DITA Implementation

Darwin Information Typing Architecture

Scott Galaba

Agenda

- How did we get here and where are we now?
- Old Process / New Process
- What is DITA, Key Features, Specifications
- DITA Implementation
- DITA Open Tool Kit
- XMetaL - DITA Authoring Tool
- Content in the DITA world
- DITA Workflow
- Topics, Concepts, Task, and References
- DITA Maps
- Conditional Reuse

How Did We Get Here?

Problem: Given the same effort and tools, it became difficult to produce high quality documentation and in different formats (the details to be presented)

Cause: The number of products has increased from 1 product to 4 family products (the future holds more)

Solution: Switched from Word to DITA (Topic-based Authoring)

Where are we now?

For the base family, publishing 17 documents in 3 formats (html, pdf, epub) for 3 products = 153

Total content deliverables = 153

Files needed to produce the content:

- 17 Map Files
- 430 Topic Files
- 546 Graphics

About the content:

- + 36K Line Count
- + 130K Word Count
- 60% Content Reuse – User's Guides
- 80% Content Reuse – Accessory Guides

Where might we be going?

- More documents
  - Additional Products
  - API
  - App and Tech Notes
- Continue to improve process
  - Review / validation
  - Automated build
  - CCMS Component Content Management System
- Potential growth into other areas

Old Process (Using Word)

- Edit word documents, once for each product
- Generate pdf one at a time

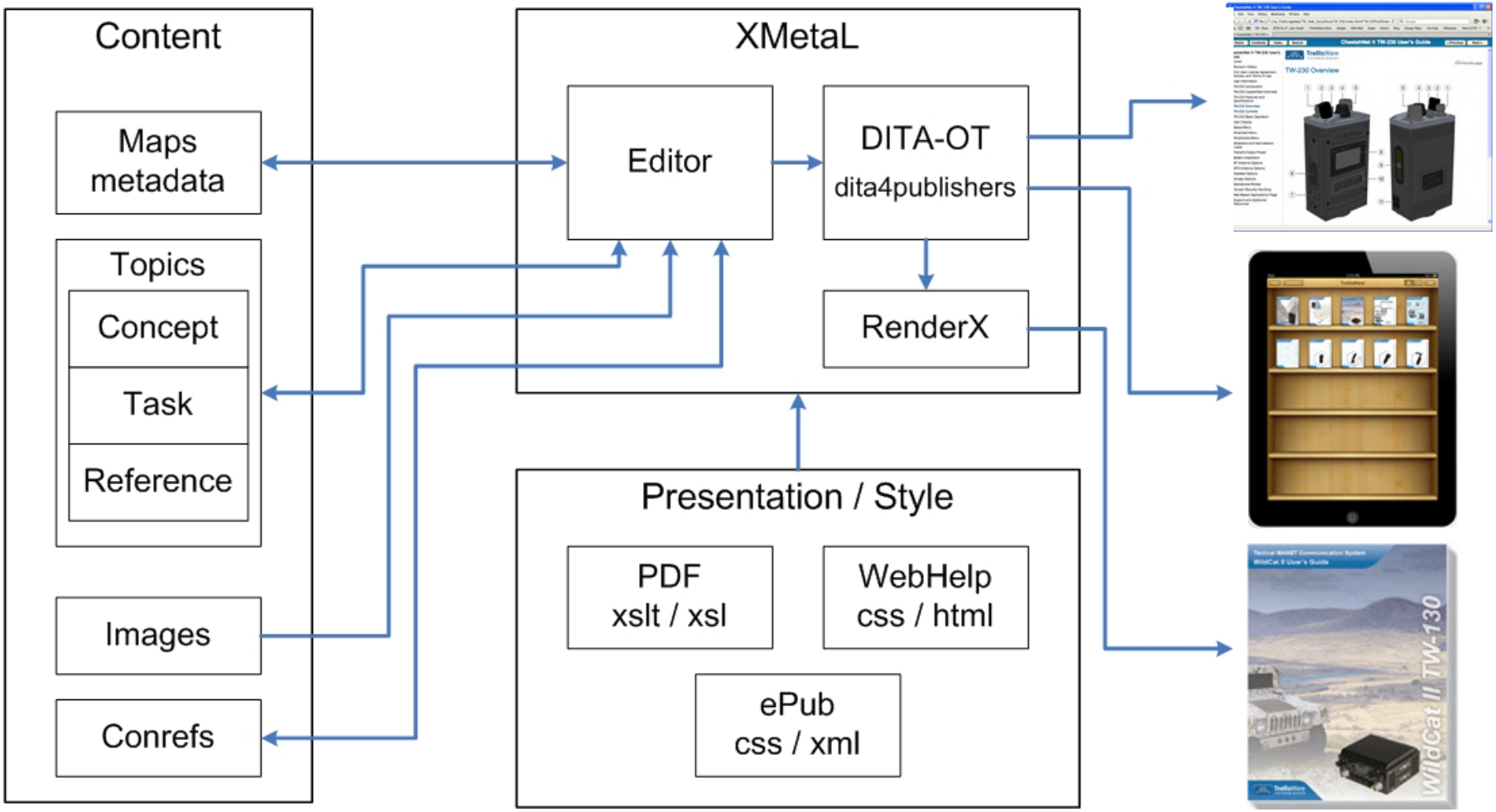
Pros	Cons
Everyone knows Word	No way to easily track changes
	No way to generate good html or epub, etc.
	Requires multiple edits of same information across documentation sets, requires cut and paste
	Time consuming process, people-intensive to maintain, open to errors
	Hard to verify changes
	Hard to work with because the style is tied to the content
	Word documents are linear: book and article style, not topic-based
	Word does not offer content reuse across documents

New Process (Using DITA)

- Documents are created and edited in the DITA format
- Generate a verify of different outputs for all products

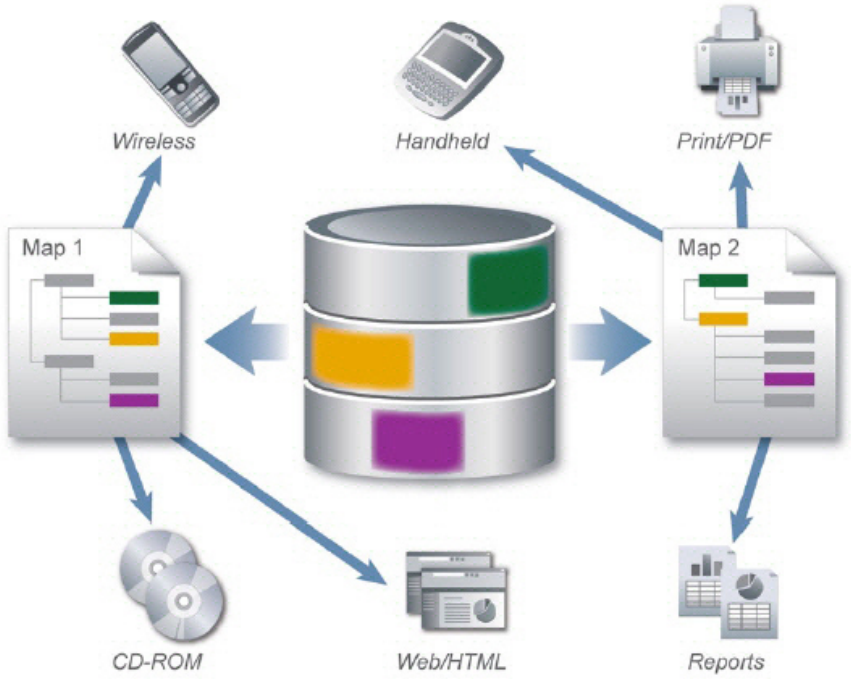
Pros	Cons
One common data content set	More complex
Write or edit it once then it reused across documents and outputs	
Many products being served from same common content	
More output types available than pdf from the same content like html and epubs	
Tracking of changes in revision control	
Separating content from format	
Provide consistent structure to the content	
Supplies a modular / topic-based content model	

DITA Implementation



The Goals

- To create content in a reusable form that is separate from presentation / style
- Provide a solution the uses single sourcing and content reuse to reduce content production and support cost, increase speed of content delivery, and improved customer product satisfaction
- To deliver content that is available anytime, anywhere, on any device



You Already Know DITA

Microsoft Word	HTML	DITA
	<html>	<topic> <concept><task><reference>
	<body>	<body> <conbody><taskbody><refbody>
Heading 1	<h1>	<title>
Paragraph	<p>	<p>
Image	<image>	<image>
Table	<table>	<table>

What is DITA?

- **DITA = Darwin Information Typing Architecture**

- **Darwin:** DITA utilizes principles of inheritance for specialization, new elements can evolve from existing elements

- **Information Typing:** Focuses on the meaning and purpose of the content (Classifying content as either a Concept, Task, or Reference topic)

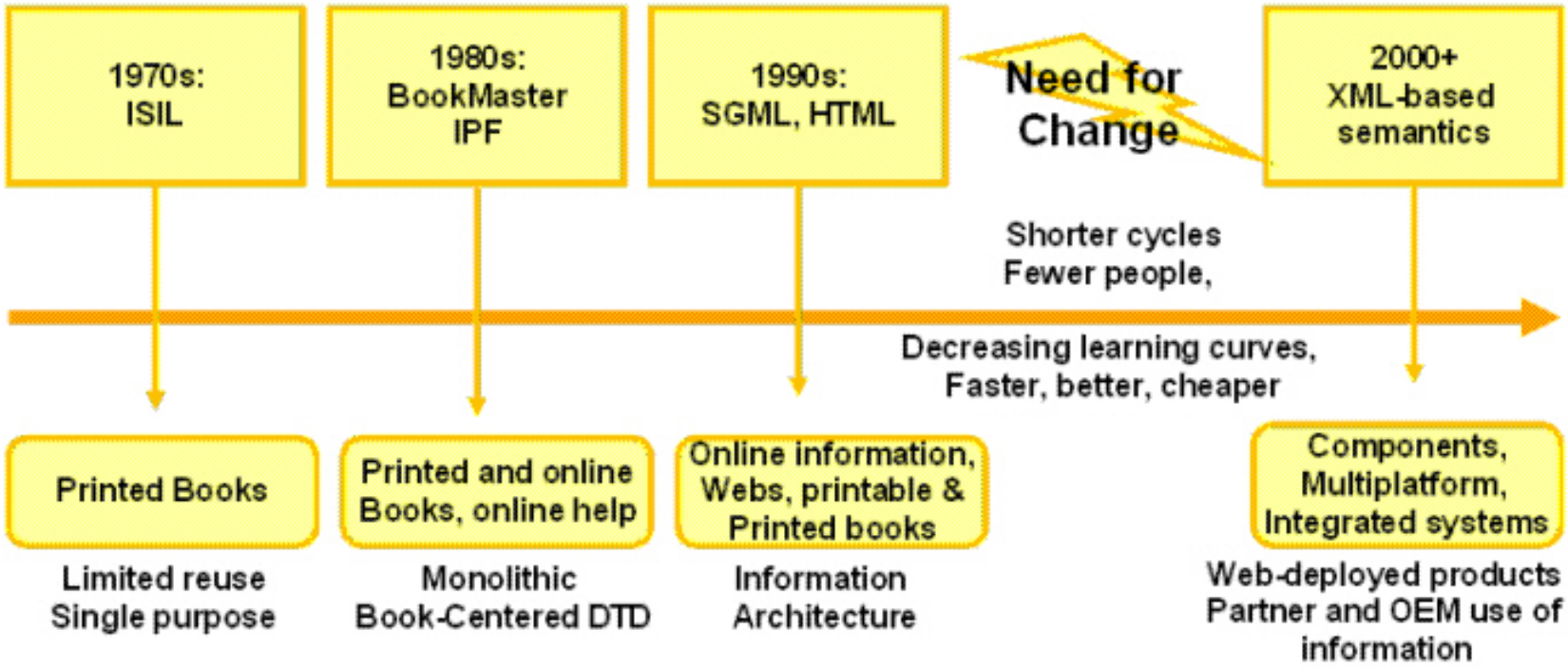
- **Architecture:** An architecture that supports the authoring, managing, and publishing topic-oriented content

Darwin Information Typing Architecture (DITA) is an XML-based, end-to-end architecture for authoring, producing, and delivering readable information as discrete, typed topics

What is DITA, really?

- An eXtensible Markup Language (XML) standard
- An XML vocabulary to describe content
- DITA components:
  - Document Type Definitions (DTDs for Topic and Map)
  - DITA Open Toolkit for generating output
  - Documentation
- Built for topic-oriented authoring
- An accepted industry standard for technical communications
- A way to work in XML without having to design your own structure
- Cost-effective way to create, publish, reuse, and exchange structured content
- Separates *content* from *presentation*
- All DITA topics are defined in terms of schemas and documents are validated against them as part of the authoring or processing process.

Markup History



DITA History

- Developed by IBM corporation as a successor/replacement for IBMIDDoc (a "book-centric" information model)
- Donated by IBM to OASIS (Organization for the Advancement of Structured Information Standards)
- Initial DITA Technical Committee:
  - XML tool vendors (Arbortext, Blast Radius, Idiom, Rascal, Syntext)
  - Consultants (Comtech, Innodata, Mulberrytech)
  - Companies (BMC, Boeing, IBM, Intel, Lucent, Nokia, Sun)
  - Organizations (National Library of Medicine, US Department of Defense)
- DITA 1.0 finalized by DITA Technical Committee February 2005
- DITA 1.0 formally approved by OASIS June 2005
- DITA 1.1 formally approved August 2007
- DITA 1.2 formally approved December 2010 with new Learning and Training Content (L&TC) Specialization
- In the current DITA 1.2 standard there are over 520 elements defined
  - Basic = 171, TC = 215, L&TC = 142

Core Design Principles of DITA

Core Principles	Description
Topic Orientation	Discrete units of information covering a specific subject with a specific intent
Topic Granularity	Topics combine with other topics into information sets
Consistency	DITA DTDs (Document Type Definition) guarantee (validated) that DITA information types follow identical information structures
Separation of Content	Separation of content (specific topics) from context (links to other topics, files, navigation) Not just separation of content from formatting!!
Inheritance	Has a top-level “generic” information type from which other types inherit their structures
Specialization	Ability to extend basic information types for special uses

How do we ensure consistency of information and formatting across many authoring Groups ?

DITA has rules!

Rules for authoring and rules to transform / creating outputs

The rules are set by Document Type Definition (DTD):

- Element definitions
- Required or not?
- Number of elements allowed and whether ordering rules apply
- What the elements can contain
  - Other elements
  - Text
  - Attributes
  - Predefined attribute values

DITA Key Features

- **Modular Content** – "topic-based authoring"
  - Content organized into small, reusable modules called topics
  - Topics organized using maps for publication and delivery
  - Clearly distinct information types: concept, task, and reference
- **Maps** – a collection of topics
- **Domains** – a set of elements available across multiple topic or map types
  - Programming, Software, User Interface
- **Metadata** – available to both topics and maps used for describing and passing content about the topic or map
- **Specialization** – allows for new element types to be created based on core DITA elements
- **Conrefs** – content references, link-based reference to content
- **Conditional Reuse** – content flagged for at the Audience, Platform and Product level

DITA Open Tool Kit

General-purpose, cross-platform, open-source publishing system

http://dita-ot.sourceforge.net/

- Maintained by IBM and DITA technical community through SourceForge
- The DITA Open toolkit is XML with a starter set of stylesheets (XSLs) and schemas (DTDs)
- It is free and has been integrated into leading DITA editors and CMS tools
- Uses Apache Ant to transform DITA files into different outputs
  - (ant -f build.xml mapfile transformtype)
- Provides plug-in architecture for adding functionality
  - DITA For Publishers, used to produce epub
- Out-of-the-box, transforms, produces:
  - HTML
  - PDF (via XSL-FO output compiler to pdf, like RenderX)
  - WebHelp

DITA and DITA-OT Versions

Current versions:

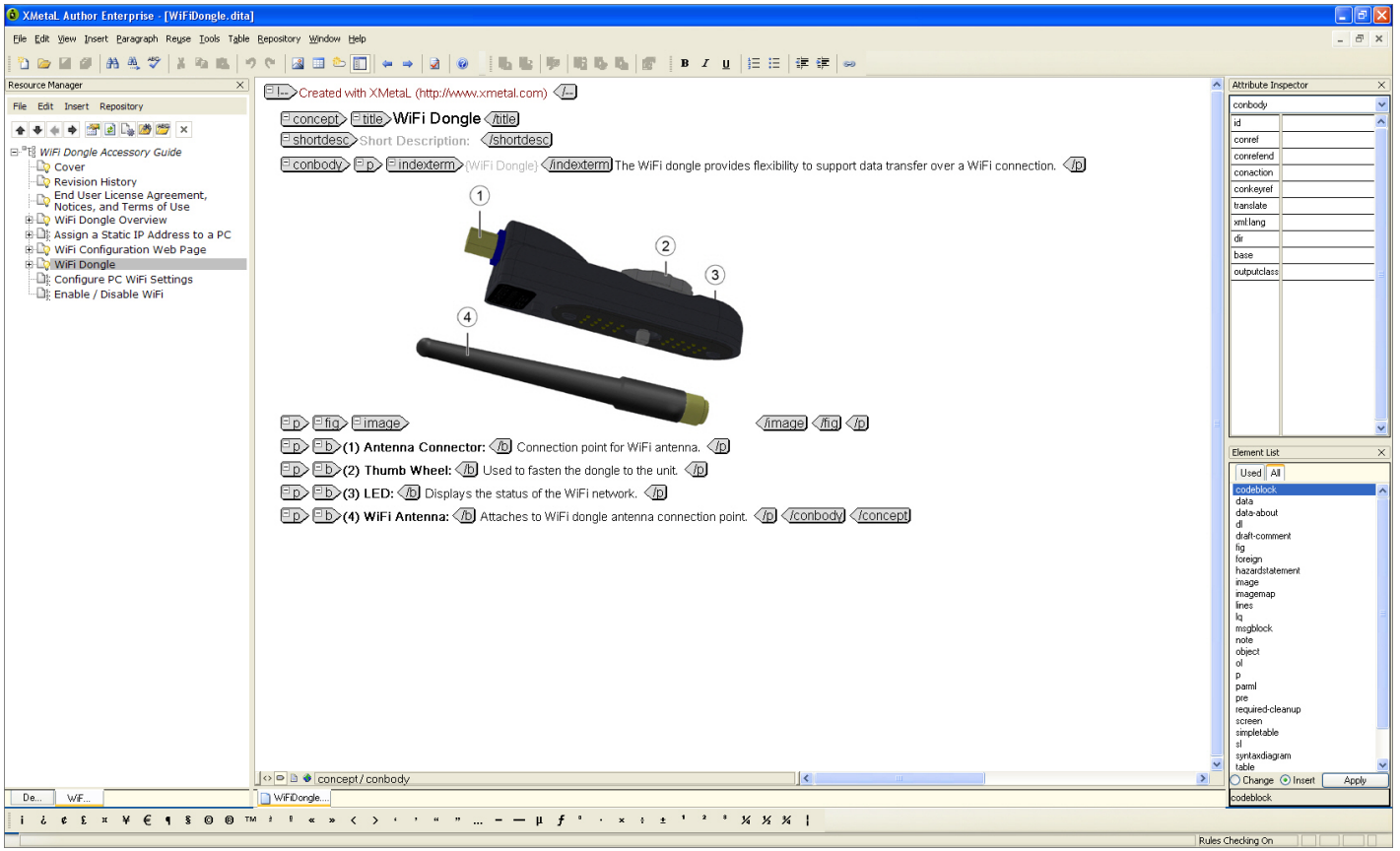
- DITA Standard: 1.2
- DITA Open Tool Kit: 1.5.4

DITA Versions	DITA-OT Versions
1.0	1.0.X
	1.1.X
	1.2.X
1.1 (preliminary)	1.3
	1.3.1
1.1	1.4.X
	1.4.3
1.2	1.5.2
	1.5.3
	1.5.4

DITA Authoring Tool – XMetaL Author Enterprise

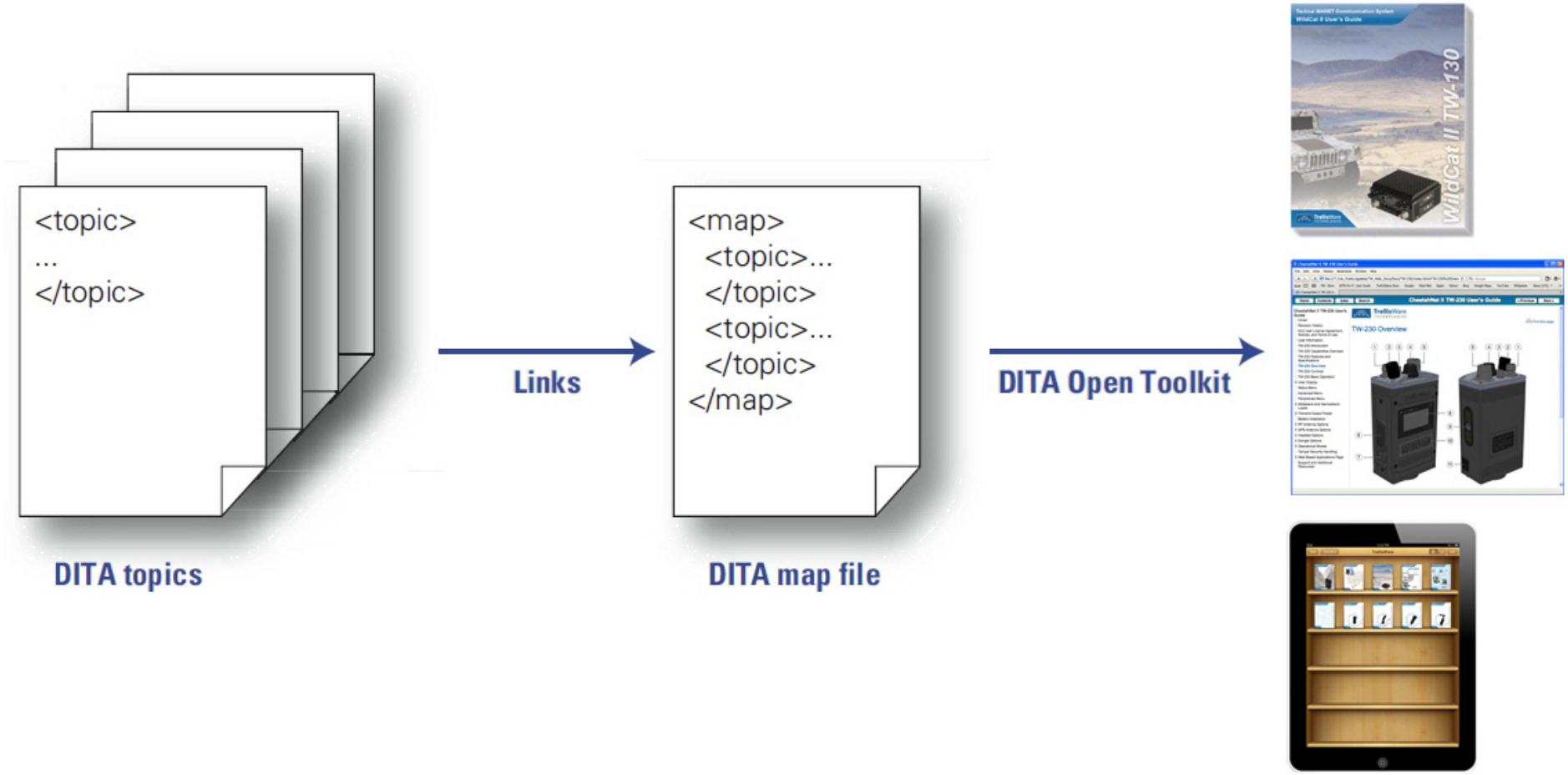
For creating and editing DITA topics and maps, and for publishing

- DITA-aware
- DITA rules are embedded
- DITA 1.2 Elements
- Validates topics and maps to DITA 1.2
- Creates well-formed structure
- Text, Tag, Normal editing views
- DITA-OT 1.5.4
- RenderX for PDF
- Publishing Interface to 21 deliverables



DITA Single Sourcing Publishing Overview

One map used to create multiple outputs

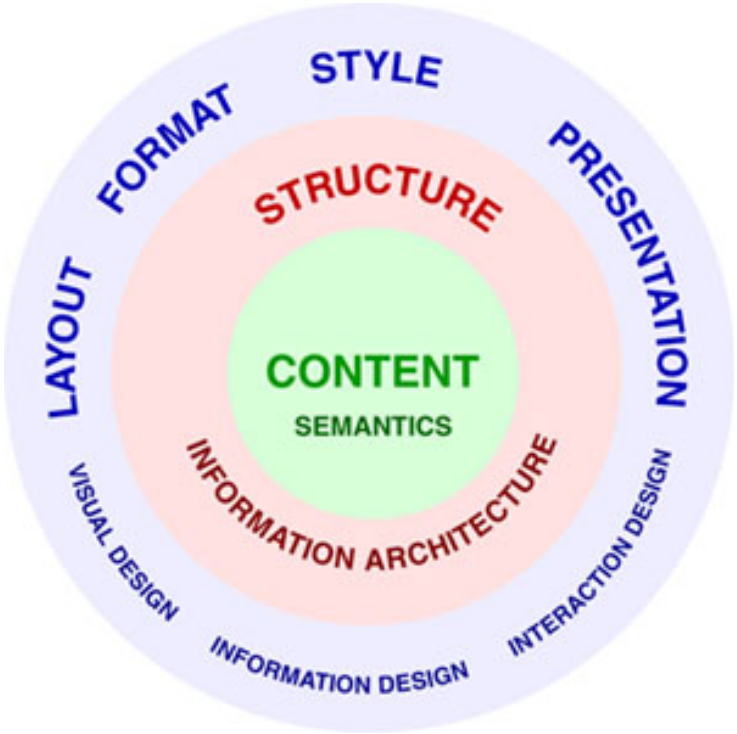


What is Content in the DITA World?

- It's not the Presentation or the Structure!
- DITA Separates the Presentation Layer from Content
  - Adds Structure to the Content
  - Tags Content with Meaning (semantics) by Metadata

- The three layers use different "markup"
  - Style – color, font-size, margin-left
  - Structure - <p>, <image>, <table>
  - Semantics <menucascade>, <uicontrol>

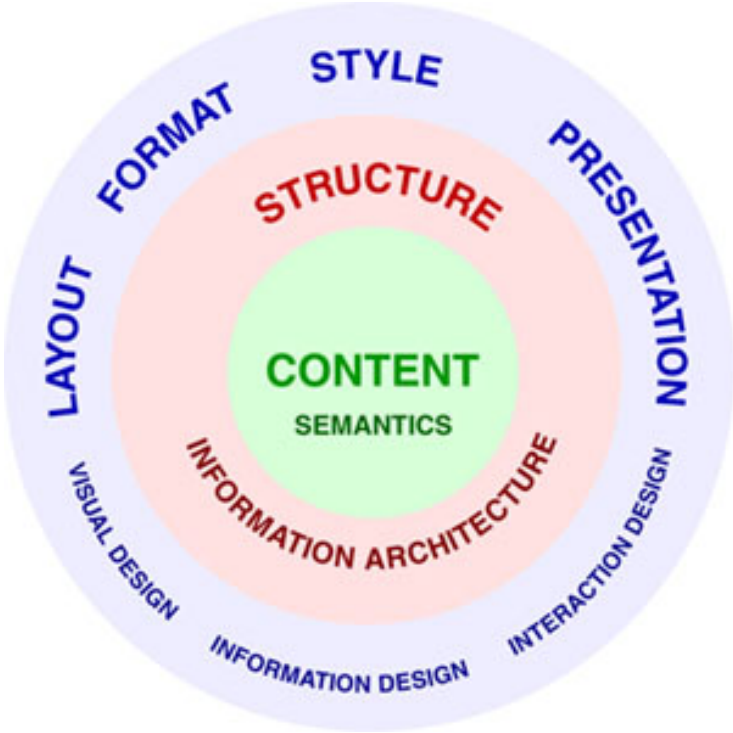
- The three layers use different technologies
  - XSLT Stylesheets (CSS)
  - XML Schemas (DTDs)
  - XML/DITA Documents



Three Different Professions

The three layers are the work of different professionals

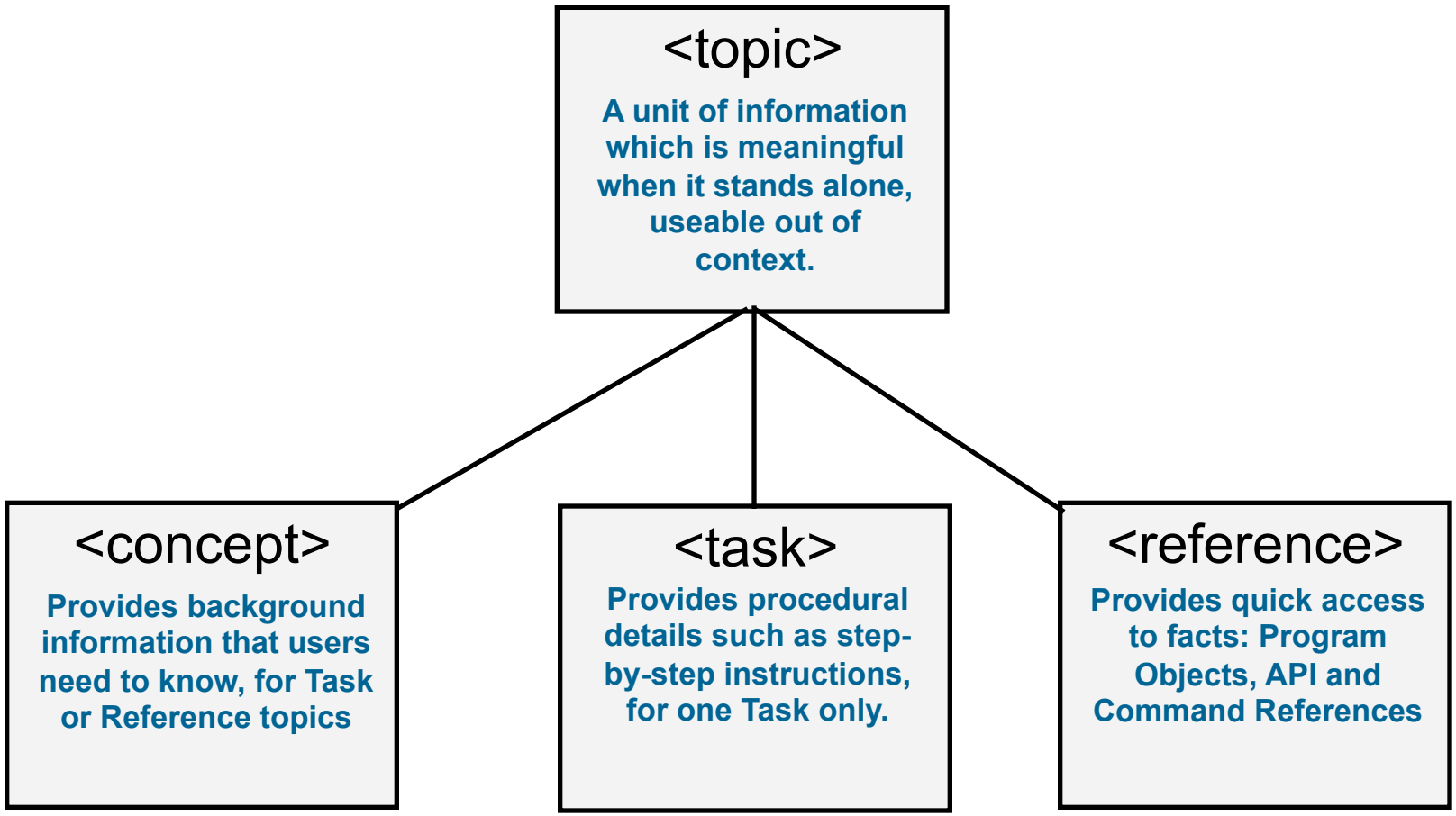
- Designers (Information, Interaction, Visual) for Style
- Information Architects for Structure
- Authors for Content and Metadata



DITA Topic

- A chunk of information specific to a single subject
- Short enough to be specific to a single subject or answer a single question
- Should be able to stand-alone, usable out of context
- Generic topic is the bases for all specialized topic types (Darwin inheritance):
  - From the “Generic topic" type created: task, concept, reference topics
- DITA prescribes three information types:
  - Concept
  - Task
  - Reference
- No formal restriction on topic length
- Topic file type can be either .xml or .dita extension
  - .dita extension used at TrellisWare

DITA Core Topic Types – The “IT” in DITA



DITA Core Topic Types - continued

Concept Topics

- Concept topics introduce the background or overview information for task or reference topics
- Concept topics should not describe task or reference information

Task Topics

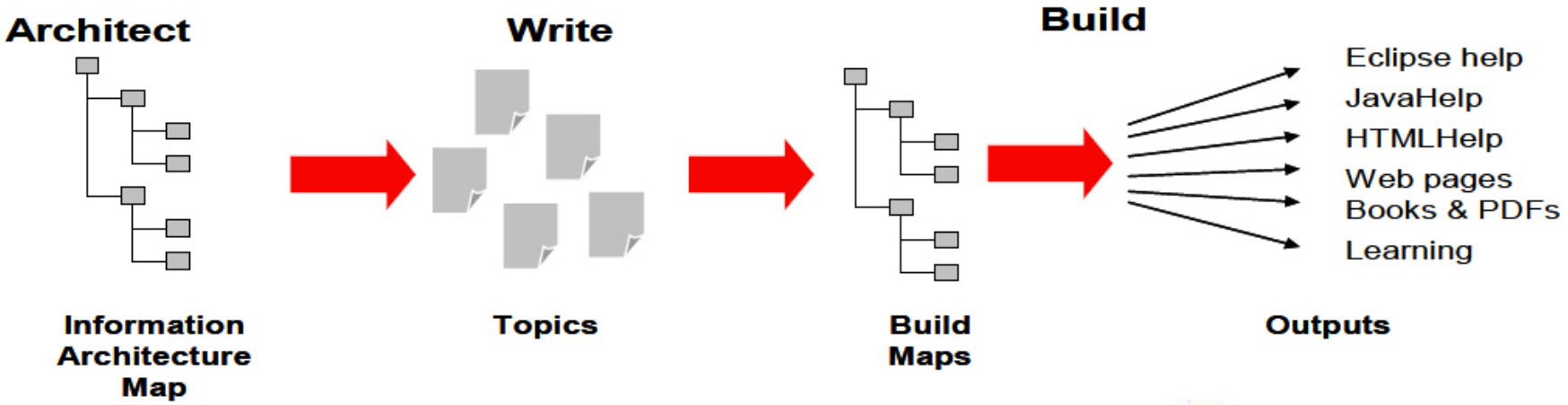
- Task topics describe the steps of a particular task, or provide an overview of a higher-level task
- In a task topic, describe how to do only one task
- Task topics should not describe conceptual or reference information

Reference Topics

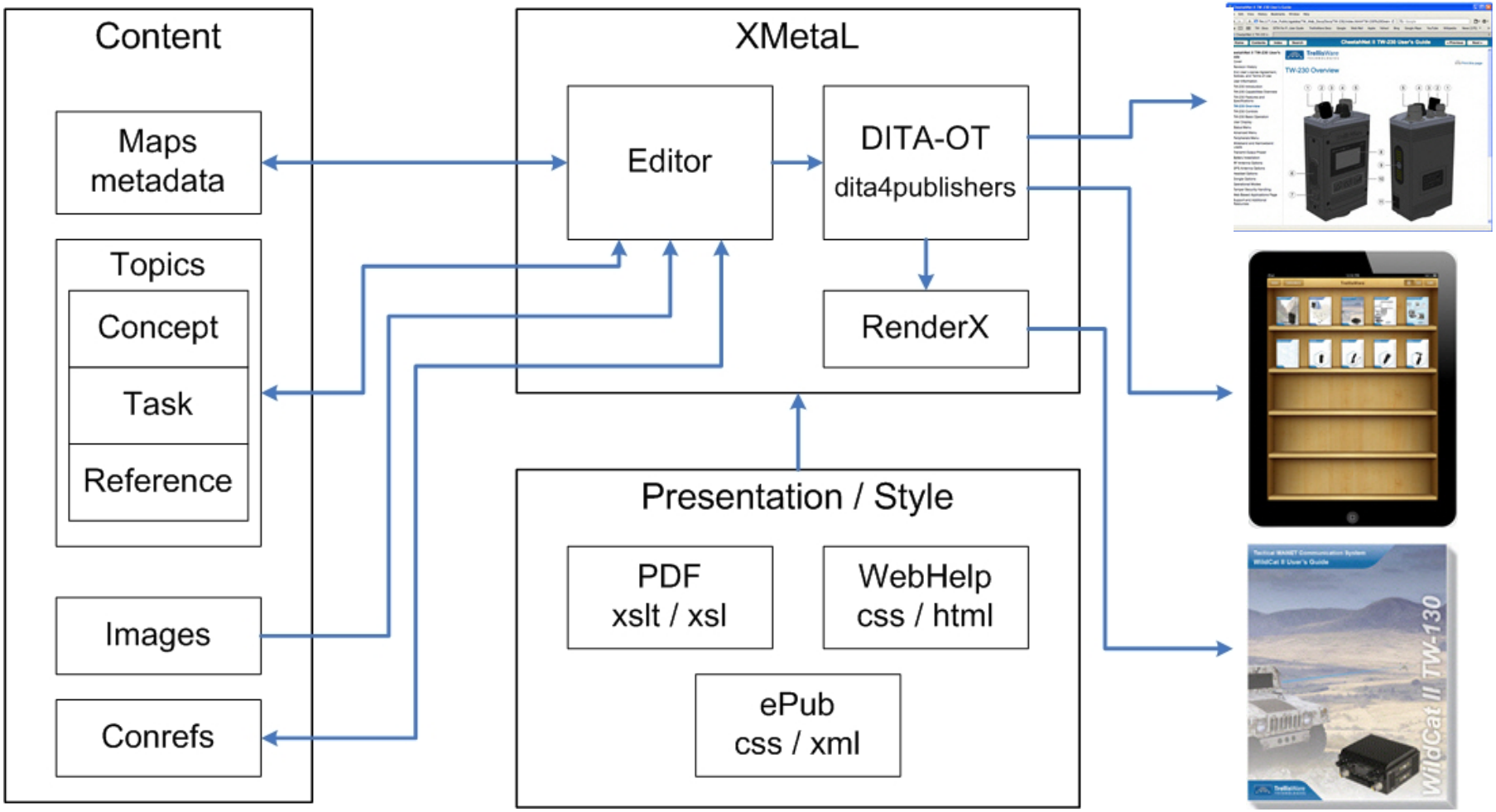
- Reference topics should be designed for quick scanning of information
- Typical uses: Messages and codes, Part descriptions, Program objects, APIs, Command reference
- Reference topics should not describe conceptual or task information

Typical DITA Workflow

1. Analyze the content, procedure, or information as it applies to the DITA information types.
2. Identify the tasks.
3. Identify the concepts and references needed to support the task.
4. Create the topics (tasks, concepts, references).
5. Use DITA maps to assemble topics for document deliverable.
6. Publish and deliver the content.



DITA Implementation

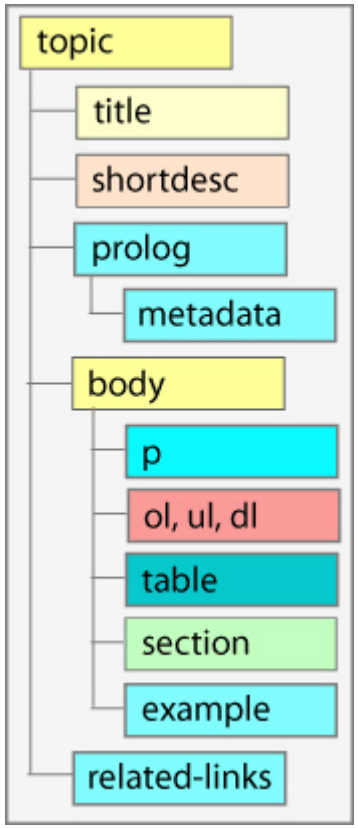


DITA Topic Structure

A topic has only three required elements:

- An ***id*** attribute in the main topic tag
- A ***title***
- A ***body***

```
<topic id="1">
  <title>My Topic</tilte>
  <shortdesc>About my topic…</shortdesc>
  <body>
   <p>Some content</p>
   <p>Some more content</p>
  </body>
</topic>
```

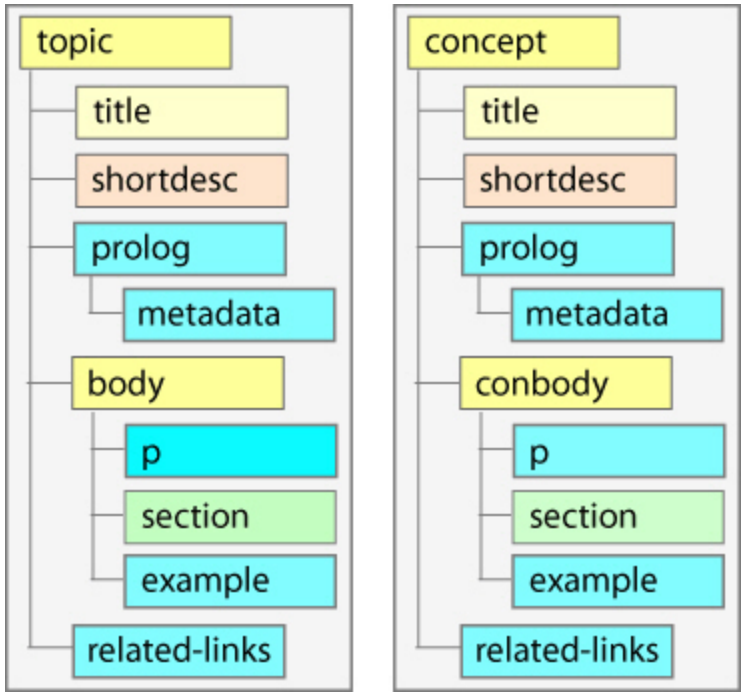


DITA Concept Topic Structure

Specializes topic element names and topic structure:

- Root element is renamed ***concept***
- Body element is renamed ***conbody***

```
<concept id="2">
  <title>My Concept topic <title/>
  <shortdesc>About my..</shortdesc>
  <conbody>
   <p>Some content</p>
   <p>Some more content</p>
   </conbody>
</concept>
```



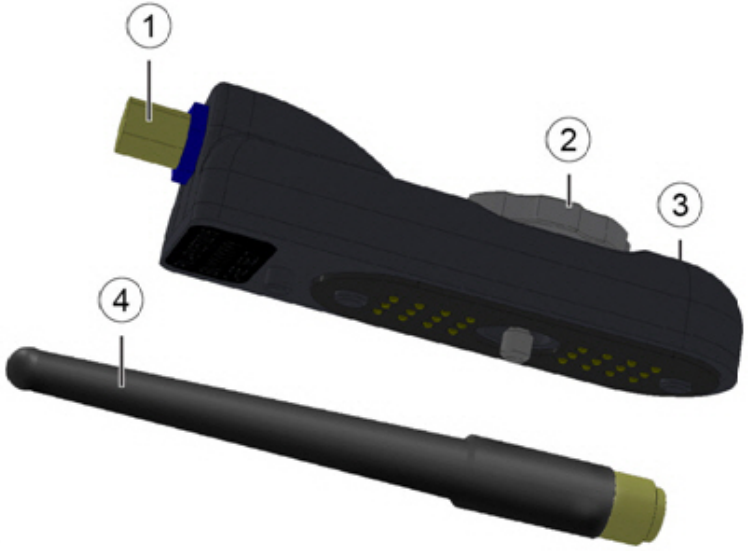
Concept Topic Tag Example

<!-- Created with XMetaL (http://www.xmetal.com) -->

<concept> <title>WiFi Dongle</title>

<shortdesc>Short Description:</shortdesc>

<conbody> <indexterm>{WiFi Dongle}</indexterm> The WiFi dongle provides flexibility to support data transfer over a WiFi connection.



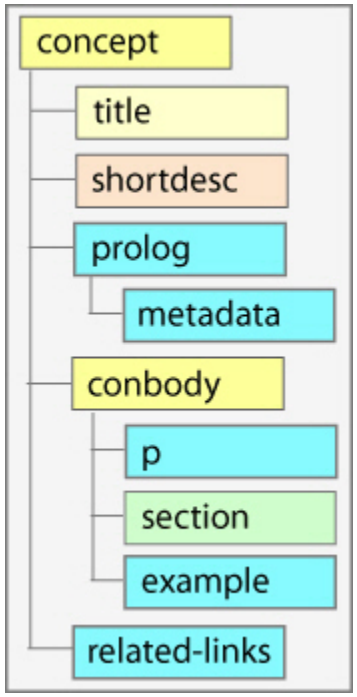
</image> </fig>

 (1) Antenna Connector: Connection point for WiFi antenna.

 (2) Thumb Wheel: Used to fasten the dongle to the unit.

 (3) LED: Displays the status of the WiFi network.

 (4) WiFi Antenna: Attaches to WiFi dongle antenna connection point. </conbody> </concept>



Concept Topic Code Example

```
 1: <?xml version="1.0"?>
 2: <!DOCTYPE concept PUBLIC "-//OASIS//DTD DITA Concept//EN" "concept.dtd">
 3: <!-- Created with XMetaL (http://www.xmetal.com) -->
 4: <concept id="concept_5BD24B8C90124F50A2836863F2EF7B9F" outputclass="pagebreak">
 5:
 6:   <title>WiFi Dongle</title>
 7:   <shortdesc>
 8:   </shortdesc>
 9:   <conbody>
10:     <p><indexterm>WiFi Dongle</indexterm>The WiFi dongle provides flexibility
11:       to support data transfer over a WiFi connection.
12:     </p>
13:     <p>
14:       <fig id="fig_F93B86989CF845CAA0474FEA7241A5CE"><image placement="inline"
15:         href="Images/Dongles/WiFi.jpg" width="415px" height="305px"></image>
16:       </fig>
17:     </p>
18:     <p>
19:       <b>(1) Antenna Connector:</b> Connection point for WiFi antenna.
20:     </p>
21:     <p>
22:       <b>(2) Thumb Wheel:</b> Used to fasten the dongle to the unit.
23:     </p>
24:     <p>
25:       <b>(3) LED:</b> Displays the status of the WiFi network.
26:     </p>
27:     <p>
28:       <b>(4) WiFi Antenna:</b> Attaches to WiFi dongle antenna connection
29:       point.
30:     </p>
31:   </conbody>
32: </concept>
```

DITA Task Topic Structure

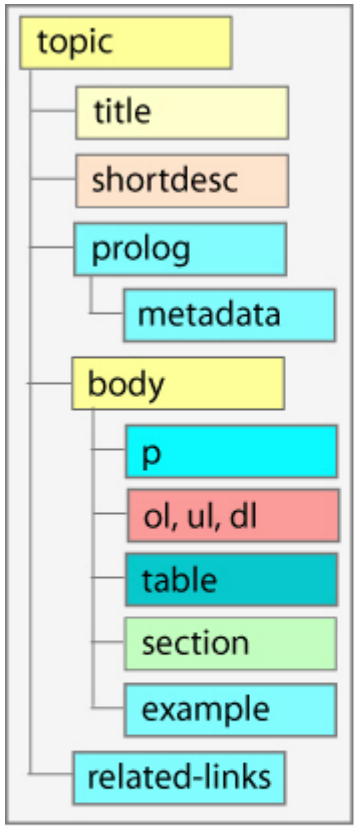
Specializes topic element names and topic structure:

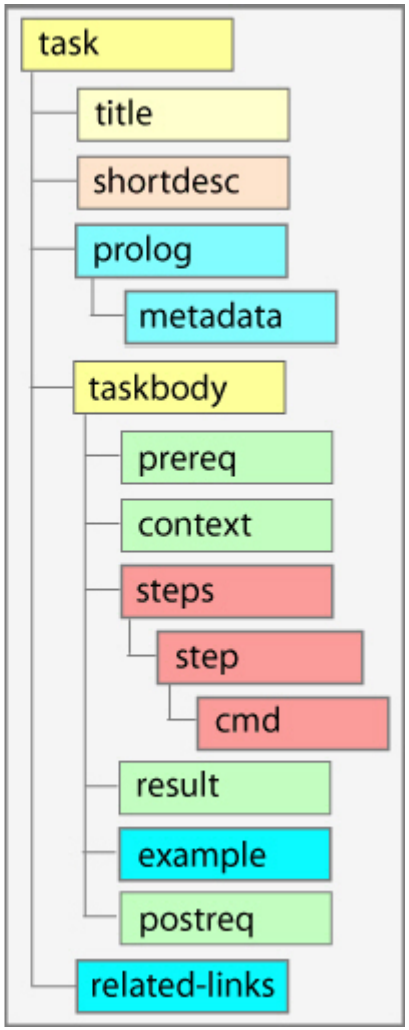
- Root element is renamed ***task***
- Body element is renamed ***taskbody***

```
<task id="3">
 <title>My Concept topic <title/>
  <shortdesc>About my concept..</shortdesc>
  <taskbody>
   <prereq/>
   <context/>
   <steps/>
  </taskbody>
</task>
```

DITA Task Topic Structure - continued

```
<taskbody>
  <prereq/>
  <context/>
  <steps>
    <step><cmd> </cmd></step>
    <step><cmd> </cmd></step>
  </steps>
  <result/>
  <example/>
  <postreq/>
</taskbody>
```





Task Topic Tag Example

<!-- Created with XMetaL (http://www.xmetal.com) -->

<task> <title>Installation of WiFi Dongle</title>

<shortdesc>Short Description:</shortdesc>

<taskbody> <context> <indexterm>{WiFi Dongle <indexterm>{Installation}</indexterm>}</indexterm>

To install the WiFi dongle: </context>

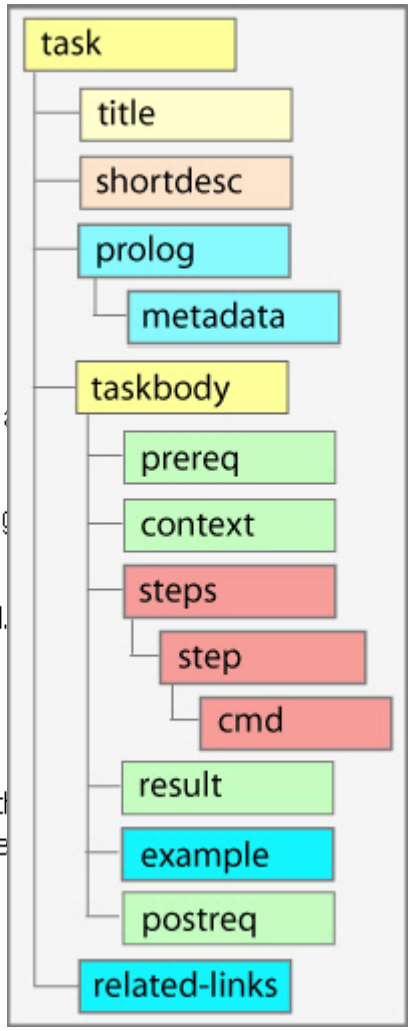
<steps>

1. <step> <cmd>Turn the unit OFF.</cmd> </step>
2. <step> <cmd>Attach the WiFi antenna to the dongle by inserting the antenna onto the WiFi antenna connector and screw the
   </cmd> </step>
3. <step> <cmd>Align the dongle center to the guide pins and use the thumbscrew to secure the dongle to the unit Align the dong
   the thumbscrew to secure the dongle to the unit. </cmd> </step>
4. <step> <cmd>Once the dongle is firmly in place, turn the unit on and it will automatically recognize that the dongle is attached.

<result>Result:

<note>**Note:** The unit will not recognize the dongle if it is attached when the unit is already in operation. The unit's power must be
the dongle. </note>

<note>**Caution:** When the WiFi dongle is attached, power is being applied to the side connector. Care should be taken not to let t
configuration to avoid shorting the unit. It should also be noted that while the unit is immersible, the WiFi dongle is not. Immersion or e
dongle is attached can permanently damage both the dongle and unit. </note> </result> </taskbody> </task>



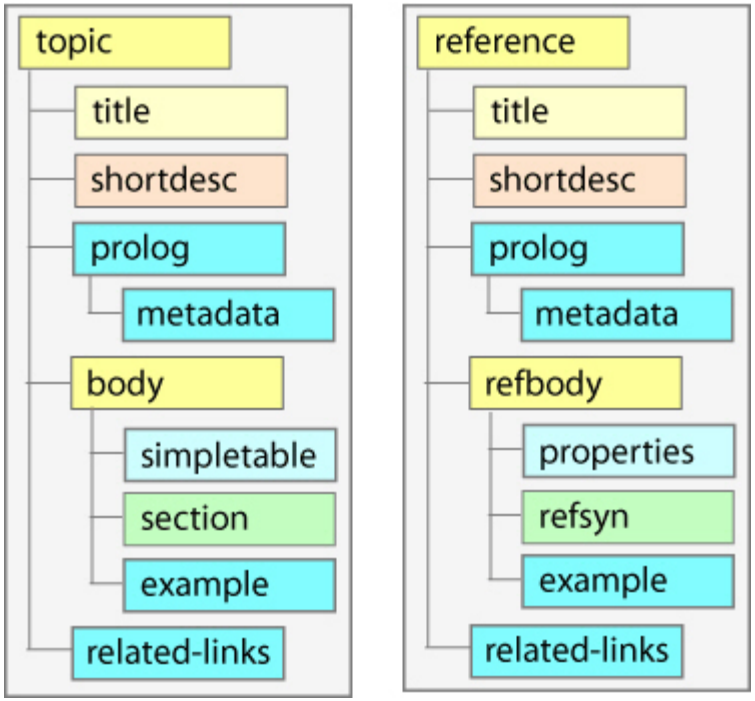
Task Topic Code Example

```
<?xml version="1.0"?>
<!DOCTYPE task PUBLIC "-//OASIS//DTD DITA Task//EN" "task.dtd">
<!-- Created with XMetaL (http://www.xmetal.com) -->
<task id="task_7CBBAA791EB849ECA9DAD6C9B90CEED0">
  <title>Installation of WiFi Dongle</title>
  <shortdesc>
  </shortdesc>
  <taskbody>
    <context><indexterm>WiFi
        Dongle<indexterm>Installation</indexterm></indexterm>
      <p> To install the WiFi dongle:
      </p>
    </context>
    <steps>
      <step>
        <cmd> Turn the unit OFF.
        </cmd>
      </step>
      <step>
        <cmd> Attach the WiFi antenna to the dongle by inserting the antenna
          onto the WiFi antenna connector and screw the antenna until it is firmly
          seated.
        </cmd>
      </step>
      <step>
        <cmd> Align the dongle center to the guide pins and use the thumbscrew
          to secure the dongle to the unit Align the dongle center to the guide pins and
          use the thumbscrew to secure the dongle to the unit.
        </cmd>
      </step>
      <step>
        <cmd> Once the dongle is firmly in place, turn the unit on and it will
          automatically recognize that the dongle is attached.
        </cmd>
      </step>
    </steps>
    <result>
      <note> The unit will not recognize the dongle if it is attached when the
        unit is already in operation. The unit's power must be turned off then back on
        to recognize the dongle.
      </note>
      <note type="caution"> When the WiFi dongle is attached, power is being
        applied to the side connector. Care should be taken not to let the unit get wet
        when operating in this configuration to avoid shorting the unit. It should also
        be noted that while the unit is immersible, the WiFi dongle is not. Immersion
        or excessive moisture while the WiFi dongle is attached can permanently damage
        both the dongle and unit.
      </note>
    </result>
  </taskbody>
</task>
```

DITA Reference Topic Structure

Specializes topic element names and topic structure:

- Root element is renamed ***reference***
- Body element is renamed ***refbody***




```
<reference id="4">
  <title>My Reference Topic <title/>
  <shortdesc>About my..</shortdesc>
  <refbody>
   <properties/>
   <refsyn/>
  </refbody>
</reference>
```

- properties = three-column table of property types
- refsyn = specialization of the section element

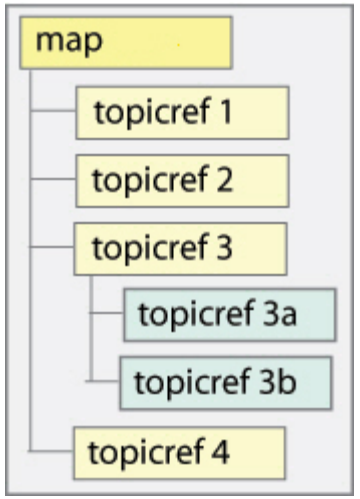
DITA Map

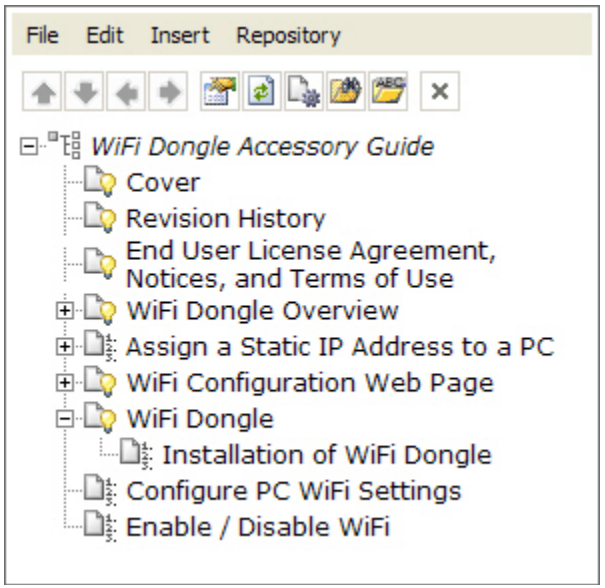
- A DITA map organizes ***topicrefs*** (references to DITA topics) in hierarchies for publishing to web, print, help, epubs, and other deliverables
- DITA maps provide a means to arrange topics
- They function as outlines or table of contents for deliverables
- Defines the navigation structure
- Allows topics to be organizes for different deliverables
- Can reuse the same topic in different deliverables
- DITA maps can contain metadata to added meaning to the content
- DITA maps have .ditamap file extension

DITA Map File Structure

A map is a nested list of ***topicrefs***

```
<map id="sample" title = “sample map">
  <topicref href="topic1.dita" />
  <topicref href="topic2.dita" />
  <topicref href="topic3.dita">
      <topicref href="topic3a.dita" />
      <topicref href="topic3b.dita" />
  </topicref>
  <topicref href="topic4.dita"  />
</map>
```





DITA Map Tag Example

DITA Map Tag Example

Metadata

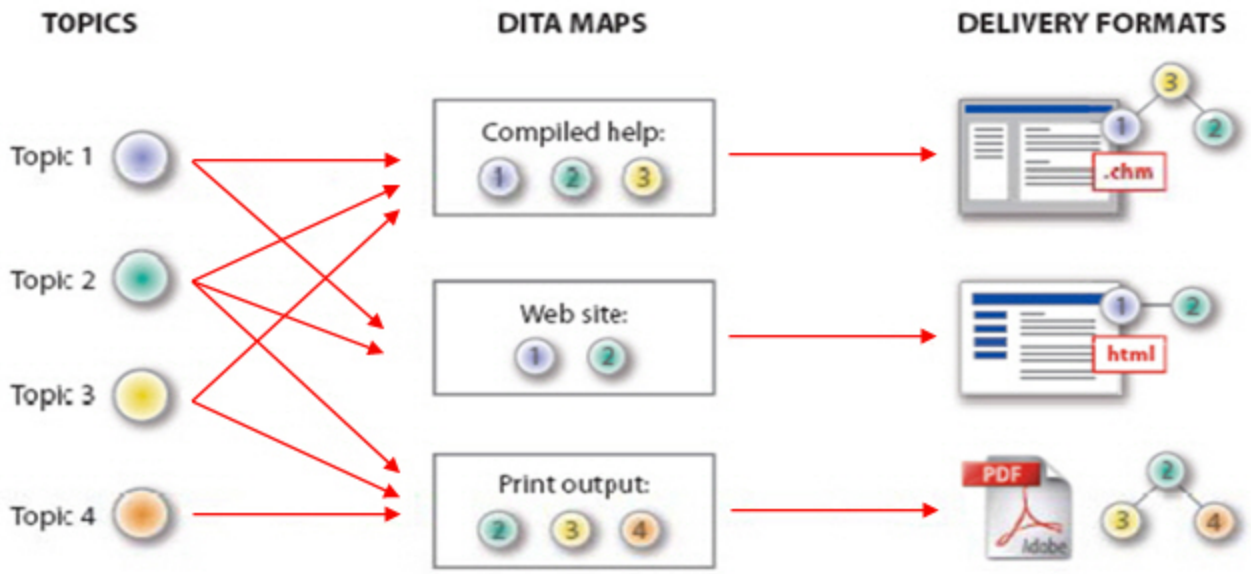
Used on the title page and placed into the header and footer of the PDF

Also placed into the metadata area of the html and epub source files

Concept and Task topics used in this presentation

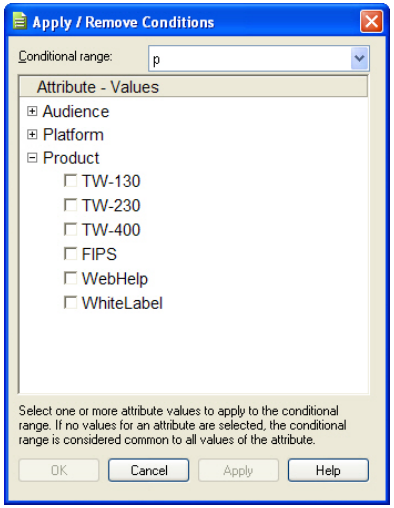
DITA Maps – Delivering Different Formats

Separate map used for a specific output



Conditional Reuse

- Applied to elements or content
- Used to target content at the Audience, Platform and Product level
  - Audience
    - Administrator
    - User
  - Platform
    - Windows XP
    - Windows 2000
    - Linux
    - Mac OSX
  - Product
    - TW-130
    - TW-230
    - TW-400
    - FIPS
    - WebHelp
    - WhiteLabel



Concept and Task Topic PDF Output Example

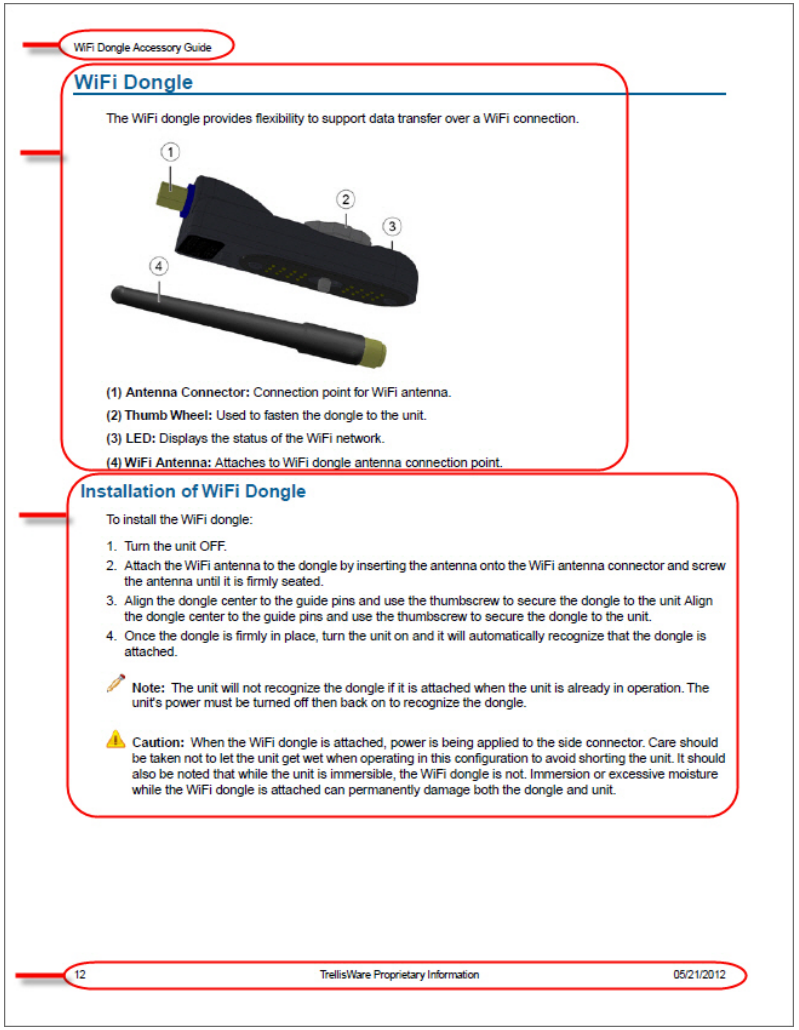
Content

Metadata - Header

Concept Topic

Task Topic

Metadata - Footer



Presentation & Style

XSLT and XSL files:

- Set page parameters and variables
  - size & layout
  - 8.5 x 11, margins
- Set font size, color and position on document
- Set header and footer
  - Font size, color and position on document
- Place metadata into the document header, footer, and title page
- etc…

Concept Topic html WebHelp Output Example

Content

DITA Map

Concept Topic

Presentation & Style

HTML files free-flow

CSS file:

- Set font size, color and position on document
- Set header and footer font size and color
- Header and footer content added as plug-in html files during transform build process
- HTML and java files control the interface
- etc…

DITA Specification

http://docs.oasis-open.org/dita/v1.2/os/spec/DITA1.2-spec.html

Each element:

- Description
- Contains
- Contained by
- Inheritance
- Example
- Attributes

3.1.1.4.4 linktext

The <*linktext*> element provides the literal label or line of text for a link. In most cases, the text of a link can be resolved during processing by cross reference with the target resource. Use the <*linktext*> element only when the target cannot be reached, such as when it is a peer or external link, or when the target is local but not in DITA format. When used inside a topic, it is used as the text for the specified link; when used within a map, it is used as the text for generated links that point to the specified topic.

Contains

Note: These models represent only the default document types distributed by OASIS. Actual content models will differ with each new document type.

Doctype	Content model
topic (base), map (base), classifyMap, subjectScheme, learningAssessment, learningBookmap, learningContent, learningMap, learningOverview, learningPlan, learningSummary	(text data or data or data-about or foreign or unknown or keyword or term or ph or b or i or sup or sub or tt or u) *(any number)*
topic (technical content), map (technical content), concept, ditabase, glossary, glossentry, glossgroup, reference, task, bookmap	(text data or data or data-about or foreign or unknown or keyword or apiname or option or parmname or cmdname or msgnum or varname or wintitle or term or abbreviated-form or ph or b or i or sup or sub or tt or u or codeph or synph or filepath or msgph or systemoutput or userinput or menucascade or uicontrol) *(any number)*
machineryTask	(text data or data or data-about or foreign or unknown or keyword or wintitle or term or ph or b or i or sup or sub or tt or u or menucascade or uicontrol) *(any number)*

Contained by

Doctype	Content model
topic (base), topic (technical content), concept, ditabase, glossary, glossentry, glossgroup, reference, task (strict), task (general), machineryTask, learningAssessment, learningContent, learningOverview, learningPlan, learningSummary	link
map (base), map (technical content), classifyMap, subjectScheme, learningMap	topicmeta
bookmap, learningBookmap	topicmeta, bookmeta

Inheritance

"- topic/linktext " when used in topics, and "- map/linktext " when used in maps.

Example

```
<link href="tzover.htm#accsqlj">
 <linktext>Accessing relational data with SQLJ</linktext>
</link>
```

Attributes

Name	Description	Data Type	Default Value	Required?
univ-atts attribute group (includes select-atts, id-atts, and localization-atts groups)	A set of related attributes, described in univ-atts attribute group			
global-atts attribute group (xtrf, xtrc)	A set of related attributes, described in global-atts attribute group			
class, outputclass	Common attributes described in Other common DITA attributes			

Parent topic: 3.1.1.4 Related links elements
Previous topic: 3.1.1.4.3 linkpool
Next topic: 3.1.1.4.5 linkinfo

Return to main page.

OASIS DITA Version 1.2 -- OASIS Standard, 1 December 2010

Reference Material

OASIS DITA Technical Committee (TC)

http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=dita

DITA 1.2 Standard

http://docs.oasis-open.org/dita/v1.2/os/spec/DITA1.2-spec.html

All DITA 1.2 elements, A to Z

http://docs.oasis-open.org/dita/v1.2/os/spec/common/alldita1.2elements_a_to_z.html

DITA Open Toolkit

http://dita-ot.sourceforge.net/

IBM - Introduction to the Darwin Information Typing Architecture

https://www.ibm.com/developerworks/xml/library/x-dita1/

https://www.ibm.com/developerworks/xml/library/x-dita2/

https://www.ibm.com/developerworks/xml/library/x-dita3/

XMetaL DITA Authoring Tools

http://na.justsystems.com

Presentation Credits

- A Short Introduction to DITA – Michael Priestly – IBM
- DITA 101 – Sarah O'Keefe – Scriptorium Publishing
- DITA and Information Architecture – Kristen Eberlein – IBM
- DITA Maturity Model – Michael Priestly, IBM and Amber Swope, JustSytems
- DITA Quick Start – Eliot Kimber – Really Strategies
- DITA Quick Start – T.S. Selvakumar – Cadence
- DITA XML and SCORM-based Learning Delivery – John Hut – IBM
- Introduction to Information Modeling with DITA – Alan Houser – Group Wellesley
- Painless XML Authoring? How DITA Simplifies XML – Bob Doyle – CMSReview
- Using DITA for Training & Support – Joan Lasselle – Lasselle-Ramsay