



Open Publish 2006

An introduction to DITA

27 July 2006

Andrew Squire, Senior Consultant, Elkera Pty Ltd

Elkera Pty Limited ABN 68 092 447 428
Suite 701, 10 Help Street, Chatswood, NSW 2067, Australia
PO Box 5280, West Chatswood, NSW 1515
Telephone +61 2 8440 6999
Facsimile +61 2 8440 6988
www.elkera.com

Contents

1	Introduction.....	1
1.1	What is DITA?.....	1
1.2	DITA history.....	1
2	What problems does DITA try to solve?.....	1
2.1	Content reuse.....	1
2.2	Content sharing and interchange.....	2
3	How does DITA solve these problems?	3
3.1	Topic orientated architecture enables content reuse.....	3
3.2	Specialization architecture enables content interchange	4
4	Other benefits of DITA.....	4
5	Who should use DITA?	5
6	Maximising the benefits from DITA	5
6.1	Information architecture	5
6.2	Consider the writers.....	5
6.3	Off the shelf tools.....	6

An introduction to DITA

Author: Andrew Squire, Senior Consultant, Elkera Pty Limited.

Date: 27 July 2006

Presented at Open Publish 2006, Sydney, Australia on 27 July 2006, under the title "Real world application development with DITA".

1 Introduction

1.1 What is DITA?

- DITA stands for Darwin Information Typing Architecture.
- DITA is an XML based architecture for authoring and publishing technical documentation.
- DITA is a topic based architecture where content is broken up into self-contained chunks:
 - content is broken into self contained chunks
 - familiar to help and web content creators

1.2 DITA history

- Developed by IBM in 2000 to replace IBMIDDoc for their large library of technical documentation.
- IBM donated DITA to OASIS in March 2004.
- Revised by OASIS and became an OASIS standard in May 2005.

2 What problems does DITA try to solve?

2.1 Content reuse

- Some organizations maintain large document sets where multiple documents may contain copies of the same content.
- A common approach to content reuse has been to copy the content and paste it into new documents. Multiple copies of the same content increase the cost of document maintenance when the common content needs to be changed, increase the cost of translation and increase the likelihood of copies becoming inconsistent.

- XML natively supports content reuse using entities. Shared content can be stored in a separate file and included into the document using an entity reference. This can cause difficulties when editing fragments because they cannot contain an XML declaration or a DOCTYPE declaration.
- W3C developed XInclude to enable content reuse. XInclude elements are added to an existing DTD such as DocBook. XInclude processing tools are integrated into publishing systems. Content can only be inserted where the XInclude element is valid. This can be inflexible unless the XInclude element is allowed everywhere. Processing errors may occur because there is no mechanism to control the structure that is inserted by XInclude. Inserted content may not be valid when resolved and this may cause problems with processing applications.
- Automated content reuse solutions have been attempted with XML and word processor based applications.
- XML based applications tend to be complex bespoke applications built around standard and proprietary DTDs. The cost of developing these applications and the complexity means they are not easily accessible to smaller organizations.
- Non-XML based applications tend to be lock content into proprietary formats that are not accessible to other applications.
- Non-XML based applications may have formatting problems because of a lack of context or may place unnecessary restrictions on content sharing to bypass this problem.

2.2 Content sharing and interchange

- Organizations may wish to share or interchange content with partners or other organizations.
- Organizations may share and interchange content using word processing files. File formats may not be compatible between the organizations. If shared content is to be reused, it may need to be restyled to suit its new use. Restyling may require considerable effort that needs to be repeated when new content is received.
- Standard DTDs such as DocBook enable sharing and interchange to occur without formatting concerns. However, different organizations may need to customize the DTD in different ways. These customizations may not be compatible with processing applications.
- Successful interchange of XML data normally requires data to be transformed by the receiving application or modifications to be made to

receiving processing applications. Modifications are ongoing as DTDs rarely remain unchanged.

3 How does DITA solve these problems?

3.1 Topic orientated architecture enables content reuse

- DITA uses a topic based architecture for content creation, management and publishing. The base information unit is called a 'topic'. This is a self contained stand-alone unit of information and is the basic unit for reuse in DITA.
- Topics are organized into an output such as a document, help system or web site using a map. Topics are included into the map using a topicref element which identifies the topic to insert. The map can be used to specify the order and groupings of topics that form the output.
- The map model allows topics to be reused in different documents without needing to make copies of the content.
- Topics and maps each have their own DTD. Validation problems that may occur when using XML entities as a method of reuse do not occur.
- The topicref element in the map is similar in function to the XInclude element. However the topicref element can only reference topics. This behaviour is enforced by DITA compliant processing applications. DITA maps will always be resolved into a structure that can be processed by publishing applications.
- DITA also provides content reuse at a lower level than the topic. All elements have a conref attribute that can be used to reference shared content. Processing applications replace the referencing element with the referenced element. The content referenced by the attribute must be the same type as the referencing element, either the same element or a specialization of the element. This ensures that conref will always be resolved into a valid structure and can be processed by publishing applications.
- DITA provides a technical framework to support content reuse. DITA compliant applications will provide for content reuse, out-of-the-box. An organization must still manage their content architecture to fit with DITA's topic based architecture.

3.2 Specialization architecture enables content interchange

- DITA provides an inheritance based method for extending the DTD called specialization.
- Specialization involves creating a new element that is based on an existing element. The "specialized" element must maintain the same content constraints as the existing element. The specialization relationship is retained in the DTD for use by processing applications. Specialization results in an inheritance hierarchy.
- The inheritance hierarchy allows processing applications to recognize and process new elements based on rules found for existing elements in the inheritance hierarchy. DITA can be customized without breaking processing applications. This enables interchange as customized DTDs can always be processed by DITA processing applications.
- Specialization also helps to reduce the cost of maintenance. Adding new elements to the DTD does not always require changes to processing applications and style sheets.

4 Other benefits of DITA

- DITA is a modern DTD with a relatively modest set of elements. Those familiar with XHTML will be familiar with many DITA elements. This should reduce the amount of training required when comparing DITA to other standard DTDs such as DocBook.
- DITA is an OASIS standard. Standards create a larger community of users that make it viable for developers and vendors to get behind the standard and produce compliant tools. This reduces the need for bespoke applications which reduces the cost for smaller organizations.
- Greater number of tools means that organizations can change tools as the market matures and better tools become available without needing to change the content behind the tools.
- DITA which is well supported by the open source community which has produced the DITA Open Toolkit (DITA OT). This toolkit provides processing to generate multiple output formats such as PDF (via XSL-FO), HTML and a number of help formats.
- It is likely that a community will develop around the standard to make it easier to find writers and developers who are experienced with it.

5 Who should use DITA?

- Organizations with product ranges based on standard or common components can significantly reduce the volume of documentation needed.
- The DITA topic and map model allows organizations to tailor content to suit the audience or output format. This is much more flexible than document centric applications where it is not easy to restructure content for different output formats without creating redundant content. The DITA Open Toolkit provides processing to generate the different output formats.
- Organizations that need to translate their content into different languages can use the component based management and reuse features of DITA to reduce translation costs.

6 Maximising the benefits from DITA

6.1 Information architecture

- Moving to DITA is not just a technical change it may involve change to your information architecture and processes to create content. If your existing architecture is document based there is substantial change to move to a topic based architecture. This change needs to be carefully planned.
- Organizations need to develop a content architecture that suits their content and the users of their content. This requires extensive content analysis and identification of content suitable for reuse.
- Organizations need to plan how to migrate existing content to the new architecture. An organization can expect to rework existing content so it is suitable for use with the new architecture.
- The information architecture must be continually managed as new content is created. An information architect must ensure topics are written to be compatible with the architecture and suitable for reuse.
- Writers must be trained in the new information architecture. Writers must be trained to create content that is suitable for reuse. This may be a significant change to the writers style.

6.2 Consider the writers

- Traditional XML developments often overlook the needs of content writers. XML developments often fail when they do not adequately

address the needs of writers. Writers should be involved from the planning stages of the development.

- Authoring tool selection is critical for writer's acceptance. Writers should be fully involved in the process for selecting tools they will use on a daily basis. Consider customising tools to automate common tasks the writers perform. The move to structured authoring should be a benefit to the writers.

6.3 Off the shelf tools

- Care needs to be taken when selecting XML based tools for use with DITA. Specialization relies on features provided by the XPath standard. These features may not be available in some XML tools. Organizations will need to check with vendors to ensure tools support DITA specialization.
- The DITA Open Toolkit (DITA OT) is an open source toolkit for processing DITA content. This toolkit provides tools for processing and resolving DITA maps and conref attributes and transforming DITA content into PDF (via XSL-FO) and HTML as well as various Help formats. DITA OT may need to be integrated with other tools to create a production publishing system.
- XML Authoring tools support for DITA is growing. XMetaL Author, Arbortext Editor (formerly Epic), Adobe Framemaker and other XML editors now provide specific customizations for DITA. However, DITA is only in its infancy at the moment and some tools may be a bit raw at present.
- Rendering applications are providing support for DITA. Commercial XSL-FO renderers can be plugged into the DITA OT. Elкера XML Print provides a DITA customization.
- There will always be the need to do some custom development of off the shelf tools. At the very least, outputs will need to be customized to achieve the formatting required by an organization.