Indexing without Context: Some Thoughts about the New World of DITA and Content Management Systems

BY FRED BROWN

Back-of-the-book indexers use "context" as a key component in their work. Context can consist of: the subject of the publication; the material in a specific page; section, or chapter; and the index headings themselves. Content management systems have a dramatic impact on indexing by removing context from the indexing process. This article examines issues about how the DITA (Darwin Information Typing Architecture) standard for content management systems affects "context" in the indexing process and how these issues can be addressed using controlled vocabulary.

DITA and Content Management Systems

Think of a cook book. DITA provides a common template for each recipe. XML is the language in which the book is written. A content management system acts as the faithful scribe for a world renown chef and author who writes a variety of cook books. Content management systems do the actual work of creating, modifying, publishing and tracking documents. Some content management systems use DITA while others do not; most use XML.

To be precise, DITA is an XML standard for content management that is being widely adopted for creating computer documentation. XML is way of coding text, somewhat similar to HTML, but focused on the meaning of the content rather than how to format it. DITA is a particular flavor of XML designed for publishing computer documentation. DITA organizes a technical publication into discrete information modules called "topics." "Topics" can be of different types: concept, task or reference. And each topic follows a specific template defined by DITA. In this article, we examine the DITA standard to illustrate some of the fundamental issues that content management systems pose for indexing.

Content management systems allow publishers to create small modular "topics" which can be mixed and matched into a variety of publications—print or online. For example, a computer chip manufacturer[1] may have a user guide, installation manual and reference sheets for each new chip or variant that they produce. Each new variant of a chip will require a new set of documentation. By reusing "topics" written previously, our computer chip manufacturer can automatically generate 75% of the documentation for a new variant of a chip—a huge saving in both time and resources. Similarly, a medical publisher could publish "topics" about cancer research in a web portal for oncologists and recycle some of the same "topics" in a text book entitled, *Introduction to Nursing.*

The key issue for indexers about DITA and content management lies in the organization of the text into discrete information modules or "topics." Managing these discrete topics independently during the publishing process leads to a loss of "context" for indexing. An indexer ends up having to index a given "topic" without knowing the main subject of the publication, what other "topics" are included and what the other index entries are. All the indexer sees is the topic at hand with no "peripheral vision" to see how the "topic" and its index entries fit into the rest of the publication (or publications).

Continuing with our cook book example, the "topics" are the individual recipes that go into each cook book. The index entries are embedded in the recipe and travel along with it into whatever book it's published in. For example, pancakes, beef stew and chocolate cake could go into any number of different books. When indexing the Sweet Peach Tarts recipe, the indexer has no idea of whether it's going into *All about Fine Pastry*, *Kids Can Cook*, or *Lovers Delights and Other Secrets*. The indexer doesn't know who the audience is, the main theme of the book, and what the other recipes and index entries are. Different terminology will be needed for pastry chefs, young children and lovers. The approach an indexer takes to each of these three books would be quite different because of the different themes and audiences. How can the indexer possibly begin to

Fred Brown has worked as a technical writer and indexer. He is coauthor of *Index It Right!: Advice from the Experts* (vol. 1) and is a recipient of the Australian and New Zealand Society of Indexers Web Index Award.

meet the needs of these quite different books, all at the same time, and without even knowing what these different books are?—and other new cook books may well be created long after the Sweet Peach Tarts recipe has been written and indexed. We should also note that the author will find it similarly challenging to write a Sweet Peach Tarts recipe that will be suitable for all the cook books.

Closed– Versus Open–System Indexing

In regular back-of-the-book indexing, index entries are created within the context of the book as a whole. Because all of the material being indexed is finite and known, this type of indexing is called "closed-system" indexing. Typically, one also knows who the audience is and an indexer experienced in the field may have a reasonable idea of the needs of the audience. All of the book's content and needs of the audience form a "context" within which the index is created. The index entries themselves also form a context in which other index entries exist.

Some back-of-the-book indexers go as far as to say that in indexing "context is everything." Context is key to:

- selecting relevant material in the text to index
- selecting the relevant terminology
- differentiating different locations where a broad topic is discussed
- gathering related information together
- directing the user to the appropriate place in the index

In open-system indexing, an infinite amount of material can be added without affecting the existing index headings or the structure of the index. Metadata is an example of open-system indexing. DITA and content management lead to open-system indexing because the amount of material is vast compared to a single publication. Material is regularly being added, revised or removed. In DITA and other content management systems, the specific index headings and the overall index structure have to work across a whole range of publications, not just one particular publication or version of that publication.

Essentially, the problem facing back-of-the-book indexing in DITA and content management systems is to create a closed-system index in an open-system environment.

Creating Index Entries

DITA and content management create complications for the finely crafted, back-of-the-book index. Index headings written for a given topic in DITA will be reused in many different publications. Unlike traditional back-of-the-book indexes, the index headings in DITA topics cannot be tailored to a specific publication.

Do Mi Stauber states, "The metatopic [main subject of the publication], whether explicitly indexed or merely kept in mind, informs every indexing decision."[2] In a technical publication, the main headings in the index exist in relationship to the title of the publication (given an explicit and accurate title). So what happens when you are indexing a topic that could go into any number of different publications, now or in the future?

Imagine a DITA topic about "Receiving Fax Messages on a Multi-Function Center (MFC)". In the *MFC User Guide*, one would wish to have the following index headings for this topic:

```
fax messages, receiving
...
receiving, fax messages
```

Note that you would **not** wish to have the following index entry with the main heading "MFC" because the whole publication is about the MFC:

```
MFC, fax messages
```

However, if this topic is reused in a publication entitled, *How to Run Your Home Office*, then you would wish to have the above index marker. As we can see, when we remove the context of the publication as whole, creating multi-purpose index markers that can fit into any publication becomes quite challenging.

One possible solution would be to include all possible index markers in the DITA topics and then remove the unnecessary index markers at the editing stage for a given publication. Within a publication, one might store rules for editing the index automatically. For example, the *MFC User Guide* could have a rule that says to remove all index markers which have "MFC" as the main heading.

Creating Cross-references

We also have the issue of where to store the "See" and "See also" cross references. Cross-references have no reference locators (page numbers) and thus cannot be attached to any specific DITA topic that will be reused in different publications. In fact, cross-references can only be specified for a given publication once all of the index headings are known. For example, in our index to the *MFC User Guide* we may need the following "See" cross reference:

```
messages. See fax messages
```

The publication *How to Run Your Home Office* will need an additional cross-reference from "printers" to "MFC" that would be unnecessary for the *MFC User Guide* because the *MFC User Guide* has no index heading for "MFC." The "See" cross-reference in the *MFC User Guide* changes to a "See also" cross-reference in *How to Run Your Home Office* because the main heading "messages" now has some subheadings. Note also that this "See also" cross-reference in *How to Run Your Home Office* has some additional items—"phone messages" and "text messaging":

```
messages
    receiving
    sending
    See also fax messages; phone messages; text messaging
...
MFC, fax messages
...
printers
    cost of
    paper requirements
    types of
    See also inkjet printers; laser printers; MFC
```

Controlled Vocabulary

When reusing index entries, DITA recommends that they be written with "maintenance" in mind[3]. As the previous examples show, creating index markers that can be "maintained," or more precisely, "reused," puts DITA into a different league from straight back-of-the-book indexing. Controlled vocabulary provides a way to simplify the problem of "maintenance."

In indexing, controlled vocabulary helps ensure consistency across different publications. For example, one could insist that all main headings and most sub-headings use preferred terms. In this case, index headings created independently in different "topics" would automatically mesh properly when compiled in an index for a specific publication.

In large open publishing environments such as DITA, vocabulary control is essential both to writers and to indexers. DITA allows for vocabulary control both internally within its own structure and externally through thesauri or other controlled vocabularies. Internally, DITA employs the concept of "domains" in relation to "topics."[4] Using domains, specialized vocabulary can be defined for a specific "topic" or shared among many topics. DITA also permits external controlled vocabularies to be used in metadata such as the Dublin Core Subject element.

The information for generating "See" and "See also" cross-references could also be stored within a controlled vocabulary. Equivalence relationships, defined in thesauri as "Used for",

store information required to generate "See" cross-references in the index. Hierarchical relationships and associative relationships, defined respectively in thesauri as "Narrower Term" and "Related Term" relationships, store information required to generate "See also" cross-references. For simplicity and ease of maintenance, hierarchical relationships would be most easily represented in the index using "See also" entries. There are five other ways to represent hierarchical relationships in an index[5], but using the "See also" cross-reference is the most common and easiest to use. Automatically generating the cross-references in the index does imply some "dumbing down" of the editing process in order to make the indexes more easily maintainable.

How you handle cross-references[6] can be influenced by the publication format. Where print publishing is being abandoned in favor of online, "See" cross-references may be replaced with double-posting the index entries directly under the synonyms themselves. In practice, "See also" cross-references can be the most difficult to deal with. Especially in smaller publications, cross-references may come up blind because there is no matching index heading for the cross-reference. One approach to minimizing the problem is to limit "See also" cross-references to just one reference only.

When automatically generating the cross-references using the controlled vocabulary, some additional rules would also need to be applied. All the terms appearing in a cross-reference would also need to appear in the generated index for the specific publication. No blind cross-references please! A "See also" cross-reference with no subheadings would need to be converted to a "See" cross-reference. A "See" cross-reference with sub-headings would need to be flagged for review because all regular index markers should use preferred terms only.

Generating Metadata

Publishing to an intranet or the internet often requires metadata to support a "smart" search function. Using controlled vocabulary to create all or most of the index markers provides the opportunity to automatically generate the metadata from the index markers. In DITA, metadata generated from the index would fit into either the keywords element (in the "topic" prolog section) or in the Dublin Core Subject element. Thus the back-of-the-book indexing work could be transformed automatically into metadata tags to make an online publication searchable.

Looking ahead

In a way, controlled vocabulary ends up providing "context" by default. Particular material is indexable because it contains concepts that exist in the controlled vocabulary. The controlled vocabulary also defines the relationships used to create the navigation within the index. And the "topics" themselves may be written using controlled vocabulary defined within DITA.

Controlled vocabulary may offer the greatest benefit for context management systems where there is a single audience for all of the publications, as is the example of the computer chip maker discussed earlier. In the examples of the medical publisher and the cook book author, where there are different audiences for the publications, both writing and indexing can become problematic depending on how diverse the needs of the different audiences are.

Because of the relative newness of DITA and content management, compared to hundreds of years of back-of-the-book indexing experience, many issues remain for indexing in DITA. The problem is fitting a closed-system process into an open-system environment—of fitting the old, tried and true into the new and innovative. We will need to borrow principles from both back-of-the-book indexing and metadata, including controlled vocabularies. Finally, we will need to look at what compromises need to be made to make indexing both workable for the customer and maintainable for the documentation team.

Notes

1. Presentation by Jim Ingram from IBM Microelectronics entitled, "From the real toward the ideal: a case study in virtual document development" given at the SIGDOC 1997 conference in Salt Lake City.
2. Do Mi Stauber. *Facing the Text–content and structure in book indexing*. p.89. (2004).
3. OASIS DITA Wiki. "Indexing issue summary." *(www.wiki.oasis-open.org/dita/Indexing_issue_summary)*
4. See "DITA typed topic specializations (infotyped topics)" *in Introduction to the Darwin Information Typing Architecture (www.ibm.com/developerworks/xml/library/x-dita1/#h8)* by IBM.
5. Brown, Fred. "Hierarchical Relationships in Indexes" *Key Words*. (October-December 2003).
6. From *Jan Wright's* experience with open-ended indexing.

References

Brown, Fred. "Vocabulary Links:// Thesaurus Design for Information Systems — seminar by Dr. Bella Hass Weinberg," Key Words (November/ December, 1998).

IBM. *Introduction to the Darwin Information Typing Architecture* (2001, revised 2005) *(www.ibm.com/developerworks/xml/library/x-dita1/)*

OASIS. "Indexing issue summary," *OASIS DITA Wiki*. *(www.wiki.oasis-open.org/dita/Indexing_issue_summary)*