

# Detailed Instructions: IXMLEmbedder with XML Documents

by Angela Howard

Allow one day at the beginning of the project for steps 1-3. Allow one to two days at the end of the project for steps 6-7. For steps 4-5, where you're using Sky or Cindex to index the book, estimate your usual amount of indexing time.

## 1. Receive the XML files from the client, make backup copies, and generate an original PDF file if necessary:

- a. Download the XML files from the client using whatever procedure the client has provided. Often, you will need to check out the files from a version control system. Put the files into a *project* directory.
- b. Back up the files: Create a subdirectory called *originals* under the *project* directory and copy all the XML files into it.
- c. Create a PDF file of the book, using whatever procedure the client provides for this, and save it to your *project* directory. Save this file as *book-original.pdf*.

## 2. Set up a project in IXMLEmbedder:

- a. In IXMLEmbedder, in the **Project** tab, create a new project file by entering the name *project* for **Project Name** and clicking **Save Project** to choose the directory. The directory you choose must be the *project* directory, the same directory where your XML files reside.
- b. Leave **use file numbers in ids** unchecked. I'll explain why in step 3b.
- c. In the **Documents** tab, under **Project Type**, check **tag set** and choose the XML type from the drop-down menu.
- d. In the **file extension** box, enter the file extension for your documents *if it differs from the default*. IXMLEmbedder assumes a default file extension for each XML tagset. For an AsciiDoc project the default is *.asciidoc*; DITA projects have a default of *.dita*; DocBook projects have a default of *.xml*; HTMLBook projects have a default of *.xhtml*; LaTeX projects have a default of *.tex*; and TEI projects have a default of *.xml*.
- e. Uncheck **report unused ids**. Usually, you will enter a lot more document IDs in the files than you will use, so unused IDs are to be expected. That's why I recommend unchecking this box. Otherwise, you will get a large number of errors in the IXMLEmbedder log file, making it difficult to see the more important errors.
- f. Click **Add** for each XML file, in order, to create the list of documents in the documents box. The **Add Document** window will open, and you need to enter the file name (without the extension), and the starting and ending page numbers for that file. Use the page numbers in the PDF file to determine the starting and ending page numbers for each chapter. Note that these page numbers are not really used by IXMLEmbedder when embedding entries; they are just entered here for your reference. Only add documents that will be indexed. For example, if there are separate files for the front matter or back matter of the book that will not be indexed, you don't need to include those in the document list.
- g. In the **Indexes** tab, click **Add** to set up an index. Most XML tagsets will only allow the creation of one index, typically a subject index. The **Add Index** window will open. To create one subject index, enter *project* as the name of the **Index (IXML) file**, and **subject** for the **code**. You will create a file called *project.xml* in step 6a.
- h. Choose a **sort order** of **word by word**, the default, which allows the book generation process in step 6e to determine the sorting. Click OK.
- i. The **cross references placement** option isn't used by XML tagsets.
- j. The **lead word** should be the same word you use to identify cross references in Sky or CINDEX, usually **see**.
- k. Go back to the Project tab and click **Save Project**.

### 3. Insert document IDs into the XML files:

- a. Open the first XML file in a text editor or XML editor.

**Tip:** I use EditPlus. Unless you're going to be using the advanced features of an XML editor, like XMLmind or Oxygen, it's not really necessary to spend the extra money for it, for the purposes of this process. If you already have an XML editor and are used to it, that's your best option. If not, a text editor is fine, but Notepad is perhaps a little too primitive. EditPlus is a text editor that allows you to edit multiple files at once, and it has colored highlighting that makes it easier to separate the tags from the text, and that's all I find necessary. The whole point of using IXMLEmbedder is that you don't have to work in the XML files very much anyway.

- b. Open IXMLEmbedder and go to the **Documents** tab. Select the first chapter of book in the documents list. Now look at the **ID generation** box. If the first chapter is chapter 1, click **Increment chapter number** to set it to 1. If it is a preface or something prior to chapter 1, leave it set to 0 (zero). Since the chapter number will be included in the document ID, and there is one chapter per file, there is no need to also include file numbers in the document ID (see step 2b).
- c. Arrange the windows on your display so that the IXMLEmbedder window is immediately to the left of your editor with the first XML file open. In IXMLEmbedder, click on **Increment sequence number and copy ID to Clipboard**. You'll notice that the sequence number starts out at 10. In the editor, place your cursor at the beginning of the first paragraph of text and type Ctrl-V to paste the document ID. The pasted document ID will look like this: `<!-- id=1.10 -->` if you are in chapter 1, or `<!-- id=10 -->` if you are in chapter 0.
- d. In IXMLEmbedder, click on **Increment sequence number and copy ID to Clipboard** again. The sequence number will increment by 10 each time. In the editor, place your cursor at the beginning of the next paragraph of text, and type Ctrl-V to paste the document ID, which will be `<!-- id=1.20 -->` or `<!-- id=20 -->`.
- e. Repeat this process throughout the file, and then for each of the XML files in turn, incrementing the chapter number as appropriate for each successive chapter.

**Tip:** I recommend inserting a document ID at the beginning of every paragraph, and at the end of the last paragraph in a section, to allow for whatever ranges you might need. You don't have to stop and think about where you might need an index entry that way. The alternative is to skim each paragraph and determine whether the information is significant enough to warrant any index entries, and that just makes this process go a lot slower. I also found it quicker to go through this process once I had changed my computer settings to automatically change focus to the window where my cursor was, rather than having to click in each window to change focus to it. You can find this setting under Ease of Access Center in the Control Panel, under "Make the mouse easier to use" and checking the box for "Activate a window by hovering over it with the mouse."

- f. Once document IDs are inserted in all the XML files, make a copy of those files a subdirectory called *labeled*. This saves a copy of all the XML files containing the document IDs so that you can repeat the embedding process later if you need to. (See step 6f.)
- g. Generate a PDF file of the book using the files containing the document IDs, so that the document IDs are shown along with the text in the PDF file. Name the generated PDF file *project\_withIDs.pdf*. While you're indexing in Sky or CINDEXT (in step 4), you'll want to be able to look at this file to see where the document IDs are. You could index the book while looking at the XML files, but they can be more difficult to read, and you will often not be able to see external figures or examples, which will make it more difficult to understand the text.

**XML variations:** Some XML types will recognize the document IDs as HTML comments, so they will not show up when you generate a PDF file. If this is the case, what I have done is alter the document ID syntax just for the purpose of generating this PDF file. Make sure the files with the document IDs in their correct syntax are saved to the *labeled* directory first (see step 3f). Then, in the *project* directory, do a global search and replace on all the document IDs to change them from `<!-- id=1.20 -->` to just `id=1.20-`. Now when you generate a PDF, these document IDs will be displayed in the PDF file. When you're done, just copy all the files from the *labeled* directory to your *project* directory so your document IDs are in the correct syntax again.

**4. Create the index in Sky or CINDEXT:**

- a. Open Sky or CINDEXT, and create a new index with the name *project*. In another window, open the *project\_withIDs.pdf* file with the document IDs you created in step 3g.
- b. Configure Sky or CINDEXT to recognize a chapter component in the locator with a period "." as the separator. Also make sure that the locators are not conflated. In Sky, under Index Options, in the Locators tab, in the Chapters tab, check "Recognize Chapter Labels", but make sure that "Merge Chapter Labels" is unchecked. Set Conflation to "None".
- c. Create index entries in Sky or CINDEXT, using the document IDs in the PDF file as locators. For example, if the document ID "`<!-- id=1.90 -->`" is at the beginning of a paragraph where you would want to place an index entry, use "1.90" in the locator field. When entering locators in Sky, I can type a locator range as "1.90-150", and it will automatically be output as 1.90-1.150 because I set conflation to none.

**XML Variations:** Keep in mind that, depending on which type of XML you're using, there are most likely some restrictions on the types of index entries that are allowed. You'll want to ensure that you don't enter the types of entries in Sky or CINDEXT that are not allowed by the XML type so you won't get errors later. Here are some things to watch out for:

Subheading levels	Sometimes only two levels of entries, a main heading and one subheading level, are allowed.
Cross references	Cross-references are sometimes only allowed from the main heading and not from subheadings.
Special characters	Special characters in index entries sometimes cause problems, and you may have to use substitutions, such as "&lt;" for "<" and "&gt;" for ">".
"See also" placement	During the embedding process (step 6c), the index tags for "see" and "see also" cross references, are placed at the beginning of the first XML file of the book. For "see" entries, this is ok, because no page number is output with a "see" entry. But with some XML variations, page 1 is output along with the cross reference, because the index tag is on page 1 of the book. In these cases, you'll need to manually move the "see also" index tags next to another index entry with a valid page number.

**5. Edit your index in Sky or CINDEXT, until it is in final form.**

**6. Embed the index entries into the XML files:**

- a. Export your index as an IXML file. I use Sky, so I choose File, then Export, and select "XML Records 'ixml' File" as the output type. This will create a file named *project.ixml*.

**Note:** For Sky version 7 only, you will need to add a couple of lines to the *project.ixml* file. Open the *project.ixml* file in your text editor. Add a `<records>` tag on a line immediately before the first `<record>` tag. At the bottom of the file, add a `</records>` tag on a line immediately after the last `</record>` tag.. (This will be fixed in Sky version 8.)

- b. In IXMLEmbedder, go to the **Indexes** tab. If you have more than one index in the list of indexes, select the index you want to embed entries for. Usually there will just be a **subject** index. Click on it to select it.
- c. Click on the **Embed** button to embed all the entries from the *project.ixml* file into the XML files. The entries are translated into the proper index tag syntax, according to which XML tagset you selected in step 2c, and inserted into the XML files in place of the corresponding document IDs. If any document IDs in the XML files are not used in your index, the document IDs will be removed, and no index tags will be inserted in their place. If your index uses a document ID that does not occur in the XML files, that will be reported in the log file. A new subdirectory called *embedded* will be created, and the XML files with embedded entries will be saved there. The XML files in the *project* directory will remain unchanged so that you can repeat this process if necessary.
- d. Look at the log file from IXMLEmbedder to see if there are any errors you need to fix. You'll need to reference the IXMLEmbedder user guide to determine what specific errors mean. If there are errors you need to correct (some log file entries are just warnings or informational messages), you can make the changes in Sky or CINDEXT and then repeat the process starting at step 6a.

- e. Once you have corrected any errors in the log file, you can generate a new PDF file from the XML files that are now embedded with index tags. Remember that the files with embedded index tags are in the *embedded* directory underneath your *project* directory. The client process that you use to generate a PDF file will probably expect the embedded files to reside in the *project* directory, so you'll need to copy the files from the *embedded* directory to the *project* directory first.
- f. Compare the index in the generated PDF file to the index generated by Sky or CINDEK, and check to make sure they are the same. Resolve any discrepancies as necessary. If you need to go back to Sky or CINDEK to make changes, you'll need to also copy the files in the *labeled* directory back to the *project* directory, and repeat the process starting at step 6a.

**7. Return all XML files with embedded index tags, and the final PDF file, to the client.**