Conventions for Writing Proper DocBook XML ("Formatting Guide")

Conventions for Writing Proper DocBook XML ("Formatting Guide")

Abstract

This guide details how to write proper DocBook XML, primarily for use with the MySQL DocBook sources used for the MySQL manual.

Document generated on: 2012-12-22 (revision: 33724)

Table of Contents

Conventions for Writing Proper DocBook XML (“Formatting Guide”) v
1. Introduction to DocBook 1
2. IDs Within a Document 3
3. Links (or References) Within a Document 5
4. Quotes 7
5. DocBook Elements We Use 9
5.1. answer 9
5.2. caution 10
5.3. colspec 10
5.4. command 10
5.5. email 10
5.6. emphasis 10
5.7. entry 10
5.8. figure 11
5.9. filename 11
5.10. graphic 11
5.11. guibutton 11
5.12. guilabel 12
5.13. guimenu 12
5.14. imagedata 12
5.15. imageobject 12
5.16. indexterm 12
5.17. informaltable 12
5.18. itemizedlist 13
5.19. keycap 13
5.20. link 13
5.21. listitem 13
5.22. literal 14
5.23. mediaobject 15
5.24. option 15
5.25. orderedlist 16
5.26. para 16
5.27. phrase 16
5.28. primary 16
5.29. programlisting 16
5.30. qandaentry 16
5.31. quandaset 17
5.32. question 17
5.33. quote 17
5.34. remark 18
5.35. replaceable 18
5.36. row 18
5.37. secondary 18
5.38. tbody 19
5.39. tgroup 19
5.40. textobject 19
5.41. title 19
5.42. ulink 19
5.43. userinput 19
5.44. xref 19
6. File Names and Guidelines for Graphics 21

Conventions for Writing Proper DocBook XML (“Formatting Guide”)

This chapter provides guidelines for writing MySQL documents using DocBook:

- Introduction to DocBook
- General document-writing conventions
- How to begin a new document
- How to structure a DocBook document
- Which DocBook XML elements should be used for which purpose
- How to use IDs, links, and quotation marks
- What to do when you begin using a DocBook element that we have not been using before
- Graphics markup and special markup

A few basic hints for editing before we start:

- **UTF-8:** Make sure your editor uses UTF-8 encoding.
- **Spaces, not tabs:** Use spaces, not tabs. Use 2 spaces to indent.
- **Unix line endings:** Make sure your editor saves your edits using Unix-style line endings (LF), rather than Windows-style (CR/LF) or Mac OS-style (CR) line endings.

Chapter 1. Introduction to DocBook

DocBook contains a large number of XML elements from which we only use a subset.

Be aware that XML (not just DocBook XML) has two special characters that cannot be used literally in text: `<` (the opening angle bracket) and `&` (the ampersand sign). Whenever you need to use those two characters literally, you have to use entities, like in this example: `<literal> & <programlisting>`.

Alternatively, you can wrap any content that contains characters that cannot be used literally within CDATA containers, like this:

```
<![CDATA[(content that might contain characters illegal in XML)]]>
```

This makes sense particularly for larger pieces of code within `<programlisting>` or similar containers.

We structure documentation using nested `section` elements. A `book` as DocBook defines it would look like this:

```
<book>
 <chapter>
  <section>
   <section>
    ...
   </section>
  </section>
 </chapter>
 <appendix>
  <section>
   ...
  </section>
 </appendix>
</book>
```

Chapter 2. IDs Within a Document

In DocBook, you may assign an ID to virtually every element used. Although this might be useful in some cases, it also adds a lot of overhead to document, as those IDs need to be managed and handled. Therefore, we use IDs only for a few elements, particularly for the structural elements like `book`, `chapter`, `appendix`, and `section`.

IDs have to be unique within a document. Note that the term “document” may refer to more than just one file, for example when you include many XML files in one “master” file.

ID names may not contain spaces or special characters. They may (and should), however, contain dashes, to make them more readable.

Chapter 3. Links (or References) Within a Document

We use the `<xref>`, `<link>`, and `<ulink>` elements to refer (or link) to other resources. Note that we prefer `<xref>` over `<link>`.

Normally, we use the `<xref>` element, like this:

```
...
<para>
Here is a link that points to <xref linkend="some-id"/>.
</para>
<section id="some-id">
<title>Important Section</title>
</section>
```

In this example, the link description is taken from the title of the referenced section, so that in most output formats, it will look like this:

```
Here is a link that points to Important Section.
```

In output formats that can display the link description as “hot” (or clickable), “Important Section” becomes a hotspot. This is, for example, true for HTML and PDF. In output formats where this is not possible (like printed books), our XSL stylesheets take care of adding other information, so that the above example might look like this:

```
Here is a link that points to Section 10.5 Important Section.
```

You should use the `<link>` element only if you have to use text other than the linked element's title as the link description, like in the following example:

```
...
<para>
Here is a link that points to an <link linkend="some-id">important section</link>.
</para>
<section id="some-id">
<title>Important Section</title>
</section>
```

For both `<xref>` and `<link>`, *do not use numbers to refer to elements!* For example, do not use something like this:

```
See  <link linkend="some-id">10.5</link>.
```

For information about (automatic) numbering, see Guidelines for Numbering.

Also, do not add other information like the words “chapter” or “section” to your link. For example, do not use something like this:

```
See section <link linkend="some-id">Important Section</link>.
```

Our XSL stylesheets take care of adding such information *in the correct language* (bear in mind translations) when appropriate.

For links referring to resources other than XML documents, you should use the `<ulink>` element, which has the following syntaxes:

```
<ulink url="target-resource">descriptive text</ulink>
<ulink url="target-resource"/>
```

The second form can be used if the content of the element is the same as the URL attribute. Here is an example:

```
The MySQL Reference Manual is
<ulink url="http://dev.mysql.com/doc/mysql/">available online</ulink>
or can be downloaded in a number of formats. For example, you can find the
PDF version here:
<ulink url="http://dev.mysql.com/get/Downloads/Manual/manual.pdf/from/pick"/>
```

Chapter 4. Quotes

The rules for entering quote characters are pretty straightforward:

- Double quotation marks: `<quote>example</quote>`
- Double quotation marks in program listings: `"` (entered verbatim)
- Single quotation marks: `'` (entered verbatim)
- Backticks: `` ` `` (entered verbatim)

See Section 5.33, “quote”.

Chapter 5. DocBook Elements We Use

Table of Contents

5.1. answer 9
5.2. caution 10
5.3. colspec 10
5.4. command 10
5.5. email 10
5.6. emphasis 10
5.7. entry 10
5.8. figure 11
5.9. filename 11
5.10. graphic 11
5.11. guibutton 11
5.12. guilabel 12
5.13. guimenu 12
5.14. imagedata 12
5.15. imageobject 12
5.16. indexterm 12
5.17. informaltable 12
5.18. itemizedlist 13
5.19. keycap 13
5.20. link 13
5.21. listitem 13
5.22. literal 14
5.23. mediaobject 15
5.24. option 15
5.25. orderedlist 16
5.26. para 16
5.27. phrase 16
5.28. primary 16
5.29. programlisting 16
5.30. qandaentry 16
5.31. quandaset 17
5.32. question 17
5.33. quote 17
5.34. remark 18
5.35. replaceable 18
5.36. row 18
5.37. secondary 18
5.38. tbody 19
5.39. tgroup 19
5.40. textobject 19
5.41. title 19
5.42. ulink 19
5.43. userinput 19
5.44. xref 19

5.1. answer

Used for answers to a question posed in a `quandaset`.

http://docbook.org/tdg/en/html/answer.html

5.2. caution

Used where a procedure may result in data loss if not carried out correctly. Used to indicate great care should be taken.

http://docbook.org/tdg/en/html/caution.html

```
<caution>
  <para>
    The following procedure should be carried out with care...
  </para>
</caution>
```

5.3. colspec

Used to denote column widths within an `informaltable`.

```
<informaltable>
<tgroup cols="2">
<colspec colwidth="30*"/>
<colspec colwidth="50*"/>
<tbody>
...
```

http://docbook.org/tdg/en/html/colspec.html

5.4. command

Used for program names.

```
To dump all databases, you can use <command>mysqldump <option>- -all-
databases</option></command>.
```

http://docbook.org/tdg/en/html/command.html

5.5. email

Used for email addresses.

```
<email>stefan.hinz@oracle.com</email>
```

http://docbook.org/tdg/en/html/email.html

5.6. emphasis

Used to emphasize important parts of text. If you want boldface, add a `role` attribute:

```
<emphasis role="bold">Note:</emphasis> Remember that ...
```

http://docbook.org/tdg/en/html/emphasis.html.

5.7. entry

Used for column entries within rows of a table.

5.8. figure

Used as a wrapper to include graphics (and possibly other media).

```
<figure>
 <title>Caption of the graphic</title>
 <mediaobject>
   <imageobject>
     <imagedata fileref="screenshot.en.png" lang="en"/>
   </imageobject>
   <imageobject>
     <imagedata fileref="screenshot.de.png" lang="de"/>
   </imageobject>
   <textobject>
     <phrase lang="en">A screenshot</phrase>
   </textobject>
   <textobject>
     <phrase lang="de">Ein Mattscheiben-Foto</phrase>
   </textobject>
 </mediaobject>
</figure>
```

Every figure should be preceded by a specific in-text reference (for example: `see Figure figure-title`, `Figure figure-title shows`, etc.). Figures should not be introduced with colons or phrases like “in the following figure,” or “as shown in this figure.” Figures float, so the lack of specific in-text references may cause incorrect placement of figures.

As for capitalization of figure captions, see Guidelines for Capitalization of Terms. Do not use periods at the end of captions.

See also Section 5.23, “mediaobject”. As for file names of graphics, see Chapter 6, *File Names and Guidelines for Graphics*.

http://docbook.org/tdg/en/html/figure.html.

5.9. filename

Used for files, directories, and paths.

```
Change the server configuration using the <filename>my.cnf</filename> option
file.
```

http://docbook.org/tdg/en/html/filename.html

5.10. graphic

DEPRECATED: DO NOT USE: Formerly used to include graphics.

http://docbook.org/tdg/en/html/graphic.html

5.11. guibutton

Used for buttons in a graphical user interface (GUI).

```
Click <guibutton>OK</guibutton> to select that option.
```

http://docbook.org/tdg/en/html/guibutton.html

5.12. guilabel

Used for labels in a graphical user interface (GUI).

```
Those options are listed in the group labeled <guilabel>Server Settings</
guilabel>.
```

http://docbook.org/tdg/en/html/guibutton.html

5.13. guimenu

Used for menus and menu items (we do not use `<guimenuitem>` and `<guimenusubmenu>` yet) in a graphical user interface (GUI).

```
To open another program window, select <guimenu>File</guimenu>, <guimenu>New
Instance Connection ...</guimenu>.
```

http://docbook.org/tdg/en/html/guimenu.html

5.14. imagedata

See Section 5.23, “mediaobject”.

5.15. imageobject

See Section 5.23, “mediaobject”.

5.16. indexterm

Used to mark index entries.

```
<para>
The Tiger
<indexterm>
 <primary>Big Cats</primary>
 <secondary>Tigers</secondary>
</indexterm>
is a very large cat indeed.
</para>
```

See also primary and secondary.

http://docbook.org/tdg/en/html/indexterm.html

5.17. informaltable

Used for tables.

```
<informaltable>
 <tgroup cols="2">
  <colspec colwidth="30*"/>
  <colspec colwidth="50*"/>
  <tbody>
   <row>
    <entry>
     <emphasis role="bold">Operating System</emphasis>
```

```
    </entry>
    <entry>
     <emphasis role="bold">File-size Limit</emphasis>
    </entry>
   </row>
   <row>
    <entry>
     Linux-Intel 32-bit
    </entry>
    <entry>
     2GB, much more when using LFS
    </entry>
   </row>
  </tbody>
 </tgroup>
</informaltable>
```

See also tgroup, tbody, row, and entry.

http://docbook.org/tdg/en/html/informaltable.html

5.18. itemizedlist

Used for lists in which each entry is marked with a bullet or other dingbat.

```
<itemizedlist>
 <listitem><para>
  This is a list element.
 </para></listitem>
 <listitem><para>
  This is another list element.
 </para></listitem>
</itemizedlist>
```

See also listitem.

http://docbook.org/tdg/en/html/itemizedlist.html

5.19. keycap

Used to wrap the text printed on a key on a keyboard.

```
Press <keycap>CTRL</keycap>, <keycap>ALT</keycap>, and <keycap>DELETE</keycap>
to invoke the Windows Task Manager.
```

http://docbook.org/tdg/en/html/keycap.html

5.20. link

Used to reference resources within a document. Note that we preferably use the `xref` element for this.

5.21. listitem

Used for items of a list. Note that a `listitem` should always contain a `para` (paragraph) element. This is needed for better rendering in various output formats.

See also itemizedlist and orderedlist.

http://docbook.org/tdg/en/html/listitem.html

5.22. literal

Used for a variety of purposes, like character set names (`<literal>latin1</literal>`) or SQL statements and SQL functions (`<literal>SELECT * FROM tbl_name</literal>`).

Here is a list of types of terms that should be wrapped in the `<literal>` element. This list is by Pearson. The way Pearson describes these terms is that they are “monospaced” or “mono.” In some cases, we use elements other than `<literal>` (this is noted in parens). When producing output for Pearson, we map *all* these terms onto mono. For other types of output, we may display some of them differently. For example, the DocBook PDF stylesheets display <command> in bold and <filename> in italic.

- arguments
- arrays
- class names
- code commands (in some cases it is more appropriate to use `programlisting`)
- constructors
- data types
- directives
- DOS programs used as commands (we use `command`, and `option` for program options)
- email addresses (we use `email`)
- events
- fields
- flags
- functions
- HTML tags (remember to properly encode instances of `<` and `&`)
- Internet addresses (we use `ulink`)
- keywords
- logical operators
- loops
- methods
- newsgroup names (we use `ulink`)
- objects
- onscreen messages
- parameters
- pointers
- procedures

- properties
- shell scripts
- statements
- structures
- symbolic constants
- system prompts
- switches (and options)
- units
- URLs (we use `ulink`)
- values
- variables

http://docbook.org/tdg/en/html/literal.html

5.23. mediaobject

Used for referencing images and other media. This element serves as a wrapper around these elements:

- `imageobject`: Used to wrap image data. An imageobject can wrap exactly one image file. If more than one image file should be included in the mediaobject, you have to specify more than one imageobject. As in the following example, this could be done for multiple languages. An imageobject contains exactly one `imagedata` tag.
- `textobject`: Used to wrap a description that shows up as the `alt` attribute to the HTML `img` tag. `textobject` contains exactly one `phrase`. If more `phrase`s are needed, you need to specify more than one `textobject`.

Apart from `imageobject` and `textobject`, `mediaobject` may contain other XML elements for media files, such as `videoobject` and `audioobject`.

```
<mediaobject>
  <imageobject>
    <imagedata fileref="screenshot.en.png" lang="en"/>
  </imageobject>
  <imageobject>
    <imagedata fileref="screenshot.de.png" lang="de"/>
  </imageobject>
  <textobject>
    <phrase lang="en">A screenshot</phrase>
  </textobject>
  <textobject>
    <phrase lang="de">Ein Mattscheiben-Foto</phrase>
  </textobject>
</mediaobject>
```

http://docbook.org/tdg/en/html/mediaobject.html

5.24. option

Used for options of software commands.

```
To dump all databases, use the <option>--all-databases</option> option of the
<command>mysqldump</command> program.
```

http://docbook.org/tdg/en/html/option.html

5.25. orderedlist

Used for lists in which each entry is marked with a sequentially incremented label.

```
<orderedlist>
 <listitem><para>
  This is a list element. It is prefixed with 1.
 </para></listitem>
 <listitem><para>
  This is another list element. It is prefixed with 2.
 </para></listitem>
</orderedlist>
```

See also listitem and itemizedlist.

http://docbook.org/tdg/en/html/orderedlist.html

5.26. para

Used for paragraphs.

```
<para>This is a paragraph.</para><para>This is another paragraph.</para>
```

http://docbook.org/tdg/en/html/para.html

5.27. phrase

See Section 5.23, “mediaobject”.

5.28. primary

Used for primary index entries. See also <indexterm> and <secondary>.

http://docbook.org/tdg/en/html/primary.html

5.29. programlisting

Used for SQL examples and other code listings.

In the output created from the XML (HTML, PDF, CHM, and so forth), everything between `<programlisting>` and `</programlisting>` is displayed literally. When you indent a program listing the indentation also shows literally.

Within `programlisting` elements, make sure that code lines do not run longer than 72 characters. Code lines should be indented using spaces, not tabs.

http://docbook.org/tdg/en/html/programlisting.html

5.30. qandaentry

Used for question/answer sets within a quandaset.

http://docbook.org/tdg/en/html/qandaentry.html

5.31. quandaset

Used for lists consisting of questions and answers, and can be divided into sections. Every entry in a `<qandaset>` must contain a `question`, but `answer` elements are optional (some questions have no answers), and may be repeated (some questions have more than one answer). Common uses for this element include reader questionnaires and lists of “Frequently Asked Questions” (FAQs).

```
<qandaset defaultlabel="qanda">
 <qandaentry>
  <question>
   <para>
    To be, or not to be?
   </para>
  </question>
  <answer>
   <para>
    That is the question.
   </para>
  </answer>
 </qandaentry>
</qandaset>
```

The `defaultlabel` attribute identifies the default label that should be used for `question` and `answer` elements:

- `quanda`: Questions are labeled “Q:” and Answers are labeled “A:”. Other similar labels may be substituted, for example, the words might be spelled out, “Question:” and “Answer:”, and the actual characters or words used are dependent on the language.
- `number`: The entries are enumerated.
- `none`: No distinguishing label precedes questions or answers.

If no value is specified, the implied presentation may be any one of these, as defined by the stylesheet. Note that each question and answer can explicitly define a label, regardless of the default label specified.

See also quandaentry, question, and answer.

http://docbook.org/tdg/en/html/qandaset.html

5.32. question

Used for questions in a quandaset.

http://docbook.org/tdg/en/html/question.html

5.33. quote

Double quotation marks (“example”) are entered like this:

```
<quote>example</quote>
```

Exception: In program listings, they're written verbatim:

```
"example"
```

The difference between “example” and "example" might not be visible in every output format (particularly in HTML). For printed material, however, it's important to use the `<quote>` element wherever appropriate.

For writing curly-quoted single literal characters, you should use the `‘` and `’` XML entities. Example:

```
Terminate an SQL statement with a &lsquo;<literal>;</literal>&rsquo; character.
```

That example will look like this:

```
Terminate an SQL statement with a “;” character.
```

Curly single quotation marks is a pretty common convention with O'Reilly and Pearson.

Single quotation marks (`'`) and **backticks** (`` ` ``) are always entered verbatim.

http://docbook.org/tdg/en/html/quote.html

5.34. remark

Used for a remark (or comment) intended for presentation in a draft manuscript.

```
<remark>[SH] This is a remark.</remark>
```

Remarks may be placed almost everywhere; for a complete list of elements that may contain remarks, see http://docbook.org/tdg/en/html/remark.html.

Remarks can be processed by XSL stylesheets in a way that they are not visible in generated output. Furthermore, it is possible to hide remarks in HTML and CHM output by means of CSS stylesheets (in HTML, the remark will still be visible in the page source code), without even processing them using XSLT. This means that there is no reason to remove remarks, unless those are not needed any more.

We start remarks with the initials of the author, put in square brackets. If you are commenting on a remark, you should not use another remark, but rather do it like this: `<remark>[SH] To be, or not to be? [WS] That is the question.</remark>`

http://docbook.org/tdg/en/html/remark.html

5.35. replaceable

Used for content that may or must be replaced by the user.

```
SELECT * FROM <replaceable>tbl_name</replaceable>
```

Here, the user is expected to replace *`tbl_name`* with the name of an actual table when issuing the `SELECT` statement. Pearson notes: All placeholders should be in mono and italics. A placeholder is pseudo text you put in code (a file name, or parameter) where the reader will need to fill in their actual info to complete.

http://docbook.org/tdg/en/html/replaceable.html

5.36. row

Used for rows in tables.

http://docbook.org/tdg/en/html/row.html

5.37. secondary

Used to mark secondary index entries.

See also primary and indexterm (for an example).

http://docbook.org/tdg/en/html/secondary.html

5.38. tbody

Used to mark the table body in tables.

http://docbook.org/tdg/en/html/tbody.html

5.39. tgroup

Used to mark column groups in tables.

http://docbook.org/tdg/en/html/tgroup.html

5.40. textobject

See Section 5.23, “mediaobject”.

5.41. title

Used for the text of the title of a formal block-level element (for example `book`, `chapter`, or `figure`).

```
<section id="my-section"><title>The Title</title> ...
```

http://docbook.org/tdg/en/html/title.html

5.42. ulink

Used for referencing resources external to the document, particularly for URLs.

```
My favorite literature is the <ulink url="http://dev.mysql.com/doc/mysql/
en">MySQL Reference Manual</ulink>.
```

See also Chapter 3, *Links (or References) Within a Document*.

http://docbook.org/tdg/en/html/ulink.html

5.43. userinput

Used for data entered by the user.

```
At the system prompt, enter <userinput>xyzzy</userinput> to gain access to the
system.
```

http://docbook.org/tdg/en/html/userinput.html

5.44. xref

Used to refer to other resources. See Chapter 3, *Links (or References) Within a Document* for information on how to use this element.

http://docbook.org/tdg/en/html/xref.html

Chapter 6. File Names and Guidelines for Graphics

Graphics should have a file name that relates to the chapter or section where the graphic appears. At least, they should contain the name of the document (the `<book>`), like in this example:

```
<graphic fileref="cygwin-packetmanager.png" format="PNG" lang="en"/>
```

If the document is likely to become translated, and if we need different graphics for each individual language, then the file name should also include the language, like in this example:

```
<graphic fileref="cygwin-packetmanager.en.png" format="PNG" lang="en"/>
```

Note that the language code is prefixed using dots (`.en`) before the file extension (`.png` in this example). The following two-character ISO language codes should be used:

- `ar`: Arabic
- `de`: German
- `en`: English/American
- `es`: Spanish
- `fr`: French
- `hi`: Hindi
- `it`: Italian
- `ja`: Japanese
- `pt`: Portuguese
- `ru`: Russian
- `zh`: Chinese

For a complete list, see the ISO 639 2-letter codes.

Figures must be 640 by 480 resolution, 256 color or 24-bit color, pcx format. This is a Pearson requirement.