

Toward Single-Source Publishing Using DocBook XML

Dan York  
Director of Training  
Mitel Networks Corporation  
Network Server Solutions Group  
(formerly e-smith, inc.)

November 8, 2001

XML Users Group of Ottawa  
Ottawa, Ontario, Canada

Copyright © 2001 Dan York

# Agenda

k

- **Introduction**
- **Examples**
- **Stylesheets**
- **Tools**
- **Extensions**
- **Resources**

▪

## **Introduction**

# Chasing the Holy Grail

k

Documentation staff have always been seeking a system that...

- **Uses a *single source file* that can be exported into *multiple output formats***
- **Allows the easy automation of production**
- **Allows for the easy customization of formatting**

## What is DocBook?

k

- **A system for writing structured documents using SGML or XML**
- **Targeted at documents about computer technology, but can be used for other subjects**
- **Also other uses such as web sites and slides (like these!)**
- **Supported by a growing number of commercial and free software/open source tools**
- **Supported by a large community of DocBook users and developers**

## DocBook History

k

- **Created around 1991 originally by HaL Computer Systems and O'Reilly & Associates**
- **Later developed by a wide range of companies including:**
  - **Novell**
  - **Digital Equipment Corporation**
  - **Hewlett Packard**
  - **Sun Microsystems**
  - **SCO**

**Also Fujitsu, Hitachi, Unisys and many others**

- **Now maintained by OASIS**

## Who uses DocBook?

k

Beyond the companies mentioned previously, DocBook is widely used within the free software/open source community, including by the following organizations and companies:

- **Linux Documentation Project**
- **FreeBSD Documentation Project**
- **GNOME Documentation Project**
- **KDE Documentation Project**
  
- **Caldera Systems**
- **Mandrakesoft**
- **Red Hat**

- **SuSE**

## DocBook Strengths

k

- **Strong user/developer community**
- **Easily customizable and extensible "standard" stylesheets**
- **Open source and commercial tools available**
- **Output to multiple formats, including:**
  - **HTML**
  - **HTML Help**
  - **Java Help**
  - **man pages (UNIX/Linux manual pages)**
  - **MIF (FrameMaker)**

- **PDF**
- **PostScript**
- **RTF (Microsoft)**
- **TeX**
- **Texinfo pages**
- **XHTML**
- **XSL Formatting Objects (FO)**

## DocBook Weaknesses

k

- **Number of elements can be intimidating.**
- **Learning curve for newcomers not used to tagged languages can be intimidating.**
- **Setting up some of the open source tools can be intimidating.**
- **Some of the open source tools do not provide the desired level of control or maturity. (ex. XML PDF generation)**

▪

## Examples

# Example - Book

k

```
<book>
  <bookinfo>
    <title>Sample Book</title>
  </bookinfo>
  <chapter>
    <title>First Chapter</title>
    <sect1>
      <title>First Section</title>
      <para>This is a paragraph.</para>
    </sect1>
    <sect1>...</sect1>
  </chapter>
  <chapter>...</chapter>
  <chapter>...</chapter>
  <chapter>...</chapter>
  <appendix>...</appendix>
  <appendix>...</appendix>
</book>
```

## Example - Article

k

```
<article>
  <articleinfo>
    <title>Sample Article</title>
  </articleinfo>
  <para>This is an introductory paragraph.</para>
  <sect1>
    <title>First Section</title>
    <para>This is a paragraph.</para>
    <sect2>
      <title>This is a section level 2</title>
      <para>Blah, blah, blah... </para>
    </sect2>
    <sect2>...</sect2>
    <sect2>...</sect2>
  </sect1>
  <sect1>...</sect1>
  <sect1>...</sect1>
  <sect1>...</sect1>
</article>
```

# Sections

k

- DocBook contains sections that can be nested.
- Either of two styles, specific (<sect1-5>):

```
<sect1>
  <sect2>
    <sect3>...</sect3>
    <sect3>
      <sect4>
        <sect5>...</sect5>
        <sect5>...</sect5>
      </sect4>
      <sect4>...</sect4>
    </sect3>
  </sect2>
  <sect2>...</sect2>
</sect1>
<sect1>...</sect1>
<sect1>...</sect1>
```

## Sections - continued

k

- **Generic (nested <section> tags):**

```
<section>
  <section>
    <section>...</section>
    <section>
      <section>
        <section>...</section>
        <section>...</section>
      </section>
      <section>...</section>
    </section>
  </section>
  <section>...</section>
</section>
<section>...</section>
<section>...</section>
```

- **No theoretical limit on depth of nesting.**

## List - Itemized

k

- **Provides a bulleted list and maps to a <ul> in HTML.**

```
<itemizedlist>  
<listitem><para>Item A</para></listitem>  
<listitem><para>Item B</para></listitem>  
<listitem><para>Item C</para></listitem>  
</itemizedlist>
```

## List - Ordered

k

- **Provides a numbered list and maps to a <ol> in HTML.**

```
<orderedlist>  
<listitem><para>Item A</para></listitem>  
<listitem><para>Item B</para></listitem>  
<listitem><para>Item C</para></listitem>  
</orderedlist>
```

## List - Variable

k

- **Provides a list of terms and definitions. Maps to a <DL> in HTML.**

```
<variablelist>
<title>Title of lists</title>
  <varlistentry>
    <term></term>
    <listitem><para></para></listitem>
  </varlistentry>
  <varlistentry>
    <term></term>
    <listitem><para></para></listitem>
  </varlistentry>
</variablelist>
```

## Image - Simple

k

- **An image is usually part of a larger <mediaobject> tag. The simple form is:**

```
<mediaobject>
<imageobject>
  <imagedata fileref="filename.jpg" format="JPG"/>
</imageobject>
</mediaobject>
```

## Image - Complex

k

- **A more complex form allows for multiple image formats and also alternative text:**

```
<mediaobject>
<imageobject>
<imagedata fileref="logo.eps" format="EPS" scale="50" scalefit="1"/>
</imageobject>
<imageobject>
<imagedata fileref="logo.jpg" format="JPG"/>
</imageobject>
<textobject>
<phrase>Our company logo</phrase>
</textobject>
</mediaobject>
```

# Tables

k

- **Tables can be created in DocBook:**

```
<table>
<title></title>
<tgroup cols="2">
  <thead>
    <row>
      <entry></entry>
      <entry></entry>
    </row>
  </thead>
  <tbody>
    <row>
      <entry></entry>
      <entry></entry>
    </row>
    <row>
      <entry></entry>
      <entry></entry>
    </row>
  </tbody>
</tgroup>
</table>
```

- **Also an `<informaltable>` tag for tables that do not need a title. Additionally, informal tables will not appear in a generated list of tables.**

# Figures and Examples

k

- `<figure>` and `<example>` tag are included and will be numbered and have titles.
- `<informalfigure>` and `<informalexample>` will not be numbered.
- **Examples:**

```
<figure>
<title>Revenues for Q1</title>
<mediaobject>
  <imageobject>
    <imagedata fileref="qlrevenue.jpg" format="JPG"/>
  </imageobject>
</mediaobject>
</figure>
```

```
<example>
<title>Sample python code</title>
<programlisting>
  print "Hello, world!"
</programlisting>
</example>
```

# Admonitions

k

- **Several forms of "admonitions" are available.**

- **"Note":**

```
<note><para>This is a note</para></note>
```

**Note**

**This is a note**

- **"Warning":**

```
<warning><para>This is a warning</para></warning>
```

**Warning**

**This is a warning**

- **"Tip":**

```
<tip><para>This is a tip</para></tip>
```

**Tip**

**This is a tip**

- **"Caution":**

```
<caution><para>This is a caution</para></caution>
```

**Caution**

**This is a caution**

- **"Important":**

```
<important><para>This is important</para></important>
```

**Important**

**This is important**

## Character Formatting

k

- `<emphasis>This is some text.</emphasis>`
- `<emphasis role="bold">This is some bold text.</emphasis>`
- `<filename>`
- `<command>`
- Many others

## Verbatim Text

k

- `<programlisting>`
- `<screen>`
- `<literallayout>`

**Can use `<![CDATA[ ] ]>` to include XML/HTML code, as in:**

```
<![CDATA[<listitem><para>This is a list item.</para></listitem>]]>
```

# Links

k

- **Based on id's found in other elements**
- **<ulink url="http://www.docbook.org/">Docbook page</ulink>**
- **<link linkend="idname">Internal link</link>**
- **<xref linkend="idname"/>**
- **Example:**

```
<sect1 id="introduction">  
...  
<para>See the <link linkend="introduction">intro section</link>  
for more info.</para>
```

# Indexing

k

- **Allows creation of indices.**
- **<indexterm> takes a primary and secondary terms**
- **Example:**

```
One XSLT processing command is "xsltproc"  
<indexterm><primary>xsltproc</primary><secondary>XSLT  
processing</secondary><indexterm>, which is part of "libxslt".
```

## Note

**It is also possible to create glossaries and bibliographies**

## Miscellaneous

k

- **<sidebar>**
- **<blockquote>**
- **<footnote>**
- **Many, many, many others...**

**<http://www.docbook.org/tdg/en/html/ref-elements.html>**  
**[<http://www.docbook.org/tdg/en/html/ref-elements.html>]**

## Example - Q&A

k

- **DocBook can also be used for question and answer sets:**

```
<?xml version="1.0"?>
<!DOCTYPE qandaset PUBLIC "-//OASIS//DTD DocBook XML V4.1.2//EN"
    "http://www.oasis-open.org/docbook/xml/4.1.2/docbookx.dtd">

<qandaset>

<qandaentry>
<question><para>What is the answer to the ultimate question?</para></question>
<answer><para>42</para></answer>
</qandaentry>

<qandaentry>
<question><para>What kind of stylesheets are available?</para></question>
<answer><para>XSLT or DSSSL</para></answer>
</qandaentry>

</qandaset>
```

## Example - Reference Page

k

- Can be used for items such as Linux/UNIX man pages.
- Example:

```
<!DOCTYPE refentry PUBLIC "-//OASIS//DTD DocBook XML V4.1.2//EN"
    "http://www.oasis-open.org/docbook/xml/4.1.2/docbookx.dtd">
<refentry>
<refmeta>
<refentrytitle>ls</refentrytitle>
<manvolnum>1</manvolnum>
</refmeta>

<refsynopsisdiv><title>Synopsis</title>
<synopsis>
ls (options)... (file)...
</synopsis>
</refsynopsisdiv>

<refsect1><title>Description</title>
<para>List information about the files...
</para>
<!--...-->
</refsect1>
<!--...-->
</refentry>
```

▪

## **Stylesheets**

## Standard stylesheets

k

- "Standard" set of "modular" stylesheets maintained primarily by Norman Walsh.
- Two sets of stylesheets:
  - XSL
  - DSSSL
- Available from <http://docbook.sourceforge.net/>  
[<http://docbook.sourceforge.net/>]

## XSL Stylesheets - Output

k

- **XSLT stylesheets can transform DocBook XML into the following formats:**
  - **HTML**
  - **HTML Help (for Windows Help)**
  - **Java Help**
  - **XHTML**
  - **XSL Formatting Objects (FO)**
- **Other tools can take FO into PDF, PostScript, etc.**

## DSSSL Stylesheets - Output

k

- **DSSSL stylesheets can transform DocBook SGML into the following formats:**
  - **HTML**
  - **MIF**
  - **RTF**
  - **TeX**
- **Toolchain typically goes from TeX into PDF, Postscript, etc.**

**Although intended to work with SGML, the openjade toolchain also supports the use of XML.**

# Customization Layers

k

- **Main strength of standard stylesheets is that they are easily customizable.**
- **Parameters found in `params.xml` and (for HTML) `chunk.xml`.**
- **Example:**

```
<?xml version="1.0"?>
<!-- HTML Stylesheet for Mitel NSSG DocBook XML documents -->

<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
                version="1.0">

<xsl:import href="/usr/share/sgml/docbook/docbook-xsl-1.45/html/docbook.xml"/>

<xsl:param name="chapter.autolabel" select="1"/>
<xsl:param name="section.autolabel" select="1"/>
<xsl:param name="section.label.includes.component.label" select="1" doc:type="boolean"/>

<!-- Insert more parameters here. -->
</xsl:stylesheet>
```

## Example - Generating HTML

k

- **Separate XSLT stylesheets for processing a single file and for generating multiple files (chunking).**

- **Example of generating a single file:**

```
xsltproc -o outfile.html /usr/share/sgml/docbook-xsl/html/docbook.xsl input.xml
```

- **Example of generating multiple files:**

```
xsltproc /usr/share/sgml/docbook-xsl/html/docbook.xsl input.xml
```

## Example - Generating PDF

k

- **Generally a two-stage process:**

1. **Generation of FO from XML**
2. **Generation of PDF from FO**

- **Example:**

```
xsltproc -o manual.fo /usr/share/sgml/docbook-xsl/fo/docbook.xsl manual.xml  
Fop -fo manual.fo -pdf manual.pdf
```

- **Example of generating multiple files:**

```
xsltproc /usr/share/sgml/docbook-xsl/html/docbook.xsl input.xml
```

▪

## Tools

## Editing

k

**Emacs** Emacs with psgml  
with [http://www.lysator.liu.se/~lenst/about\_psgml/] mode for en-  
psgml tering elements and performing validation

**vim** Vim [http://www.vim.org/] with macros  
with [http://www.lodestar2.com/software/docbook/] to speed entry  
macros

**epcEdit** http://www.epcedit.com/ [http://www.epcedit.com/] - commer-  
cial editor for both Windows and Linux

**ArborText** http://www.arbortext.com/html/products.html  
**Epic** [http://www.arbortext.com/html/products.html] - comprehen-  
sive suite of both editing and processing tools

**SoftQuad** http://www.softquad.com/ [http://www.softquad.com/] - XML  
**XMetal** editor

# Validation

k

- Most XSL processors include validation, but there are specific tools.
- `xmllint` - part of libxml [<http://www.xmlsoft.org/>]

```
xmllint --valid --noout file.xml
```

## HTML - Saxon

k

- <http://saxon.sourceforge.net/> [<http://saxon.sourceforge.net/>]
- Java-based, also "Instant" versions for Windows
- Developed by Michael Kay
- Incorporates experimental XSLT features

```
java com.icl.saxon.StyleSheet inputfile stylesheet
```

## HTML - Xalan

k

- <http://xml.apache.org/> [<http://xml.apache.org/>]
- Java-based (also a C++ version)
- Developed by Apache Project
- **Widely used in open source community**

```
java org.apache.xml.xalan.xslt.Process -IN inputfile -XSL stylesheet -OUT outfile
```

## HTML - libxslt

k

- <http://xmlsoft.org/XSLT/> [<http://xmlsoft.org/XSLT/>]
- C program. Binaries available for Linux and Windows.
- Developed by Daniel Veillard from Red Hat, originally for GNOME Project
- Multiple commands:
  - `xsltproc` - XSLT processing
  - `xmllint` - XML validation and formatting
- ***Extremely fast***  
`xsltproc -o outputfile stylesheet inputfile`

# Print - FOP

k

- <http://xml.apache.org/fop/> [<http://xml.apache.org/fop/>]
- Java-based
- Developed by Apache Project
- Current version outputs to PDF, MIF, text and PCL.
- **Examples:**  
`Fop -fo inputfile -pdf outputfile`  
`Fop -fo manual.fo -pdf manual.pdf`
- **Can be used in conjunction with Xalan to directly process files:**  
`Fop -xsl mystyles.xsl -xml manual.xml -pdf manual.pdf`

## Print - PassiveTeX

k

- <http://users.ox.ac.uk/~rahtz/passivetex/>  
[<http://users.ox.ac.uk/~rahtz/passivetex/>]
- Series of TeX macros
- Developed by Sebastian Rahtz
- Current version outputs to PDF or TeX.
- **Examples:**  
`pdfxmltex FOinputfile`  
`pdfxmltex manual.fo`
- You actually run the command twice on the same FO file to resolve all references.
- Installation can be a bit tricky, but output is regarded as high quality.



## SGML Tools

k

- **While designed for SGML, openjade and jade can also work with XML.**

```
openjade -t sgml -i html -d DSSSL-stylesheet openjadepath/pubtext/xml.dcl filename.xml
```

- **More info at:**
  - **<http://openjade.sourceforge.net/>**  
**[<http://openjade.sourceforge.net/>]**
  - **<http://www.jclark.com/jade/>** [**<http://www.jclark.com/jade/>**]

## SGML Tools, cont'd

k

Other tools provide wrappers to openjade/jade, including:

**sgmltools-lite** <http://sgmltools-lite.sourceforge.net/>  
[<http://sgmltools-lite.sourceforge.net/>]

**sgml2x** <http://sgml2x.sourceforge.net/>  
[<http://sgml2x.sourceforge.net/>]

## Commercial Tools

k

**DocParse** [http://www.commandprompt.com/products\\_DocParse.lxp](http://www.commandprompt.com/products_DocParse.lxp)  
[[http://www.commandprompt.com/products\\_DocParse.lxp](http://www.commandprompt.com/products_DocParse.lxp)] -  
converts HTML into DocBook

**ArborText** <http://www.arbortext.com/html/products.html>  
**Epic** [<http://www.arbortext.com/html/products.html>] - comprehen-  
sive suite of both editing and processing tools

**RenderX** <http://www.renderx.com/products.html>  
**XEP** [<http://www.renderx.com/products.html>] - a FO to PDF render-  
ing engine

## Other Tools

k

- xt**      <http://www.jclark.com/xml/xt.html>  
[<http://www.jclark.com/xml/xt.html>] - one of original XSLT processors. Less frequently used now.
- docbook2x**      <http://docbook2x.sourceforge.net/>  
[<http://docbook2x.sourceforge.net/>] - converts DocBook to man and Texinfo pages
- refdb**      <http://refdb.sourceforge.net/> [<http://refdb.sourceforge.net/>] - creates reference databases and bibliographies from DocBook

More tools can be found at:

- <http://www.freshmeat.net/> [<http://www.freshmeat.net/>]

- <http://www.sourceforge.net/> [<http://www.sourceforge.net/>]

# Automation

k

- **Processing can be automated through using Makefiles.**
- **Example:**

```
TARGET = installchecklist

XMLFILE = ${TARGET}.xml
HTMLFILE = ${TARGET}.html
FOFILE = ${TARGET}.fo
PDFFILE = ${TARGET}.pdf

FOFILESHEET = installconfig.xsl
HTMLSTYLESHEET = /usr/share/sgml/docbook/e-smith-html.xsl

all:      html pdf

html:
    xsltproc -o ${HTMLFILE} ${HTMLSTYLESHEET} ${XMLFILE}

pdf:
    xsltproc -o ${FOFILE} ${STYLESHEET} ${XMLFILE}
    Fop -fo ${FOFILE} ${PDFFILE}
```

- **Now make all will generate both HTML and PDF.**

# Version Control

k

- **Because XML files are all text based, the Concurrent Versions System (CVS) works extremely well.**
- **<http://www.cvshome.org/> [<http://www.cvshome.org/>]**
- **<http://www.lodestar2.com/people/dyork/talks/cvs/> [<http://www.lodestar2.com/people/dyork/talks/cvs/>]**

▪

## **Extensions**

# Slides Doctype

k

- **Creation of presentation slides from DocBook XML**
- **Can create HTML (with or without frames) and FO**
- **Uses DocBook elements within specific heirarchical framework.**
- **Formatting controlled through CSS file.**
- **Downloadable from DocBook Open Repository (SourceForge).**
- **Example:**

```
<!DOCTYPE slides SYSTEM "/usr/share/sgml/docbook/xsl-slides-1.1/slides.dtd">
<slides>
<slidesinfo>
  <title>Introduction to DocBook</title>
</slidesinfo>

<foil><title>My first slide</title>
<itemizedlist>
<listitem><para>...</para></listitem>
<listitem><para>...</para></listitem>
<listitem><para>...</para></listitem>
</itemizedlist>
```

```
</foil>  
<foil><title>My second slide</title>  
<para>... </para>  
</foil>  
</slides>
```

# Website Doctype

k

- **Creation of web sites from a collection of DocBook XML files**
- **Uses most DocBook elements within specific framework. Has separate files that control page navigation and heirarchy.**
- **Downloadable from DocBook Open Repository (SourceForge).**
- **See nwalsh.com [<http://nwalsh.com/>] as an example.**
- **Example of main file:**

```
<!DOCTYPE webpage SYSTEM "../website.dtd" [  
<!NOTATION XML SYSTEM "xml">  
<!ENTITY test1a SYSTEM "test1a.xml" NDATA XML>  
<!ENTITY test3 SYSTEM "test3.xml" NDATA XML>  
<!ENTITY about.xml SYSTEM "about.xml" NDATA XML>  
>  
<webpage id="home">  
<config param="desc" value="The Test Home Page"/>  
<config param="rcsdate" value="$Date: 2001/11/08 20:44:20 $" />  
<config param="footer" value="about.html" altval="About..." />  
<head>  
<title>Welcome to Website</title>  
<summary>Introduction</summary>  
<keywords>Norman Walsh, DSSSL, SGML, XML, DocBook, Website</keywords>  
</head>
```

```
<para> This small, somewhat contrived website demonstrates the
Website document type. Website provides a system for building static
Websites from XML content.</para>
```

```
<para>A <ulink url="txtindex.html">text-only</ulink> version is also
available, demonstrating how multiple presentations can be derived from
the same sources.</para>
```

```
<webtoc/>
```

```
<section><title>What is a Website?</title>
```

```
<para>A website is a collection of pages organized, for the purposes of
navigation, into one or more hierarchies. In Website, each page is a
separate XML document authored according to the Website DTD, a customization
of <ulink url="http://www.oasis-open.org/docbook/">DocBook</ulink>.</para>
```

```
</section>
```

```
</webpage>
```

- 

## **Conclusion**

## So have we reached the Holy Grail?

k

**Yes, if you are willing to:**

- **Sacrifice some formatting control**
- **Experience complexity in setting up tools**

**No, if you:**

- **Need precise formatting control**
- **Do not have the technical expertise to set up the tools**

## Resources

k

**OASIS DocBook home page (also mailing lists):**

- <http://www.oasis-open.org/docbook/>  
[<http://www.oasis-open.org/docbook/>]

**"*DocBook: The Definitive Guide*" home page:**

- <http://www.docbook.org/> [<http://www.docbook.org/>]

**DocBook Open Repository:**

- <http://docbook.sourceforge.net/> [<http://docbook.sourceforge.net/>]

**Books:**

- ***DocBook: The Definitive Guide*, by Norman Walsh and Leonard Muellner, published by O'Reilly & Associates, October 1999 (new version available online)**

## Resources, cont'd

k

### Linux Documentation Project Author Guide:

- <http://www.linuxdoc.org/LDP/LDP-Author-Guide/>  
[<http://www.linuxdoc.org/LDP/LDP-Author-Guide/>]

### Norman Walsh's documentation pages:

- <http://www.nwalsh.com/docs/> [<http://www.nwalsh.com/docs/>]

### Mitel NSSG documentation process:

- <http://www.e-smith.org/docs/docprocess.html>  
[<http://www.e-smith.org/docs/docprocess.html>]

## Questions?

k

### ***Contact Info:***

**Dan York, Director of Training**

**Mitel Networks Corporation Network Server Solutions Group**

- **dan\_york@mitel.com**
- **dyork@Lodestar2.com (personal address)**
- **+1-613-263-4312**

**This presentation will be available from:**

- **<http://www.lodestar2.com/people/dyork/talks/docbook/>  
[<http://www.lodestar2.com/people/dyork/talks/docbook/>]**