

A Gentle Introduction to XSL

Sarah O'Keefe
Scriptorium Publishing
www.scriptorium.com



Doc FRAME

Why XSL is like a medical procedure

- “This won’t hurt a bit.”
- “You may feel some slight discomfort.”



Doc FRAME

Agenda

- About the presenter
- What is XSL?
- What do I need to work with XSL?
- XSL basics
- XSL commands



Doc FRAME

About the presenter

- Consultant
- Coauthor, *FrameMaker 7: The Complete Reference*, *FrameMaker for Dummies*, and other books
- Founder, Scriptorium Publishing



Doc FRAME

What is XSL?

- Extensible Stylesheet Language
- World Wide Web Consortium specification (w3c.org)
- Open standard
- Transforms XML into other formats



Doc FRAME

XSL versus XSLt

- XSL for browser presentation
- XSLt to create new files
- XSL used generically in this session



Doc FRAME

What do I need to work with XSL?

- Text editor or XML/XSL editor
 - Our choice: Marrowsoft Xslerator
- XSL parser/processor
 - Our choice: Saxon
- XML source files
- Output specifications



Doc FRAME

XSL basics

- Tree view of XML file
- Processing through the tree
- Procedural versus declarative



Doc FRAME

My first XSL commands

- `<xsl:template>`
- `<xsl:value-of>`
- `<xsl:apply-templates>`



Doc FRAME

My first XSL template

Source `<Para>This is a paragraph.</Para>`

Transformation

```
<xsl:template match="Para">
  <p>
    <xsl:apply-templates/>
  </p>
</xsl:template>
```

Result `<p>This is a paragraph.</p>`

Testing the transformation

`saxon file.xml transform.xsl`



Doc FRAME

Transformation components

- Paragraphs (block elements)
- Text ranges (inline elements)
- Output files
- Cross-references
- Graphics
- Navigation



Doc FRAME

Paragraph transformation

- Search and replace
- Straightforward
- Often more layers in XML than output



Doc FRAME

Text ranges

- Similar to paragraph transformation



Doc FRAME

Managing output files

- Default is one output file.
- Saxon can specify one output file name.
- `<xsl:document>` for multiple output files (XSL 1.1).



Doc FRAME

Commands for output files

- `<xsl:document>`
- `generate-id` function



Doc FRAME

Cross-references

- Must convert reference into specific file name
- Must pick up text for hyperlink from somewhere



Doc FRAME

XSL concepts for cross-references

- id function
- Complex ancestor references
 - ancestor-or-self::Section[1]



Doc FRAME

Graphics

- Graphic files not affected by XSL
- Can transform graphic references into standard hyperlinks



Doc FRAME

XSL concepts for graphics

- unparsed-entity-uri function
- Recursive template



Doc FRAME

Recursion

- A recursive template is a template that calls itself.
- Used to process complex content.
- Each time the template runs, a small chunk of content is separated out.
- Very common in XSL.



Doc FRAME

Navigation

- Table of contents
- Previous/Next links
- Index



Doc FRAME

XSL concepts for navigation

- `<xsl:for-each>` loops
- Axis references
- `<xsl:choose>` statements



Doc FRAME

Table of contents

- Generate based on XML structure
- Create links



Doc FRAME

Previous/Next links

- Link based on structure
- Must calculate for many different scenarios



Doc FRAME

Index

- Complex transformation
- Process index entries
- Sort index entries
- Group index entries
- Process “see” and “see also” references



Doc FRAME

Buy this book

- *XSLT Programmer's Reference*
 - Michael Kay
 - Wrox Press
 - ISBN: 1861005067



Doc FRAME

Software used in this demo

- Xslerator: www.marrowsoft.com
- TextPad: www.textpad.com
- Saxon: saxon.sourceforge.net
- DocFrame: www.docframe.com



Doc FRAME

White papers

- XML overview:
www.scriptorium.com/structure.pdf
- Structure implementation:
www.scriptorium.com/str_implementation.pdf



Doc FRAME

Q & A



Doc FRAME

Contact information

Scriptorium Publishing
Research Triangle Park, NC, USA
www.scriptorium.com
sales@scriptorium.com
919-481-2701 x105



Doc FRAME