Introduction to XSL Formatting Objects (FO) (for SVDIG)

Simon Bate
Scriptorium Publishing

About me

- Simon Bate
- Sr. Technical Consultant at Scriptorium
- 30+ years publications experience
- Certified Technical Trainer (CompTIA CTT+)
- 4 years of DITA and DITA OT experience

About Scriptorium

- Content Strategy for Technical Communication

- Analyze
- Propose
- Implement
- Transform
- Train

Overview

- What is XSL-FO?
- Basic structure
- Pages and areas
- Attribute sets
- Modifying the PDF2 plugin

What XSL-FO?

- W3C recommendation for XML formatting
  - XSLT (Transformation)
  - XSL-FO (Formatting Objects)
- XML dialect
- Versions
  - 1.0 – 2001
  - 1.1 - 2006
- Uses “fo” namespace (<fo:root>)

Overview

- Use XSL to transform XML to XSL-FO
- Use FO processor to generate PDF
  - Apache FOP (free)
  - Antenna House
  - RenderX XEP
- Generates pages for all reading orders, e.g.:
  - Lines left-to-right, top-to-bottom
  - Lines top-to-bottom, right-to-left

Basic structure

```
<fo:root xmlns:fo=”...”>
  <fo:layout-master-set>
    <fo:simple-page-master master-name=”A4”>
      <fo:region-body region-name=”body”>
        <!-- Define region -->
      </fo:region-body>
      <!-- define other regions -->
    </fo:simple-page-master>
  </fo:layout-master-set>

  <fo:page-sequence master-reference=”A4”>
    <fo:flow flow-name=”body”>
       <!-- Content -->
    </fo:flow>
    <!-- static content -->
  </fo:page-sequence>
  <fo:page-sequence ... />
</fo:root>
```

Pages and areas

- Browsers can flow text without limits
- Pages have physical limits
  - Orientation
  - Reading order
- Above and below text
  - region-before/region-after
- Sidebars
  - region-start/region-end

Page sequences

- A page sequence defines one or more pages
  - Think “chapter”
- The page master defines the layout
- Content is placed in flows
  - Defined by <fo:flow>
  - Generates multiple pages
- A page sequence can have multiple flows
  - xsl-region-body
  - xsl-region-before, -after, -start, -end

Flows and static content

- Page sequences can contain
  - Flows - Information that flows from page to page
  - Static content - is the same from page to page
    - Think running heads, folios

-
  ```
  <fo:page-sequence>
    <fo:static-content flow-name=”xsl-region-before”>
      <fo:block>...</fo:block>
    </fo:static-content>
    <fo:flow flow-name=”body”>
      <fo:block>...</fo:block>
    </fo:flow>
  </fo:page-sequence>
  ```

Text areas

- Block areas
  - Paragraphs, tables, lists
  - <fo:block>
- Line areas
  - Blocks consist of one or more text lines
- Inline areas
  - words, graphics
  - <fo:inline>

Blocks

- A flow consists of multiple, nested blocks
- Blocks can define
  - Space before and after
  - Margins
  - Borders
  - Padding
  - Fonts
  - Alignment

Example block

```
<fo:block line-height="50pt" font-size="24pt"
     font-family="Helvetica">
  <fo:block font-style="italic">
    <xsl:value-of
      select="*[contains(@class, ' topic/title ')]"/>
  </fo:block>
</fo:block>
```

Attribute sets

- XSL construct

```
<xsl:attribute-set name=”set-name”>
   <xsl:attribute name=”attr-name”>value</...>
```

- Include in XSL

```
<xsl:include href=”...-attrs.xsl”/>
```

- Use as part of an FO object

```
<fo:block xsl:use-attribute-sets="set-name">...
```

- Attribute set can contain xsl:use-attribute-sets
- Defined in:
  - plugins\(pdf)\cfg\fo\attrs\*-attr.xsl

Resources

- XSL Formatting Objects Developer's Handbook
  - Doug Lovell
  - Sams Publishing, 2003
- Definitive XSL-FO
  - G. Ken Holman
  - Prentice Hall, 2003
- W3Schools.com
  - XSL-FO Tutorial
  - XSL-FO Reference

Modifying the OT PDF2 transform

- Plugin to a plugin
  - PDF2 transform contains hooks
- Excellent docs at:
  - http://dita-ot.sourceforge.net/1.6/
  - http://dita-ot.sourceforge.net/1.6/readme/dita2pdf-customization.html
- Just change what you need
  - Let the PDF2 transform do the rest

Plugin structure

- cfg/catalog.xml
  - fo/attrs/custom.xsl
  - fo/xsl/custom.xsl
  - fo/font-mappings.xml
- custom.xsl could just contain templates
  - I decided to use xsl:import statements
  - Parallel XSL file structure
    - XSL files
    - Attribute set files

How the OT generates PDFs

- Preprocesses DITA and Map files
  - Conrefs
  - Xrefs
  - others
- Creates merge file
  - Map
  - Merged topics
- Post-processes merge file for indexing

How to find templates?

- Good search tool
  - Agent Ransack
- DITA OT XSL matches on @class
- Search for template match statement
- Sometimes necessary to search merge file

Finding correct attribute set

- Examine output .fo file
  - See integrator changes to copy .fo file
  - Good to reformat for readability
- Find unique attributes
  - Look in -attr files
- Sometimes need to add
  - <xsl:comment>
  - <xsl:message>

Questions