

Answers to the final examination on XSLT

Christian Rinderknecht

11 December 2006

1 Making a table of contents

The purpose is, given an XML document representing a book made of chapters and sections (whose titles are specified as attributes), to write an XSL transform that extracts the table of contents, followed by the original document where all attributes are changed into elements. For example, from the document

```
<?xml version="1.0" encoding="UTF-8"?>
<book>
  <title>The XML Book</title>
  <author>John Doe</author>
  <intro>I</intro>
  <chapter title="Alpha">
    <quote author="Smith">Foo</quote>
    <section title="V">A</section>
    <section title="W">B</section>
  </chapter>
  <chapter title="Beta">
    Foreword
    <section title="X">
      C
      <section title="Y">D</section>
      C'
    </section>
  </chapter>
  <chapter title="Gamma">
    <section title="Z">E</section>
  </chapter>
</book>
```

we want the output

```
<?xml version="1.0" encoding="UTF-8"?>
<book>
  <title>The XML Book</title>
  <author>John Doe</author>
  <toc>
    <intro/>
    <chapter>Alpha<section>V</section>
      <section>W</section>
    </chapter>
    <chapter>Beta<section>X<section>Y</section>
      </section>
    </chapter>
    <chapter>Gamma<section>Z</section>
  </chapter>
</toc>
<intro>I</intro>
<chapter>
  <title>Alpha</title>
  <quote>
    <author>Smith</author>Foo</quote>
  <section>
    <title>V</title>A</section>
  <section>
    <title>W</title>B</section>
</chapter>
<chapter>
  <title>Beta</title>
  Foreword
  <section>
    <title>X</title>
    C
  <section>
    <title>Y</title>D</section>
  C'
</section>
</chapter>
<chapter>
  <title>Gamma</title>
  <section>
    <title>Z</title>E</section>
</chapter>
</book>
```

Answer. One possible transform is

```
<?xml version="1.0" encoding="UTF-8"?>

<xsl:stylesheet version="2.0"
    xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:output method="xml" version="1.0" indent="yes"/>

<!-- Matching the document root and running two transforms -->
<xsl:template match="book">
    <xsl:copy>
        <xsl:copy-of select="title|author"/>
        <toc>
            <xsl:apply-templates select="intro|chapter" mode="toc"/>
        </toc>
        <xsl:apply-templates select="intro|chapter"/>
    </xsl:copy>
</xsl:template>

<!-- Extracting the table of contents -->
<xsl:template match="intro|chapter|section" mode="toc">
    <xsl:copy>
        <xsl:value-of select="@title"/>
        <xsl:apply-templates select="section" mode="toc"/>
    </xsl:copy>
</xsl:template>

<!-- Elements are invariant -->
<xsl:template match="*">
    <xsl:copy>
        <xsl:apply-templates select="@*" />
        <xsl:apply-templates />
    </xsl:copy>
</xsl:template>

<!-- Attributes become elements -->
<xsl:template match="@*">
    <xsl:element name="{name()}">
        <xsl:value-of select="."/>
    </xsl:element>
</xsl:template>

</xsl:stylesheet>
```