Schematron for word-processing documents

Andrew Sales

Andrew Sales Digital Publishing

XML London, 7th June 2015

Background

- Why use Word to capture XML?
  - cost
  - skills, familiarity
  - legacy workflows & content
  - dual approach: markup *and* typesetting
- Cons
  - working in unstructured environment
  - underlying markup hidden

Quality

- If you do use Word, you need (ideally):
  - consistently-applied styles
  - well-designed template
- All styled *Normal* produces sub-optimal results

Approaches

- Before OOXML/ODF: macros
- After: Schematron is possible
  - it's all XML behind the scenes
  - benefit of XML output from validation (SVRL)
  - write XPaths (XSLT, XQuery…) rather than bespoke code
  - abstraction possible
  - standards-based (including source markup!)

Types of rule: unexpected styles

"All paragraph styles in the body of the document must be a member of a controlled list of styles."

```
<pattern id="unexpected-para-style">

<let name="allowed-para-styles"
     value="('articlehead', 'bodytext', 'bibhead', 'bib')"/>

<rule context="w:p[not(parent::w:ftr)
                   and not(parent::w:footnote)
                   and not(parent::w:endnote)][w:r]">
<report
        test="not(w:pPr/w:pStyle/@w:val =
              $allowed-para-styles)">unexpected para style
              '<value-of select="w:pPr/w:pStyle/@w:val"/>';
              expected one of:
              <value-of select="$allowed-para-styles"/>
</report>
</rule>
</pattern>
```

Unexpected sequence of styles

“The first bibliographic citation must be immediately preceded by a bibliography heading.”

```
<pattern id="missing-bib-heading">
<rule
      context="w:p[w:pPr/w:pStyle/@w:val='bib']
      [not(preceding::w:p[w:pPr/w:pStyle/@w:val
      = 'bib'])]">
<assert
      test="preceding::w:p[w:pPr/w:pStyle/@w:val
            = 'bibhead']">
            no bibliography heading found
</assert>
</rule>
</pattern>
```

Format of datatypes, e.g. dates

"A date in a bibliographic citation must conform to the format YYYY-MM-DD."

```
<pattern id="bad-date">
<rule
      context="w:r[w:rPr/w:rStyle/@w:val
               ='bibdate']">

<assert test=". castable as xs:date">
text styled as 'bibdate' must be in the format
'YYYY-MM-DD';
got '<value-of select="."/>'</assert>
</rule>
</pattern>
```

Co-occurrence constraints

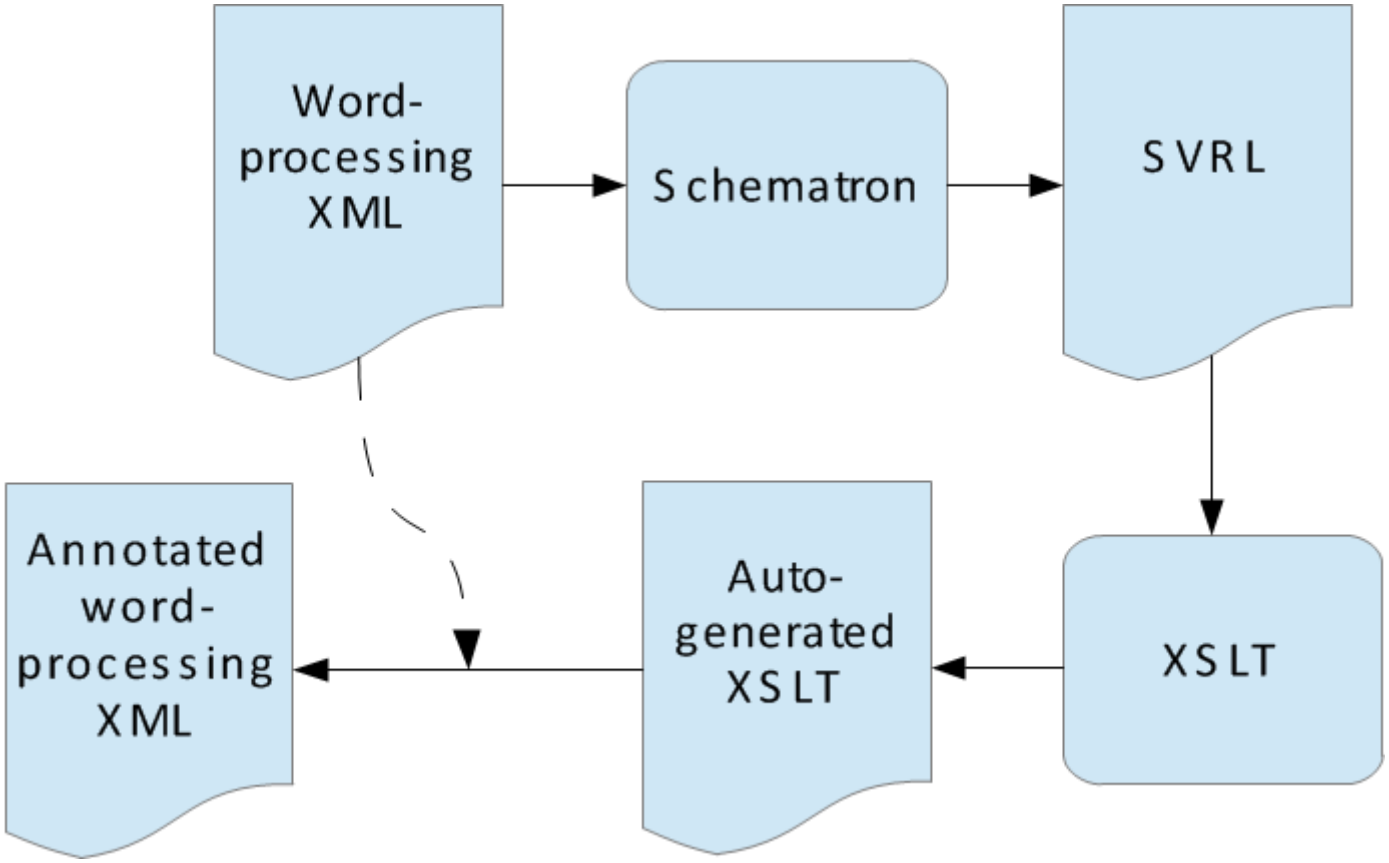
"Every citation reference must have a corresponding citation number in the bibliography."

```
<pattern id="broken-citation-link">

<let name="citation-refs"
 value="//w:r[w:rPr/w:rStyle/@w:val ='bibref']"/>

<rule context="w:r[w:rPr/w:rStyle/@w:val
                = 'bibnum']">
<assert test=". = $citation-refs">
could not find a citation reference to this
citation:
'<value-of select="."/>'</assert>
</rule>
</pattern>
```

Visualisation



Visualisation (2)

- Demo(s)…
- Errors limited to a renderable location

Simplification

- Flat structure & verbose markup mean tedious rule-writing
- Options:
  - simplify the rules
  - simplify the source
  - domain-specific language?

Simplified rules

```
<pattern id="expected-preceding-style"
     abstract="true">
<rule context="w:p[w:pPr/w:pStyle/@w:val
           = $context-style]
              [not(preceding::w:p[w:pPr/w:pStyle/@w:val
               = $context-style])]">
<assert test="preceding::w:p
             [w:pPr/w:pStyle/@w:val
             = $expected-preceding-style]">
first occurrence of style '<value-of select="$context-style"/>'
has no preceding style '<value-of select="$expected-preceding-
style"/>'
</assert>
</rule>
</pattern>
```

```
<pattern id="missing-bib-heading"
         is-a="expected-preceding-style">
<param name="context-style" value="'bib'"/>
<param name="expected-preceding-style"
       value="'bibhead'"/>
</pattern>
```

Simplified source

```
<doc>
<sect>
<p style="articlehead">The application of Schematron schemas to word-
processing
documents</p>
<p style="bodytext">As traditional print-based publishing has made the
transition into the digital age, a convention has developed in some
quarters of capturing or even typesetting content using word-
processing applications.</p>

<!-- lots more here... -->

<p style="heading 2">References</p>
<p style="bib"><span style="bibnum">[1]</span>
<url address="http://www.ecma-
international.org/publications/standards/Ecma-376.htm"
>http://www.ecma-international.org/publications/standards/Ecma-
376.htm</url>.
Retrieved <span style="bibdate">2015-03-08</span>.</p>
<!-- etc. -->
</sect>
</doc>
```

DSL

- More declarative, schema-like
- Can drive auto-generation of Schematron schema

Style schema

```
<Document>
  <Ref name="articlehead"/>
  <OneOrMore>
    <Ref name="bodytext"/>
  </OneOrMore>
  <Optional>
    <Group>
      <Ref name="bibhead"/>
      <OneOrMore>
        <Ref name="bib"/>
      </OneOrMore>
    </Group>
  </Optional>
</Document>
```

Other office documents

- E.g. spreadsheets
- Demo…

Conclusion

- Quality control through Schematron possible although XML may be “hidden”
- Errors can be presented in context to user in familiar environment
- Simplify: rules/source; DSL?
- Applicable to other office document types