An XML model of CSS3 as an XLaTeX-TeXML-HTML5 stylesheet language

S. Sankar, S. Mahalakshmi and L. Ganesh

Abstract

HTML5 [1] and CSS3 [2] are popular languages for Web development. However, HTML with CSS is prone to errors and difficult to port, so we propose an XML version of CSS that can be used as a standard for creating stylesheets and templates across different platforms and pagination systems. XLaTeX [3] and TeXML [4] are some examples of XML that are close in spirit to TeX that can benefit from such an approach. Modern TeX systems like XƎTeX and LuaTeX use simplified `fontspec` macros to create stylesheets and templates. We use XSLT to create mappings from this XML-stylesheet language to `fontspec`-based TeX templates and also to CSS3. We also provide user-friendly interfaces for the creation of such an XML stylesheet.

1 Comparison of OpenOffice and CSS stylesheets

Nowadays, most modern applications have implemented an XML package format including an XML implementation of stylesheets: InDesign has its own IDML [5] (InDesign Markup Language) XML package format and Microsoft Word has its own OOXML [6] format, which is another ISO standard format. As they say ironically, the nice thing about standards is that there are plenty of them to choose from. However, instead of creating one more non-standard format, we will be looking to see how we can operate closely within current standards. Below is a sample code derived from OpenOffice document format:

```
<style:style style:name="Heading_20_1"
    style:display-name="Heading 1"
    style:family="paragraph"
    style:parent-style-name="Heading"
    style:next-style-name="Text_20_body"
    style:default-outline-level="1"
    style:class="text">
+<style:font-face
    style:name="Adobe Caslon Pro Bold"
    svg:font-family="’Adobe Caslon Pro Bold’"
    style:font-family-generic="roman"
    style:font-pitch="variable" />

<style:text-properties fo:font-size="115%"
    fo:font-weight="bold"
    style:font-size-asian="115%"
    style:font-weight-asian="bold"
    style:font-size-complex="115%"
    style:font-weight-complex="bold" />
</style:style>
```

```
-<style:style style:name="Heading_20_2"
    style:display-name="Heading 2"
    style:family="paragraph"
    style:parent-style-name="Heading"
    style:next-style-name="Text_20_body"
    style:default-outline-level="2"
    style:class="text">

<style:font-face style:name="Arial"
    svg:font-family="Arial"
    style:font-family-generic="swiss"
    style:font-pitch="variable" />

<style:text-properties fo:font-size="14pt"
    fo:font-style="italic"
    fo:font-weight="bold"
    style:font-size-asian="14pt"
    style:font-style-asian="italic"
    style:font-weight-asian="bold"
    style:font-size-complex="14pt"
    style:font-style-complex="italic"
    style:font-weight-complex="bold" />
</style:style>
```

The equivalent CSS style is listed below:

```
heading1{
          font-family: Adobe Caslon Pro Bold;
          font-size:14pt;
          font-style: normal;
          font-variant: normal;
          font-weight: bold;
          line-height: 16pt;
          text-align: left;
          color: black;
          background-color: none;
          text-decoration: none;
          text-transform: normal;
        }

heading2{
          font-family: Arial;
          font-size:14pt;
          font-style: italic;
          font-variant: normal;
          font-weight: bold;
          line-height: 16pt;
          text-align: left;
          color: black;
          background-color: none;
          text-decoration: none;
          text-transform: normal;
        }
```

When comparing the above two style coding standards, the Cascading Style Sheet (CSS) is a simple and straightforward formulation without complex namespaces, additional attributes and other details which are not mandatory to form a style.

1.1 Advantages of CSS style patterns

- CSS is simple to author.
- CSS makes it possible for the entire style and layout to be abstracted out of the HTML, so the HTML has only the content.
- Different stylesheets can be used for different media without the user having to explicitly choose one; e.g., printers, desktop monitors and other smaller portable devices.
- Implementing CSS is straightforward for the HTML engines.

1.2 Disadvantages of CSS style patterns

While new additions to CSS3 provide a stronger, more robust feature-set for layout, CSS is still at heart a styling language (for fonts, colours, borders and other decoration), and not a layout language (for blocks with positions, sizes, margins, and so on). These limitations mean that creating fluid layouts generally requires hand-coding of CSS.

2 Introducing the Cascading Style Sheet Markup Language (CSSML)

SASS [7] is a meta-language on top of CSS that is used to describe the style of a document clearly and structurally, with more power than flat CSS. SASS provides a simpler, more elegant syntax for CSS and also implements various features that are useful for creating manageable stylesheets.

SASS is an extension of CSS3 and provides several useful features which can handle nested rules, common variables, etc. However, it is not an XML model and cannot be validated using a DTD/schema.

We will introduce here the Cascading Style Sheet Markup Language (CSSML), an XML version of CSS that can be used as a standard for creating stylesheets and templates across different platforms and pagination systems. It is also an extension of CSS3 to handle nested rules as in SASS, and can be validated using DTD/schema.

We also hope CSSML will eventually evolve to circumvent all the current limitations in CSS and XSL-FO, especially the rules of placement of figures and tables in multi-column layout of text. However, we want to keep the CSSML as a clean data model by not introducing a scripting language on top of it as is the case with SASS.

3 Definition of CSSML

In general, our Cascading Style Sheet Markup Language (CSSML) specifies style format details in a well-structured XML format. The style names used in CSSML are similar to CSS; the only difference is that the style names are defined as XML tag elements. Here is an example to explain the difference between CSS and CSSML style coding:

```
  CSS: font-family: Adobe Caslon Pro Bold;
CSSML: <font-family>Adobe Caslon Pro Bold
       </font-family>
```

The main advantage of the CSSML tag pattern is that we can validate the CSSML document using XML Schema or XML DTD which is not possible in CSS. We can write our own XML DTD/schema to validate the CSSML document as follows:

- Elements and attributes that must/may be included, and are permitted in the structure.
- The structure as specified by a regular expression syntax.
- How character data is to be interpreted, e.g. as a number, a date, etc.

However, creating the CSSML document is not as simple as creating CSS. CSSML needs XML tagging for all the data, and the user needs to wrap all the details with appropriate XML elements. To avoid such difficulties, we have provided a user interface to create CSSML automatically with appropriate XML elements.

4 Namespaces

In this section we compare namespaces in CSSML, CSS, and XML.

4.1 CSSML

```
<styles xmlns="http://www.tnq.co.in/CSSML">
<namespace prefix="html"
           value="http://www.w3.org/1999/xhtml"/>
<namespace prefix="tux"
           value="http://www.tnq.co.in/TUX"/>
  <html:p>
   <style><text-color>yellow</text-color></style>
  </html:p>
  <tux:p>
   <style><text-color>blue</text-color></style>
  </tux:p>
</styles>
```

4.2 CSS

```
@namespace html "http://www.w3.org/1999/xhtml";
@namespace tux  "http://www.tnq.co.in/TUX";
html|p { display: block; color: yellow; }
tux|p  { display: block; color: blue; }
```

4.3 XML

```
<?xml version="1.0"?>
<?xml-stylesheet href="xml.css"?>
<article
  xmlns:html="http://www.w3.org/1999/xhtml"
  xmlns:tux="http://www.tnq.co.in/TUX">
  <html:p>This is some text</html:p>
  <tux:p>This is another text</tux:p>
</article>
```

4.4 Sample CSSML coding

```
<styles xmlns="http://www.tnq.co.in/CSSML"
        xmlns:tux="http://www.tnq.co.in/TUX">
  <tux:section1>
   <style>
     <font-family>lmmono10-regular</font-family>
     <font-url>http://www.ctan.org/tex-archive/
         fonts/lm/fonts/opentype/public/lm/
         lmmono10-regular
     </font-url>
     <font-size unit="pt">11</font-size>
     <font-style>normal</font-style>
     <font-variant>normal</font-variant>
     <font-weight>Bold</font-weight>
     <font-face>lmmono10.otf</font-face>
     <line_height unit="pt">13.2</line_height>
   </style>
   <tux:section-label id="B12">
    <style>
      <text-indent unit="pt">6</text-indent>
      <rule-color>black</rule-color>
      <text-transform>normal</text-transform>
      <vertical-align>bottom</vertical-align>
    </style>
   </tux:section-label>
   <tux:section-title1 id="B13">
    <style>
      <text-indent unit="in">0</text-indent>
      <text-align>justify</text-align>
      <word-break>hyphenate</word-break>
      <column-span>all</column-span>
      <vertical-align>bottom</vertical-align>
    </style>
   </tux:section-title1>
  </tux:section1>
</styles>
```

5 CSSML to CSS3 conversion

CSSML provides a more elegant syntax for CSS and implements various features that are useful for creating stylesheets and LaTeX templates. CSSML allows us to use formatting, nested rules, inline imports, etc., all with CSS compatibility. XSLT is used to transform the CSSML XML to CSS format or to a LaTeX class file using the `fontspec` package.

5.1 Formatting

```
<tux:paragraph
     xmlns:tux="http://www.tnq.co.in/TUX">
 <style>
   <font-name>Times</font-name>
   <font-size unit="pt">11</font-size>
   <line_height unit="pt">13</line_height>
   <text-indent unit="pt">11</text-indent>
 </style>
</tux:paragraph>
```

```
@namespace tux "http://www.tnq.co.in/TUX";
tux|paragraph { font-family: Times;
                font-size:   11pt;
                line-height: 13pt;
                text-indent: 11pt; }
```

5.2 Nesting

```
<tux:section1>
 <style>
   <font-family>Times</font-family>
   <font-size unit="pt">11</font-size>
   <font-weight>Bold</font-weight>
   <line_height unit="pt">12</line_height>
 </style>
 <tux:section-label id="B12">
   <style>
     <text-indent unit="pt">6</text-indent>
   </style>
 </tux:section-label>
 <tux:section-title id="B13">
  <style><text-align>left</text-align></style>
 </tux:section-title>
</tux:section1>
```

```
@namespace tux "http://www.tnq.co.in/TUX";
tux|section1 { font-family: Times;
               font-size:   11pt;
               line-height: 12pt;
               font-weight: bold; }

tux|section1>tux|section-label
{ text-indent:6pt; }

tux|section1>tux|section-title
{ text-align:left; }
```

5.3 Selecting nodes

XPath is used to select nodes instead of CSS selectors. Here are some examples of CSS to XPath mappings:

CSS selectors	XPath pattern
`h1p`	`h1//p` (matches any `p` element that is a descendant of an `h1` element)
`h1>p`	`h1/p` (matches any `p` element that is a child of an element `h1`)
`p:first-child`	`*[1]/self::p` (matches element `p` when `p` is the first child of its parent)
`h1+h2`	`h1/following-sibling::*[1] /self::h2` (matches any `h1` element immediately preceded by an element `h2`)

The above table just provides a rough idea of how it is done, and by no means provides an exhaustive list of all CSS3 selectors.

6 CSSML to TeX font conversion

We use the `fontspec` package for font definitions. This package allows users of XeTeX or LuaTeX to load OpenType fonts in a LaTeX document. No font installation is necessary, and font features can be selected and used as desired throughout the document.

XeTeX and LuaTeX also allow fonts to be loaded by file name instead of font name. When you have a very large collection of fonts, you will sometimes not wish to have them all installed in your system's font directories. In this case, it is more convenient to load them from a different location on your disk.

6.1 Font declaration example

In CSSML:

```
<font-group>
 <font-family>Times</font-family>
 <font-style-1>Times CG</font-style-1>
 <font-style-2>Times-Bold</font-style-2>
 <font-style-3>Times-Italic</font-style-3>
 <font-style-4>Times-BoldItalic</font-style-4>
 <font-style-5>Times-BoldSC</font-style-5>
</font-group>
```

Using `fontspec` in LaTeX:

```
\fontspec[
  BoldFont = Times-Bold.otf,
  ItalicFont = Times-Italic.otf,
  BoldItalicFont = Times-BoldItalic.otf,
  SmallCaps = Times-BoldSC.otf,
]{Times.otf}
```

6.2 Paragraph style example

In CSSML:

```
<tux:paragraph>
 <style>
  <font-name>Times</font-name>
  <font-size unit="pt">11</font-size>
  <line_height unit="pt">13</line_height>
  <text-indent unit="pt">11</text-indent>
 </style>
</tux:paragraph>
```

Using `fontspec` in LaTeX:

```
\def\normalsize{%
 \fontsize{11}{13}%
 \fontspec{Times}%
 \paraindent=11\p@
}
```

6.3 Section heading style example

In CSSML:

```
<tux:section1>
 <style test="section1">
  <font-family>Times</font-family>
  <font-size unit="pt">11</font-size>
  <line-height unit="pt">13</line-height>
 </style>
 <tux:section-label>
  <style test="section1">
    <font-variant>Bold</font-variant>
  </style></tux:section-label>
 <tux:section-title>
  <style test="section1">
    <text-align>center</text-align>
    <margin-top unit="pt">12</margin-top>
    <margin-bottom unit="pt">6</margin-bottom>
  </style>
 </tux:section-title>
</tux:section1>
```

In LaTeX:

```
\newcommand\section{\@startsection
  {section}%
  {1}%
  {\z@}%
  {-12\p@ \@plus -2\p@ \@minus -2\p@}%
  {6\p@}%
  {\centering\fontsize{11}{13}%
   \selectfont\bfseries}%
}
```

References

[1] http://www.w3.org/TR/html5. A vocabulary and associated APIs for HTML and XHTML. W3C Working Draft, 25 May 2011.

[2] http://www.w3.org/TR/2011/REC-css3-selectors-20110929. Selectors Level 3. W3C Recommendation, 29 September 2011.

[3] http://omega.enstb.org/xlatex. A DTD/schema which is very close to LaTeX.

[4] http://getfo.org/texml. An XML syntax for TeX.

[5] http://blogs.adobe.com/indesignsdk/category/idml. IDML for representing InDesign content.

[6] http://xml.openoffice.org/general.html. OpenOffice.org XML file format.

[7] http://sass-lang.com. Syntactically Awesome Stylesheets.

◇ S. Sankar, S. Mahalakshmi and L. Ganesh
TNQ Books and Journals
Chennai, India
sankar (at) tnq dot co dot in