Harmonization of Office Document Format and Web Page Format Driven by Separating Presentation from Content

Fang Chun-yan [1, a *], Li Ning[2,b] and Hou Xia[3,c]

[1] No 1 An-Ding-Men East Street, Beijing 100007, China

[2,3] 35 Bei-Si-Huan Zhong-Lu, Beijing 100101, China

[a]fangcy@cesi.cn, [b]ningli.ok@163.com, [c]houxia@bistu.edu.cn

Keywords: Format harmonization; office document format; presentation-content separation; HTML; CSS; UOF.

Abstract. We propose a new type of style markup language to address the need for separation of content and presentation in office document formats, allow the reuse and sharing of styles in a standard way, and attain good interoperability with HTML/CSS as well as good compatibility with the current UOF specification. We begin with a review of the development of the office document format represented by UOF and the web document format HTML/CSS. We note that in spite of independent development both types of document format are moving towards separation of content and presentation, and therefore good interoperability is expected. We illustrate by example what the proposed mark-up language could look like and give an example to illustrate its use in transforming a sample HTML/CSS page into UOF. In the conclusion we note the importance of such a mark-up language for the future of the web office, or Office 2.0, which is considered by many to be the future of office applications in the age of cloud computing.

Introduction

Reflowable documents allow their presentation to adapt to various media, and so they are widely used in the daily office document processing and in electronic publishing. Some popular reflowable document format standards include: ISO/IEC standards Open Document Format or ODF (ISO 26300:2006) [2], Office Open XML or OOXML (ISO 29500:2008) [3], and Uniform Office Format or UOF (GB/T20916- 2007) [4]. HTML was originally designed for screen media. Since its invention in 1980 it has become the most widely used document format for web pages.

Office documents and web pages are the two most commonly used reflowable document types, though they have been developed along different directions [1].

Although SGML and XML both have made separating presentation from content as their mission, none of the XML based reflowable formats has completely fulfilled this task. This is due, in part, to users' preference for editors using the What You See Is What You Get (WYSIWYG) paradigm, which focus on the visual appearance of each document part according to the style of that part. Content and presentation are typically mixed together in such editors in a hybrid description, and this enables the user to see the result instantly. When content and presentation are described separately, a user may need to perform some additional operations to see what the document will actually look like on a printed page or on a web page. Today, people have realized that the WYSIWYG editing mode has many drawbacks, and editors are gradually changing to the What You See Is What You Mean (WYSIWYM) model [6], in which a user specifies the semantic categories for the various parts of the document.

Beyond all doubt [7-8], CSS has been very successful in separating presentation from content completely and in an easy and standard manner. CSS is used to specify the presentation of a document written in a markup language. CSS allows the same web page content to be presented in different styles for different rendering methods and devices, and it also allows the author and the reader to choose their preferred styles when browsing the web page. As with HTML, CSS is maintained in

W3C and has developed quickly. The first version of CSS was published in 1996. Newly developed CSS3 is already supported by many browsers.

For a long time, office document formats and HTML/CSS were developing separately. They have different application areas, and interoperability between them has been largely neglected. However, they are becoming strongly interrelated nowadays. Documents do need to be displayed on different media. Reflowable documents should be able to interoperate so that a document created by a system in one application area can be transformed into a corresponding document that can be properly handled by a system focused on a different application area. Unfortunately, at present there are few tools which can successfully transform, for example, HTML pages with CSS stylesheets into a corresponding office document and vice versa. To address this need, in late 2014 Apache started a project [9], called Corinthia, to develop a toolkit for converting between and editing common office file formats. It is designed to cater to multiple classes of platforms - desktop, web, and mobile - and relies heavily on web technologies such as HTML, CSS, and JavaScript, for representing and manipulating documents.

This paper proposes a style markup language as a standard approach to separating presentation from content in office documents and a way to make styles sharable between documents. We hope that it can be adopted and integrated into the new document format for web office, which will become increasingly important in the background of cloud computing.

STYLES IN THE WEB DOCUMENT FORMAT

HTML uses XML-like elements and attributes to describe web pages. Additionally, HTML allows multimedia objects, forms, and scripts to be embedded to provide an interactive user interface. Since its introduction in 1980, its trend has clearly been towards separation of content and presentation.

The early versions of HTML had no clear intension to separate presentation from content. Then in 1991, Berners-Lee [10] proposed 18 elements used for creating very simple web pages. In the most recent version, HTML5, the number of element exceeds 110. The HTML elements can be classified into the following categories: 1) structural markup describing the editing semantics, like h1, p, ul, etc.; 2) presentational markup describing the appearance of the text, like font, strike, u, center, b, i, etc.; and 3) hypertext markup making parts of a document into links to other documents. The addition of the structural markup clearly indicates a movement towards separation. Moreover, many presentational markup elements, e.g., b and i, have become deprecated under the HTML 4.0 and illegal in HTML5. On the other hand, the use of the attribute @class associated with user defined meanings is encouraged in the elements span and div to indicate more precise semantics. For example, <span class="author"> and <div class="invoice"> clearly indicate the semantics of the corresponding content. Finally, as CSS began to be supported by more and more browsers, the design of web pages started moving towards semantic HTML, in which markup is used to reinforce the semantics as opposed to presentation. All these indicate a clear trend in HTML towards separation of presentation from content and its semantics.

In addition to the movement towards separation, HTML/CSS is also beginning to include concepts from word processing. Before HTML5, all content was displayed in one page. HTML5 has introduced a new set of elements, e.g., section, article, header, footer, and aside, to enhance the ability of paginated structure expression. In particular, the new element section, which is similar to the corresponding concept in the office document format, enables paging of the web pages, thus approaching the page formatting capabilities of office documents. Similarly, CSS3, the new version under development since 1998, is providing new features that allow CSS to more adequately style documents for paged display. Also, CSS3 is divided into several parts called modules, unlike earlier versions of CSS, which were large single specifications defining the various features. Major specification modules include: backgrounds and borders, basic box, cascading and inheritance, color, generated and replaced content, fonts, generated content for paged media, template layout, marquee, media queries, selectors, and basic user interface. These refinements further shorten the distance between the office document formats and the web document format and provide almost equivalent capabilities to describe the layout, e.g., fonts, paragraphs, box, etc., as word processing systems provide. However, there are still major differences in other elements.

STYLES IN THE OFFICE DOCUMENT FORMAT

Office document formats are the other type of most widely used reflowable document formats. They are used by office software and Software Development Kits (SDKs) for preparing office documents. Among the major office document formats used today [2-4], UOF is perhaps the most representative. The first version of UOF was published in 2007. Currently UOF 2.0 is under development and will be published soon. UOF has harmonized many concepts occurring in ODF and OOXML and can interoperate well with them.

UOF separates presentation from content to some extent. In the packaged container for a UOF document there is a separate XML file describing common styles shared by all the elements of that document. In UOF, an element can reference a style by the style identifier. However, UOF does not force such referencing and allows individual paragraphs or runs to specify their own styles in the element properties. UOF allows the user to embed pure logic data structure into a UOF document package and set up relations between the logic data and UOF elements [11]. This provides another way to separate presentation from content. However, it is also optional in UOF, actually few authors use the feature to achieve separation.

Another reflowable document format worth mentioning here is XSL-FO. XSL-FO is a markup language for XML document formatting and is most often used to generate PDFs. XSL-FO is part of XSL (Extensible Stylesheet Language), a set of W3C technologies designed for the transformation and formatting of XML data [12]. XSL-FO has no semantic markup in the way it is meant in HTML. And, like UOF, it stores all of the document's data within itself. XSL-FO documents are rarely created as original documents; instead, they are used as an intermediate XSL transformation in some process, such as printing. An XSL-FO document is usually transformed from logical XML data with a ready-made stylesheet, and it can be further transformed into other presentational formats, like PDF, automatically by the target format processor. In this sense, one can think that the logical XML data belongs to the content and the stylesheet belongs to the styles. XSLT has the advantages that the stylesheet can be shared by different documents (or XML data) and the styles can be specified in a very flexible way.

Sharing styles is no doubt efficient for typesetting documents provided that content and presentation are well separated. A valuable effort carried out in ISO in 2003 was to specify style libraries to be used in DSSSL [13]. We adapt this idea to establish a style library to be shared among reflowable documents. The following sections focus on how to establish such a style library.

PROBLEMS WITH SHARING STYLES AND INTEROPERABILITY

Based on the above comments about the current CSS and office document formats, we can identify problems that make sharing styles and interoperability difficult at the present time. First, there are the following problems in sharing styles among office document formats and web page formats.

1. The different formats specify styles in different ways with no standard specification, making it difficult to share styles. Even the vocabularies are different. For example, the text levels in UOF are specified using auto-numbering styles, which are quite different from their specification in HTML.
2. Office document styles and web page styles may be embedded inside the document content rather than all collected together in a separate module (although this is strongly discouraged in HTML5).

3. For office documents the styles can only be shared in the scope of individual documents. If two documents use the same styles, each document has to store the complete style specification. This is inefficient in terms of storage and transmission of documents over a network.
4. Office document styles are specified differently than CSS styles, and it is not clear that they are transformable. Even if they are, we still don't know what a standard way to transform them is. This is harmful to interoperability.

One might think that XSLT provides an easy solution to the transformation problem. However, separating presentation from content using XSLT is not ideal due to the following reasons.

1. The XSLT stylesheets are not easy to design. So far there are no efficient tools to author stylesheets. Based on the comments above, especially item 2, the stylesheets would likely be very large and complex, and therefore developing them would be very difficult and error-prone.
2. Although a stylesheet can be shared by documents of the same type, for example UOF documents, it could not be shared by documents of different types, such as OOXML and ODF. Each type would require its own stylesheet.
3. The ways to specify styles in XSLT are different from either CSS or office document formats.
4. CSS allows a document's style to be influenced by multiple style sheets. One style sheet could inherit or "cascade" from another, permitting a mixture of stylistic preferences controlled equally by the site designer and the user. But multiple stylesheets are not allowed in XSL transformation.

As a summary, office document formats and HTML/CSS have different design purposes, and they were developed independently in the years past. But in the background of electronic publishing, a single document will likely need to be published on different media and therefore need to use style relevant to media other than the one for which it was originally intended; separation of presentation from content is perhaps the only way out. Today, HTML and office document formats are finally approaching closer to each other. Compared with office document formats, HTML is more successful in separating presentation from content, and CSS has provided increasingly rich styles for HTML. We believe that if presentation were separated from content in office documents like it is in HTML/CSS documents, many of the problems cited above would be drastically reduced. For example, transformation stylesheets could focus only on the information in the style section without having to be able to extract style from content elements. The challenge, then, is how to make this separation successful in the office document formats in a way that is compatible with current word processing software and how to let documents share the styles as much as possible. As a large amount of office software will be evolved into web office or Office 2.0 under the environment of cloud computing, documents will be switched frequently among different formats for different purposes. Therefore it becomes necessary to improve the present office document format.

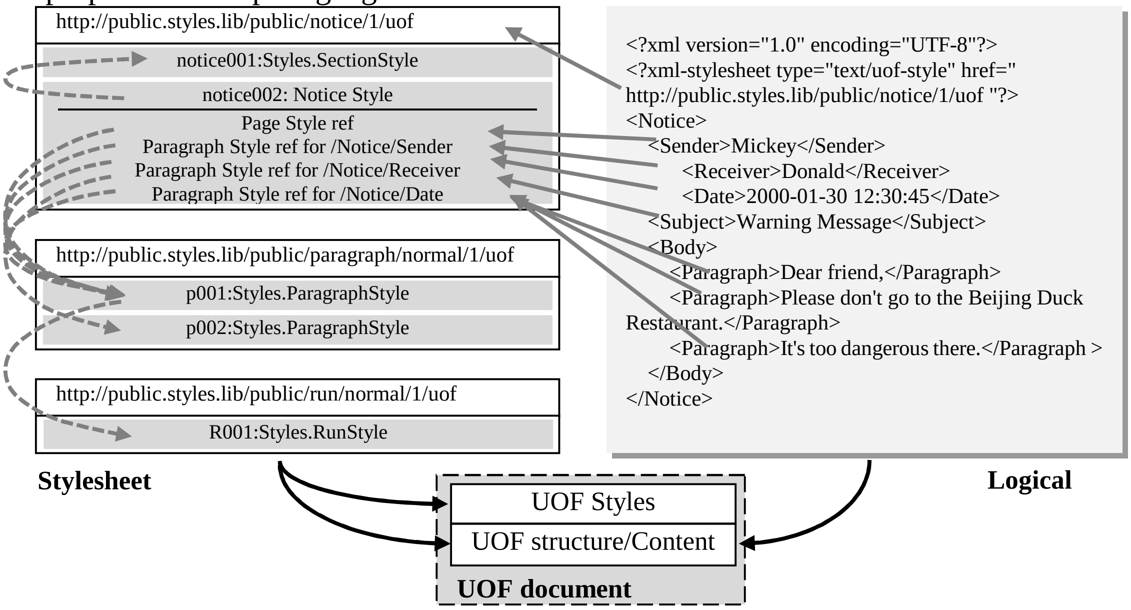
DESIGN OF THE STYLE MARKUP LANGUAGE FOR OFFICE DOCUMENT FORMAT

In this section we propose a style markup language to achieve separation of style and content as well as bring office format systems and HTML/CSS formatting closer. We illustrate the proposed markup language with a simple example. Recall that we want to satisfy the following targets:

1. Presentation can be completely separated from content. In other words, the content contains pure logical data only, just as in XSLT. Even the current design of HTML is not ideal for this. No matter how much focus is placed on semantics by the HTML standards, the syntax of the grammar still allows authors to use elements in ways that mix style and content. The situation in office systems is even worse.
2. Styles can be shared inside and outside documents. Currently, CSS styles and XSLT stylesheets can be reused or shared among documents, but only as a whole. Finer granularity of sharable styles is more desirable, e.g., at paragraph or run levels. A complex style should be able to reuse simple styles. For this purpose, a global naming mechanism, like namespaces identified by URIs, is needed together with a directory structure to store and locate the styles preferred.

3. Compatible with current UOF format. We hope that the change in format will not impact the current UOF implementations. Instead of redesigning UOF format, we want to define a style markup language that can be used on top of existing UOF style mechanisms to specify various styles in a standard way. The styles specified by the markup language will constitute a new type of stylesheet which can be easily integrated into a UOF document. With the style markup language, a UOF document no longer needs to physically include all the style specifications it uses; rather, the presentation of the document can be generated on the fly according to the logical data and the stylesheet, much like XSL-FO.
4. The styles can come from different namespaces, e.g., some from CSS and some from UOF. The specification of styles should be consistent with these dialects so that they can be easily merged into the target format.
5. Styles specified with different formats, e.g., UOF or CSS, can be identified and associated with each other so that a standard mapping among different styles can be created. This is to prevent the chaos caused by arbitrary interpretation of styles.

Fig. 1 shows an example of a UOF document consisting of content and references to a stylesheet that uses our proposed markup language.


http://public.styles.lib/public/notice/1/uof
notice001:Styles.SectionStyle
notice002: Notice Style
Page Style ref
Paragraph Style ref for /Notice/Sender
Paragraph Style ref for /Notice/Receiver
Paragraph Style ref for /Notice/Date
http://public.styles.lib/public/paragraph/normal/1/uof
p001:Styles.ParagraphStyle
p002:Styles.ParagraphStyle
http://public.styles.lib/public/run/normal/1/uof
R001:Styles.RunStyle
Stylesheet
Logical
UOF Styles
UOF structure/Content
UOF document

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/uof-style" href="
http://public.styles.lib/public/notice/1/uof "?>
<Notice>
  <Sender>Mickey</Sender>
      <Receiver>Donald</Receiver>
      <Date>2000-01-30 12:30:45</Date>
  <Subject>Warning Message</Subject>
  <Body>
    <Paragraph>Dear friend,</Paragraph>
    <Paragraph>Please don't go to the Beijing Duck
Restaurant.</Paragraph>
    <Paragraph>It's too dangerous there.</Paragraph >
  </Body>
</Notice>
```

Fig. 1. Sample document with stylesheet reference.

The part of Fig. 1 labeled "Logical Data" forms the content of the document. Note that it contains no style specification explicitly. Instead, it associates the UOF style http://public.styles.lib/public/notice/1/uof using an <?xml-stylesheet> element. It can also associate other stylesheets from different format systems, e.g. CSS, by simply using a different URI. The organization and specification of styles in our proposed language is the subject of the remainder of this section.

We propose a standard organization to store a set of common styles to be shared by documents. The styles can be classified into the following categories based on the current specifications of CSS and UOF: font styles, run styles, paragraph styles, page styles, cell styles, table styles, auto-numbering styles, etc. Each style set is registered with a URI. A style is located by a URL indicating the class path; for example, http://public.styles.lib/public/paragraph/normal/1/uof means the first UOF style for normal paragraphs in the public style library public.styles.lib. A standard organization may specify better structures for classifying the styles and provide the metadata to allow search engines to locate desired styles efficiently. The text file contains the public styles should be either standardized or publicly available on the Internet

We now illustrate our proposed markup language. A typical style can be defined as follows.

```
<?xml version="1.0" encoding="UTF-8"?>
<Styles formatNamespace="http://format.namespace/uof"
targetNamespace="http://public.styles.lib/public/paragraph/normal/1">
   <Style id="p001" name="paragraph-normal-1" class=”Styles.Paragraph”
    >
      <![CDATA[
        <RunStyle ref="http://public.styles.lib/public/run/normal/1/uof"/>
        <OutlineLevel>9</OutlineLevel>
        <Alignment>both< /Alignment>
        <Indent>4</Indent>
        <LineSpace>single< LineSpace>
        <ParagraphSpace>0</ParagraphSpace>
        …
      ]]>
   </Style>
   <Style id=…>…
   </Style>
</Styles>
```

In this example, a set of UOF styles is defined and identified in the target namespace http://public.styles.lib/paragraph/normal/1/uof, among which a style called paragraph-normal-1 is specified using UOF elements. It is identified as p001 inside the style set. It can also be referenced externally using the registered URL, http://public.styles.lib/public/paragraph/normal/1/uof#p001. This paragraph style references another run style identified as http://public.styles.lib/public/run/normal/1/uof. The attribute @class indicates that this is a Paragraph style in the UOF styles. This hierarchical information of styles will be used to construct the presentation of the UOF document later.

```
<?xml version="1.0" encoding="UTF-8"?>
<Styles formatNamespace="http://format.namespace/css"
targetNamespace="http://public.styles.lib/public/paragraph/normal/1">
   <Style id="p001" name="paragraph-nomal-1" class="P">
      <![CDATA[
        font-family: SongTi;
        font-size: 12pt;
        text-align: justify;
        text-indent: 1cm;
        line-height: 1; …
      ]]>
   </Style>
   <Style>...
</Styles>
```

In the second example the target namespace http://public.styles.lib/public/paragraph/normal/1/css indicates that this is the same set of styles as in http://public.styles.lib/public/paragraph/normal/1/uof listed above but specified in CSS format.

In this way, a style hierarchy can be established where styles can reference each other. Style mapping among different formats is also available. The path in the target namespace tells relationship of mapping. The styles with common parent are counterpart styles, and the attribute @formatNamespace indicates the document format which the styles are designed for, e.g., UOF by http://format.namespace/uof or CSS by http://format.namespace/css. Moreover, a user can define private styles which are not to be shared. In this case, the user can specify a private target namespace, e.g., file://localhost/path/my-paragraph-style.xml. The style specification is the same as above but may have higher priority than public styles.

The style of the whole document can be specified in a similar way. The following is an example of a document’s style.

```
<st:Styles xmlns:st="http://schemas.netuof.org/2014/styles"
   xmlns:ust="http://schemas.uof.org/cn/2009/styles"
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xsi:schemaLocation="http://schemas.netuof.org/2014/styles file:Styles.xsd"
   formatNamespace="http://format.namespace/uof"
   targetNamespace="http://public.styles.lib/public/notice/1/uof">
   <st:Style id="notice001" name="page style" class="Styles.Section">
      <![CDATA[
           <PageSize>A4</PageSize>
          <Orientation>horizontal</PageSize>
          <Margin>20mm</Margin>
     ]]>
   </st:Style>
   <st:Style id="notice002" name="Notice Style" class="DocStyle">
      <![CDATA[
<Section styleRef="notice001"/>
<Paragraph             styleRef="http://public.styles.lib/public/paragraph/normal/1/uof#p001"
content="/Notice/Sender"/>
<Paragraph styleRef="http://public.styles.lib/public/paragraph/normal/1/uof#p001"
content="/Notice/Receiver"/>
<Paragraph styleRef="http://public.styles.lib/public/paragraph/normal/1/uof#p001"
content="/Notice/Date"/>
<Paragraph styleRef="http://public.styles.lib/public/paragraph/normal/1/uof#p002"
content="/Notice/Suject"/>
<Paragraph styleRef="http://public.styles.lib/public/paragraph/normal/1/uof#p001"
content="/Notice/Body//Paragraph"/>
]]>
   </st:Style>
</st:Styles>
```

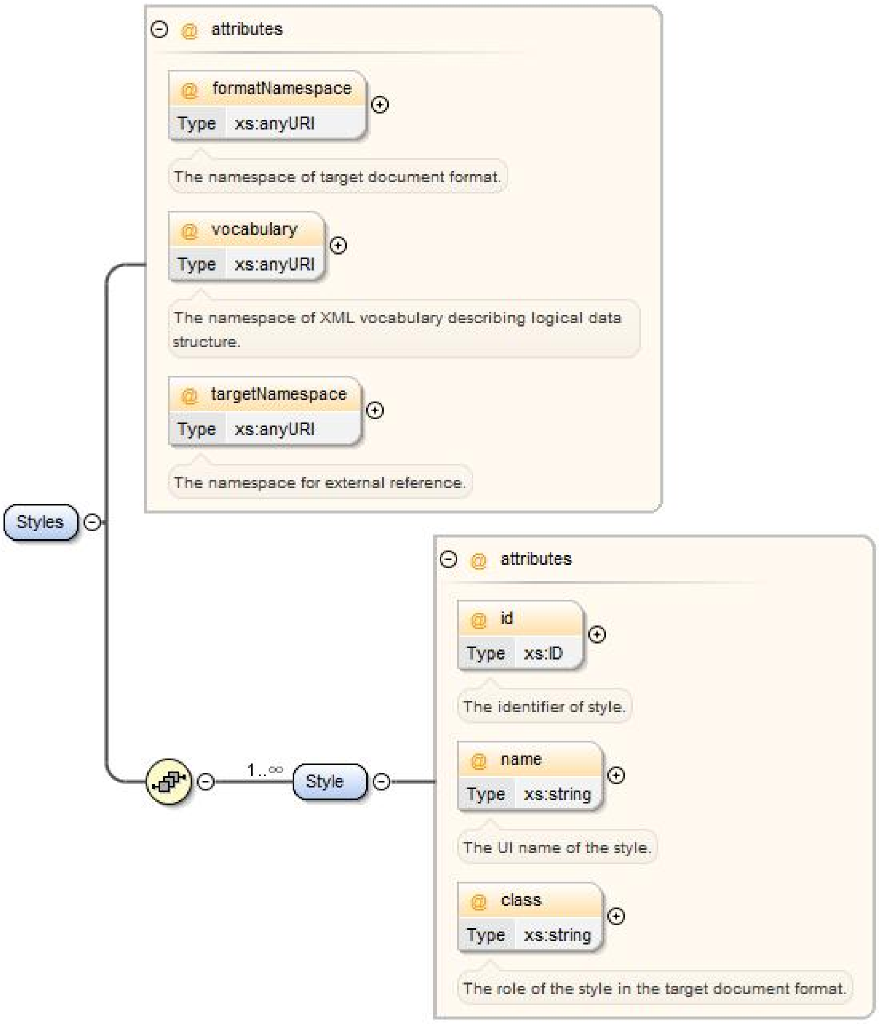
The document style set is identified as http://public.styles.lib/public/notice/1/uof. It is for notices using the UOF format. The attribute @vocabulary indicates that the namespace of the notice document type is http://vocabulary.namespace/notice. The first part notice001 specifies the general page style, including page size, orientation and margin. It is the Section Style in the UOF Styles. The second part notice002 specifies the notice document style. It has a section which uses the above page style. Under the section, there are five types of paragraph which correspond to sender, receiver, date, title and body of the notice. The first three and the last one follow the paragraph style of http://public.styles.lib/public/paragraph/normal/1/uof#p001, while the third one follows the style of http://public.styles.lib/public/paragraph/normal/1/uof#p002. Note that the single element Paragraph may generate more than one actual paragraph depending on how many data elements match the XPath expression in @content.

Fig. 2 shows the formal definition of the style markup language in XML Schema.

The style markup language can meet the targets listed above. It is the task of the style markup language processor or office software to parse the styles specified by the style markup language and generate the document’s presentation in the user interface, similar to the way a web page described in HTML+CSS is processed by the web browser before being displayed.

INTEROPERABILITY BETWEEN DOCUMENT STRUCTURES

The interoperability between the office document format and HTML/CSS depends on the common points of elements and attributes representing style and structure in both kinds of documents. HTML and CSS have many elements related to multimedia and interactive user interface, for example, form, canvas, events, video, audio, etc. These do not have counterparts in UOF. On the other hand, UOF also has many elements related to editing semantics, like auto-numbering, character grid, typesetting rules, change tracking, etc., that HTML/CSS does not have. It is hard to calculate the interoperability between HTML5/CSS3 and UOF accurately due to the different vocabularies [14-15]. But for text and layout, the two formats have similar block-level and inline-level elements, most of which are transformable. We estimate that the interoperability is about 75%.



The namespace of target document format.

The namespace of XML vocabulary describing logical data structure.

The namespace for external reference.

The identifier of style.

The UI name of the style.

The role of the style in the target document format.

Fig. 2. Style markup language schema

Both document format systems have concepts of block level and inline level elements. Block-level elements are elements that create large blocks of content like paragraphs or page divisions. They start new lines of text and can contain other blocks as well as inline elements and text or data. Inline elements are elements that do not start new lines, and they generally only contain other inline tags and text or data. Similar concepts exist in office document formats too.

However there are still some significant differences. For example, HTML has the block-level element called div, a box container for grouping other HTML elements. An HTML document consists of many div boxes and paragraph boxes, thus the whole document can be considered as a flow of boxes. Presentation of an HTML document requires, among other things, locating the position of each box. There is no such element in UOF. However, UOF has an element called section which holds successive paragraphs with the same layout. Sometimes, the div box in HTML can be described by the object positioned by an anchor in UOF too. The presentation of a UOF document is thus to locate the position of each section as well as the position of each paragraph and object within the section. Another difference is that HTML has the element aside which defines some content aside from the content it is placed in. UOF does not have such an element, but it has a structure to record comments. HTML’s aside can be expressed by a comment in UOF, however the location is decided by the rendering program and will not necessarily be the same as in HTML.

The steps to transform a document from HTML+CSS into UOF include: 1) to analyze the styles in the original CSS; 2) to transform the CSS styles into UOF styles and add them into the UOF document; 3) to transform the HTML elements into UOF elements and add them into the UOF document; 4) to associate the UOF elements with the styles in the UOF document.

In the Appendix we show an excerpt from the W3C web page Cascading Style Sheets (CSS) Snapshot 2010 [16] and the corresponding UOF-like document to illustrate the effect.

To transform a document from UOF into HTML+CSS is more difficult without the help of a style markup language, as it is by no means easy to abstract styles from content in a hybrid document. For example, a hybrid UOF document of Fig. A2 is like:

```
<uof:WordProcessing xmlns:uof="http://schemas.uof.org/cn/2009/uof">
  <uof:Section>
    <uof:SectionProperty>
      <uof:Margin left="90.0" top="72.0" right="90.0" bottom="72.0"/>
      <uof:paper height="841.9" width="595.3"/>
      <uof:headerPosition toMargin="42.55"/>
      <uof:footerPosition toMargin="49.6"/>
      <uof:PaperOrientation>portrait</uof:PaperOrientation>
      <uof:VerticalAlignment>top</uof:VerticalAlignment>
      <uof:WritingMode>t2b-l2r-0e-0w</uof:WritingMode>
      <uof:color>#ffffff</uof:color>
    </uof:SectionProperty>
    <uof:Paragraph>
      <uof:ParagrapProperty>
        <uof:OutlineLevel>2</uof:OutlineLevel>
        <uof:Space>
          <uof:Before>
            <uof:Absolute>24.0</uof:Absolute>
          </uof:Before>
          <uof:After>
            <uof:Auto/>
          </uof:After>
        </uof:Space>
        <uof:Background color="#ffffff"/>
      </uof:ParagrapProperty>
      <uof:Run>
        <uof:RunProperty>
          <uof:Font west="Arial" size="20.5" color="#005a9c "/>
          <uof:IsBold>true</uof:IsBold>
          <uof:SpaceAdjust>18.0</uof:SpaceAdjust>
        </uof:RunProperty>
        <uof:Text> Cascading Style Sheets (CSS) Snapshot 2010 </uof:Text>
      </uof:Run>
    </uof:Paragraph>
    <uof:Paragraph>
      ......
    </uof:Paragraph>
  </uof:Section>
</uof:WordProcessing>
```

It is hard to transform it into CSS+HTML in Table A1 and Table A3.

In the future, we suggest to eliminate the hybrid mode of styles and content in UOF and instead to separate the presentation from content and specify the styles using the style markup language. If this is done, the procedure will be straightforward, much like in the transformation from HTML+CSS into UOF. Separation of content from style in office document formats is certainly feasible as ODF has already separated styles from content from the very beginning.

CONCLUSIONS

This paper proposed a style markup language. The purpose is to improve the interoperability between the office document format and the web document format so that they can be transformed easily in the future. The style markup language can promote the separation of presentation and content and enable us to standardize styles and reuse them, share styles between documents, decrease the size of documents, reduce the burden on file storage and transmission, and so on. They are particularly important to web office or Office 2.0, which many believe will be the main form of office applications in the age of cloud computing.

Some innovative contributions made in this paper include:

1. The style markup language is neither the same as the styles in the office document format, e.g., UOF, nor the same as CSS for HTML. It can incorporate any style specified in different dialects and associate the styles with each other. In this sense, it is a meta-markup language for styles. Such a design concept is proposed here for the first time.
2. The document structures of UOF and HTML are briefly compared. This paper indicates that it is hard to transform between current office documents and HTML/CSS documents, as office documents usually are a hybrid of styles and content. The transformation should become easier if the styles are separated from the content with the help of the style markup language. A sample transform example was shown to illustrate the feasibility.
3. The application of a style markup language will not cause significant change in UOF. All the styles and content can still follow the syntax and semantics of the current UOF standard. The only notable change in future UOF is to deprecate the hybrid mode of styles and content and to save the styles and content separately. However, this will not prevent the application from generating conventional UOF documents for temporary use so that they can be processed by current word processing software.

In a word, the style markup language serves as a bridge for UOF toward separating presentation from content. However, there are still some problems worth researching. Firstly, can we tell clearly what belongs to presentation and what belongs to content; for example, is an image in the background presentation or content? Sometimes the possibility and degrees of separation are as subjective as the content itself. Secondly, styles in the office documents are highly related to editing semantics, for example, the style of auto-numbering affects how numbered lists and headings are edited. Can we find the proper way to specify the styles while giving enough consideration to editing semantics? Moreover, can the styles be used as flexibly as in XSLT? Luckily the meta-style markup language does not need to care much about how the styles are specified in a particular format. However, future UOF and other office document format specifications should care. Lastly, the styles have to be managed by proper organizations, and efficient tools are required to create, submit, search and retrieve the styles. It may increase the complexity of the applications. It is the mainstream of office applications that will finally decide whether the design of the style markup language is successful or not.

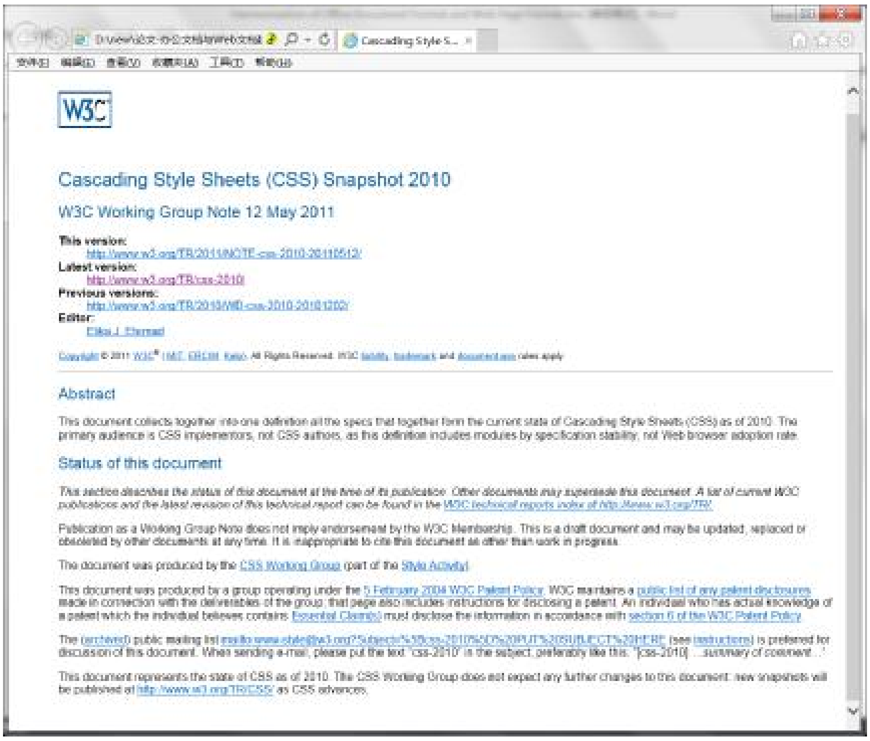
ACKNOWLEDGMENTS

This work was partially supported by the Project of Construction of Innovative Teams and Teacher Career Development for Universities and Colleges under Beijing Municipality (IDHT20130519).

The authors would like to express sincere thanks to Professor Lawrence Henschen for his kind help to finish this paper.

APPENDIX

We chose an excerpt from W3C web page Cascading Style Sheets (CSS) Snapshot 2010 [15] as an example and transformed it into a UOF-like document. The page includes a HTML document with a CSS stylesheet. Some unused features are removed from the styles to keep it as short as possible. The UOF tags are also simplified and translated from Chinese into English. Figure A1 shows the original document browsed in IE9. Figure A2 shows the transformed result in YOZO Office 2013.



Cascading Style Sheets (CSS) Snapshot 2010

W3C Working Group Note 12 May 2011

This version:
http://www.w3.org/TR/2011/NOTE-css-2010-20110512/
Latest version:
http://www.w3.org/TR/css-2010
Previous versions:
http://www.w3.org/TR/2010/WD-css-2010-20101202/
Editor:
Elika J. Etemad

Copyright © 2011 W3C® (MIT, ERCIM, Keio), All Rights Reserved. W3C liability, trademark and document use rules apply.

Abstract

This document collects together into one definition all the specs that together form the current state of Cascading Style Sheets (CSS) as of 2010. The primary audience is CSS implementors, not CSS authors, as this definition includes modules by specification stability, not Web browser adoption rate.

Status of this document

This section describes the status of this document at the time of its publication. Other documents may supersede this document. A list of current W3C publications and the latest revision of this technical report can be found in the W3C technical reports index at http://www.w3.org/TR/.

Publication as a Working Group Note does not imply endorsement by the W3C Membership. This is a draft document and may be updated, replaced or obsoleted by other documents at any time. It is inappropriate to cite this document as other than work in progress.

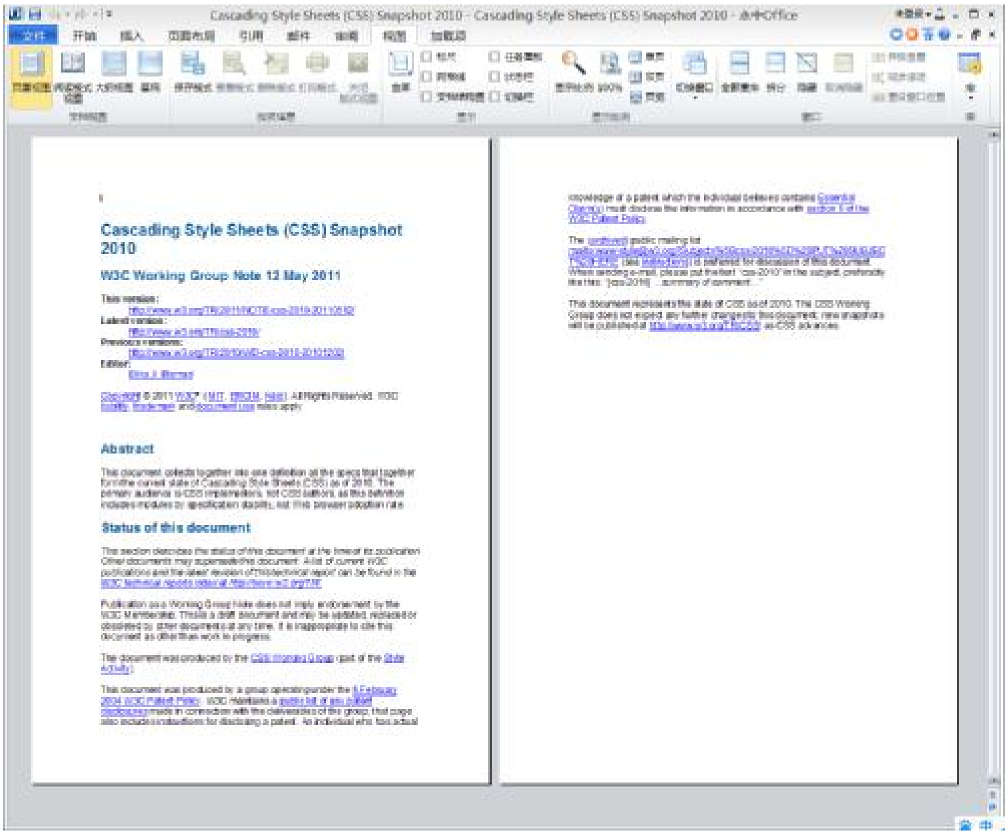
The document was produced by the CSS Working Group (part of the Style Activity).

This document was produced by a group operating under the 5 February 2004 W3C Patent Policy. W3C maintains a public list of any patent disclosures made in connection with the deliverables of the group; that page also includes instructions for disclosing a patent. An individual who has actual knowledge of a patent which the individual believes contains Essential Claim(s) must disclose the information in accordance with section 6 of the W3C Patent Policy.

The (archived) public mailing list mailto:www-style@w3.org?Subject=%5Bcss-2010%5D%20PUT%20SUBJECT%20HERE (see instructions) is preferred for discussion of this document. When sending e-mail, please put the text "css-2010" in the subject, preferably like this: "[css-2010] …summary of comment…"

This document represents the state of CSS as of 2010. The CSS Working Group does not expect any further changes to this document: new snapshots will be published at http://www.w3.org/TR/CSS/ as CSS advances.

Figure A1. The original document browsed in IE9


Cascading Style Sheets (CSS) Snapshot 2010 - Cascading Style Sheets (CSS) Snapshot 2010 - 永中Office
文件
开始
插入
页面布局
引用
邮件
审阅
视图
加载项

Cascading Style Sheets (CSS) Snapshot 2010

W3C Working Group Note 12 May 2011

This version:
http://www.w3.org/TR/2011/NOTE-css-2010-20110512/
Latest version:
http://www.w3.org/TR/css-2010/
Previous versions:
http://www.w3.org/TR/2010/WD-css-2010-20101202/
Editor:
Elika J. Etemad

Copyright © 2011 W3C® (MIT, ERCIM, Keio), All Rights Reserved. W3C liability, trademark and document use rules apply.

Abstract

This document collects together into one definition all the specs that together form the current state of Cascading Style Sheets (CSS) as of 2010. The primary audience is CSS implementors, not CSS authors, as this definition includes modules by specification stability, not Web browser adoption rate.

Status of this document

This section describes the status of this document at the time of its publication. Other documents may supersede this document. A list of current W3C publications and the latest revision of this technical report can be found in the W3C technical reports index at http://www.w3.org/TR/.

Publication as a Working Group Note does not imply endorsement by the W3C Membership. This is a draft document and may be updated, replaced or obsoleted by other documents at any time. It is inappropriate to cite this document as other than work in progress.

The document was produced by the CSS Working Group (part of the Style Activity).

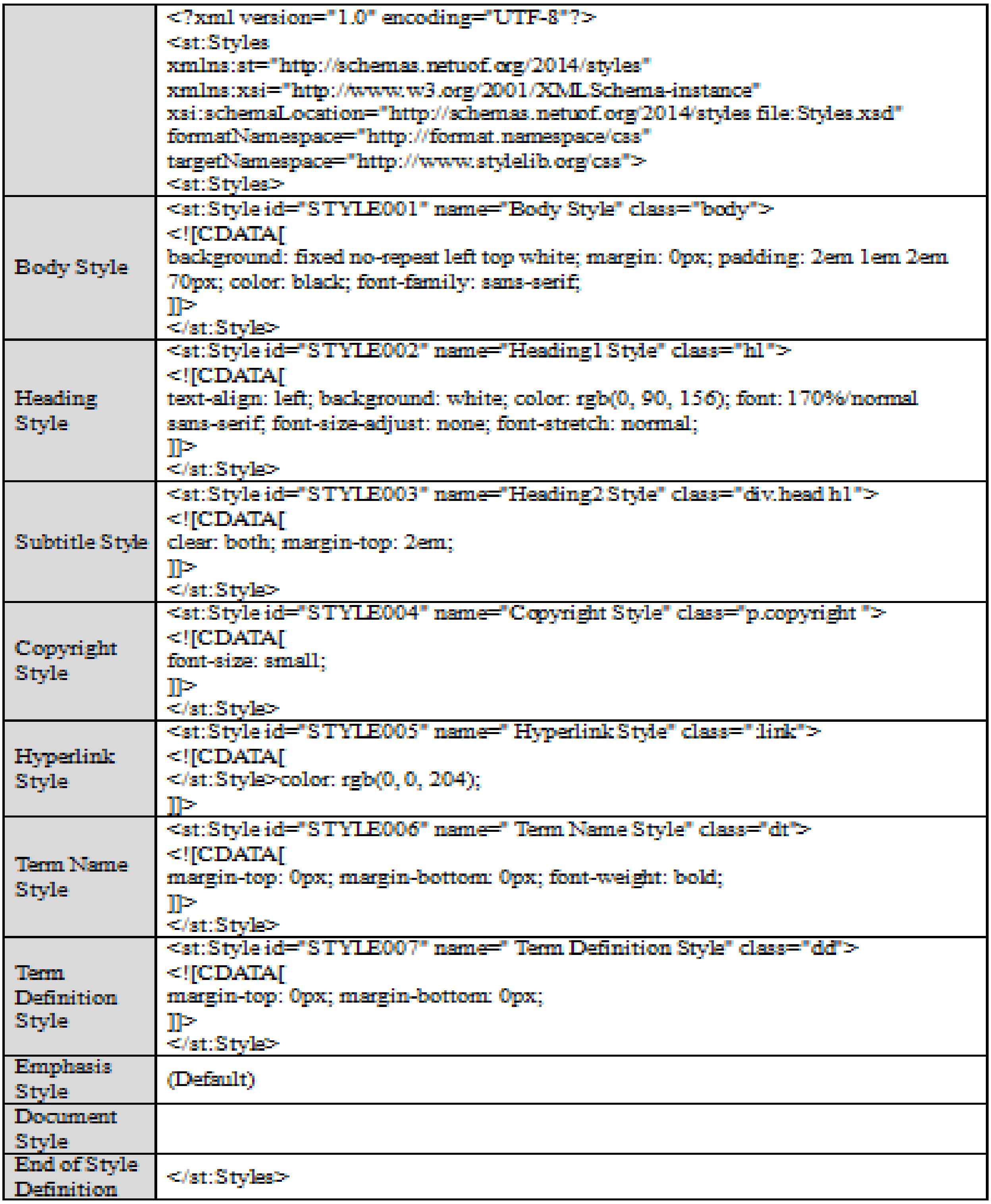
This document was produced by a group operating under the 5 February 2004 W3C Patent Policy. W3C maintains a public list of any patent disclosures made in connection with the deliverables of the group; that page also includes instructions for disclosing a patent. An individual who has actual knowledge of a patent which the individual believes contains Essential Claim(s) must disclose the information in accordance with section 6 of the W3C Patent Policy.

The (archived) public mailing list mailto:www-style@w3.org?Subject=%5Bcss-2010%5D%20PUT%20SUBJECT%20HERE (see instructions) is preferred for discussion of this document. When sending e-mail, please put the text "css-2010" in the subject, preferably like this: "[css-2010] …summary of comment…"

This document represents the state of CSS as of 2010. The CSS Working Group does not expect any further changes to this document: new snapshots will be published at http://www.w3.org/TR/CSS/ as CSS advances.

Figure A2. The transformed result in YOZO Office 2013

Table A1 and Table A2 show the CSS and UOF stylesheets for these two documents. Table A3 shows the HTML file associated with the CSS styles. Table A4 shows the data file associated with the UOF styles.



	<?xml version="1.0" encoding="UTF-8"?> <st:Styles xmlns:st="http://schemas.netuof.org/2014/styles" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://schemas.netuof.org/2014/styles file:Styles.xsd" formatNamespace="http://format.namespace/css" targetNamespace="http://www.stylelib.org/css"> <st:Styles>
Body Style	<st:Style id="STYLE001" name="Body Style" class="body"> <![CDATA[background: fixed no-repeat left top white; margin: 0px; padding: 2em 1em 2em 70px; color: black; font-family: sans-serif;]]> </st:Style>
Heading Style	<st:Style id="STYLE002" name="Heading1 Style" class="h1"> <![CDATA[text-align: left; background: white; color: rgb(0, 90, 156); font: 170%/normal sans-serif; font-size-adjust: none; font-stretch: normal;]]> </st:Style>
Subtitle Style	<st:Style id="STYLE003" name="Heading2 Style" class="div.head h1"> <![CDATA[clear: both; margin-top: 2em;]]> </st:Style>
Copyright Style	<st:Style id="STYLE004" name="Copyright Style" class="p.copyright "> <![CDATA[font-size: small;]]> </st:Style>
Hyperlink Style	<st:Style id="STYLE005" name=" Hyperlink Style" class=":link"> <![CDATA[</st:Style>color: rgb(0, 0, 204);]]>
Term Name Style	<st:Style id="STYLE006" name=" Term Name Style" class="dt"> <![CDATA[margin-top: 0px; margin-bottom: 0px; font-weight: bold;]]> </st:Style>
Term Definition Style	<st:Style id="STYLE007" name=" Term Definition Style" class="dd"> <![CDATA[margin-top: 0px; margin-bottom: 0px;]]> </st:Style>
Emphasis Style	(Default)
Document Style	
End of Style Definition	</st:Styles>

Table A1. The CSS stylesheet

	<?xml version="1.0" encoding="UTF-8"?> <st:Styles xmlns:st="http://schemas.netuof.org/2014/styles" xmlns:uof="http://format.namespace/uof" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://schemas.netuof.org/2014/styles file:Styles.xsd" formatNamespace="http://format.namespace/uof" targetNamespace="http://www.stylelib.org/uof"> <st:Styles>

Body Style	<st:Style id="STYLE001" name="Page Style" class="Styles.Section">
	<![CDATA[
	    <uof:Margin left="90.0" top="72.0" right="90.0" bottom="72.0"/>
	    <uof:paper height="841.9" width="595.3"/>
	    <uof:headerPosition toMargin="42.55"/>
	    <uof:footerPosition toMargin="49.6"/>
	    <uof:PaperOrientation>portrait</uof:PaperOrientation>
	    <uof:VerticalAlignment>top</uof:VerticalAlignment>
	    <uof:WritingMode>t2b-l2r-0e-0w</uof:WritingMode>
	    <uof:color>#ffffff</uof:color>
	]]>
	</st:Style>

Heading Style	<st:Style id="STYLE002" name="Heading Run Style" class=" Styles.Run"> <![CDATA[<uof:Font west="Arial" size="20.5" color="#005a9c "/> <uof:IsBold>true</uof:IsBold> <uof:SpaceAdjust>18.0</uof:SpaceAdjust>]]> </st:Style>

	<st:Style id="STYLE003" name="Heading Paragraph Style" class=" Styles.Paragraph">
	<![CDATA[
	            <uof:OutlineLevel>2</uof:OutlineLevel >
	    <uof:Space>
	        <uof:Before>
	            <uof:Absolute>24.0</uof:Absolute >
	        </uof:Before>
	        <uof:After><uof:Auto/></uof:After>
	    </uof:Space>
	    <uof:Background color="#ffffff"/>
	]]>
	</st:Style>

Subtitle Style	<st:Style id="STYLE004" name=" Subtitle Style" class=" Styles.Paragraph"> <![CDATA[… (omitted)]]> </st:Style>
Copyright Style	<st:Style id="STYLE005" name=" Copyright Run Style" class=" Styles.Run"> <![CDATA[<uof:Font west="Arial"/>]]> </st:Style>

	<st:Style id="STYLE006" name="Copyright Paragraph Style" class=" Styles.Paragraph">
	<![CDATA[
	    <uof:Sapce>
	        <uof:Before>
	            <uof:Auto/>
	        </uof:Before>
	        <uof:After>
	            <uof:Auto/>
	        </uof:After>
	    </uof:Sapce>
	]]>
	</st:Style>

Hyperlink Style	<st:Style id="STYLE007" name="Hyperlink Style" class=" Styles.Run"> <![CDATA[<uof:Font west="Arial" color="#0000ff"/> <uof:Underscore lineType="single"/>]]> </st:Style>
Term Name Style	<st:Style id="STYLE008" name="Term Name Style" class=" Styles.Paragraph"> <![CDATA[<uof:Font west="Arial" color="#0000ff"/> <uof:IsBold>true</uof:Isbold>]]> </st:Style>

Term Definition Style	<st:Style id="STYLE009" name="Term Definition Style" class=" Styles.Paragraph">
	<![CDATA[
	            <uof:Indent>
	                <uof:Left>
	                    <uof:Absolute value="36.0"/>
	                </uof:Left>
	            </uof:Indent>
	]]>
	</st:Style>

Emphasis Style	<st:Style id="STYLE010" name="Emphasis Style" class=" Styles.Run"> <![CDATA[<uof:IsItalic>true</uof:IsItalic >]]> </st:Style>

Document Style	<st:Style id="STYLE100" name="W3C Document Style" class="DocStyle">
	<![CDATA[
	<uof:WordProcessing xmlns:uof="http://schemas.uof.org/cn/2009/uof">
	    <uof:Section section_style="STYLE001"/>
	    <uof:Paragraph run_style="STYLE002" paragraph_style="STYLE003" content="/Doc/Title"/>
	    <uof:Paragraph run_style="STYLE002" paragraph_style="STYLE003" content="/Doc/SubTitle"/>
	    <uof:Paragraph paragraph_style="STYLE008" content="/Doc/Versions/Version/@title"/>
	    <uof:Paragraph paragraph_style="STYLE009" content="/Doc/Versions/Version"/>
	    <uof:Paragraph paragraph_style="STYLE008" content="/Doc/Editor/@title"/>
	    <uof:Paragraph paragraph_style="STYLE009">
	        <uof:Run run_style="STYLE007" link="/Doc/Editor/@Contact" content="/Doc/Editor"/>
	    </uof:Paragraph>
	    <uof:Paragraph run_style="STYLE005" paragraph_style="STYLE006" content="/Doc/Copyright">
	        <uof:Run style="STYLE007" link="/Doc/Copyright//A/@href" content="/Doc/Copyright//A"/>
	    </uof:Paragraph>
	    <uof:Paragraph run_style="STYLE002" paragraph_style="STYLE003" content="/Doc/Abstract/@title"/>
	    <uof:Paragraph run_style="STYLE005" paragraph_style="STYLE006" content="/Doc/Abstract"/>
	    <uof:Paragraph run_style="STYLE002" paragraph_style="STYLE003" content="/Doc/Status/@title"/>
	    <uof:Paragraph run_style="STYLE005" paragraph_style="STYLE006" content="/Doc/Status/Paragraph">
	        <uof:Run run_style="STYLE010" content="/Doc/Status/Paragraph/Emphasis"/>
	        <uof:Run run_style="STYLE007" link="/Doc/Status/Paragraph//A/@href"
	            content="/Doc/Status/Paragraph//A"/>
	    </uof:Paragraph>
	</uof:WordProcessing>
	]]>
	</st:Style>

End of Style Definition	</st:Styles>

Table A2. The UOF stylesheet

```
<HTML lang="en">
  <HEAD>
    <META content="IE=10.000" http-equiv="X-UA-Compatible"/>
    <TITLE>Cascading Style Sheets (CSS) Snapshot 2010</TITLE>
    <META http-equiv="Content-Type" content="text/html; charset=utf-8"/>
    <LINK href="W3C-WG-NOTE.css"
      rel="stylesheet" type="text/css"/>
    <META name="GENERATOR" content="MSHTML 10.00.9200.17054"/>
  </HEAD>
<P class="copyright"><A href="http://www.w3.org/Consortium/Legal/ipr-notice#Copyright"
rel="license">Copyright</A> © 2011 <A href="http://www.w3.org/"><ACRONYM title="World Wide Web&#10;
Consortium">W3C</ACRONYM></A><SUP>®</SUP> ( <A href="http://www.csail.mit.edu/"><ACRONYM
title="Massachusetts Institute&#10; of Technology">MIT</ACRONYM></A>, <A
href="http://www.ercim.eu/"><ACRONYM title="European Research Consortium for Informatics and&#10;
Mathematics">ERCIM</ACRONYM></A>, <A href="http://www.keio.ac.jp/">Keio</A>), All Rights Reserved. W3C
<A href="http://www.w3.org/Consortium/Legal/ipr-notice#Legal_Disclaimer">liability</A>, <A
href="http://www.w3.org/Consortium/Legal/ipr-notice#W3C_Trademarks">trademark</A> and <A
href="http://www.w3.org/Consortium/Legal/copyright-documents">document use</A> rules apply.</P>
      <HR title="Separator for header"/>
    </DIV>
    <H2 class="no-num no-toc" id="abstract">Abstract</H2>
<P>This document collects together into one definition all the specs that together form the current state of Cascading Style
Sheets (CSS) as of 2010. The primary audience is CSS implementors, not CSS authors, as this definition includes modules
by specification stability, not Web browser adoption rate. </P>
    <H2 class="no-num no-toc" id="status">Status of this document</H2>
<P><EM>This section describes the status of this document at the time of its publication. Other documents may supersede
this document. A list of current W3C publications and the latest revision of this technical report can be found in the <A
href="http://www.w3.org/TR/">W3C technical reports index at http://www.w3.org/TR/.</A></EM></P>
<P>Publication as a Working Group Note does not imply endorsement by the W3C Membership. This is a draft document
and may be updated, replaced or obsoleted by other documents at any time. It is inappropriate to cite this document as other
than work in progress. </P>
    <P>The document was produced by the <A href="http://www.w3.org/Style/CSS/members">CSS Working Group</A>
(part of the <A href="http://www.w3.org/Style/CSS">Style Activity</A>). </P>
<P> This document was produced by a group operating under the <A
href="http://www.w3.org/Consortium/Patent-Policy-20040205/">5 February 2004 W3C Patent Policy</A>. W3C
maintains a <A href="http://www.w3.org/2004/01/pp-impl/32061/status" rel="disclosure">public list of any patent
disclosures</A> made in connection with the deliverables of the group; that page also includes instructions for disclosing
a patent. An individual who has actual knowledge of a patent which the individual believes contains <A
href="http://www.w3.org/Consortium/Patent-Policy-20040205/#def-essential">Essential Claim(s)</A> must disclose the
information in accordance with <A
href="http://www.w3.org/Consortium/Patent-Policy-20040205/#sec-Disclosure">section 6 of the W3C Patent
Policy</A>. </P>
    <P>The (<A href="http://lists.w3.org/Archives/Public/www-style/">archived</A>) public mailing list <A
href="mailto:www-style@w3.org?Subject=%5Bcss-2010%5D%20PUT%20SUBJECT%20HERE">mailto:www-style
@w3.org?Subject=%5Bcss-2010%5D%20PUT%20SUBJECT%20HERE</A> (see <A
href="http://www.w3.org/Mail/Request">instructions</A>) is preferred for discussion of this document. When sending
e-mail, please put the text “css-2010” in the subject, preferably like this: “[<!---->css-2010<!---->] <EM>…summary of
comment…</EM>”</P>
<P>This document represents the state of CSS as of 2010. The CSS Working Group does not expect any further changes to
this document: new snapshots will be published at <A
href="http://www.w3.org/TR/CSS/">http://www.w3.org/TR/CSS/</A> as CSS advances. </P>
  </BODY>
</HTML>
```

Table A3. The HTML file associated with the CSS styles

```
<?xml-stylesheet type="text/uof-style" href="http://www.stylelib.org/uof#STYLE100"?>
<Doc>
   <Title>Cascading Style Sheets (CSS) Snapshot 2010</Title>
   <SubTitle>W3C Working Group Note</SubTitle>
   <Versions>
      <Version title="This version:">http://www.w3.org/TR/2011/NOTE-css-2010-20110512/</Version>
      <Version title="Latest version::">http://www.w3.org/TR/css-2010/</Version>
      <Version title="Previous versions:"
         >http://www.w3.org/TR/2010/WD-css-2010-20101202/</Version>
   </Versions>
   <Editor title="Editor:" Contact="http://fantasai.inkedblade.net/contact">Elika J. Etemad</Editor>
   <Copyright>
      <A href="http://www.w3.org/Consortium/Legal/ipr-notice#Copyright">Copyright</A> © 2011 <A
href="http://www.w3.org/">W3C</A> ( <A href="http://www.csail.mit.edu/">MIT</A>, <A
href="http://www.ercim.eu/">ERCIM</A>, <A href="http://www.keio.ac.jp/">Keio</A>), All Rights Reserved. W3C <A
href="http://www.w3.org/Consortium/Legal/ipr-notice#Legal_Disclaimer">liability</A>, <A
href="http://www.w3.org/Consortium/Legal/ipr-notice#W3C_Trademarks">trademark</A> and <A
href="http://www.w3.org/Consortium/Legal/copyright-documents">document use</A> rules apply. </Copyright>
   <Abstract title="Abstract"> This document collects together into one definition all the specs that together form the
current state of Cascading Style Sheets (CSS) as of 2010. The primary audience is CSS implementors, not CSS authors, as
this definition includes modules by specification stability, not Web browser adoption rate. </Abstract>
   <Status title="Status of this document">
      <Paragraph>
         <Emphasis>This section describes the status of this document at the time of its publication. Other documents may
supersede this document. A list of current W3C publications and the latest revision of this technical report can be found in
the <A href="http://www.w3.org/TR/">W3C technical reports index at http://www.w3.org/TR/.</A></Emphasis>
      </Paragraph>
      <Paragraph>Publication as a Working Group Note does not imply endorsement by the W3C Membership. This is a
draft document and may be updated, replaced or obsoleted by other documents at any time. It is inappropriate to cite this
document as other than work in progress. </Paragraph>
      <Paragraph>The document was produced by the <A href="http://www.w3.org/Style/CSS/members">CSS Working
Group</A> (part of the <A href="http://www.w3.org/Style/CSS">Style Activity</A>). </Paragraph>
      <Paragraph> This document was produced by a group operating under the <A
href="http://www.w3.org/Consortium/Patent-Policy-20040205/">5 February 2004 W3C Patent Policy</A>. W3C
maintains a <A href="http://www.w3.org/2004/01/pp-impl/32061/status" rel="disclosure">public list of any patent
disclosures</A> made in connection with the deliverables of the group; that page also includes instructions for disclosing a
patent. An individual who has actual knowledge of a patent which the individual believes contains <A
href="http://www.w3.org/Consortium/Patent-Policy-20040205/#def-essential">Essential Claim(s)</A> must disclose the
information in accordance with <A
href="http://www.w3.org/Consortium/Patent-Policy-20040205/#sec-Disclosure">section 6 of the W3C Patent
Policy</A>. </Paragraph>
      <Paragraph>The (<A href="http://lists.w3.org/Archives/Public/www-style/">archived</A>) public mailing list <A
href="mailto:www-style@w3.org?Subject=%5Bcss-2010%5D%20PUT%20SUBJECT%20HERE">mailto:www-style@
w3.org?Subject=%5Bcss-2010%5D%20PUT%20SUBJECT%20HERE</A> (see <A
href="http://www.w3.org/Mail/Request">instructions</A>) is preferred for discussion of this document. When sending
e-mail, please put the text “css-2010” in the subject, preferably like this: “[<!---->css-2010<!---->] <Emphasis>…summary
of comment…</Emphasis>”</Paragraph>
      <Paragraph>This document represents the state of CSS as of 2010. The CSS Working Group does not expect any
further changes to this document: new snapshots will be published at <A
href="http://www.w3.org/TR/CSS/">http://www.w3.org/TR/CSS/</A> as CSS advances. </Paragraph>
   </Status>
</Doc>
```

Table A4. The data file associated with the UOF styles

REFERENCES

[1] LI, N.; Tian, Y.A.; Hou, X.; Liang, Q. A Discussion on Relationship between Revisable and Non-revisable Document Formats. Acta Electronica Sinica 2008, 36, 128-132.

[2] ISO/IEC 26300:2006. Information technology - Open Document Format for Office Applications (OpenDocument). 2006.

[3] ISO/IEC 29500-1:2008. Information technology - Document description and processing languages - Office Open XML File Formats - Part 1: Fundamentals and Markup Language Reference. 2006.

[4] GB/T20916-2007. Specification for the Chinese office file format. 2007.

[5] W3C. HTML5. http://www.w3.org/TR/html5/ (accessed on 7 July, 2014).

[6] WYSIWYM. http://en.wikipedia.org/wiki/WYSIWYM (accessed on 7 July, 2014).

[7] Dodds, L. XSL and CSS: One Year Later. http://www.xml.com/pub/a/2000/06/21/deviant/index.html (accessed on 7 July, 2014).

[8] Lie, H.W.; Day, M. Printing XML: Why CSS Is Better than XSL. http://www.xml.com/pub/a/2005/01/19/print.html (accessed on 7 July, 2014).

[9] Corinthia Home. http://cwiki.apache.org/confluence/display/Corinthia/Corinthia+Home (accessed on 5 February, 2015)

[10] W3C. Tags used in HTML. http://www.w3.org/History/19921103-hypertext/hypertext/WWW/MarkUp/Tags (accessed on 7 July, 2014).

[11] LI, N; MU, Y.M.; DONG, H. Separation and Combination of Content and Appearance in Document Format. Acta Electronica Sinica 2007, 35, 375-378.

[12] W3C. Extensible Stylesheet Language (XSL) Version 1.1. http://www.w3.org/TR/xsl11/ (accessed on 7 July, 2014).

[13] ISO/IEC TR 19758. Information technology - Document description and processing languages - DSSSL library for complex compositions. 2003.

[14] Li, N.; Liang, Q.; Hou, X., Tian Y.A. Interoperability Measurement of Documents, Journal of Beijing Information Science & Technology University 2011, 26, 6-12.

[15] Xu, Z.W. The Analysis and Improvement of Interoperability between XSL-FO and Office Document Format. MSc thesis, Beijing Information Science and Technology University, Beijing, 2014.1.

[16] W3C. Cascading Style Sheets (CSS) Snapshot 2010. http://www.w3.org/TR/css-2010/ (accessed on 7 July, 2014).