文档格式中“内容”与“表现”的分离与融合

李 宁,牟永敏,董 慧,方春燕
(北京信息工程学院计算机科学与工程系,北京 100101)

摘 要: 置标语言产生的一个初衷是要做到文档内容与表现的分离. 现有技术中公开的文档记录格式一般是通过定义与显现格式相关的 XML Schema 来支持混合的文档格式与内容信息,不能很好地达到内容与表现分离的目的. 针对《中文办公软件文档格式规范》制定的需要,本文提出了一种新的格式记录方法,即在文档中记录文档格式描述和用户数据内容两个部分,并建立两者的关联,使格式与内容做到既可分离又可融合. 既能适应办公软件“所见即所得”的编辑要求,又保证了可以与其它应用很好地集成.

关键词: 文档格式; 文档处理; 办公软件; 可扩展置标语言
中图分类号: TP317.1 **文献标识码**: A **文章编号**: 0372-2112 (2007) 02-0375-04

Separation and Combination of Content and Appearance in Document Format

LI Ning, MU Yong-min, DONG Hui, FANG Chun-yan
(*Department of Computer Science and Technology, Beijing Information Technology Institute, Beijing* 100101, *China*)

Abstract: One of the original intensions of markup languages is to separate the content and appearance. However, in most nowadays document formats, XML Schema are defined to combine content with appearance, thus they do not match the target. To meet the requirements from setting up the format standard for Chinese office document, a new kind of document format is proposed in this paper. In the proposed format, the information about appearance and the information of user's data are separated but coexisted in the same document, while association is setup between these two. This document format can assure that the format data can either be separated from or combined with the user's data, which makes WYSIWYG editing and system integration successful.

Key words: document format; document processing; office software; extensible Markup language (XML)

1 引言

文档的编辑长期以来采用所见即所得 (What-You-See-Is-What-You-Get, WYSIWYG) 的方式. 人们希望从编辑器输出的文档及其排版格式可以原封不动地显示在屏幕上或打印到纸张上. 在今天看来,这种方式有很多弊端,一方面,作家和编辑各有分工,文档作者的排版能力很难达到专业的水平. 实际上,经作者排版的文章到了报章杂志编辑的手里一般都要再次排版,浪费了人力和时间;另一方面,有些出版部门要求作者自己排版后提交稿件,给作者带来诸多烦恼,很多作者的电脑里并没有规定的排版软件或字形字号;再者,媒体种类繁多,显示/打印设备千差万别,现实情况往往是“所见非所得”[1].

近年来,人们开始重新审视排版需求并探讨替代的方式. 一般认为,文档的编排可分为两种类型[2],一类是“所见即所得”的交互式编排;另一类是离线的自动排版方式,后者的前提是文档做到格式与内容分离,通过定义排版规则(式样单),并施加到不带格式信息的文档内容之上,从而得到排版后的文档显现效果. 这两类方式的适用对象不同. 第一类方式适合专业人员编排个性化的文档,例如,报章和杂志的版面等等,这一类编辑工具有很长的历史,产品十分丰富,如 Microsoft Office、Adobe FrameMaker 和 InDesign 等. 第二类排版需要事先定义规则(式样单),然后计算机自动将文档内容排成所需的格式. 这类排版也有广泛的用途,例如,为提取的文档自动加载格式、个性化阅读、多种介质上的文档显现、文档的高效存储与传送、文档与应用系统的集成等等. 第二类排版方式比起第一类有许多优势,例如,文档的作者不需要关心排版的效果,可以专注于编写文档内容;可以定义不同的式样单使文档排成不同的格式,不同的显现仅仅是式样单不同而已,无需修改文档;不同的显现介质都可有不同的式样单,对于特定的介质,文档可以找到最适合的式样单,最恰当地显现出来;给定排版规则,可以保证同类文档有一致的显现形式;文档的交换仅需传送内容,可以节省网络带宽和存储资源;文档系统与其他应用系统能够方便地集成,利于信息的共享和检索[3,4],等等.

收稿日期:2006-04-21;修回日期:2006-10-10
基金项目:北京市教育委员会科技发展计划(No. KM200510772003)

为了方便办公文档的信息交换,中文办公软件基础标准工作组正在制定国家标准《中文办公软件文档格式规范》(简称为 UOF,Uniform Office-document Format).它建立在可扩展置标语言 XML 基础之上.置标语言产生的初衷就是要做到内容与表现(格式)的分离[5].然而,作为文档编辑工具使用的文档记录格式,一方面要尽可能地支持"所见即所得"的友好的编辑风格,这要求格式与内容混合编排;另一方面为了支持与其他应用的集成,最好还能做到格式与内容分离,这两者本身存在一定的矛盾[6].目前,无论是国际上主流的、基于 XML 的办公文档格式(例如 OASIS Open Document Format ——ODF[7],Microsoft Office Open XML Format[8]),还是制定中的中文办公软件文档格式 UOF,所描述的内容主要是格式信息,文档的数据内容散布在格式信息之中.也就是说,为了支持办公软件"所见即所得"的第一类编排方式,办公文档格式大多采用内容与格式混排的方法.这种文档格式是否可以做到格式与内容分离呢?

2 在办公文档格式中支持用户数据的可能途径

办公文档格式中支持格式与内容分离的关键是要支持用户数据,即文档中真正的逻辑内容.例如,无论编排成表格形式还是文本形式,一个报价单中必然包含货品名称和价格,这就是用户数据.为了能够交换,现在用户数据也普遍采用 XML 来描述,并且同时提供用户定义 Schema(User Defined Schema,UDS)来规定数据的结构,以便对数据加以验证.电子公文[9~11]就是一种较为典型的用户 XML 数据.下面就以电子公文为例,阐述办公文档格式支持用户数据的可能的途径.

电子公文的主要结构包括"页眉"、"主体"和"版记"等等部分,如图 1.

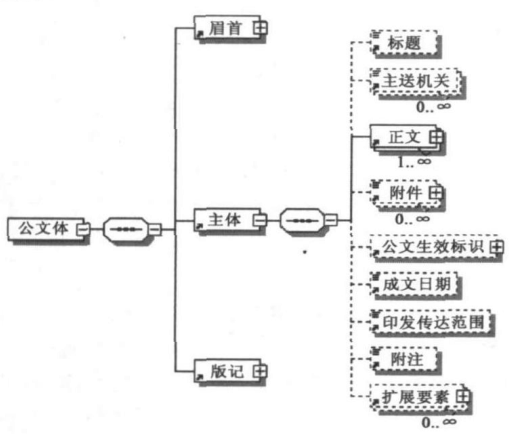

图 1 电子公文的结构片段*

作为办公文档,电子公文需要用 UOF 结构加以记录.UOF 文档一般由"节"构成,"节"可包含若干"段落",段落又可包含若干"句"."句"中的文本串是最终显现的内容.另外"节"、"段落"和"句"各有描述它们特性的属性.如图 2.

要使 UOF 能够描述电子公文的显现格式,同时能够从 UOF 文档中抽取电子公文数据,可以考虑以下几种方式:

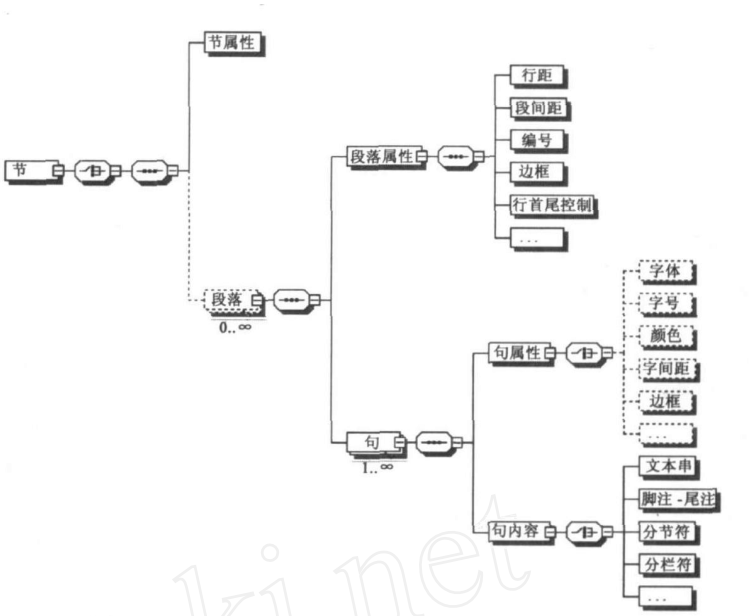

图 2 UOF 的结构片段**

(1) 依靠 Schema 的扩展能力

W3C XML Schema 规范留有一些机制用于 Schema 的扩展,主要是命名空间(Name space)以及"Any"元素与属性.在电子公文应用中,可以将 UOF 元素和用户数据分为两个命名空间,在 UOF 空间中预留出若干"Any",如"句内容"下允许出现"Any".通过该"Any"元素放置电子公文空间的"标题"和"主送机关"等信息.采用这种方式的好处是可以整体定义命名空间,并采用标准的 XML 分析器进行分析验证.但是这种方法也有致命的缺点.很显然,在 UOF 中采用"Any"来支持用户定义数据的最大的问题是不能做到各命名空间各自透明——必须顾及用户数据在 UOF 中允许出现的位置,而这在制定 UOF 的时候是不可能知道的.另外也会使 Schema 复杂化,并产生非限定性问题[12].因此,"Any"仅适合用于 UOF 的扩展,不宜使用这个方法来支持用户数据.

(2) 文档过滤

文档过滤是在分析器处理办公文档之前过滤元素标记(Tags),即在处理格式内容时忽略用户数据的内容,在处理用户数据时忽略格式内容.例如,在处理 UOF 格式时,忽略掉任何来自其它命名空间的元素,这样就可允许在 UOF 文档中随意插入电子公文空间的元素.这种方式的优点是比较灵活,命名空间透明,不相互影响.但是这种方法的最大问题在于,UOF 的编辑过程会打乱其它空间的元素结构.例如,如果将电子公文的元素插入在文字表格的单元格之中,对表格单元的任何排序都可能改变电子公文元素的逻辑结构,从而导致用户数据无法通过 UDS 验证.这个问题的根源在于内容对格式的依赖.Microsoft Office 2003 中对用户数据的处理就采用了文档过滤的方式[13],因此难以避免这个问题.可见标记过滤的

* 为清晰起见,对电子公文的结构作了简化.

** 为清晰起见,对 UOF 的结构作了简化.

方式并不理想.

3　解决方案

上述两种方法的表明,为支持用户数据将格式与内容信息合二为一并非上策.我们提出的方法是,将文档描述分为UOF格式树(格式描述)和UDS实例树(用户数据)两部分,通过关联机制建立起两者的联系,同时记录在文档之中.

为了支持用户数据,在UOF中建立了“用户数据集”,如图3.

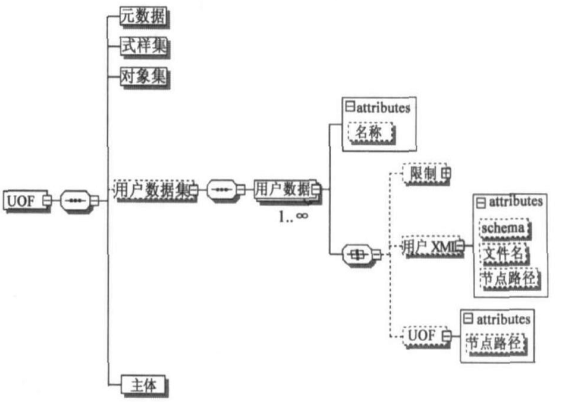

图3　UOF中的“用户数据集”

“用户数据集”作为一个UOF格式树和UDS实例树关联节点的索引,记录了每一对关联节点的对应关系.图中“用户数据”的子元素“用户 XML”包含三个属性:“schema”即UDS;“文件名”(可选)指明存放用户数据的XML文件;“节点路径”通过XPath[14]指向UDS实例树的节点.另一个“用户数据”子元素“UOF”则指向对应的UOF格式树节点.子元素“限制”用于指定具有这种对应关系的UOF文档节点是否可以修改、打印或浏览,以替代常规的“公文域”.子元素“名称”则用于命名映射关系.

图4是一个具体的电子公文UOF文档中的“用户数据集”内容.其中可以看到,电子公文“gongwen.xml”中的节点“/公文体/眉首/份数序号”对应于UOF格式树中节点“/uof:UOF/uof:文字处理/字:主体/”下第2个段落的第2句(/uof:UOF/uof:文字处理/字:主体/字:段落[2]/字:句[2]).

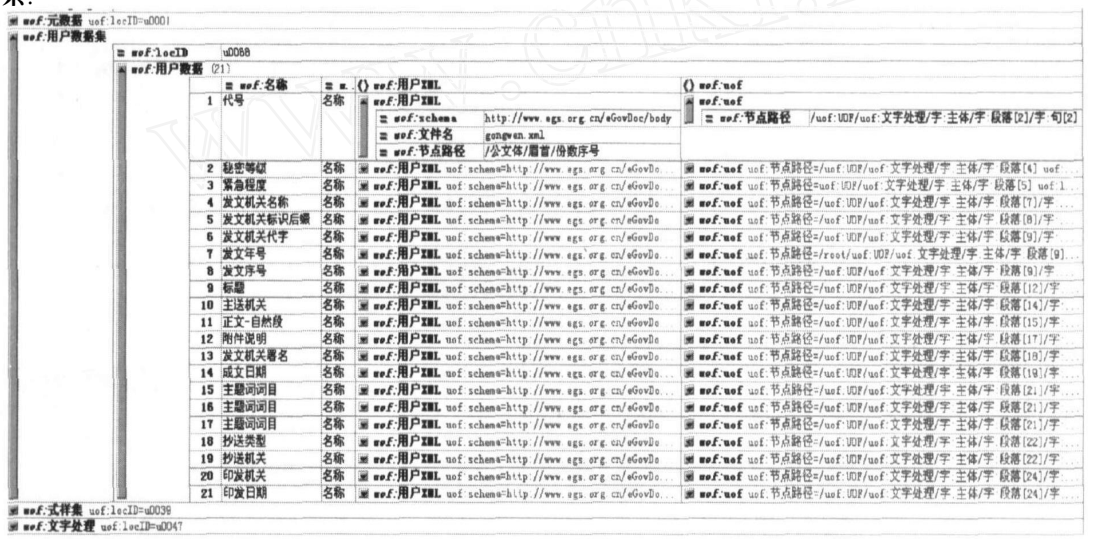

图4　一个具体的电子公文UOF文档中的“用户数据集”

通过这种映射方法,北京信息工程学院目前已经完成了从UOF提取用户数据从而生成电子公文XML数据的XSLT式样单(Stylesheet).实践证明这种方法是完全可行的.这个方法的优点是:文档的格式信息与用户数据互不影响,可以很方便地分别提取,并用UOF和UDS验证.因为有“用户数据集”显式指定对应关系,用户数据不再依赖于格式信息,避免了前述方案的缺点.另外,XML节点的对应十分灵活方便,可以把格式树的任何节点与任何用户数据对应.在实现中要注意的是,“用户数据集”的内容应在UOF格式树构造完成后生成.另外,办公软件的开发仍然需要考虑如何实现用户数据的导入导出,如何使用户可以随时建立用户数据节点和格式树节点的关联,如何在文档的编辑过程中维护这种关联,等等.由于篇幅限制,本文恕不详细讨论这些内容.

《中文办公软件文档格式规范》的设计充分考虑到对用户XML数据的支持,不但在UOF Schema中加入了“用户数据集”这样的机制,在文档存储格式中也考虑到了这一点,为中文办公软件厂商在开发中实现这一功能提供了必要的条件.

4　总结

《中文办公软件文档格式规范》即将发布,它将为办公文档的信息交换奠定基础.中文办公软件文档格式除了要支持“所见即所得”编辑方式,从而格式与内容混合编排之外,需要能够同时支持格式与内容分离,既保留第一类交互式排版直观、可读性强的优点,又使之具备第二类排版的优点,办公软件的处理能力便可大大增强.然而,这两类编排方式本身存在矛盾.现有技术中公开的文档记录格式一般是通过定义与格式相关的XML Schema来支持混合的文档格式与内容数据,不能很好地达到内容与表现分离的目的,难以与其它应用系统集成.针对上述问题,本文提出了一种新的文档结构.即在文档中记录文档格式信息和用户数据两个部分,并通过“用户数据集”建立两

者的关联,使格式与内容做到既可分离又可融合.既能适应办公软件"所见即所得"的编辑要求,又保证了可以与其它应用很好地集成.该方法已经被《中文办公软件文档格式规范》所采纳,并在电子公文处理的实际应用中获得了成功.

参考文献:

[1] 李宁.中文办公软件标准化几个问题的探讨[J].信息技术与标准化.2003,(12):48 - 50.

[2] Deach S. What is XSL-FO and when should I use it? [J]. The Seybold Report,2002,2(17):1 - 8.

[3] 吴劲,陈泽琳.基于部分匹配的 XML 文本文档向量检索模型[J].电子学报.2002,30(12A):2169 - 2171.
Wu Jin, Chen Ze-lin. Vector retrieval modeling using partial march pattern for text-rich XML documents[J]. Acta Electronica Sinica,2002,30(12A):2169 - 2171. (in Chinese)

[4] 徐海渊,吴泉源.贾焰.基于 Key 的 XML 连续查询算法[J].电子学报.2003,31(2):284 - 286.
Xu Hai-yuan, Wu Quan-yuan, Jia Yan. Key-based XML algorithm for continual query[J]. Acta Electronica Sinica,2003,31(2):284 - 286. (in Chinese)

[5] Alschuler L. ABCD. SGML, A User's Guide to Structured Information[M]. Boston: International Thomson Computer Press, 1995. 31 - 32.

[6] Goldfarb C, Prescod. Paul. XML Handbook (5th Edition)[M]. New Jersey: Prentice Hall PTR,2003. 350 - 394.

[7] OASIS. Open Document Format for Office Applications (OpenDocument) v1.0[S/OL]. http://www.oasis-open.org/committees/download.php/12572/OpenDocument-v1.0-os.pdf, 2006-04-15.

[8] Microsoft. Microsoft Office Open XML Formats Overview [R/OL]. http://www.microsoft.com/office/preview/developers/fileoverview.mspx,2006-04-15.

[9] GB/T 19667.1-2005,基于 XML 的电子公文格式规范 第 1 部分:总则[S].

[10] GB/T 19667.2-2005,基于 XML 的电子公文格式规范 第 2 部分:公文体[S].

[11] 段荣婷,徐维.基于 XML 技术的中国电子公文结构设计研究[J].现代图书情报技术.2004(1):67 - 71.

[12] Vlist E. XML Schema[M]. Sebastopol: O'Reilly,2002. 104 - 105.

[13] Lenz E, McRae M, St. Laurent S. Office 2003 XML[M]. Sebastopol: O'Reilly,2004. 144 - 147.

[14] W3C. XML Path Language (XPath) Version 1.0[S/OL]. http://www.w3.org/TR/xpath,2006-04-15.

作者简介:

李 宁 男,博士,1964 年生于北京,英国 University of Kent 信息技术专业博士,北京信息工程学院计算机科学与工程系副主任.研究方向为置标语言技术、中文信息处理、多媒体. E-mail: ningli@public2.bta.net.cn