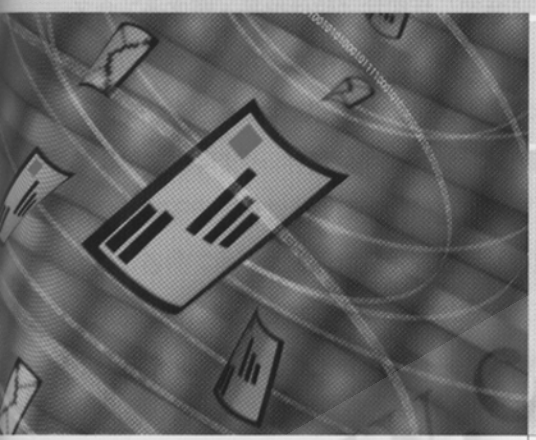
基于XML和工作流的期刊出版系统的设计与实现

Design and Implement of Journal Publishing System Based on XML and Workflow



邹振亚 （中央民族大学计算机科学与技术系 100081）

摘 要：本文分析了目前期刊出版发行流程和存在的问题，提出并设计实现了一套基于XML和工作流的出版发行系统，并对系统实现过程中所涉及的关键技术进行了分析研究。可以较好的解决出版物全文上网问题。

关键词：出版发行 XML 工作流 Apache Cocoon

1 引言

随着互联网的高速发展，信息共享和电子化逐渐成为时代的要求，传统的印刷媒体(新闻报纸等)，也正逐渐拓宽到网络化出版发行领域，目前的期刊网络化出版的实现方式基本是各大期刊网建立全文收录数据库，这种方式并不理想，因为各个编辑部的稿件编排格式、排版印刷软件有所不同，因此收录的时候部分期刊可以采用编辑部提供的电子版进行转换入库上网(如中国期刊网开发了CAJ格式，可以将方正排版文件转换生成CAJ文件，通过专用的阅读器阅读，或者可以转换为PDF格式)，而其余的期刊或编辑部未提交插图文件的，收录单位需先对样刊或其中插图手工扫描转换，这样既给编辑部和收录单位增加了工作量，又使封装文字和插图信息的电子版文档质量降低。

基于以上的现状，本文阐述了一种期刊出版发行的思路，设计和实现了一套基于XML和工作流的期刊出版系统。既实现了期刊编辑部的出版系统自动化，又实现了科技文档数据的标准化，快速高质量的收录入库和网络发行。

2 系统设计与实现

整个系统由Tomcat 4.1 + Cocoon 2.04 + J2SE1.41+SqlServer2000架设，采用Cocoon下的XSP编写。功能设计和数据流如图1所示。

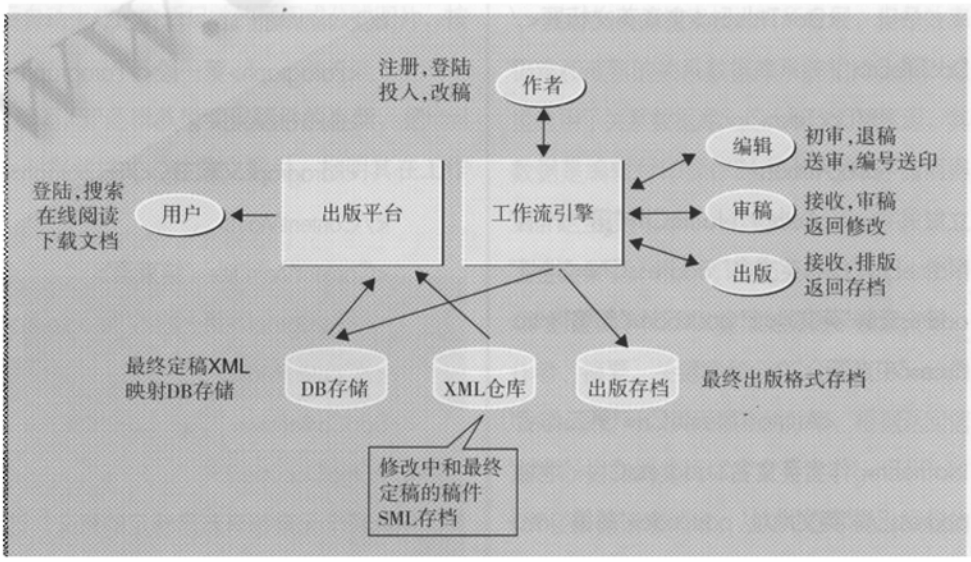


图 1

2.1 数据存储

2.1.1 格式选择

XML作为SGML的子集，最初就是为信息标准化所设计，近年来的不断的发展，相关技术已经日趋成熟。选择XML格式作为稿件存储格式有下述优点：

* XML树状层次信息结构存储稿件的内容非常适合，可以方便的提取索引，方便修改具体单项信息。

* 采用XML结构存储稿件，可以完全以内容为中心，从而分离了不必要的格式信息，而只要设计统一的XSL模板即可格式化为单独期刊具体页面格式。

* XML格式存储稿件更加有利于资料的共享和标准化，可以方便的转换为任意的格式(html，RTF，PDF，WML等)。

这里设计了下面的结构存储稿件，文字部分仅为示例：

```
<JournalDocument>
  <JournalName>期刊名称</JournalName>
  <SubmitTime>提交时间</SubmitTime>
  <EditStatic>编审状态</EditStatic>
  <DocTitle>
    <DocTitleCn>文章中文标题</DocTitleCn>
    <DocTitleEn>文章英文标题</DocTitleEn>
  </DocTitle>
  <DocAuthor>
    <Author NameCn="第一作者" NameEn="作者英文名" addressCN="地址" addressEn="英文地址" postcode="邮编" introduce="中文简介"/>
    <Author NameCn="第二作者" NameEn="作者英文名" addressCN="地址" addressEn="英文地址" postcode="邮编" introduce="中文简介"/>
    <Author NameCn="其他作者" NameEn="作者英文名" addressCN="地址" addressEn="英文地址" postcode="邮编" introduce="中文简介"/>
  </DocAuthor>
  <DocAbstract>
    <DocAbstractCN>中文摘要</DocAbstractCN>
    <DocAbstractEn>英文摘要</DocAbstractEn>
  </DocAbstract>
  <DocKeywords>关键字</DocKeywords>
  <DocLibID>中图法分类号</DocLibID>
  <DocID>文章编号</DocID>
  <DocContent>
    <ContentMain title="前言">
      <Paragraph>第一段</Paragraph>
      <Paragraph>第二段</Paragraph>
    </ContentMain>
    <ContentMain title="方案">
      <Paragraph>第一段</Paragraph>
      <illustration id="插图/公式/图表1"/>
      <Paragraph>第二段</Paragraph>
    </ContentMain>
    <ContentMain title="模型 ">
      <Paragraph>第二段</Paragraph>
    </ContentMain>
    <ContentMain title="实现">
      <Paragraph>第一段</Paragraph>
      <illustration id="2"/>
      <Paragraph>第二段</Paragraph>
    </ContentMain>
    <ContentMain title="结束语">
      <Paragraph>第一段</Paragraph>
    </ContentMain>
  </DocContent>
</JournalDocument>
```

结构每个元素都有自己特定的意义。这里DocContent主要存储了稿件的文字信息，这也是我们检索所主要关心的。对于科技文档中的图片、公式、图表由于目前Web浏览器的限制，则一律采用图片的解决方法，在xml文档中仅存储路径信息。至于如何编写和显示xml留给投稿等模块实现。

2.1.2 xml数据仓库

为了便于把大量的稿件存储管理，设计了简单的本地xml存储仓库。

* 已经出版稿件的xml文件以标准的“文章编号”命名；建立同名的文件夹用于存储插图，插图按“稿件编号＋顺序号”命名，建立每月出版稿件的索引xml文件，以年月命名。这一功能由编辑付印时自动生成。

* 修改中的稿件xml文件以“投稿编号“命名，建立同名的文件夹用于存储插图，插图按“投稿编号＋顺序号”命名，建立索引xml文件。这一功能由投稿时自动生成，编审中进行操作，出版前转存为出版稿件，修改相应索引。

2.1.3 出版存档

为了保存出版的印刷实际格式，建立了出版存档的仓库，采用xml文件建立索引，保存出版社的具体排版印刷文件格式。

2.1.4 检索数据库

为了网络出版检索的高效，采用了xml映射存储到数据库，因为这里xml设计的结构比较简单，所以可以很方便的映射存储到数据库。采用SqlServer2000作为检索数据库，建表如图2所示，通过XSP脚本把定稿付印的xml稿件分字段映射存储在数据库中。其中“DocContent”字段建立全文索引。

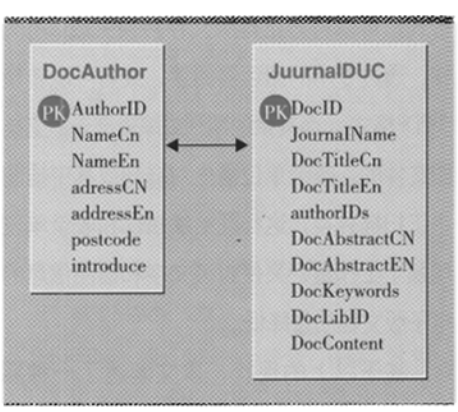


图 2

2.2 工作流引擎与出版平台设计

工作流引擎设计遵循编辑部运作模式编写引擎，目的是实现投稿编审出版流程自动化。图3所示为流程分析。

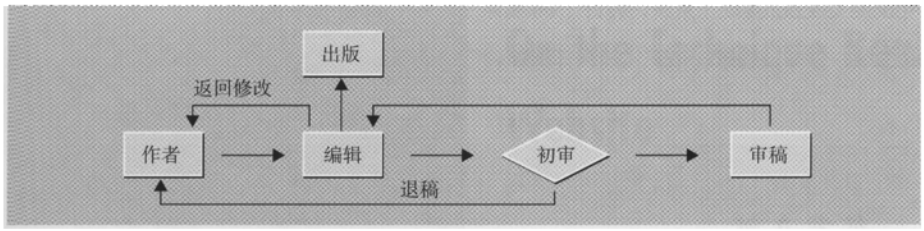


图 3

整个平台以开放源代码项目Apache Cocoon 2作为出版平台服务框架，Apache Cocoon是一个XML发布框架，它将XML和XSLT技术在服务器应用程序中的应用提升到一个新级别。Cocoon的宗旨是提升管道化SAX处理的性能和扩展性，通过对内容、逻辑和样式的分离来营造一个灵活的环境。它定义了一个标准的内容管理系统。

作为一个开发平台，Cocoon最吸引人的地方在于它提供了品种丰富的生成器、转换器和串行化器。这里主要使用了一下组件：DirectoryGenerator生成器可以将目录列表转换成XML格式，以便从中生成SAX事件，这样方便建立xml文件仓库索引；JSPGenerator生成器可以根据JSP页来生成XML和SAX事件，ServerPagesGenerator生成器可以根据XSP页来生成XML和SAX，便于编写工作流引擎；HTMLSerializer可以根据SAX事件来生成HTML响应；PDFSerializer可以根据SAX事件，使用Apache FOP格式输出处理器来生成PDF文档，实现网上发行。

2.2.1 投稿模块

使用JavaScript与html的iframe元素编写所见即所得的基于浏览器的客户端，支持IE3和Netscape6以上的浏览器。实现了以内容为中心的客户端编辑功能。作者只需按照向导的提示，一步步把自己的稿件填写即可，这样作者完全不用考虑繁杂的对文档的细致格式的设置(如粗细体，字号，对齐方式，图片位置等)。作者稿件(目前一般大部分用word编写)如果含有插图和公式，可以使用编辑软件的导出web页功能，目前的编辑软件底层都开始支持xml，但是各个厂家的导出格式不同，所以这里只需要其导出后自动生成的合适尺寸的插图或公式图片，按照投稿向导要求上载即可。

以上获取的是简单的html格式的稿件，通过编写XSP逻辑单来实现稿件的存储，把html稿件通过特定的xslt格式化为前面我们设计好的xml存储格式。并建立索引，标明其在工作流中的状态。

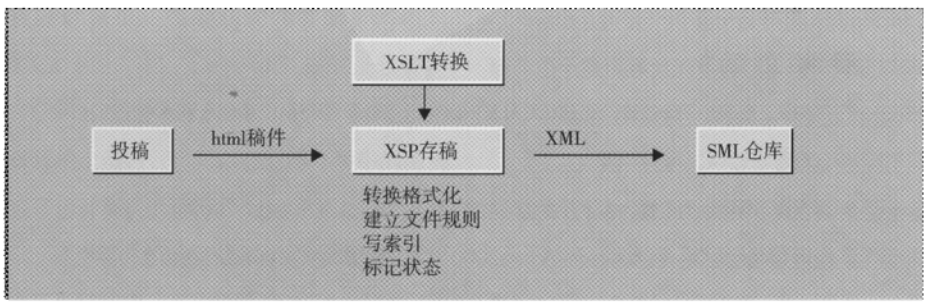


图 4

2.2.2 编审付印模块

编辑角色负责控制整个工作流。可以浏览xml仓库的索引，查看新稿件，这里采用xslt把xml格式化为html便于编辑网上浏览，可以确定退稿（删除存档，同时发信给作者）或审稿。对于审稿返回的稿件根据修改意见返回作者修改或定稿（设置文章编号）。在每月付印日期选择本月定稿稿件建立本月刊发文章索引确定付印(修改索引，标记文章状态，映射存储入检索数据库)。

发送给出版社的xml文档可以附加Fhtml.xslt（根据期刊印刷格式编写，可以用html的元素和xsl样式精确控制文本块和上下页标，使网页效果与印刷本显示相同）模板，在cocoon下转化为html格式便于编辑排版。或者采用cocoon的FOP，根据相应的xml2PDF.xslt（利用xsl-fo特性，根据期刊印刷格式编写，精确定位内容，设置字体，分栏页标等）模板，把本期xml稿件格式化为一个分页PDF文档，效果与期刊印刷版本一致，可以作为制版的对照。

2.2.3 出版平台

这一部分是选择cocoon 2平台的最大优势，这里通过编写xsp逻辑单建立了一个用户认证系统，方便给具有帐号的用户提供访问期刊全文的权限。

虽然xml的搜索技术有所发展，但是效率仍难和成熟的关系数据库系统相比，所以这里选用了关系数据库SqlServer作为搜索源，其数据是编辑付印操作时由xml仓库存储而得到。这里使用sqlserver对DocContent字段建立了全文索引，用户可以通过search.xsp（使用XSP的逻辑单和ESQL编写）页面，搜索文章、作者、摘要、关键字和全文。利用Cocoon的Cache设计可以实现很高的效率，根据返回信息可以提交浏览具体文章全文，getfullDoc.XSP会根据请求消息，从XML仓库提出相应的xml文档结合编写好的xml2PDF.xslt进行串行化，生成相应的PDF电子文档，提供下载。

3 改进的方案

本文只是单机的试验系统，如果要实现各种期刊的收录入库也可以很好的扩展，通过建立主收录服务系统，定期增量式同步各期刊的xml付印定稿和相应的期刊的格式化xsl文档，同步各期刊的搜索数据库建立主索引数据库，可以很方便的实现收录入库和网上发行。

基于Cocoon的平台可以更好的格式化xml文档为丰富的格式，可以考虑提供给各种客户端使用，如无线用户使用的wml ，甚至可以格式化为RTF文档格式。

随着XML相关技术的发展，目前办公软件逐渐的向底层支持xml化发展。可以考虑编写专用的投稿客户端软件，把公式用MathML，插图用SVG实现，使整个稿件作为xml存储更加集成化。

4 结束语

根据xml标准化的思路，这里提出了一种期刊一体化的出版方案，基于xml出版平台Apache Cocoon实现了系统，设计了xml稿件的生成、存储、检索、格式化的具体操作，实现了工作流程自动化，方便实现网上全文发布。随着xml技术的发展，相信科技稿件的标准化存储技术会逐渐成熟，全文的网络化发行系统也会日趋完善。

参 考 文 献

1 Roger Riggs Programming Wireless Decices with the Java 2 Platform[M],Micro Edition Addison Wesley 2001。

2 Qusat Mahmound Learning Wireless Java O'Reilly[M], 2001。

3 Yu Feng,Dr.Jun Zhu Wireless Java Programming with Java 2 Micro Edition[M] Smas 2001。

4 John W.Muchow Core J2ME Technology and MIDP[M] Sun Microsystems Press 2001。

5 王森 Java手机程式设计入门[M]，知城数位，台湾，2001。

6 焦祝军、张威，J2ME无线通讯技术应用开发[M]，北京电子希望出版社，2002。