

北京大学学报(自然科学版)
Acta Scientiarum Naturalium Universitatis Pekinensis
doi: 10.13209/j.0479-8023.2015.027

流式文档到固定版式文档的可逆转换方法研究

李宁^{1,2,†} 刘寅^{1,2} 梁琦^{1,2} 冯雪^{1,2}

1. 北京信息科技大学计算机学院, 北京 100101; 2. 数字出版技术国家重点实验室, 北京 100871;
† E-mail: ningli.ok@163.com

摘要 分析以往流式文档和固定版式文档一体化处理方法的局限, 提出一种可逆转换方法, 即将文档的转换信息记录于目标文档中, 以便于源文档的重构。还论述了可逆转换的原理、关键技术、实验结果、技术优势以及未来展望。选择标文通(UOF)作为流式文档格式, CEBX 作为固定版式文档格式, 利用该方法成功地实现了标文通文档到 CEBX 文档的可逆转换, 实验结果表明, 该方法取得了较好的效果。

关键词 版流一体化; 文档格式; 文档转换; UOF; CEBX

中图分类号 TP312

Research on Reversible Transformation from Re-flowable Document into Fixed-layout Document

LI Ning^{1,2,†}, LIU Yin^{1,2}, LIANG Qi^{1,2}, FENG Xue^{1,2}

1. Computer School, Beijing Information Science and Technology University, Beijing 100101; 2. State Key Laboratory of Digital Publishing Technology, Beijing 100871; † E-mail: ningli.ok@163.com

Abstract Considering the limitation of traditional methods to integrate re-flowable document and fixed-layout documents, a reversible transformation method was proposed which recorded the additional information used to restore the original document in the target. The authors discussed the principle, key techniques, experiment result, advantages and future work of the reversible transformation. Selecting UOF (Uniform Office-document Format) as the source re-flowable document format and CEBX as the target fixed-layout document format, a successful reversible transformation from UOF into CEBX was implemented. Experiment shows that the method has fairly good result.

Key words re-flowable document and fixed-layout document integration; document format; document transformation; UOF; CEBX

由于流式文档在浏览时按流式灌排的方式进行版面计算和绘制, 难以避免跑版, 即某种条件下看到的文档页数或对象的位置, 在另一种条件下会发生改变(例如多一页或少一页等)。固定版式文档在生成时已经把版面内容固定下来, 可以有效避免跑版问题。但是, 固定版式文档也存在着难以克服的缺点, 例如不方便编辑修改, 检索效率不高, 难以进行信息提取或与其他应用结合, 特别是不易实现

电子出版所需要的版面重排功能^[1]。

为了实现从固定版式文档中还原流式信息, 最早进行的是版面理解方面的研究。版面理解一般包含两个过程: 首先通过版面分析, 对版面内的图像、图形、表格、文本信息的区域和位置关系进行自动分割和判断; 然后通过版面理解, 将版面的物理几何结构映射为逻辑结构(将切分出的区域加上逻辑标签)^[2-4]。这些方法的缺点是: 结果的准确性

数字出版技术国家重点实验室开放课题(BX9161123405)和北京市属高等学校创新团队建设与教师职业发展计划项目(IDHT20130519)资助

收稿日期: 2014-06-26; 修回日期: 2014-10-25; 网络出版时间: 2014-12-01 10:20

不易保证;算法一般针对特定的文档类型,无法适用于灵活多变的结构;采用内置的规则或语法,难以灵活修改,或通过样本学习加以改进;更主要的是,版面理解的重点是判断区域的上下文关系和层次嵌套,识别的粒度较粗(最多为段落级别),不能满足流式文档编辑的需要。

为了能够将流式文档和固定版式文档相结合,扬长避短,近年研究比较多的方案是,扩展现有的固定版式文档格式或者流式文档格式,即:1)以固定版式文档中精确定位的版面描述为基础,在其中附加流式逻辑结构信息,用于面向流式的应用(如重排、抽取表格结构等)^[1,5-7];2)在流式文档中规定更为精确、丰富的版面格式化方法(如特定的样式、确定的排版算法和规则等),使得流式文档在不同环境下的显现效果趋近一致^[8]。这两种方法有一定成效,但由于难以在同一份文档中等同兼顾固定版式信息和流式信息,很难直接将二者进行简单融合。特别是,由于固定版式文档和流式文档长期独立发展,有大量的固定版式和流式文档遗存下来,很难从源头上融合二者。

分析固定版式文档的形成过程,可以发现,虽然固定版式文档可以借助工具直接生成,但是绝大部分的固定版式文档是从流式文档转换而来的。特别是在办公文档处理领域,借助办公软件进行文档排版,然后生成固定版式的公文进行发布,是最为普遍的方式。由于流式文档处于固定版式文档的上游,编辑语义一般不被固定版式文档所支持,因此在流式文档转换成固定版式文档的过程中,编辑语义一般是被丢弃的,这决定了流式文档一旦被转换成固定版式文档,就难以还原为流式文档^[9]。

我们的想法是,在流式文档转换成固定版式文档的过程中,除了将固定版式文档格式支持的版面内容转换过来以外,将一些关键的转换信息也记录在目标格式中。当需要对固定版式文档进行编辑时,借助这些转换信息,从固定版式文档重构出流式文档,实现流式文档到固定版式文档的可逆转换。本文选择标文通(UOF, Uniform Office-document Format)作为源流式文档格式,选择CEBX(Common E-document of Blending XML)作为目标固定版式文档格式,尝试进行可逆转换。转换中仅利用CEBX允许的用户自定义数据记录转换信息,而未对CEBX和UOF格式做任何改变。

1 UOF 与 CEBX 文档格式

流式文档一般包含元数据、式样、书签、超级链接、对象、节(最大的排版单元,不同页面式样的文档内容形成不同的分节)、段落、句及其他元素和属性。这些内容按一定的层次结构进行描述,形成流式文档格式。

标文通是我国自主知识产权的中文办公文档格式标准,简称UOF^[10-11]。UOF采用XML作为文档格式描述语言,包括字处理、电子表格和演示文稿三部分,形成独立、完整、开放及可扩展的文档描述体系,得到各主流办公软件的支持,如永中Office、金山WPS、Microsoft Office及OpenOffice.org等。UOF目前有3个版本:UOF 1.0、UOF 1.1和UOF 2.0(标准草案)。其中UOF 1.0和UOF 1.1是单文件结构,UOF 2.0是多文件结构,以ZIP打包的形式保存各个文件。本文针对UOF 2.0字处理部分进行研究,其简化的文档结构如图1所示。

固定版式文档一般包含以下内容:文件描述对象、组对象、页面集合、页面对象、活动对象、图片对象及字体对象、流对象、数字对象以及引用机制等。

CEBX即XML混成的公共电子文档,是北京大学与北京方正阿帕比技术有限公司于2009年推出的一种独立于软件、硬件、操作系统和呈现设备的文档格式标准^[12-13]。CEBX采用“容器+文件”的方式来描述和存储数据,容器类似一个虚拟存储系统,将各类数据描述文件组织起来,并提供特定的访问控制接口。在CEBX中,数据描述采用XML技术,以版式描述信息为基础,辅以版面对象的结构化信息,然后经过压缩放入容器中。

CEBX文档分为4个层次:物理层、资源层、版式信息层和流式信息层。其中物理层是CEBX文档的数据存储形式。在CEBX中,用XML描述数据文件,所有的数据文件按XDA(XML-based document archive)文档打包结构描述并存储。资源层表示文档中的各个图元对象,包括文字和图像等。版式层通过页对象组成页树(PageSet),在每个页面内,使用资源定义的图元按块(PageBlock)形成页面中的内容(Content)。简化的CEBX文档结构如图2所示。

UOF文档到CEBX文档的转换,主要是建立

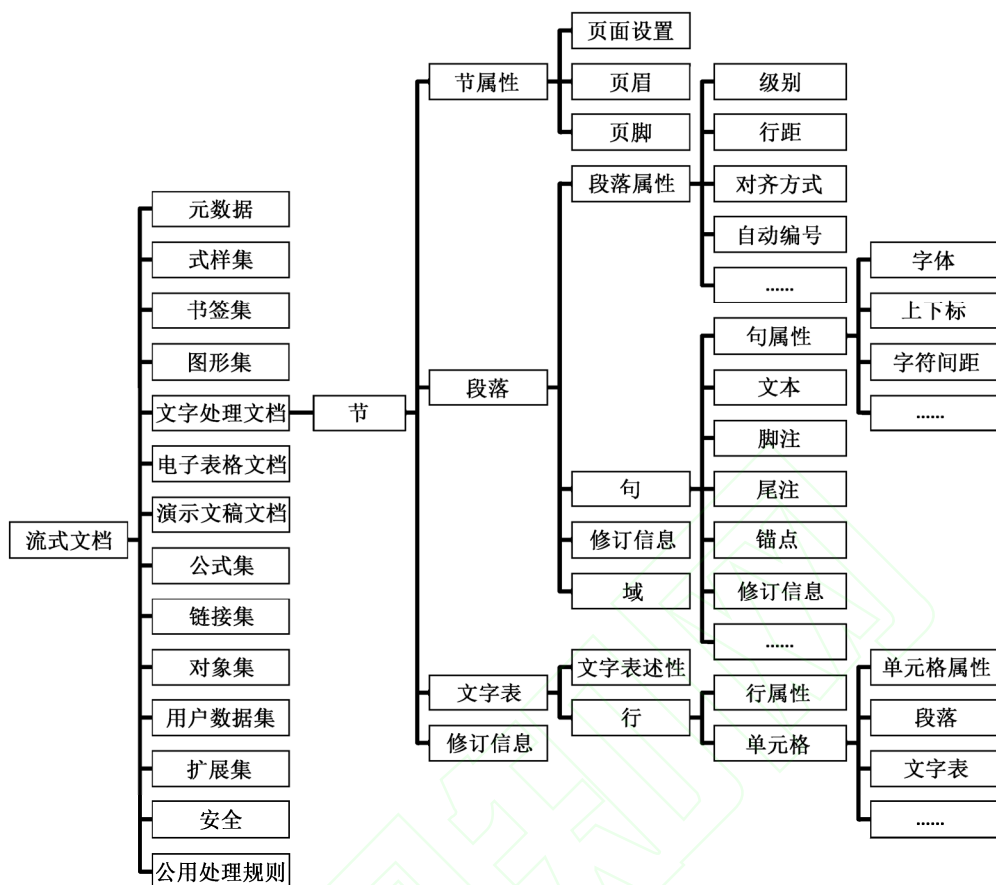


图 1 简化的 UOF 流式文档结构
Fig. 1 Simplified structure of the re-flowable document UOF

UOF 文档内容到 CEBX 文档各个页面内容的映射。

为了记录必要的转换信息，需要在 XDA 包中增加相关的数据文件。读者可能认为 CEBX 中的流式信息层(图 2 中的 StructureRoot)描述文档的流式结构和样式信息，理应可以利用它来记录相关的转换信息^[7]。但是，经过分析，CEBX 中的流式信息层主要用于内容的重新排版，以实现屏幕自适应的目标。它是预设的，不符合任何一个流式文档格式标准，我们无法利用它来记录转换信息。

CEBX 中，允许用户在压缩包中加入自己的数据文件。为此我们构造了一个 UOF 集用于记录转换信息，并将其集成到 XDA 打包结构中，如图 3 所示。

UOF 信息集包含重构 UOF 所需的信息，具体含义如下：1) UOF_INFO 目录，存放 UOF 信息集的目录；2) _content.xml 文件，为源 UOF 文档去除文本后的逻辑框架信息，包含分节、段落、句等编辑

语义信息；3) record.info 文件，记录 UOF 转换到 CEBX 文档过程中的转换信息，以实现可逆转换；4) _styles.xml 文件，为 UOF 的式样集，记录源 UOF 文档中的式样信息。

2 UOF 到 CEBX 文档的可逆转换方法

实现 UOF 到 CEBX 文档的可逆转换有一种简单的方法，即在保存 CEBX 目标文档的同时，把 UOF 源文档也完整地保存下来，将它们打包成一个文件，编辑时直接提取出流式文档来处理。该方式固然简单，但是难以保持目标文档与源文档的一致性；特别是，源文档与目标文档中所有内容均重复保存两份，造成资源浪费。

本文提出的转换方法尽可能地共享源文档和目标文档中的信息。例如，对于文本和图像等内容数据，仅在 CEBX 中保存，UOF 逻辑框架中仅含有流式文档的逻辑结构，在重构时才把 CEBX 中保存的

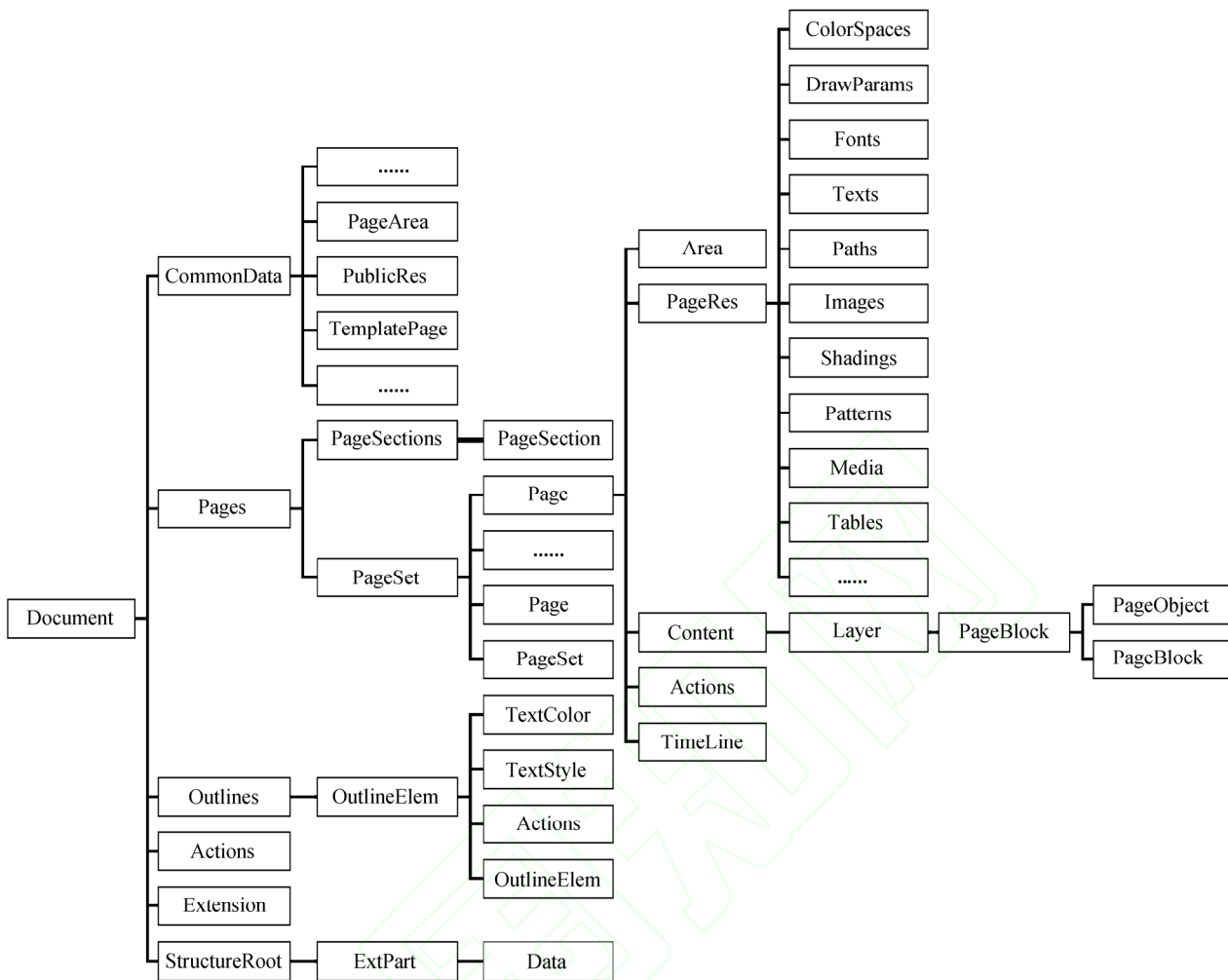


图 2 简化的 CEBX 固定版式文档结构

Fig. 2 Simplified structure of the fixed-layout document CEBX

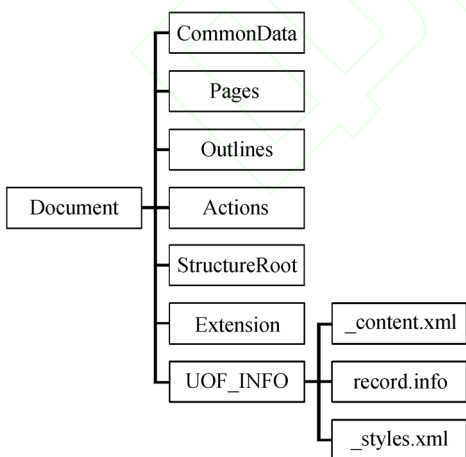


图 3 集成 UOF 信息集的 CEBX 文档

Fig. 3 CEBX with UOF information integrated

内容数据回填到 UOF 逻辑框架中,还原出源 UOF 文档。因此,在转换的开始,要进行 UOF 的框架提

取,主要是去除 UOF 中的文本内容和多媒体数据内容,并将它们在源文档中的位置与转换后 CEBX 文档中的相应内容数据建立映射关系。

实现 UOF 到 CEBX 文档的可逆转换,重点要处理好三部分内容的映射,即段落、列表和多媒体数据。

2.1 段落的映射

文字是组成文档的基本内容,因而也是转换的重点。在 UOF 中,组成段落的基本成分是句(run)。而在 CEBX 中,各个图元(object)都是相互独立的,没有诸如句和段落之类的逻辑关系。因此如何将 CEBX 中的文字关联到源 UOF 文档中的句和段落是首要解决的问题。

虽然在 CEBX 中没有段落的概念,但是 CEBX 中有块(Block)的概念。为了解决上述问题,我们将

UOF 中的每个段落映射到 CEBX 对应的块中，再将段落中的句映射到 CEBX 块内的图元，便可以确定哪些图元内的文本来自于源 UOF 文档中的同一个段落。将来从 CEBX 文档中提取文本时，便可以以 CEBX 文档中的块内图元为单位，依次还原源 UOF 文档中的句和段落，将其填入 UOF 逻辑框架中，实现 UOF 文档内容的重构。

这里遇到的困难是，当 UOF 文档中的某个段落，或段落中的句在 CEBX 中跨页出现时，情况会变得复杂，如图 4 所示。

CEBX 在换页时需要结束当前页面，并初始化一个新的页面。在新增的页面中，上一页未写完的段落中的句会写到新页面里的新块中，而上一页未写完的句也会被写入新块里的新图元，因此会造成一对多的映射关系，难以还原上层的结构。

为解决上述问题，转换过程中构造每个页面时，每写完一个段落，就累加一次总的块数，每写完一个句，就累加一次总的图元数，将之记录于 record.info 中。再将实际页面中的块数和图元数与 record.info 中记录的块数和图元数进行对比，便可以准确判断出跨页的句或段落，并加以重构。在每个页面中重构 UOF 段落的算法如下。

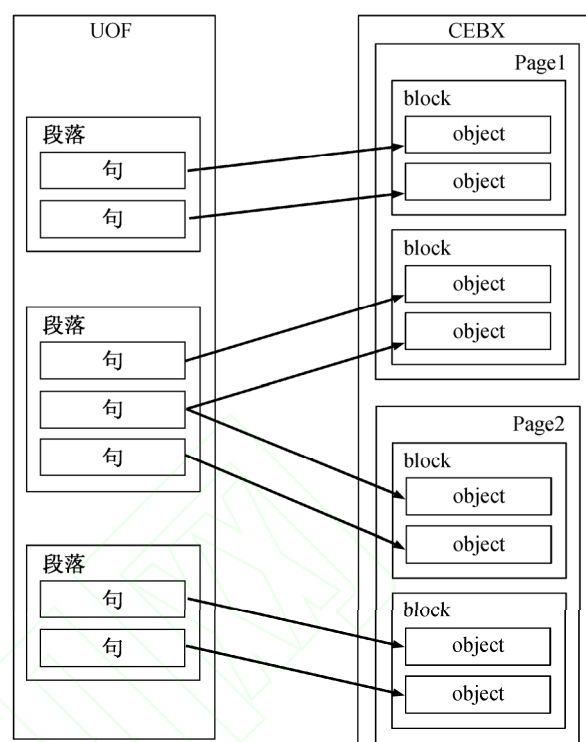


图 4 UOF 中段落和句的跨页映射

Fig. 4 Mapping of paragraphs and runs in UOF crossing pages

```

For each Page
{
    Let  $n_b$  be the number of Blocks recorded in record.info during transformation
    Let  $n_b'$  be the number of Blocks found in current Page
    Let  $n_o$  be the number of Objects recorded in record.info during transformation
    Let  $n_o'$  be the number of Objects found in current Page
    If  $n_b = n_b'$  // There is no UOF-Paragraph crossing Pages
    Then
    {
        // Construct UOF-Paragraphs
        For each Block
        {
            Take each Object as UOF-Runs
            Construct UOF-Paragraph by all the UOF-Runs
        }
    }
}
Else //  $n_b' = n_b + 1$ , there is UOF-Paragraph crossing pages
{
    // Construct first UOF-Paragraph
    If  $n_o' = n_o$  // There is no UOF-Run crossing pages
    
```



```

Then
{
    Take each Object in last Block in last Page as UOF-Run
    Take each Object in first Block in current Page as UOF-Runs
    Construct first UOF-Paragraph by all the UOF-Runs
}
Else //  $n_0' = n_0 + 1$ , there is UOF-Run crossing pages
{
    Take each Object except last in last Block in last Page as UOF-Runs
    Take last Object in last Block in last Page plus first Object in first Block in current Page as UOF-Run
    Take each Object except first in first Block as UOF-Runs
    Construct first UOF-Paragraph by all the UOF-Runs
}
// Construct rest UOF-Paragraphs
For each Block except first
{
    Take each Object as UOF-Runs
    Construct UOF-Paragraph by all the UOF-Runs
}
}
}

```

2.2 列表的映射

UOF 的列表是用自动编号集表示的。每个列表项的编号由“段落属性”中的“自动编号信息”以及相关属性控制。另外，UOF 中文档的大纲也是通过列表实现的，UOF 中对段落进行的不同层次的编号决定该段落所在的章节层次，因此实现 UOF 列表的重构是转换的另一个重点。

在 CEBX 中，列表的编号和列表项的内容并没有区别。UOF 与 CEBX 列表的对比如下。

显现式样：

1. 计算机科学
2. 软件工程

UOF 表示：

```

<字:段落_416B>
<字:段落属性_419B 式样引用_419C="id00000">
    <字:自动编号信息_4186 编号引用_4187="bn00010" 编号级别_4188="1"/>
</字:段落属性_419B>
<字:句_419D><字:文本串_415B>计算机科学</字:文本串_415B></字:句_419D>
</字:段落_416B>
<字:段落_416B>

```

```

    <字:段落属性_419B 式样引用_419C="id00000">
        <字:自动编号信息_4186 编号引用_4187="bn00010" 编号级别_4188="1"/>
    </字:段落属性_419B>
    <字:句_419D><字:文本串_415B>软件工程</字:文本串_415B></字:句_419D>
</字:段落_416B>
CEBX 表示:
<Text ID="100" Font="98" Size="14">
    <TextCode x="30.37" y="167.616">1.</TextCode>
</Text>
<Text ID="103" Font="98" Size="14">
    <TextCode x="37.77" y="167.616"> 计算机科学
</TextCode>
</Text>
<Text ID="108" Font="106" Size="14">
    <TextCode x="30.37" y="177.216">2.</TextCode>
</Text>
<Text ID="111" Font="106" Size="14">
    <TextCode x="37.77" y="177.216"> 软件工程
</TextCode>

```

</Text>

因此，仅仅从 CEBX 中无法辨别某个图元是普通段落还是列表编号，导致无法从 CEBX 文档中重构 UOF 的列表。为此，在 UOF 文档到 CEBX 文档的转换过程中，记录转换后的列表编号所在页面的页码以及页面坐标(如下)，以便在重构 UOF 列表时，可以分析出哪些内容是列表编号，从而解决上述问题。

```
<Pages>
  <Page>
    <number>1</number>
```

```
<list_label><x>30.37</x><y>167.616</y></list_label>
```

```
<list_label><x>30.37</x><y>177.216</y></list_label>
```

```
</Page>
```

```
</Pages>
```

2.3 多媒体数据的映射

在 UOF 和 CEBX 文档格式中，多媒体数据的表示方式比较接近，都是通过引用外挂多媒体文件来实现的。

在 UOF 中，所有多媒体文件都保存在 data 目录中，并具有相应的路径名。在 UOF 的对象数据集(objectdata.xml)中，相应会有一个唯一标识符，指向相应的文件。在文档内容中，通过对象数据集的标识符引用具体的多媒体数据。

与 UOF 不同，CEBX 文档的每个页面都有一个 Res 目录，其中存放该页所使用的多媒体文件，每个文件都具有文件名。在页面资源描述文件(PageRes.xml)中相应存在一个元素标识符指向相应的文件。在页面内容(Content.xml)中，块(PageBlock)中的图元(PageObject)通过相应资源的元素标识符引用具体的多媒体数据。

UOF 和 CEBX 文档中，一般多媒体内容的表示方式见表 1。

本文的转换过程中，目标 CEBX 文档是通过调用 CEBX Builder Kernel 中的接口生成的。在资源描述文件中，资源的标识符是根据内容出现的顺序自动生成的。相应地，所引用多媒体资源的文件名是根据上述标识符自动生成的，二者都无法自定义。而 UOF 中是先有多媒体资源，再生成多媒体标识符，多媒体数据文件名和标识符都可以任意指定。正因为如此，在重构 UOF 时，首先，将所有页

表 1 UOF 与 CEBX 的多媒体内容表示方式

Table 1 Representations of multimedia content in UOF and CEBX

文档类型	表示方式
UOF	/data ----- <对象数据 标识符="obj001"> <路径>/data/image001.jpg</路径> </对象数据>
	Graphics.xml ----- <图形 标识符="img001"> <图片数据引用>obj001</图片数据引用> </图形>
	Page_n/Res ----- image001.jpg ----- <Image ID="01">
	Page_n/Page Res_1.xml ----- <ImageFile>image001.jpg</ImageFile> </Image>
CEBX	----- <PageBlock> <PageObject BaseRef="01"/> </PageBlock>

面 Res 目录下的多媒体文件拷贝到 UOF 框架中的 data 目录下；之后，创建数据对象集文件 objectdata.xml，在对象集中按顺序定义所有数据对象并设置标识符引用；最后，从前到后按顺序遍历 UOF 框架中所有的段落锚点中的多媒体数据引用，综合标识符引用、数据对象和 CEBX 中的多媒体信息，重构 UOF 多媒体文档内容。

UOF 中的其他对象，例如表格、页眉、页脚、表注、图注、链接等，与 CEBX 中版块(Piece)的类型有较好的对应，因此不难从这些 CEBX 对象中重构 UOF 的相应内容。

3 实验结果

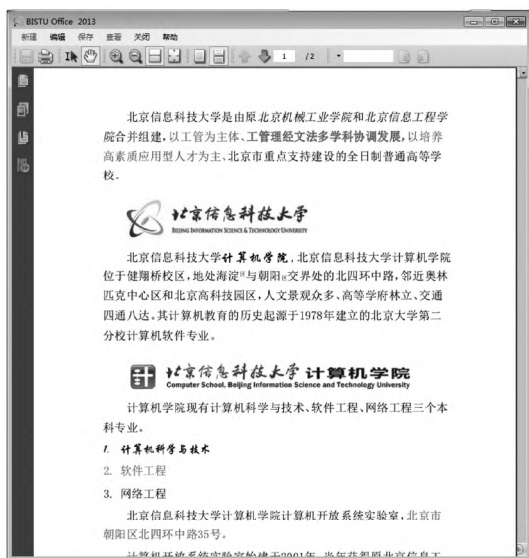
下面从使用效果、存储效率和转换速度 3 个方面展示实验结果。

图 5 为一个样本文档从 UOF 格式转换为 CEBX 格式，并在 CEBX 浏览器中看到的效果，以及从该 CEBX 文档重构 UOF 文档后，在编辑工具(永中 Office 插件)中看到的效果。主要说明的是，在用户界面中，重构过程是用户透明的，用户体验如同直接打开 CEBX 文档进行编辑。

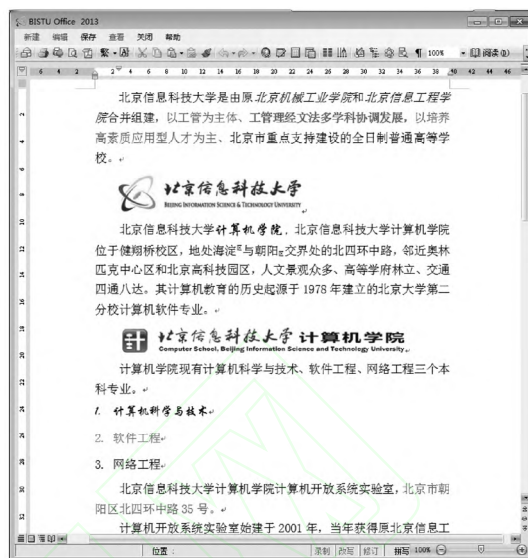
从图 5 可以看到，原来流式文档的编辑语义均得到较好的保留。对比 Microsoft Office 2013，同

样内容的 OOXML^①文档通过 Microsoft Office 2013 转换为 PDF 文档, 再从 Microsoft Office 2013 读入

该 PDF 文档进行编辑, 反复 4 次后其结果如图 6 所示。



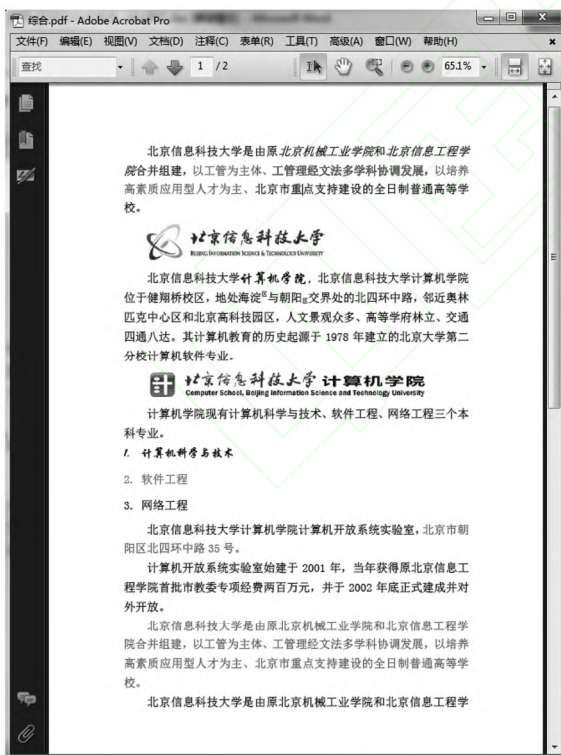
(a) 在CEBX浏览器中的效果



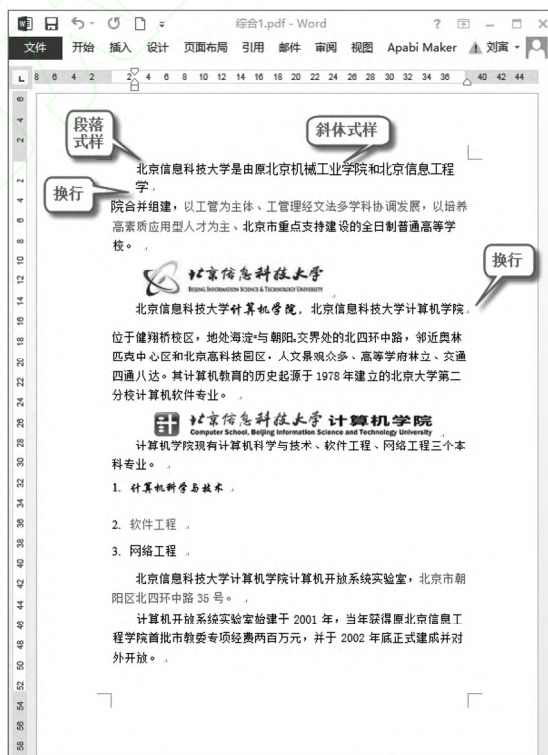
(b) 在编辑工具中的效果

图 5 带转换信息的 CEBX 样本文档的浏览和编辑效果

Fig. 5 Browsing and editing effect of sample CEBX document with transformation information



(a) 在PDF浏览器中的效果



(b) 在Microsoft Office 2013中的编辑效果

图 6 PDF 文档的浏览和编辑效果

Fig. 6 Browsing and editing effect of PDF document

① Microsoft Office 所使用的开放文档格式国际标准。

从图 6 可见, Microsoft Office 2013 通过版面理解重构出来的流式文档, 在很多地方发生了编辑语义的改变。而我们的方案不论反复读写几次均不会出现上述问题。

为测试存储效率, 我们选取 5, 10, 50, 100, 500 和 1000 页的文本类 UOF 文档进行实验, 转换成 CEBX 文档后, 转换信息所占的比例如图 7 所示。

从图 7 可见, 转换信息在目标 CEBX 文档数据中仅占很小的比例, 并且随着文档大小的增加, 所占比例越来越小。对于含有多媒体内容的文档, 实验结果与文本类的基本一致, 因篇幅关系, 此处不再赘述。

为测试重构 UOF 文档所用的时间, 分别选取不同大小的, 含文本类内容和图片内容的 CEBX 文档进行实验, 结果如图 8 所示。

从图 8 可见, 当文档中仅含有文本内容时, CEBX 文档的大小与重构时间基本呈线性正比关系, 基本处于可接受的范围之内。当文档仅含图片

内容时, 重构时间受 CEBX 文档大小的影响越来越小。虽然重构时间依然随着 CEBX 文档的增大而增加, 但是增长的速度越来越慢。随着多媒体内容的增加, 多媒体数据的处理(解压与压缩)成为影响重构时间的主要因素。

4 结论

本文选择 UOF 和 CEBX 文档, 尝试在流式文档向固定版式文档转换时, 在目标文档中增加转换信息, 以便从目标文档中重构出源文档, 实现流式文档到固定版式文档的可逆转换, 达到对固定版式文档进行再次编辑的目的。该方法无需对源 UOF 文档格式和目标 CEBX 文档格式作任何改变, 突破了以往版流一体化方法的局限。实验结果表明, 该方法效果良好。

这个方法亦可用于其他的文档格式转换。大多数不同类型的文档格式经过转换后, 均存在不同程度的信息损失。将转换信息记录于目标文档格式中, 实际上是将损失的信息保留下来, 用于日后重构源文档, 可以实现近乎无损的可逆转换。

本课题选择在国内具有一定应用基础的文档格式进行研究, 具有一定的现实意义和应用价值。UOF 是我国的办公文档格式国家标准, CEBX 是方正公司的通用电子文档格式, 均具有自主知识产权, 实现这两种格式的可逆转换亦有助于自主标准的推广和文档处理产业链的形成。

本研究在一些方面还有待进一步深入。首先, 文档之间的可逆转换目前依赖于目标文档的生成软件和重构软件, 如果转换信息以及转换软件是私有的, 则不利于文档间持久的互操作。因此需要对

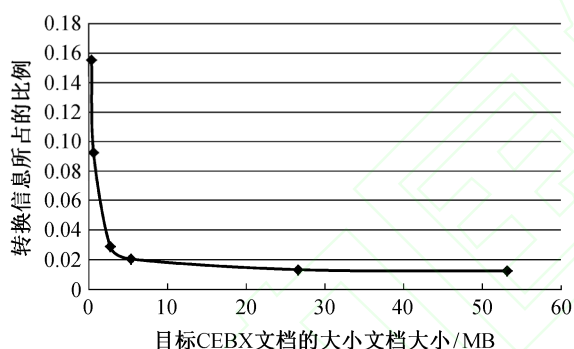


图 7 转换信息在目标 CEBX 文档中所占空间的比例
Fig. 7 Space proportion the transformation information occupied in the target CEBX file

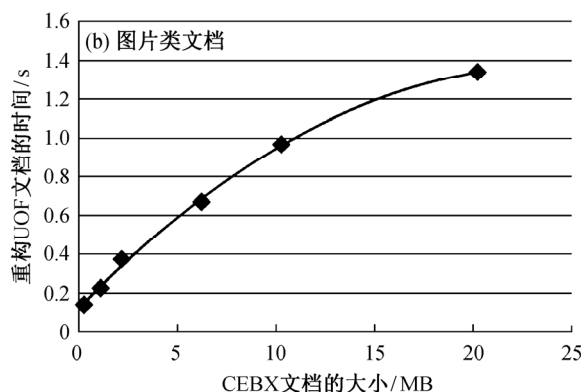
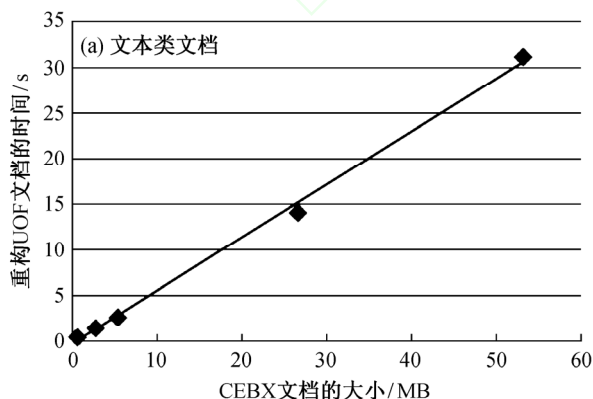


图 8 不同内容的文档大小与重构时间的关系

Fig. 8 Relationship between file size and reconstruction time for documents with different content

转换信息的记录结构进行标准化,这样可以促成开放的可逆转换算法。其次,转换方法需要考虑更多的因素,例如,对于跨页的表格处理以及复杂公式的可逆转换等,目前尚未深入研究。再者,由于式样的复杂性以及 UOF 与 CEBX 在式样描述上的差异,加之这方面 CEBX 未能提供完善的接口,迫使转换程序要在 CEBX 文档中完整保留 UOF 的式样信息,导致一部分数据的冗余。这要希冀不同的文档格式标准的制定者未来进一步合作,制定出更加和谐完善的标准体系。

参考文献

- [1] 李宁, 田英爱, 侯霞, 等. 办公文档与固定版式文档格式关系探讨. 电子学报, 2008, 36(B12): 128 - 132
- [2] Mao S, Rosenfeld A, Kanungo T. Document structure analysis algorithms: a literature survey. SPIE Electronic Imaging, 2003, 5010: 197-207
- [3] Nagy G. Twenty years of document image analysis in PAMI. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000, 22(1): 38-62
- [4] Namboodiri A M, Jain A K. Document structure and layout analysis // Digital Document Processing. London, 2007: 29-48
- [5] Hardy M R B, Brailsford D F. Mapping and displaying structural transformations between XML and PDF // Proc The 2002 ACM Symposium on Document Engineering. New York: ACM Press, 2002: 95-102
- [6] Hardy M R B, Brailsford D F, Thomas PL. Creating structured PDF files using XML templates // Proc The 2004 ACM Symposium on Document Engineering. New York: ACM Press, 2004: 99-108
- [7] 仇睿恒, 汤帜. 一种在电子出版中融合固定版面与流式信息的方法. 电子学报, 2012, 40(11): 2267 - 2281
- [8] 荣明军, 梁金晶. 一种 UOF 文档保存和还原版式的方案设想. 北京信息科技大学学报: 自然科学版, 2010, 25(S2): 23 - 26
- [9] 李宁, 梁琦, 侯霞, 等. 文档互操作性度量. 北京信息科技大学学报: 自然科学版, 2011, 26(2): 6 - 12
- [10] 中文办公软件基础标准工作组. GB/T 20916-2007. 中文办公软件文档格式规范[S]. 北京: 中国标准出版社, 2007
- [11] 李宁, 吴新松. 中文办公软件文档格式规范(1.0、1.1 版)使用指南. 长沙: 湖南师范大学出版社, 2010
- [12] 汤帜, 仇睿恒, 王毅. CEBX: 新一代结构化版式文档技术. 北京信息科技大学学报: 自然科学版, 2010, 25(S2): 11 - 15
- [13] 汤帜, 刘丽, 丁力, 等. CEBX v1.2 标准手册. 北京: 北京北大方正电子出版社, 2012