

# 基于双重签名的 WORD 文档在线处理方案

On-line WORD Documents disposing system based on double signature

许淑华 (绍兴文理学院数理信息学院 浙江绍兴 312000)

齐鸣鸣 (绍兴文理学院工学院 浙江绍兴 312000)

**摘要:**如何处理基于 Web 的在线 Word 文档及其安全性是网络化办公系统的两个重要内容。设计了集成客户机的 Word2003 和 MSPaint 软件资源的两个控件,结合可视化的手写签名和数字签名,提出了一个基于双重签名的 Word 文档在线处理方案。

**关键词:**控件 可视化 手写签名 数字签名

## 1 引言

微软的 Word 是功能强大的办公软件,为人们普遍所熟悉和使用。随着基于 Web 的自动化办公系统在企事业单位中的广泛运用,在线处理和共享 Word 文档显得非常必要。当前,微软已经提供了相关的控件支持 Word 文档的在线处理。这种处理方法简单实用,不足之处主要表现在:

(1) 把 Word 文档直接存放在服务器的发布目录下。因此发布目录的读写权限设置复杂,不易维护,也容易给服务器带来安全隐患。

(2) 由于没有存放在相应的数据库中,Word 文档不易管理,容易被篡改,不安全。

(3) 采用无压缩的二进制数据形式进行传输,效率不高。

目前,已经有多篇文献在单机环境下<sup>[1,2]</sup>和在 Web 环境下<sup>[3]</sup>讨论 Word 的开发和应用。这些文献叙述没有涉及网络环境下的 Word 文档的在线处理和鉴定控制措施。本文根据实际应用项目,从分布式的角度出发,详细论述了如何设计控件调用客户端本地环境的 Word2003 和 MSPaint 的软件资源。并在数字签名的基础上,分析了文档的双重签名和验证流程,实现了一种基于 Web 的 Word 文档在线处理方案。

## 2 WORD XML 和 WORD 组件对象

Word XML 旨在镜像传统的 .doc 文件中的信息。早期的 Word 以二进制形式进行数据的存储和传输。

但已经无法满足网络环境的挑战和完全不同的应用程序之间的数据传输。Word2003 以及以上版本扩展了以前版本的二进制文件的功能,采用了基于 XML(Extensible Markup Language)的 Word XML 方案。该方案使得 XML 的应用程序都可以访问和处理采用新文件格式的数据,强化了数据管理、数据恢复以及其他应用软件的互操作性。此外,安全性问题也大大减少,因为存储为 XML 的信息实质上是纯文本。因此,数据可以无障碍地通过防火墙。

Word 本身是个组件对象集合。用户可以根据这些组件对象接口调用 Word 的应用程序。图 1 给出了 Word 的主要对象结构图。其中 Application 对象表示 Word 应用程序本身,Document 表示正在处理活动文档,Selection 对象表示文档中当前选定的区域,Range 对象表示文档中的一个连续区域。Bookmarks 对象与 Range 和 Selection 对象类似,用于在文档中标记一个位置,或者用作文档中的文本容器。

## 3 设计与实现

根据 Word XML 的特点和均衡分布的要求,方案采用 J2EE 框架,分视图层、逻辑层和数据层三层结构,主要设计包括视图层的两个控件的设计和两个流程设计:文档的签名和存储以及文档的验证和查阅。

### 3.1 控件设计

#### 3.1.1 基于 Word2003 的办公控件

该控件集成了本地客户机的 Word2003 软件资源,向用户提供了一个基于 B/S 的在线办公手段。利

用 Word2003 的对象接口,根据 VC6.0++ 的 MFC 开发 ActiveX 的方法<sup>[4]</sup>,主要步骤如下:

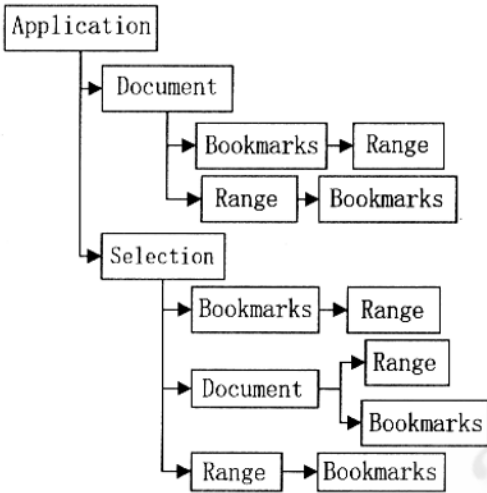


图 1 Word 的主要对象结构图

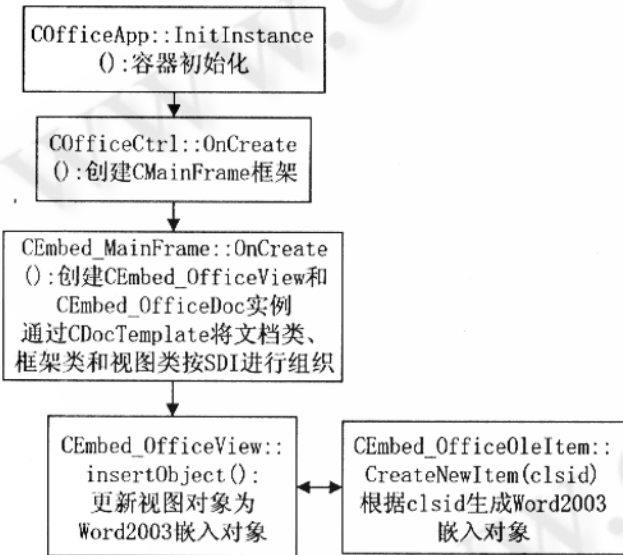


图 2 控件插入 Word2003 文档视图时的主类调用流程

(1) Word2003 的相关对象描述类的导入: 本文根据 Word2003 的安装目录下的 MSWORD. OLB 文件,利用 ClassWizard 工具,在创建的 ActiveX 工程中导入和添加了 Word2003 的部分组件对象描述类 CApplication、CDocument、Cselection、Cshapes 和 CinlineShapes。

(2) Word2003 文档视图的插入:ActiveX 工程起

初缺乏框架类、视图类和文档类,所以本文利用 ClassWizard 生成 CEmbed\_MainFrame、CEmbed\_OfficeOleItem、CEmbed\_OfficeDoc 和 CEmbed\_OfficeView 四个类。图 2 给出控件插入 Word2003 文档视图时的主类调用流程。

其中,CEmbed\_OfficeView::insertObject() 是核心过程,根据通过 CLSIDFromProgID(L"Word.document",&clsid)) 获得 clsid,然后通过 pCEmbed\_OfficeOleItem->CreateNewItem(clsid) 获得 Word 的 Document 嵌入对象。

(3) 接口方法的定义和方法的实现:控件的接口必须在 odl 文件中做具体描述。主要方法的伪代码描述如下:

```
void COfficeCtrl:: LoadXmlData ( VARIANT
buffer)
{ //根据压缩的 XML 数据打开 Word 文档
① 装载文档的 XML 压缩数据;
② 定义数据流缓冲区 CString strBuffer;
③ 获得字节流 strBuffer= buffer.pbVal;
④ 将 strBuffer 存储到本地文件 wordTemp.
zip;
⑤ 解压本地文件 wordTemp. zip 得到 wordT-
emp.xml;
⑥ 调用 CDocument.Open 方法装载和显示本地
文件 wordTemp.xml。
}
VARIANT COfficeCtrl:: GetXmlData()
{ //根据 Word 文档获得 XML 的压缩数据
① 获得文档的 XML 压缩数据;
② 定义 VARIANT buffer;
③ 利用 CDocument 的 SaveAs 方法将文档数据
以 XML 格式存储到本地 wordTemp.xml 文件;
④ 压缩 wordTemp.xml 得到 wordTemp.zip;
⑤ 根据 wordTemp.zip 根据得到压缩的数据
strBuffer;
⑥ 定义返回值类型为字符指针 buffer.vt=
VT_BYREF;
⑦ buffer.pbVal= strBuffer.AllocSysString
();
⑧ 返回 buffer。
}
```

```

BOOL CInsertFrmCtrl::Sign()
{ //插入签名位图
① BOOL bResult = TRUE;
② 定义 Word 的部分组件对象
CInlineShapes wdShapes;
CSelection wdSel;
CDocument wdDoc;
CApplication wdApp;
③ 获得 Word 的 Selection 对象接口指针
LPDISPATCH lpDisp;
lpDisp = m_pSelection->GetIDispatch();
④ 定义相关数据

```

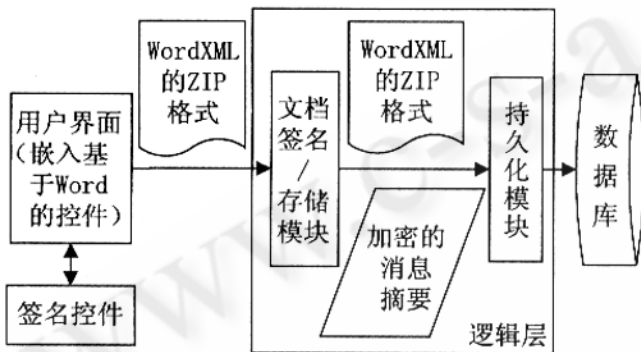


图 3 文档的签名和存储流程

```

COleVariant vTrue((short)TRUE),
vFalse((short)FALSE),
vOpt((long)DISP_E_PARAMNOTFOUND, VT_
ERROR);
⑤ 获得插入的 Word 文档类的指针
CEmbed_WordDoc * pDoc = GetDocument
();
⑥ 绑定嵌入的 WORD 对象
wdDoc.AttachDispatch(lpDisp);
⑦ 获得 Word 的 InlineShapes 对象
wdApp = wdDoc.GetApplication();
wdSel = wdApp.GetSelection();
wdShapes = wdSel.GetInlineShapes();
⑧ 验证本地 pict.wtmp 文件是否存在, 否则返
回 FALSE;
⑨ 插入签名位图 wdShapes.AddPicture
("pict.wtmp", vFalse, vTrue, vOpt);
⑩ 释放 wdDoc, wdApp, wdSel, wdShapes 接

```

口, 并返回 bResult。

```

}
```

### 3.1.2 基于 MSPaint 的手写签名控件

MSPaint 是 Window 系统的画图软件, 功能强大, 非常便于用户的手写签名。同时, 微软也提供了该软件的可嵌入对象编程模型接口 Paint. Picture 供用户调用。基于微软提供的 MSPaint 的对象接口, 利用类似开发办公控件的方法, 设计了一个集成了 MSPaint 界面的 ActiveX 控件。该控件提供了一个用户可在视图层上用鼠标进行手写签名的有效手段。

(1) 主类创建。利用 ClassWizard 生成 CSig-natureFrame、CSignatureOleItem、CSignatureDoc 和 CSignatureView 四个类。插入 MSPaint 文档视图的主类调用流程类似办公控件的设计。不同点在于, 在 CSignatureView:

```

OnInsertObject() 中, 通过 CLSIDFromProgID
(L"Paint. Picture", &clsid) 获得 clsid, 然后
在 CSignatureOleDoc->CreateNewItem(clsid)
获得 MSPaint 的嵌入文档对象。

```

(2) 控件方法实现。该控件的主要方法是获得签名位图数据。主要伪代码如下:

```

void CSignatureCtrl::GetSignatureData(
VARIANT buffer)

```

```

{ //签名位图文档存放在本地

```

```

① 获得嵌入对象;

```

```

② 根据嵌入对象得到文件接口指针 pPersist-
File;

```

```

③ 利用该接口指针 pPersistFile 保存签名位图
数据到本地 pict.wtmp。

```

```

}

```

### 3.2 双重签名和验证的流程设计

双重签名和验证涉及视图层的 Word 文档内的手写签名和验证以及逻辑层的 Word 文档的数字签名和验证。

#### 3.2.1 文档的签名和存储

文档的签名和存储分为三步:

(1) 在视图层, 用户利用签名控件进行可视化的手写签名, 然后以位图图片的格式自动插到文档中光标所指定的位置。

(2) 逻辑层的文档签名/存储模块对所提交的文档的 XML 的 ZIP 数据生成 MD5 消息摘要, 并且根据

用户角色匹配的私钥加密此消息摘要。

(3) 将此文档的 XML 的 ZIP 数据和已加密的消息摘要通过持久化模块导入到后台数据库。图 3 给出文档的签名和存储流程。

### 3.2.2 文档的查阅和验证

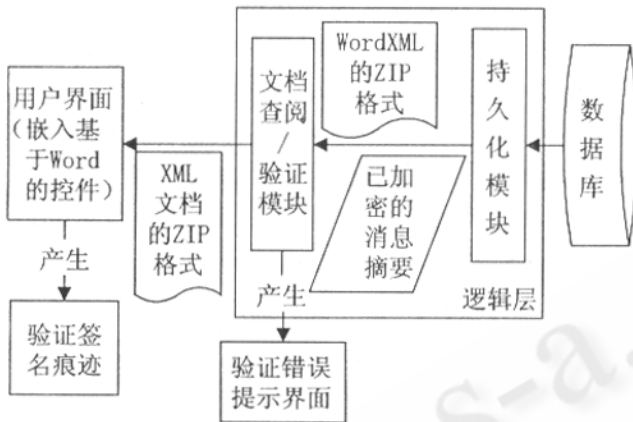


图 4 文档的查阅和验证流程

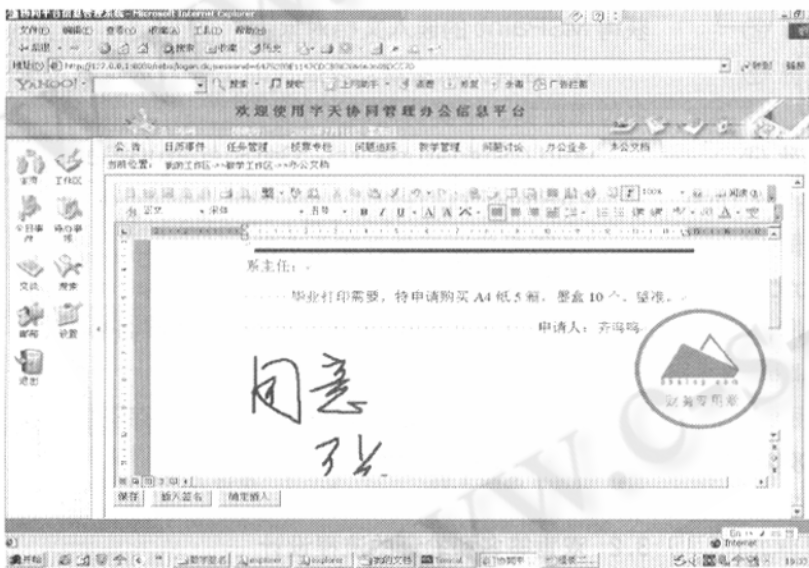


图 5 一个真实的运行界面

当用户网上查阅和验证文档时，此过程分为三步：

(1) 逻辑层的文档查阅/验证模块通过持久化模块得到文档的 XML 的 ZIP 数据和已加密的消息摘要。

(2) 根据文档作者角色对应的公钥，文档查阅/验证模块对已加密得消息摘要进行解密。此外重新生成文档的 ZIP 数据的 MD5 消息摘要。比较两者值来验

证文档的安全性。如果验证出错，文档查阅/验证模块则会派生出验证错误提示界面。

(3) 经过解压，视图层的办公控件打开文档数据。用户可以根据文档中的电子痕迹来验证文档的逻辑有效性。图 4 给出了文档的查阅和验证流程。

### 3.3 运用及分析

由于教学管理要求，某高校的计算机系办公流程要求多人在线轮流处理相同的 Word 文档，同时还要求保留文档的处理痕迹并保证其完整性和不可否认性。该方案解决了上述问题，目前已经应用于该高校内部办公项目上。图 5 给出了一个真实的运行界面。在使用中，该方案的优点体现在以下几个方面：

(1) Word 文档以压缩的 XML 数据集中存放在服务器的数据库中，便于管理和共享，达到文档的集中式管理和分布式使用的要求。

(2) 两个控件很好地扮演了药引子的作用。由于它们本身并不包含 Word2003 和 MSPaint，所以并不大，便于实时下载，无需预先安装。

(3) Word XML 是基于文本格式的，经过基于 ZIP 格式的压缩，其大小大大减少，便于数据进行提交和传输，缩短了服务器的相应时间。

(4) 基于文档消息摘要的物理安全的保证和基于文档内的手写签名的电子痕迹使得文档更具安全性和透明性。

(5) 利用角色定义一对密钥，克服了一户一对密钥，大大减少了密钥对的数量。同时，密钥对中的私钥和公钥分别存放在角色表的 PrivateKey 和 PublicKey 数据项中，简化了密钥的管理、分配和维护。

## 4 结束语

根据网络化办公的应用需求，该方案在传统的数字签名基础上，融合了用户的手写签名，实现了在线 Word 文档的双重签名和验证。实际应用表明，该方案具有一定的应用价值。

根据控件的一些应用缺陷，下一步工作将采用 Java Applet 取代控件来调用客户机的软件资源，克服服务器平台的局限性和控件的不安全性。

(下转第 103 页)

### 参考文献

- 1 刘敬学、费奇, 一种实现 VFP 与 Word 动态数据交换的简便方法[J], 计算机工程, 2005, 31(11): 198~199.
- 2 王玲、汪琦等, 在 Word 中建立数字电路逻辑符号库[J], 计算机系统应用, 2006, (02): 65~68.
- 3 方路平、章文彬, WEB 环境下得 WORD 文档生成技术讨论[J], 计算机系统应用, 2005, (05): 87~90.
- 4 Don Benage 等, 应用 Visual Studio 6.0 构建企业解决方案[M], 北京: 人民邮电出版社, 1999: 188~229.