

基于 OLE 和 VBA 的数据库与 WORD 数据交互研究

林 宫

(福州大学科技处, 福建 福州 350002)

摘要: 分析了当前科研项目管理信息系统中存在的数据交互问题, 并对 OLE、VBA 与 WORD 的自动化技术主要特征进行研究, 提出一种基于 OLE 和 VBA 的数据库与 WORD 文档数据交互的解决方案. 该方案对 WORD 文档模板的格式化和安全保护策略进行了设计, 并在某科研管理信息系统实施中验证了它的实用性和有效性.

关键词: OLE; VBA; 数据库; WORD 文档; 数据交互

中图分类号: TP311.13

文献标识码: A

The data interaction study between database and WORD based on OLE and VBA

LN Gong

(Department of Science and Technology, Fuzhou University, Fuzhou, Fujian 350002, China)

Abstract This article analysis the data interaction problem in the current information system of science research project management and study the automatic technology characteristic between database and WORD program based on OLE and VBA in order to giving a data interaction solution between them. It designs a WORD document templates format and security protection policy and verify the practicality and efficiency of this project in the environment of a actual science management information system.

Keywords OLE; VBA; database; WORD document; data interaction

科研项目管理部门大多根据管理的需要, 以 Microsoft 公司的 WORD 软件制作成标准的上报材料文档模板提供给用户, 而用户提交的 WORD 文档往往包含了上报材料所需的全部数据. 若能使现有的科研项目管理信息系统, 智能地从 WORD 电子文档中采集相关数据, 同时, 将管理信息系统中的数据直接导出生成 WORD 文档, 这样既方便用户上报科研项目的相关材料, 避免重复输入数据, 提高数据的准确性, 又能满足管理部门的管理流程和需求. 针对目前很多科研项目管理信息系统与 WORD 文档间数据交互都不能提供良好和便捷支持的问题, 开展这方面的研究, 对科研项目管理信息系统的建设具有重要的实际意义^[1, 2]. 本文结合某科研管理信息化工作的现状和需求, 提出一种基于 OLE 和 VBA 的数据库与 WORD 文档数据交互的解决方案. 在该方案中设计了 WORD 2000 文档模板的格式化及安全保护策略, 并结合某管理信息系统实现了 MS SQL Server 2000 数据库与 WORD 2000 文档之间的数据交互.

1 OLE、VBA 与 WORD 的自动化技术分析

OLE (Object Linking and Embedding) 是一种 Windows 系统中的进程交互通信机制, 实际上是一种编程接口. OLE 自动化能把 Office 等程序的对象链接或嵌入到应用程序系统中, 实行对 OLE 组件的编程式控制, 使得用户不仅能操纵驻留在自己应用程序里的对象, 而且能操纵驻留在系统上其他程序中的对象, 实现不同应用程序间的数据交互和资源信息共享. WORD 应用程序提供了很强的 OLE 服务功能, 用 VC、VB、C++ Builder 等开发的系统软件均能方便地以客户的方式向 WORD 应用程序提交命令, 取得并设置 WORD 对象的属性, 操作这些对象, 完成对象之间的传输通讯, 实现不同系统平台间的

收稿日期: 2006-03-06

作者简介: 林宫 (1964-), 男, 工程师.

基金项目: 福建省自然科学基金资助项目 (A0410014); 福建省教育厅科研资助项目 (JB05070)

数据交互,真正做到完全控制 WORD 应用程序中对象,包含自定义的 VBA(Visual Basic for Applications)宏代码^[3]. OLE 的引入使软件的开发方法从面向过程转为面向对象,是操作系统及其运行的应用程序的集成部分.

VBA 是 Microsoft 公司专为 WORD 等进行二次开发的工具,是新一代的标准宏语言,它能被所有的 Microsoft 可编程应用程序所共享,具有自动操作和控制 WORD 等应用程序对象的方法和属性,且 WORD 还提供了访问本地或远程数据库的编程接口.它引入事件驱动的机制用于组织应用程序,并实现了 Microsoft 应用程序的自定义功能和扩展功能.

WORD 自动化服务器包含很多可编程的 OLE 自动化对象,如 Application, Document, BookMarks, FormFields, Section, Tables, Words 等^[4],其主要的 OLE 自动化对象的层次关系如图 1 所示.

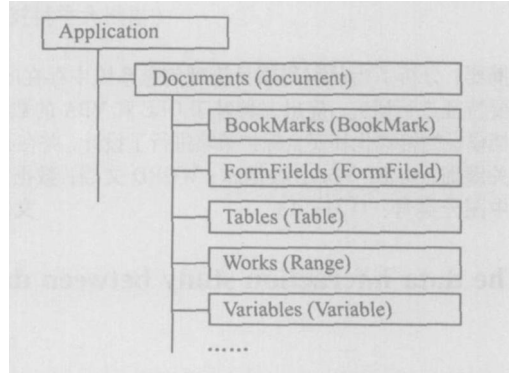


图 1 WORD 主要的 OLE 自动化对象的层次关系
Fig 1 Hierarchical relation of main OLE auto-object in WORD

经研究分析发现:OLE、VBA 的自动化技术可以创建并能完全控制 WORD 程序的操作,同时可利用 WORD 中的 BookMark, FormField, Table, Range 等对象及其属性来实现 WORD 文档的格式化,即按一定的要求规范地组织和布局 WORD 文档的数据和格式;然后,再根据需要通过 WORD 文档的窗体保护、Variable 对象等对文档实施安全保护和数据信息隐藏,防止用户破坏文档模板所制定的格式化模式以及随意更改文档的内容.

2 数据库与 WORD 文档间数据交互的系统模型

实现不同系统平台间数据交互的前提是数据的格式必须符合通讯协议规定的要求.因此,基于 OLE 和 VBA 的数据库与 WORD 文档的数据交互,首先,要对提供给用户的 WORD 文档模板进行格式化的设计,保证它们间数据交互的正确可靠.其次,必须对 WORD 文档进行安全保护,完成 WORD 文档格式化的实施.同时,考虑到 WORD 文档信息人工输入时不可避免会产生错误,所以 WORD 模板还应有充分的容错和报错机制,并可用 VBA 代码完成对错误数据的容错识别及提供相关识别信息.本文数据交互的设计思想:①在 MIS 系统中创建 OLE 对象操作 WORD,将 WORD 文档中的数据按一定格式采集到数据库中,或将数据库中的数据导入到 WORD 文档中;②用 WORD 宏、VBA 的自动化技术操纵 WORD 文档并连接所需的数据源,完成 WORD 文档与数据库间的数据交互通讯.这种 WORD 与数据库有效结合的数据交互策略,实现了 WORD 与数据库的数据共享,提高了系统的柔性.数据库与 WORD 文档间数据交互的系统模型如图 2 所示.



图 2 数据库与 WORD 文档间数据交互的系统模型
Fig 2 System model of data interaction between database and WORD document

3 WORD 文档模板的格式化和安全保护策略设计

WORD 以其操作使用简便、编辑功能强大,已成为人们日常管理工作中较为普遍采用的一种图文编辑处理和对数据进行加工的工具.但 WORD 文档中的信息形式多种多样,既有汉字、标点、数字、各种符号、表格及公式,还包括各种 ActiveX 控件、表单、窗体、域和书签等对象.控制 WORD 文档中的各种对象并对文档中的数据进行准确定位和读写,保证数据通讯的正确和快捷,是数据库与 WORD 文档数据交互的一个重要基础条件;并且 WORD 文档在人工填写过程中,经常会由于疏忽,致使内容或格式被修改.所以,必须让使用者只能在文档中指定的位置上填写内容,而在其他位置则不允许修改文档内容.因此,需要对 WORD 文档模板进行格式化和安全保护策略设计.

3.1 WORD 文档模板的格式化

WORD 文档中的域相当于文档中可能发生变化的数据或邮件合并文档中套用信函、标签中的占位符, 其中, 链接和引用域可插入指定的书签所代表的文字或图形, 但文档中必须有该书签的定义; 等式和公式域可用数学公式计算数值并插入在文档的表格或常规文本中。并且, 窗体域和 ActiveX 控件可以用 VBA 编写宏自定义其行为, 对其进行控制操作和数据交互^[3]。所以, 人们可根据具体的需求在 WORD 文档中的某一指定位置或表格中插入文字型窗体域、书签、链接和引用域、等式域、编辑域控件或命令按钮控件等, 将 WORD 的对象和数据规范地布局在文档中的指定位置。在人工编辑 WORD 文档时, 通过在文字型窗体域中直接输入数据, 或编写编辑域控件相应的图形用户界面及处理程序, 在用户窗体中输入数据, 而对某些标准化的数据的输入则可通过在数据字典选择数据实现, 并自动检查输入的信息是否符合规范要求, 然后将数据传入 WORD 文档中指定的位置。同时也可用文字型窗体域的计算功能得到书签处的值或通过引用域、公式域等得到所需的数据。当完成 WORD 文档的编辑后, 执行命令按钮定制的宏, 对文档所填写的数据内容进行有效性检查, 保证文档数据的正确有效。而 OLE 或 VBA 操作 WORD 文档时, 只须对文档中的对象及属性进行控制, 然后将数据按一定的格式要求写到 WORD 文档中。通过对 WORD 文档模板进行上述处理即可达到实现其格式化的目的。

3.2 WORD 文档模板的安全保护及 VBA 代码的封装

WORD 文档中的格式化部分, 只允许在 WORD 文档中指定的区域范围内进行数据交互, 一般不能随意修改文档中受保护的数据, 因此, 必须对此部分内容增加节标记并进行文档的窗体保护, 以实现文档实施格式化, 达到用户只能填写信息, 但不能改变文档的布局 and 对象等。而其自由格式部分, 则允许用户随意编辑文档中此部分内容。同时, 为防止非法用户修改文档工程中的对象和文档数据, 需对文档模板的工程进行加密保护和为文档编辑设置权限。另外, 在 WORD 文档数据交互通讯时, 需解除文档的窗体保护, 才能完成数据交互, 数据交互结束时再对文档实行对应的窗体保护。

虽然, VBA 技术可以实现 WORD 的自动化功能, 但 VBA 采用的是程序代码与文档相集成, 其安全性受到了影响, 因而在实际应用中受到一定的限制。而 DLL(Dynamic Link Library) 则为它提供了一个全新的支持, 通过 VB 编程技术, 将从 WORD 文档中导出的用户窗体、类模块等的 VBA 代码, 制作成 DLL 文件, 即可完成 WORD 文档中 VBA 代码的封装, 供 WORD 文档调用^[4]。VBA 代码封装的目的: ① VBA 代码与文档相分离, 提高了 WORD 文档的安全性; ② 保护 VBA 代码的安全, 避免他人通过解密等简单方法就能窥知过程代码, 维护了开发者的软件版权; ③ 将复杂的 VBA 代码制作成动态链接库, 加快了代码的运行速度和各程序间的协同作业; ④ 简化调用过程, 并使活动文档中的代码数量大大降低, 减小了文档文件, 提高了文档的运行速度。

4 数据库与 WORD 文档的数据交互的实现

由于本研究的 WORD 文档包含有格式化和自由格式 2 部分的数据信息, 所以主要是实现 WORD 文档中格式化部分的数据与数据库之间的数据交互, 而把整个文档另存成一个文件, 并建立其与数据库的连接, 供需要时由 MIS 系统调用 WORD 程序查阅该文档内容。数据库与 WORD 文档间的数据交互通讯有多种途径, 以下主要讨论基于 OLE、VBA 实现 SQL Server 数据库与 WORD 文档数据交互的方法。

4.1 基于 OLE 的数据交互

C++ Builder 的强大功能和简单易用的优点已成为一种人们常用的 MIS 系统前台开发工具, 以 C++ Builder 开发的某科研管理信息系统为例, 讨论用 C++ Builder 操纵 WORD 自动化服务器实现数据库与 WORD 文档间的数据交互过程。该系统采用浏览器/服务器(Browser/Server) 架构, 后台采用 SQL Server 作为数据库服务器。

在 C++ Builder 中实现 OLE 自动化, 需要定义一个可变类型(Variant) 变量来接收 CreateOLEObject 函数生成的自动化服务器实例。CreateOLEObject 函数在 ComObj 单元中定义, 使用时要在窗体的单元文件中包含 `utils.h`, `comobj.hpp`, `word_2k.h` 文件^[5]。由上面的讨论可知, 访问 WORD 文档中的域或书

签等对象即可获取所需的相应数据. 具体实现时, 定义一个接收 OLE 对象的可变类型变量, 然后, 创建生成 WORD 对象并打开 WORD 文档, 通过 OlePropertyGet, OlePropertySet, OleFunction, OleProcedure 等向 WORD 的 OLE 服务程序提交操纵 WORD 命令, 对激活的文档进行数据交互, 操作结束后清除 WORD 对象, 以免内存溢出. 以下是将 WORD 数据导入到数据库中的主要代码:

```

// 获得 WORD 自动化对象 Application
WordApp= CreateOleObject("Word Application");
// 通过 Application 获取 Documents 对象
WordDocs= WordApp OlePropertyGet("Documents");
// 打开 WORD 文档:
WordDoc= WordDocs OleFunction("Open", "文档文件名");
// 得到窗体域中的值
WordDoc OlePropertyGet("窗体域名称"). OlePropertyGet("Caption")
// 解除文档保护
WordDoc OleFunction("Unprotect", "密码");
// 得到书签位置的值得值
WordDoc OlePropertyGet("BookMarks"). OleFunction("Item", WideString("书签名")). OlePropertyGet("Range");
// 取得其它对象的值
.....
// 完成 WORD 文档与数据库间的数据交互
.....
// 保护文档
WordDoc OleFunction("Protect", wdAllowOnlyFormFields, true, "密码");
// 关闭文档, 并释放所创建的对象
WordDoc OleFunction("Close");
WordApp OleProcedure("Quit");
WordApp = Unassigned
WordDocs = Unassigned
WordDoc = Unassigned

```

此外, 在 WORD 数据导入到数据库时, 为防止破坏原文档的数据以及还需对文档进行其他后续操作等, 先通过程序复制所选择的 WORD 文档, 然后打开对应的文档, 判断其否是为满足数据交互要求的文档, 并根据实际情况采集管理信息系统所需的数据. 对于将数据库中的数据写入到 WORD 文档中, 只需通过 MS 系统连接数据库并筛选所需的数据写入到 WORD 文档中相应的位置即可.

4.2 基于 VBA 的数据交互

WORD 程序提供了 Microsoft Query, ADO, SQL API 等多种操纵 SQL Server, Oracle, Access 及其他数据库的接口^[3]. 对上述几种访问数据库接口的研究发现: Microsoft Query 是一个 WORD 查阅数据的组件工具, 它只能对数据进行查询, 不能对数据进行修改; ADO 和 SQL API 的方法均需要 VBA 编程支持. SQL API 是直接通过 ODBC API 访问数据库, 其方法复杂、调试困难、但运行速度快, 在进行大量数据查询或编辑时可考虑采用此方法. 而 ADO 是 Microsoft 公司提供的针对各种数据源的高级编程接口, 它支持对大多数数据库的操作访问. 因此, 基于 VBA 在 WORD 中应用 ADO 访问数据库是一种十分理想的方法.

本文采用邮件合并的方式通过 VBA 代码控制将数据库中的数据传输到 WORD 文档, 而将 WORD 文档数据写到数据库中则用 ADO 方法. 具体实现是: 在打开的 WORD 文档中, 用 VBA 直接建立起文档与其所需的本地或远程的数据库连接, 并应用 WORD 提供的邮件合并功能, 将从数据库中筛选的数据导出到 WORD 文档中, 这种方案只能对数据进行查询, 不能对数据进行修改; 而用 VBA 创建 ADO

对象, 执行 Select Insert Update Delete 等命令操纵数据库, 则可将 WORD 文档中的数据写入到数据库的表中或对数据库进行其它编辑操作^[4]. 主要代码如下:

1) 数据库导入到 WORD 文档的主要代码:

//连接并筛选邮件合并所需的数据

```
ActiveDocument Merge OpenDataSource Name = "",
    ConfirmConversions = False, ReadOnly = False, LinkToSource = False,
    AddToRecentFiles = False, WritePassWordTemplate = "", Revert = False,
    Format = wdOpenFormatAuto, Connection = "DSN=数据源名称;
    UID=用户帐号; PWD=密码; APP=Microsoft Query
    WSID=工作站名(客户端); Network=DBMSOCN",
    SQLStatement = "SELECT 字段列表 FROM 表名", SQLStatement1 = ""
//执行邮件合并操作并将结果生成一新的文档.
```

```
With ThisDocument Merge
    . Destination = wdSendToNewDocument
    . Execute
End With
```

2) WORD 文档数据导入到数据库的主要代码:

//定义 ADO 对象

```
Dim con As ADODB.Connection
```

```
Dim rec As ADODB.Recordset
```

//创建 ADO 对象

```
Set con = New ADODB.Connection
```

```
Set rec = New ADODB.Recordset
```

//连接数据库

```
con.Open "Provider=MSDASQL; DRIVER=SQL Server; SERVER=数据库服务器名;
    UID=用户帐号; PWD=密码; DATABASE=数据库名"
```

//执行数据库的 Select Insert update 等操作

.....

5 结语

MS 系统是提高工作效率和水平的重要手段, 其系统数据的智能采集和有效利用是推进管理工作信息化的重要环节. 在某科研项目管理信息系统开发中, 采用本文思想, 实现了基于 OLE 和 VBA 的数据库与 WORD 文档数据交互, 获得了较为理想的效果, 不仅可以大大减少管理工作的工作量, 推进管理工作的信息化进程; 同时也提高了 MS 系统的柔性和开发效率, 简化了系统的开发过程、节省了开发的成本, 为今后开发实施此类系统提供了一种思路和有益的尝试.

参考文献:

- [1] 董 丽, 陈和平. 基于 FTP 和 OLE 的科技项目网上申报管理系统的设计和实现 [J]. 武汉科技大学学报: 自然科学版, 2004 27(1): 80-83
- [2] 印 鉴, 曹王华, 杨 敏, 等. 科研项目管理系统的设计与实现 [J]. 计算机应用研究, 2005(3): 214-216
- [3] Microsoft Corporation. 全面掌握 Microsoft Office 2000 应用程序开发 [M]. 北京博彦科技发展有限公司, 译. 北京: 清华大学出版社, 2000
- [4] David Bector. Microsoft Office 2000 VBA 基础 [M]. 北京超品计算机有限责任公司, 译. 北京: 人民邮电出版社, 2000
- [5] Borland/Inprise C++ Builder 5 开发人员指南 [M]. 梁志刚, 汪 浩, 译. 北京: 机械工业出版社, 2000