

基于 C#.NET 的 Word 报告生成功能开发

叶明,张铮

YE Ming,ZHANG Zheng

北京航空航天大学 经济管理学院,北京 100083

School of Economic Management, Beihang University, Beijing 100083, China

E-mail:ymlake@sina.com

YE Ming,ZHANG Zheng.Development of Word report generation function based on C#.net.Computer Engineering and Applications,2008,44(9):104-106.

Abstract: Automatic report generation function is an important component of office automation system.In this paper,Using C#.net calling COM to control Word,the text and tables in the document are inserted into Word report separately.Meanwhile, brings up the cursor positioning methods to control the document format and location,and the content is inserted into the corresponding position.

Key words: C#.net;Word document;report generation

摘要:自动生成 Word 文档报告功能是办公自动化系统中的重要组成部分。采用调用 COM 组件实现 C#.net 对 Word 的控制的技术,将文档中的文字部分和表格部分分别插入到 Word 报告中,同时提出了使用文档光标跟随定位的方法来控制文档的起始位置和格式,并在相应的位置上插入对应的内容。

关键词:C#.net;Word 文档;报告生成

文章编号:1002-8331(2008)09-0104-03 **文献标识码:**A **中图分类号:**TP311.52

1 引言

Word 是世界上使用范围最广的文字处理系统。虽然许多应用程序开发工具都有其自身的报表生成系统,但用 Word 作为应用程序的报表输出工具却具有许多不可替代的优点:第一、可以生成图、文、表并茂的文档;第二、可以生成更为复杂的报表,包括多层次嵌套表格、斜线表格等;第三、用户可以定制和修改文档模板;第四、用户可以对生成的文档做进一步的加工;第五、生成的文档,用户可以进行转储和发布。

可见,用 Word 作为应用程序的报表输出工具可以进一步扩展应用程序的功能,提高系统数据资料的利用率和可共享性,便于用户对所生成的报表或文档进行二次加工和重复使用,这是那些应用程序开发工具自带的报表生成系统所不可比拟的。Word 决不仅仅是一个字处理系统,它是一个集成化环境,它集成了 Microsoft Visual Basic,可以通过编程来实现对 Word 功能的扩展^[1]。

Word 提供的强大的编程接口技术使得我们利用各种编程工具通过程序对其进行控制成为可能。在 C#.net 中控制 Word 主要有两种方法:一种是使用 CreateObject 启动 Word,然后以 Ole 方式对 Word 进行控制。但是这种方法在 C#.net 编程时没有代码提示,对于熟悉 Word 属性和方法的开发人员来说,使用该方法能够真正做到完全控制 Word 文件。但是正如上面所

提到的,这种方式要求开发人员十分熟悉 Word 接口中的属性和方法,无形中带来了很大的不便。另外一种是通过真正的 COM 技术,将 Office 软件目录中文件 MSWORD11.OLB 中的类库全部导入到 C#.net 中,利用 COM 技术编程,便可以在源码中使用各种操作函数来操作 Word。C#.net 将会自动将库文件转化为 DLL 组件,这样只要在源码中创建该组件对象即可达到操作 Word 的目的。

同时,对于大多数应用程序的报告或报表的一部分来说,其文档和表格的格式是相对固定的,如报告书的封面,个人简历表等,因此在报告具体实现的方法中,很多开发人员往往事先设计好相应的模板文件,在需要用程序填写内容的地方插上书签或特定的关键字,然后在报告生成的过程中把相应的内容插入或者替换到特定的地方^[1,2]。这种方法的好处是在程序外部设计好文档的风格,减少了代码量,提高了执行的效率。但是这种方法却为日后程序的升级和维护带来了很大的不便。

通过上面的描述可以发现,总体来说,通过 COM 技术来实现对 Word 的操作对于大多数编程者来说要相对容易得多,本文所采用的方法也是这种方法。同时,将要介绍的报告文档生成的方法是根据系统用户所选择的需要添加到报告中的内容,将文字和表格等内容动态的插入到报告当中,并运用文档光标跟随和定位的方法,在程序内部对文档的格式进行初步的设

基金项目:国家自然科学基金(the National Natural Science Foundation of China under Grant No.70371007);北京市自然科学基金(the Natural Science Foundation of Beijing City of China under Grant No.9052006)。

作者简介:叶明(1983-),男(汉族),博士生,主要研究方向:复杂数据分析、数据挖掘;张铮(1974-),男(汉族),博士,主要研究方向:复杂数据分析、数据挖掘。

收稿日期:2007-08-07 **修回日期:**2007-10-14

定。运用这种方法,笔者在国家某部委决策支持系统开发项目中,实现了 Word 文档格式的报告的自动化生成。

2 程序结构

在报告生成的过程中,主程序首先打开一份新的空白文档,扫描用户操作跟踪器,如果不为空,则根据跟踪器中的内容调用生成文档方法集中的对应的方法插入到文档报告当中,完后将文档保证至服务器。具体结构方式见图 1 所示。

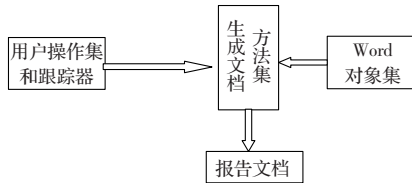


图 1 程序实现结构图

用户操作集合跟踪器:该方法集合主要跟踪并记录用户操作的类型和结果(主要是文字和表格),以构成插入报告的备选内容。

生成文档方法集:该部分是报告生成功能的核心部分,主要方法包括打开和关闭文档,插入文字,表格以及文档光标的定位等。在文档具体生成的过程中,主程序首先调用插入文字的方法插入文档的标题,再调用格式设定的方法来设定相应的标题格式。然后调用文档光标定位的方法来重新设定文档下一段内容的起始位置,依次进行下去。如果插入的内容是表格,则主程序则调用插入表格的方法进行表格的插入,同样插入一段表格之后再调用文档光标定位的方法来设定文档下一段内容的起始位置。该方法集合在程序实现的过程中编译成 DLL 文件,在具体报告的生成过程中以动态调用。

Word 对象集:即 Word 所对应的 COM 组件,为程序提供底层的方法和属性。其中常用的对象有 Word 应用程序对象(Microsoft.Office.Interop.Word.Application)、Word 文档对象(Microsoft.Office.Interop.Word._Document)等。上述两个对象中基本包括了对 Word 文档操作的基本方法和属性如文档的打开(Microsoft.Office.Interop.Word.Application.Open)和保存(Microsoft.Office.Interop.Word.Application.SaveAs)^[2]、在文档相应的位置插入某些特定的内容(Microsoft.Office.Interop.Word._Document.Range.InsertAfter)等。开发人员可以调用其中的接口来实现对 Word 文档的基本操作。

报告文档:该文档是首先在服务器端生成一个空白文档,然后依次插入报告的文字和图表等,完成之后保存在服务器端以便客户下载使用。

以上便是报告生成的主要逻辑框架,在下一节中将具体说明主要部分的具体实现方法。

3 程序实现

在生成文档的方法集合中,主要封装了对 Word 文档的主要操作的一些方法,下面将依次介绍这些方法中的文档的打开和关闭、插入文字和图表以及文档光标定位等方法。

3.1 打开和关闭 Word 文档

创建、获取以及存储一个 Word 文档对象是对 Word 操作的基础,程序在生成一份完整的报告之前必须首先在内存中创建一个新的文档对象。在报告完成后,须将报告保存在服务器

指定的位置中存储起来,以便用户下载使用。上述代码比较常见,由于篇幅限制这里就不一一列举。

3.2 插入文字及表格

在 Word 中插入文字的方法比较简单,在文档指定的位置中调用 Word 命名空间中 Document 对象中的 InsertAfter 方法即可,具体的代码语句如下:

```

Object start,end;
String Content;
oDoc.Range(ref start,ref end).InsertAfter(Content);
  
```

这里,start 和 end 变量指定文档光标的位置,其变量类型为 C#.net 中的基类型。变量 Content 即表示要插入文档的内容,变量类型为字符串类型。

若要在 Word 文档中插入一个内存中的二维表格,其形式如表 1 所示。

表 1 表格示例

时间段	GDP 增长/%	就业增长/%	就业弹性
“六五”	84.885	18.711	0.220
“七五”	94.781	9.099	0.096
“八五”	178.521	6.092	0.034
“九五”	77.703	-6.914	-0.089

其具体实现方法是这样的:首先根据内存中二维表格的行数和列数在文档中光标所在的位置插入一个对应的二维表格,再将内存中表格的内容依次插入到 Word 文档表格当中。具体是实现代码语句如下:

```

//获得内存表格中的行数和列数
DataTable dt;
int row,col;
row=dt.Rows.Count+1;
col=dt.Columns.Count;
//在文档中插入一个空白表格
oWord.ActiveDocument.Tables.Add(oDoc.Range(ref start,ref end),
row,col,ref oMissing,ref oMissing);
//将内存中表格的内容插入到文档表格当中
for(int j=1;j<=col;j++)
oWord.ActiveDocument.Tables[oWord.ActiveDocument.Tables.Count].
Cell(1,j).Range.Text=dt.Columns[j-1].Caption.ToString();
for(int i=2;i<=row;i++)
for(int j=1;j<=col;j++)
oWord.ActiveDocument.Tables[oWord.ActiveDocument.Tables.Count].
Cell(i,j).Range.Text=dt.Rows[i-2][j-1].ToString();
  
```

以上便是文档中插入文字和表格的实现方法。在插入文字或者表格之前,必须首先获得文档光标的位置,以便在指定的位置上插入相应的内容。下面将具体介绍文档在生成的过程中跟踪文档光标移动的具体方法。

3.3 文档光标位置跟踪

一份文档在新生成的时候,文档中的光标的位置为 0。主程序在文档中插入新的内容之后,光标也将随之移动,光标移动的新位置主程序必须要即时地跟踪和定位,以便获得文档要插入的下一项内容的起始位置。

如果文档中插入一个字符串,那么文档光标的位置移动比较简单,移动的量即为所插入字符串的长度。下面的代码则表示程序在文档中插入一个 Content 字符串后,文档光标的移动和重新定位:

```
oDoc.Range(ref start,ref end).InsertAfter(Content);
//计算插入文档的长度
Int length=Content.Length;
//获得新的位置
start=(object)((int)start+length);
end=(object)((int)end+length);
```

如果文档中插入一个二维的表格,那么文档光标的位置的移动就比较复杂。在这种情况下,光标移动的量需这样计算:对二维表中每个单元格里面的字符串长度求和,然后再减去二维表中的回车字符的数量,其中回车字符的数量为二维表中单元格的个数加上二维表的行数(在 Word 二维表格中,每一行的末尾都有一个回车字符)。具体实现代码如下:

```
//求二维表中每个单元格里面的字符串长度总和
Int length=0;
for (int i=1;i<=oWord.ActiveDocument.Tables[oWord.ActiveDocu-
ment.Tables.Count].Rows.Count;i++)
for (int j=1;j<=oWord.ActiveDocument.Tables[oWord.ActiveDocu-
ment.Tables.Count].Columns.Count;j++)
length=length+oWord.ActiveDocument.Tables[oWord.ActiveDocu-
ment.Tables.Count].Cell(i,j).Range.Text.Length;
//计算二维表中回车字符的个数并将之减去
length=length-oWord.ActiveDocument.Tables[oWord.ActiveDocu-
ment.Tables.Count].Rows.Count*oWord.ActiveDocument.Tables[oWord.Ac-
tiveDocument.Tables.Count].Columns.Count*2;
//计算新的光标的位置
start=(object)((int)start+length+oWord.ActiveDocument.Tables[oWord.
ActiveDocument.Tables.Count].Rows.Count*oWord.ActiveDocument.Tables
[oWord.ActiveDocument.Tables.Count].Columns.Count+oWord.ActiveDocu-
ment.Tables[oWord.ActiveDocument.Tables.Count].Rows.Count+1);
end=(object)((int)end+length+oWord.ActiveDocument.Tables[oWord.
ActiveDocument.Tables.Count].Rows.Count*oWord.ActiveDocument.Tables
[oWord.ActiveDocument.Tables.Count].Columns.Count+oWord.ActiveDocu-
ment.Tables[oWord.ActiveDocument.Tables.Count].Rows.Count+1);
```

上述代码中,变量 *start* 和 *end* 共同定位了文档光标的位置。读者可以发现,两个变量的值是相同的,那是因为如果光标

的两个定位值 *start* 和 *end* 相等的话,那么该光标值所对应的位置就是文档中的一个具体的点,这个点往往是文档中即将插入内容的起始位置。事实上,如果变量 *start* 和 *end* 处在不同的位置,那么 *start* 和 *end* 两个位置之间的内容即表示文档光标所选定的内容,通过程序可以控制该内容的格式如字体类型、居中、大小、加粗等。

以上方法便是程序在插入不同的内容后对文档光标位置的定位和跟踪,通过对光标的跟踪可以控制插入文档的位置和格式,初步实现了不通过 Word 模版和标签的方式来控制文档格式的目的。

本文所列举的程序在 C#.net 和 Word2003 中调试通过。

4 总结

使用 Word 作为应用程序的报告输出工具在利用性、兼容性和报告的可维护性上都有比较大的优势,同时由于 Word 提供的强大的编程接口技术,利用 C#.net 这种开发工具可以高效地生成各种专业的 Word 格式报告。本文所提供的方法不同于广泛使用的 Word 模版技术,而是通过跟踪和控制 Word 文档内部光标的位置来控制文档的内容的起始和相应的格式,并在相应的位置上插入指定的内容。这样使得在文档的内容实现上也更灵活,最终为报告的自动生成功能的实现提供了一种新的实现思路。

参考文献:

- [1] 鲁保玉,杨新芳.用 Delphi 生成 Word 报告及动态结构表格[J].计算机软件与应用,2007,4(3):180-183.
- [2] 孔令彦,董蓬勃,姜青香,等.使用 Visual Basic 操纵 Microsoft Word 对象生成报告文档[J].计算机工程与应用,2003,39(36):115-117.
- [3] 曹刚,韩作生.一种基于 Word 文档的手写批注实现方法[J].计算机工程与应用,2005,37(11):100-101.
- [4] 闫新珠,王秀芹.在 VC 中利用 Word 生成测量报告[J].地矿测绘,2006,22(1):32-33.
- [5] Kimmel P.C# 编程深度搜索[M].金旭日,伊尚锋,徐焱,译.北京:清华大学出版社,2003.
- [6] 李洪兴.模糊控制的插值机理[J].中国科学:E 辑,1998,28(3):259.
- [7] 王国俊.模糊推理的全蕴涵三 I 算法[J].中国科学:E 辑,1999,29(1):43-53.
- [8] 吴望名.参数 Kleene 系统中的广义重言式[J].模糊系统与数学,2000,14(1):1-7.
- [9] 王国俊.模糊推理的一个新方法[J].模糊系统与数学,1999,13(3).
- [10] 张兴芳,孟广武.蕴涵算子族及其应用[J].计算机学报,2007,30(3):448-453.
- [11] 彭家寅.基于某些常见蕴涵算子的模糊推理全蕴涵三 I 约束算法[J].自然科学进展,2005,15(5):539.
- [12] 王国俊.一种新型的三 I 算法及其逻辑基础[J].自然科学进展,2003,13(6):575.
- [13] Dubois D.Fuzzy sets in approximate reasoning,Part 2:logic approaches[J].Fuzzy Sets and Systems,1991,40(1):143.
- [14] 王国俊.非经典数理逻辑与近似推理[M].北京:科学出版社,2000.

(上接 63 页)

6 结论

本文给出了参数 IIP 系统所对应蕴含算子族 R_{p-n} 及其伴随三角模 T_{p-n} , 并且基于蕴含算子族 R_{p-n} 讨论了 FMP 及 FMT 模型的三 I 算法及 α -三 I 算法。借助于蕴含算子族 R_{p-n} , 计算得到的 A^* , B^* 中含有参数 p , 是参数 p 的函数, 该函数的性质有待进一步研究。

参考文献:

- [1] Zadeh L A.Outline of new approach to the analysis of complex systems and decision processes[J].IEEE Trans on Systems,Man and Cybernetics,3:28-33.
- [2] 王国俊,兰蓉.系统 H_α 中的广义重言式理论[J].陕西师范大学学报,2003,31(2):1-11.