

文章编号: 1005—0523(2005)01—0080—03

用 Delphi 建立数据库与 Excel 之间的数据通信

涂晓斌¹, 汤鹏志¹, 祖建樱²

(1. 华东交通大学 基础科学学院, 江西 南昌 330013; 2. 南昌大学 软件学院, 江西 南昌 330029)

摘要:介绍了 Delphi 与 Excel 之间的通信机制. 基于 OLE 自动化技术, 利用 Delphi 创建一个应用程序. 该程序可遍历当前数据库表中的所有数据, 并从中选取有用的信息进行分类汇总, 制作报表.

关 键 词: Delphi; Excel; 数据库; 数据通信.

中图分类号: TP391. 6 **文献标识码:** A

0 引 言

数据报表作为企事业单位上报和下达的重要信息载体, 随着信息化建设的不断推进, 在实际的工作中得到了前所未有的应用. 因此, 数据报表已经成为管理信息系统中重要的一项功能, 并且, 由于数据的多样性和统计信息的增加, 数据报表的系统实现变得越来越复杂.

Delphi 是一个高效的可视化数据库管理信息系统开发工具, 但是 Delphi 开发环境中提供的报表控件在制作复杂报表时显得不够理想, 不管是以前版本提供的 Quick Report 控件组, 还是 Delphi 7 提供的 Rave 控件组, 都不能让用户对生成的报表进行改动, 且程序控制很难实现. Excel 作为现代办公常用的电子表格制作工具, 以它的易操作性和实用性, 得到了各行业办公人员的青睐. 本文根据实际应用实践, 介绍利用 Delphi 编程控制 Excel 生成报表的各种方法.

1 Delphi 与 Excel 之间的通信机制

Excel 采用面向对象的方法开发. 所谓面向对

象, 是指将数据和对数据的操作放在一起, 组成一个相互依存、不可分开的整体, 这个整体就是对象. 面向对象的程序设计方法将数据和程序封装到对象中, 应用程序中的每个对象都包含了程序代码和数据. 为了能够对 Excel 程序进行二次开发, 微软提供了对 Excel 程序以及 Excel 程序中的对象进行访问. Excel 中的工作表、工作簿、图形以及 Excel 应用程序本身都是对象, 因此都可以对这些对象进行访问和控制. 要控制 Excel 做报表, 必须了解 Excel 的对象、方法和属性. Excel 对象具有紧密的级联关系. 其中, Application 是 Excel 的一个主要对象, 其它对象由此而派生. 对于 Excel 而言, 对象间的级联关系为 Application—>WorkBook (工作簿)—>WorkSheet (工作表)—>Cell (单元格).

Delphi 中提供了能够对 Excel 中各个对象进行操作的组件, 通过 Servers 组件面板中的 TExcelApplication 组件、TExcelWorkbook 组件和 TExcelWorksheet 组件可以直接操纵 Excel 应用程序、工作簿、工作表. 在将数据写入 Excel 中之前首先要连接 Excel 应用程序, 实现代码如下:

```
try
    ExcelApplication1.Connect;
except
```

```

MessageDlg( 'Excel 可能没有安装', mtError,
[mbOk], 0);
Abort;
end;

```

其中 ExcelApplication¹ 是 TExcelapplication 组件的名字. 在连接 Excel 应用程序后, 需要创建一新的工作簿, 并将增加的工作簿和 TExcelWorkbook 组件连接, 实现代码如下:

```

empty := EmptyParam;
//添加一个新工作簿
ExcelApplication1. Workbooks. Add(empty, 0);
//组件和工作簿连接
ExcelWorkbook1. ConnectTo ( Excelapplication1.
Workbooks. Item[
ExcelApplication1. Workbooks. Count ] );
//激活工作簿
ExcelWorkbook1. Activate;

```

其中, empty 为 OleVariant 类型的变量, ExcelWorkbook¹ 是 TeexcelWorkbook 组件的名字. 在连接了工作簿后需要将 TeexcelWorksheet 组件和工作表连接, 实现代码如下:

```

//组件和工作表连接
ExcelWorksheet1. ConnectTo (( ExcelWorkbook1.
Worksheet[1] as __worksheet));

```

其中, EexcelWorksheet¹ 是 TeexcelWorksheet 组件的名字, 该语句实现和工作簿中的第一个工作表连接. 准备工作完成以后, 就需要将数据库中数据写入 Excel 单元格中. 单元格的定位可通过 TeexcelWorksheet 组件的 Range 属性完成. 具体实现代码如下:

```

//将数据库中的数据写入 Eexcel
ExcelWorksheet1. Range[R, R]. Value := Fields.
Fields[i]. Value;

```

其中, R 表示要写入数据的单元格名称, 格式为“A1”、“B1”等. 用下面三种方法也可将数据写入“A1”单元格.

```

ExcelWorksheet1. application. cells. item[1, 1];
ExcelWorksheet1. application. cells(1, 1);
ExcelWorksheet1. application. cells[1, 1]. Value;

```

2 数据交换程序开发技术

Excel 表格的控制, 主要包括数据的导入、修改; 单元格的合并、边框的控制; 表格的复制、粘贴等.

当报表格式一定的情况下, 表格的复制、粘贴显得尤为重要, 这样, 可以先制作一个文件模板, 然后按照实际需要输出多页报表即可.

假若 Delphi 应用程序中放置了两个 TComboBox 组件分别用来选取数据库别名和数据库表名, 在窗体创建时应在数据库别名列表框中显示系统中所有已定义的数据库别名, 设窗体名为 TSec __TableForm, 组合框名为 DatabaseComboBox, 实现的程序代码如下:

```

procedure TSec __TableForm. FormCreate (Sender:
TObject);
begin
Button2. Enabled := False;
Screen. Cursor := crHourglass;
with DatabaseComboBox do
begin
items. Clear;
Session. GetAliasNames(Items);
end;
Screen. Cursor := crDefault;
If DatabaseComboBox. Items. Count < 1 then
DatabaseComboBox. Items. Add( '数据库别名没定义, 请您先定义. ');
end;

```

假若 Delphi 应用程序中放置了一个 TListBox 控件用来存放从数据库表中选取的有关字段名, 将该数据库表中这些字段对应的数据写入 Excel 的程序代码如下:

```

procedure TSec __TableForm. WriteData;
var Col: Char;
i, Row, j: integer;
R: String;
begin
Col := 'A'; //第一列为 A
Row := 1; //第一行用于导入数据库表字段名
for i := 0 to ListBox1. Items. Count - 1 do
begin
R := String(Col) + IntToStr(Row);
ExcelWorksheet1. Range[R, R]. Value2 := List-
Box1. Items. Strings[i];
Col := Chr(Ord(Col) + 1);
end;
with table1 do
begin

```

```
//从第一条记录开始
first;
Col := 'A'; //第一列为 A
Row := 2; //从第二行开始
while not eof do
begin
//从数据库每一行中逐个读取各个字段数值
for j:=0 to ListBox1.Items.Count-1 do
begin
//指定范围
R := String(Col) + IntToStr(Row);
//将数据库中的数据写入 Excel 中
ExcelWorksheet1.Range[R,R].Value2 :=
Table1.Fieldbyname (ListBox1.Items[j]).AsString;
//得到下一列,依次为 A、B、C……
Col := Chr(Ord(Col)+1);
end;
Inc(Row); //下一行
Col := 'A'; //重新指向第 1 列
Next; //下一条记录
end;
end;
end;
```

3 报表制作应注意的问题

3.1 报表格式的选择

报表格式的选择对信息系统报表的实现方法起着决定性的影响。如果在报表的格式要求比较严格的情况下,应当采用模板的方式产生报表。由于模板在数据导入之前就已经按照标准制定好,所以

只要在程序中控制模板的复制与粘贴,然后编程实现数据输入指定位置即可。而对于报表格式多变的情况,由于数据的不同,需要合并单元格或者控制边框,可以直接在程序中自动控制报表的生成。

3.2 打印

对于 Excel 报表的打印,最好不要在程序中进行控制,因为报表往往需要签字或者进行审查,有许多报表都包含平面图或示意图,为了有效的控制打印质量,最好通过程序控制输出或显示 Excel 文件,以便修改;另一个重要的原因是 Excel 具有强大的排版功能,而这正是选择 Excel 导出报表的重要原因。

3.3 报表时间和表头

报表时间要有用户可以控制的输出。表头的制作要在事先做好格式,控制输出时,只改动那些诸如上报单位、下达单位、负责人等数据,这样既保证了系统的效率,又不失其实用性。

4 结束语

用 Delphi 控制 Excel 来完成数据库管理信息系统的数据报表功能,是 Delphi 制作复杂报表的最佳选择,因为 Delphi 不但能控制数据的导出与导入,而且可以完成当前 Excel 应用中的大部分功能。如果深入研究 Visual Basic for Application(VBA)就可以制作出符合实际需要的各种 Delphi 控制 Excel 的控件。

参考文献:

- [1] 肖庆航,陆定淑,等. Delphi 7 数据库开发教程[M]. 北京:清华大学出版社,2004.
- [2] 蒋方帅. Delphi5 程序员指南[M]. 北京:人民邮电出版社,2000.

Building up Data Communication between Database and Excel By Delphi

TU Xiao-bin¹, TANG Beng-zhi¹, ZU Jian-ying²

(1. School of Natural Science, East China Jiaotong University, Nanchang 330013; 2. College of Software, Nanchang University, Nanchang 330029, China)

Abstract: It introduces the communication mechanism between Delphi and Excel and utilizes Delphi to establish a application program. Based on OLE automatic technology, The program can ransack all datas in the database, choose useful message to classify and make report form.

Key words: Delphi; Excel; database; data communication