

基于 Delphi+SQL Server 交叉报表的实现*

Realization of Cross Report Based on Delphi+SQL Server

易嵩杰 宋庆帧 李陶深
Yi Songjie Song Qingzhen Li Taoshen

(广西大学计算机与信息工程学院 南宁 530004)

(Coll. of Comp. & Info. Engi., Guangxi University, Nanning, 530004)

摘要 阐述 Delphi 中制作报表的主要控件及其功能,并以企业物流信息管理系统为例,介绍基于前端开发平台 Delphi 和数据库端 SQL Server,在不需第三方控件的情况下,交叉报表的实现。

关键词 交叉报表 Delphi SQL 动态生成

中图法分类号 TP391.13

Abstract Exemplified by the enterprise logistic manage system, the main controls and functions of cross report processing in Delphi are described, the cross report based on front developing platform of Delphi and database of SQL Server under the situation of not needing the other controlling person are also introduced.

Key words cross report, Delphi, SQL, dynamic creating

所谓交叉报表是指提取数据并将其变换成列名称,从而使数据能够以交叉部分格式进行编排的表格形式。在大型管理信息系统开发过程中,系统开发人员需要考虑的一个重要问题是如何设计和输出报表,尤其大量的交叉报表。如财务中的多栏式报表就是交叉报表。Delphi 是一个出色的可视化数据库系统开发工具,但是用 Delphi 设计复杂报表比较烦琐,操纵起来比较困难,QuickReport 控件常常不能满足数据库报表的需要,报表的制作、输出难以实现。为了获得复杂的、高质量的报表,常常需要借助第三方报表工具。本文利用 Delphi 中制作报表的主要控件和 SQL Server 的存储过程,实现了交叉报表的输出,免去了第三方报表工具。

1 Delphi 中制作报表控件

Delphi 中的 QuickReport 报表组件可以迅速地设计出符合西方人习惯用的报表。QuickReport 报表组件由以下部分组成:

(1) TQuickRep: 用于设置报表外观的主要组件,它包括标题 (TitleBand)、列标头

(ColumnHeaderBand)、细节 (DetailBand)、总结 (SummaryBand)、页眉 (PageHeaderBand)、页脚 (PageFooterBand) 等带区，其 DataSet 属性指定主报表将从哪一个数据集中获得数据。

(2) TQRLabel 组件：用于在报表中显示静态文本，与 Label 的性质相似；

(3) TQRExpr 组件：可以通过 expression 属性为报表设计各种计算；

(4) TDBText 组件：用于通过设置 DataSet 属性和 DataField 属性来指定数据源，用来显示数据库表格中记录的字段，可以是字符串字段、数值字段、日期字段和备注字段的值；在处理报表中，常用到数据访问控件 TQuery。TQuery 是 SQL 语句控件，可使用 SQL 语言对数据库进行操作，或者执行数据库中的存储过程，程序将根据 Query 返回的结果来生成报表。

2 交叉报表实现

下面通过一个企业物流信息管理系统实例，分别从前端开发平台 Delphi 和数据库端 SQL Server 这两个部分，讨论在不需要第三方控件的情况下，如何实现交叉报表的输出。

2.1 报表要求与分析

在某手机公司开发的一个企业物流信息管理系统中，设置了一总部商品库存报表，该报表要求汇总所有销售区库存数量。由于该公司的销售区个数是可变的，所以该报表是一个典型的动态数据源下交叉统计报表，其相关表结构如表 1。

2.2 数据库端的实现

交叉报表对于数据分析是二维的，即表的行和列。在交叉报表中行和列的交叉点构成了数据格，这就是交叉报表要分析、统计的数据对象，显然数据格的数据是建立在

表 1 总部商品库存报表的结构

字段名	键	描述	类型	长度	非空
mobcode	1	手机串号	varchar	50	1
mobname	0	手机名称	varchar	50	1
comShortName	0	销售区	varchar	50	1

行和列的基础之上的。在 Delphi 中不能直接建立交叉报表，因此我们需要在数据库中做一些变化，具体的做法是：首先建立一个临时表，将交叉报表中显示的列从相关表中提取出来，作为该临时表的字段，然后再将报表中需要显示的行分组合计插入新建的临时表中，最后按行分组查出统计数据，这就基本上形成了所要显示的交叉报表的数据源。这一处理过程在 SQL Server 2000 中可用存储过程实现，具体的实现代码如下：

```

declare @sql varchar (255), @sq varchar (800), @cou varchar (800)
/* 判断临时表 mobzhouse 是否存在，存在则删除 */
if object_id ('mobzhouse') is not null
    drop table mobzhouse
/* 初始化变量 */
set @sql=''
set @sq='手机名称，'
set @cou=''
/* 用游标取出交叉报表中要显示的列，本例中就是手机销售区 */
declare temptabl cursor for select comShortName from mobhouse
open temptabl
declare @a varchar (50)

```

```

fetch temptabl into @a
while @@fetch_status=0
begin
    set @sql=@sql+@a+' int, '
    set @sq=@sq+'sum (isnull ('+@a+', 0)) as '+@a+', '
    set @cou=@cou+'sum (isnull ('+@a+', 0)) +'
    fetch next from temptabl into @a
end
close temptabl
set @sql=@sql+' 手机名称 varchar (50) '
exec ('create table mobzhouse ('+@sql+')') /* 生成临时表 mobzhouse */
/* 将报表中需要显示的行分组插入新建的临时表 mobzhouse 中,即各个销售区的各类手机
的数目合计 */
open temptabl
fetch temptabl into @a
while @@fetch_status=0
begin
    exec ('insert into mobzhouse (手机名称, '+@a+') select mobname,
    count (mobcode) from mobhouse where comShortName =''' +@a+''' group by
    comShortName, mobname')
    fetch next from temptabl into @a
end
close temptabl
deallocate temptabl
set @cou=@cou+'0 as 合计'
/* 查询所有销售区库存汇总数量 */
exec ('select '+@sq+@cou+' from mobzhouse group by '' 手机名称 ''')

```

2.3 Delphi 中交叉报表的动态生成

在 Delphi 中, 报表不会自动生成, 要在选定大小的 QuickRep (相当于纸张) 上画表格, 而且是一根一根的画。不论是竖线还是横线, 都通过 QRShape 控件实现, 一根线就是一个控件。为了更好的表现实际效果, 还要做好报表中每根线的定位工作。但是在交叉报表中, 列名是从表中提取数据而生成的, 因此随着表中的数据不同, 每张报表的格式都不一样。由于无法预知表中会出现什么样的数据, 也就无法确定报表中的格式, 所以不可能为所有可能的结果信息都设计一种统一的报表格式。为了解决这一问题, 可采用动态生成报表技术。具体实现思路如下:

- (1) 定义动态数组类型, 数组的每个元素为一个类;
- (2) 根据 Query 返回的字段个数设置数组长度, 分配内存, 在程序中动态的生成各个报表控件对象, 并设置其属性;
- (3) 生成报表, 依据客户操作预览报表或打印报表;

(4) 在报表生成预览或打印完成以后，将释放所建立的控件。

下面是主要程序代码。

```

var
    ..... // 分别定义 TQRdbtext、TQRshape、TQRExpr、TQRlabel 数组以及其他变量
begin
    with query1 do //执行存储过程
        begin
            close;      sql.clear;
            sql.Add ('exec Procedure—mobhouse');      open;
        end;
    fieldlen := query1.FieldCount - 1; //字段个数
    setlength (qrdtext, fieldlen); // 设置数组 qrdtext 长度
    setlength (qrshape, 3 * (fieldlen - 1)); // 设置数组 qrshape 长度
    setlength (qrexpr, fieldlen); // 设置数组 qrexpr 长度
    setlength (qrlabel, fieldlen); // 设置数组 qrlabel 长度
    interval := 529 div fieldlen; //设定字段间间隔
    j := 1;
    for i := 0 to 3 * (fieldlen - 1) - 1 do
begin
    qrshape [i] := tqrshape.create (quickrep1); //创建 TQRShape 对象
    with qrshape [i] do //设置 qrshape [i] 的属性 (动态画线)
        begin
            if (i > 0) and (i < (fieldlen - 2)) then //画列标题带区的线
                parent := detailband1;
            if (i > (fieldlen - 1)) and (i < 2 * (fieldlen - 1) - 1) then //画细节带区的线
                parent := SummaryBand1;
            if (i > 2 * (fieldlen - 1)) and (i < 3 * (fieldlen - 1) - 1) then //画总结带区的线
                parent := ColumnHeaderBand1;
            left := 189 + j * interval; top := 0; height := 20; width := 1;
            shape := qrsVertLine; enabled := true; visible := true;
        end;
        j := j + 1;
    end;
end;
for i := 0 to fieldlen - 1 do
begin
    qrlabel [i] := tqrlabel.create (quickrep1); //创建 TQRlabel 对象
    with qrlabel [i] do //设置 qrlabel [i] 的属性
        begin
            parent := ColumnHeaderBand1; left := 190 + i * interval; top := 7;

```

```

height: =17; width: =interval-3; alignment: =tcenter;
autosize: =false; font.Size: =8; enabled: =true; visible:
= true;
end;
qrdbtext [i]: =tqrdbtext.create (quickrep1); // 创建 TQRdbtext 对象
with qrdbtext [i] do//设置 qrdbtext [i] 的共有属性
begin
parent: =detailband1; left: =190+i*interval; top: =1;
height: =18 width: =interval-3; alignment: =tcenter;
autosize: =false; dataset: =query1; font.Size: =8;
end;
qrexpr [i]: =tQRExpr.create (quickrep1); // //创建 TQRExpr 对象
with QRExpr [i] do//设置 qrexpr [i] [i] 的共有属性
begin
parent: =SummaryBand1; left: =190+i*interval; top: =
4;
height: =15; width: =interval-3; alignment: =tcenter;
autosize: =false; font.Size: =7; enabled: =true; visible:
= true;
end;
end;
for i: =0 to fieldlen-1 do
begin
//设置 qlabel [i] 的标题, 对应交叉报表的列名
qlabel [i] .caption: = query1.fields [i+1] .fieldname;
// 设置 qrdbtext [i] 字段域, 显示交叉报表中行和列的交叉点的数据
qrdbtext [i] .datafield: = query1.fields [i+1] .fieldname;
// 设置 qrexpr [i] 中的 expression 属性, 统计每一列的和
qrexpr [i] .Expression: ='sum (qu-zsale.'+ query1.fields [i+1] .fieldname +')';
end;

```

3 结束语

本文叙述了一种基于 Delphi+SQL Server 的交叉报表制作方法, 该方法在不需安装第三方插件的情况下, 较好地实现交叉报表的输出, 减少项目开销, 充分发挥了已有软件的功能。

参考文献

- 1 李香敏. SQL Server 2000 Programmer's Guide 程序员指南. 北京: 北京希望电子出版社, 2000.
- 2 清宏计算机工作室. Delphi 数据库开发. 北京: 机械工业出版社, 2000.