

# 可视化动态报表系统的 UML 建模研究与实现

袁裕辉 (广东省经济管理干部学院 510262)

**摘要:** 文章通过对报表系统的研究,利用 UML 技术,建立了基于对象的可视化动态报表系统的模型,该报表系统为用户自行开发的一种通用报表系统,它具有学习功能,在应用程序执行时能根据可视化的用户需求动态生成报表,并且以二进制文件的形式作为模板加入到报表知识库中,供以后调用、修改。

**关键词:** 报表模型 面向对象 统一建模语言 DELPHI

## 1 引言

以往的通用报表需要事先定义好一系列变量用来设计报表的格式及预置参数,然后在进行二次开发时根据实际的需要对变量赋值。由于传统报表设计一般将报表结构定义为一个关系型数据表,因此虽然在一定程度上减轻了报表编程工作,报表复用性差、报表格式单一、开放性不强的问题没有从根本上得到解决,主要体现在:

- (1) 需要逐项定义报表参数,报表定义不是“所见即所得”,定义过程不直观;
- (2) 报表系统不具有自动学习功能,在同一次运行时不能生成不同风格的报表,通用性不强;
- (3) 当报表格式变化较大时,难以满足用户动态添加新报表的要求,开放性不够;
- (4) 不能实现软件复用。

本文基于 OO(面向对象)特性,从对象的角度来分析报表的属性和相关操作,将报表类定义为一系列具有相同操作的报表对象的集合,从而从一个全新的角度提出一种具有学习功能的报表生成系统,并介绍它的模型、原理及实现。该报表系统解决了传统通用报表系统存在的问题,它具有以下优点:

- ① 该系统基于 OO 将报表定义为一个类,任一报表文件是报表类的一个实例,报表系统具有复用性;
- ② 该报表系统具有学习功能,能在应用程序运行时,根据用户在屏幕上可视化自定义报表的类型、报表式样、数据源、计算公式等用户需求参数,动态生成报表文件,

通用性强;

- ③ 报表文件以二进制文件的形式作为模板加入到报表知识库中,供以后调用、修改,开放性好;
- ④ 操作界面简单并能对数据进行常规统计工作。

## 2 基于 OO 的报表知识库模型分析

从 OO 观点来看,所有的报表对象具有相同的操作集和信息结构(属性集),报表类是报表对象的集合,因而可以将报表模型定义为一个类。记报表对象的集合为 T,报表类数据结构集为 P,报表模型可定义为:

$$TReport; : = <Tname, Inh, P, Coi, Itf> \quad (1-1)$$

$$P = \{d1, d2, \dots, dn\}$$

式中, TRname 为报表类的名称(或称为类标识); Inh 为类的继承性描述;  $P = \{d1, d2, \dots, dn\}$ , 为各子类的数据结构集; Coi 为类的操作集的具体实现; Itf 为类的对外接口,报表类通过公布各子类的相关属性值来接受用户输入需求参数。P 包含了用户的全部需求 R, 即 R 为 P 的 Public 属性,  $R \in P$

$$R = \{d1, d2, d3, d4, d5, d6, d7, d8\}, R \in P \quad (1-2)$$

$$d1 = \{flKind, RptName, HeardCaption, BottomCaption, Page, Direct\} \quad (1-3)$$

$$d2 = \{obkind, obPointer, (x1, y1), (x2, y2)\} \quad (1-4)$$

$$d3 = \{TableNameI\} \quad I = 1, 2, \dots \quad (1-5)$$

$$d4 = \{(x1, y1), (x2, y2), Text, Font, TableName, FieldName\} \quad (1-6)$$

d5={(x1,y1),(x2,y2),Text,Font,FormualStr,RepID, StatType} (1-7)

d6={(x1,y1),(x2,y2),Text,Font,RepID,Direct} (1-8)

d7={(x1,y1),(x2,y2),LineWidth} (1-9)

d8={FixedSize,Size,Name,Color,Style} (1-10)

式中 d1-- 报表模式子类, 它的实例属性有: 报表类型 flKind (单页表、纵横表、主从/复合表)、报表名称 RptName、页眉 HeadCaption、页脚 BottomCaption、纸张大小 Page、纸张走向 Direct (纵向、横向);

d2-- 当选对象子类, 它的实例属性有: 当前所选的对象类型 obKind、对象所在区域((x1,y1),(x2,y2));

d3-- 数据源子类, 它的实例属性有: 数据表 TableName, 同一张报表可以从多个数据源中取数据;

d4-- 数据文本子类, 它的实例属性有: 数据表中的字段名 TableName.FieldName, 及显示标签的位置 ((x1,y1),(x2,y2))、内容 Text、字体 Font, 数据文本子类的的文本内容从数据源子类中取值;

d5-- 公式子类, 它的实例属性有: 数据源中的表列的计算公式 FormualStr、统计类别 StatType (如求和、平均、最大值、最小值等)、及显示标签的位置 ((x1,y1),(x2,y2))、内容 Text、字体 Font;

d6-- 静态文本子类, 它的实例属性有: 报表标题、报表头等文本标签的位置 ((x1,y1),(x2,y2))、内容 Text、字体 Font, 及方向 Direct (纵向或横向);

d7-- 线子类, 它的实例属性有: 线位置((x1,y1),(x2,y2))、线宽 LineWidth; 用户可以利用线子类动态设计不同的表格式样;

d8-- 字体子类, 它的实例属性有: 字体的固定大小 FixedSize, 字体大小 Size, 字体名称 Name, 字体颜色 Color, 字体风格 Style;

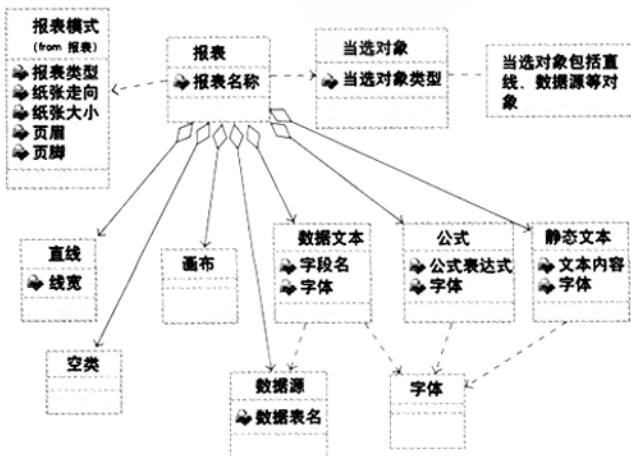


图 1 报表类建模

### 3 报表模型的 UML 建模

可以用 UML 语言来描述公式 (1-1) ~ (1-10) 所定义的报表模型, 如图 1 (建模工具为 Rose98, 开发环境为 DELPHI 5)。图 1 中数据源子类、数据文本子类、公式子类综合描述了报表的数据源、数据项、计算结果; 报表模式子类描述了报表的模式, 静态文本子类、线子类综合描述了报表的式样。图 1 的报表类的数据结构可用 Pascal 定义如下(仅以线子类 d7 为例):

Type

```
{线对象}
PLineObject=^LineObject; {直线类型指针}
LineObject = Record {画线}
x1,y1:Integer; {起点}
x2,y2:Integer; {终点}
LineWidth:Integer; {线宽}
end;
```

### 4 基于对象的动态报表系统的原理及实现

图 2 是在 DELPHI5 开发环境下开发的面向对象动态生成智能报表设计系统的用户界面, 该系统界面由系统菜单、工具栏、报表区域三部分组成。工具栏第一栏从左开始分别为“新建报表”、“打开报表”、“保存报表”、“预览报表”、“数据源”、“离开”、“调整”、“划线”、“擦线”、“静态文本”、“数据文本”、“公式”、“求和”、“上移”、“下移”、“左移”、“右移”、“画网格”等图标。

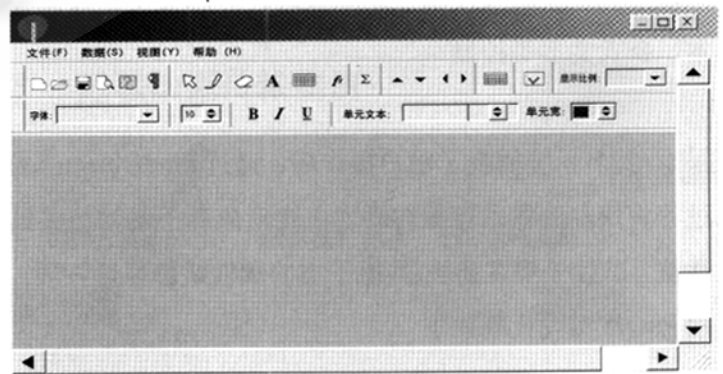


图 2 基于对象的动态报表系统的用户界面

下面来详细讨论系统的实现原理 (由于篇幅所限, 源代码省略, 开发环境为 Delphi 5):

(1) 用户向系统发出新建报表的消息时, 报表系统动态生成一个报表对象, 系统处于“空闲”状态, 等待用

户定义用户需求参数,用户通过操作工具栏上的图标,在报表区域可视化定义报表对象中各子类实例属性值。

(2) 系统学习用户的可视化的需求R,生成报表对象:

① 当用户要求新建报表时,系统动态生成以上各子类,并通过Procedure TReportPaintBox.NewReport过程初始化“画布”上的报表的各属性初值。

② 后用户根据其需求R在系统屏幕(画布)上可视化画出、修改预期的报表格式,系统通过procedure TReportPaintBox.MouseDown过程捕获用户的操作,程序代码如下:

```
procedure TReportPaintBox.MouseDown(Button:
TMouseButton; Shift: TShiftState; X, Y: Integer);// 捕获
Mouse 动作
var
x1,y1,x2,y2:integer;
Point: TPoint;
begin
inherited MouseDown(Button,Shift,X,Y);
if BtnOption=btnAllow then //系统处于用户定义
状态:
begin
if not(ssShift in Shift) then FindObject(x,y); //
查找鼠标附近的对象
end
else
... //画线控制、删除直线、静态文本、数据文
本、公式文本等控制
end;
```

③ 当系统捕获了用户操作后,通过 FindObject(x,y)过程将用户的需求参数转换成报表对象各子类的相关属性值,以线子类在为例(其他子类的属性赋值原理相同),用 Pascal 实现代码如下:

```
Procedure TReportPaintBox.FindObject(x,y:integer); /
/ 查找鼠标附近的对象
var i,xMin,xMax,yMin,yMax:integer;
OldKind:integer;
begin
{从直线对象中查找}
For i:=0 to LinesList.Count -1 do
begin
```

```
PobLine:=LinesList.Items [i]; //线对象集合
if (PobLine^.x1 =PobLine^.x2 ) or (PobLine^.y1
=PobLine^.y2) then //判断是否为直线
begin
{将用户定义的曲线位置值赋值给线对象的相
关属性}
xMin:=Min(PobLine^.x1,PobLine^.x2);
xMax:=Max(PobLine^.x1,PobLine^.x2);
yMin:=Min(PobLine^.y1,PobLine^.y2);
yMax:=Max(PobLine^.y1,PobLine^.y2);
end;
end;
MoveObject:=False;
if OldKind<>obtNone then Invalidate ;
end;
```

(3) 系统将可视化的报表转换成二进制的报表文件保存并加入知识库,供以后调用、修改;

(4) 当用户调出已建的报表文件时,系统动态生成报表类的实例,并将报表对象以可视化界面的形式转换成用户需求,并供用户修改。

## 5 结束语

本文提出的可视化动态生成报表生成系统,从OO出发,将报表系统定义为一个报表类,将用户需求定义为报表类的Public属性,在实际应用时,用户只需要可视化给出报表的模式、样式和数据等外观效果,系统即能动态生成满足要求的各类报表,同时由于将系统定义为类,针对特定要求可以以继承的方法派生新的子类,便于报表系统的扩充。这种报表生成系统能自成一体,适合大型通用软件的报表开发,有很强的通用性。■

### 参考文献

- 1 基于Windows通用财务报表系统的设计与实现,计算机工程与应用, No1, 1998。
- 2 基于Client/Server方式的生产管理报表系统,计算机应用, VOL20, No3, Mar. 2000/9/20。
- 3 Oracle数据库应用系统中的报表生成及其关键技术,微电脑应用, No8, 1999。
- 4 Object Oriented Analysis and Design Using UML, A Whitepaper By Mark Collins—Cope of Ratio Group。