用 PB 实现通用数据选择输入窗对象技术

谢梅源　（温州职业技术学院 325000）

摘要：在中大型管理信息系统中都要进行从大量的用户数据字典进行选择输入设计，本文将给出用 POWERBUILER 设计通用的数据选择输入窗对象技术的思路，并给出设计成功的关键程序及应用实例。

关键词：POWERBUILDER　数据窗口技术　通用数据选择输入对象

1　引言

中大型管理信息系统在输入设计时都要面临：如何实现从数据库中大量用户字典数据中选择输入大批量数据的问题。

中大型管理信息系统中，特别是一些行业信息系统中，为了系统的数据输入的规范性，在系统分析阶段，建立了很多的用户数据字典。同时在系统实施中，复杂的实际企业环境为这些字典集合，积累了极其多的数据。

因此在进行选择性的输入设计时往往会面临以下几个问题：

（1）对于任一用户数据字典，首先要考虑采用什么编码方式来为计算机输入选择的快速、简洁、用户友好的进行提供可能；

（2）对于数据输入选择过程的复杂的用户选择，设计要求应满足其返回信息的多样性；

（3）对于不同的用户数据字典、甚至不同的系统作为数据源的数据选择输入设计，是否能避免重复开发设计的问题；

（4）数据选择输入对象应具有：输入界面统一、极易用、用户友好等特点；

（5）数据选择输入对象在可重用性的基础上，还应能提供极简单的对象调用接口；

（6）数据选择输入对象的调用应是可据参数灵活定制的，以满足不同的要求。

本文作者用 POWERBUILER 设计通用数据选择输入窗对象，解决以上问题。

2　PB 中 DataWindow 技术的特点

2.1　DataWindow 介绍

（1）DataWindow 是一个 PB 中访问操作 DATABASE 的主要对象容器；

（2）DataWindow 功能很灵活，但最终生成的还是 SQL 脚本；

2.2　DataWindow 可以根据传入的字符串来动态创建

```
szSQLSelect = " select emp. id, emp. ename from emp"
szSyntax = SQLCA. syntaxFromSQL( szSQLSelect, szError)
dw_1. create( szSyntax, szError2 )
```

以上代码创建 DW 对象，即在 dw_1 控件中来显示 emp 表中 id，ename 列值。

3　设计思路

3.1　定义分离的参数，使此对象的调用做到灵活

为了使此对象具有通用性，满足不同开发人员使用该对象的需求，必须做到灵活性，因此该对象就不能与数据库中的某些用户数据字典表进行绑定，而通过 SQL 来动态创建。

（1）把与用户数据字典表相关一个查询语句通过 3 个字符串参数来传递。例如：查询 emp 表中 ename 以‘AB’开头所有人员，其相应的查询语句为：select emp. id, emp. ename from emp where emp. ename like ‘AB% ’。则 3 个参数值分别为“emp. id, emp. ename”、“emp”、“emp. ename like ‘AB% ’”

（2）在列表中显示列并不一定是 SELECT 子句中所有的列，例如显示 SELECT 子句列表中的第 3、4、7 列，则使用一个字符串参数，其值以 "3,4,7" 字符串来表示。同样，返回到父窗口的列表也用一个字符串参数来传递。

（3）该对象要把列表值返回到父窗口，因为返回

的列的类型不一定相同,返回的列数事先也并不知道,因此可定义一个变长的 any 类型的数组来存放返回的列值。

由于使用此对象都需要传递以上这些参数,所以把它们放在一个结构 s_select 中,如下所示:

colist	string	select 列表
tablist	string	FROM 子句列表
whereclause	string	WHERE 子句
dispcols	string	select 列表中在控件中显示的列号列表,用列号标识每一列
returncols	string	select 列表中返回值列号列表
returnvalue[ ]	any	返回列值

3.2 定义功能,对传入的参数集进行转换,转换成符合 SQL 语义的 SQL 语言

定义一些函数和事件,对于上述传入的参数进行相应的转换。

(1) 定义函数,根据传入的 SQL 参数,转化成符合 SQL 语义的查询语句,动态地创建数据源;

(2) 定义函数,对传入的 dispcols,returncols 参数值进行分离,得到列号;

(3) 定义函数,根据列号在数据源中得到该列的数据类型;

(4) 定义函数,对传入的 dispcols 字符串参数进行转化,从数据源中得到相应的列及列的数据类型;

(5) 定义函数,在数据源中隐藏不显示的列

(6) 定义函数,对传入的 returncols 字符串参数进行转化,从数据源中得到相应列的及列的数据类型;

(7) 在一个 keydown 事件,获得用户选中记录在数据源窗口的行号;

(8) 定义函数,把返回列值放入 any 数组中;

3.3 在对象函数中,由传入的 SQL 语言字符串,动态创建 DataWindow

3.4 设计灵活的对象显示容器,将上面 DW 的结果在容器上显示

在此对象上定义了一系列函数和时间,但数据窗口对象是不具有继承性的,如果希望两个不同的数据窗口上有相同的行为,唯一的方法就是重写代码。这显然不是好的面向对象的编程风格,解决的方法就是定义一个数据窗口用户对象。所以在 DataWindow (dw_1) 放在数据窗口用户对象(dwa_select)上。

最后定义显示的窗口对象,其上的数据窗口控件从 dwa_select 对象继承。

4 具体技术实现与关键代码

4.1 数据选择输入窗工作的主要流程步骤

(1) w_select 窗口从父窗口得到相应参数。

(2) 在 w_select 窗口 open 事件中调用 dwa_select 用户对象的初始化函数 initialize( )。

- 根据传入 SQL 语句动态的创建 DW
- 根据显示列号列表参数得到显示相应列
- 根据返回列号列表参数得到返回相应列
- 隐藏不显示列,并调整列的位置

(3) 用户选择列表中的某一条记录,得到该记录在数据窗口的行号。

(4) 把用户选中行相应的返回列存在 any 型数组中。

(5) 通知父窗口取回所选值,返回。

4.2 建立 dwa_select 用户对象的步骤

(1) 打开用户对象绘制器,选择" 新建" 和" 标准可视化用户对象" ;然后选择数据窗口作为类型。以 dwa_select 的名字存盘,dwa_select 是数据窗口祖先。

(2) dwa_select 用户对象上创建一个数据窗口控件 dw_1 及 9 个静态文本 st_1,st_2,st_3,st_4, st_5,st_6, st_7,st_8,st_9,功能是分别用来显示用户检索返回的记录及每一条记录在该页中的行号。Dwa_select 如图 1 所示的用户对象。

4.3 在 dwa_select 对象上定义的一些函数和事件的关键代码

(1) 根据 SQL 动态的创建数据源

//把 SQL 参数转化成符合 SQL 语义的字符串,不

检索任何数据

ls_sqlstring = ´select ´ + as_columnlist + ´ from ´ + as_tablelist + " where 1 = 0"

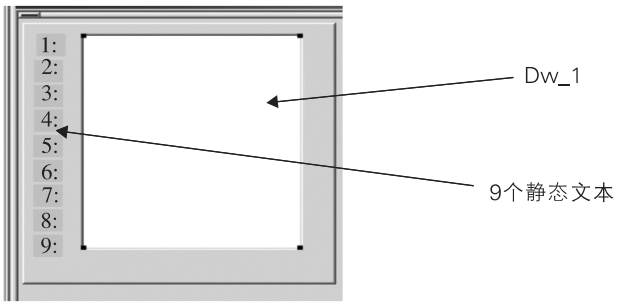

图 1　dwa_select 对象

//生成有效的 DataWindow SQL SELECT 语句字符串的语法

ls_syntax = itran_work. syntaxfromsql (ls_sqlstring, is_presentation, is_error)

if ls_syntax = ´ then return -1 //生成语法失败!

li_return = dw_1. create (ls_syntax, is_error) //动态创建 dw_1 数据源

if li_return < 0 then return -1 //创建 dw_1 失败

ii_col_count = integer (dw_1. describe (´datawindow. column. count´)) //总列数

(2) 根据列号,从 dw_1 数据源中得到相应列数据类型。

例如:ai_colnum = 3,则返回第 3 列的数据类型

ls_coltype = dw_1. describe (´#´ + string (ai_colnum) + ´. coltype´)

(3) 注册显示列,把列号及数据类型通过两个实例数组中来保存。

(4) 注册返回列,根据列号,得到返回列。

(5) 隐藏不显示列,如把 dw_1 中第 i 列隐藏

ls_modi + = ´#´ + string (i) + ´. visible = 0´

dw_1. modify (ls_modi)

(6) 在 keydown 事件中根据按下键的不同,可以选中列表中的某一记录,得到该记录在数据窗口的的行号,进行返回。

例如:当前数据窗口中可视记录的第一、二条记录在整个数据窗口是行号。

case key1!, KeyNumpad1!

il_selectedrow = long (dw_1. object. datawindow. firstrowonpage)

case key2!, KeyNumpad2!

ll_firstrow = long (dw_1. object. datawindow. firstrowonpage)

il_selectedrow = ll_firstrow + 1

……

以此类推,可以得到每一条可视记录在整个数据窗口的行号。

(7) 根据用户提供 WHERE 子句检索满足条件的记录,并用以填充选择列

ls_select = ´select ´ + is_columnlist + ´ from ´ + is_tablelist

if as_criteria < > " " then ls_select + = ´ where ´ + as_criteria

li_return = dw_1. setsqlselect (ls_select)

il_rowcount = dw_1. retrieve ( )

(8) 返回当前被选中的行,将该行为返回列值依次填入 any 类型的 aa_value[ ] 数组。

例如:将数据类型为数值型的选中行某列值放入该 aa_value[ ] 数组中。

ll_selectedrow:被选中行的行号;ii_returncol[i]:返回的某列;

aa_value[i] = dw_1. getitemnumber (il_selectedrow, ii_returncol[i])

(9) 在 w_select 窗口 open 事件中进行初始化,关键代码如下:

s_select = message. powerobjectparm //接收参数

if dw_select. initialize (sqlca, s_select. colist, s_select. tablist,

s_select. dispcols, s_select. returncols, ´ue_notify´) < 1 then

messagebox ("提示:", dw_select. is_error)

end if

if uo_select. showlist (s_select. whereclause) < 1 then this. triggerevent (´ue_notify´) //无有效选择

5　利用数据选择输入窗实现的实例

笔者在物资管理系统中,利用该对象实现数据选择的功能。

(1)在物资系统中,物资品名 wzd_pm 表结构如下:

Wzdm	物资代码(主键)
Wzlx	物资类型(1 – 设备,2 – 消耗品)
Wzpm	物资品名
Gg	规格
Pydm	拼音代码
Wzmb	物资别名
Bmpydm	别名拼音代码
Dj	单价

(2) 物资入库登记功能

例如,对电视机进行入库登记,在父窗口即物资入库窗口(w_wzrk)输入物资品名的首音码 D,回车,则首音码以 D 开头的所有物资品名都满足条件,在子窗口(w_select)中以列表的形式显示,如果满足条件的记录较多,则应分页显示由用户选择;如果没有满足条件,则应给用户提示信息。该功能类似智能拼音输入法,用户可以多输入首音码,减少返回的记录数,便于选择。用户在子窗口中选择符合条件的某一记录,再通知父窗口取回所需的信息。

(3) 在 w_wzrk 窗口关键代码如下

w_wzrk 窗口的 dw_1 控件的回车事件中描述程序如下:

```
s_select sele_info //定义一个 s_select 结构变量
ls_colname = lower( this. getcolumnname( ) ) //获得 dw_1 中的列名
if ls_colname = "wzd_pm_wzpm" then
ls_pydm = upper( trim( this. gettext( ) ) ) //获取输入的首音码
sele_info. colist = " wzd_pm. wzdm,wzd_pm. wzlx, wzd_pm. wzpm,wzd_pm. gg,wzd_pm. pmdm,wzd_pm. wzbm,wzd_pm. bmpydm "
sele_info. tablist = " wzd_pm"
sele_info. whereclause = " upper( wzd_pm. pmdm) like '" + LS_PYDM + " %' or upper( wzd_pm. bmpydm) like '" + LS_PYDM + " %' "
sele_info. dispcols = "1,2,3,4,6"
sele_info. returncols = "1,2,3,4"
openwithparm( w_select, sele_info)
sele_info = message. powerobjectparm
if sele_info. suceed = 1 then
ll_wzdm = long( sele_info. returnvalue[1]) //在父窗口取得返回行的第 1 列的值
```

6 用户数据字典表设计要求

(1) 必须有一字段(常为主键,或至少为候选键),记录用户数据字典中数据项的唯一性 ID 信息。

(2) 另有一字段,记录用户数据项的缩写编码。

(3) 缩写编码,可为数字编码,名称的五笔缩写码,但从实践经验总结,以名称信息的拼音缩写码最为实用。

7 结束语

在中大型信息系统开发中,需要构建许多通用用户对象,本文中的用户对象用 PB 开发,但其思路与方法同样也可用于其它语言开发用户通用对象。

参考文献

1 Ryan K. Stephens, Ronald R. Plew 著,何玉洁等译,《数据库设计》北京:机械工业出版社, 2001。

2 晓通网络数据库研究所编著,《PowerBuilder 开发手册》,呼和浩特 内蒙古人民出版社, 1999。

3 John W. Satzinger Robert B. Jackson Stephen D. Burd 著,朱群雄、汪晓男等译,《系统分析与设计》,北京 机械工业出版社, 2002。

4 萨师煊、王珊编著,《数据库系统概论》,北京 高等教育出版社, 2000。