

doi: 10.7690/bgzdh.2014.01.026

# Delphi7 中 DBGrid 控件鼠标滚轮事件的实现方法

张弦弦<sup>1</sup>, 蒙庆芳<sup>1</sup>, 王平<sup>1</sup>, 康林<sup>2</sup>

(1. 西南自动化研究所信息中心, 四川 绵阳 621000; 2. 总装重庆军代局驻绵阳地区军代室, 四川 绵阳 621000)

**摘要:** 针对 Delphi7 软件中 TDBGrid 控件内数据量超出屏幕时无法用鼠标进行上下滚动的问题, 设计了 2 种支持鼠标滚轮的方法。通过自行设计包含鼠标滚动事件的 TDBGrid 控件或通过对 Delphi7 自身 TApplicationEvents 控件的控制, 实现数据的翻页或逐条滚动。分析结果证明: 2 种方式都通过截获 Windows 操作系统中的消息, 达到对 TDBGrid 控件支持鼠标轮的滚动事件的目的。

**关键词:** 控件; 事件; 截获消息; 鼠标轮; 滚动

**中图分类号:** TP311.1 **文献标志码:** A

## Realization of DBGrid Controller Mouse Wheel Event in Delphi7

Zhang Xianxian<sup>1</sup>, Meng Qingfang<sup>1</sup>, Wang Ping<sup>1</sup>, Kang Lin<sup>2</sup>

(1. Information Center, Southwest Automation Research Institute, Mianyang 621000, China;

2. PLA Presentation Office in Mianyang District, PLA Military Representation Bureau of General Equipment Headquarters in Chongqing, Mianyang 621000, China)

**Abstract:** When data quantity of TDBGrid controller in Delphi7 software out of screen, the operator can not use mouse to roll up and down. Therefore, design 2 methods to supporting mouse wheel. Through design TDBGrid controller including mouse wheel event or by controlling in TApplicationEvents controller of Delphi7, realize turning over data screen or scrolling data one by one. The analysis data proves that: 2 methods can intercept information of Windows operation system, then realize TDBGrid controller supporting mouse wheel rolling event.

**Keywords:** controller; event; interception message; mouse wheel; rolling

### 0 引言

鼠标轮的滚动功能是一种基础的鼠标功能, 当我们浏览网页或者编辑文档时, 鼠标轮上下移动, 网页或文档随之自动上下滚动。目前, 很多软件都支持鼠标轮的滚动事件, 但在使用 Delphi7 语言开发应用的过程中, 若界面上显示的数据放在一个 TDBGrid 控件中, 当数据量过多超出整个屏幕时, 无法调用鼠标的滚动事件, 只能使用鼠标点击 TDBGrid 控件右侧的滚动条实现屏幕中数据的上下滚动, 给用户造成极大的不便。

TDBGrid 控件以网格的形式显示和编辑数据库中的数据, 是 Delphi 进行数据操作中使用最频繁、功能最强大的控件。由于 Delphi7 的大多数控件中都只支持 MouseDown(鼠标下移), MouseUp(鼠标上移), MouseMove(鼠标移动)等事件, 而不直接支持 MouseWheel(鼠标滚动)事件。为此, 笔者设计了 2 种对 DBGrid 控件实现鼠标滚轮事件的方法。

### 1 原理

Windows 是一个消息驱动的操作系统, 在 TDBGrid 控件中滚动鼠标轮时, Windows 会产生一个对应的消息。Delphi 把这一消息放在 Messages.pas 单元中, 命名为 WM\_MOUSEWHEEL。于是可以通

过截获 WM\_MOUSEWHEEL 消息, 使其支持鼠标轮的滚动事件。

其实现方法包括以下 2 种: 1) 自行设计一个 TDBGrid 控件, 该控件包括 OnMouseWheel、OnMouseWheelUp、OnMouseWheelDown 3 个事件, 并编码实现截获 WM\_MOUSEWHEEL 消息的功能; 2) 在 ApplicationEvent 控件的 OnMessage 事件里编码实现截获 WM\_MOUSEWHEEL 消息的功能。

### 2 设计与编程

1) 方法一。

派生一个 TDBGrid 控件, 取名为 NewDBGrid 控件, 如图 1。

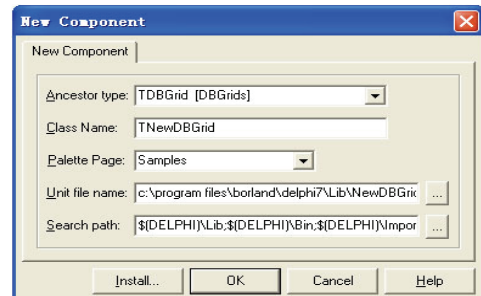


图 1 创建 TDBGrid 控件

在 NewDBGrid.pas 中增加 OnMouseWheel、OnMouseWheelUp、OnMouseWheelDown 事件(图 2)。

收稿日期: 2013-08-14; 修回日期: 2013-09-24

作者简介: 张弦弦(1980—), 女, 四川人, 工程师, 从事计算机软件应用开发研究。



图 2 增加控件事件

在 NewDBGrid.pas 中增加如下代码<sup>[1]</sup>:

```
//定义私有变量 fOnMouseWheel,
fOnMouseWheelUp, fOnMouseWheelDown 以保存过程指针
private
    fOnMouseWheel: TMouseEvent;
    fOnMouseWheelUp:
TmouseWheelUpDownEvent;
    fOnMouseWheelDown:
TmouseWheelUpDownEvent;
//定义 WndProc 方法
public
    procedure WndProc(var Msg: Tmessage);
override;
    published
        //添加 3 个鼠标轮滚动事件
        property OnMouseWheel: TmouseWheelEvent
read fOnMouseWheel write fOnMouseWheel;
        property OnMouseWheelUp:
TmouseWheelUpDownEvent read
fOnMouseWheelUp write fOnMouseWheelUp;
        property OnMouseWheelDown:
TmouseWheelUpDownEvent read
fOnMouseWheelDown write fOnMouseWheelDown;
    end;
//载入 WndProc, 以截获 WM_MOUSEWHEEL 消息
procedure TNewDBGrid.WndProc(var Msg:
Tmessage);
var
    MousePoint: Tpoint;
    Handled: Boolean;
    shift: TshiftState;
begin
    //若截获的消息为 WM_MOUSEWHEEL
    if (Msg.Msg = WM_MOUSEWHEEL) then
    begin
        //从当前消息中获取鼠标的位置
        MousePoint.X := LOWORD(Msg.Lparam);
        MousePoint.Y := HIWORD(Msg.Lparam);
        Handled := False;
        // Wparam > 0 时, 鼠标向上滚动
        if (Msg.Wparam > 0) then
        begin
            //调用存储在 fOnMouseWheelUp 元素中的方法
```

```
        if Assigned(fOnMouseWheelUp) then
fOnMouseWheelUp(self,shift,MousePoint,Handled);
        end
        else //鼠标向下滚动
        begin
            //调用存储在 fOnMouseWheelDown 元素中的
方法
            if Assigned(fOnMouseWheelDown) then
fOnMouseWheelDown(self,shift,MousePoint,Handled);
            end;
            //最后调用存储在 fOnMouseWheel 元素中的方
法
            if Assigned(fOnMouseWheel) then
fOnMouseWheel(self,shift,Msg.Wparam,MousePoint,
Handled);
                if Handled then exit;
            end;
        end;
    end.
```

在工程中创建一个窗口, 命名为 FrameDWEEdit。放入控件 TNewDBGrid, 命名为 DBGridDW, 在它的 OnMouseWheelUp 事件中增加如下代码<sup>[2]</sup>:

```
procedure
TFrameDWEEdit.DBGridDWMouseWheelUp(Sender:
Tobject;
    Shift: TshiftState; MousePos: Tpoint; var Handled:
Boolean);
//其中:
// Shift: 记录鼠标轮滚动时是否有其他的虚拟键
按下。
// MousePos: 鼠标的坐标, 由 X,Y 坐标组成。
// Handled: 表示鼠标轮的滚动是否处理。
Begin
    //向 DBGridDW 发 pageup 按键的消息, 使其数
据记录向上翻一页
    PostMessage(DBGridDW.Handle,
WM_KEYDOWN, VK_PRIOR, 0);
    PostMessage(DBGridDW.Handle,
WM_KEYDOWN, VK_DOWN, 0);
end;
在 OnMouseWheelDown 事件中增加如下代码:
procedure
TFrameDWEEdit.DBGridDWMouseWheelDown(Send
er: Tobject;
    Shift: TshiftState; MousePos: TPoint; var Handled:
Boolean);
begin
    //向 DBGridDW 发 pagedown 按键的消息, 使其
数据记录向下翻一页
    //为了体现数据的连续性, 保留上一页的最后一条记录
    PostMessage(DBGridDW.Handle,
```

```

WM_KEYDOWN, VK_NEXT, 0);
PostMessage(DBGridDW.Handle,
WM_KEYDOWN, VK_UP, 0);
end;

```

2) 方法二。

FrameDWEEdit 窗口中放入控件 TDBGrid, 命名为 DBGridData, 以列表形式显示数据<sup>[3]</sup>。增加控件 TApplicationEvents, 命名为 ApplicationEventsData, 用于截获 WM\_MOUSEWHEEL 消息, 如图 3 所示。



图 3 增加控件 TApplicationEvents

在 ApplicationEventsData 控件的 OnMessage 事件中编写如下代码<sup>[4]</sup>(如图 4):

```

//控制 TDBGrid 的记录条滚动
Procedure
TFrameDWEEdit.ApplicationEventsDataMessage(var
Msg: tagMSG; var Handled: Boolean);
begin
//判断是否接收到 WM_MOUSEWHEEL 消息
if(DBGridData.Focused)And(Msg.message=WM
_MOUSEWHEEL)then
begin
if Msg.wParam>0 then //当 wParam 参数为正
数时, 表示鼠标轮向上滚动
//发送消息, 记录向上翻
SendMessage(DBGridData.Handle,

```

```

WM_KEYDOWN, VK_UP, 0)
Else //wParam 参数为负数时, 发送消息, 记
录向下翻
SendMessage(DBGridData.Handle,
WM_KEYDOWN, VK_DOWN, 0);
Handled := True;
end;
end;
end;

```

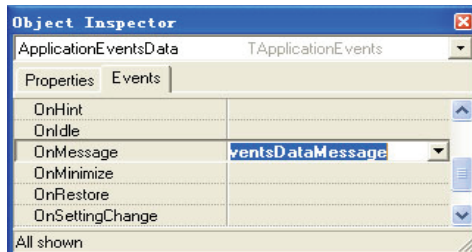


图 4 编写控件消息事件

### 3 结束语

上述 2 种鼠标滚动方法在 Delphi 开发中得到了广泛的应用, 其区别是: 方法一实现的是多行数据记录的上下滚动, 类似于翻页效果; 方法二实现的是一行数据记录的上下滚动, 类似于点击 TDBGrid 右边轮动条的上下按钮的效果。程序员可根据需求选择使用鼠标实现数据记录的翻页或上下滚动。

### 参考文献:

- [1] 李慧, 刘欣. Delphi 程序开发范例宝典[M]. 北京: 人民邮电出版社, 2012: 369-375.
- [2] 宋一兵, 赵景波, 李春艳. Delphi7 基础教程[M]. 北京: 机械工业出版社, 2005: 278-294.
- [3] 王璐, 官琴. 基于 PowerPC 嵌入式平台的 WindML 图形驱动设计[J]. 兵工自动化, 2013, 32(5): 95-96.
- [4] Delphi 帮助手册[S]. Borland 公司, 2002.

\*\*\*\*\*

(上接第 87 页)

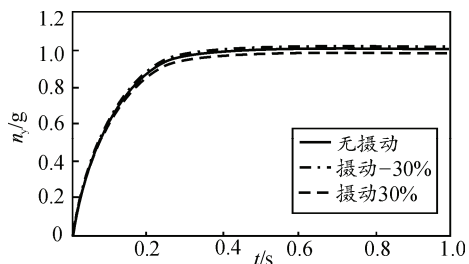


图 5 经典 PID 参数扰动时法向过载响应曲线

### 4 结论

笔者以滚转导弹为研究对象, 建立了动力学模型。采用专家 PID 控制方法设计了该导弹的控制系统, 并在 Matlab 中对该控制系统进行仿真。从仿真结果来看: 该控制器满足相关的性能, 具有较好的鲁棒性和自适应性, 能够有效应对气动与运动耦合。下一步, 笔者将改进专家 PID 的参数和规则设定。

### 参考文献:

- [1] 王旭刚, 王中原, 史金光. 滑翔增程弹制导与控制系统设计[J]. 南京理工大学学报, 2011, 35(3): 304-308.
- [2] 于剑桥, 梅跃松, 孟宏志. 鸭式气动布局单通道控制滚转导弹转速设计[J]. 弹道学报, 2010, 22(2): 32-34.
- [3] 袁天保, 刘新建, 秦子增. 自旋弹道导弹模型及运动特性分析[J]. 系统仿真学报, 2004, 16(6): 1125-1127.
- [4] 张鸣, 祝学军, 陈海东. 滚转导弹耦合性分析[J]. 导弹与航天运载技术, 2005(4): 47-51.
- [5] 王旭刚, 王中原. 弹体滚速和舵机时间常数对炮弹制导精度的影响[J]. 南京理工大学学报, 2011, 35(2): 183-186.
- [6] 钱杏芳. 导弹飞行力学[M]. 北京: 北京理工大学出版社, 2011: 249-251.
- [7] 刘金琨. 滑模变结构控制 Matlab 仿真[M]. 北京: 清华大学出版社, 2005: 94-102.
- [8] 白冬梅, 吕丽霞. 专家整定 PID 控制系统[J]. 兵工自动化, 2004, 23(4): 46-47.
- [9] 纪宇潇, 张继华, 刘旺开, 等. 基于专家系统的热动力试验台测控系统[J]. 兵工自动化, 2013, 32(6): 56-58.