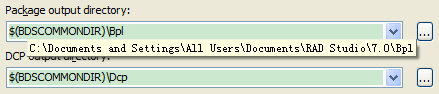
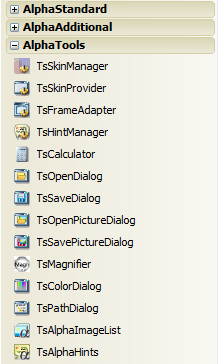
AlphaControls是一个标准的独创的皮肤控件集合，它具有很多新的属性可以添加到普通的控件中以增强程序界面效果，同时还能够添加一些行为到普通的控件中，并且使用了一些新技术。AlphaControls是一个易于使用和强大的工具，用于开发具有独创的皮肤和无皮肤的业务/媒体应用程序。  
  
在这里我使用8.51FS进行学习使用。

1. 安装前需先卸载以前的版本（如果是第一次安装，请跳过此步骤），移除IDE里的AlphaControls控件包，然后到bpl文件夹和dcp文件夹，删除所有（acnt\*.\*)文件，如图1所示：  
   

2.解压压缩包到文件夹，把这个文件夹加到"Library Path"；

**例如：**.增加library path 环境变量路径C:\Program Files\Embarcadero\RAD Studio\11.0\Components\AlphaControls8.51FS \acnt848\acnt\_regdelphiXE5 (操作菜单：Tools->Environment Options->Library->Library Path，点ADD，再OK)

实际路径请根据自己的路径修改

3.运行Delphi XXXX，打开"acntXXXX\_R.dpk"，编译此运行期库，再打开"acntXXXX.dpk"，安装；XXXX为具体版本号  
4.安装完毕后，就可以在组件面板上看到AlphaControls组件了，如下图所示：  


**XE5安装总结分享：（32位和64位的文件名称有明显的数字64区别，注意版本）**

以32位操作系统安装为例

XE5安装很简单，复制AlphaControls到任何位置，我习惯放新建Components文件夹所有的插件都放在里面，启动软件DELPHI XE5并添加插件库路径

双击acntDelphiXE5\_R.dpk,然后对右侧面的面板中的acntDelphiXE5\_R，鼠标右键，选择complie，

然后再双击acntDelphiXE5.dpk，然后对右侧面的面板中的acntDelphiXE5，鼠标右键，选择Install

完成后，就可以在VCL应用程序模式下，组件面板上找到AlphaControls相关的三个组件菜单

下面进行测试AlphaControls使用：  
1.新建一个应用程序窗体，在窗体上放置一个TsSkinManager；  
2.设置TsSkinManager其"SkinDirectory"属性为皮肤Skins文件夹路径；  
3.然后再选定其"SkinName"属性所要设置的皮肤名称，比如"UnderWater"，如下图所示：  
  
4.设置其"Active"属性为True，编译运行如下图所示：  
  
5.也可以通过设置"InternalSkins"属性，把皮肤文件编译到应用程序里面，记住"SkinName"属性还要选择"UnderWater (内建皮肤)"。

6.选中FORM1上的任何空间，点击左边的third part controls，可以关联第三方的空间  
7.另一个重要的控件是TsSkinProvider，放置一个到窗体上，它们的一些代码设置如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 | |  | | --- | |  | | uses sConst, sDefaults;    procedure TForm1.FormCreate(Sender: TObject);  begin    with sknmngr1 do              //TsSkinManager    begin      with AnimEffects do      begin        //-------------------窗体移动时透明度--------------------        BlendOnMoving.Active := False;        BlendOnMoving.BlendValue := 170;        BlendOnMoving.Time := 1000;        //-------------------按钮效果--------------------        Buttons.Events := [beMouseEnter, beMouseLeave, beMouseDown, beMouseUp];        //-------------------对话框隐藏时效果--------------------        DialogHide.Active := True;        DialogHide.Mode := atAero;        DialogHide.Time := 0;        //-------------------对话框显示时效果--------------------        DialogShow.Active := True;        DialogShow.Mode := atFading;        DialogShow.Time := 0;        //-------------------窗体隐藏时效果--------------------        FormHide.Active := True;        FormHide.Mode := atAero;        FormHide.Time := 0;        //-------------------窗体显示时效果--------------------        FormShow.Active := True;        FormShow.Mode := atFading;        FormShow.Time := 0;        //-------------------最小化时效果--------------------        Minimizing.Active := True;        Minimizing.Time := 200;        //-------------------页切换时效果--------------------        PageChange.Active := True;        PageChange.Time := 200;        //-------------------皮肤切换时效果--------------------        SkinChanging.Active := True;        SkinChanging.Mode := atFading;        SkinChanging.Time := 100;      end;      Effects.AllowGlowing := True;//允许按钮和边框图标有特殊发光效果，如Win7的标题按钮发光      ExtendedBorders := True;     //基于分层技术的窗体阴影      HueOffset := 0;              //允许改变当前皮肤的调色板      IsDefault := True;           //定义为当前管理器作为默认的应用程序管理器      with MenuSupport do      begin        ExtraLineFont.Name := '宋体';        ExtraLineWidth := 32;      //附加行宽度        IcoLineSkin := 'ICOLINE';  //图标行风格        UseExtraLine := False;     //是否使用附加行，      end;      Saturation := 0;             //改变当前皮肤的饱和度      SkinnedPopups := True;       //弹出菜单是否应用皮肤      SkinningRules := [srStdForms, srStdDialogs, srThirdParty]; //换肤规则      //-------------------让第三方控件也换肤，在属性器里设置比较好-------------------      ThirdLists[ord(tpEdit)].Add('TEdit');      ThirdLists[ord(tpCheckBox)].Add('TCheckBox');    end;    with sknprvdr1 do                    //TsSkinProvider    begin      AddedTitle.ShowMainCaption := True;//是否显示窗体标题      AddedTitle.Text := '附加的文本';   //允许定义一个额外的文本将在窗体的标题后显示      AllowBlendOnMoving := True;        //是否允许窗体移动时透明度      AllowExtBorders := True;           //是否允许窗体阴影      CaptionAlignment := taLeftJustify; //标题文本对齐方向      DrawNonClientArea := True;         //是否要绘制非客户区      GripMode := gmNone;                //当值是gmRightBottom时，窗体右下角有个三角形      MakeSkinMenu := False;             //是否在窗体的系统菜单添加“皮肤”菜单项      MenuLineSkin := '';                //菜单显示皮肤样式      ResizeMode := rmStandard;          //指定调整大小模式的形式，为rmBorder，则改变时只出现边框      ScreenSnap := False;               //指定是否窗体捕捉到屏幕边缘，主动粘边      ShowAppIcon := True;               //指定是否应用程序图标将显示在窗体的标题      SnapBuffer := 10;                  //当ScreenSnap为True，指定窗体边缘和屏幕边缘之间最大像素      //TitleButtons                     //添加标题按钮      //TitleIcon                        //添加标题图标      TitleSkin := '';                   //标题显示皮肤样式      UseGlobalColor := True;            //是否使用全局颜色    end;  end; |

7.放置一些控件测试效果，结果如下图所示：  


此插件基本上没有教程，只能看附带的DEMO，文件夹中有源文件自己编译运行查看，根目录下有EXE可以看DEMO效果