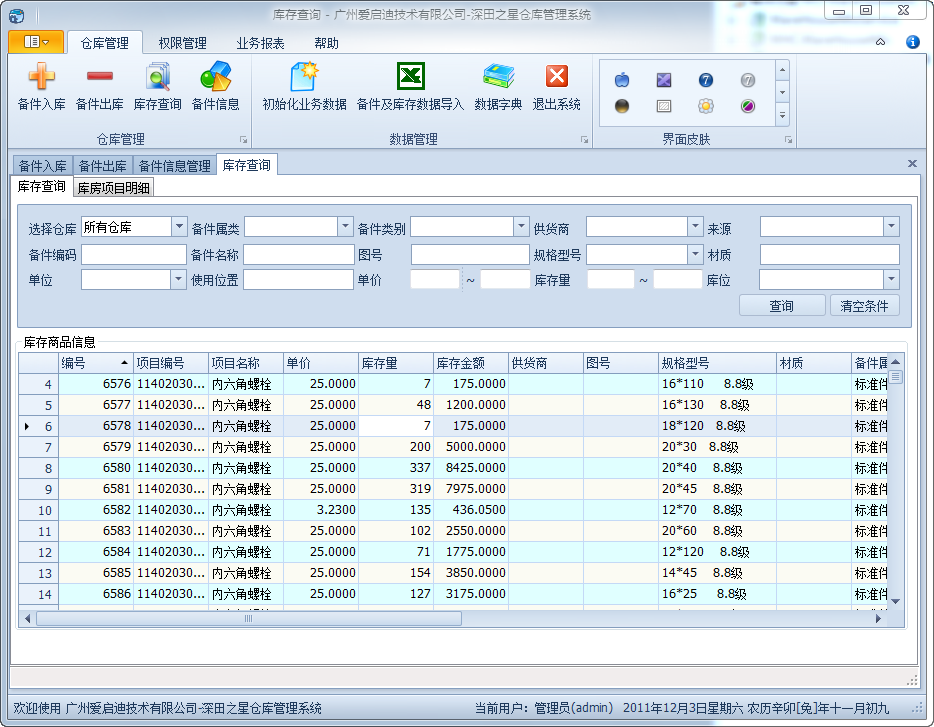
从事Winform开发很多年了，由于项目的需要，设计过各种各样的界面效果。一般来说，运用传统的界面控件元素，合理设计布局，能够设计出比较中规中矩的标准界面；利用一些换肤的控件或者部分界面组件，能够设计出相对好看一些的界面效果，如以前很盛行的ActiveSkin、IrisSkin和DotNetSkin等，这些能够对传统的界面元素进行换肤，确实比标准灰色的界面控件好看了很多。不过随着界面控件的组件发展，目前一般倾向于是用较为大型的控件组，他们除了提供设计得体的界面控件外，还提供了非常多种绚丽多彩的界面样式供选择，如DotNetBar、netadvantage、DevExpress等大型界面控件组。

无论界面设计如何变化，一般基本原则都是为客户提供直观、易用、体验效果较好的界面效果哦，从Office的发展历程我们也可以看到整体的界面效果趋向，从开始的标准控件到目前的Ribbon窗体，从单色调变化到绚丽多彩的界面样式，都给我们提供很好的界面设计参考，大型的界面组件也是模仿这一趋势。言归正传，我们来谈谈详细一点的东西，目前我趋向于采用一种基于Ribbon样式的界面，以及一种基于传统界面结合OutLook样式的界面设计。

 一、基于Ribbon样式的界面



 以上的界面样式，是一种比较大气、符合Office界面效果的界面布局，通过把不同的功能块集中在不同的面板上显示，确实简洁、美观很多，本文不重复介绍该界面效果的优劣，我们主要来集中看看下面的另外一种界面效果。

二、基于OutLook样式的界面设计

其实OutLook样式很早就有，也可以在很多公开的控件组中看到，如果仅仅是追求部分的OutLook界面效果，而不是整体性的方案，那么CodeProject上的这款开源Outlook组件，估计是其中的佼佼者（<http://www.codeproject.com/Articles/43181/A-Serious-Outlook-Style-Navigation-Pane-Control>）。不过由于需求的是整体性效果，而且还要考虑更多控件界面样式的一致性及美观性，因此我们还是基于DevExpress界面组来设计这样的OutLook界面效果，会显得更加美观大方一点。界面效果如下所示。



以上通过把一个系统很多相关的功能模块放到一颗树上进行分类展示，对于一个比较复杂的人力资源管理系统或者其他复杂功能的系统，也是一个比较合理的布局方式，另外OutLook工具条还是可以隐藏起来，节省右边多文档界面的数据展示空间，这样整体还是比较合理及美观的。其中右边的布局，还可以通过SplitContainer方式把它分成多个模块，然后客户想哪个数据显示面板大一点，拖动一下就可以了，这样不至于数据比较多的时候，导致显示布局不好的情况。下面我们来介绍下如何实现以上的界面布局效果。

**1、创建一个基于DevExpress.XtraEditors.XtraForm基类的窗体。如下代码所示**

[复制代码](javascript:void(0);)

public partial class MainForm : DevExpress.XtraEditors.XtraForm

{

public MainForm()

{

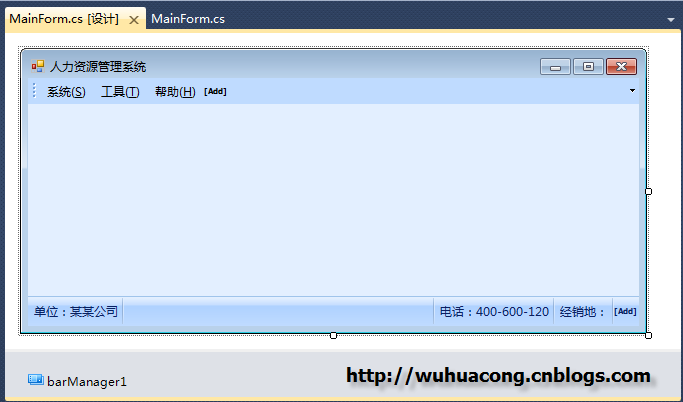
InitializeComponent();

}

}

[复制代码](javascript:void(0);)

**2、在界面设计中，在DevExpress工具箱Navigation & Layout里面拖动添加一个BarManager控件到窗体中，并删除默认的工具栏Tools，并添加一些菜单项和状态条数据。如下所示。**



**3、添加PanelControl和NavBarControl，添加一些测试功能按钮。**

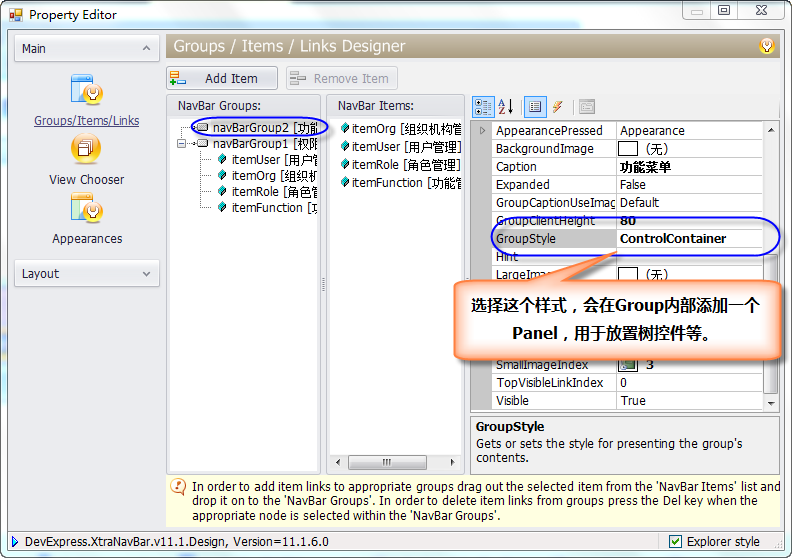
由于我们需要使用MDI多文档界面效果，因此先设置Mainform的IsMdiContainer属性为True。

然后添加一个PanelControl，设置其Dock为Top布局，为该控件ContentImage设置一个背景图片（事先用ＰＳ设计好，保存为png格式即可），添加几个小Lable，设置其的图片和文字。

最后拖入一个NavBarControl控件到界面中，设置其Dock为Left布局，通过控件的右键菜单上的“Run Designer"进入设计界面，先随便添加一些NavBarGroup和NavBarItem项目，粗略设置得到界面效果如下所示。



注意，在NavBarcontrol里面，默认是没有一个Panel可以添加一些特殊的控件，如树、按钮等，默认只有BarItem对象可以添加进入。为了在一个NavBarGroup里面添加这样的控件，需要修改NavBarGroup的属性才行，如下所示。



**4、添加DockManager和XtraTabbedMdiManager 控件实现多文档布局，其最终将以Tab方式进行展现。**

继续在上面的窗体中添加DockManager控件和XTraTabbedMdiManager控件，这两个控件可以实现在右边以Tab方式展现多文档布局，这样对用户操作来说，可以一次性打开多个窗体进行操作，方便很多，也必将美观，是一种常见的布局展现。为了在窗体激活的时候，在顶部显示关闭按钮，其他的不显示，那么需要设置XTraTabbedMdiManager控件的ClosePageButtonShowMode=InActiveTabPageHeader即可实现了。最终Tab效果如下所示。



以上就是我设计的一个系统界面的具体操作流程，其实很多时候，介绍总是很快，摸索总是很慢，这个就是知识积累的效率提升。当然，要设计好一个系统界面，除了考虑界面的布局美观性、还要考虑图标的协调性、还有就是整体的框架，要可以比较好的适应这些布局控件的操作，不能太过臃肿或者难以阅读。