

 万水计算机技术实用大全系列

# Borland Delphi 2005 C# Builder

# 精彩编程实例集锦

罗斌 等编著



包含所有实例的源代码



中国水利水电出版社  
[www.waterpub.com.cn](http://www.waterpub.com.cn)

万水计算机技术实用大全系列

**Borland Delphi 2005**  
**C# Builder 精彩编程实例集锦**

罗 斌 等编著

中国水利水电出版社

## 内 容 提 要

本书用 200 个独立实例程序介绍了 Borland Delphi 2005 C# Builder 最新的编程技术。书中所有实例程序摒弃语法说教, 侧重实际应用, 主要包括: 使用 GDI+ 技术开发图形文字特效程序技巧、使用 ADO.NET 开发数据库程序技巧、使用 ASP.NET 开发 Web 应用程序技巧以及 Web Service 程序设计技巧等。在本书中, 还介绍了托管状态下的文件操作、系统管理、注册表应用、网络通信、界面设计等难得一见的实用技术, 是 Borland Fans 进军 Microsoft .NET Framework 的重要参考书。本书所有实例程序在配套光盘中均有源代码, 让您所见即所得、所学即所用。

本书适合于使用 Borland Delphi 2005 C# Builder 进行 Microsoft .NET Framework 程序开发的初、中级用户参考, 也可作为高等院校及计算机培训学校在校学生的学习案例。

### 图书在版编目 (CIP) 数据

Borland Delphi 2005 C# Builder 精彩编程实例集锦 / 罗斌等编著. —北京: 中国水利水电出版社, 2006

(万水计算机技术实用大全系列)

ISBN 7-5084-3537-0

I. B... II. 罗... III. ①软件工具—程序设计②C 语言—程序设计  
IV. ①TP311.56②TP312

中国版本图书馆 CIP 数据核字 (2006) 第 001445 号

书 名	Borland Delphi 2005 C# Builder 精彩编程实例集锦
作 者	罗 斌 等编著
出版 发行	中国水利水电出版社 (北京市三里河路 6 号 100044) 网址: www.waterpub.com.cn E-mail: mchannel@263.net (万水) sales@waterpub.com.cn
经 售	电话: (010) 63202266 (总机)、68331835 (营销中心)、82562819 (万水) 全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	北京市天竺颖华印刷厂
规 格	787mm×1092mm 16 开本 35 印张 866 千字
版 次	2006 年 1 月第 1 版 2006 年 1 月第 1 次印刷
印 数	0001—5000 册
定 价	58.00 元 (含 1CD)

凡购买我社图书, 如有缺页、倒页、脱页的, 本社营销中心负责调换  
版权所有·侵权必究

# 前 言

为了适应 Internet 的快速发展,微软公司于 2000 年推出了 .NET 战略,以此作为新一代应用程序的开发平台。 .NET 战略反映了微软这位软件巨人对未来软件的思考,囊括了微软所有的最新技术成果。在 .NET 时代,程序员将会从以 Windows 为中心的应用程序逐渐革新为以 Web 应用程序为中心,就像当年从 DOS 转向 Windows 一样,它带来了编程模式的又一次飞跃。作为一家在 Windows 平台开发工具领域享有崇高声誉的厂商, Borland 公司在微软发布 .NET 平台之后,迅速取得了微软 .NET 平台的授权并且推出自己的面向 .NET 平台的开发工具 C# Builder。由于 C# Builder 基于 Microsoft .NET Framework,并且是在微软推出功能相当强大的 Visual Studio.NET 2003 之后的产品,因此它受到了许多程序员的质疑,成为 Borland 有史以来最具争议的开发工具。那么, C# Builder 是否真正是一个作秀的产品呢?我们来听听 Borland 大中华区首席技术官李维先生在介绍 C# Builder 时的一席话:

“Borland 已经在开发工具领域竞争了近 20 年,深深了解如何提供使用者需要的开发工具,因此 C# Builder 不但内含了其他 .NET 开发工具提供的功能,例如开发 WinForm、WebForm、Web Service、ASP.NET 和 ADO.NET 等, C# Builder 更提供了其他 .NET 开发工具无法企及的强大功能。”

C# Builder 最初是以一个独立的开发工具出现的,在 Delphi 2005 中, Borland 公司将 Delphi for.NET 和 C# Builder 集成在同一环境中,成为名符其实的为 .NET 企业级开发量身打造的解决方案。本书将用立竿见影的实例程序全面展示使用 Borland Delphi 2005 C# Builder 开发各种 Microsoft .NET Framework 应用程序的最新魅力。全书由简到繁、从易到难共分为程序界面编程实例、程序控制编程实例、文件操作编程实例、系统管理编程实例、网络开发编程实例、GDI+特效编程实例、ADO.NET 数据访问编程实例和 ASP.NET Web 应用编程实例共八章 200 个独立编程实例,每一个编程实例都给出了简要的编写说明、具体的实现步骤和完整的源代码(见光盘),每一个编程实例都至少介绍一种方法、解决一个问题,这些方法和问题可能与语言有关、可能与系统有关、可能与应用有关,总之,它们都可能是您在使用 Borland Delphi 2005 C# Builder 开发 Microsoft .NET Framework 应用程序时可能遇到的问题。

本书适合于使用 Borland Delphi 2005 C# Builder 进行 Microsoft .NET Framework 程序开发的初、中级用户参考,也可作为高等院校及计算机培训学校在校学生的学习案例。书中的所有实例程序均在 Borland Delphi 2005 C# Builder/Microsoft Office 2003/SQL Server 2000/Microsoft Windows Server 2003/Microsoft Visual Studio.NET 2003 环境中编译执行成功,因此,请在以上环境中进行实例程序的学习。

本书所有内容和思想并非一人之力所能及,而是凝聚了众多编程高手多年的心得并经

过充分的提炼和总结而成，在此对他们表示崇高的敬意和衷心的感谢！本书编写人员包括罗斌、罗兴禄、雷治英、罗帅、雷玲、曹勇、雷丽、李平、雷平、邓承会、雷国忠、王年素、雷浩、李佳星、范刚强、杨琴、罗顺文、吴多、王彬、张志红、罗兰、刘显国和刘婷，终稿由罗斌统筹完成。由于作者水平有限，其中也可能存在一些疏漏和不当之处，敬请读者批评指正。联系邮箱 [binluobin@163.com](mailto:binluobin@163.com) 或 [binluobin@sina.com](mailto:binluobin@sina.com)。

罗斌

2005年11月于重庆长寿

# 目 录

## 前言

<b>第 1 章 程序界面编程实例</b> .....	1	实例 31 如何创建中英文菜单	83
实例 1 如何创建换肤窗体	1	实例 32 如何创建上下文菜单	86
实例 2 如何创建闪屏窗体	3	实例 33 如何创建工具栏菜单	88
实例 3 如何创建分隔窗体	5	实例 34 如何动态增加删除菜单	93
实例 4 如何创建托盘窗体	8	实例 35 如何在窗体上实现超链接	95
实例 5 如何创建移动窗体	11	实例 36 如何在消息框中绘制表格	97
实例 6 如何创建渐变色窗体	12	<b>第 2 章 程序控制编程实例</b> .....	99
实例 7 如何创建全屏幕窗体	15	实例 37 如何设置打印机	99
实例 8 如何创建最顶层窗体	16	实例 38 如何设置打印页面	103
实例 9 如何创建文字形窗体	18	实例 39 如何实现打印预览	106
实例 10 如何创建选项卡窗体	20	实例 40 如何屏蔽鼠标消息	109
实例 11 如何创建多文档窗体	23	实例 41 如何模拟键盘输入	111
实例 12 如何创建渐隐渐显窗体	28	实例 42 如何获取设置输入法	114
实例 13 如何拖动无标题栏窗体	30	实例 43 如何制作记事本程序	116
实例 14 如何最大化最小化窗体	32	实例 44 如何创建多线程程序	124
实例 15 如何创建登录对话框	34	实例 45 如何直接返回到桌面	127
实例 16 如何创建浏览对话框	36	实例 46 如何限制软件使用次数	129
实例 17 如何创建图形组合框	39	实例 47 如何实现程序互斥运行	131
实例 18 如何设置错误提示框	42	实例 48 如何发送消息关闭程序	132
实例 19 如何设置浮动信息框	44	实例 49 如何仿真键盘按键状态	134
实例 20 如何设置公共对话框	46	实例 50 如何实现控件拖放功能	145
实例 21 如何设置消息提示框	49	实例 51 如何使用剪贴板操作文本	150
实例 22 如何设置默林精灵动作	52	实例 52 如何使用缓冲输入输出流	153
实例 23 如何设置应用程序光标	57	实例 53 如何控制进度条执行速度	156
实例 24 如何实现菜单历史列表	59	实例 54 如何创建开机自启动程序	159
实例 25 如何设置窗体启动位置	65	实例 55 如何创建字体选择对话框	161
实例 26 如何设置工具栏位置	66	实例 56 如何获取窗体中所有控件	165
实例 27 如何设置窗体状态栏	68	实例 57 如何获取重定向输出结果	166
实例 28 如何设置窗体标题栏	72	实例 58 如何获取用户共享文件夹	168
实例 29 如何设置菜单功能键	74	实例 59 如何获取程序启动文件夹	170
实例 30 如何创建自绘式菜单	78	实例 60 如何获取程序集反射内容	171

实例 61	如何获取设置当前工作目录 .....	173	实例 97	如何判断文件是否已经存在 .....	268
实例 62	如何获取程序运行环境变量 .....	176	实例 98	如何在文件中查找指定文本 .....	269
实例 63	如何获取程序文件版本信息 .....	178	实例 99	如何在指定路径中搜索文件 .....	272
实例 64	如何控制列表视图显示方式 .....	180	实例 100	如何将文本存为 Word 文件 .....	276
实例 65	如何控制图像控件显示方式 .....	184	实例 101	如何自动创建 Excel 文件 .....	278
实例 66	如何拖动滚动条浏览图像 .....	186	<b>第 4 章 系统管理编程实例 .....</b>	<b>282</b>	
实例 67	如何使用方向键浏览图像 .....	188	实例 102	如何获取本地主机名 .....	282
实例 68	如何使用指针等不安全代码 .....	190	实例 103	如何获取当前用户名 .....	283
实例 69	如何使用反射查找特殊成员 .....	193	实例 104	如何获取当前计算机名 .....	284
实例 70	如何使用反射获取类型成员 .....	196	实例 105	如何获取 .NET 版本号 .....	285
实例 71	如何使用哈希表 (反) 序列化 .....	200	实例 106	如何获取操作系统版本号 .....	286
实例 72	如何使用哈希码加密数据 .....	204	实例 107	如何获取操作系统目录 .....	288
实例 73	如何解析正则表达式集合 .....	207	实例 108	如何获取系统安装日期 .....	289
实例 74	如何使用 Monitor 同步线程 .....	210	实例 109	如何获取系统当前日期时间 .....	291
实例 75	如何模拟多线程资源共享 .....	213	实例 110	如何获取系统最近启动时间 .....	292
实例 76	如何启动电话拨号程序 .....	215	实例 111	如何获取系统配置信息 .....	294
实例 77	如何启动控制面板程序 .....	217	实例 112	如何获取系统驱动器信息 .....	299
实例 78	如何启动屏幕保护程序 .....	218	实例 113	如何获取系统所有打印机 .....	301
<b>第 3 章 文件操作编程实例 .....</b>	<b>221</b>		实例 114	如何获取系统显示分辨率 .....	303
实例 79	如何播放多媒体文件 .....	221	实例 115	如何获取系统已安装字体 .....	304
实例 80	如何操作程序帮助文件 .....	223	实例 116	如何获取系统特殊文件夹 .....	305
实例 81	如何操作文件系统对象 .....	226	实例 117	如何获取系统驱动程序 .....	309
实例 82	如何读写独立存储文件 .....	228	实例 118	如何获取系统启动方式 .....	311
实例 83	如何切分组合大型文件 .....	231	实例 119	如何获取系统日志信息 .....	312
实例 84	如何加密解密任意文件 .....	235	实例 120	如何读写事件日志信息 .....	316
实例 85	如何建立程序关联文件 .....	242	实例 121	如何获取当前计算机服务 .....	321
实例 86	如何解析文件全路径名 .....	244	实例 122	如何获取计算机服务程序 .....	324
实例 87	如何获取文件启动方式 .....	246	实例 123	如何获取计算机服务说明 .....	325
实例 88	如何获取文件尺寸大小 .....	247	实例 124	如何启动停止计算机服务 .....	327
实例 89	如何获取设置文件属性 .....	249	实例 125	如何获取计数器类别名称 .....	330
实例 90	如何获取设置文件时间 .....	253	实例 126	如何计算性能计数器数值 .....	334
实例 91	如何获取设置文件夹时间 .....	256	实例 127	如何新增系统性能计数器 .....	338
实例 92	如何获取设置文件夹属性 .....	259	实例 128	如何启动关闭指定进程 .....	341
实例 93	如何获取指定目录所有文件 .....	262	实例 129	如何获取系统指定进程信息 .....	344
实例 94	如何获取指定目录所有子目录 .....	264	实例 130	如何获取指定进程模块信息 .....	349
实例 95	如何获取文件夹所有文件目录 .....	265	实例 131	如何获取指定进程线程信息 .....	352
实例 96	如何判断文件夹是否已经存在 .....	267	实例 132	如何挂起休眠终止线程 .....	358

实例 133	如何监视系统文件变化 .....	361	实例 167	如何在数据表格中更新记录 .....	458
实例 134	如何拦截系统关机消息 .....	366	实例 168	如何使用数据适配器增加记录 .....	461
实例 135	如何通过 IP 获取主机信息 .....	367	实例 169	如何使用数据读取器获取记录 .....	464
实例 136	如何启动停止系统驱动程序 .....	369	实例 170	如何使用事务删除数据库记录 .....	465
实例 137	如何获取语言文化名称信息 .....	372	实例 171	如何使用参数化方式增加记录 .....	469
实例 138	如何获取语言文化详细信息 .....	376	实例 172	如何在 XML 文件中增加记录 .....	472
<b>第 5 章 网络开发编程实例 .....</b>	<b>378</b>		实例 173	如何使用 XML 查找 数据库记录 .....	476
实例 139	如何创建虚拟目录 .....	378	实例 174	如何使用 XML 查找 关系表记录 .....	479
实例 140	如何自制网页浏览器 .....	382	实例 175	如何获取 XML 文件 数据表设计 .....	481
实例 141	如何收听网络电台节目 .....	385	实例 176	如何使用内存流操作 无文件 XML .....	484
实例 142	如何获取电子邮件数量 .....	387	实例 177	如何直接读取 Excel 文件记录 .....	488
实例 143	如何向服务器上传文件 .....	391	实例 178	如何将记录转换成 Excel 文件 .....	491
实例 144	如何从服务器下载文件 .....	393	实例 179	如何创建数据库主从关系表 .....	494
实例 145	如何创建邮件发送程序 .....	395	实例 180	如何分表显示数据表主从关系 .....	497
实例 146	如何创建邮件接收程序 .....	398	实例 181	如何合并显示数据表字段信息 .....	500
实例 147	如何创建通信服务器程序 .....	401	实例 182	如何获取数据表字段设置信息 .....	502
实例 148	如何创建通信客户端程序 .....	404	实例 183	如何读取数据库记录图像字段 .....	505
实例 149	如何创建聊天服务器程序 .....	406	实例 184	如何使用数据视图作为数据源 .....	508
实例 150	如何创建聊天客户端程序 .....	409	实例 185	如何获取记录求和求平均值 .....	510
<b>第 6 章 GDI+特效编程实例 .....</b>	<b>412</b>		<b>第 8 章 ASP.NET Web 应用编程实例 .....</b>	<b>513</b>	
实例 151	如何拷贝贝屏幕图像 .....	412	实例 186	如何创建 Web Service 服务程序 .....	513
实例 152	如何动态显示图像 .....	414	实例 187	如何创建 Web Service 客户程序 .....	516
实例 153	如何异形化显示图像 .....	415	实例 188	如何获取 Web 服务器 变量集合信息 .....	518
实例 154	如何无级旋转显示图像 .....	418	实例 189	如何获取浏览 Web 页的 浏览器信息 .....	520
实例 155	如何无级旋转显示文字 .....	420	实例 190	如何隐藏传递 Web 页间 的调用参数 .....	524
实例 156	如何控制图像显示质量 .....	421	实例 191	如何操作 Web 客户端 Cookie 数据 .....	526
实例 157	如何无级缩放图像大小 .....	424	实例 192	如何实现全屏显示和关闭 Web 页 .....	528
实例 158	如何绘制渐变拉伸图形 .....	428			
实例 159	如何保存图形绘制结果 .....	433			
实例 160	如何动态绘制时钟指针 .....	439			
实例 161	如何设置图像填充方式 .....	441			
实例 162	如何绘制网格画刷列表 .....	444			
实例 163	如何转换图像文件格式 .....	446			
实例 164	如何获取图像像素值 .....	450			
<b>第 7 章 ADO.NET 数据访问编程实例 .....</b>	<b>453</b>				
实例 165	如何使用数据绑定浏览记录 .....	453			
实例 166	如何使用数据绑定更新记录 .....	456			



实例 193 如何在 Web 页中实现  
回车键跳转 ..... 529

实例 194 如何设置 Web 页图像  
控件对齐方式 ..... 532

实例 195 如何在 Web 页中绘制  
条形图和饼图 ..... 535

实例 196 如何在 Web 页中显示  
Excel 风格表格 ..... 537

实例 197 如何为 DataGrid 控件  
添加删除按钮 ..... 539

实例 198 如何使用 DataGrid 控件  
分页浏览记录 ..... 542

实例 199 如何动态产生 Web 页控件  
并处理事件 ..... 547

实例 200 如何在 Web 页中使用图形  
码校验登录 ..... 549

# 第 1 章 程序界面编程实例

## 实例 1 如何创建换肤窗体

### 实例目的

此实例主要介绍了如何根据图形图像样式创建程序窗体界面并进行变换的编程技术，即通常所说的程序窗体界面换肤。实例程序执行后，将显示一个不规则的图形窗体界面，如图 1-1 所示；在图 1-1 中单击“皮肤一”按钮，则显示该程序另一个不规则的图形窗体界面，如图 1-2 所示；在图 1-2 中单击“皮肤二”按钮，则显示图 1-1 所示的不规则图形窗体界面。实例程序可以通过单击“皮肤一”、“皮肤二”按钮在这两种不规则的图形窗体界面间任意切换，外观效果类似于东方影都播放器和 Windows 媒体播放器。



图 1-1

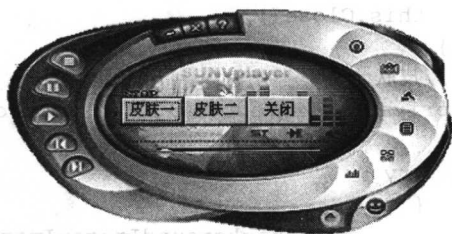


图 1-2

### 实现方法

在目前流行的大多数多媒体播放器和工具软件中，如 Windows 媒体播放器、东方影都、金山影霸等，它们的界面通常都是一些不规则的、相当个性化的图形图像界面，并且都备有多个不同的皮肤（不规则的图形窗体界面）供用户进行选择，使用户感到赏心悦目。在 .NET 中，可以通过设置应用程序窗体 WinForm 的公共属性来实现这种个性化的窗体界面。默认情况下，.NET 创建的 Windows Forms Application 应用程序窗体（WinForm）有一个 TransparencyKey 属性，该属性能够设置程序窗体透明区域的颜色，它使得 WinForm 窗体背景具有相同 TransparencyKey 属性所指定颜色的窗体区域透明显示；在窗体的透明区域执行的任何鼠标操作（如鼠标单击）都将传递到该透明区域下的窗口，如同这部分透明窗体完全不存在一样。因此，如果以个性化的图形图像作为 WinForm 窗体的背景图像，并且指定 WinForm 窗体的 TransparencyKey 属性的颜色值与个性化窗体图像中的不需要部分（透明）的颜色一致，则程序换肤窗体界面就大功告成。下面是实例程序据此在 Delphi 2005 C# Builder 中通过设置应用程序窗体 WinForm 的 TransparencyKey 属性实现创建程序换肤窗体界面的操作步骤：

- (1) 启动 Delphi 2005，选择 File\New\Windows Forms Application-C#\Builder 菜单，创建

一个 Borland Delphi 2005 C# Builder Windows Forms 应用程序框架。

(2) 在画图或其他图像处理软件中制作两幅不规则的图像并保存为 Show1.bmp 和 Show2.bmp 图像文件, 注意其透明色(不需要的部分颜色)应为一个确定的颜色, 该颜色必须与程序窗体 WinForm 的 TransparencyKey 属性指定的颜色值一致, 实例程序选择的是白色, 如图 1-1 和 BCB1-2 所示。然后复制这两个图像文件到程序运行文件夹中。

(3) 在程序窗体 WinForm (WinForm.cs) 设计视图中选择该设计窗体 WinForm, 在 Object Inspector 的属性页中设置其 TransparencyKey 属性为 White, 设置其 FormBorderStyle 属性为 None, 设置其 ShowInTaskbar 属性为 False; 单击 BackgroundImage 属性右边的“...”按钮, 在弹出的文件打开对话框中选择步骤二制作的其中一幅图像文件后, 单击“打开”按钮返回。

(4) 选择 View\Tool Palette 菜单, 指向 Tool Palette 的 Windows Forms, 在程序窗体设计视图上 (WinForm) 拖放三个 Button 控件, 在 Object Inspector 的属性页中设置三个 Button 控件的 Text 属性分别为“皮肤一”、“皮肤二”和“关闭”。

(5) 为“皮肤一”、“皮肤二”和“关闭” Button 控件的 Click 鼠标单击事件添加如下所示的实现代码:

```
//关闭
private void button3_Click(object sender, System.EventArgs e)
{
    this.Close();
}
//皮肤一
private void button1_Click(object sender, System.EventArgs e)
{
    try
    {
        this.BackgroundImage=Image.FromFile("Show2.bmp");
    }
    catch(Exception Ex)
    {
        MessageBox.Show(Ex.Message,"信息提示",
            MessageBoxButtons.OK,MessageBoxIcon.Information);
    }
}
//皮肤二
private void button2_Click(object sender, System.EventArgs e)
{
    try
    {
        this.BackgroundImage=Image.FromFile("Show1.bmp");
    }
    catch(Exception Ex)
    {
        MessageBox.Show(Ex.Message,"信息提示",
            MessageBoxButtons.OK,MessageBoxIcon.Information);
    }
}
```

最后, 编译执行程序即可。

**注意：**如果你的操作系统设置的颜色质量为“最高32位”，可能按照此种方法创建的程序换肤窗体界面无效，在此情况下，在系统显示属性中将颜色质量设置为较低的值（如16位）即可。

### 实例源代码

有关程序项目文件全部源代码及执行程序，请参考配套光盘“源代码\第1章 程序界面编程实例\实例1 如何创建换肤窗体”。

## 实例2 如何创建闪屏窗体

### 实例目的

此实例主要介绍了如何在显示应用程序主窗体前显示闪屏窗体的编程技术。实例程序启动执行后，将出现一个无标题栏的闪屏窗体，指示程序正在初始化，如图2-1所示，在间隔几秒钟后该闪屏窗体自动关闭，进入程序主窗体界面。



图2-1

### 实现方法

闪屏启动窗体通常用于需要进行长时间初始化整理的大型软件中，例如 Delphi 2005 就是一个典型的闪屏启动窗体。Delphi 2005 在启动时，会首先弹出一个闪屏窗体，指示当前正在加载的程序模块，加载完成后，才进入 Delphi 2005 的设计界面。在.NET 中，创建闪屏启动窗体可以通过两个 WinForm 实例结合定时器实现，一个 WinForm 实例用作程序主窗体，另一个 WinForm 实例用于闪屏窗体，在程序主窗体的 Load 事件中直接加载闪屏窗体即可。

下面是实例程序在 Delphi 2005 C# Builder 中实现闪屏窗体效果的操作步骤：

(1) 启动 Delphi 2005，选择 File\New\Windows Forms Application-C#Builder 菜单，创建一个 Borland Delphi 2005 C# Builder Windows Forms 应用程序框架。

(2) 选择 File\New\Windows Form-C#Builder 菜单，系统会自动生成一个 WinForm1 窗体 (WinForm1.cs)，此窗体将用作于闪屏窗体显示。

(3) 选择 View\Tool Palette 菜单，指向 Tool Palette 的 Components，在闪屏窗体设计视图上 (WinForm1) 拖放一个 Timer 组件 (timer1)。如果希望闪屏窗体停留的时间较长，可以在 Object Inspector 的属性页中设置 Timer 组件 (timer1) 的 Interval 属性为“5000 (毫秒)”或更长时间。

(4) 为 Timer 组件 (timer1) 的 Tick 事件添加如下所示的实现代码:

```
//关闭闪屏窗体
private void timer1_Tick(object sender, System.EventArgs e)
{
    this.Close();
}
```

(5) 为闪屏窗体 WinForm1 的 Load 事件和 Closed 事件添加如下所示的实现代码:

```
//关闭定时器
private void WinForm1_Closed(object sender, System.EventArgs e)
{
    this.timer1.Stop();
}
//启动定时器
private void WinForm1_Load(object sender, System.EventArgs e)
{
    this.timer1.Start();
}
```

(6) 在 WinForm1 (WinForm1.cs) 闪屏窗体设计视图中选择该设计窗体 WinForm1, 在 Object Inspector 的属性页中设置 WinForm1 窗体的 FormBorderStyle 属性为 None, 设置 ShowInTaskbar 属性为 False; 单击其 BackgroundImage 属性右边的 “...” 按钮, 在弹出的文件打开对话框中选择一幅图像文件后, 单击 “打开” 按钮返回。则此幅图像就成为闪屏窗体图像。

(7) 选择 View\Tool Palette 菜单, 指向 Tool Palette 的 Windows Forms, 在程序主窗体设计视图上 (WinForm) 拖放一个 Button 控件 (button1)。然后在 Object Inspector 的属性页中设置 Button 控件 (button1) 的 Text 属性为 “重新启动闪屏窗体”, 并为 Button 控件 (button1) 的鼠标单击事件 Click 添加如下所示的实现代码。

```
//重新启动闪屏窗体
private void button1_Click(object sender, System.EventArgs e)
{
    this.Hide();
    WinForm1 MySplashForm=new WinForm1();
    MySplashForm.ShowDialog();
    this.Show();
}
```

(8) 为应用程序主窗体 WinForm 的 Load 事件添加如下所示的实现代码:

```
//启动闪屏窗体
private void WinForm_Load(object sender, System.EventArgs e)
{
    this.button1_Click(null,null);
}
```

最后, 编译执行程序即可。

### 实例源代码

有关程序项目文件全部源代码及执行程序, 请参考配套光盘 “源代码\第 1 章 程序界面编程实例\实例 2 如何创建闪屏窗体”。

### 实例3 如何创建分隔窗体

#### 实例目的

此实例主要介绍了如何创建分隔窗体使应用程序窗体中的不同控件随窗体缩放而同步缩放的编程技术。实例程序执行后，在最上面的列表视图中选择一人，则该人员的简历会出现在左下角的文本框中，照片会出现在右下角的 PictureBox 控件中，任意拖放窗体大小，PictureBox 控件和列表视图控件也会随之改变，也可以在各个控件边际间拖动以改变各个控件的可视空间，如图 3-1 所示。



图 3-1

#### 实现方法

在 .NET Windows Forms 应用程序中，调整同一窗体中各个控件间的布局关系主要有两个因素需要考虑：一是窗体中各个控件的 Dock 属性的设置，二是 Splitter 控件的使用。

Splitter 控件通常用于在运行时调整窗体中停靠控件的大小，如 Windows 资源管理器，它的数据窗格所包含的信息在不同的时间有不同的宽度。如果一个控件可由 Splitter 控件调整其大小，则当用户将鼠标指针指向该控件的未停靠的边缘时，鼠标指针将更改外观，指示该控件的大小是可以调整的。若要调整 Splitter 控件的停靠控件的大小，应该将希望调整大小的控件停靠到一个容器（窗体）的边缘，然后将 Splitter 控件停靠到该容器的同一侧。若要确保 Splitter 控件不会将停靠的控件尺寸调整得太小而使用户无法使用，则应该设置其 MinExtra 和 MinSize 属性。MinExtra 和 MinSize 属性确定停靠到左侧和右侧（如果是水平 Splitter 控件，则停靠到顶部和底部）的控件可调整到的最小尺寸。如果 Splitter 控件停靠到的窗体上的其他控件显示特定的边框样式，则可以使用 BorderStyle 属性来匹配停靠到该窗体上的控件的边框样式。

Windows 窗体中的控件 (Control) 的 Dock 属性用于获取或设置控件停靠到父容器（窗体）的哪一个边缘，它是一个 DockStyle 枚举类型，DockStyle 枚举类型的枚举成员及其说明如表 3-1 所示。

**注意：**当控件停靠到其容器的一条边缘时，调整容器的大小时，始终使控件与那条边缘对齐。如果多个控件停靠到一条边缘，那么这些控件将按照它们的 Z 顺序并排出现，控件在 Z 顺序中的位置越高，距离容器边缘就越远。如果选择了 Left、Right、Top 或 Bottom，则调整控件的指定边缘和相对边缘的尺寸至包含控件的相应边缘的尺寸。如果选择了 Fill，则调

整控件的所有四条边以匹配包含控件的边缘。

表 3-1

成员名称	说明
Bottom	该控件的下边缘停靠在其包含控件的底部
Fill	控件的各个边缘分别停靠在其包含控件的各个边缘，并且适当调整大小
Left	该控件的左边缘停靠在其包含控件的左边缘
None	该控件未停靠
Right	该控件的右边缘停靠在其包含控件的右边缘
Top	该控件的上边缘停靠在其包含控件的顶端

下面是实例程序在 Delphi 2005 C# Builder 中通过使用 Splitter 控件和设置控件的 Dock 属性实现创建分隔窗体以实现同步缩放窗体控件效果的操作步骤：

(1) 启动 Delphi 2005，选择 File\New\Windows Forms Application-C#Builder 菜单，创建一个 Borland Delphi 2005 C# Builder Windows Forms 应用程序框架。

(2) 选择 View\Tool Palette 菜单，指向 Tool Palette 的 Windows Forms，在程序窗体设计视图上 (WinForm) 拖放一个 ListView 控件 (listView1)，在 Object Inspector 的属性页中设置其 Dock 属性为 Top，设置其 FullRowSelect 属性为 True，设置其 GridLines 属性为 True，设置其 View 属性为 Details，单击 Columns 属性右边的“...”按钮，弹出“ColumnHeader 集合编辑器”对话框，在该对话框中单击“添加”按钮，系统自动为 ListView 控件 (listView1) 添加一个列，然后在右边的“columnHeader1 属性”列表中设置该列的属性。其中，Text 属性为列标题名，TextAlign 属性为列标题的对齐形式 (Left、Center、Right)，Width 属性为列标题的宽度。请据此在 ListView 控件 (listView1) 中添加“姓名”、“职务”和“联系地址”三个列标题。

(3) 选择 View\Tool Palette 菜单，指向 Tool Palette 的 Windows Forms，在程序窗体设计视图上 (WinForm) 拖放一个 Splitter 控件 (splitter1)，在 Object Inspector 的属性页中设置其 Dock 属性为 Top。

(4) 选择 View\Tool Palette 菜单，指向 Tool Palette 的 Windows Forms，在程序窗体设计视图上 (WinForm) 拖放一个 RichTextBox 控件 (richTextBox1)，在 Object Inspector 的属性页中设置其 Dock 属性为 Left。

(5) 选择 View\Tool Palette 菜单，指向 Tool Palette 的 Windows Forms，在程序窗体设计视图上 (WinForm) 拖放一个 Splitter 控件 (splitter2)，在 Object Inspector 的属性页中设置其 Dock 属性为 Left。

(6) 选择 View\Tool Palette 菜单，指向 Tool Palette 的 Windows Forms，在程序窗体设计视图上 (WinForm) 拖放一个 PictureBox 控件 (pictureBox1)，在 Object Inspector 的属性页中设置其 Dock 属性为 Fill，设置其 SizeMode 属性为 StretchImage。

(7) 为了在程序启动时在 ListView 控件 (listView1) 中添加列表项的数据信息，为应用程序窗体 WinForm 的 Load 事件添加如下所示的实现代码。

```
//初始化列表视图数据
private void WinForm_Load(object sender, System.EventArgs e)
{
```

```
string[] MyStrs1={"张颖","销售代表","复兴门 245 号"};
this.listView1.Items.Add(new ListViewItem(MyStrs1));
string[] MyStrs2={"王伟","副总裁","罗马花园 890 号"};
this.listView1.Items.Add(new ListViewItem(MyStrs2));
string[] MyStrs3={"李芳","销售代表","芍药园小区 78 号"};
this.listView1.Items.Add(new ListViewItem(MyStrs3));
}
```

(8) 为了实现选择 ListView 控件 (listView1) 中不同的行时出现不同的信息, 为 ListView 控件 (listView1) 的 SelectedIndexChanged 事件添加如下所示的实现代码。

//显示人员详细信息

```
private void listView1_SelectedIndexChanged(object sender, System.EventArgs e)
{
    try
    {
        ListViewItem MyItem=this.listView1.SelectedItems[0];
        if(MyItem.Text=="张颖")
        {
            this.richTextBox1.Text="获北京大学心理学学士学位。她同时完成了冷食的艺术。张颖是国际美食协会的会员。";
            this.pictureBox1.Image=System.Drawing.Image.FromFile("EMPID1.bmp");
        }
        else if(MyItem.Text=="王伟")
        {
            this.richTextBox1.Text="王伟获南京大学商业学士学位,获该校国际营销博士学位。他能说流利的法语和意大利语并能阅读德语。他加入公司时是销售代表,被提拔为销售经理并升任销售副总裁。王伟是销售管理圆桌协会,北京商业总会和太平洋周边进口协会的成员。";
            this.pictureBox1.Image=System.Drawing.Image.FromFile("EMPID2.bmp");
        }
        else if(MyItem.Text=="李芳")
        {
            this.richTextBox1.Text="李芳获北京学院化学学士学位。她同时完成了食品零售管理资格程序。李芳被雇用为销售员并提升为销售代表。";
            this.pictureBox1.Image=System.Drawing.Image.FromFile("EMPID3.bmp");
        }
    }
    catch(Exception Ex)
    {
        // MessageBox.Show(Ex.Message,"信息提示",
            MessageBoxButtons.OK,MessageBoxIcon.Information);
    }
}
```

最后, 编译执行程序即可。

**说明:**实例程序所使用的图片和示例内容来自 MS Office 2000, 在运行程序前应该从 Office 2000 的示例数据库 Northwind 中复制这些图片文件到程序的运行文件夹中, 并改名为程序设置的图像文件名。

### 实例源代码

有关程序项目文件全部源代码及执行程序, 请参考配套光盘“源代码\第1章 程序界面编



程实例\实例 3 如何创建分隔窗体”。

## 实例 4 如何创建托盘窗体

### 实例目的

此实例主要介绍了如何创建托盘窗体程序的编程技术。实例程序执行后，选择“视图\显示托盘”菜单，则程序主窗体隐藏，在系统任务栏通知区域（托盘）上显示一个图标，表示程序仍在运行；用鼠标右键单击程序托盘图标，在弹出的浮动菜单中选择“显示背景一”菜单，则托盘图标和程序主窗体会同时显示，效果如图 4-1 所示；如果选择“显示窗体”菜单，则会只显示程序窗体而不会同时显示托盘图标；选择其他菜单会实现菜单标题所示的功能。

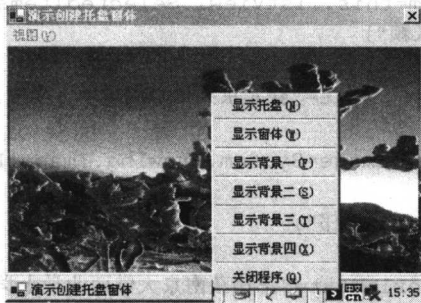


图 4-1

### 实现方法

托盘窗体通常用于在后台运行且没有用户界面的进程，但在任务栏的状态通知区域中显示图标的服务类应用程序，如金山词霸、SQL Server 服务管理等。这类程序在运行时，一般情况下不显示程序主窗体，而是在系统任务栏通知区域（托盘）中放置一个图标，表明程序仍在运行。在 .NET 中，创建托盘窗体程序可以使用 NotifyIcon 组件实现。用户只需要设置 NotifyIcon 组件属性就可以创建托盘窗体，NotifyIcon 组件（类）常用的公共属性包括：

(1) ContextMenu 属性，该属性用于获取或设置托盘图标的快捷菜单。

(2) Icon 属性，该属性用于获取或设置当前托盘图标。

(3) Text 属性，该属性用于获取或设置当鼠标停留在状态（托盘）区域图标上时显示的工具提示文本。

(4) Visible 属性，该属性用于获取或设置一个值，该值指示图标在任务栏的状态通知区域（托盘）中是否可见。

一般情况下，不需要处理 NotifyIcon 组件的默认事件。如果需要使 NotifyIcon 组件实现更多的功能，例如响应鼠标双击等，则可以处理 NotifyIcon 组件的 DoubleClick 事件、MouseDown 事件。NotifyIcon 组件常见的公共事件包括：

(1) Click 事件，该事件当用户单击状态区域的图标时发生。

(2) DoubleClick 事件，该事件当用户在任务栏的状态通知区域中双击图标时发生。