

.NET开发经典名著

Pro WPF in C# 2012: Windows Presentation
Foundation in .NET 4.5

WPF编程宝典

——使用C# 2012和.NET 4.5
(第4版)

[美] Matthew MacDonald 著
王德才 译



Apress®

清华大学出版社

.NET 开发经典名著

WPF 编程宝典

——使用 C# 2012 和 .NET 4.5(第 4 版)

[美] Matthew MacDonald 著
王德才 译



清华大学出版社

Matthew MacDonald

Pro WPF in C# 2012: Windows Presentation Foundation in .NET 4.5

EISBN: 978-1-4302-4365-6

Original English language edition published by Apress Media. Copyright © 2012 by Apress Media.
Simplified Chinese-Language edition copyright © 2013 by Tsinghua University Press. All rights reserved.

本书中文简体字版由 Apress 出版公司授权清华大学出版社出版。未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

北京市版权局著作权合同登记号 图字：01-2013-5009

本书封面贴有清华大学出版社防伪标签，无标签者不得销售。

版权所有，侵权必究。侵权举报电话：010-62782989 13701121933

图书在版编目(CIP)数据

WPF 编程宝典——使用 C# 2012 和 .NET 4.5(第 4 版) / (美)麦克唐纳(MacDonald, M.) 著;王德才 译.
—北京:清华大学出版社, 2013.8

(.NET 开发经典名著)

书名原文: Pro WPF in C# 2012: Windows Presentation Foundation in .NET 4.5

ISBN 978-7-302-32773-8

I. ①W… II. ①麦… ②王… III. ①Windows 操作系统—程序设计 ②C 语言—程序设计
IV. ①TP316.7 ②TP312

中国版本图书馆 CIP 数据核字(2013)第 136208 号

责任编辑:王 军 李维杰

装帧设计:牛艳敏

责任校对:成凤进

责任印制:李红英

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社 总 机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者:清华大学印刷厂

装 订 者:三河市新茂装订有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:59.5 字 数:1600 千字

版 次:2009 年 8 月第 1 版 2013 年 8 月第 4 版 印 次:2013 年 8 月第 1 次印刷

印 数:1~4000

定 价:128.00 元

产品编号:052335-01

致 谢

如果没有他人相助，没有哪位作者能凭一己之力完成一书籍。在撰写本书时，我极大地受惠于整个 Apress 团队，包括 Mark Powers，他全程负责本书的编辑工作；还有 Sharon Wilkey 和 Linda Seifert，他们加紧完成了审稿工作；还要感谢多位默默奉献的幕后工作者，他们负责编写索引页面、绘图以及最后的校对工作。

Fabio Claudio Ferracchiati 为本书提供了颇有见地且合乎时宜的技术分析评论，在此向他表达最诚挚的感谢。本书也吸收了来自各个 WPF 团队的众多博客的部分精髓，这些团队一直热衷于挖掘 WPF 最深层次的技术。我鼓励所有希望学习 WPF 未来版本的读者向他们学习。最后衷心感谢爱妻以及 Nora、Razia、Paul 和 Hamid，你们的支持给了我笔耕不辍的动力。感谢所有为本书作出贡献的人士！

作者简介

Matthew MacDonald 是一位作家、教育家，曾三次荣膺微软 MVP。他迄今已经撰写了十多本有关.NET 编程的书籍，包括 *Pro Silverlight 5 in C#*(由 Apress 于 2012 年出版)和 *Beginning ASP.NET 4.5 in C#*(由 Apress 于 2012 年出版)。他还曾撰写 *Your Brain: The Missing Manual*(由 O'Reilly Media 于 2008 年出版)一书，该书讲述如何最大限度地激发大脑潜能，出版后受到广大读者的热烈欢迎。Matthew 目前与妻子和两个女儿居住在多伦多。

技术编辑简介

Fabio Claudio Ferracchiati 是一位前沿技术领域的多产作家。他已经撰写了十几本有关.NET、C#、Visual Basic 以及 ASP.NET 的编程书籍。他是.NET 领域微软认证的解决方案开发专家(MCSD)，现居意大利罗马。可以访问他的博客 Ferracchiati.com。

前 言

.NET 问世之初便引入了一些重要的新技术，包括编写 Web 应用程序的全新方法(ASP.NET)、连接数据库的全新方法(ADO.NET)、新的类型安全的语言(C#和 VB.NET)以及托管的运行时(CLR)。在这些新技术中，其中一项重要技术是 Windows 窗体，它是用于构建 Windows 应用程序的类库。

尽管 Windows 窗体是一个功能完备的工具包，但它绑定到旧式的核心 Windows 技术。最重要的是，Windows 窗体依靠 Windows API 创建标准用户界面元素的可视化外观，如按钮、文本框和复选框等。所以这些要素在本质上是不可定制的。例如，如果希望创建时髦的光晕按钮，就需要创建自定义控件，并使用低级的绘图模型为按钮(各种不同的状态)绘制各个方面的细节。更糟的是，普通窗口被切割成不同的区域，每个控件完全拥有自己的区域。所以没有较好的绘制方法可将一个控件的内容(如按钮背后的辉光效果)延伸到其他控件所占的区域中。更不要指望实现动画效果，如旋转文本、闪烁按钮、收缩窗口以及实时预览等，因为对于这些效果必须手工绘制每个细节。

WPF(Windows Presentation Foundation)通过引入一个使用完全不同技术的新模型改变了所有这一切。尽管 WPF 也提供了大家熟悉的标准控件，但它“自行”绘制每个文本、边框和背景填充。所以 WPF 的功能更强大，可以改变渲染屏幕上所有内容的方式。使用这些特性，可重新设置常见控件的样式(如按钮)，并且通常不需要编写任何代码。同样，可使用变换对象旋转、拉伸、缩放以及扭曲用户界面中的所有内容，甚至可使用 WPF 动画系统对用户界面中的内容进行变换。并且因为 WPF 引擎将在窗口上渲染的内容作为单独操作的一部分，所以能处理任意多层相互重叠的控件，即使这些控件具有不规则的形状且是半透明的也同样如此。

在 WPF 这些新特性的背后是基于 DirectX 的功能强大的基础结构，DirectX 是一套硬件加速的图形 API，通常用于开发最前沿的计算机游戏。这意味着可使用丰富的图形效果，而不会损失性能，而使用 Windows 窗体实现此类效果会严重影响程序运行的性能。实际上，甚至可使用更高级特性，例如对视频文件和 3D 内容的支持。使用这些特性以及优秀的设计工具，可创建出令人赏心悦目的用户界面和可视化效果，而使用 Windows 窗体技术是无法实现这些效果的。

还有必要指出，可使用 WPF 的标准控件和简单的可视化外观来构建普通 Windows 应用程序。实际上，在 WPF 中，可以像在旧式 Windows 窗体模型中那样方便地使用通用控件。更值得一提的是，WPF 增强了商业开发人员所需要的特性，包括大幅改进的数据绑定模型、一套用于打印及管理打印队列的新类，以及用于显示大量格式化文本的文档特性。甚至提供了用于构建基于页面的应用程序的模型，这种应用程序可在 Internet Explorer 中流畅运行，

并能从 Web 站点启动，所有这些操作都不会出现常见的安全警告和令人讨厌的安装提示。总之，WPF 将以前 Windows 开发领域中的精华与当今的创新技术融为一体，得以构建现代化的富图形用户界面。

关于本书

本书深刻地介绍 WPF 技术，面向了解 .NET 平台、C# 语言以及 Visual Studio 开发环境的专业开发人员。在学习本书前，不需要具备使用以前版本 WPF 的经验，而使用过 WPF 的开发人员可以通过阅读每章开头的“新增功能”来了解 WPF 新特性。

本书全面描述 WPF 的所有主要特性，从 XAML(用于定义 WPF 用户界面的标记语言)到 3D 绘图和动画。本书个别之处会编写涉及 .NET Framework 其他特性的代码，如用于查询数据库的 ADO.NET 类。本书中不讨论这些内容。但如果需要了解有关 .NET(而非特定于 WPF)的特性的更多信息，请参阅 Apress 出版的许多专门介绍 .NET 的书籍。

内容概览

本书共包括 33 章。如果刚开始学习 WPF，将发现按照章节顺序阅读本书是最容易的方法，因为后续章节常用到前面章节中演示的技术。

下面是本书每一章的主要内容：

第 1 章：WPF 概述 介绍 WPF 的体系结构，WPF 的 DirectX 基础设施，以及新的能自动改变用户界面尺寸的设备无关度量系统。

第 2 章：XAML 介绍用于定义用户界面的 XAML 标准。该章将讨论为什么创建 XAML 以及 XAML 的工作原理，并将用不同的编码方法创建基本的 WPF 窗口。

第 3 章：布局 深入研究在 WPF 窗口中用于组织元素的布局面板。该章将分析不同布局策略，并将构建一些普通类型的窗口。

第 4 章：依赖项属性 介绍 WPF 如何使用依赖项属性来支持重要特性，如数据绑定和动画。

第 5 章：路由事件 介绍 WPF 如何使用事件路由在用户界面元素中发送冒泡路由事件或隧道路由事件，还介绍所有 WPF 元素都支持的一组基本鼠标、键盘以及多点触控事件。

第 6 章：控件 分析所有 Windows 开发人员都十分熟悉的控件，如按钮、文本框和标签，还讨论它们在 WPF 中的区别。

第 7 章：Application 类 介绍 WPF 应用程序模型。在该章中您将看到如何创建单实例和基于文档的 WPF 应用程序。

第 8 章：元素绑定 介绍 WPF 数据绑定。在该章中您将看到如何将任意类型的对象绑定到用户界面。

第 9 章：命令 介绍 WPF 命令模型，使用 WPF 命令模型可将多个控件连接到同一个逻辑操作。

第 10 章：资源 介绍如何使用资源在程序集中嵌入二进制文件，以及如何在整个用户界面中重用重要的对象。

第 11 章：样式和行为 解释 WPF 样式系统，使用 WPF 样式可为一整组控件应用一套通用属性值。

第 12 章：形状、画刷和变换 介绍 WPF 中的 2D 绘图模型。在该章中您将学习如何创建形状、使用变换改变元素，以及使用渐变画刷、图像画刷和平铺图像画刷绘制特殊效果。

第 13 章：几何形状和图画 深入分析 2D 绘图。在该章中您将学习如何创建包含弧线和曲线的复杂路径，以及如何高效地使用复杂图形。

第 14 章：效果和可视化对象 介绍低级图形编程。在该章中您将使用像素着色器应用 Photoshop 风格的效果，手动构建位图，并为了优化绘图性能而使用 WPF 的可视化层。

第 15 章：动画基础 研究 WPF 的动画框架，通过 WPF 动画框架可使用简单的声明式标记将动态效果集成到应用程序中。

第 16 章：高级动画 研究更高级的动画技术，如关键帧动画、基于路径的动画以及基于帧的动画。该章还将列举一个详细示例，展示如何使用代码创建和管理动态的动画。

第 17 章：控件模板 介绍如何通过插入定制的模板来为任意 WPF 控件提供动态的新外观(以及新行为)，您还将看到如何使用模板构建能够换肤的应用程序。

第 18 章：自定义元素 研究如何扩展现有的 WPF 控件，以及如何创建自己的控件。在该章中您将看到几个示例，包括基于模板的颜色拾取器、可翻转的面板、自定义的布局容器，以及执行自定义绘图的装饰元素。

第 19 章：数据绑定 展示如何从数据库获取信息，将获取的信息插入到自定义的数据对象中，并将这些数据对象绑定到 WPF 控件。您还将学习如何借助虚拟化技术提高大型数据绑定列表的性能，以及如何使用验证方法捕获编辑错误。

第 20 章：格式化绑定的数据 展示将原始数据转换为包含图片、控件以及选择效果的富数据显示的一些技巧。

第 21 章：数据视图 分析如何在数据绑定窗口中使用视图在数据项列表中导航，以及应用过滤、分类和分组。

第 22 章：列表、树和网格 带您浏览 WPF 中的富数据控件，包括 ListView、TreeView 和 DataGrid。

第 23 章：窗口 分析 WPF 中窗口的工作原理。在该章中您还将学习如何创建不规则形状的窗口，以及如何使用 Vista 玻璃效果，您还将通过定制任务栏跳转列表、缩略图以及图标重叠实现大部分 Windows 7 特性。

第 24 章：页面和导航 介绍如何使用 WPF 构建页面，以及保持跟踪导航历史。该章还将介绍如何构建驻留于浏览器中的 WPF 应用程序，这种应用程序可从 Web 站点启动。

第 25 章：菜单、工具栏和功能区 分析面向命令的控件，如菜单和工具栏。在该章还将使用可免费下载的 Ribbon 控件尝试更富有现代气息的用户界面。

第 26 章：声音和视频 介绍 WPF 媒体支持。在该章中您将看到如何控制声音和视频的播放，以及如何合成动画和生动鲜活的效果。

第 27 章：3D 绘图 研究 WPF 中对绘制 3D 图形的支持。在该章将学习如何创建和变换 3D 对象，以及如何为 3D 对象应用动画效果，甚至还会看到如何在 3D 表面上放置可交互的 2D 控件。

第 28 章：文档 介绍 WPF 的富文档支持。在该章中您将学习如何使用流文档以尽可能

便于阅读的方式呈现大量文本，并将学习如何用固定文档显示准备打印的页面，甚至还将学习如何使用 RichTextBox 控件提供文档编辑功能。

第 29 章：打印 演示 WPF 的打印模型，可通过打印模型在打印文档中绘制文本和图形。在该章中您还将学习如何管理页面设置和打印队列。

第 30 章：与 Windows 窗体进行交互 分析如何在同一个应用程序——甚至在同一个窗口中，结合使用 WPF 和 Windows 窗体内容。

第 31 章：多线程 介绍如何创建具有良好响应能力，在后台执行耗时任务的 WPF 应用程序。

第 32 章：插件模型 展示如何创建可扩展的、能动态发现和加载独立组件的应用程序。

第 33 章：ClickOnce 部署 展示如何使用 ClickOnce 安装模型部署 WPF 应用程序。

使用本书的前提条件

为运行 WPF 4.5 应用程序，计算机必须安装 Windows 7、Windows 8 或带有 Service Pack 2 的 Windows Vista，还需要 .NET Framework 4.5。为创建 WPF 4.5 应用程序(并打开本书中提供的示例项目)，需要安装 Visual Studio 2012，Visual Studio 2012 中包含了 .NET Framework 4.5。

还有一种选择。不使用任何版本的 Visual Studio，可使用 Expression Blend(一种面向图形的设计工具)来构建和测试 WPF 应用程序。总体而言，Expression Blend 是面向图形设计人员的工具，他们使用该工具创建绚丽夺目的内容；而对于编写大量代码的编程人员来说，Visual Studio 则是理想工具。本书假定使用的是 Visual Studio。如果准备使用 Expression Blend，务必选用明确支持 WPF 的版本(与某些 Visual Studio 版本绑定在一起的版本仅用于 Metro 开发，不支持 WPF)。到撰写本书时为止，支持 WPF 的 Expression Blend 版本是称为 Blend + Sketchflow Preview for Visual Studio 2012 的预览版本，网址是 <http://tinyurl.com/cgar5lz>。

代码示例和 URL

查看 Apress 网站或 www.prosetech.com 以下载最新的示例代码是个好主意。测试在本书中介绍的大部分更复杂的示例需要用到这些代码示例，因为在本书示例中那些较次要的细节通常被忽略了。本书关注最重要的部分，以免为阐明概念而无谓地占用过多篇幅。

为下载本书的源代码，可访问 Web 站点 <http://www.prosetech.com> 查找本书的页面，也可从 <http://www.tupwk.com.cn/downpage> 下载本书的源代码。您将发现在本书中提及的链接的列表，从而不需要键入任何内容就可以找到重要工具和例子。

反馈

本书力争成为 WPF 编程爱好者的最佳辅导和参考资料。为达到该目标，您的评论和建议对我们来说是非常有帮助的。您可将本书的缺点、优点及其他反馈信息直接发送到邮箱 wkservice@vip.163.com，我们将不胜感激。

目 录

第 I 部分 基础知识

第 1 章 WPF 概述	3
1.1 Windows 图形演化	3
1.1.1 DirectX: 新的图形引擎	4
1.1.2 硬件加速与 WPF	4
1.2 WPF: 高级 API	4
1.3 分辨率无关性	5
1.3.1 WPF 单位	6
1.3.2 系统 DPI	7
1.3.3 位图和矢量图形	9
1.4 WPF 体系结构	10
1.5 WPF 4.5	13
1.5.1 WPF 工具包	14
1.5.2 Visual Studio 2012	14
1.6 小结	16
第 2 章 XAML	17
2.1 理解 XAML	17
2.1.1 WPF 之前的图形用户界面	17
2.1.2 XAML 变体	19
2.1.3 XAML 编译	19
2.2 XAML 基础	20
2.2.1 XAML 名称空间	21
2.2.2 代码隐藏类	22
2.3 XAML 中的属性和事件	24
2.3.1 简单属性与类型转换器	25
2.3.2 复杂属性	26
2.3.3 标记扩展	28
2.3.4 附加属性	29
2.3.5 嵌套元素	30
2.3.6 特殊字符与空白	32
2.3.7 事件	34
2.3.8 完整的 Eight Ball Answer 示例	35
2.4 使用其他名称空间中的类型	36
2.5 加载和编译 XAML	38
2.5.1 只使用代码	38
2.5.2 使用代码和未经编译的 XAML	40
2.5.3 使用代码和编译过的 XAML	42
2.5.4 只使用 XAML	44
2.6 小结	45
第 3 章 布局	47
3.1 理解 WPF 中的布局	47
3.1.1 WPF 布局原则	47
3.1.2 布局过程	48
3.1.3 布局容器	48
3.2 使用 StackPanel 面板进行简单布局	50
3.2.1 布局属性	52
3.2.2 对齐方式	52
3.2.3 边距	53
3.2.4 最小尺寸、最大尺寸以及显式地设置尺寸	54
3.2.5 Border 控件	56
3.3 WrapPanel 和 DockPanel 面板	57
3.3.1 WrapPanel 面板	57
3.3.2 DockPanel 面板	58
3.3.3 嵌套布局容器	59
3.4 Grid 面板	60
3.4.1 调整行和列	63
3.4.2 布局舍入	64
3.4.3 跨越行和列	65
3.4.4 分割窗口	66
3.4.5 共享尺寸组	69
3.4.6 UniformGrid 面板	72

3.5	使用 Canvas 面板进行基于坐标的布局	72	5.4.2	焦点	111
3.5.1	Z 顺序	73	5.4.3	获取键盘状态	112
3.5.2	InkCanvas 元素	74	5.5	鼠标输入	113
3.6	布局示例	76	5.5.1	鼠标单击	115
3.6.1	列设置	76	5.5.2	捕获鼠标	115
3.6.2	动态内容	77	5.5.3	鼠标拖放	116
3.6.3	组合式用户界面	79	5.6	多点触控输入	118
3.7	小结	80	5.6.1	多点触控的输入层次	119
第 4 章	依赖项属性	83	5.6.2	原始触控	119
4.1	理解依赖项属性	83	5.6.3	操作	122
4.1.1	定义依赖项属性	84	5.6.4	惯性	124
4.1.2	注册依赖项属性	84	5.7	小结	125
4.1.3	添加属性包装器	86	第 II 部分 进一步研究 WPF		
4.1.4	WPF 使用依赖项属性的方式	87	第 6 章	控件	129
4.1.5	共享的依赖项属性	88	6.1	控件类	129
4.1.6	附加的依赖项属性	88	6.1.1	背景画刷和前景画刷	130
4.2	属性验证	90	6.1.2	字体	132
4.2.1	验证回调	90	6.1.3	鼠标光标	136
4.2.2	强制回调	91	6.2	内容控件	137
4.3	小结	93	6.2.1	Content 属性	138
第 5 章	路由事件	95	6.2.2	对齐内容	140
5.1	理解路由事件	95	6.2.3	WPF 内容原则	141
5.1.1	定义、注册和封装路由事件	95	6.2.4	标签	142
5.1.2	共享路由事件	96	6.2.5	按钮	142
5.1.3	引发路由事件	96	6.2.6	工具提示	145
5.1.4	处理路由事件	97	6.3	特殊容器	151
5.2	事件路由	99	6.3.1	ScrollViewer	152
5.2.1	RoutedEventArgs 类	100	6.3.2	GroupBox	154
5.2.2	冒泡路由事件	100	6.3.3	TabItem	154
5.2.3	处理挂起的事件	103	6.3.4	Expander	155
5.2.4	附加事件	103	6.4	文本控件	158
5.2.5	隧道路由事件	105	6.4.1	多行文本	158
5.3	WPF 事件	106	6.4.2	选择文本	159
5.3.1	生命周期事件	106	6.4.3	拼写检查	160
5.3.2	输入事件	108	6.4.4	PasswordBox	162
5.4	键盘输入	108	6.5	列表控件	162
5.4.1	处理按键事件	109	6.5.1	ListBox	163
			6.5.2	ComboBox	166

6.6	基于范围的控件	166	8.2.1	Source 属性	212
6.6.1	Slider	167	8.2.2	RelativeSource 属性	212
6.6.2	ProgressBar	168	8.2.3	DataContext 属性	213
6.7	日期控件	169	8.3	小结	214
6.8	小结	171	第 9 章 命令		215
第 7 章 Application 类		173	9.1	理解命令	215
7.1	应用程序的生命周期	173	9.2	WPF 命令模型	216
7.1.1	创建 Application 对象	173	9.2.1	ICommand 接口	217
7.1.2	派生自定义的 Application 类	174	9.2.2	RoutedCommand 类	217
7.1.3	应用程序的关闭方式	176	9.2.3	RoutedUICommand 类	218
7.1.4	应用程序事件	177	9.2.4	命令库	219
7.2	Application 类的任务	179	9.3	执行命令	220
7.2.1	显示初始界面	179	9.3.1	命令源	220
7.2.2	处理命令行参数	180	9.3.2	命令绑定	221
7.2.3	访问当前 Application 对象	181	9.3.3	使用多命令源	223
7.2.4	在窗口之间进行交互	182	9.3.4	微调命令文本	224
7.2.6	单实例应用程序	184	9.3.5	直接调用命令	224
7.3	程序集资源	189	9.3.6	禁用命令	225
7.3.1	添加资源	189	9.3.7	具有内置命令的控件	227
7.3.2	检索资源	190	9.4	高级命令	229
7.3.3	pack URI	192	9.4.1	自定义命令	229
7.3.4	内容文件	193	9.4.2	在不同位置使用相同的命令	230
7.4	本地化	193	9.4.3	使用命令参数	232
7.4.1	构建能够本地化的用户界面	194	9.4.4	跟踪和翻转命令	233
7.4.2	使应用程序为本地化做好准备	194	9.5	小结	237
7.4.3	管理翻译过程	195	第 10 章 资源		239
7.5	小结	200	10.1	资源基础	239
第 8 章 元素绑定		201	10.1.1	资源集合	239
8.1	将元素绑定到一起	201	10.1.2	资源的层次	241
8.1.1	绑定表达式	202	10.1.3	静态资源和动态资源	242
8.1.2	绑定错误	203	10.1.4	非共享资源	243
8.1.3	绑定模式	203	10.1.5	通过代码访问资源	244
8.1.4	使用代码创建绑定	205	10.1.6	应用程序资源	244
8.1.5	使用代码检索绑定	206	10.1.7	系统资源	245
8.1.6	多绑定	207	10.2	资源字典	246
8.1.7	绑定更新	210	10.2.1	创建资源字典	246
8.1.8	绑定延迟	211	10.2.2	使用资源字典	247
8.2	绑定到非元素对象	211	10.2.3	在程序集之间共享资源	248

10.3	小结	251	12.2.6	VisualBrush 画刷	297
第 11 章	样式和行为	253	12.2.7	BitmapCacheBrush 画刷	298
11.1	样式基础	253	12.3	变换	299
11.1.1	创建样式对象	256	12.3.1	变换形状	300
11.1.2	设置属性	257	12.3.2	变换元素	302
11.1.3	关联事件处理程序	258	12.4	透明	303
11.1.4	多层样式	259	12.4.1	使元素半透明	303
11.1.5	通过类型自动应用样式	261	12.4.2	透明掩码	304
11.2	触发器	262	12.5	小结	306
11.2.1	简单触发器	262	第 13 章	几何图形和图画	307
11.2.2	事件触发器	264	13.1	路径和几何图形	307
11.3	行为	266	13.1.1	直线、矩形和椭圆图形	308
11.3.1	获取行为支持	266	13.1.2	使用 GeometryGroup 组合形状	309
11.3.2	理解行为模型	267	13.1.3	使用 CombinedGeometry 融合 几何图形	311
11.3.3	创建行为	268	13.1.4	使用 PathGeometry 绘制曲线和 直线	313
11.3.4	使用行为	270	13.1.5	微语言几何图形	318
11.3.5	Blend 中的设计时行为支持	271	13.1.6	使用几何图形进行剪裁	319
11.4	小结	271	13.2	图画	320
第 III 部分 图画和动画			13.2.1	显示图画	322
第 12 章	形状、画刷和变换	275	13.2.2	导出插图	324
12.1	理解形状	275	13.3	小结	326
12.1.1	Shape 类	276	第 14 章	效果和可视化对象	327
12.1.2	矩形和椭圆	277	14.1	可视化对象	327
12.1.3	改变形状的尺寸和放置形状	278	14.1.1	绘制可视化对象	328
12.1.4	使用 Viewbox 控件缩放形状	280	14.1.2	在元素中封装可视化对象	329
12.1.5	直线	282	14.1.3	命中测试	332
12.1.6	折线	283	14.1.4	复杂的命中测试	334
12.1.7	多边形	284	14.2	效果	338
12.1.8	直线线帽和直线交点	286	14.2.1	BlurEffect 类	338
12.1.9	点划线	287	14.2.2	DropShadowEffect 类	339
12.1.10	像素对齐	288	14.2.3	ShaderEffect 类	340
12.2	画刷	289	14.3	WriteableBitmap 类	342
12.2.1	SolidColorBrush 画刷	290	14.3.1	生成位图	342
12.2.2	LinearGradientBrush 画刷	290	14.3.2	写入 WriteableBitmap 对象	343
12.2.3	RadialGradientBrush 画刷	292	14.3.3	更高效的像素写入	345
12.2.4	ImageBrush 画刷	294			
12.2.5	平铺的 ImageBrush 画刷	295			

14.4	小结	347	16.4	基于帧的动画	399
第 15 章	动画基础	349	16.5	使用代码创建故事板	402
15.1	理解 WPF 动画	349	16.5.1	创建主窗口	403
15.1.1	基于时间的动画	349	16.5.2	创建 Bomb 用户控件	405
15.1.2	基于属性的动画	350	16.5.3	投弹	406
15.2	基本动画	351	16.5.4	拦截炸弹	409
15.2.1	Animation 类	351	16.5.5	统计炸弹和清理工作	410
15.2.2	使用代码创建动画	353	16.6	小结	412
15.2.3	同时发生的动画	358	第 IV 部分 模板和自定义元素		
15.2.4	动画的生命周期	358	第 17 章	控件模板	417
15.2.5	Timeline 类	359	17.1	理解逻辑树和可视化树	417
15.3	故事板	362	17.2	理解模板	422
15.3.1	故事板	363	17.2.1	修饰类	424
15.3.2	事件触发器	363	17.2.2	剖析控件	426
15.3.3	重叠动画	366	17.3	创建控件模板	428
15.3.4	同步的动画	367	17.3.1	简单按钮	429
15.3.5	控制播放	367	17.3.2	模板绑定	430
15.3.6	监视动画进度	371	17.3.3	改变属性的触发器	431
15.4	动画缓动	373	17.3.4	使用动画的触发器	434
15.4.1	使用缓动函数	373	17.4	组织模板资源	435
15.4.2	在动画开始时应用缓动与在动画 结束时应用缓动	374	17.4.1	分解按钮控件模板	436
15.4.3	缓动函数类	375	17.4.2	通过样式应用模板	438
15.4.4	创建自定义缓动函数	377	17.4.3	自动应用模板	439
15.5	动画性能	379	17.4.4	由用户选择的皮肤	440
15.5.1	期望的帧率	380	17.5	构建更复杂的模板	442
15.5.2	位图缓存	382	17.5.1	嵌套的模板	443
15.6	小结	384	17.5.2	修改滚动条	445
第 16 章	高级动画	385	17.5.3	控件模板示例	450
16.1	动画类型回顾	385	17.6	可视化状态	451
16.1.1	动态变换	386	17.7	小结	452
16.1.2	动态改变画刷	390	第 18 章	自定义元素	453
16.1.3	动态改变像素着色器	392	18.1	理解 WPF 中的自定义元素	454
16.2	关键帧动画	393	18.2	构建基本的用户控件	456
16.2.1	离散的关键帧动画	395	18.2.1	定义依赖项属性	456
16.2.2	缓动关键帧	395	18.2.2	定义路由事件	459
16.2.3	样条关键帧动画	396	18.2.3	添加标记	460
16.3	基于路径的动画	397	18.2.4	使用控件	462

18.2.5	命令支持	462
18.2.6	深入分析用户控件	465
18.3	创建无外观控件	466
18.3.1	修改颜色拾取器的代码	466
18.3.2	修改颜色拾取器的标记	467
18.3.3	精简控件模板	469
18.4	支持可视化状态	472
18.4.1	开始编写 FlipPanel 类	473
18.4.2	选择部件和状态	475
18.4.3	默认控件模板	476
18.4.4	使用 FlipPanel 控件	482
18.4.5	使用不同的控件模板	483
18.5	创建自定义面板	485
18.5.1	两步布局过程	485
18.5.2	Canvas 面板的副本	488
18.5.3	更好的 WrapPanel 面板	489
18.6	自定义绘图元素	492
18.6.1	OnRender()方法	493
18.6.2	评估自定义绘图	494
18.6.3	自定义绘图元素	495
18.6.4	创建自定义装饰元素	497
18.7	小结	498

第 V 部分 数据

第 19 章	数据绑定	501
19.1	使用自定义对象绑定到数据库	501
19.1.1	构建数据访问组件	502
19.1.2	构建数据对象	504
19.1.3	显示绑定对象	505
19.1.4	更新数据库	508
19.1.5	更改通知	508
19.2	绑定到对象集合	510
19.2.1	显示和编辑集合项	510
19.2.2	插入和移除集合项	513
19.2.3	绑定到 ADO.NET 对象	514
19.2.4	绑定到 LINQ 表达式	515
19.3	提高大列表的性能	518
19.3.1	虚拟化	518

19.3.2	项容器再循环	519
19.3.3	缓存长度	519
19.3.4	延迟滚动	520
19.4	验证	521
19.4.1	在数据对象中进行验证	521
19.4.2	自定义验证规则	526
19.4.3	响应验证错误	528
19.4.4	获取错误列表	529
19.4.5	显示不同的错误指示符号	530
19.4.6	验证多个值	533
19.5	数据提供者	535
19.5.1	ObjectDataProvider	536
19.5.2	XmlDataProvider	539
19.6	小结	541

第 20 章 格式化绑定的数据

20.1	数据绑定回顾	543
20.2	数据转换	544
20.2.1	使用 StringFormat 属性	545
20.2.2	值转换器简介	546
20.2.3	使用值转换器设置字符串的格式	547
20.2.4	使用值转换器创建对象	549
20.2.5	应用条件格式化	551
20.2.6	评估多个属性	552
20.3	列表控件	554
20.4	列表样式	555
20.4.1	ItemContainerStyle	556
20.4.2	包含复选框或单选按钮的 ListBox 控件	557
20.4.3	交替条目样式	559
20.4.4	样式选择器	561
20.5	数据模板	564
20.5.1	分离和重用模板	566
20.5.2	使用更高级的模板	567
20.5.3	改变模板	569
20.5.4	模板选择器	570
20.5.5	模板与选择	573
20.5.6	改变项的布局	577

20.6	ComboBox 控件	578
20.7	小结	581
第 21 章	数据视图	583
21.1	View 对象	583
21.1.1	检索视图对象	584
21.1.2	视图导航	584
21.1.3	以声明方式创建视图	587
21.2	过滤、排序与分组	588
21.2.1	过滤集合	588
21.2.2	过滤 DataTable 对象	591
21.2.3	排序	592
21.2.4	分组	593
21.2.5	实时成型	598
21.3	小结	599
第 22 章	列表、树和网格	601
22.1	ListView 控件	601
22.1.1	使用 GridView 创建列	602
22.1.2	创建自定义视图	606
22.2	TreeView 控件	613
22.2.1	创建数据绑定的 TreeView 控件	614
22.2.2	将 DataSet 对象绑定到 TreeView 控件	617
22.2.3	即时创建节点	618
22.3	DataGrid 控件	621
22.3.1	改变列的尺寸与重新安排列	622
22.3.2	定义列	623
22.3.3	设置列的格式和样式	628
22.3.4	设置行的格式	629
22.3.5	显示行细节	630
22.3.6	冻结列	631
22.3.7	选择	632
22.3.8	排序	632
22.3.9	编辑	633
22.4	小结	635
第 VI 章 窗口、页面和富控件		
第 23 章	窗口	639
23.1	Window 类	639
23.1.1	显示窗口	641
23.1.2	定位窗口	642
23.1.3	保存和还原窗口位置	642
23.2	窗口交互	644
23.2.1	窗口所有权	646
23.2.2	对话框模型	647
23.2.3	通用对话框	648
23.3	非矩形窗口	649
23.3.1	简单形状窗口	649
23.3.2	具有形状内容的透明窗口	651
23.3.3	移动形状窗口	653
23.3.4	改变形状窗口的尺寸	653
23.3.5	组合到一起: 窗口的自定义控件模板	655
23.4	Windows 7 任务栏编程	658
23.4.1	使用跳转列表	659
23.4.2	改变任务栏图标和预览	663
23.5	小结	667
第 24 章	页面和导航	669
24.1	基于页面的导航	669
24.2	基于页面的界面	670
24.2.1	创建一个具有导航窗口的基于页面的简单应用程序	670
24.2.2	Page 类	672
24.2.3	超链接	673
24.2.4	在框架中驻留页面	675
24.2.5	在另一个页面中驻留页面	677
24.2.6	在 Web 浏览器中驻留页面	678
24.3	页面历史	678
24.3.1	深入分析 WPF 中的 URI	678
24.3.2	导航历史	679
24.3.3	维护自定义的属性	681
24.4	导航服务	682
24.4.1	通过编程进行导航	682
24.4.2	导航事件	683

24.4.3	管理日志	684	第 26 章	声音和视频	731
24.4.4	向日志添加自定义项	685	26.1	播放 WAV 音频	731
24.4.5	使用页函数	689	26.1.1	SoundPlayer 类	731
24.5	XAML 浏览器应用程序	692	26.1.2	SoundPlayerAction 类	733
24.5.1	创建 XBAP 应用程序	692	26.1.3	系统声音	733
24.5.2	部署 XBAP 应用程序	693	26.2	MediaPlayer 类	734
24.5.3	更新 XBAP 应用程序	695	26.3	MediaElement 类	735
24.5.4	XBAP 应用程序的安全性	695	26.3.1	使用代码播放音频	736
24.5.5	完全信任的 XBAP 应用程序	696	26.3.2	处理错误	737
24.5.6	组合 XBAP/独立应用程序	697	26.3.3	使用触发器播放音频	737
24.5.7	为不同的安全级别编写代码	697	26.3.4	播放多个声音	739
24.5.8	在网页中嵌入 XBAP 应用程序	702	26.3.5	改变音量、平衡、速度以及 位置	740
24.6	WebBrowser 控件	702	26.3.6	将动画同步到音频	742
24.6.1	导航到页面	703	26.3.7	播放视频	744
24.6.2	构建 DOM 树	704	26.3.8	视频效果	744
24.6.3	使用 .NET 代码为网页添加 脚本	706	26.4	语音	747
24.7	小结	708	26.4.1	语音合成	747
第 25 章	菜单、工具栏和功能区	709	26.4.2	语音识别	749
25.1	菜单	709	26.5	小结	751
25.1.1	Menu 类	710	第 27 章	3D 绘图	753
25.1.2	菜单项	710	27.1	3D 绘图基础	753
25.1.3	ContextMenu 类	712	27.1.1	视口	754
25.1.4	菜单分隔条	713	27.1.2	3D 对象	754
25.2	工具栏和状态栏	714	27.1.3	摄像机	761
25.2.1	ToolBar 控件	714	27.2	深入研究 3D 绘图	765
25.2.2	StatusBar 控件	717	27.2.1	着色和法线	766
25.3	功能区	718	27.2.2	更复杂的形状	769
25.3.1	添加功能区	719	27.2.3	Model3DGroup 集合	769
25.3.2	应用程序菜单	720	27.2.4	使用材质	771
25.3.3	选项卡、组与按钮	722	27.2.5	纹理映射	773
25.3.4	富工具提示	724	27.3	交互和动画	777
25.3.5	带有快捷键提示的键盘访问	725	27.3.1	变换	777
25.3.6	改变功能区的大小	726	27.3.2	旋转	778
25.3.7	快速访问工具栏	729	27.3.3	飞过	779
25.4	小结	730	27.3.4	跟踪球	781
			27.3.5	命中测试	782
			27.3.6	3D 表面上的 2D 元素	786