

<<WPF编程宝典>>

图书基本信息

书名：<<WPF编程宝典>>

13位ISBN编号：9787302252696

10位ISBN编号：7302252696

出版时间：2011-6

出版时间：清华大学

作者：麦克唐纳

译者：王德才

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.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，其中的一项重要技术是Windows窗体，它是用于构建Windows应用程序的类库。

尽管Windows窗体是一个成熟且功能完整的工具包，但它使用的是在过去十年中基本没有变化的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，如果希望创建一个外观时髦的光晕按钮，需要创建一个自定义控件，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。

更糟糕的是，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，没有比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。

Windows演示基础（Windows Presentation Foundation，WPF）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也提供了熟悉的标准控件，但是它“自行”绘制每个文本、边框和背景填充。

所以，WPF能提供更强大的功能，可以改变渲染屏幕上所有内容的方式。

使用这些特性，可以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，可以使用变换对象旋转、拉伸、缩放以及扭曲用户界面中的所有内容，甚至可以使用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，所以它能够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。

这意味着可以使用丰富的图形效果，而不会损失性能，而使用Windows窗体实现这类效果会严重影响程序运行的性能。

实际上，甚至可以使用更高级特性，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），可以创建出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。

尽管通常最关注WPF中最前沿的视频、动画以及3D特性，但有必要指出，可以使用WPF的标准控件和简单的可视化外观来构建常规Windows应用程序。

实际上，使用WPF中的通用控件与使用Windows窗体中的通用控件同样容易。

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、一套用于打印以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，这种应用程序可以在Internet Explorer中无缝地运行，并且能够从Web站点加载，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开发领域中的精华和当今的创新技术融为一体，来构建现代的、富图形用户界面。

尽管Windows窗体应用程序还将继续维持数年，但是从事新的Windows开发项目的开发人员应当首选WPF。

关于本书 本书针对了解.NET平台、C#语言以及Visual Studio开发环境的专业开发人员，深入分析了WPF。

<<WPF编程宝典>>

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“新增功能”中突出显示的WPF新特性。

本书全面描述了所有WPF的主要特性，从XAML（用于定义WPF用户界面的标记语言）到3D绘图和动画。

本书很少编写涉及.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。

但如果需要更多有关.NET而非特定于WPF的特性，请参考Apress出版的许多专门介绍.NET的书籍。

内容概述 本书一共包括33章。

如果刚开始学习WPF，将会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，以及新的能够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以及它的工作原理，并且将使用不同的编码方法创建一个基本的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。

该章将分析不同的布局策略，并且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开发人员都非常熟悉的控件，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。

在该章将会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。

在该章将会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样式可以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：几何图形和图画 深入分析2D绘图。

在该章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 介绍低级的图形编程。

在该章将使用像素着色器应用Photo风格的效果，手动构建位图，并且为了优化绘图性能使用WPF的可视化层。

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

第16章：高级动画 研究了更高级的动画技术，如关键帧动画、基于路径的动画以及基于帧的动画。

还将分析一个详明的示例，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 介绍如何通过插入一个定制的模板为任意WPF控件提供动态的新外观（以及新行为）。

还将看到如何使用模板构建能够换肤的应用程序。

<<WPF编程宝典>>

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。

还将学习如何使用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中导航，以及应用过滤、分类和分组。

第22章：列表、网格和树 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。

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，这种应用程序可以从Web站点加载。

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。

在该章将会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对象添加动画。

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富文档支持。

在该章将学习如何使用流文档以尽可能容易阅读的方式展现大量文本，并将学习如何使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，通过该打印模型可以在打印文档中绘制文本和图形。

在该章还将学习如何管理页面设置和打印队列。

第30章：与Windows窗体的交互 分析如何在一个应用程序——甚至在一个窗口中，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性的、在后台执行耗时的WPF应用程序。

第32章：插件模型 展示如何创建可扩展的、能够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 为了运行WPF 4应用程序，计算机中必须安装Windows 7、Windows Vista或带有Windows XP SP2。

还需要.NET Framework 4。

为了创建WPF 4应用程序（以及为了能够打开本书中所提供的示例项目），需要安装Visual Studio 2010，Visual Studio 2010中包含了.NET Framework 4。

还有一种选择。

不使用任何版本的Visual Studio，可以使用Expression Blend（一种面向图形的设计工具）来构建和测试WPF应用程序。

总之，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则是理想的工具。

本书假定使用的是Visual Studio。

如果想学习更多有关Expression Blend的内容，可以参阅有关该主题的专门书籍（顺便提一下，为使用WPF 4创建应用程序，需要Expression Blend 4，在撰写本书时它还是一个beta版本）。

代码示例和URL

<<WPF编程宝典>>

内容概要

本书针对了解，NET平台、C#语言以及Visual Studio开发环境的专业开发人员，深入分析了WPF。

在学习《WPF编程宝典:C#(2010版)》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“新增功能”中突出显示的WPF新特性。

本书全面描述了所有WPF的主要特性，从XAML(用于定义WPF用户界面的标记语言)到3D绘图和动画。

《WPF编程宝典:C#(2010版)》很少编写涉及.NET Framework其他特性的代码，如用于查询数据库的ADO.NET类。

《WPF编程宝典:C#(2010版)》中不讨论这些内容。

但如果需要更多有关.NET而非特定于WPF的特性，请参考Apress出版的许多专门介绍.NET的书籍。

<<WPF编程宝典>>

作者简介

麦克唐纳(Matthew MacDonald)，是一位作家、教育家，也是微软最有价值专家。他撰写了十多本有关.NET编程的书籍，包括Pro Silverlight 3 in C#（2009年Apress出版）和Pro ASP.NET 3.5 in C#（2007年Apress出版）以及本书的前一版本。目前他与妻子以及两个女儿居住在多伦多。

<<WPF编程宝典>>

书籍目录

第1章 WPF概述

- 1.1 Windows图形演化
 - 1.1.1 DirectX：新的图形引擎
 - 1.1.2 硬件加速与WPF
- 1.2 WPF：高级API
 - 1.2.1 Windows窗体将继续保留
 - 1.2.2 DirectX也将继续保留
 - 1.2.3 Silverlight
- 1.3 分辨率无关性
 - 1.3.1 WPF单位
 - 1.3.2 系统DPI
 - 1.3.3 位图和矢量图形
- 1.4 WPF体系结构
- 1.5 WPF4
 - 1.5.1 新特性
 - 1.5.2 WPF工具包
 - 1.5.3 VisualStudio2010
- 1.6 小结

第2章 XAML

- 2.1 理解XAML
 - 2.1.1 WPF之前的图形用户界面
 - 2.1.2 XAML变体
 - 2.1.3 XAML编译
- 2.2 XAML基础
 - 2.2.1 XAML名称空间
 - 2.2.2 代码隐藏类
- 2.3 XAML中的属性和事件
 - 2.3.1 简单属性与类型转换器
 - 2.3.2 复杂属性
 - 2.3.3 标记扩展
 - 2.3.4 附加属性
 - 2.3.5 嵌套元素
 - 2.3.6 特殊字符与空白
 - 2.3.7 事件
 - 2.3.8 完整的EightBall示例
- 2.4 使用其他名称空间中的类型
- 2.5 加载和编译XAML
 - 2.5.1 只使用代码
 - 2.5.2 使用代码和未经编译的XAML
 - 2.5.3 使用代码和编译过的XAML
 - 2.5.4 只使用XAML
- 2.6 XAML2009
 - 2.6.1 自动事件连接
 - 2.6.2 引用
 - 2.6.3 内置类型

<<WPF编程宝典>>

2.6.4 高级的对象创建

2.7 小结

第3章 布局

3.1 理解WPF中的布局

3.1.1 WPF布局原则

3.1.2 布局过程

3.1.3 布局容器

3.2 使用StackPanel面板进行简单布局

3.2.1 布局属性

3.2.2 对齐方式

3.2.3 边距

3.2.4 最小尺寸、最大尺寸以及显式地设置尺寸

3.2.5 Border控件

3.3 WrapPanel面板和DockPanel面板

3.3.1 WrapPanel面板

3.3.2 DockPanel面板

.....

第4章 依赖项属性

第5章 路由事件

第6章 控件

第7章 Application类

第8章 元素绑定

第9章 命令

第10章 资源

第11章 样式和行为

第12章 形状、画刷和变换

第13章 几何图形和图画

第14章 效果和可视对象

第15章 动画基础

第16章 高级动画

第17章 控件模板

第18章 自定义元素

第19章 数据绑定

第20章 格式化绑定的数据

第21章 数据视图

第22章 列表、网格和树

第23章 窗口

第24章 页面和导航

第25章 菜单、工具栏和功能区

第26章 声音和视频

第27章 3D绘图

第28章 文档

第29章 打印

第30章 与Windows窗体的交互

第31章 多线程

第32章 插件模型

第33章 ClickOnce部署

章节摘录

版权页：插图：

<<WPF编程宝典>>

编辑推荐

《WPF编程宝典:C#(2010版)》：尊敬的读者：多年来，NET开发人员一直使用Windows窗体工具包创建Windows应用程序。

尽管Windows窗体是一项相当成熟的技术，但存在一个严重的局限性：它依赖于近十年来基本未变的Windows关键组件。

WPF改变了这一切。

WPF仍然包含您熟悉的标准控件，但它使用性能卓越的DirectX引擎“自行”绘制控件的每个细节，可提供一系列令人赏心悦目的新功能。

例如，您不必编写代码即可定制精美的诸如按钮的常用控件，还可旋转、拉伸、缩放和扭曲用户界面元素来制作生动鲜活的动画效果。

您可以用WPF得心应手地完成过去用Windows窗体难以完成甚至几乎无法完成的任务，如播放媒体文件或创建带有纹理的三维图形。

WPF 4是迄今为止最激动人心的WPF版本。

它新增了DataGrid等专业控件，支持多点触控屏幕和新任务栏等Windows 7功能，增强了文本渲染功能，提供了更逼真的动画以及PhotoShop风格的像素着色器。

《WPF编程宝典——C#2010版》全面介绍WPF平台，浓墨重彩地描述wPF 4的新功能，将帮助您设计、构建以及润色WPF作品。

无论您是初出茅庐的WPF新手，还是使用过WPF旧版本的WPF专业人员，《WPF编程宝典:C#(2010版)》都将成为您的良师益友。

祝您通过《WPF编程宝典:C#(2010版)》学到丰富的WPF知识，享受到美妙的编程乐趣！

MattheW MacDonald (Microsoft MVP.MCSD)

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>