Broadview®
www.broadview.com.cn

全面系统，深入实用
通俗易懂，实例丰富

WPF
专业编程指南

◆ 李应保 著

電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

WPF
专业编程指南

李应保 著

電子工業出版社
Publishing House of Electronics Industry
北京•BEIJING

内 容 简 介

《WPF 专业编程指南》是一本 WPF 编程的专业参考书，全书通过大量的实例深入阐述了 WPF 中的传递事件、传递命令、相关属性、附加属性、逻辑树和视觉树等基本概念；介绍了各种画笔、画刷的使用方法；深入讨论了 WPF 中的各种控件以及这些控件在窗口或页面上的排版，并进而阐述了控件的风格和模板及数据绑定等相关技术。

本书对 WPF 中的图形系统及图形和排版的变换原理也进行了深入的探讨，并在此基础上讨论了 WPF 中的动画技术。多媒体不是 WPF 专有的技术，但本书介绍了在 WPF 中使用多媒体的实用技术。用户控件和自定义控件是 WPF 中比较深入的内容，本书最后两章对这一课题进行了深层次的研究，通过对 Ribbon 控件的开发，不仅可以了解开发用户控件和自定义控件的方法，而且可以体会 WPF 项目的组织及多种 WPF 技术细节的综合运用。

本书可供.NET 桌面及互联网应用程序的开发人员、项目管理人员或准备进入这一领域的相关工程技术人员，以及大专院校相关专业的师生参考学习。

图书在版编目（CIP）数据

WPF 专业编程指南 / 李应保著.—北京：电子工业出版社，2010.1
ISBN 978-7-121-10011-6

I. W… II.李… III.窗口软件，Windows Vista—用户界面—程序设计 IV.TP316.7

中国版本图书馆 CIP 数据核字（2009）第 220308 号

策划编辑：袁金敏
责任编辑：高洪霞
特约编辑：顾慧芳
印　　刷：北京天宇星印刷厂
装　　订：涿州市桃园装订有限公司
出版发行：电子工业出版社
　　北京市海淀区万寿路 173 信箱　　邮编 100036
开　　本：860×1092　1/16　　印张：34　　字数：828 千字
印　　次：2010 年 1 月第 1 次印刷
印　　数：4000 册　　定价：68.00 元（含光盘 1 张）

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线：（010）88258888。

前　言

2008年在世界经济历史上是一个不平凡的一年，在这一年中，美国的五大投资银行有两家破产。金融危机席卷全球，美国的失业率在过去的一年从4%飙升到了10%，有些城市的失业率甚至高达20%；加拿大的就业数字也是惨不忍睹，多伦多的失业率剧升到了12%（2009年8月数据）。在这样暗淡的经济背景下，有关WPF的工作却在快速增长，过去两年，和WPF相关的工作职位在北美和欧洲每半年翻一番，WPF初级职位的薪水在5万美元到7万美元之间，高级职位在10万美元以上。一般来说，新的开发平台在中国的应用比欧美要晚2~3年，但随着互联网的普及和软件开发外包到中国，这一迟后时间正在缩短。可以预见，在未来几年内，和WPF相关的工作职位也会在中国快速增长；所以，学习WPF编程技术正当其时，拥有WPF技术必将使你在职场上更加挥洒自如，也就是说，**学习WPF就是获取新的工作机遇。**

WPF是基于.NET的新一代界面开发平台，它实现了桌面应用程序和互联网应用程序的统一编程，实现了程序员长期梦想的数据驱动用户界面，融合了动画、多媒体的功能，跨越了图形和控件、控件和排版等技术上的界限，在很短的时间内实现并超越了Flash和PDF的相关功能。近年来微软在软件开发上的投资额已和中等国家的GDP相当，其中.NET是微软最大的投资项目。在.NET 2.0之后，微软停止了WinForm的开发，而WPF实际上是微软在今后一段时间内唯一要不断投入的用户界面开发平台，这一平台在Vista、Windows 7 和Window Server上得到了广泛的支持。建立在WPF上的应用程序将会自动随着.NET的不断开发而自动拓展新的功能：把WPF和WCF（Windows Communication Foundation）技术相结合，可以很容易地实现面向服务的软件架构（SOA，Service Oriented Architecture）；WPF对图形流的支持，使得开发GIS应用程序更加方便；基于SilverLight和Ajax技术是互联网开发的新热点。所以，企业把应用程序的界面建立在WPF之上，不仅可以极大地缩短开发周期，而且可以把同一技术用在不同的项目上，从而极大地降低开发成本。比如过去开发桌面应用程序和互联网应用程序一直是两个不同的开发团队，而使用WPF，我们只需要一个开发团队。在过去的20年内，微软一直是用户界面开发的领跑者；若企业把应用程序建立在WPF之上，就不会担心落伍。所以，**应用WPF就是降低企业的开发成本**。

本书特点

本书深入浅出地介绍了WPF中的各种新概念，使用了大量图表和实例力图以整体的形式把WPF展现在读者的面前。笔者推崇Scott Meyers的写作风格（Scott Meyers的C++系列丛书在软件界有广泛的影响——笔者注），即以散文的笔调描述技术细节，以避免枯燥的叙述；希望读者在阅读本书时有一种像朋友在一起聚会的感觉，边喝啤酒边聊天，在聚会结束时，您会发现自己已经掌握了WPF技术。因此，**阅读本书是通向WPF专业编程的捷径。**

由于WPF是在.NET 3.0 之后引入的，故读者在使用WPF之前应该已经熟悉 .NET的编程环境、C#语言等基本知识。本书使用简单的UML（Unified Modeling Language）描述WPF类和类间的关系，若您具有UML的基本知识，对阅读本书会有帮助。书中的例子在Visual Studio 2008 和.NET FrameWork

3.5上调试过，有时笔者也使用了微软的Expression Blend 2.0调试，但后者不是必需的。

本书光盘使用说明

本书的配书光盘含有约100MB的源程序，所用的语言为C#和XAML。所有的例程在Visual Studio 2008 和.NET Framework 3.5上调试通过，笔者在创建某些例程时，使用过Microsoft Blend 2.0。Microsoft Blend工具在创建WPF界面时非常有用，但对于运行本书的例程不是必需的。本光盘的内容是对本书的补充，因书中着重介绍WPF编程模型和基本概念，光盘中则含有完整的源代码。

1．光盘中的目录使用Yingbao.Chapterxx格式，如第1章的例程在Yingbao.Chapter1的目录下，第2章的例程在Yingbao.Chapter2的目录下等。全书共18章，整个光盘含有18个目录。

2．在每个目录下有一个相应的Visual Studio解决方案文件，其文件名采用Yingbao.Chapterxx.sln格式，如第1章Visual Studio解决方案文件名为Yingbao.Chapter1.sln，第2章Visual Studio解决方案文件名为Yingbao.Chapter2.sln等。该文件中含有一个或多个项目，如Yingbao.Chapter13解决方案中含有六个项目。运行某个项目，您需要在Visual Studio中的Solution explore窗口下单击鼠标右键，在弹出菜单中选择“Set as StartUp Project”条目，然后，您只要按下“F5”功能键，或在Visual Studio 的“Debug”菜单下选择“Start Debug”即可。

3．例程中命名空间的名字采用Yingbao.Chapterxx的格式，如第1章所有例程中的类都在Yingbao.Chapter1命名空间中，第2章的所有例程中的类都在Yingbao.Chapter2命名空间中。虽然对于本书的例程来说使用不同的命名空间不是必需的，但对于组织大型项目，使用命名空间是良好的习惯。

4．例程中的类名、属性名、域名、方法名采用通用电气（GE）编程规范，读者也可用其他的公司或自己公司的编程规范。

感谢

写作是一个费时费力的工作，笔者在写作本书的时候，得到了家人的支持和理解。电子工业出版社的杨福平副总编和袁金敏编辑对本书的出版做了大量的工作，笔者在此表示衷心感谢。

联系方式

最后，若您有什么建议和意见或者发现书中的错误，请和笔者联系：Yingbao.Li@gmail.com。

李应保

2009年9月8日于加拿大

目　录

第一篇　WPF编程基础

第二篇　WPF专业程序员必备

第三篇 图形和动画

第四篇 开发WPF产品

第一篇　WPF编程基础

第1章 WPF应用程序

本章讨论WPF应用程序的创建，运行及退出时的技术细节。涉及创建WPF应用程序的三个重要的类：Application、Window和Page、应用程序的异常处理、发布等话题。

1.1 WPF应用程序的创建

在Windows操作系统中，所有的应用程序都在自己独立的进程中运行，WPF也不例外。每个进程都有自己独立的内存地址，进程间的数据是隔离的，进程之间不会相互影响。当应用程序创建进程时，同时创建一个或多个线程。线程在自己的进程空间中运行，在底层，WPF仍然使用Windows的消息驱动机制来实现事务处理。

创建WPF应用程序从创建Application类开始。WPF的主要功能是人机交互，我们可以把当代人机界面（UI）程序归为两大类：一类是桌面（desktop）应用程序；一类是互联网（Web）应用程序。WPF首次实现了对这两类应用程序的统一编程。

让我们来创建一个简单的WPF程序：

```
using System;
using System.Windows;
namespace Yingbao.Chapter1
{
     public class HelloWPF
    {
        [STAThread]
        public static void Main()
        {
            Window win = new Window();
            win.Title = "WPF application";
            win.Content = "Hello WPF!";
            win.Show();
            Application app = new Application();
            app.Run();
        }
    }
}
```

首先，用Yingbao.Chapter1声明命名空间，并将在本书的所有例程中使用这一惯例。第1章的例程在Yingbao.Chapter1的命名空间中，第2章的例程将在Yingbao. Chapter2的命名空间中等。

然后，引入了.NET中的两个命名空间: System和System.Windows。 System命名空间提供.NET的基本服务，System.Windows中含有WPF的基本服务。Microsoft把WPF所用的类，都放在System.Windows的命名空间或其下面的命名空间中，如System.Windows.Controls、System.Windows.Input，等等。在Visual Studio的项目下，引入相应的Assembly, 这样就可以使用这些命名空间中的类了。

Main函数是WPF程序的入口点，它必须是静态（static）的。所有WPF程序，都必须在单一线程公寓(STA)模型中运行。STA来源于COM，若对此概念不熟悉，可以参考COM的相关著作或文章。在