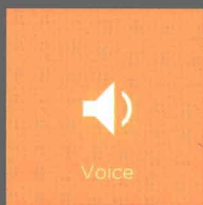
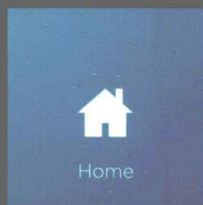


Windows Phone 应用程序开发

李春旭 刘泽宏 编著



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

Windows Phone应用程序开发

李春旭 刘泽宏 编著

电子工业出版社

Public Information and Communications Industry

北京 - BEIJING

推荐序一

近年来，随着全球移动网络及智能手机的迅速发展，智能手机供应商之间的竞争不仅是硬件设计的竞争，智能手机开发平台以及智能手机应用生态系统更是必争之地。2011年2月，全球最大的手机制造商诺基亚与全球最大的软件公司微软签订了战略合作协议，这一强强联手，充分利用了诺基亚在智能手机设计及微软在桌面软件开发平台上各自的优势，意在激烈的市场竞争中取胜。

2012年4月，第一版 Window Phone (Tango) 正式在中国上市，立即吸引了广大智能手机应用开发爱好者，大家纷纷尝试凭借网上一些零散资料来学习开发并发布自己的应用。春旭和泽宏是 Window Phone 开发平台的忠实拥趸者，在第一时间涉足这一开发平台，开发了若干 Window Phone 应用，并在开发过程中积累了很多第一手经验，本书就是在这样的背景下编写出来的。春旭和泽宏自身作为开发者，从应用开发者切身需要出发，为有意加入 Windows Phone 生态系统应用程序开发工作的爱好者提供了快速的指南。本书面向从事 Windows Phone 应用设计、开发和测试的个人或团队，涵盖了从入门到进阶的应用开发教程，还涉猎了 Windows Phone Mango 和 Tango 编程基础及延伸的相关内容。从开发环境的部署、相关工具的使用，介绍至最后的应用程序的注册与发布，让开发者能够清晰地了解 Windows Phone 生态系统中应用的生命周期和详细开发流程。

除基本的开发内容讲解外，本书还别具特色地介绍了一些对开发者在实际开发中颇有裨益 Windows Phone 的扩展知识，包括 Silverlight Toolkit 工具包中部分控件的使用，Windows Azure 云服务与 Windows Phone 的结合，开放平台在 Windows Phone 上的使用，Windows Phone 上的开源工具等内容。

《Windows Phone 应用程序开发》是一本内容生动、组织流畅、实用易行的开发指南。为了方便读者理解，每一章都配有大量的图例和代码实例，读者可以轻松、便捷地掌握 Windows Phone 的编程技巧。

祝大家阅读快乐，并享受你的开发旅程！

微软高级软件测试主管 朱素艳

推荐序二

五年来，iPhone 从第一代发展到现在的 iPhone 4S，彻底颠覆了人们对手机的印象，并创造了一个以移动 app 为载体的移动互联网时代，Android 平台紧跟 iPhone，以其免费、开放的商业模式迅速成长起来。而当年几乎是唯一的智能手机操作系统——Windows Mobile 却在这几年中丢城失地，似乎微软将与这次移动互联网的浪潮失之交臂。然而，无论是市场还是用户都不会再给微软时间去分析失利的原因并逐一解决，重塑 Windows Mobile 的形象。微软有世界上顶尖的决策层，自然也会意识到这一点，所以微软 2011 年推出了全新的 Windows Phone 7 手机操作系统，除内核有一定延续性外，Windows Phone 与 Windows Mobile 完全不同，不同的市场定位、不同的用户界面、不同的操作体验、不同的运营模式、不兼容的应用格式，等等，Windows Mobile 也将被微软舍弃。可以说微软是以壮士断腕的魄力，快刀斩乱麻的精神重返移动互联网市场，Windows Phone 必将带来微软在移动互联网时代的新篇章。

虽然 Windows Phone 此时进入市场面临时间上的劣势，但是 iPhone 和 Android 由于其产品基因的限制，决定了二者都有一定的劣势，苹果公司作为一个消费类电子产品公司，一直都是通过销售硬件来获得利润，封闭的产品保证了充盈的利润率，而正是这样也限制了产品种类的规模，导致现在市面上只有一款 iPhone 手机，也正是由于这个原因，虽然苹果公司率先提出个人电脑（PC）的概念，但却在个人电脑市场上输给了坚持开放原则的 IBM 电脑（IBM-PC）。而 Android 系统坚持以最开放的姿态推广其手机系统，开放源代码可以让任何个人或组织基于 Android 做二次开发，硬件上也没有任何限制，这样做导致的后果是 Android 系统版本众多，用户体验千差万别，硬件配置众多，手机性能良莠不齐，应用开发难度增大，用户使用成本大大提高。此外，现在 iPhone 和 Android 手机同质化严重，用户界面、图标更加相似，必会给用户带来审美疲劳，而此时 Windows Phone 以一个简洁、清新的用户界面出现，吸取了 iPhone 和 Android 开发模式中的长处，走出了一条既易于开发又保证用户体验的合作共赢之路，同时也避免了 iPhone 和 Android 的劣势，在可预见的将来，Windows Phone 必将与 iPhone 和 Android 三分天下，共同促进移动互联网市场的繁荣。

很高兴有机会在本书出版前先于读者通读此书，本书非常适合作为 Windows Phone 入门和进阶的参考图书，全书结构清晰，按照开发者学习的思路由浅入深地组织书籍内容，内容充实，思路清晰，行文连贯。可见两位作者作为有着丰富开发经验的开发者，能够很好地引导读者学习 Windows Phone 开发。更加难能可贵的是，本书

还结合国内实际情况，重点介绍了几个 SNS SDK 的使用，这些 SDK 都是移动应用开发者最可能使用到的。

同时也希望读者能够通过此书，全面学习 Windows Phone 开发，能够把握 Windows Phone 平台迅速成长的机会，与 Windows Phone 共成长，在移动互联网的大潮中赢得属于自己的一席之地。

微软资深开发经理 徐泽勇



Windows Phone 7 是微软推出的全新手机操作系统，其前身是微软的 Windows Mobile 操作系统，但 Windows Phone 7 并不是 Windows Mobile 6.5 的升级版本，而更像是一款全新的手机操作系统。Windows Phone 7 拥有全新的 UI 设计、功能设计和操作体验。Windows Phone 7 采用 Metro UI 设计，简洁朴素的 UI 效果与鲜艳明快的配色风格给人耳目一新的感觉。在用户体验方面，Windows Phone 7 严格控制手机生产厂家的硬件配置，从而保证了不同手机生产商的每部 Windows Phone 7 手机都拥有媲美 iPhone 的流畅操作体验。

Windows Phone 7 是微软开拓移动互联网版图的一个重要产品，必将投入大量的人力、物力进行产品的完善和市场推广，现在已与诺基亚达成战略合作的协议，诺基亚及其他手机生产厂商已推出了数十款 Windows Phone 7 手机，在海外上市并获得广泛的好评。国外多家著名的咨询公司一致认为 Windows Phone 7 手机由于其平台体验的一致性和合作模式的开放性必将与 Android 手机、iPhone 三足鼎立。

随着移动互联网的蓬勃发展，国内 Android 和 iOS 开发者均数量巨大，并且仍存在开发人员紧缺的情况，根据前面的分析，对未来 Windows Phone 7 的需求也不会少于这两个平台，而国内现在 Windows phone 7 的开发人员短缺，主要是因为 Windows Phone 7 手机在国内上市不久，并且国内相关的开发书籍较少。

本书旨在为开发者提供全面、详细、易于学习和理解的 Windows Phone 7 应用开发理论和经验介绍，丰富 Windows Phone 7 开发技术类书籍品种。

本书以 Windows Phone 7 Mango/Tango SDK 为学习对象，针对 Windows Phone 7 开发所需要的关键技术进行深入浅出的讲解，覆盖了 Windows Phone 7 应用开发所有可能遇到的技术难点。内容不仅涉及开发环境部署、应用程序生命周期、基本控件、页面导航、数据绑定、网络访问、数据存储、设备访问、消息推送等基本功能，还包括 SilverLight Toolkit 扩展控件、Metro UI 设计、动画设计、多任务、云存储、常用第三方开发 SDK 介绍、应用发布等开发中必不可少的内容。循序渐进的知识点讲解、由浅入深的原理剖析及结合作者开发中积累的经验介绍等特点，使这本书适用于希望学习或者从事 Windows Phone 7 应用开发的入门和进阶读者。

本书的读者应有一定的面向对象编程基础，有从事软件开发的代码编写经验，对 Windows Phone 7 应用开发有兴趣。若有一定的 .NET 或者其他手机平台应用开发经验将更有助于对本书的理解。Windows Phone 7 支持 C# 和 Visual Basic.NET 两种语言开发，但是从开发者数量、语言学习的难易程度、未来发展前景等方面考虑，C# 都优于 VB.NET，所以本书以 C# 语言作为主要开发语言。

目 录

第 1 章 Windows Phone 7 简介及其开发环境部署	1
1.1 Windows Phone 7 简介.....	1
1.1.1 Windows Phone 7 背景、规范和标准.....	1
1.1.2 Windows Phone 7 与 Windows Mobile, iOS, Android 的对比.....	4
1.1.3 Windows Phone 7 新特性.....	5
1.2 开发环境部署.....	7
1.2.1 Windows Phone SDK 的安装与介绍.....	7
1.2.2 Zune 的安装与使用.....	9
1.2.3 Expression Blend for Windows Phone 介绍.....	10
1.2.4 Windows Phone Tango 应用开发.....	11
第 2 章 Windows Phone 7 应用程序初探	12
2.1 第一个 Windows Phone 7 应用.....	12
2.1.1 项目创建和代码编写.....	12
2.1.2 项目文件说明.....	18
2.1.3 运行和部署.....	18
2.1.4 调试.....	19
2.2 模拟器使用介绍.....	19
2.2.1 功能简介和限制说明.....	19
2.2.2 性能分析器.....	19
2.3 应用的生命周期.....	20
第 3 章 Windows Phone 的基本控件	23
3.1 输入输出控件.....	23
3.1.1 TextBlock.....	23
3.1.2 TextBox.....	25
3.1.3 PasswordBox.....	29
3.1.4 ProgressBar.....	30
3.1.5 键盘布局 (InputScope).....	32
3.2 按钮控件.....	34
3.2.1 Button.....	34

3.2.2	HyperlinkButton	37
3.2.3	RadioButton	38
3.2.4	CheckBox	40
3.2.5	Slider	43
3.3	滚动控件	47
3.3.1	ScrollViewer	47
3.3.2	ListBox	49
3.3.3	滚动控件的性能优化	55
3.4	多媒体控件	58
3.4.1	Image	58
3.4.2	InkPresenter	60
3.4.3	MediaElement	61
3.5	其他控件	62
3.5.1	WebBrowser	62
3.5.2	BingMap	64
3.6	Windows Phone 自定义控件和用户控件	64
3.6.1	用户控件	64
3.6.2	自定义控件	69
第 4 章	Windows Phone 7 应用程序主题、布局与导航	71
4.1	样式和主题	71
4.1.1	画刷	71
4.1.2	主题和静态资源	75
4.2	页面 UI 布局	75
4.2.1	UI 界面简介	75
4.3	页面内布局方式	79
4.3.1	StackPanel 布局	79
4.3.2	Grid 布局	80
4.3.3	Canvas 布局	82
4.4	页面导航	83
4.4.1	Pivot 介绍	83
4.4.2	Panorama 介绍	84
4.4.3	页面间导航	86
4.5	页面方向	87

第 5 章 数据绑定及数据访问	89
5.1 数据集合	89
5.1.1 集合类型介绍及用途	89
5.1.2 集合访问 (Linq)	97
5.2 数据绑定	101
5.2.1 数据绑定原理及好处	101
5.2.2 数据绑定模式介绍	105
5.2.3 数据绑定中的数据转换	106
5.2.4 数据绑定 API 及事件介绍	108
5.3 MVVM 模型编程	108
5.3.1 MVVM 介绍	108
5.3.2 命令绑定和 ICommand	110
5.3.3 MVVM 编程实例	111
第 6 章 Windows Phone 7 Toolkit 中的扩展控件	118
6.1 Silverlight Toolkit	118
6.2 输入与按钮控件	119
6.2.1 AutoCompleteBox	119
6.2.2 ContextMenu	127
6.2.3 DateTimeConverter	130
6.2.4 ToggleSwitch	133
6.3 滑动与显示	134
第 7 章 Metro UI 设计实现	143
7.1 Metro 设计简介	143
7.1.1 包豪斯风格起源	143
7.1.2 Windows Phone 中 Metro 的诞生	147
7.1.3 现代程序中的应用	149
7.2 Metro UI 设计原则	153
7.2.1 一般原则	153
7.2.2 字体	154
7.2.3 瓷片 (Tiles)	154
7.2.4 图片和照片	155
7.2.5 图标	156

7.2.6	布局	157
7.2.7	Panorama 和 Pivot 设计精要	159
第 8 章	Windows Phone 7 动画实践	164
8.1	变形	164
8.1.1	RotateTransform (旋转变形)	164
8.1.2	ScaleTransform (缩放变形)	165
8.1.3	SkewTransform (倾斜变形)	167
8.1.4	TranslateTransform (移动变形)	167
8.1.5	TransformGroup (变形组)	168
8.1.6	MatrixTransform (矩阵变形)	169
8.2	矢量图	170
8.2.1	直线 (Line) 和折线 (Polyline)	172
8.2.2	多边形 (Polygon)	177
8.2.3	椭圆 (Ellipse) 和矩形 (Rectangle)	177
8.2.4	Z-Index	179
8.2.5	拉伸效果	180
8.2.6	Path 绘图	180
8.2.7	路径标记语法	184
8.3	位图	185
8.3.1	操作像素	185
8.3.2	位图与矢量图相结合	187
8.3.3	位图的裁剪和遮罩	188
8.4	动画制作	190
8.4.1	动画制作简介	190
8.4.2	Storyboard 详解	191
8.4.3	关键帧	192
8.4.4	Easing Functions	194
第 9 章	设备访问	195
9.1	设备信息访问	195
9.1.1	设备信息	195
9.1.2	用户信息	196
9.1.3	网络信息	197
9.2	系统任务调用	197
9.2.1	邮件任务 (EmailComposeTask)	198

9.2.2	通话任务 (PhoneCallTask)	198
9.2.3	短消息任务 (SmsComposeTask)	198
9.2.4	浏览器任务 (WebBrowserTask)	199
9.2.5	图像选择任务 (PhotoChooserTask)	199
9.2.6	媒体播放器任务 (MediaPlayerLauncher)	200
9.2.7	商城相关任务 (MarketplaceSearchTask)	200
第 10 章	互联网访问	202
10.1	请求数据	202
10.2	发送数据	204
10.3	实例: 创建 RSS 阅读器	208
第 11 章	Windows Azure 云服务	216
11.1	简介	216
11.1.1	Windows Azure 的托管服务与 Windows Phone	219
11.1.2	Windows Azure 存储服务	220
11.1.3	SQL Azure	220
11.1.4	Windows Azure Marketplace DataMarket	221
11.2	Windows Azure Toolkit for Windows Phone	222
第 12 章	本地存储	224
12.1	独立存储 (Isolated Storage)	224
12.1.1	独立存储的概述与类型	224
12.1.2	独立存储设置 (IsolatedStorageSettings)	225
12.1.3	独立存储文件 (IsolatedStorageFile)	229
12.2	本地数据库	236
12.2.1	概述	236
12.2.2	部署本地数据库	238
12.2.3	定义数据上下文	238
12.2.4	在 Windows Phone 应用中创建数据库	242
12.2.5	使用数据库	243
12.2.6	数据库安全性	245
第 13 章	Windows Phone 的消息推送	246
13.1	概述	246
13.2	推送通知的类型	248

13.2.1	Toast 通知	248
13.2.2	Tile 通知	249
13.2.3	Raw 通知	250
13.3	使用 Toast 通知	250
13.4	使用 Tile 通知	261
13.5	Raw 通知的使用	270
第 14 章	Windows Phone 的多任务处理	271
14.1	音频后台代理	272
14.2	PeriodTask 后台代理	281
14.2.1	概述	281
14.2.2	实例——创建可使用后台计划的应用程序	284
14.3	后台文件传输	295
14.3.1	BackgroundFileTransfer 概述	295
14.3.2	BackgroundFileTransfer 实例	297
第 15 章	开放平台在 Windows Phone 上的应用	315
15.1	开放平台简介	315
15.2	在 Windows Phone 上的新浪微博开放平台	316
15.2.1	授权模式	316
15.2.2	SDK 的使用说明	317
15.2.3	API 接口介绍	317
15.2.4	使用实例	318
15.3	在 Windows Phone 上的人人网开放平台	321
15.3.1	平台介绍	321
15.3.2	API 接口介绍	321
15.4	在 Windows Phone 上的腾讯微博开放平台	324
15.4.1	平台介绍	324
15.4.2	OAuth 授权	324
15.4.3	API 接口	325
第 16 章	Windows Phone 开源代码工具	336
16.1	GB2312 编 / 解码工具	336
16.2	图像编/解码工具	337
16.2.1	图像格式	337
16.2.2	将图像保存至文件	338

16.2.3 过滤图像.....	339
16.2.4 加载 GIF 动画.....	340
16.3 排序列表工具.....	340
第 17 章 Windows Phone 应用发布.....	342
17.1 注册开发者.....	343
17.2 Windows Phone 广告平台.....	344
17.3 Windows Phone 的应用程序认证要求.....	344
17.3.1 基本要求.....	344
17.3.2 认证过程.....	345
17.4 结语.....	349

第 1 章 Windows Phone 7 简介及其开发环境部署

1.1 Windows Phone 7 简介

1.1.1 Windows Phone 7 背景、规范和标准

Windows Phone 7 是由微软发布的一款智能手机操作系统。微软在发行这个操作系统时，主要的销售对象定位于一般消费者市场，而非以前 Windows Mobile 系统所瞄准的企业市场。在 Windows Phone 7 中，微软将其人机交互界面（User Interface）套用了一种称为“Metro”的设计语言（此设计理念最初被用于 Zune 上），并将微软旗下的其他软件集成到这个操作系统中，以严格控制运行它的硬件规范。Windows Phone 7 的上市，宣告了饱受诟病的 Windows Mobile 系列彻底退出了手机操作系统市场。全新的 Windows Phone 7 完全放弃了 Windows Mobile 5 和 Windows Mobile 6x 的操作界面，而且程序互不兼容，只是沿用了过去的版本号，Windows Mobile 6.5 之后就是 Windows Phone 7，但是它们其实是两个完全不同的系统。

2010 年 2 月 15 日，微软首席执行官史蒂夫·鲍尔默在于巴塞罗那举办的 2010 年全球移动通信大会（MWC2010）上说：“我荣幸地向各位介绍 Windows Phone 系列手机操作系统。在市场充斥着相同功能及体验的产品境况下，我激励团队要拿出一个可以带来全新移动体验的解决方案”。

鲍尔默说到：“手机的发展要求产品真正契合用户生活脚步、满足社交需求并提供各种无缝体验。Windows Phone 7 系列手机操作系统的发布，是这一历程的转折点。”

微软 Windows Phone 7 的界面外观比大多数手机系统要简洁。黑色背景下的亮色图标显得十分醒目。开机屏幕拥有四个大尺寸的方形图标，分别代表手机、联系人、电子邮件和短信功能。微软将这种设置称之为“Live Tiles（活动瓷片）”，此界面下方是进入相册与 Xbox Live 游戏的入口。微软在新系统中再次强调了手机社交功能的重要性，用户可以通过手机上的联系人名单、相册直接对 Facebook 或 Windows Live 上的个人社交主页进行更新，同时也可以查看好友在社交网站上的页面或照片。

Windows Phone 7 基于 Windows Embedded CE 7.0 内核，整个操作系统的架构如图 1-1 所示，由硬件层（Hardware）、内核层（Kernel）、系统层（System）和应用层（Applications）四部分组成。

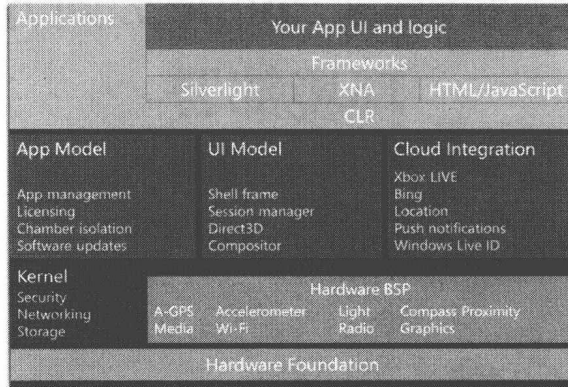


图 1-1 Windows Phone 操作系统架构

1. CLR 虚拟机

与 Android 使用 Java 虚拟机来运行应用程序的结构类似，Windows Phone 7 使用 .NET CLR 虚拟机来运行应用程序。

2. 软件结构体系

Windows Phone 7 中的软件层，即软件结构体系，主要由两部分组成，如图 1-2 所示，分别是 Screen 部分和 Cloud 部分。Screen 部分由“Tools”和“Runtime”组成，Cloud 部分则由“Portal Services”和“Cloud Services”组成。Screen 部分可以理解为本地，Cloud 部分可以理解为云端，它们都是 Windows Phone 7 软件体系中不可缺少的部分。

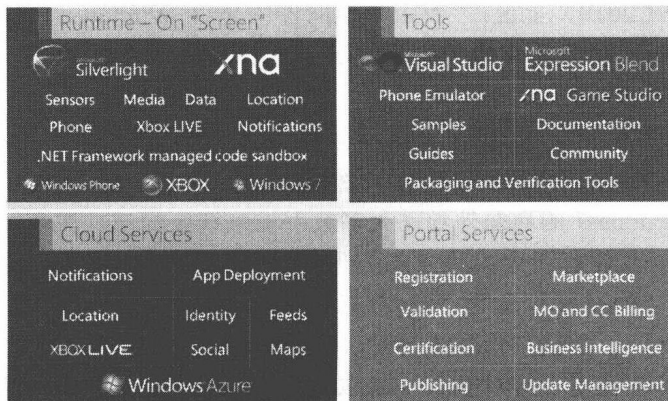


图 1-2 Windows Phone 的体系结构

Screen 部分中的“Tools”是开发应用所必需的开发工具和技术支持，“Runtime”则是开发应用的 Framework，提供所需要的 API 和功能。Windows Phone 7 提供了两种 Framework，分别是 Silverlight Framework 与 XNA Framework。Silverlight Framework 是以 XAML 文件为基础的应用程序设计框架，用来开发基本应用、网络应用、多媒体应用和控件。XNA Framework 则是用来开发基础的游戏设计框架的，如开发 2D 游戏、3D 游戏和游戏控件。

Silverlight Framework 和 XNA Framework 都是在 .NET 平台上的应用程序开发架构，能够有效地协助开发人员开发应用程序，而且开发人员只要稍微修改现有的 Silverlight 应用程序或 XNA 应用程序，就可以将 Silverlight 应用程序或 XNA 应用程序移植到 Windows Phone 7 上执行。

Cloud 部分中的“Portal Services”包含开发者开发应用程序所必需的注册账号、认证、发布、更新管理以及 Marketplace 的付费管理。Windows Phone 7 开发必须拥有注册的账号才能进行真机测试，否则只能在模拟器上进行测试。“Cloud Services”则是云端服务所需要的服务器 API。

3. 编程语言

微软当前只允许开发者使用 C# 和 VB 来进行 Windows Phone 7 的应用开发。此外，微软为了规范 Windows Phone 7 的用户体验，对开发者开发应用进行了严格的约束，开发者必须严格遵循这些开发约束和条款来进行应用开发。

4. Windows Phone 7 的上市日期

2010 年 10 月 11 日发布 Windows Phone 7 操作系统，随后多家厂商发布数款搭载 Windows Phone 7 系统的手机。

2011 年 5 月 4 日，发布 Windows Phone 7 Mango（芒果）版，支持多任务等 500 多项新功能，最重要的是带来了对于东亚地区以及其他地区更多语言的支持，而这其中就包含对中文、日语等的支持。它是 Windows Phone 7 里程碑式的一个版本。

2012 年 3 月 23 日，微软发布 Windows Phone 7 Tango 版。

目前 Windows Phone 7 已在中國大陸地区等多个国家和地区上市。

现在上市的 Windows Phone 7 手机已有数十款，包括诺基亚、三星、LG、HTC、中兴、Dell 等多个著名手机生产厂家生产的手机，尤其以诺基亚发布的 Lumia 800、Lumia 710、Lumia 900 最受大家喜爱。这些手机大致可以分为两类，一类是在 Mango 版本之前发布的手机，统称为一代机；一类是 Mango 版本之后发布的手机，称为二代机。二代机的硬件性能相比一代机更胜一筹，此外，二代机都统一预装了 Windows Phone 7 Mango 版本，而一代机通过升级也可以更新到最新的版本。

国内市面上已经有大量的 Windows Phone 7 水货手机，以其低廉的价格和全新的用户体

吸引了一部分喜欢尝鲜的用户，2012 年国内行货的上市也刺激了 Windows Phone 7 手机的消费，国内一些著名软件公司也开发出了数量众多的 Windows Phone 7 应用，此书截稿时，已经有了腾讯的全线产品的应用，人人、豆瓣等 SNS 应用，新浪微博，优酷、土豆、PPTV 等视频应用等。

1.1.2 Windows Phone 7 与 Windows Mobile, iOS, Android 的对比

Windows Phone 7 无论从用户体验还是应用程序开发架构的角度来看都是与 Windows Mobile 完全不同的产品，虽然内核采用的是 Windows Mobile 7.0 的内核，但是 Windows Phone 7 不兼容任何 Windows Mobile 的原有应用，被视为微软为抗衡 Andorid 和 iOS 两大手机操作系统创作出的壮士断腕的产品，宁可舍弃所有 Windows Mobile 的应用也要保证 Windows Phone 7 的用户体验和产品质量。本书前面也提到 Windows Phone 7 是一款面向最终用户的智能手机操作系统，相对封闭，不具有 Windows Mobile 所具有的定制性和可扩展性，也就是说用户拿到的 Windows Phone 7 手机将是同一个样子，这点与 iPhone 手机很相似，不会像 Android 那样，每个手机厂商都可以对 UI 进行修改和定制。除 OEM 厂商和运营商之外，Windows Phone 7 也暂不支持开发者使用 C/C++ 编写应用程序或者动态链接库。

Windows Phone 7 对硬件和屏幕分辨率有明确的要求，现在只支持 800×480 像素的屏幕分辨率。手机所使用的硬件也是对 Windows Phone 7 进行优化和适配过的，所以保证了系统运行的流畅性，这点与 iPhone 手机相似，iPhone 手机从软件到硬件都是由苹果公司自己采购或生产的，操作流畅性通过了充分的优化。而对于 Android 而言，不同手机厂家采购不同的硬件设备，只能通过厂家自己去优化，甚至有些硬件的驱动都需要手机厂家自己去完成，优化的效果完全取决于手机生产厂家的技术水平，这也造成了市面上的 Android 手机良莠不齐的现象。不过多个不同厂家生产拥有的好处是产品的多样性和产品生产量的保证，相比 iPhone 只有一个厂家，每年生产一款手机，必然将带来用户审美疲劳以及产品同质化严重的后果，所以 Windows Phone 7 采取的是既允许多个厂家生产又严格控制硬件的策略。

从开发者的角度来看，Windows Phone 7 要求屏幕分辨率是 800×480 像素，并且要求支持最少四点触摸，减少了开发者的适配工作量。而 Android 手机的一个弊端是任何听说过或没听说过的分辨率都有，从 320×240 像素到 1280×720 像素，有不少于 10 种的屏幕分辨率，而开发者开发的应用程序为保证所有手机的用户都能使用，就要对所有分辨率进行适配，一旦适配得不充分，将出现部分机型布局错误的情况，而 Windows Phone 7 只有一种分辨率，无适配成本。

当然，在 2010 年 Windows Phone 7 刚刚推出时，800×480 像素的分辨率是主流偏高端的，随着显示技术的进步，越来越多的高端手机采用了 HD 分辨率的屏幕，而 Windows Phone 的屏幕分辨率逐渐在高端市场失去优势。支持更高的分辨率这一重任就要等 Windows Phone 8 来完成了。笔者预计高端 Windows Phone 机型将采用约 1200×720 像素的分辨率，