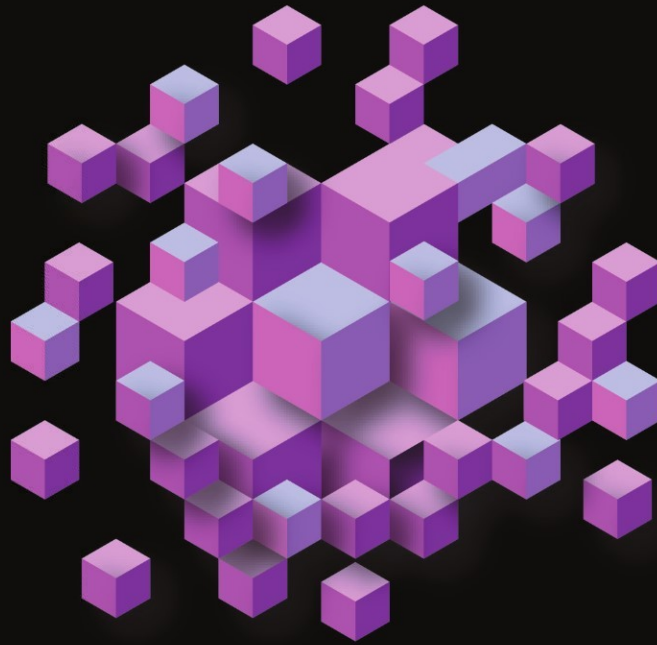


Master the art of creating authentic mobile applications on Microsoft's newest platform



Pro
**Windows Phone App
Development**

SECOND EDITION

Rob Cameron

Apress®

Pro Windows Phone App Development



Rob Cameron

Apress®

Pro Windows Phone App Development

Copyright © 2011 by Rob Cameron

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed. Exempted from this legal reservation are brief excerpts in connection with reviews or scholarly analysis or material supplied specifically for the purpose of being entered and executed on a computer system, for exclusive use by the purchaser of the work. Duplication of this publication or parts thereof is permitted only under the provisions of the Copyright Law of the Publisher's location, in its current version, and permission for use must always be obtained from Springer. Permissions for use may be obtained through RightsLink at the Copyright Clearance Center. Violations are liable to prosecution under the respective Copyright Law.

ISBN-13 (pbk): 978-1-4302-3936-9

ISBN-13 (electronic): 978-1-4302-3937-6

Trademarked names, logos, and images may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, logo, or image we use the names, logos, and images only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

President and Publisher: Paul Manning

Lead Editor: Jonathan Hassell

Development Editor: Matthew Moodie

Technical Reviewer: Robert Garrett

Editorial Board: Steve Anglin, Mark Beckner, Ewan Buckingham, Gary Cornell, Morgan Ertel, Jonathan Gennick,

Jonathan Hassell, Robert Hutchinson, Michelle Lowman, James Markham, Matthew Moodie, Jeff Olson,

Jeffrey Pepper, Douglas Pundick, Ben Renow-Clarke, Dominic Shakeshaft, Gwenan Spearing, Matt Wade,

Tom Welsh

Coordinating Editors: Debra Kelly and Jennifer L. Blackwell

Copy Editor: Jim Compton

Compositor: Apress Production (Christine Ricketts)

Indexer: SPI Global

Artist: SPI Global

Cover Designer: Anna Ishchenko

Distributed to the book trade worldwide by Springer Science+Business Media New York, 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail orders-ny@springer-sbm.com, or visit www.springeronline.com.

For information on translations, please e-mail rights@apress.com, or visit www.apress.com.

Apress and friends of ED books may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Special Bulk Sales–eBook Licensing web page at www.apress.com/bulk-sales.

Any source code or other supplementary materials referenced by the author in this text is available to readers at www.apress.com. For detailed information about how to locate your book's source code, go to www.apress.com/source-code/.

*To my beautiful wife, Ally, whose love and patience is unending,
and to my lovely daughters Amanda and Anna, who make me so proud every day.*

Rob

Contents at a Glance

Contents at a Glance	iv
Contents	v
About the Author	xviii
About the Technical Reviewer	xix
Acknowledgments	xx
Introduction.....	xxi
Chapter 1: Introduction to Windows Phone and Marketplace.....	1
Chapter 2: User Interface Development	53
Chapter 3: Handling Input on Windows Phone.....	153
Chapter 4: Windows Phone Data Persistence and Networking Programming Model	233
Chapter 5: Integrating with the Windows Phone Platform	305
Chapter 6: Windows Phone Advanced UI Development	349
Chapter 7: Advanced MVVM, Services and App Connect	415
Chapter 8: XNA Framework for Games and Applications.....	481
Chapter 9: Advanced XNA Framework Development.....	539
Chapter 10: Building Top-Rated Applications for Windows Phone	583
Index	481

Contents

Contents at a Glance.....	iv
Contents	v
About the Author	xviii
About the Technical Reviewer	xix
Acknowledgments	xx
Introduction	xxi
Who This Book Is For	xxi
How This Book Is Structured	xxii
Prerequisites	xxiii
Downloading the Code.....	xxiii
Contacting the Author.....	xxiii
Chapter 1: Introduction to Windows Phone and Marketplace	1
Why Should I Care?.....	2
Smartphone Growth and the Nokia Partnership	2
The Mobile Market.....	3
Current Microsoft Developers.....	3
Non-Microsoft Developers	4
Windows Phone Platform Overview	4

Phone Design Philosophy.....	4
Hardware Capabilities.....	7
Software Capabilities.....	9
Windows Phone 7.5	11
Consumer Features.....	11
Developer Features.....	11
Windows Phone Developer Tools.....	12
Silverlight Tools and Framework Overview.....	12
Silverlight for Windows Phone.....	12
Hello for Windows Phone.....	13
XNA Game Studio and Framework Overview.....	21
Hello XNA for Windows Phone	22
Implement the Game	28
The Windows Phone Marketplace	40
App Hub Registration.....	40
Submitting Applications to Marketplace.....	41
Windows Phone Marketplace Test Kit	41
Beta Testing Your Application.....	41
Targeted Application Distribution	41
Windows Phone Web Marketplace	44
Windows Phone Marketplace on the PC	45
Marketplace on the Device	46
Limiting Distribution of Applications.....	50
Conclusion	52
Chapter 2: User Interface Development.....	53

UX Performance Improvements.....	54
Designing for Windows Phone.....	54
Design Approach.....	54
Design Resources	55
Designer and Developer Mechanics	59
User Interface Development in Visual Studio 2010	66
WMAppManifest.xml File	69
Anatomy of a Windows Phone Application	70
Anatomy of a PhoneApplicationPage	78
Page Navigation.....	81
.NET Framework for Silverlight	84
XAML Overview	86
Key Silverlight UI Base Classes.....	92
User Interface Development with Expression Blend 4.....	97
Layout System	97
Layout Controls Sample and Blend Behaviors	99
Data Binding	103
Sample Data	118
Control Styling	120
Control Templates.....	126
Animation Fundamentals.....	130
Creating an Animation Storyboard.....	130
Creating Animations in Blend	133
Creating Animations in Code	138
Application Layout.....	138

Panorama.....	138
Pivot.....	144
Windows Phone 7.5 User Experience Enhancements	146
New Controls	146
New Controls Demonstration	147
Implicit Styles	151
Conclusion	152
Chapter 3: Handling Input on Windows Phone	153
The Keyboard.....	154
Physical Keyboard	154
Soft Input Panel (SIP) Keyboard	154
Programming with the Keyboard	154
Touch Input.....	160
Single-Point Touch.....	161
Raw Touch with Mouse Events.....	164
Multi-Point Touch	177
Accelerometer	196
Accelerometer Simulation	202
Location	203
Understanding How It Works	203
Programming with Location.....	204
Windows Phone Emulator Location Simulation	206
Microphone Input	207
Compass Sensor	210

Compass Background	210
Coding with the Compass	211
Dead Reckoning Navigation	217
Gyroscope Sensor	221
Gyroscope Background	221
Gyroscope Sample	222
Motion “Sensor”	224
Motion API Background	225
Motion API Benefits.....	225
Camera Sensor	226
Camera Sensor Background	227
Video Capture	231
Conclusion	232
Chapter 4: Windows Phone Data Persistence and Networking Programming Model	233
Application Data Persistence.....	233
IsoStore File IO.....	235
Object Persistence	239
Isolated Storage Explorer.....	241
Tool Functionality.....	241
Using the Tool	242
Local Database	242
Architecture	242
Local Database Sample	243
Application Updates and Local Database.....	255
Local Database Security and Connection String Options.....	255

Web Browser Control	255
WebBrowser Control Sample	256
WebBrowser Control Scripting Sample.....	258
HTML5 Support	262
Asynchronous Programming	263
Background Threads.....	263
Connecting Windows Phone to Services and Feeds	270
HttpRequest and WebClient	271
Windows Communication Foundation Support.....	271
Web Services	271
REST+OData	277
Plain Old REST+JSON	280
Sockets	289
Bing Maps Control	290
Licensing	290
Programmatically Add a Pin	290
Data Bind to a Collection.....	292
Application Execution Model	296
User Experience	297
Event Lifecycle.....	297
Managing State.....	299
Running Under Lock Screen	301
Fast Application Switching	302
Conclusion	303
Chapter 5: Integrating with the Windows Phone Platform	305

Device Information 305

- Identifying Unique Users 305
- Device Information 306
- System Environment Information 308
- Network Status 308
- Socket Network Information 310
- DeviceInfo Code 312
- System Tray 313
- Marketplace Status and Trial Mode 314

Tasks 315

Clipboard 334

Alarms and Reminders 335

- THE ALARM CLASS 336
- The Reminder Class 337
- Alarms and Reminder Sample Project 337

Contacts and Calendar 343

- Contacts and Calendar Background 343
- Coding with Contacts and Calendar 344

Conclusion 348

Chapter 6: Windows Phone Advanced UI Development 349

The Model-View-ViewModel Architecture 349

- MVVM Overview 350
- Pick an MVVM Helper SDK 360
- GalaSoft MVVM Light Toolkit 360

Silverlight for Windows Phone Toolkit	374
Installation and Overview	374
The AutoCompleteBox Control	377
The ContextMenu Control	379
The DatePicker and TimePicker Controls.....	379
The ListPicker Control.....	381
The ToggleSwitch Control.....	382
The WrapPanel Control	383
The HubTile Control	385
The LongListSelector Control.....	387
Creating Transitions and Interactivity	394
Toolkit Page Transitions	394
Visual State Manager.....	401
The RichTextBox Control.....	408
The Microsoft Advertising SDK.....	409
Getting Started.....	409
Adding the Advertising Control	409
Conclusion	413
Chapter 7: Advanced MVVM, Services and App Connect.....	415
Advanced MVVM.....	415
The next section covers how to access syndicated services like RSS feeds from Windows Phone.Syndicated Services.....	421
Showing Progress.....	424
Lazy Load Images	427
Data Bind to Anything	430

Encryption Services..... 433
 Secure Sockets Layer433
 Securely Encrypting Data433

Working with Images on Windows Phone 440
 Images Classes.....440

The Windows Phone Media Library 441

Working with Video and Audio Media..... 446
 Supported Codecs and Containers.....446
 Video and Audio in the Emulator.....446
 Progressive Download Video446
 Streaming Video450
 DRM452
 Audio Support452

App Connect -Extending the picture Hub..... 454
 Extending the pictures Hub.....455

App Connect - music+videos Hub Integration 464
 Additional Assets465
 Testing music+videos Hub Applications.....466
 Debugging music+videos Hub Applications466
 music+videos Hub Sample Application466

App Connect – Bing Search Integration..... 472
 Building App Connect Search Applications Overview473
 App Connect Search Application Code Sample473
 Testing an App Connect Search Application478

Conclusion	479
Chapter 8: XNA Framework for Games and Applications	481
Xbox LIVE Gamer Services.....	482
Alien Shooter	482
From Demo to Game	483
Game Management.....	486
Screens and Menus	487
AlienShooter Game Structure	489
The Content Project	489
Text and Menus	494
AlienShooter Game Play	499
Sprite Animation	499
Game Object Class.....	503
Enemy Class	507
The Hero Ship Class.....	508
The Missile Class	512
The Game Status Board Class.....	513
Updated GameplayScreen Class.....	515
Collision Detection and Memory Management	518
XNA and Silverlight Composite Rendering.....	520
How Does Shared Graphics Work?	520
Shared Graphics Project Templates.....	520
The SharedGraphicsDeviceManager Class and Rendering.....	521
Silverlight and the Game Loop.....	521
Loading Content and Initialization	521

Basic Shared Graphics Demo	522
Shared Graphics with UIElementRenderer.....	524
Shared Graphics with Page Navigation.....	530
Background Graphics Infrastructure Classes	530
Silverlight Project Modifications	534
Create the Content Projects	535
Modified RichBackgroundRenderer Class	536
Conclusion	537
Chapter 9: Advanced XNA Framework Development.....	539
AlienShooter Enhancements.....	539
Load and Save Game State.....	539
Tombstone Support	544
Particle System.....	545
AlienShooter with Silverlight	548
3D Game Development.....	554
3D Game Concepts	554
Rendering 3D Primitives	557
The Game Class	564
Rendering 3D Models.....	568
3D Augmented Reality	578
XAML Code.....	578
Rendering the 3D Model in Space	579
Windows Phone Profiler	581
Conclusion	582

Chapter 10: Building Top-Rated Applications for Windows Phone.....	583
Design First	583
Use the Right Architecture	584
Code for the Best User Experience	584
Panorama Tombstone.....	585
Pivot Tombstone	585
ListBox Tombstone	586
Globalization and Localization.....	586
Globalization	587
Localization.....	594
Live Tiles API	602
Tile Update Details.....	602
Tile Images	603
Multi-Tasking and Background Processing	607
Agents.....	608
Types of Scheduled Tasks	608
Constraints for All Scheduled Tasks	608
Periodic Task Constraints	609
Resource Intensive Task Constraints.....	609
Coding with Periodic Agents.....	609
Background File Transfer Service.....	610
Background Audio.....	610
Microsoft Push Notification Services	611
Raw Push Notification Description	612

Toast Push Notification Description.....612

Tile Notification.....613

How Microsoft Push Notifications Work614

Updating Application Tile via Web URI Subscription.....615

Conclusion617

Index.....481

About the Author



■ **Rob Cameron** Employed by Microsoft since 2001, Rob Cameron is an Architect Evangelist with Microsoft Corporation based out of Atlanta, Georgia. As part of Microsoft's Developer & Platform Evangelism team, Rob focuses on development tools and technologies targeting mobile devices, gaming, and embedded devices for telecommunications, cable, and media & entertainment companies. Rob has also worked on six books for Apress including authoring the latest *Pro Windows Phone App Development*. He has a master's degree in information technology management and a bachelor's degree in computer science. Visit Rob's blog at <http://blogs.msdn.com/RobCamer>.

About the Technical Reviewer



■ **Rob Garrett** has worked with SharePoint since the early beta version of MOSS 2007 and has leveraged his talents for SharePoint architecture and design with Portal Solutions – a SharePoint consultant company in Rockville, MD. Rob has extensive background in .NET technologies and has developed software for Microsoft Frameworks since the early days of C++ and MFC. In pursuit of his career dreams, Rob left his birthplace in England for a journey to the United States on Thanksgiving Day 1999. Upon arrival, he enjoyed his first American meal from a gas station.

Acknowledgments

I would like to acknowledge the incredible support provided by the Apress team who patiently waited as I struggled to balance all the demands in my life the past year. Jonathan Hassell has been there with me through all of my books with Apress, and I sincerely appreciate his support. Debra Kelly has been a trooper prodding me along to keep this book on track with the help of Jennifer Blackwell through the final push.

Introduction

Authoring a book on mobile development with a title that starts with “Pro” is a daunting task. Put in too much introductory material, and you lose the knowledgeable developer looking to take his application to the next level. Breeze through key concepts, and the cross-platform professional mobile developer may struggle learning how to work with targeting a relatively new platform in Windows Phone.

Another challenge is culling down what to cover between Silverlight and the XNA Framework. Both are fabulous and interesting technologies so I strived to cover both in a single text, though this book is more geared toward Silverlight developers who are interesting in learning about the XNA Framework as well.

However, with the combination of Silverlight and XNA Framework rendering available in Windows Phone 7.5 and the Windows Phone OS 7.1 SDK, understanding XNA Framework development becomes even more relevant to Silverlight developers, and vice versa. Consider the airlines demo Microsoft showed off at MIX 11, where you can navigate a 3D rendering of the airplane to pick your seat, and you can see the amazing possibilities available with Shared Graphics Rendering.

This book provides a comprehensive overview of XAML and Silverlight since it is a critical aspect of Windows Phone development. It also focuses on many of the more challenging topics such as correctly architecting Silverlight for Windows Phone applications using the Model-View-ViewModel architecture as well as working with services, encryption, and gestures.

There are many tidbits throughout this book extracted from many long hours of working with top media brands to bring their content and applications to Windows Phone. I hope that sharing this knowledge of memory and CPU optimization gleaned from these experiences helps to bring your applications to the top of the rankings in the Windows Phone marketplace.

This update includes extensive coverage of new features available to developers on Windows Phone, including programming new sensors such as the Compass, Gyroscope, Camera, and Motion sensors. I also cover the new API enhancements available with Silverlight 4 compatibility as well as the extensive improvements in phone integration such as contacts and calendar integration.

Who This Book Is For

If you are an existing Silverlight developer, this book will provide details on how to leverage your Silverlight skills for Windows Phone. If you are a mobile developer looking to port an existing application to Windows Phone, then this book will give you the information you need, covering the core concepts and techniques fundamental for Silverlight for Windows Phone development.

Having an understanding of the .NET Framework will help, but if you are an experienced developer new to the .NET Framework, you will still be able to learn Windows Phone development using this book because much of the development is in the XAML markup language as well as in code.

■ **Note** You still may want to grab a C# programming text, such as Adam Freeman's *Introducing Visual C# 2010* (Apress, 2010) if you are not confident with C#.

The example code in this book is written in C#. However, much of the development is in XAML, not in C#, so if you are a VB.NET developer, the markup is almost exactly the same. For the examples that do include C# code in the code-behind, the code translates pretty easily, as the .NET Framework for Silverlight for Windows Phone is language agnostic.

How This Book Is Structured

This book consists of chapters that focus on individual topic areas of Windows Phone and is intended to be read in the order in which it is written; however, I strive to point to related sections in other chapters so if you need to skip ahead to a specific topic feel free to do so.

I spend a bit more time on Silverlight than the XNA Framework, but both are covered throughout the book. Chapter 1 starts with an introduction to the Windows Phone Platform, the AppHub marketplace, and Silverlight, and it concludes with a detailed introductory sample on the XNA Framework. Chapter 2 focuses exclusively on Silverlight UX development. Chapter 3 covers input, leveraging examples both for Silverlight and the XNA Framework. With the fundamentals for professional development on Windows Phone out of the way, Chapter 4 covers data persistence and the network programming model with Silverlight. It covers the new and exciting features available such as local database as well as Fast Application Switching, which end-users will love.

Chapter 5 covers integration with the Windows Phone platform, starting with coverage of how to obtain device and network information. It also covers launchers and chooser Tasks, including the new BingMapsDirectionsTask and BingSearchTask objects, which enables amazing functionality with just a few lines of code in your applications. Chapter 5 also includes coverage of the new alarm and reminder APIs as well as contacts and calendar integration. Chapter 6 covers advanced UI development, including extensive coverage of MVVM development, the Silverlight toolkit for Windows Phone, and the new RichTextBox control available in Windows Phone OS 7.1.

Chapter 7 builds on the MVVM coverage in Chapter 6, adding coverage for typical scenarios that you can extend to enable well-architected Windows Phone apps. Chapter 7 also covers App Connect, which enables integration with the pictures hub, music+video hub, and Bing Search integration.

Chapter 8 dives in deep on XNA Framework development for 2D applications, building on the introductory sample from Chapter 1. It includes coverage of how to leverage the Game Management sample from AppHub to create a professional-looking application. Chapter 8 also includes coverage of the new Shared Graphics Programming model where Silverlight and XNA Framework can render together. Chapter 9 starts off with adding polish to the Chapter 8 2D game development sample, covering saving and load as well supporting tombstoning in XNA. A simple particle system is added to the 2D game to add explosions. The game is also migrated to take advantage of shared graphics for menus. Chapter 9 then delves into the world of 3D game development, covering working with primitives and models in 3D. It also shows how to use the new Camera and Motion sensor to render a 3D object in space, which can enable interesting 3D Augmented Reality experiences.

Chapter 10 rounds out the book with additional coverage of MVVM and how to integrate globalization and localization so that you can expand the market for your applications. Chapter 10 also covers the Tile API, background processing, and push notifications.

Prerequisites

You will need a version of Visual Studio 2010, with Service Pack 1 of Visual Studio 2010 applied. You will also need to install Windows Phone Developer Tools, available here:

http://create.msdn.com/en-us/home/getting_started

You will want to download the Silverlight Toolkit for WP7, available here:

<http://silverlight.codeplex.com>

I also highly recommend that you review the Design Resources for Windows Phone, available here:

[http://msdn.microsoft.com/en-us/library/hh202915\(v=VS.92\).aspx](http://msdn.microsoft.com/en-us/library/hh202915(v=VS.92).aspx)

Downloading the Code

The code is available in zip file format in the Source Code/Download section of the Apress web site. Please review the readme.txt for setup instructions.

Contacting the Author

To reach the author, please go to his blog and click the Email link to send him an email:

<http://blogs.msdn.com/RobCamer>



Introduction to Windows Phone and Marketplace

Windows Phone, updated in Windows Phone 7.5, presents an exciting opportunity for developers to build and monetize mobile applications. It represents a major new investment into mobile computing by Microsoft, and in many ways is a major break from the past. To build applications for Windows Phone, it is important to understand its target consumer audience and design philosophy.

Up until roughly 2007, Windows Mobile was a growing, well-received platform targeting primarily enterprise users. Windows Mobile was taking share from competitors such as Palm and going head-to-head with RIM Blackberry, but overall it was considered fairly successful. The release of the iPhone and its consumer focus and applications store it turned the mobile device market on its head, and was a major wake-up call for Microsoft. Tack on the surge by Google's Android operating system, and the Smartphone market became hyper-competitive.

Windows Phone represents a sea change in approach by Microsoft toward mobile computing. Although Windows Mobile 6.x and earlier resembles a miniaturized version of desktop Windows, with its iconic Start screen, Windows Phone is very different with a user interface that is more similar to the ZuneHD interface – on steroids.

When the ZuneHD was introduced, it received positive feedback from critics, but it wasn't highly marketed toward consumers. It also did not have a third-party application Marketplace. Still, it was an opportunity for Microsoft to try out new mobile computing concepts on a mass audience of a few million users.

Learning from its own stumbles, taking what works from Zune, and then looking at the existing mobile landscape, Microsoft came up with Windows Phone with a dramatically different user experience from Windows Mobile, a new mobile development paradigm, and a completely different approach in the market.

With Windows Phone, Microsoft makes the end-user consumer the top priority, whereas in the past, with Windows Mobile, the original equipment manufacturer (OEM) was Microsoft's primary customer. This does not mean that Microsoft is OEMing the device directly. Instead, Microsoft spec'd out the hardware and software for the platform such that OEMs can take that specification as a basis for further innovation. OEMs can build devices with or without a keyboard, or with slightly different physical sizes, different screen types, and additional features like a camera and more storage, to name a few options.

Windows Phone devices have an 800 x 480 screen resolution, the exact same user interface and APIs, and the same range of sensors no matter what the OEM or Mobile Operator. This means that developers can build an application and be confident that it will run on all available devices.

■ **Note** Microsoft has stated that it may add a new screen resolution in the future and will provide guidance and tooling to ease development and ensure applications can run on all screens.

New Windows Phone 7.5 devices can include a Gyroscope sensor and a compass sensor available via APIs. I cover hardware in more detail later in the chapter. This may make you nervous as a developer, but the Windows Phone product team provides a software layer in the “Motion Sensor” that abstracts out the lack of Gyroscope hardware in Windows Phone RTM devices. I cover the Motion Sensor in more detail in Chapter 3 and Chapter 9.

From a software standpoint, the programming model for Windows Phone is very different from Windows Mobile. With Windows Mobile, developers created applications with either C++ or .NET Windows Forms via the .NET Compact Framework. For Windows Phone, the available programming models are Silverlight and the XNA Framework. In Windows Phone 7, Silverlight and the XNA Framework could call non-visual APIs. Windows Phone 7.5 includes “Shared Graphics,” which allows Silverlight and XNA Framework graphics, which is covered in Chapter 9.

I cover the new software and capabilities in more detail in the following sections, but first let’s answer the question of why develop for Windows Phone.

Why Should I Care?

As a developer, you may wonder whether you should invest in building applications for Windows Phone. The answer is “yes!” Windows Phone offers unique application features and integration to developers, allowing you to build amazing user experiences. I cover more on that in the next section as well as Marketplace, but let’s first discuss market potential.

Smartphone Growth and the Nokia Partnership

The mobile phone has become a major computing platform, experiencing incredible growth. As I write this, Smartphone shipments make up approximately 10% of the mobile phone market worldwide, which includes feature phones, etc. It is expected that the share of Smartphone devices will grow to 70% of the market over the next three or four years.

Much of the growth potential is still ahead. A new or remade player in the market, such as Android and Windows Phone, can come on to the scene and capture the imagination of both developers and consumers with the right level of marketing and engineering commitment. Android has become a major player in the market along with iOS. Windows Phone is set to become a major player as well with the partnership with Nokia, a worldwide leader in the Smartphone market. The partnership demonstrates Microsoft’s commitment to move into the market.

With the Nokia partnership, Nokia has committed to building phones based on the Windows Phone operating system. Nokia brings a world-class organization, a worldwide retail presence, and amazing hardware engineering to Windows Phone.

Nokia announced that it will launch its Windows Phone in six European nations in 2011, increasing the worldwide development opportunity for Windows Phone developers, given Nokia’s strong presence worldwide.

In March of 2011, IDC predicted that Windows Phone will overtake iPhone and BlackBerry by 2015 (http://seattletimes.nwsourc.com/html/technologybrierdudleysblog/2014627927_idc_windows_phone_to_overtake.html).

In April Of 2011, Gartner predicted Windows Phone will overtake Blackberry in 2013 and will overtake iPhone in 2015 (http://wmpoweruser.com/gartner-predicts-windows-phone-7-will-overtake-blackberry-in-2013-overtake-iphone-in-2015/?utm_source=twitterfeed&utm_medium=twitter).

While these are third-party predictions, it is a decisive break from previous predictions regarding Windows Mobile. Time will tell how this story unfolds, but there is no doubt that the Nokia partnership is a major upheaval of the Smartphone marketplace in favor of Windows Phone. As a developer you do not want to miss out on this worldwide opportunity and market.

The Mobile Market

The Smartphone segment of the mobile device market is expected to continue significant double-digit growth in the foreseeable future. This growth stems from new users purchasing devices as well as existing function phone and feature phone users moving to Smartphones. Despite the hype, it is way too early to suggest that one platform or another has “won” the Smartphone war. There is plenty of new growth to be had in the market before competing platforms will need to focus on taking share from one another.

Take Android as an example. It has stormed onto the market and has outsold iOS devices so far in 2011. Regarding Windows Phone, Microsoft CEO Steve Ballmer claims that his company is “all-in” with respect to commitment to the mobile market and Windows Phone. The partnership with Nokia announced in early 2011 demonstrates the high-level of commitment and desire needed to bring a robust third ecosystem to market.

In describing my perspective on what Windows Phone means to Microsoft, I like to use an analogy. When Microsoft first introduced the Xbox game console, the general consensus in the market was that the Sony PlayStation had “won” the market and there was no way the Xbox could catch up. Today, I don’t think anyone would question Microsoft’s investment or commitment with respect to the Xbox 360. Likewise, who would have guessed that Nintendo would have made a comeback like it did with the Wii? Competition is a wonderful thing for consumers. Expect even more competition and dynamism in the mobile device market!

Current Microsoft Developers

If you are an existing .NET Framework, Silverlight, or XNA Game Studio developer, you are well on your way to having the skills needed to create great Windows Phone applications. However, there are many unique aspects to mobile device development, as well as features that are specific to Windows Phone, that require new skills. This book will present to you the information you need to bring your existing skills to bear on this new and exciting platform and to build great mobile application experiences.

■ **Note** Windows Phone development initially supported C# only at launch. Since launch, Microsoft has added support for Visual Basic .NET development for both Silverlight applications and XNA Framework game development.

If you are new to Silverlight and XNA Game Studio, I spend time in this chapter introducing you to the basic concepts in these technologies to get you started. Combine this chapter with the online

references available, and you will have the necessary foundation to proceed with the rest of this text. If you are an experienced Silverlight developer, you can skip Chapter 2 and parts of Chapters 4 and 5 that cover topics familiar to Silverlight developers.

If you are an existing Windows Mobile developer, you have a great depth of understanding of .NET and deep knowledge of mobile development considerations as well. This book will help you bring your existing knowledge to Silverlight and XNA Game Studio so that you can port your existing applications to Windows Phone.

Non-Microsoft Developers

If you are an existing mobile developer on another platform, this book will help you bring your mobile development skills to bear on Windows Phone. Windows Phone applications are written in C#, which is similar to the Java language and has roots in C, so many of the language constructs will be straightforward. For additional information on C# in general, check out the Apress title *Beginning C# 2010: From Novice to Professional*, Second Edition, by Adam Freeman.

It is challenging to write a “Pro” book and not cover some of the fundamentals. It is for this reason that I spend an entire chapter on Silverlight development to quickly help experienced mobile developers who are new to Silverlight to get started. Don’t worry, though, there are plenty of advanced topics to help you build a world-class application.

Windows Phone Platform Overview

In this section, I provide an overview of the design philosophy behind Windows Phone. I also cover the available development model to familiarize you with underlying platform architecture and developer capabilities.

Phone Design Philosophy

When Microsoft introduced Windows Phone, its executives spoke about “integrated experiences” and “smart design.” Without a doubt, Windows Phone presents a unique user interface as compared to other current Smartphone operating systems, as shown in Figure 1–1.

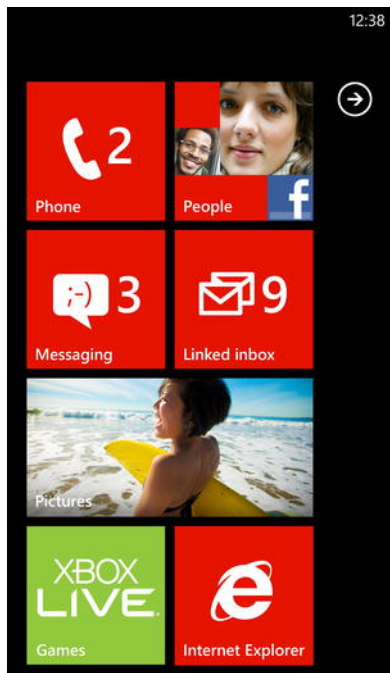


Figure 1–1. Windows Phone Start screens

The Windows Phone Start screen presents a lot of information in a clear fashion. Information and content is presented in a straightforward, easy to navigate manner without distraction. Controls have a minimalistic feel, allowing the user to focus on the content. Your applications should take the same approach.

Also shown in Figure 1–1 are several top-level tile icons pointing to the built-in “hubs,” such as the People, Office, Pictures, Zune, Marketplace, and Xbox LIVE hubs. The hubs are the built-in, top-level panoramic experiences, meaning they scroll left and right to present content in a rich fashion. Figure 1–2 shows the pictures hub. Notice that the focus is on the content, not “chrome.”

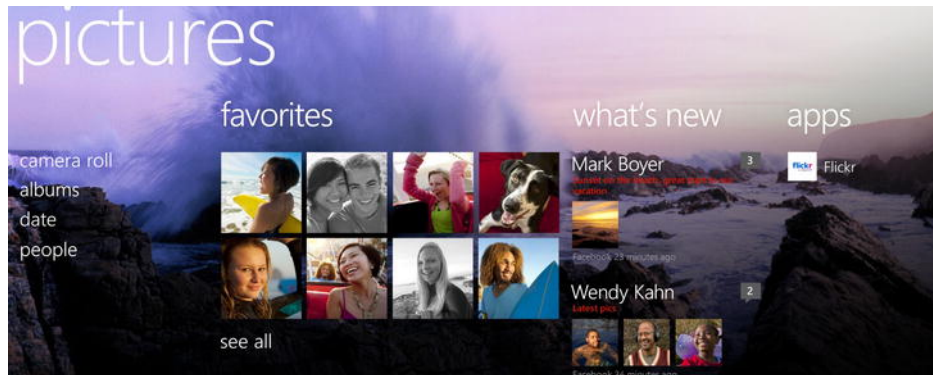


Figure 1–2. Windows Phone pictures hub Panorama experience

Figure 1–2 shows a wide view of the full user interface width of the Panorama experience. A user swipes left and right to view content in addition to scrolling up and down on each viewable section. Hubs provide an integrated view on data. For example, if you register an Exchange, Facebook, Twitter, LinkedIn, and Windows Live account, photos from social networks are pulled together into a single integrated view under “what’s new” in the pictures hub. Photos from the camera roll are shown as well as applications that work with the pictures hub such as Flickr and Photobucket. This allows the user to explore images from all sources as well as know which applications can enhance the picture hub experience.

The Panorama control is a single page for all Panorama panes. There is a little bit of “peek” on the right that shows the next Panorama pane to indicate that you should swipe left or right with only a single heading visible at any time. In addition to Panorama, the other major navigation paradigm is called Pivot Outlook on Windows Phone, and it uses a *Pivot* control as the main landing page. The *Pivot* control is virtualized, designed to display large amounts of data. The *Pivot* shows multiple headers to indicate that you should swipe left and right, and it does not have the “peek” like the Panorama. Figure 1–3 shows an example *Pivot* interface running in the Emulator.