

# Introducing Windows Phone 8 App Development

Lee Stott

Microsoft

@lee\_stott

<http://www.microsoft.com/uk/faculty>

<http://dev.windowsphone.com>



# Agenda

## Introducing Windows Phone 8

New app platform in WP8

Application development models

WP8 version of WinRT

Supported application models

Getting started with WP8 app development

New features overview

Using the Windows Phone Developer Tools

WP7x compatibility

# New Windows Phone 8 Hardware

Beautiful new hardware from Nokia, HTC, Samsung and more...



# Modern Smartphone Platform



New multicore chipset

New graphics processor

Increased RAM: 1GB or 512MB

More Screen resolutions

Removable, encryptable storage

NFC

# Shared Windows Core

Windows 8 and Windows Phone 8 Share Many Components At The Operating System Level



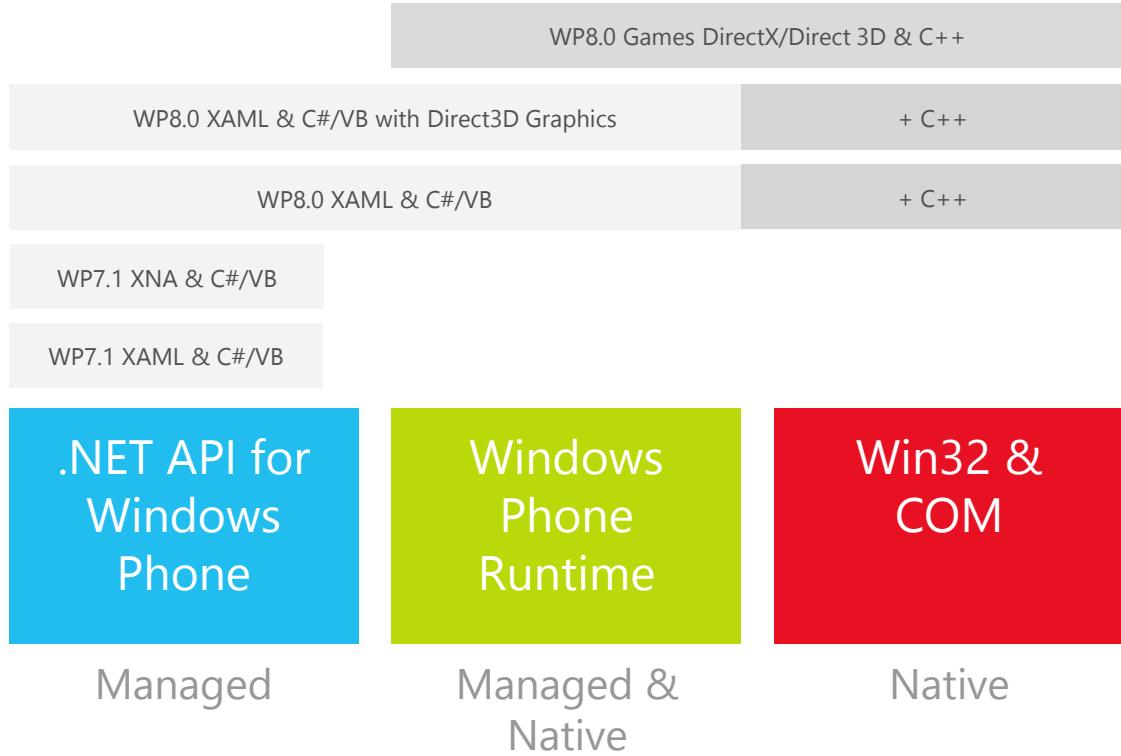
# What 'Shared Core' Does – And Doesn't - Mean

- Shared Core means
  - OS components such as the kernel, networking, graphics support, file system and multimedia are the same on both Windows 8 and Windows Phone 8
  - Hardware manufacturers work with the same driver model on both platforms
  - Windows Phone gets the support for multi-core and other hardware features that Windows has had for years
  - These solid, common foundations makes it easier to extend the Windows Phone platform into the future
- It doesn't mean
  - Windows 8 and Windows Phone 8 developers work to exactly the same APIs
    - (though you will see more commonality as new features are introduced to both platforms in the future)

# Windows Phone 8 Programming APIs

Windows Phone 8 supports

- Managed app dev using the WP7.1, WP8.0 .NET and WinPRT APIs
- Native app dev using WinPRT and Win32
- Games dev using the WP7.1 XNA framework
- Games dev using Direct3D or DirectX



# Some Key Differences

It's important to design for the platform differences as well as similarities



## Screen Size

### Windows Phone

800x480, 1280x720, 1280x768

Portrait, Landscape

### Windows

1024x768 →

Portrait, Landscape, Snapped



## Controls

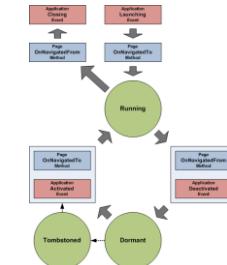
### Windows Phone

Panorama, Pivot, ListPicker

LongListSelector

### Windows

GridView, ListView, Semantic  
Zoom, FlipView



## Lifecycle

### Windows Phone

Launched from start/apps list.

Tombstones apps

### Windows

Resumes existing apps

No tombstoning

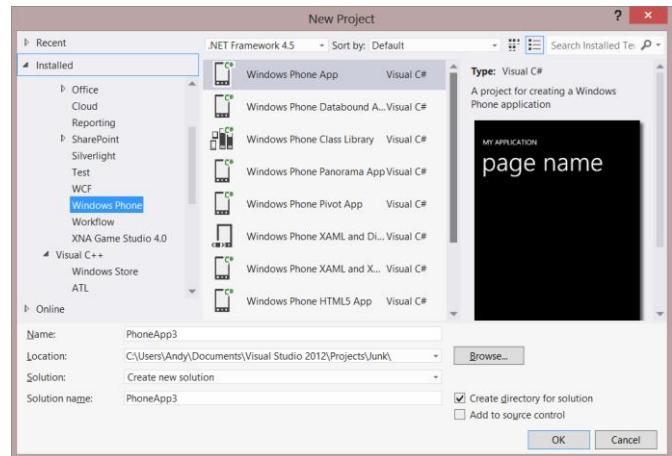
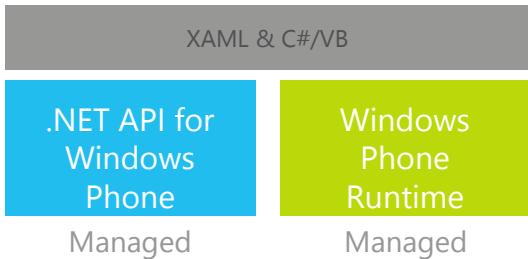
Windows Phone 8 offers many additional ways of building apps compared to Windows Phone OS 7.1

## App Models



# XAML UI with Managed Code

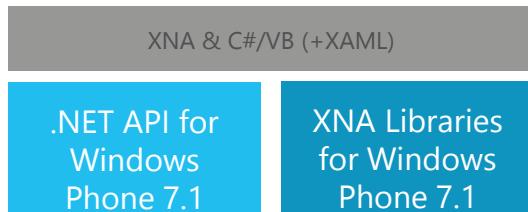
- The most common way to build apps for Windows Phone
- UI defined using XAML
- Logic written using C# or Visual Basic .NET
- Access .NET APIs and Windows Phone Runtime APIs



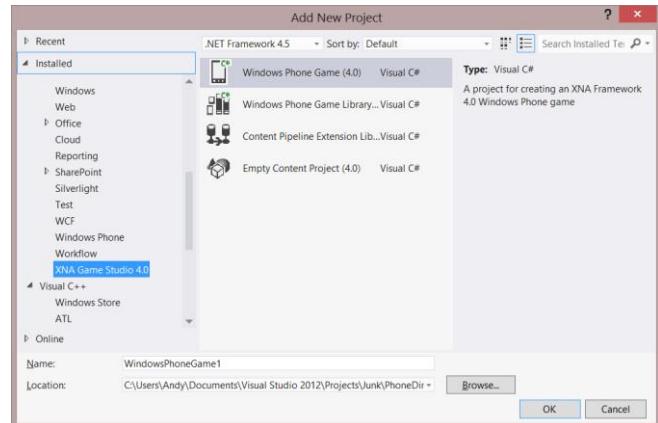
# DEMO 1: XAML and Managed Code

# XNA Games using Managed Code

- You can develop games for Windows Phone using the XNA framework
- Same functionality as in Windows Phone OS 7.1
- Logic written using C# or Visual Basic .NET
- Access .NET 7.1 APIs, not Windows Phone 8 APIs
- Same new project templates as Visual Studio 2010



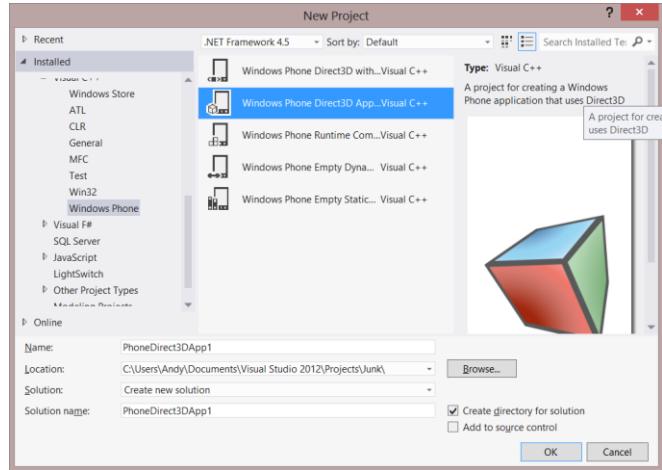
Managed



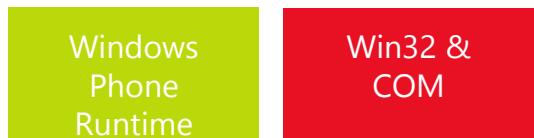
## DEMO 2: XNA and Managed Code

# Direct3D App

- Direct3D app written entirely in native code, and which use only Direct3D for its UI
- Games development – significant sharing of code base with a PC version
- Access Windows Phone Runtime APIs – significant subset of the Windows 8 SDK
- Easier to share native components such as compute engines, graphic libraries and API sets



Direct3D & C++

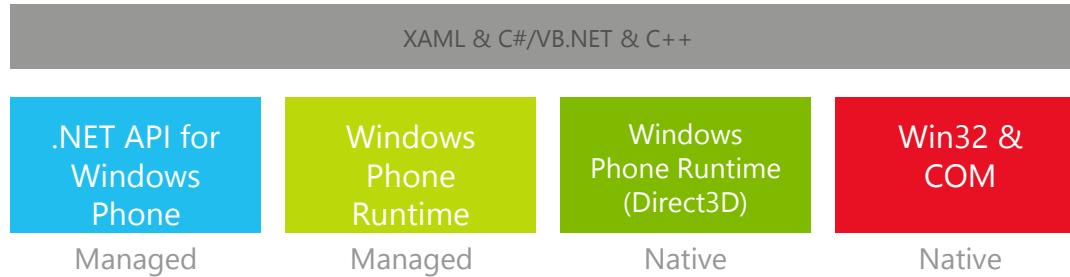


Native

Windows Phone

# Direct3D and XAML

- Developers can also build managed apps using XAML that incorporate graphics created by a Direct3D native library
  - Allows addition of powerful graphics to XAML UIs
- Use the Windows Phone Direct3D with XAML App project template
  - Available in Visual C#, Visual Basic and Visual C++ categories in the Add New Project dialogue



# DEMO 3: Direct3D Games

# XAML/Managed plus Native Code

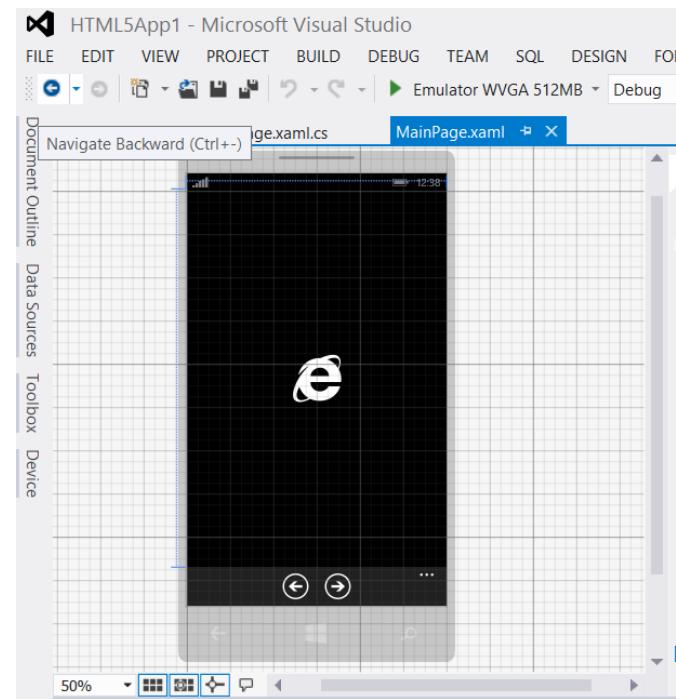
- Managed apps can also interact with native libraries
  - Add C++ Dynamic Link Library or Windows Phone Runtime Component projects to a managed XAML solution
- Win32 API set supports Winsock and File I/O functions to ease porting of existing native code libraries
- Implement compute-intensive components in native code for increased performance
  - E.g. Image processing, compute modules, document rendering...



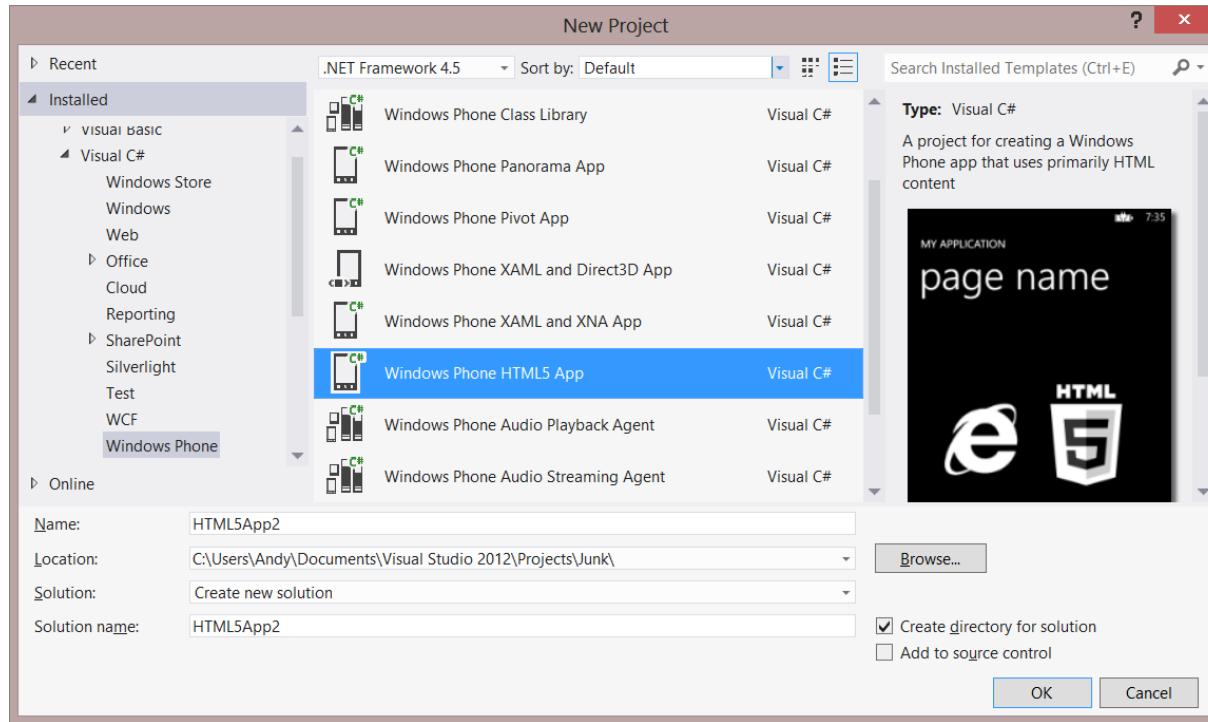
## DEMO 4: Managed and Native Component Interop

# HTML5 App Development

- Windows 8-style HTML5/JavaScript app development is not supported on Windows Phone 8
  - Windows Phone Runtime projections to C#/VB.NET and C++ only
- However, Windows Phone 8 includes Internet Explorer 10
  - Another shared codebase with Windows 8
  - Great support for HTML5 (2 x feature support compared to Windows Phone 7.5)
  - New JavaScript processing engine (4 x faster than Windows Phone 7.5)
- This same browser is at the heart of the WebBrowser control
  - You can build HTML5-based apps rendered in the WebBrowser against local or web-based content



# HTML5 App Project Template



## DEMO 5: Managed App Displaying HTML Content

 Windows Phone | Dev Center

Dashboard Develop Community Help



Welcome to the Windows Phone Dev Center.

**SUBMIT APP**

Join Dev Center and publish your app in the Windows Phone Store.

**GET SDK**

Download the tools to build great Windows Phone apps.

# Getting Started Developing for Windows Phone 8.0

# Getting The Tools

- The Windows Phone Developer Center is your base for all things Windows Phone related!
- In Windows Phone 7.x, used to be called AppHub and was at <http://create.msdn.com>
- Now, at <http://dev.windowsphone.com> !
- Get the SDK - FREE download!
- Read articles, browse and download samples and participate in the community forums
- Submit apps for testing and publication to the Windows Phone Store

# Getting a Windows Phone Developer Account

- You do not need a Windows Phone Developer account to download the SDK and start developing apps
- You do need a developer account to unlock a phone for development and to submit apps for testing and publication in the Windows Phone Store (formerly known as Windows Phone Marketplace)
- To get a Developer Account:
  - Included if you have an MSDN subscription
  - **Free to students who have a Dreamspark subscription**
  - <https://www.dreamspark.com/Product/Product.aspx?productid=26>
  - \$99 charge per annum for individual developers
- Register for an account at the Windows Phone Developer Center  
<http://dev.windowsphone.com>

# Development PC Requirements

- Your computer must meet the following system requirements to run Windows Phone SDK 8.0:

Supported operating systems	Windows 8 64-bit (x64) client versions
Hardware	8 GB of free disk space 4 GB of RAM (recommended) 64-bit (x64) motherboard
Windows Phone Emulator	Windows 8 Pro or higher (for Hyper-V) and Second Level Address Translation (SLAT)

# Second Level Address Translation (SLAT)

- SLAT is required to run the WP8 emulator.
- PCs that support SLAT are Intel-based processors that start with i (e.g., i3, i5, i7, i9) or any CPUs based on the Nehalem, Westmere, or Sandybridge micro-architectures.
- To determine if your machine supports SLAT, perform the following steps:
  - Download SysInternals/TechNet CoreInfo at  
<http://technet.microsoft.com/en-us/sysinternals/cc835722>
  - Run "coreinfo -v"
    - If you have "\*" next to "EPT" you should be good.
    - If you have a "-" next to EPT then you should start looking for a new computer :)

# Coreinfo Output

- If you see the below, you're in good shape for running the WP8 emulator on this machine:

```
Coreinfo v3.02 - Dump information on system CPU and memory topology
Copyright (C) 2008-2011 Mark Russinovich
Sysinternals - www.sysinternals.com

Intel(R) Core(TM) i5 CPU      M 450 @ 2.40GHz
Intel64 Family 6 Model 37 Stepping 5, GenuineIntel
HYPERVISOR      - Hypervisor is present
VMX            * Supports Intel hardware-assisted virtualization
EPT            * Supports Intel extended page tables (SLAT)
```

- If you see the below, you can't run the WP8 emulator on this machine:

```
C:\>Coreinfo>Coreinfo.exe -v

Coreinfo v3.04 - Dump information on system CPU and memory topology
Copyright (C) 2008-2012 Mark Russinovich
Sysinternals - www.sysinternals.com

Intel(R) Core(TM)2 Quad CPU    Q9000 @ 2.00GHz
Intel64 Family 6 Model 23 Stepping 10, GenuineIntel
HYPERVISOR      - Hypervisor is present
VMX            * Supports Intel hardware-assisted virtualization
EPT            - Supports Intel extended page tables (SLAT)
```

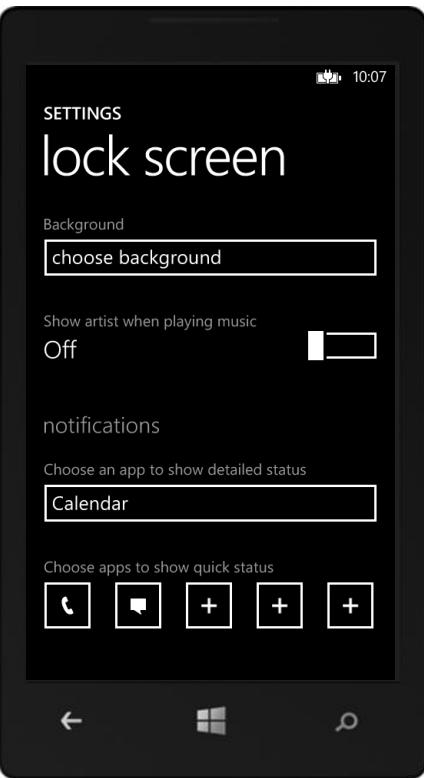
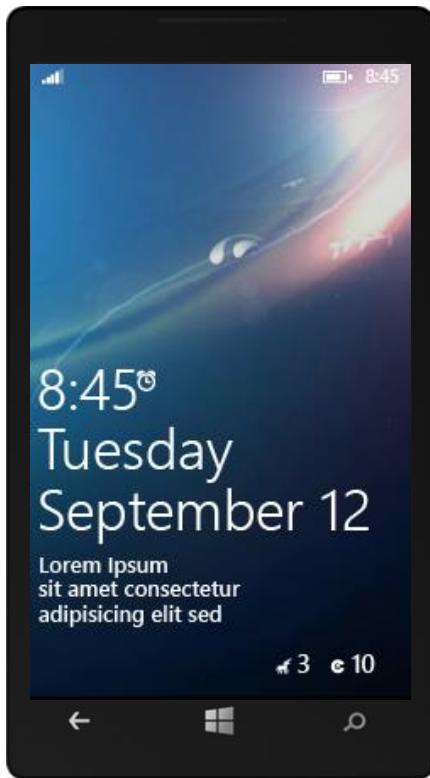
# Overview of New Features for Developers in Windows Phone 8.0

# Tile Templates and Tile Sizes

- Windows Phone 8 supports three Tile templates
  - Flip – flips from front to back (similar to the WP 7.1 Tile template)
  - Iconic – clean iconic layout designed to reflect Windows Phone design principles
  - Cycle – cycles through up to nine images



# Lock Screen on Windows Phone 8



- You can register your app as a lock screen provider, which enables:
  - User can select your app to show detailed status on the lock screen
  - Can select your app as one of the five apps to show quick status (icon and count)
  - Can select your app as the lock screen background image provider

# New Launchers

- Launchers are APIs that help a user perform common tasks
- Invoke part of the phones' built-in capabilities to perform tasks such as
  - Take a photo
  - Add a contact
  - Send an email or SMS message
  - Etc...
- New Launchers in Windows Phone 8:
  - SaveAppointmentTask
  - ShareMediaTask
  - MapDownloaderTask
  - MapUpdaterTask
  - MapsTask
  - MapsDirectionsTask

# Maps

- Windows Phone 8 has new Maps controls
- Bing Maps control from Windows Phone OS 7.1 is still supported, but deprecated
- The new Maps controls use technology supplied by Nokia
- New features:
  - Vector-based for faster rendering
  - Four cartographic map modes
  - Light and dark color modes
  - Display landmark and pedestrian features

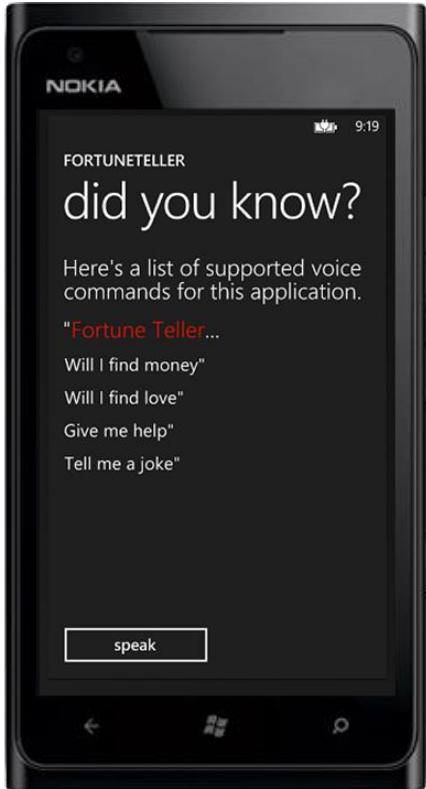


# Location and Location Tracking

- New Location API in the Windows Phone Runtime API set
  - Similar to Windows 8 Location API
- Windows Phone OS 7.1 .NET Location API still supported
- Background location-tracking apps
  - Run continuously in the background when the user navigates away from the app
  - Enables scenarios such as Run Trackers, turn-by-turn navigation



# Speech



- Windows Phone 8 allows you to enable users to robustly interact with your app using their voice
- Two types of voice interaction are new:
  - **Voice Commands** - Allows users to deep-link into your app by holding down the Start button and speaking a prefix you specify for your app, followed by a command that you define.
  - **Speech Recognition and Text-to-Speech APIs** - While in the context of your app, allow users to provide input using their voice, and readout text to users via text-to-speech

# Wallet

Credit & Debit Cards

Loyalty & Membership Cards

Access Saved Deals

Supports NFC 'Tap to Pay'



# UI Controls

New Controls in Windows Phone 8 SDK

- LongListSelector
  - Flat lists
  - Grouped lists – with headers
  - Jump List
  - Formerly in the Silverlight Toolkit
- Pivot and Panorama now in ROM
- WebBrowser control now based on Internet Explorer 10



# 3 Screen Resolutions



WVGA  
800 x 480  
15:9



WXGA  
1280 x 768  
15:9



720p  
1280 x 720  
16:9

# Camera and Photos

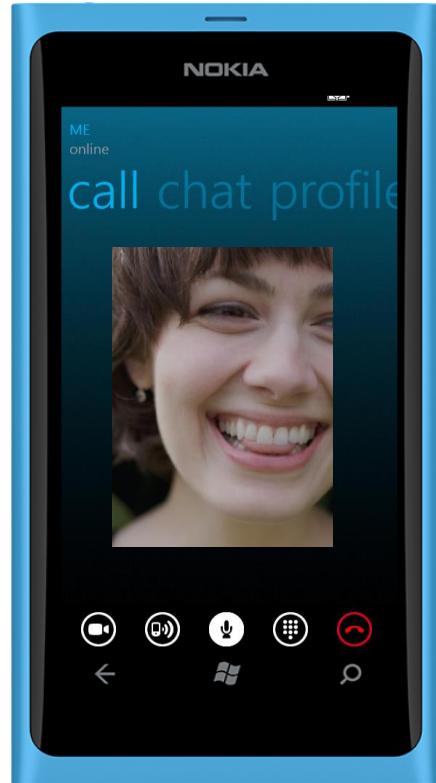
- Lenses
  - Type of extensibility available to apps that provide unique camera functionality via the camera APIs
  - As a lens, your app provides the user with a viewfinder experience and interacts directly with the camera
- New Picture.GetPreviewImage() method to return thumbnail images
- New Photos extensibility features
  - In addition to extending the Photos Hub, the photo app picker, and the share picker, your app can now integrate with the photo edit picker or register as a rich media editor
- Background Photo Auto-upload
  - Automatically upload photos to a photo storage service using a resource-intensive background agent
- Advanced camera capture APIs for apps that require fine control of the camera
  - Use the PhotoCaptureDevice class to control photo properties such as ISO, white balance, and exposure. You can even programmatically adjust the focus position.
  - The AudioVideoCaptureDevice class offers a similar level of control for video and audio recording

# Media

- Music media library
  - Add and remove music files from the user's music collection with the **SaveSong** and **Delete** methods of the **MediaLibrary** class
- Pictures media library
  - In Windows Phone 8, for each picture saved on the phone, the following two thumbnail images are automatically created and can be retrieved by your apps:
    - A small thumbnail image. You can retrieve this thumbnail with the **GetThumbnail** method of the **Picture** class.
    - A preview image that has the same dimensions as the phone's screen. You can retrieve this preview image with the new **GetPreviewImage** extension method of the **Picture** class.
- Background Audio
  - New **PlayStateChangedEventArgs** returns data when the state of the **BackgroundAudioPlayer** changes

# VoIP and Video Chat

- Incoming VoIP calls work like any other call
- Integrates with built-in phone features
- VoIP apps continue to run in the background
- Available to all developers



# Bluetooth and NFC



- Create apps that communicate with other phones using Bluetooth technology
- Bluetooth API enables the following scenarios for Windows Phone 8:
  - App-to-app communication
  - App-to-device communication
- Proximity API enables:
  - App-to-app connection using Bluetooth technology
  - Establish a Wi-Fi, Bluetooth, or Wi-Fi Direct connection between your app and an instance of your app on a proximate device
  - Send data between devices using NFC.
  - Use a phone to interact with NFC tags

# Better Sharing with NFC



# Other Networking Enhancements

- Windows.Networking Windows Phone Runtime API
  - New networking API highly compatible with Windows 8
- Incoming Sockets
  - Supported in both System.Net.Sockets and Windows.Networking.Sockets, which means your app can listen for an incoming network connection
- Support for IPV6
- Winsock native API support
  - Helps porting of existing native libraries that use the Winsock API

# Storage – Read from SD Card

## MicroSD Support

- Apps can use the Microsoft.Phone.Storage API to read files from an SD card
- User can also use SD card to extend device storage for their personal files and media
  - Store photos, music, videos
- Can install apps from a MicroSD store



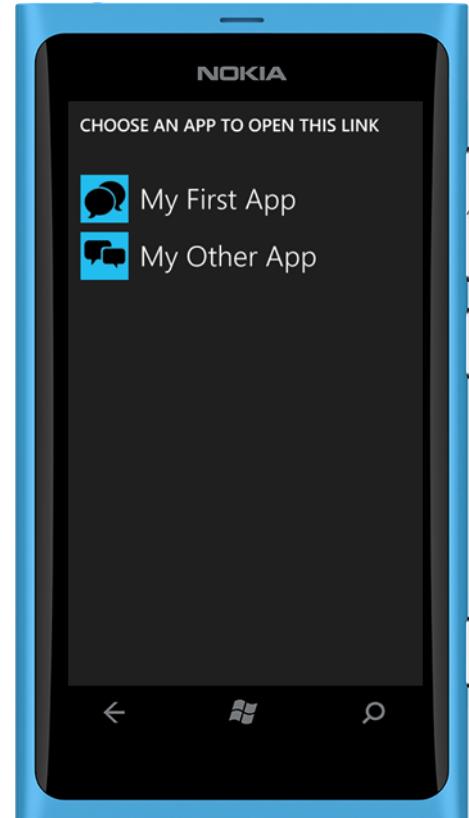
# Custom Contact Store

- Create a set of contacts owned exclusively by the app, though visible through the phone's built-in Contacts app
- APIs are provided to create, update, delete, and query the app's contacts
- Sync the app's contact list with a remote list maintained by the app's cloud service

# File and Protocol Associations

Enables App to App Communication

- App can register a File Association
  - Automatically launch your app when a file of the registered type is received as an email attachment or opened in the browser, or through Sharepoint
  - One app can launch another by sending it a file of the registered type
- App can register a protocol
  - Allows your app to automatically launch when another app 'opens' a special URI
    - Protocol is the first part of a URI, e.g.  
**myprotocol:/ShowProducts?CategoryID=aea6ae1f**
    - App launches another and passes it data in the remainder of the launch URI



# In-App Purchase

Build apps with higher and recurring revenue opportunity

## Extend reach

Attract a larger user base with free baseline experience

## Upsell

Upsell users to purchase paid digital goods and services that enhance or extend the app experience in the app itself

## Customize

Add and modify different paid options for different user segments

# In-App Purchase used for digital content

Offer **digital** assets

- ⌚ **Durables:** buy once & own forever, e.g. new game levels, maps, game items
- ⌚ **Consumables:** game currency, movie rentals, access to digital magazines for 6 months, etc.



# Ready for Business

## 1 Complete Security Platform.

Secure Boot.

Bitlocker Encryption.

## 2 Flexible App Distribution.

Business controlled.

## 3 Device Management.

Enable integration with software management systems.

## 4 Familiar Office apps



# Company Hub



apps for me news headlines alert center my profile

Get Ready  
The Get Ready mobile app is a pilot version to be made available to ...  
★★★★★

MS Path Finder  
Microsoft Real Estate & Facilities (RE&F) group leverages the Wind... Phone 7  
★★★★★

DriveTime Player  
DriveTime is a monthly on-demand, on-the-go audio program featuring interviews  
★★★★★

Sports Leads New Wave of Content Coming to Xbox  
Kinect to Create 3-D Home

XBOX MUSIC  
Microsoft's Xbox to sell music  
Kinect Technology to Help Satellites Dock in Space

SmartGlass Integrates Xbox with Tablets, Phones  
MSI Announces Windows 8 Slider

Your password will expire in 2 days  
Fix for the latest Windows 8 pre-view build now available at the download center

TICKET STATUS  
POP Account sync issue  
6/21/2012

Marketplace: Ergonomic Keyboard has been approved  
6/21/2012

KEVIN GALLO  
Busy in presentation  
Team meeting due in 45mins  
14 Unread Emails

map of San Francisco showing office locations

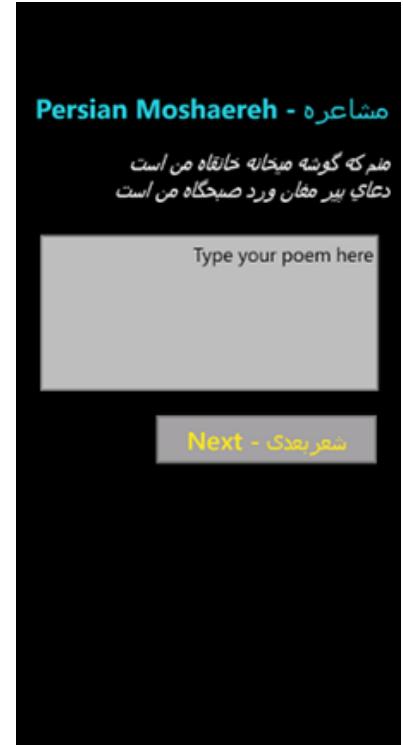
Team Members  
HR Contacts

Windows Phone

# Localization and Globalization

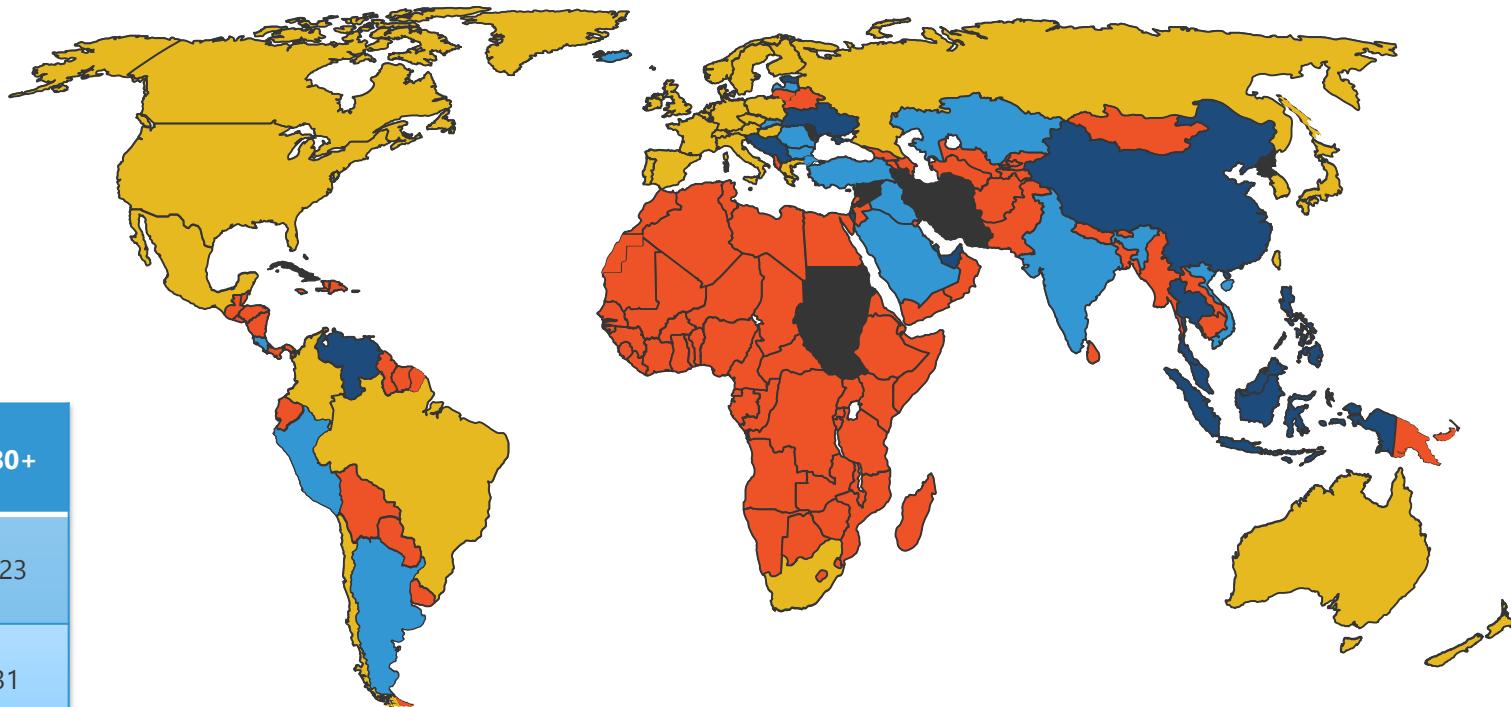
## RTL and BiDirectional Support

- Windows Phone 8 supports languages such as Arabic, Hebrew and Persian that are read from right to left
  - **FrameworkElement.FlowDirection** property allows right-to-left layout in **StackPanels** and other content controls
  - **FlowDirection** property also enables RTL text display in text controls
- Sometimes RTL languages contain content that flows from left to right. For example, a network share name such as \\server1\folder\file.ext, should always be displayed LTR
  - In such cases, you can use the [Run](#) element to contain the LTR text, and then set the [FlowDirection](#) property on the [Run](#) element to "LeftToRight"



# Consumer Store Coverage: Over 180 countries

120+ new



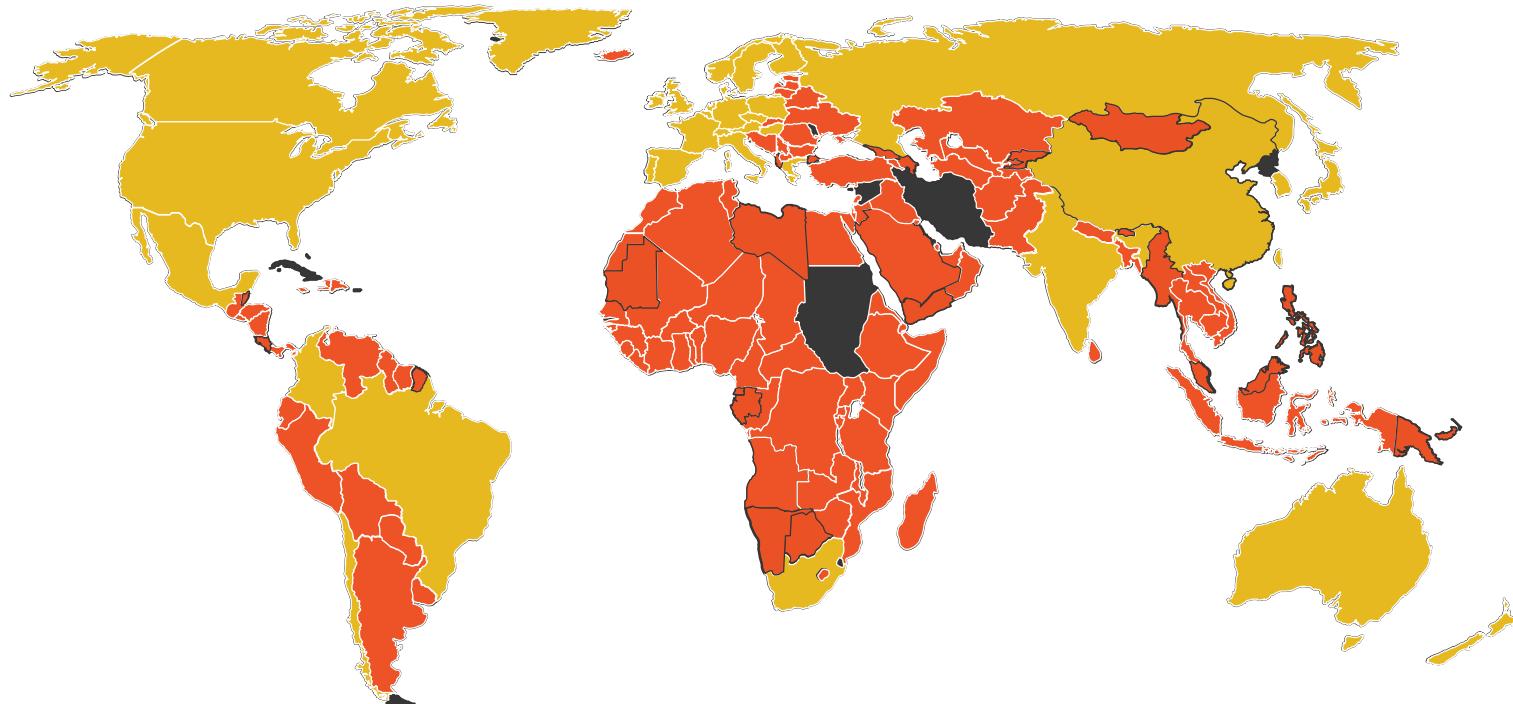
	180+
	123
	31

**29 Countries**  
Windows Phone 7.0

**63 Countries**  
Windows Phone 7.5

**180+ Countries**  
Windows Phone 8

**Developer:** Over 180 countries  
140+ new



**38 Countries**  
Windows Phone 7.x

**180+ Countries**  
Windows Phone 8.0

# Languages: 50 languages

26 New

## Windows Phone 8

50 Languages

## Windows Phone 7.5

24 Languages

Chinese Simplified

Chinese Traditional

Czech

Danish

Dutch

English (US)

Finnish

French

German

Greek

Hungarian

Indonesian

Italian

Japanese

Korean

Latvian

Lithuanian

Malay

Polish

Portuguese (Portugal)

Portuguese (Brazil)

Russian

Spanish (Spain)

Swedish

Albanian

Arabic

Azerbaijani (Latin)

Belarusian

Bulgarian

Catalan

Croatian

English (UK)

Estonian

Filipino

Hebrew

Hindi

Kazakh

Macedonian

Norwegian (Bokmål)

Persian

Romanian

Serbian (Latin)

Slovak

Slovenian

Spanish (Mexico)

Thai

Turkish

Ukrainian

Uzbek (Latin)

Vietnamese

# Windows Phone 8 Apps Run Faster ☺

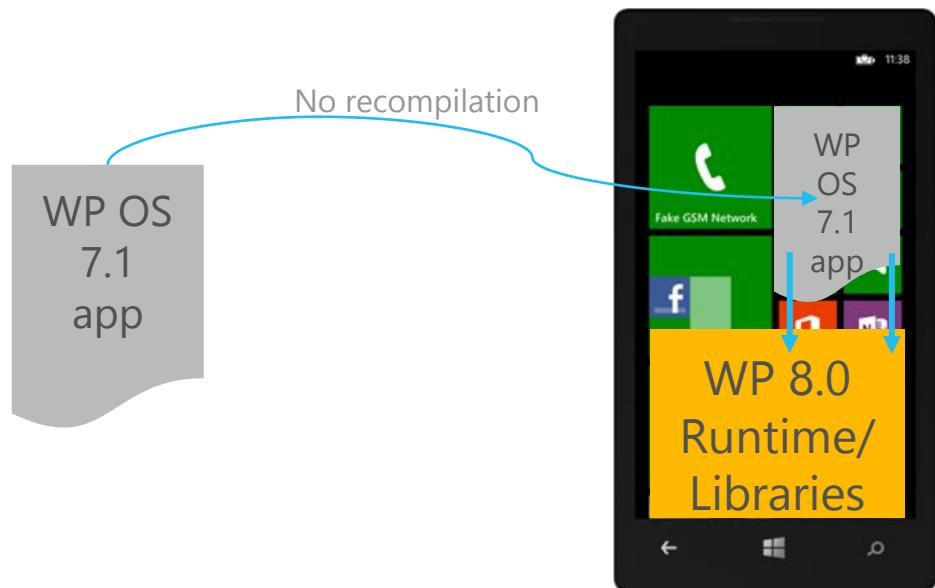
- Managed Apps are NGEN'd for you (compiled to native) in the Windows Phone Store in Windows Phone 8, so typically start and run faster ☺
- When you build your app in Visual Studio, the code is not compiled into a native image, but into a machine-independent Common Intermediate Language (CIL) binary file (formerly known as Microsoft Intermediate Language, or MSIL)
- When you submit your app to the Windows Phone Store, you submit the CIL file
  - On submission, CIL file is converted to optimized Machine Dependent Intermediate Language, or MDIL
  - When a user downloads your app to a device, it is pre-jitted to a native image
-

# Windows Phone 7.x App Compatibility



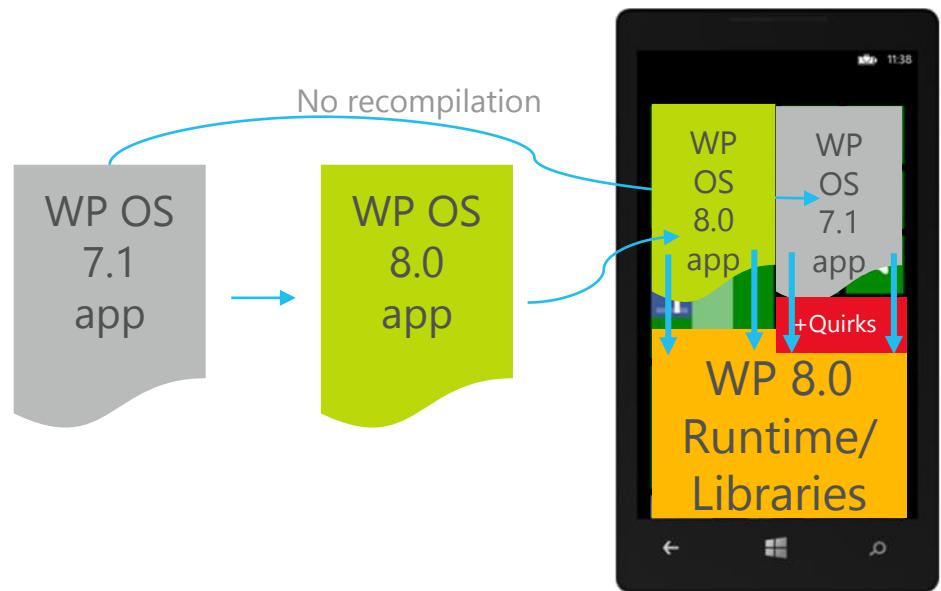
# Windows Phone OS 7.1 Apps on Windows Phone 8

- In general, the Windows Phone app platform enables apps that target Windows Phone OS 7.1 to run without modification or recompilation on Windows Phone 8
- Same APIs may exhibit slightly different behaviour in WP 8.0 compared to WP 7.1
  - See the topic **Windows Phone app platform compatibility** in the documentation



# Windows Phone OS 7.1 Apps on Windows Phone 8

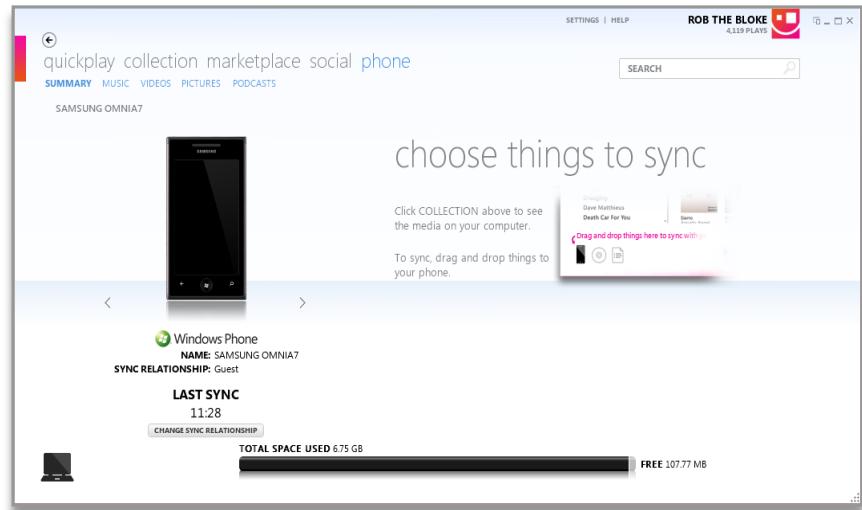
- Instead of breaking WP 7.1 apps running with the WP 8.0 runtime, quirks mode is applied to retain WP 7.1 behaviour
- Apps that are upgraded to WP 8.0 and recompiled run on the phone without quirks mode being applied



# Using the Windows Phone 8.0 Development Tools

# Connecting to the Windows Phone

- If you are developing apps for Windows Phone OS 7.1, you must install the Zune software to connect to the Windows PC
  - Also used to synchronise media
  - Perform phone updates
- Windows Phone 8 devices require only a USB connection

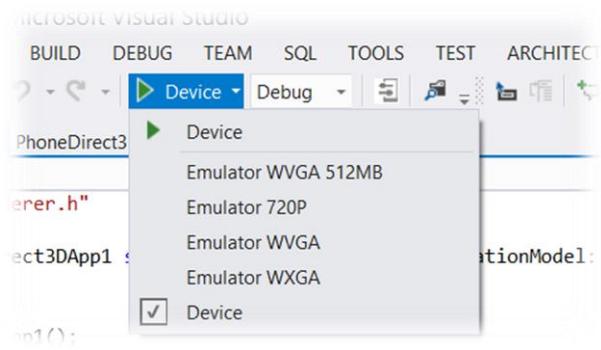


# Developer Phone Registration

- Before you can deploy your own programs to the device you need to register it as a developer device
- This is done once for a particular phone
- Registered developers can register up to three devices
- Registered students can register one device



# Deploying to the Phone



- Visual Studio lets you select the target device for your program when you run it
- The development environment is exactly the same for both platforms
- You can debug in exactly the same way for each too

# Applications on the Phone

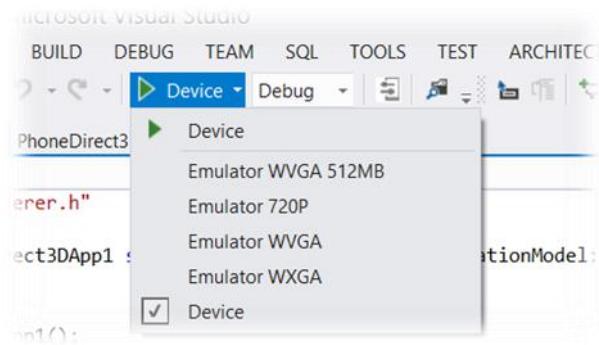
- Once you have deployed an application the phone device it is stored on the device for later use
- You are limited to ten of your own applications on the phone at any one time
- You can also send compiled versions of your application to other registered developers for them to use on their developer devices

# Visual Studio Debugging

- Visual Studio provides an exceptional debugging experience
- This experience extends to Windows Phone
- You can do all the debugging actions in Windows Phone that you can do with a Windows PC application
  - Breakpoints
  - Single Stepping
  - Viewing and modifying variables

# To Test Your app as an Optimized Native Image

- Select **Device** as the deployment target. You cannot test native code generation using the emulator
- Deploy and run your app on the device by using **Ctrl+F5** (Start without debugging) or **Alt+F1** (Start Windows Phone Performance Analysis) to test the app as an optimized native image.
- Test the app's performance and responsiveness



# The Windows Phone Emulator

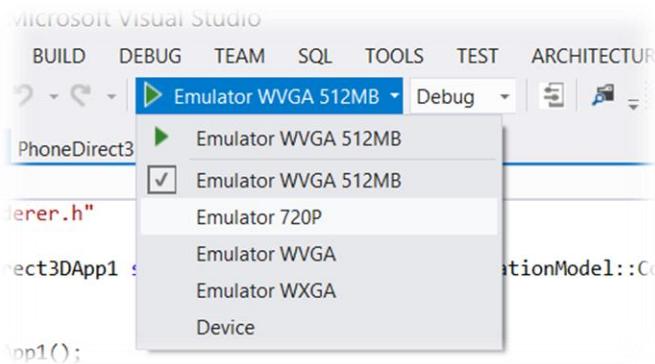
- The Windows Phone emulator runs as a Hyper-V virtual machine on your Windows PC
- It contains the same software as a “real” phone, but built for the Windows PC platform
- The emulator is supplied with the Windows Phone SDK
- You can perform location and orientation simulation using the emulator
- You can use the Simulation Dashboard in Visual Studio to manage the emulator environment
  - Lock and unlock the phone
  - Control the quality and availability of the network connection



# Emulator Performance

- The emulator is not a reliable way of predicting how a program will perform on a real device
  - The processor in a Windows PC may be more powerful than the one in the phone
- The emulator is for functional testing only
- If you have any concerns about performance you should ensure you run your program on a real device
- There are profiling tools that you can use to inspect the activity of your programs, both in the device and in the emulator

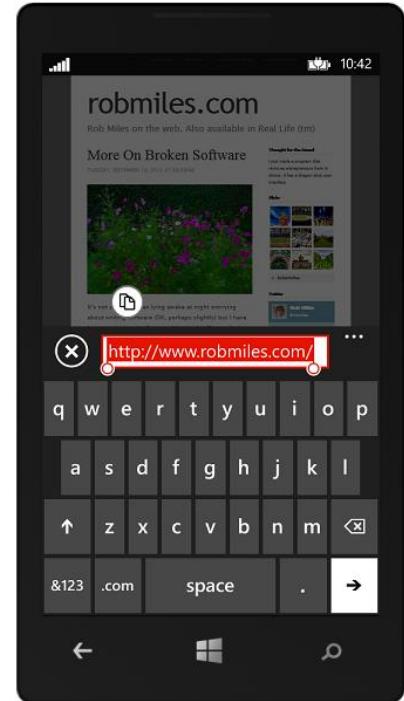
# Deploying to the Emulator



- Visual Studio lets you select the target device for a program when it runs
- The emulator is started the first time you deploy to it
  - It will then remain running until you stop it, and maintain its own local storage during that time
- Emulators provided for each of the new screen resolutions, plus WVGA 512MB to emulate reduced memory Windows Phone 7.5 devices

# Emulator Interface

- You can use the PC mouse to control the emulator
- If you have a multi-touch display you can use multiple touchpoints to pinch and zoom on the display
- The emulator will emulate the touch keyboard so you can type by clicking on the keys
- Use the PgUp/PgDn keys to switch between the emulator software keyboard and using your PC keyboard for text input



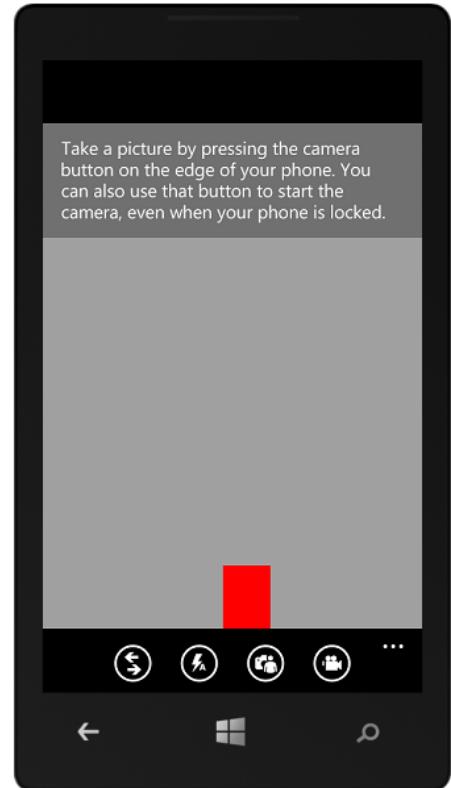
# Emulator Orientation

- Use the command bar at the right side of the emulator to access an extra set of tools
- These allow you to rotate the emulator into different orientations
- Use this to test how your software responds to orientation changes

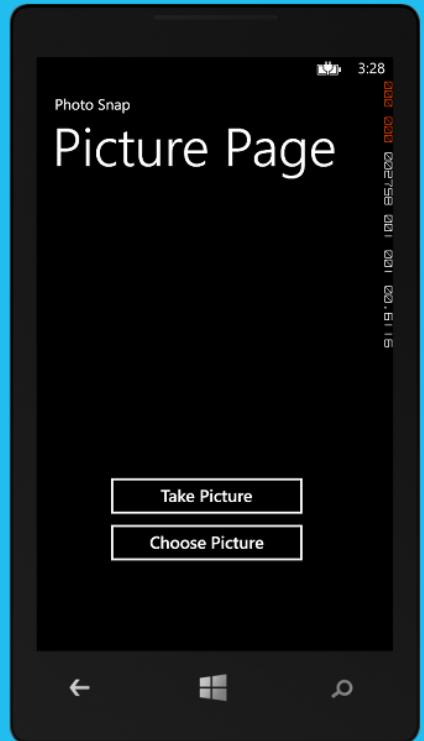


# Camera Emulation

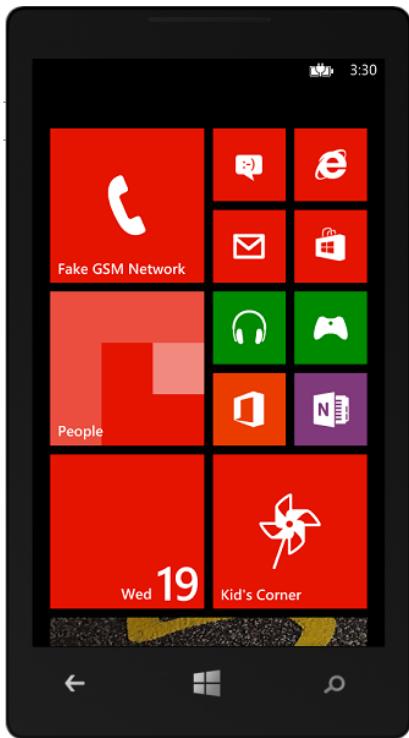
- The camera emulator just “takes” a simple photograph and returns it
- You can use this to show your applications are invoking the camera task and receiving the result correctly



# Demo 5: Photo Snap



# Location Emulation

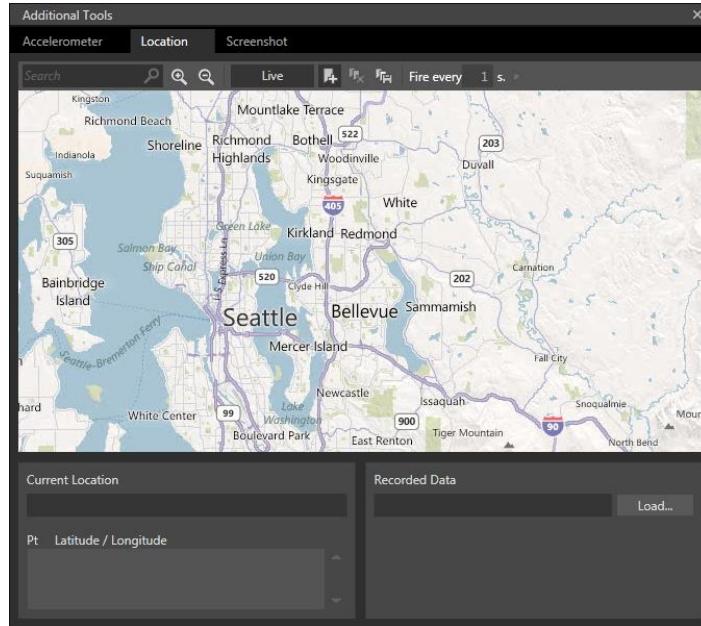


- Windows Phone applications can use the emulator to generate location data for them
- To do this we open the Additional Tools pane on the emulator

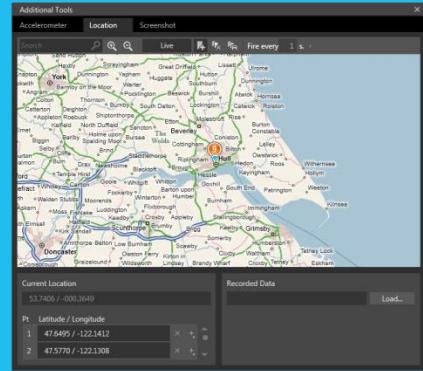
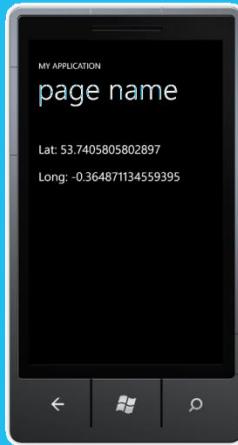
Additional Tools

# Additional Tools

- This is the Location tab in the Additional Tools
- You can click on any location to “move” the emulator to that position
- You can also place pushpins to describe a route, and then “replay” that route

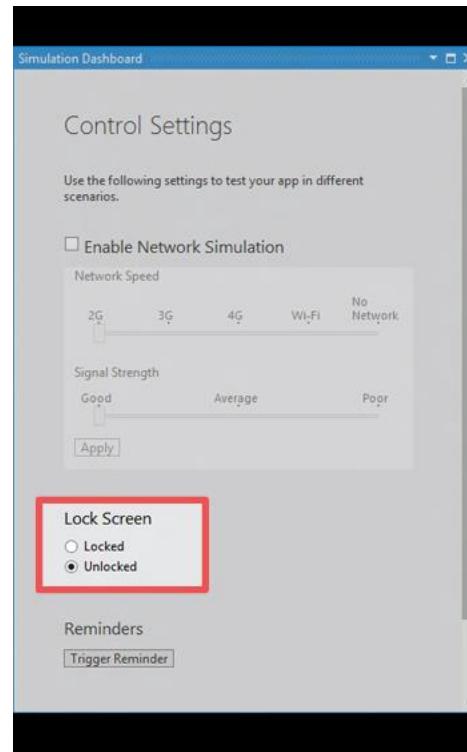


# Demo 6: Location Demo



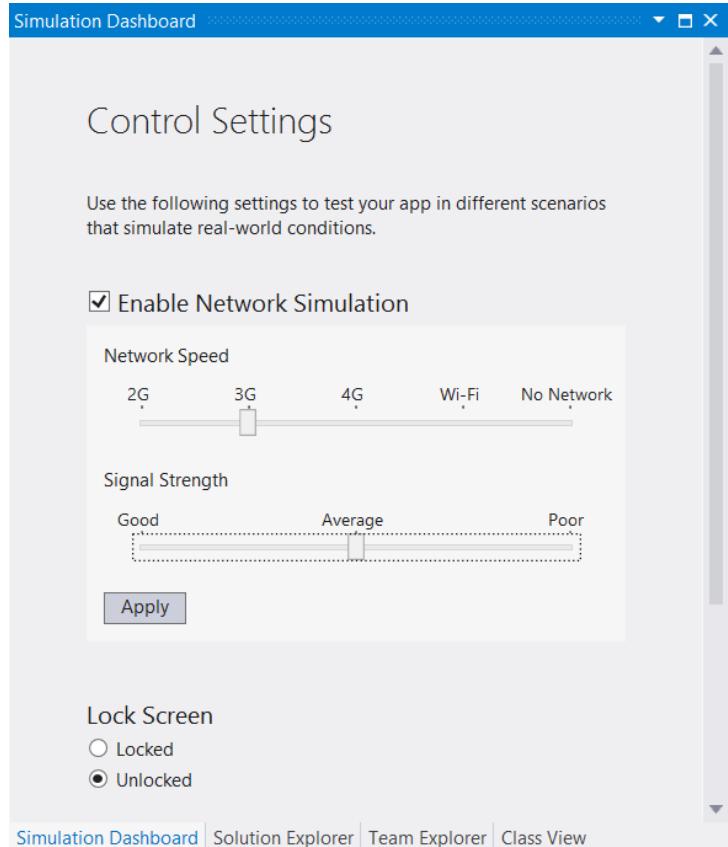
# Lock the Screen in the Emulator

- Showing the Lock Screen in the emulator is a good way of testing what happens when your app is deactivated
- Using the Keyboard:
  - Press F12 on your computer keyboard twice
- Using the Simulation Dashboard:
  - On the Visual Studio Tools menu, open the Simulation Dashboard



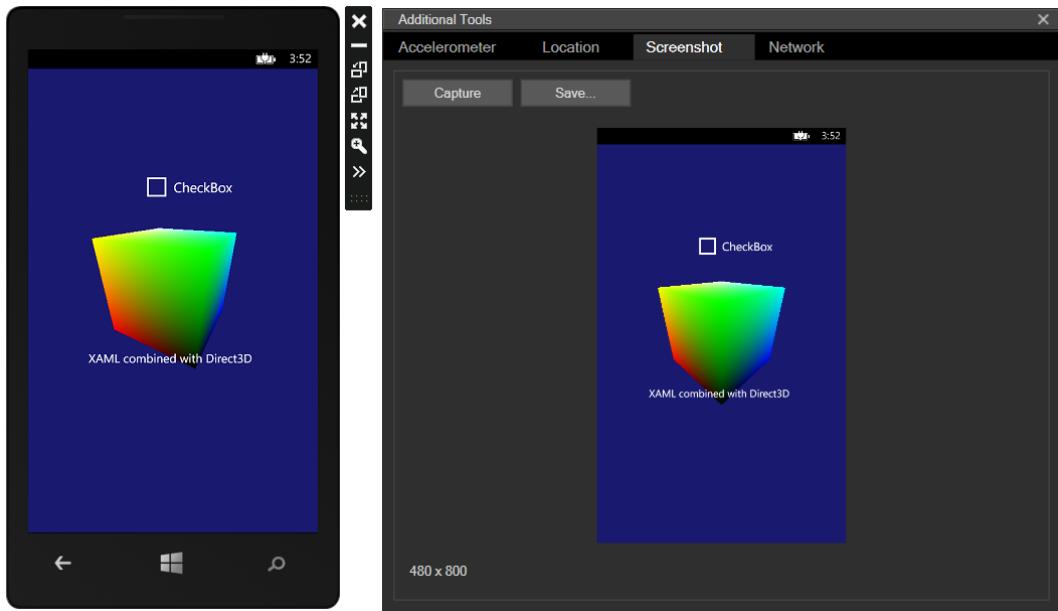
# Network Simulation

- Simulation Dashboard also includes Network Simulation
- Test your app on the emulator under different simulated network conditions



# Capturing Screenshots

- It is very useful to be able to take screenshots of the emulator
- These can be used in documentation and also to promote the application in the Marketplace
- The emulator Additional Tools pane provides a Screenshot tab

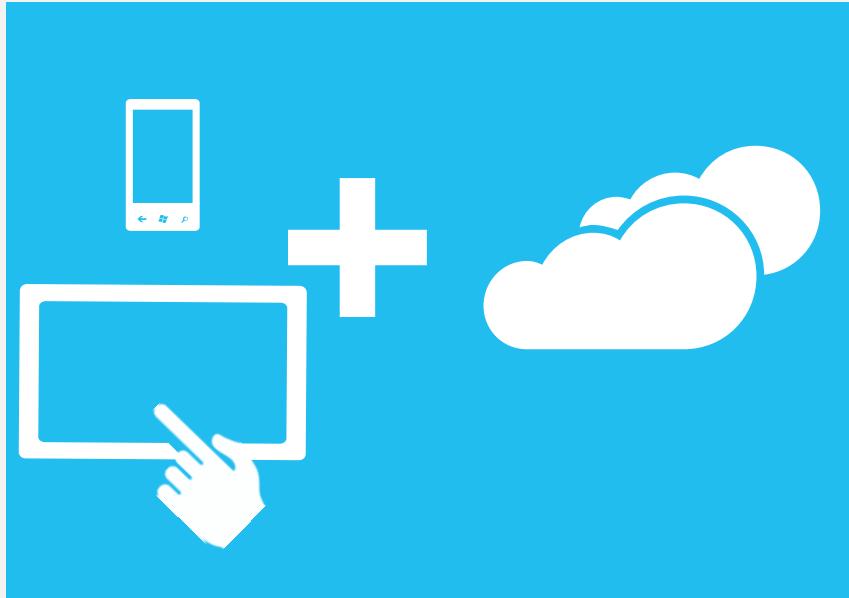


# Capturing Screenshots on a Real Device

- New in Windows Phone 8!
- Press the power and Windows buttons simultaneously
- Easy to share screenshots on social networks



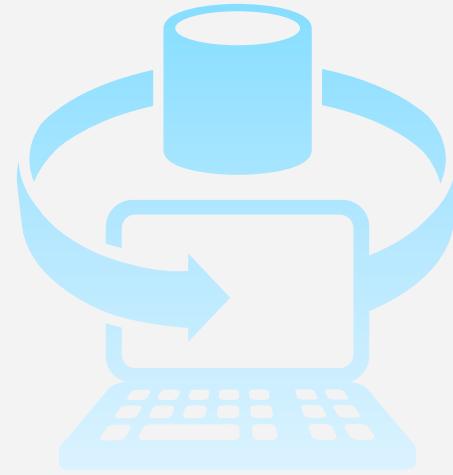
# Three Reasons for Device + Cloud



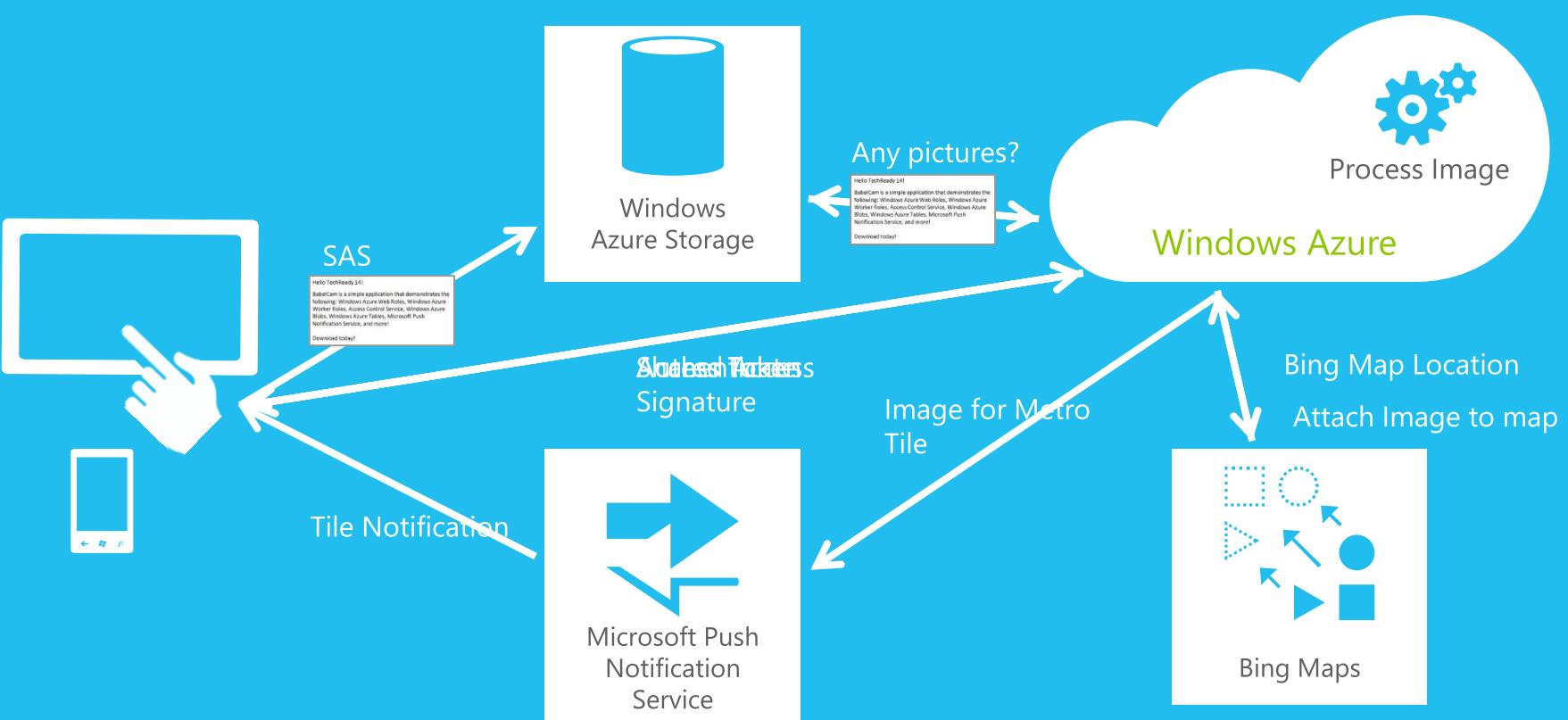
- 1 • Allows new application scenarios
- 2 • The cloud levels the playing field
- 3 • The cloud provides a way to reach across device platforms and a larger pool of resources from which to pull

# Why Windows Azure?

- PaaS: you build it, Windows Azure runs it
- Automatic O/S patching
- Elasticity and scale
- Utility billing
- Higher-level services
  - ACS, Caching, CDN, Traffic Manager
- <http://www.windowsazure.com/education>



# Send Notification



# Notifications: Different services

Windows 8:

Windows Push Notification Service (**WNS**)

Windows Phone:

Microsoft Push Notification Service (**MPNS**)

iOS:

Apple Push Notification Service (**APNS**)

Android:

Cloud To Device Messaging (**C2DM**)

# Windows Azure Toolkits for Devices

- Easier for device developers to use Windows Azure

Windows Phone



<http://bit.ly/watwp7>

iOS



<http://bit.ly/watios>

Android



<http://bit.ly/watandroid>

Windows 8



<http://bit.ly/Lcthn1>



big data



database



storage



traffic

## application building blocks



caching



messaging



identity



media



CDN



networking

# Summary

Windows Phone 8 has very many new features

The shared core means that at a low level, components of the OS are the same between Windows 8 and Windows Phone 8

Windows Phone 8 supports XAML+managed, XNA+managed, Direct3D+native app development, and hybrids of those

Apps built for Windows Phone OS 7.1 in general run without modification or recompilation on Windows Phone 8



The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be

interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.

MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.