MVA Microsoft Virtual Academy

Game Development in Unity3D for Windows Phone 8.1

Microsoft

Elissa Tong | Technical Evangelist
elistong@microsoft.com

Agenda

01 | Setup Development Environment

02 | Create a Unity Game

03 | Hardware Modifications

04 | Understand API Changes

MVA Microsoft Virtual Academy

01 | Setup Development Environment

Microsoft

System Requirements

Unity Engine	• Unity 4.2+ supports Windows Store Apps & Windows Phone 8.0 • Unity 4.5.3+ supports Universal Apps & Windows Phone 8.1 • Tip: always use the most recent Unity builds
Windows PC	• Windows 8.x Professional+, 64 bit • Microsoft Visual Studio Professional, Premium, Ultimate 2013 • Update 2+ supports Universal Apps • Microsoft Visual Studio Express 2013 • Windows Phone devices
Mac OS X	• Boot Camp 5+ • Windows 8.x Professional & up 64-bit ISO • Windows 7 USB/DVD download tool • Parallels 9+

System Requirements Resources

Windows PC

Unity:

- unity3d.com/unity/download

Windows OS:

- http://windows.microsoft.com/en-us/windows/downloads

Visual Studio IDE:

- http://www.visualstudio.com/downloads/download-visual-studio-vs

Mac OS X

Boot Camp:

- microsoftstore.com/store/msusa/html/pbPage.Help_Win7_usbdvd_dwnTool
- http://windows.microsoft.com/en-us/windows-8/install-windows-on-mac
- help.apple.com/bootcamp/mac/5.0/help
- digitaltrends.com/computing/how-to-install-windows-8-on-a-mac/

Parallels:

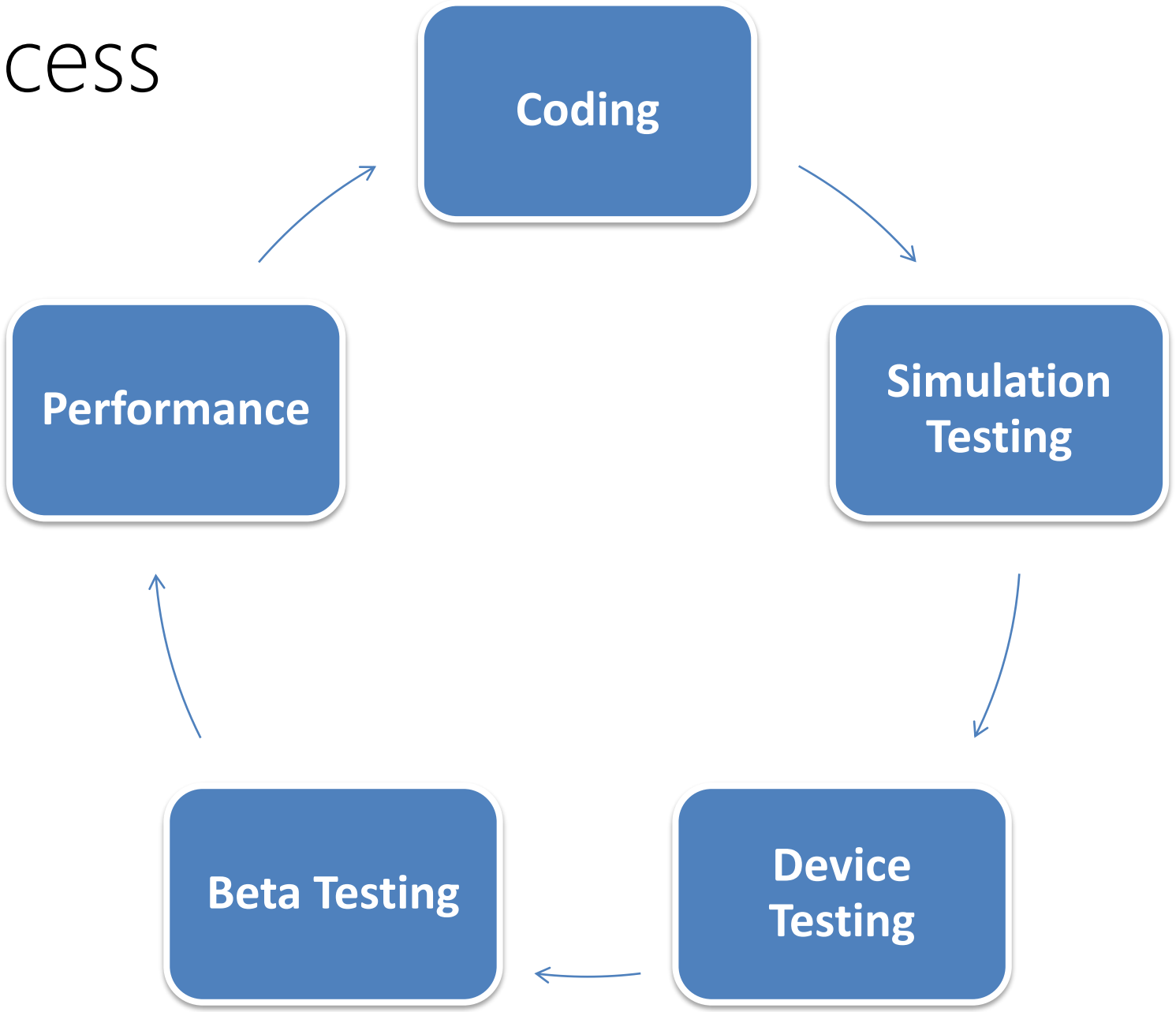
- download.parallels.com/desktop/v9/ga/docs/en_US/Parallels%20Desktop%20User's%20Guide/

MVA Microsoft Virtual Academy

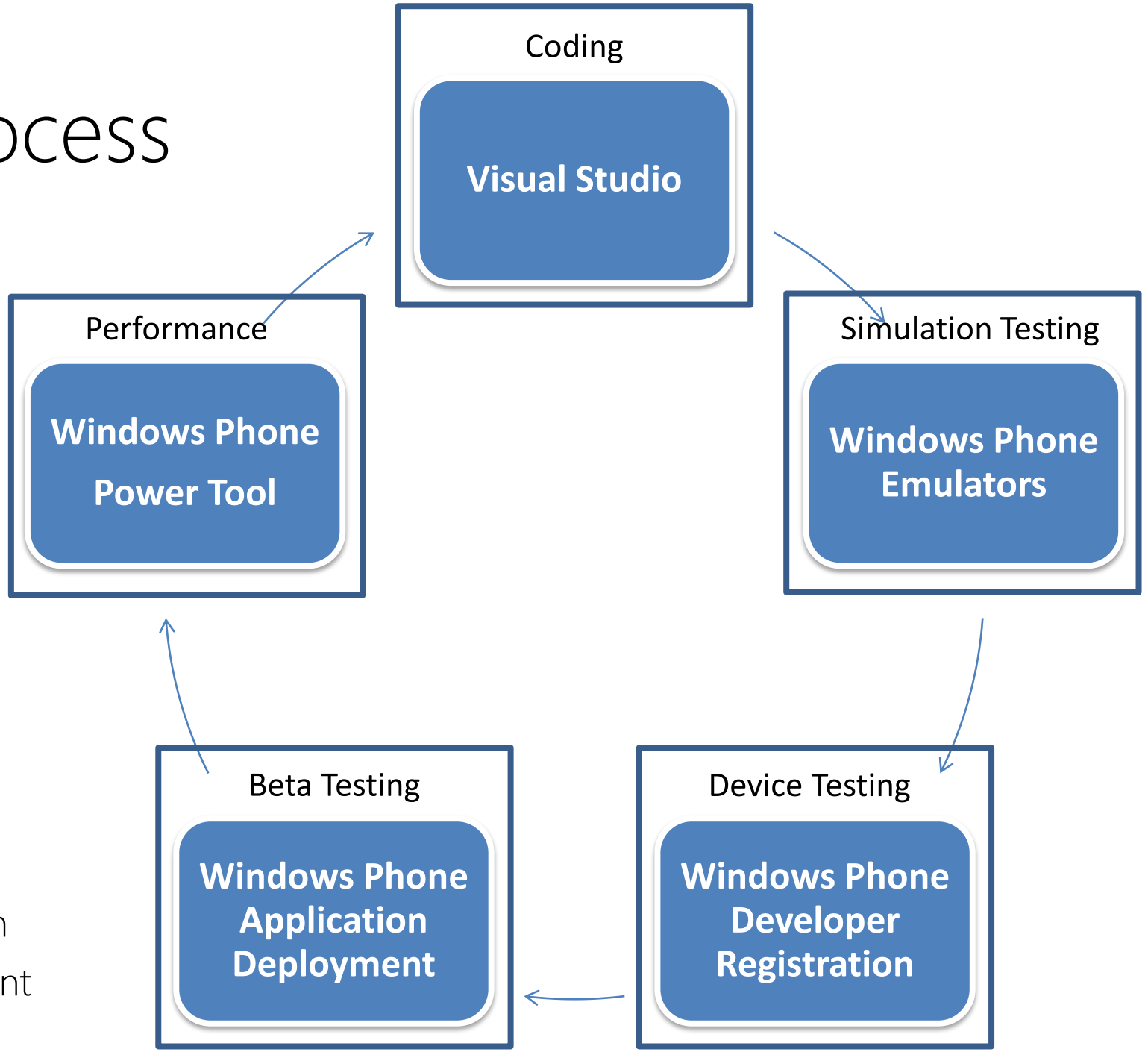
01 | Development Tools

Microsoft

Mobile Game Development Process



Mobile Game Development Process



Windows Phone Tools:

Windows Phone

Windows Phone Emulator

Hyper-V Manager

Dev Center App for WP8

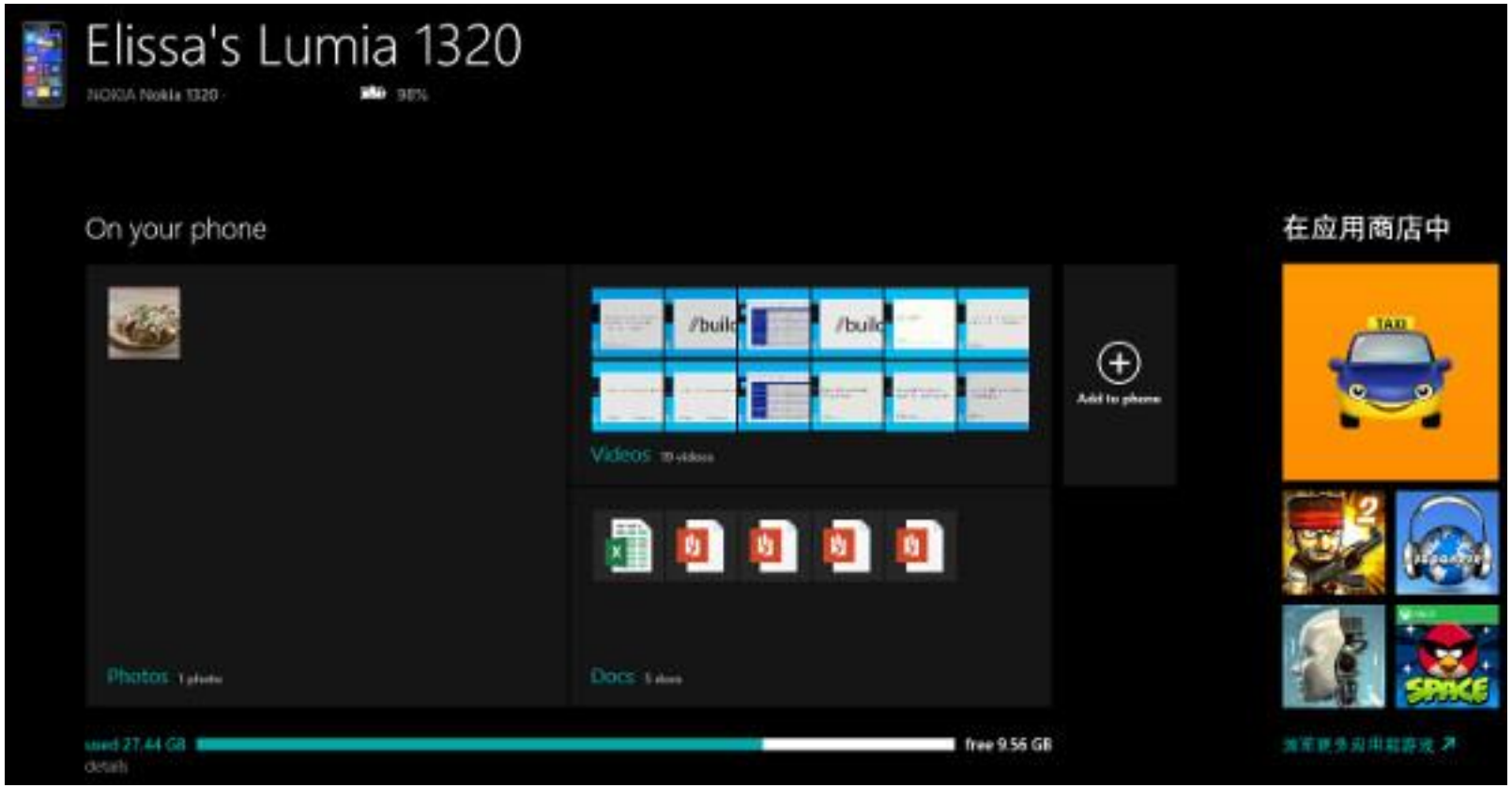
Windows Phone Power Tools

Windows Phone Developer Registration

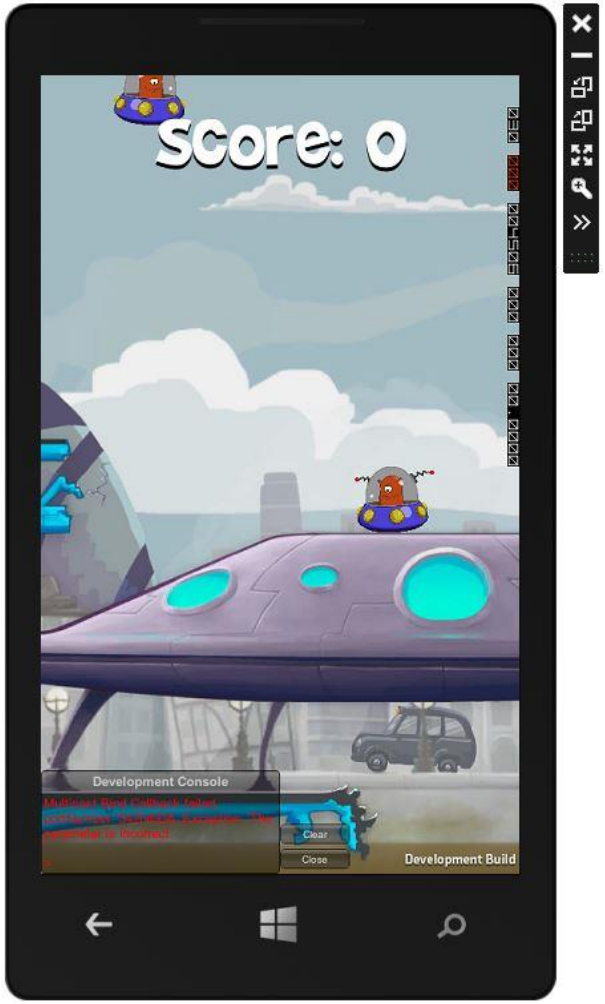
Windows Phone Application Deployment

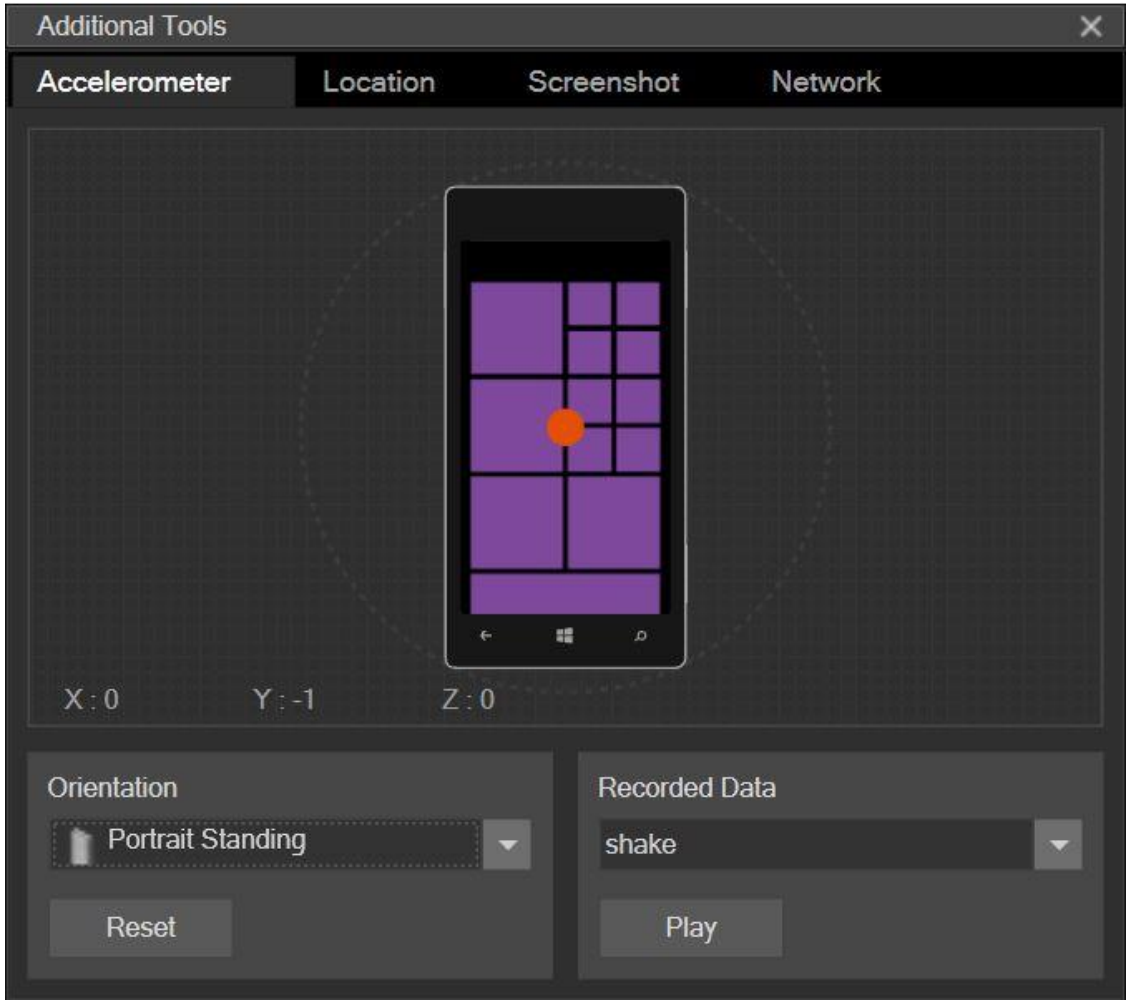
Windows Phone Tool

- Connect your phone to computer's USB port
- Upload/Download photos, videos, music, and documents



Emulators (8.0)

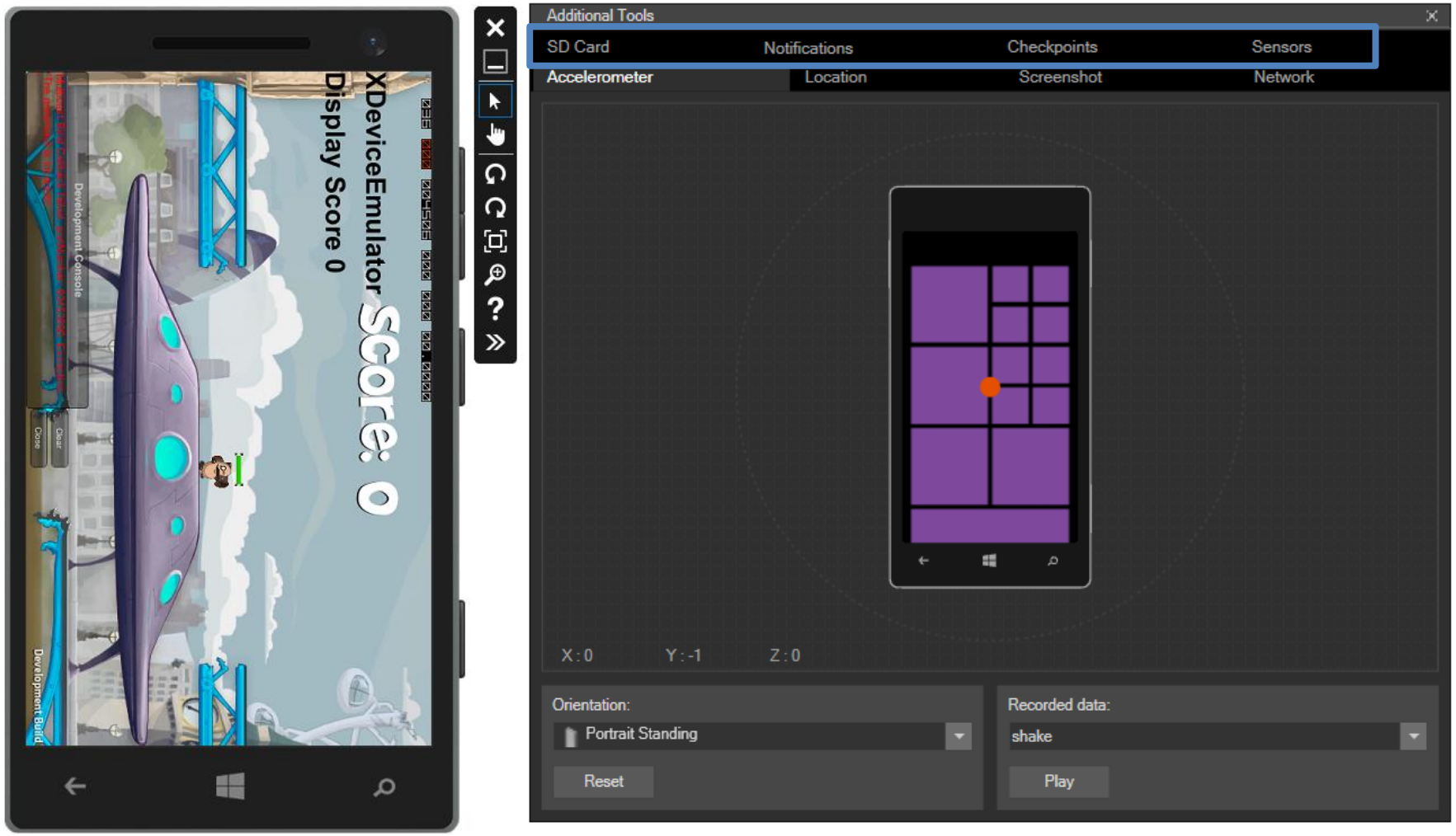




Functionality:

- Accelerometer
- Location
- Screenshot
- Network

Emulator (8.1)



Added Functionalities:

- SD Card
- Notifications
- Checkpoints
- Sensors

http://msdn.microsoft.com/en-us/library/windows/apps/dn629629.aspx

Frame Rate Counters

Sample Code

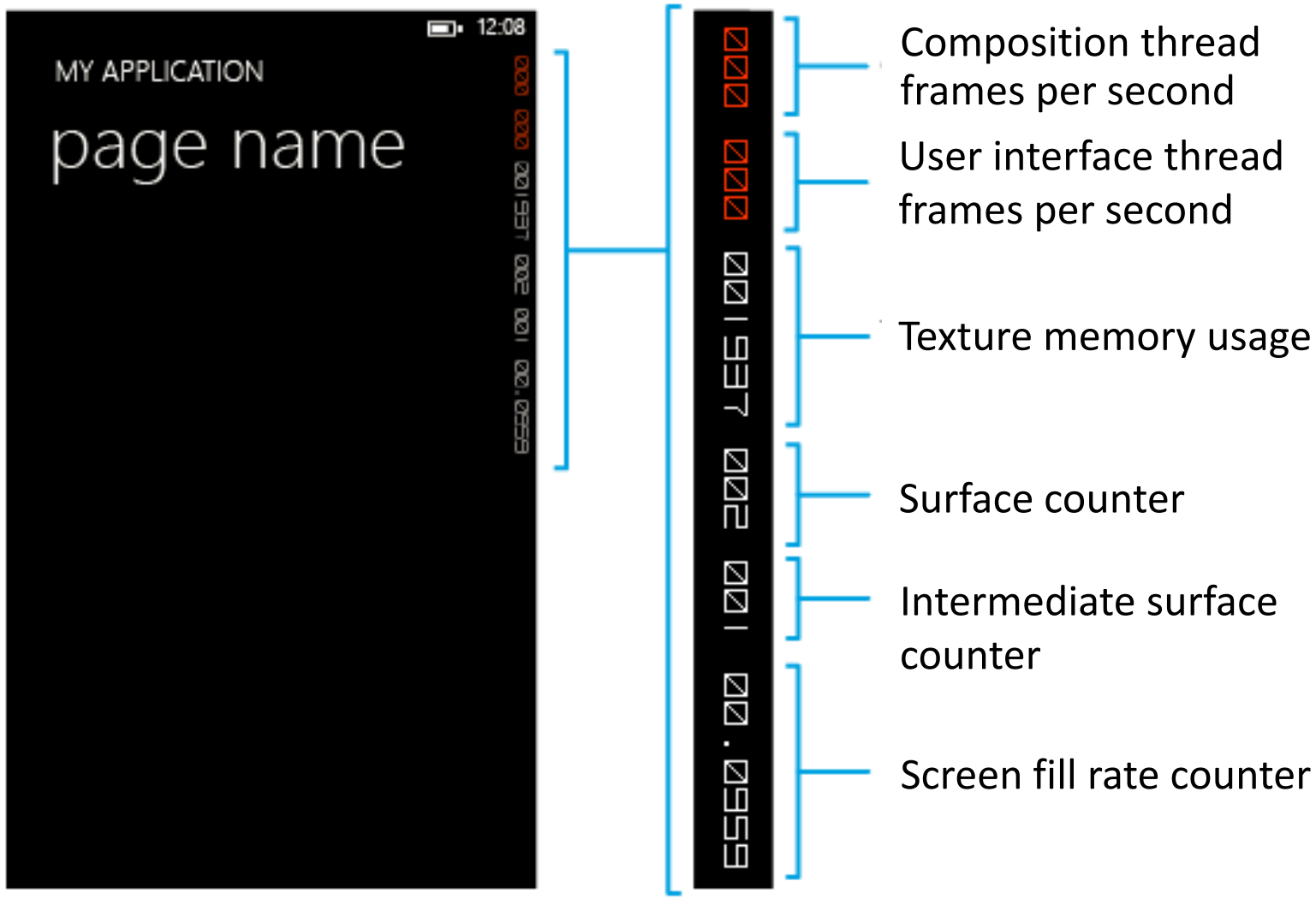
1

App.xaml.cs:

```
Application.Current.Host.Settings.E
nableFrameRateCounter = true;
```

2

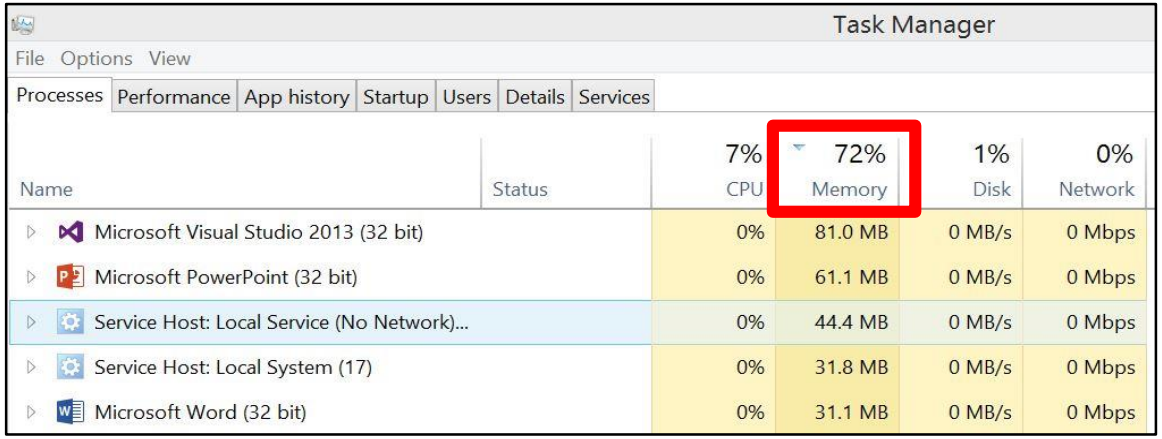
```
// ((1024 * 1024) = 1048576
string memUsage =
(Microsoft.Phone.Info.DeviceStatus.
ApplicationCurrentMemoryUsage / 104
8576.0f).ToString() + " MB";

string peakMemUsage = (Microsoft.Ph
one.Info.DeviceStatus.ApplicationPe
akMemoryUsage / 1048576.0f).ToStrin
g() + " MB";
```



http://msdn.microsoft.com/library/windows/apps/gg588380(v=vs.105).aspx

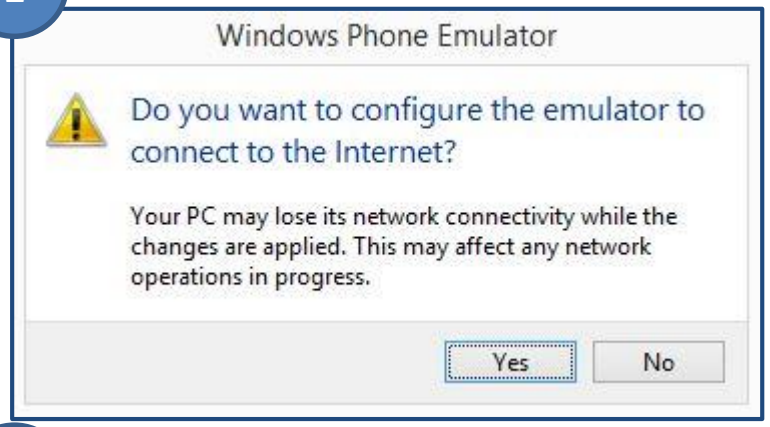
Emulator: Memory Issues

- Solution: End any running process & software not in use
- http://support.microsoft.com/kb/2911380/en-us

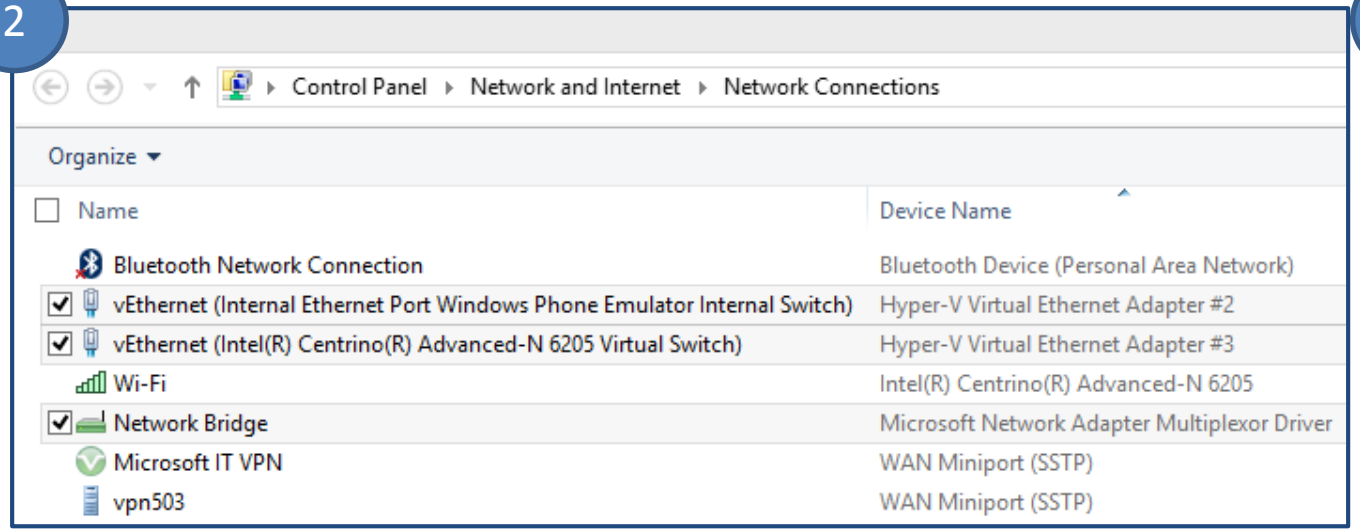


Emulator: Internet Issues

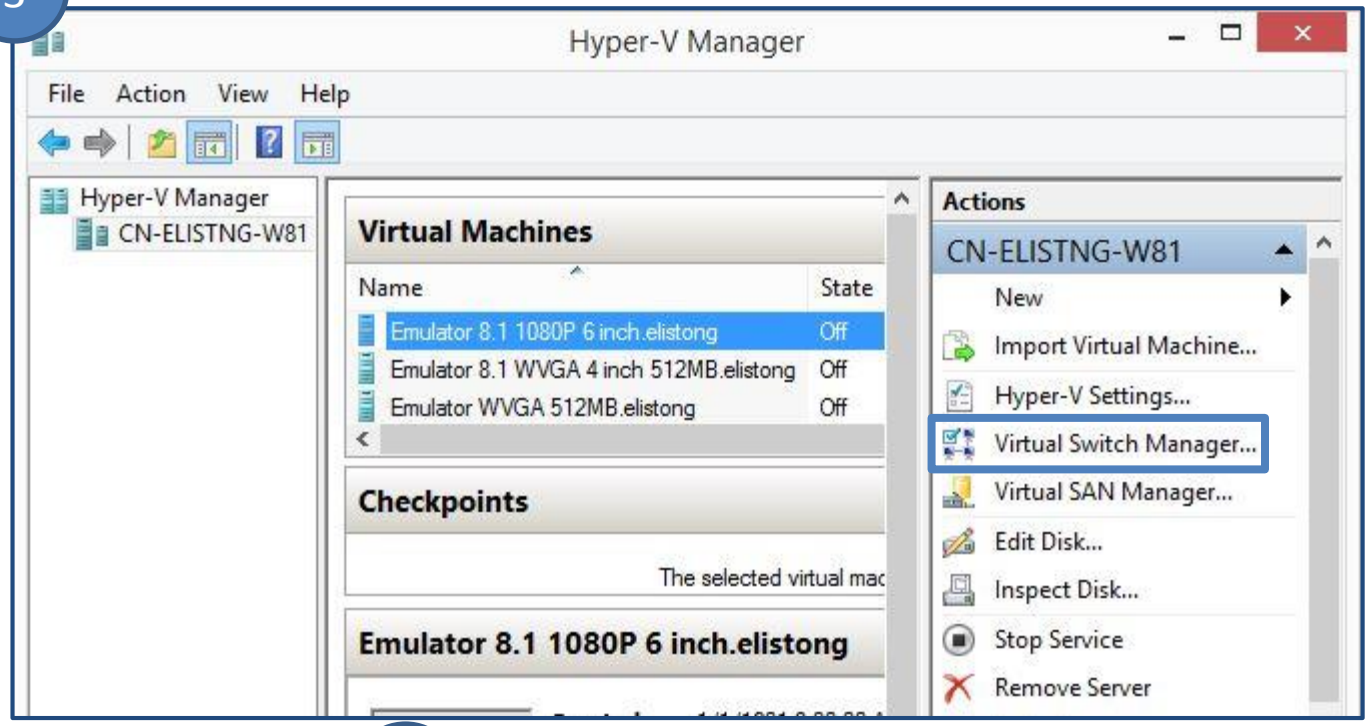
1



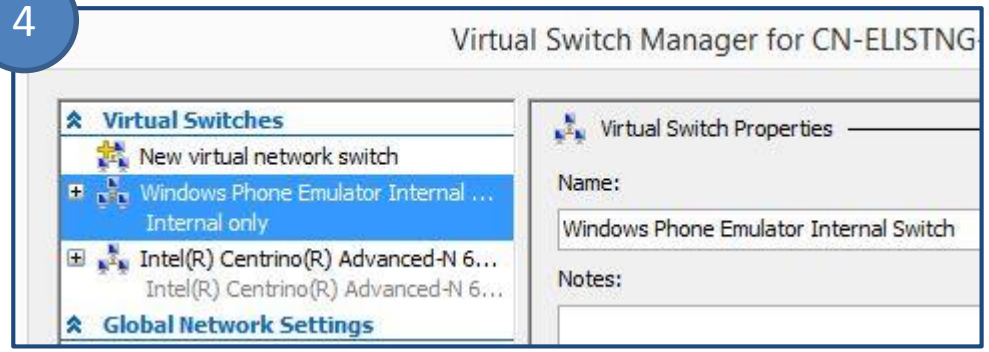
2



3

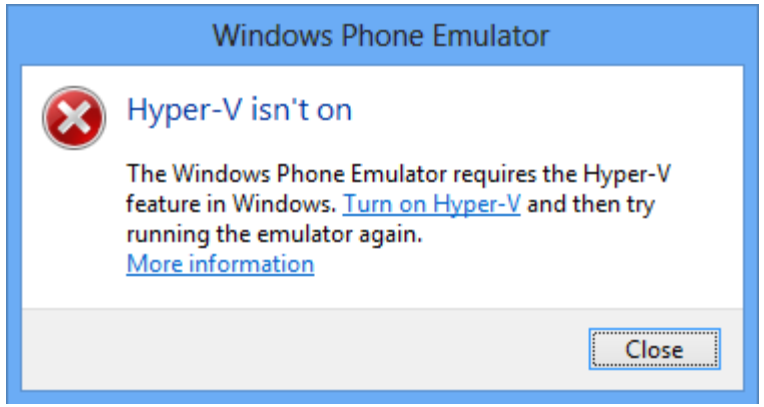


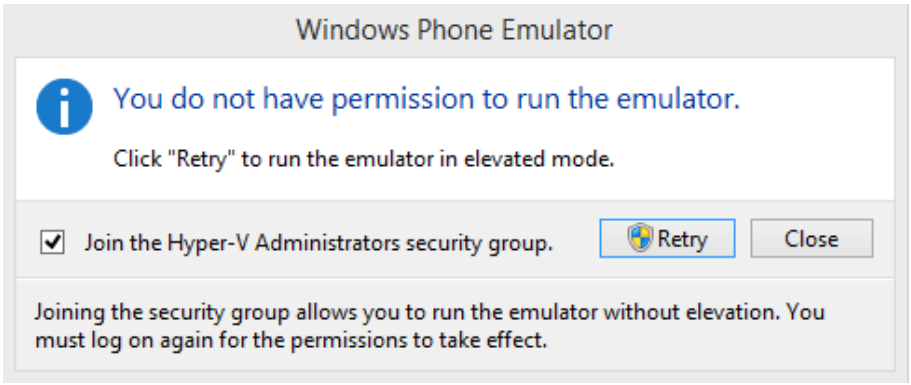
4



Emulator: Virtualization Issues

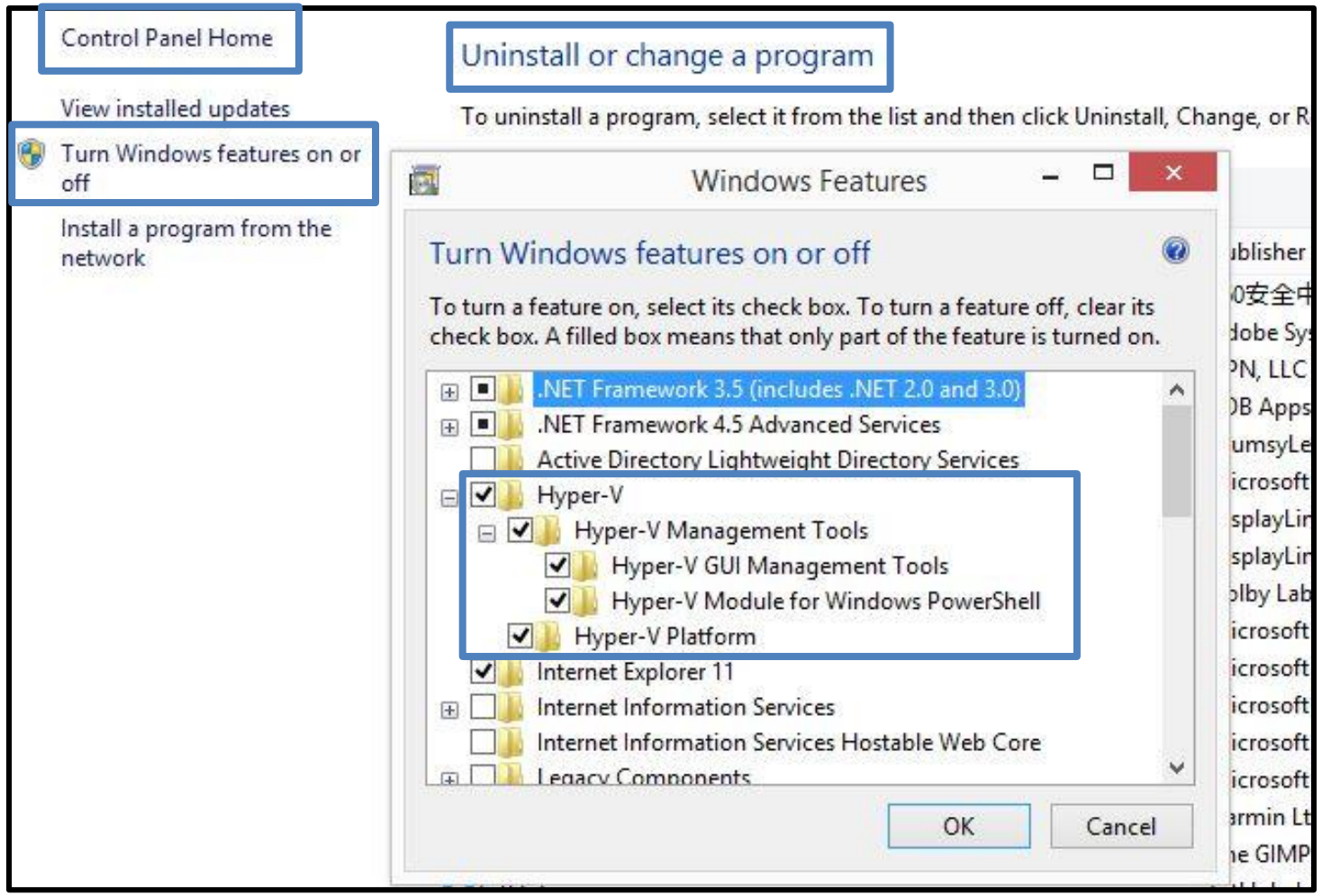
- Related to hardware virtualization
- http://msdn.microsoft.com/en-us/library/windows/apps/jj863509(v=vs.105).aspx





Hyper-V

- Step 1: Turn on Hyper-V in Windows features

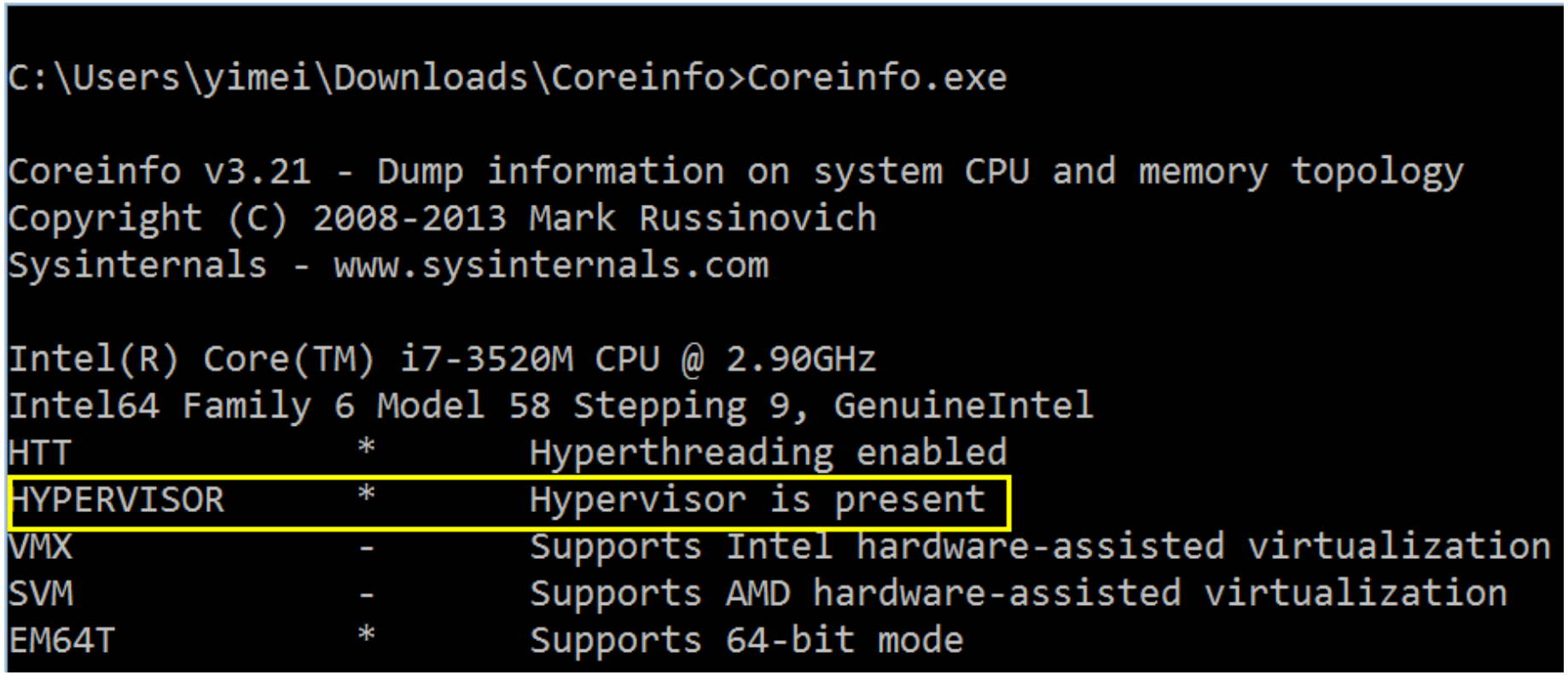


Windows Phone Emulator uses Hyper-V Virtualization Technology

Hyper-V

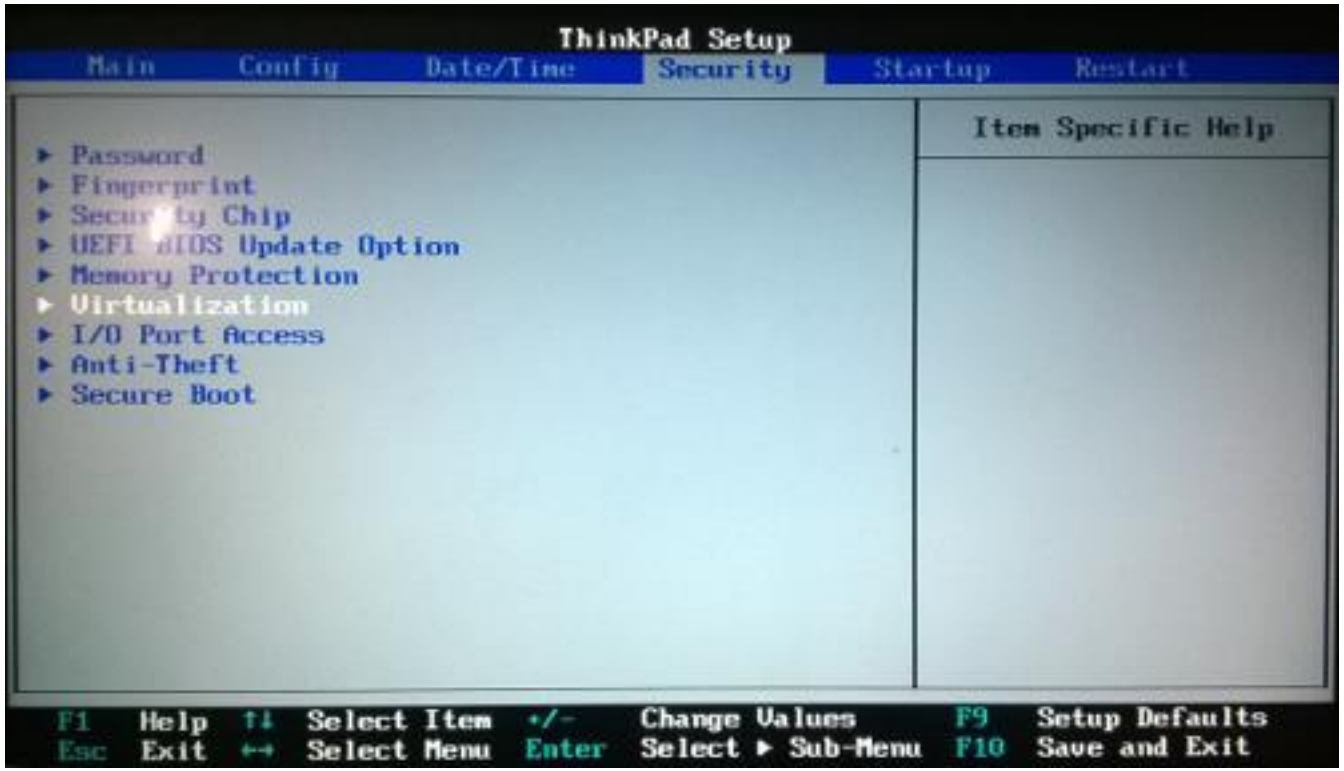
- Step 2: Does your computer support Virtualization ?
- http://technet.microsoft.com/enus/sysinternals/cc835722.aspx

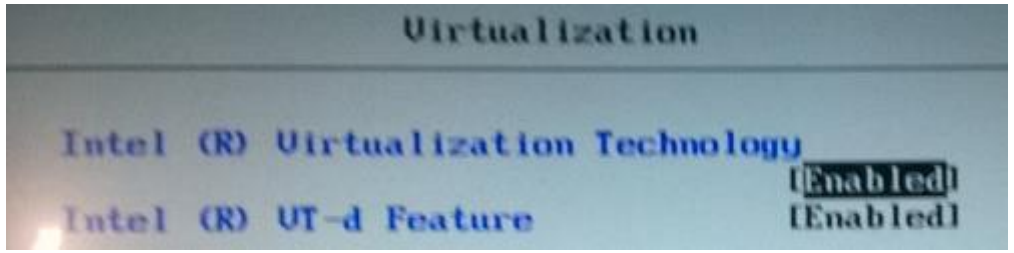
Download Coreinfo.exe



Hyper-V

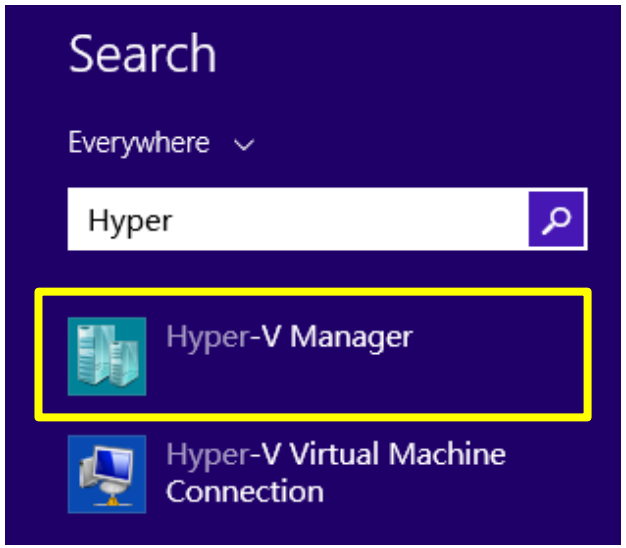
- Step 3: Check if Virtualization is enabled in your BIOS settings

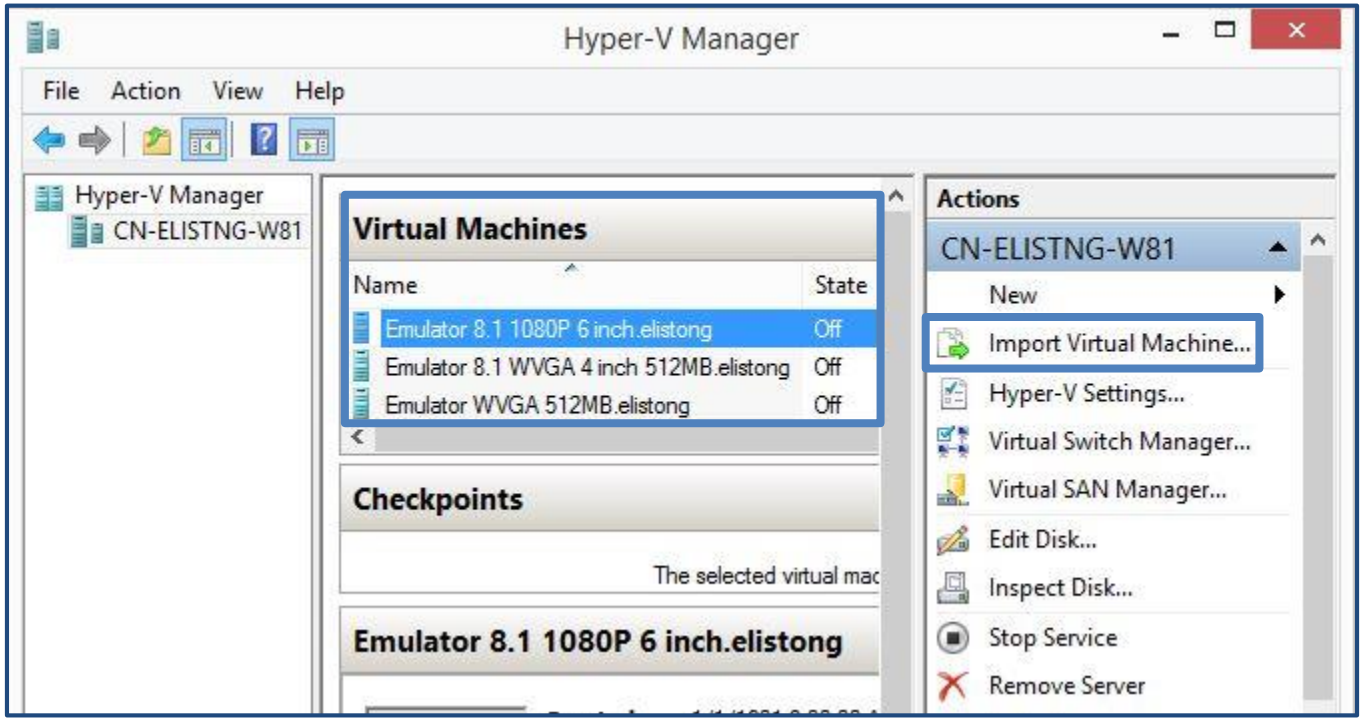


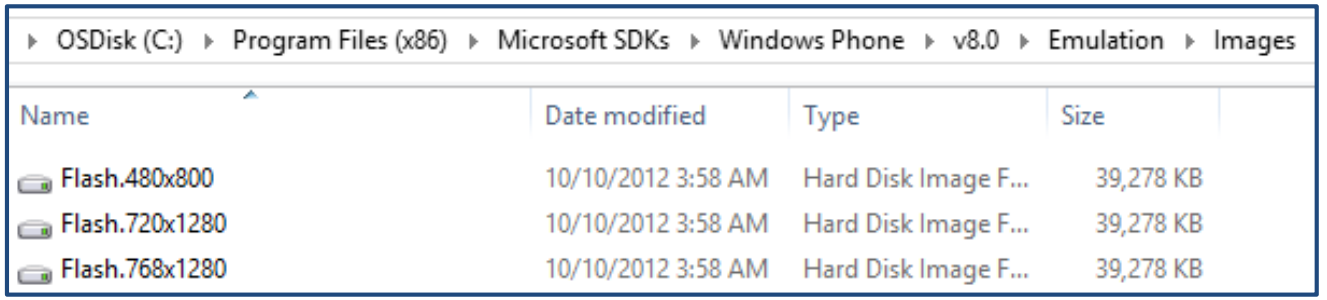


Hyper-V

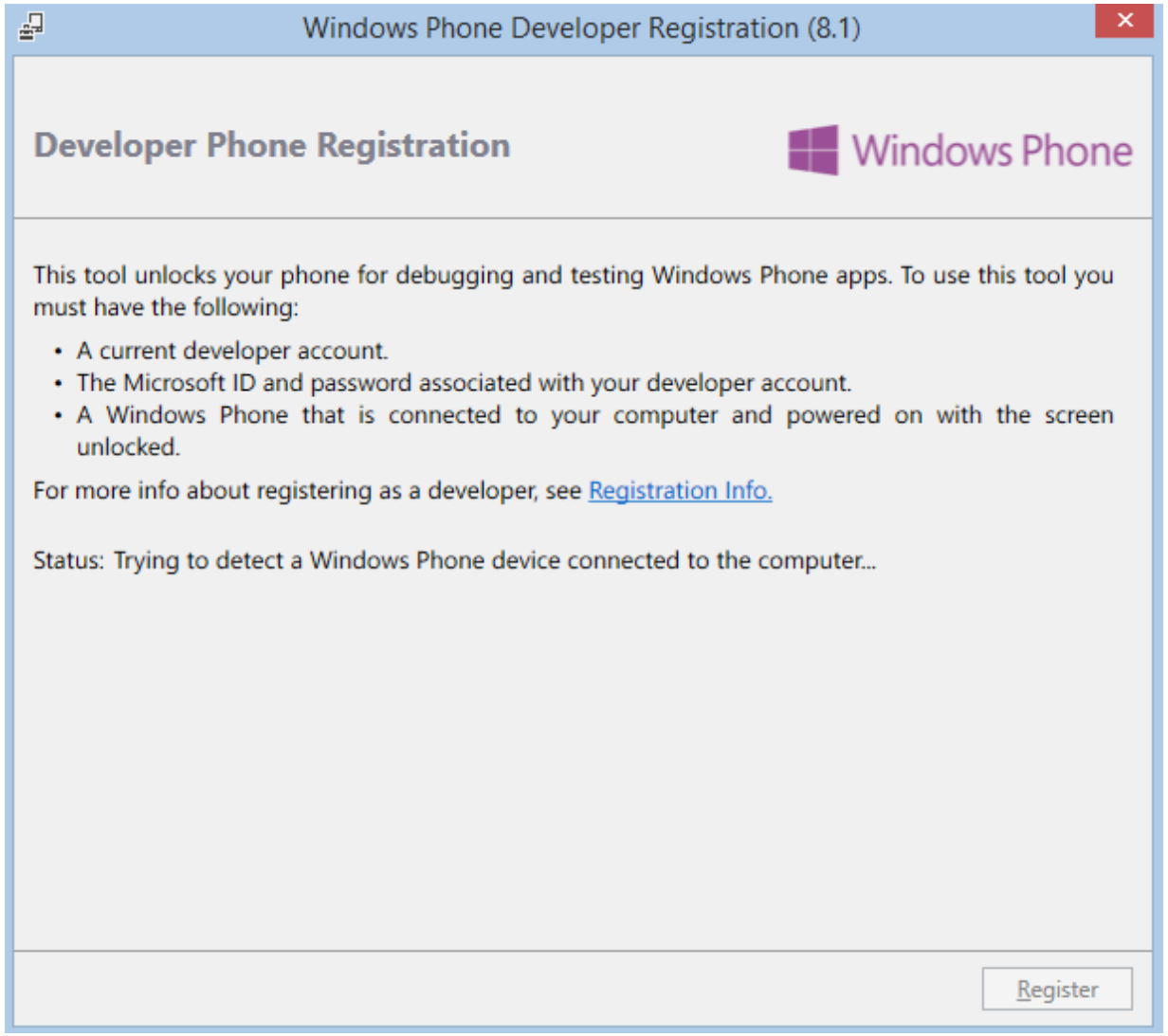
- Step 4: Hyper-V Manager







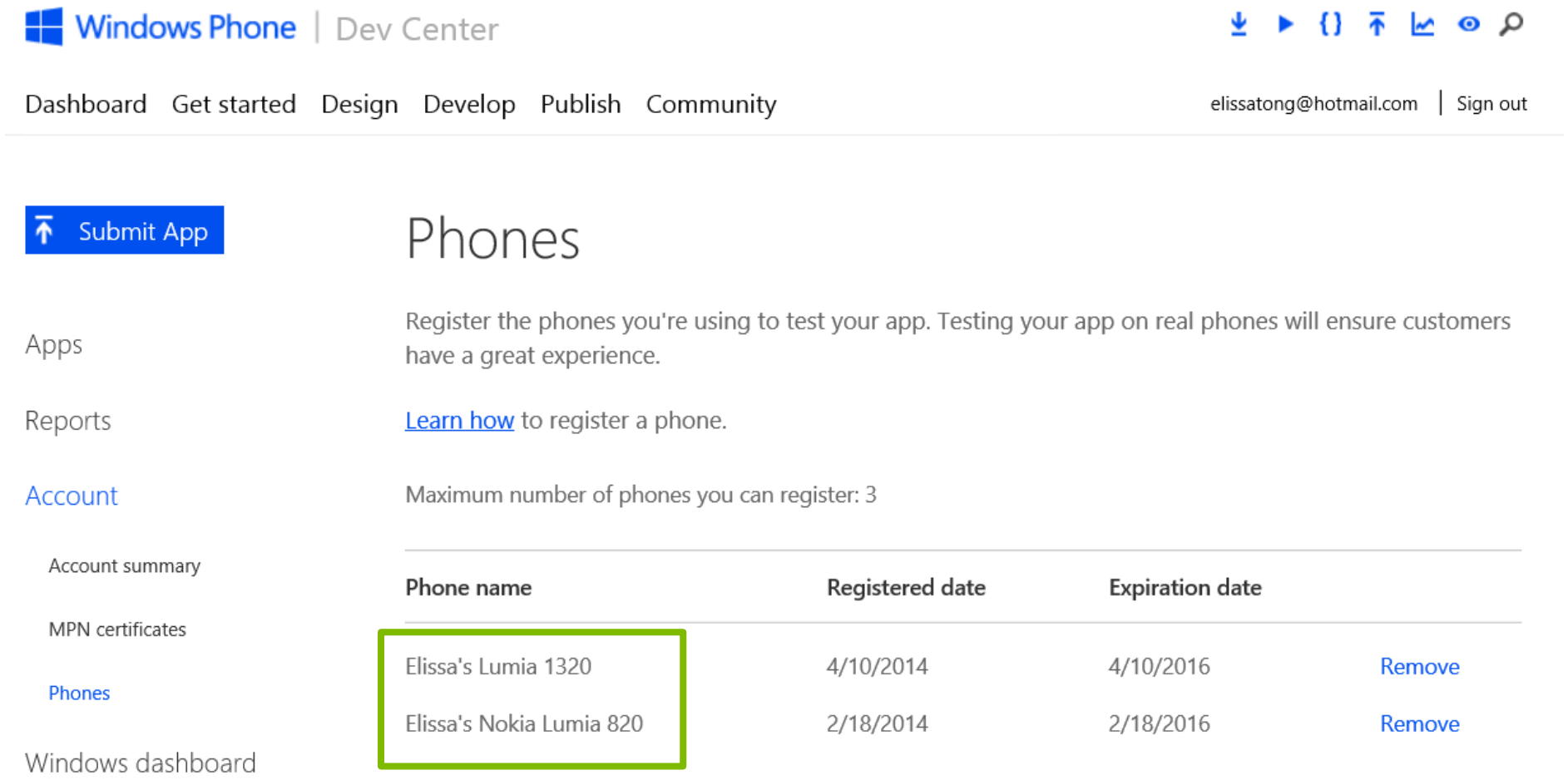
Developer Phone Registration



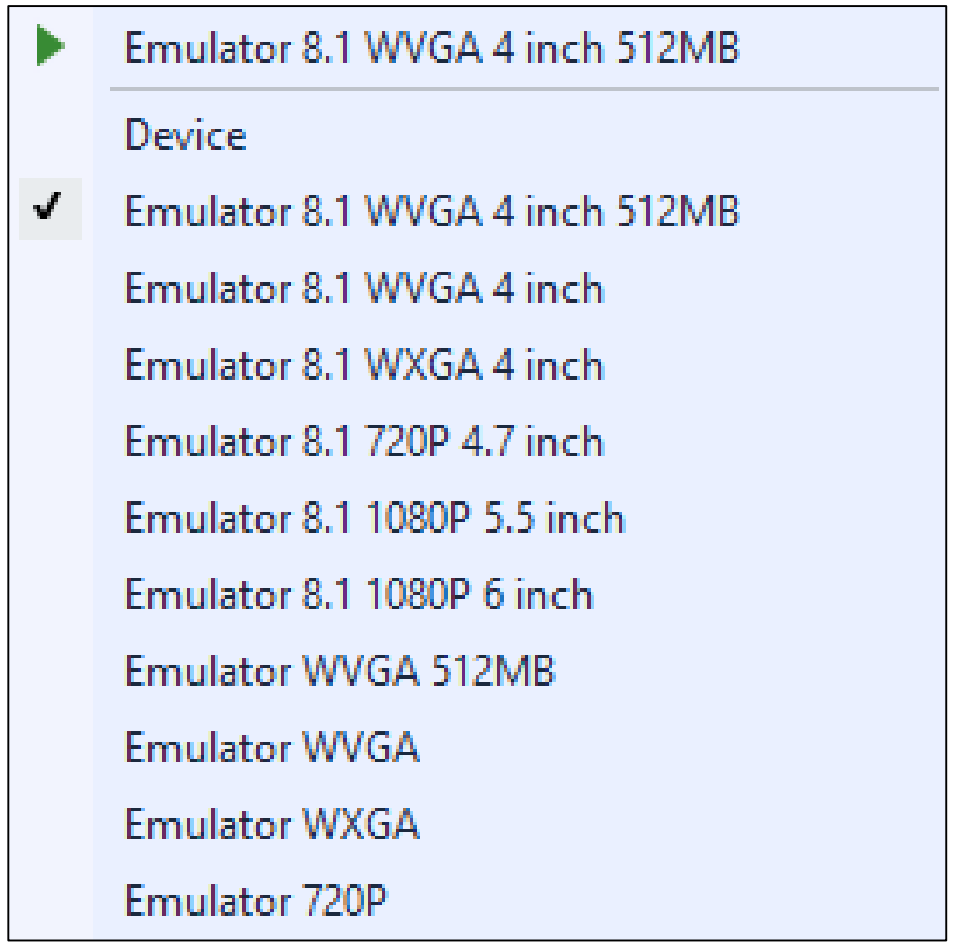
- Test on a device!
- Firstly, register device for development
- Possible issues:
  http://irisclasson.com/2013/10/28/problems-and-fixes-when-registering-a-windows-phone-8-device-to-deploydebug-applications

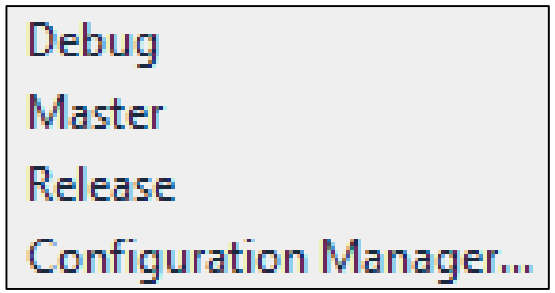
Developer Phone Registration

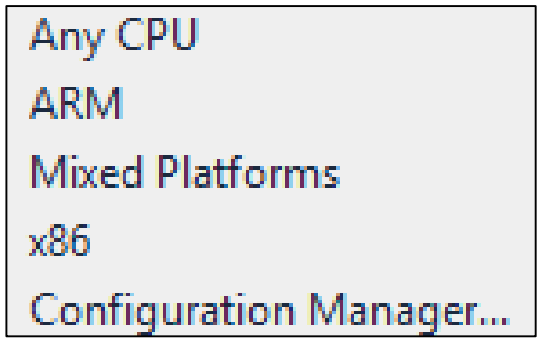
- https://dev.windowsphone.com/en-us/Account/Devices

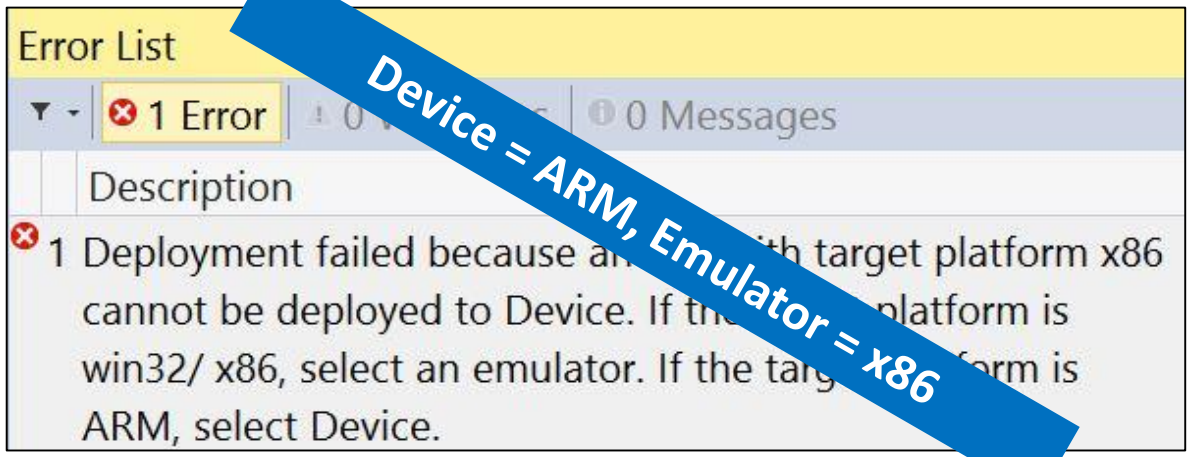


Device Testing

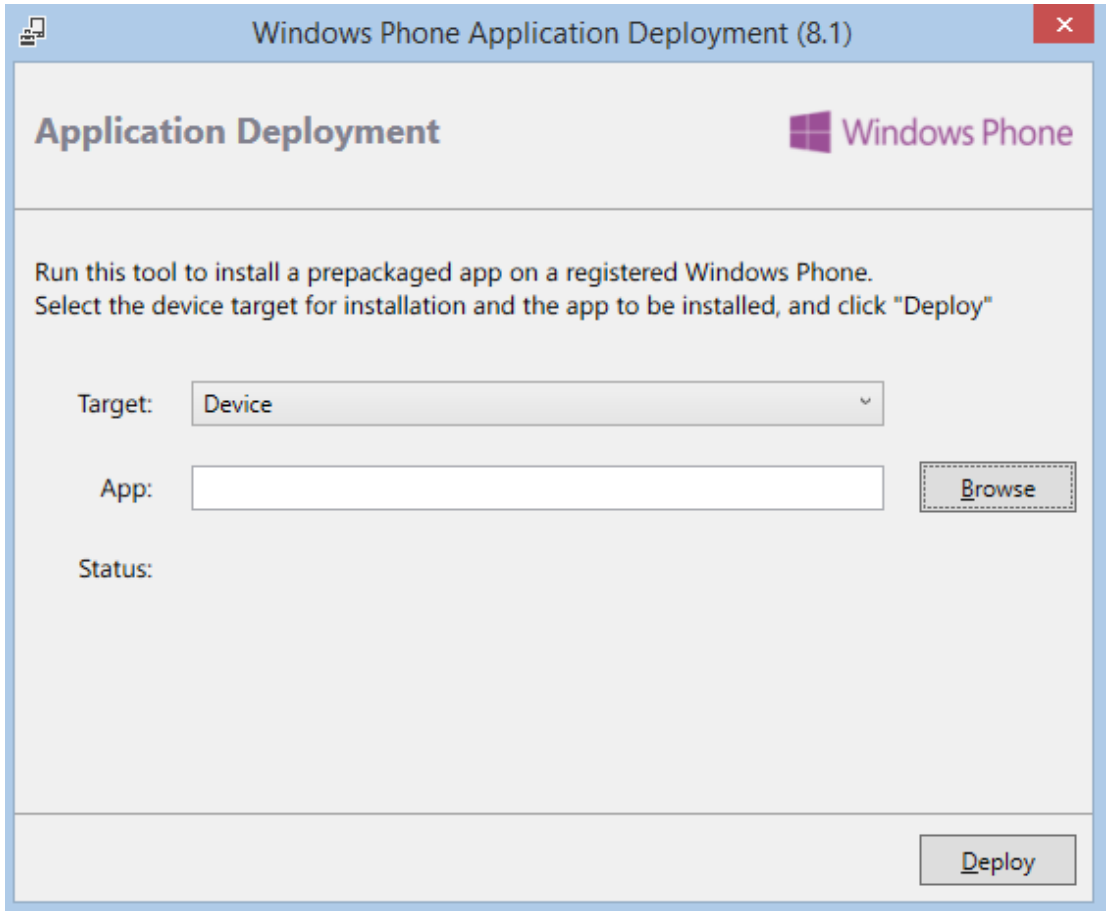








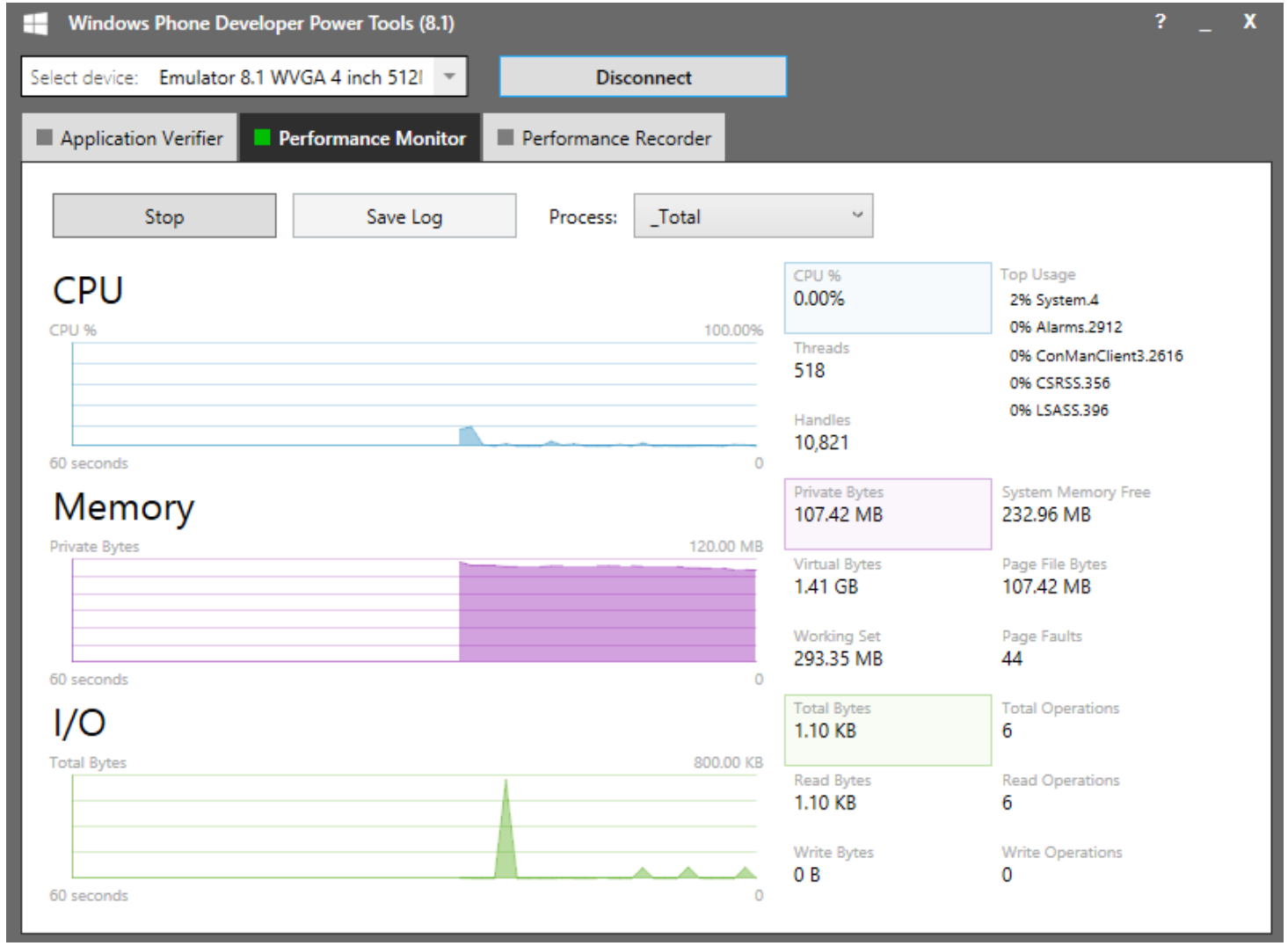
Application Deployment



Instructions

1. Select your Target
2. Browse to Windows Phone XAP
3. Click Deploy

Windows Phone Power Tools



http://msdn.microsoft.com/library/windows/apps/dn629255.aspx

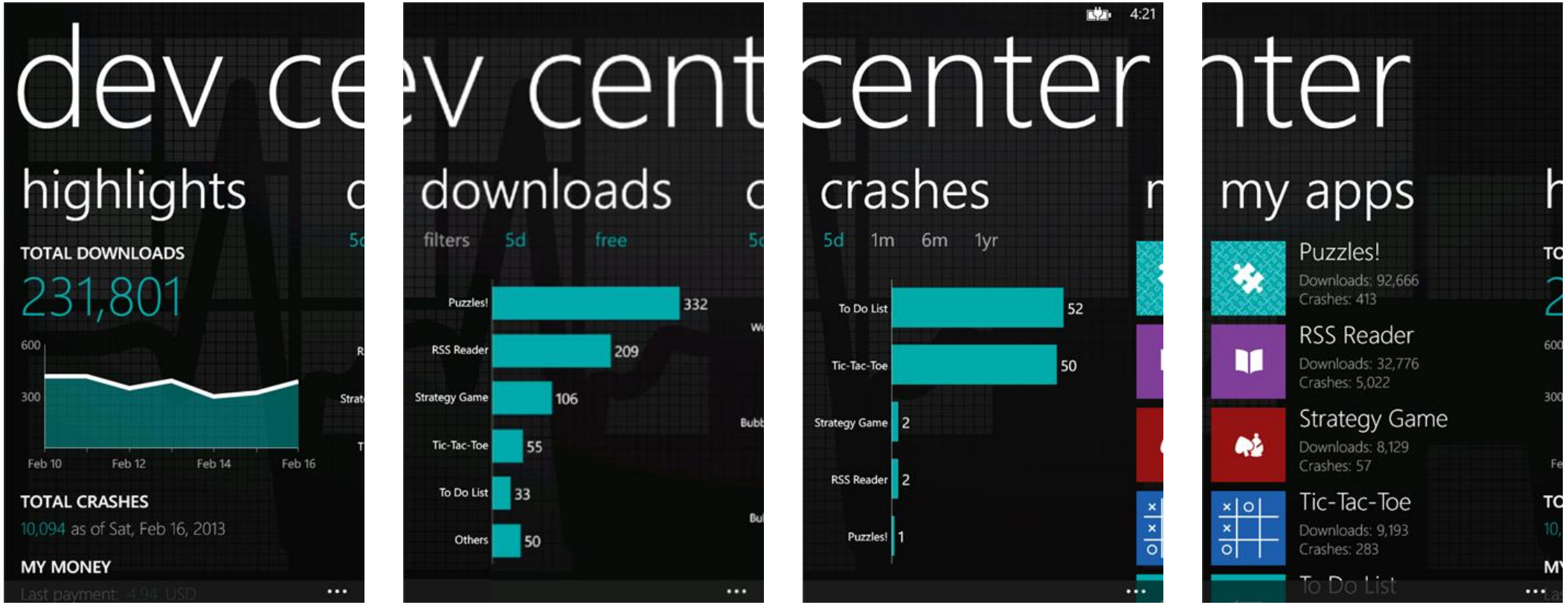
Application Verifier
Detect subtle programming errors in native code: memory leaks, exceptions…

Performance Monitor
Capture real-time performance metrics and visualize them graphically.

Performance Recorder
Collect system-wide logs and analyze them on your computer.

Dev Center App

- Install Dev Center from the Windows Phone Store
- Functionalities: App info, downloads, crashes, user reviews, etc.



prime[31] Unity Plugins

- For Windows Store & Windows Phone 8 plugins
  - IAP
  - Trial
  - Social
  - Ads
  - Live Tiles
  - Push Notifications
  - Flurry Analytics
  - & more!

Additional Tools

Visual Studio 2013 Tools for Unity

http://visualstudiogallery.msdn.microsoft.com/20b80b8c-659b-45ef-96c1-437828fe7cf2

Visual Studio Productivity Power Tools 2013

http://visualstudiogallery.msdn.microsoft.com/dbcb8670-889e-4a54-a226-a48a15e4cace

Prime31 Unity Plugins

https://prime31.com/

Dev Center App

http://www.windowsphone.com/zh-cn/store/app/dev-center/2d3063c2-4b29-4e69-9c03-50b67b0e6aec

Protobuf-net for WP8 .NET

http://code.google.com/p/protobuf-net/

Setup Summary

Windows Store Apps	Windows Phone 8.0	Windows Phone 8.1	Universal Apps	Windows 8
Supports Mac OS X	Boot Camp or Parallels	Visual Studio	Windows Phone Emulators	Virtualization Hyper-V
Phone Registration Tool	Windows Phone Power Tools	Dev Center App	Prime31 Unity Plugins	Visual Studio Tools for Unity

and more…

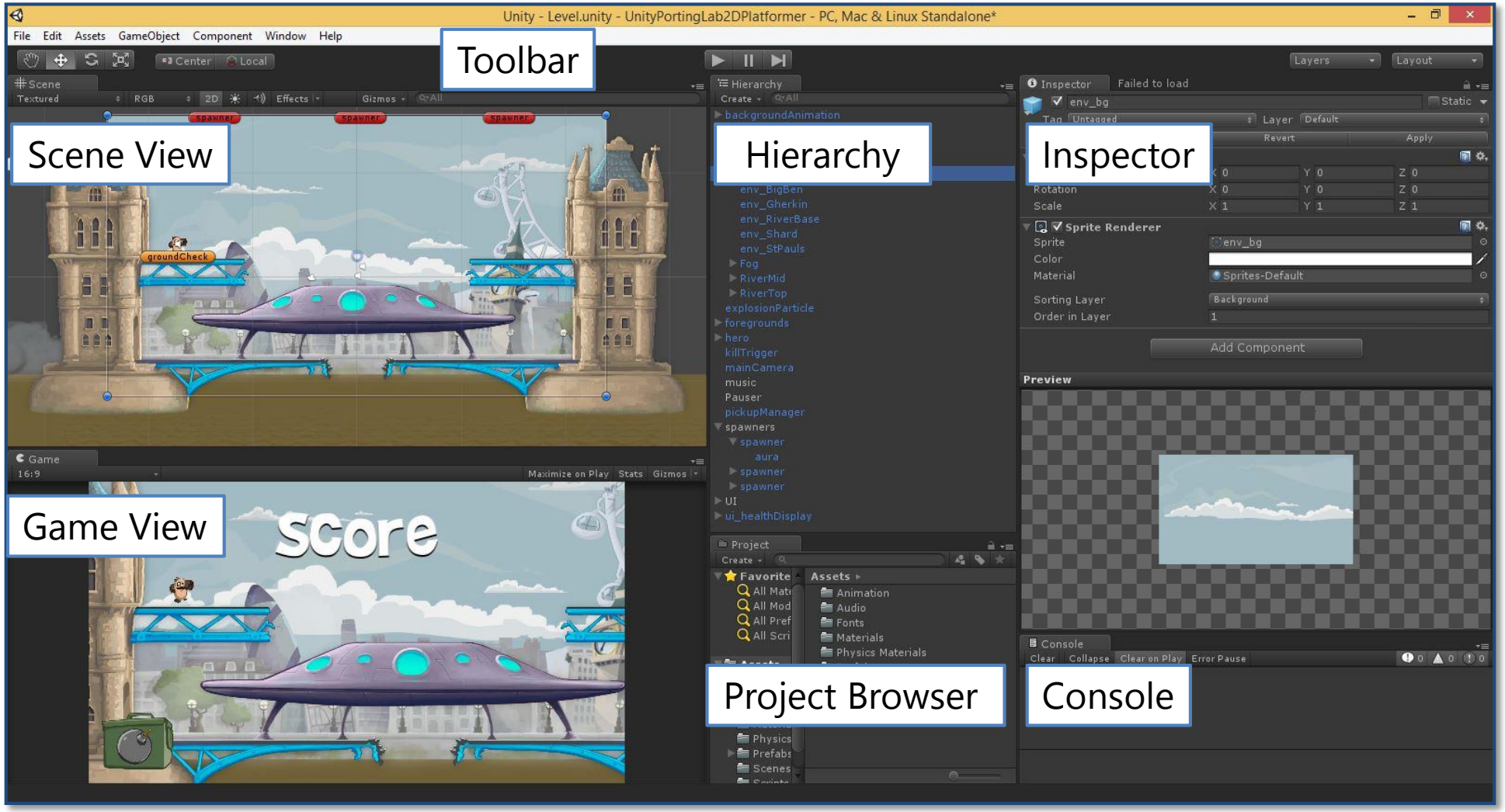
MVA Microsoft Virtual Academy

02 | Create a Unity Game

Microsoft

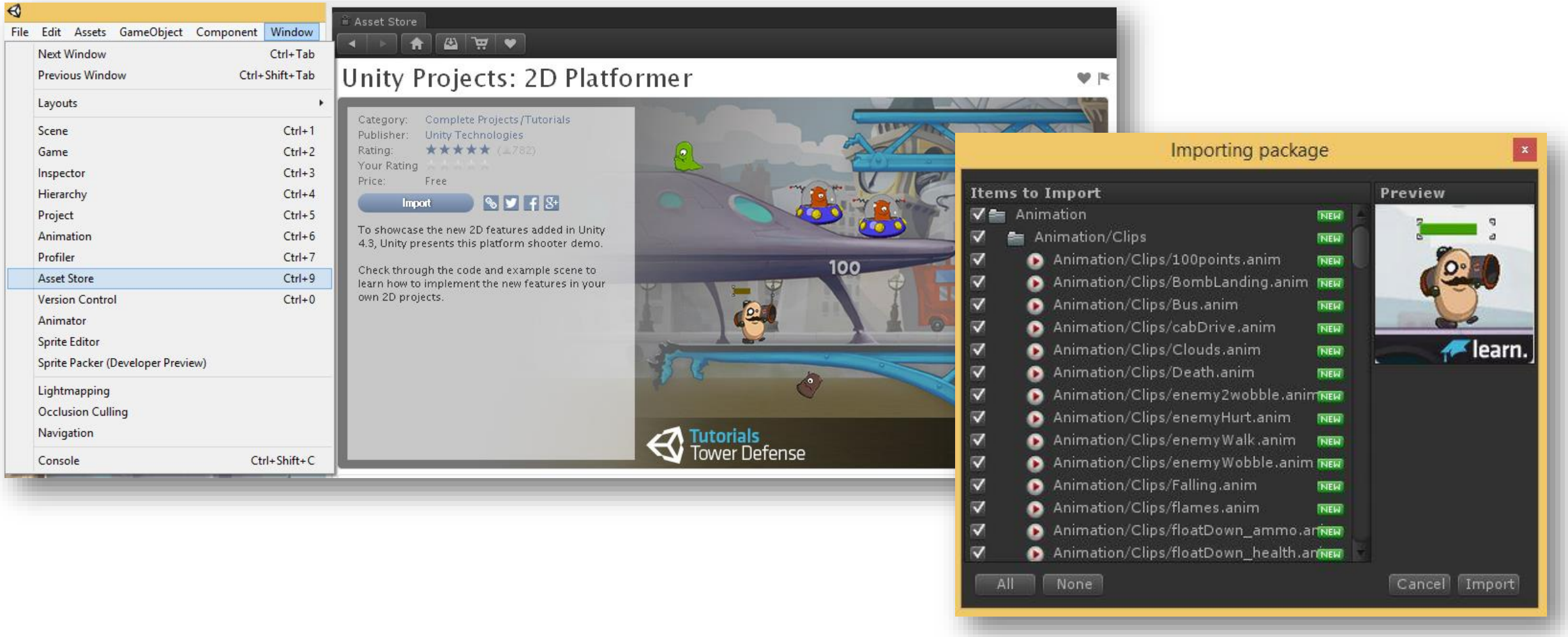
Unity3D Editor

- http://docs.unity3d.com/Manual/LearningtheInterface.html



Asset Store

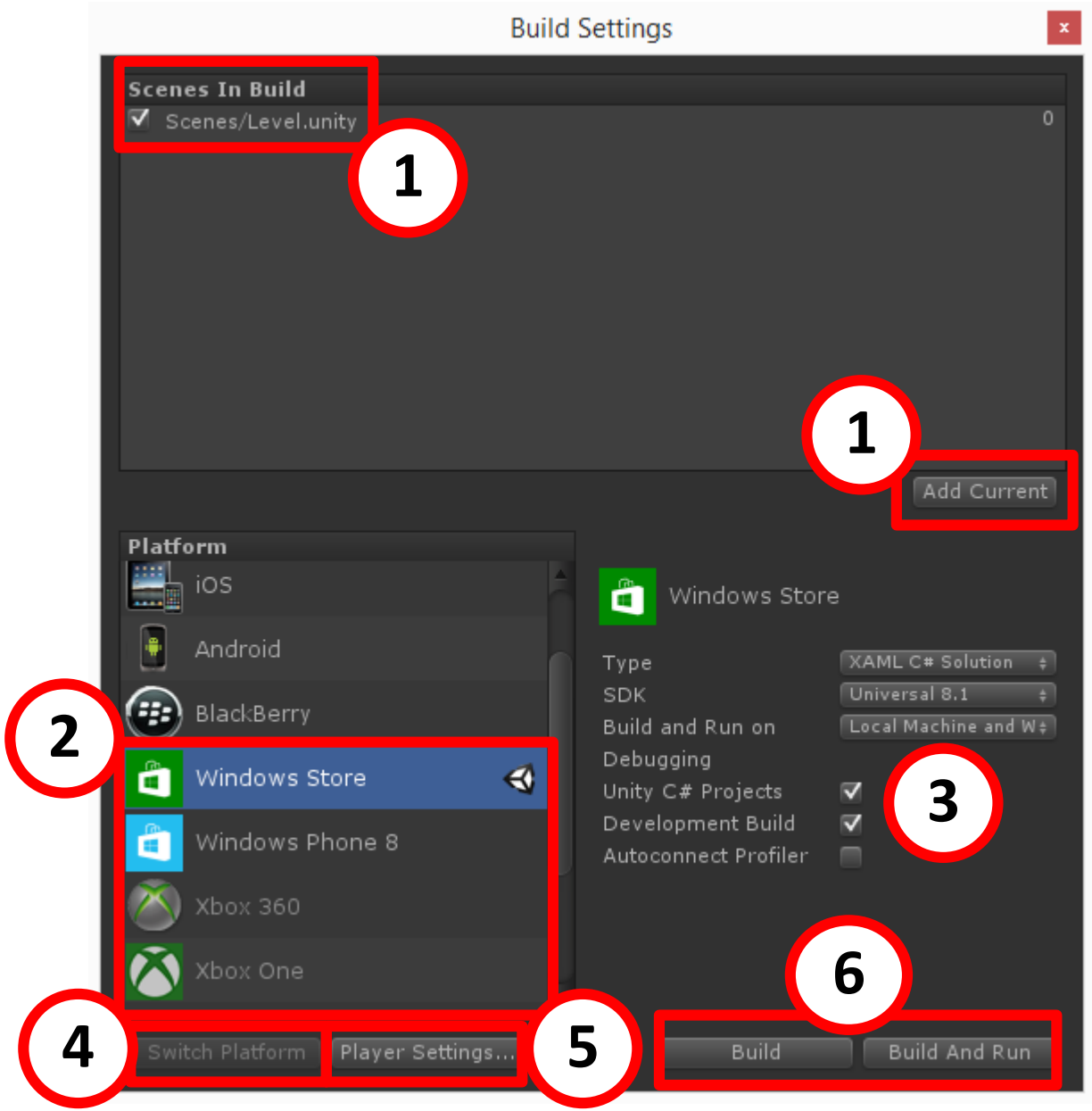
Download models, environments, and useful tools!



DEMO

Create a Unity Game

Build Settings



1. Add needed game **Scenes**
2. Select a platform
3. Change platform settings
4. Click **Switch Platform**
5. (Optional) Modify **Player Settings**
6. Click **Build** or **Build And Run**

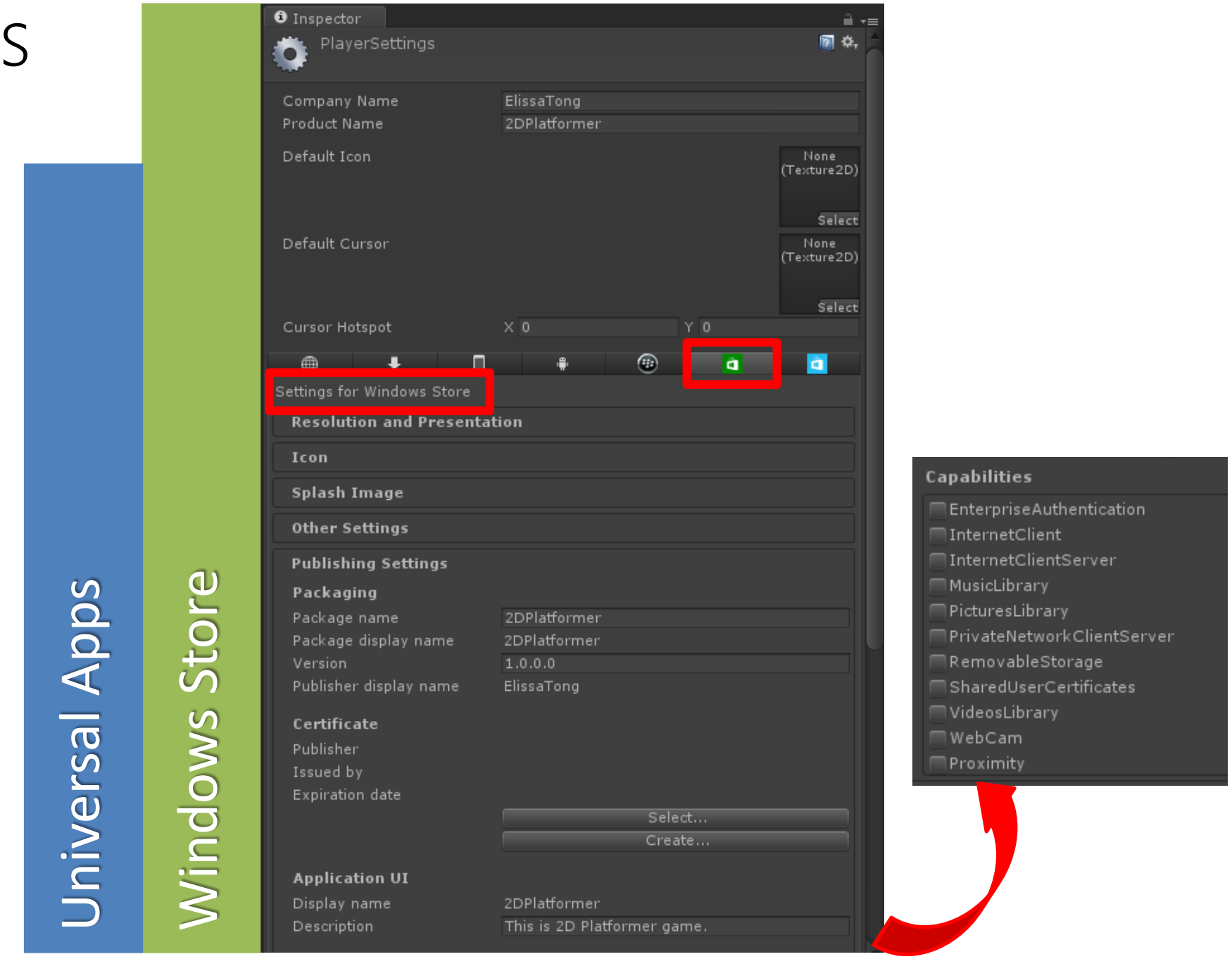
Player Settings

Player Settings

changes settings in

Package.appxmanifest

of your Visual Studio project



Create a Unity Game

Unity编辑

Asset Store

GameObject

Component

Prefabs

Build Settings

Player Settings

Visual Studio Builds & Deployment

and more…

MVA Microsoft Virtual Academy

03 | Hardware Modifications

Microsoft

Multiple Resolutions

- Windows Phone resolutions: WVGA, WXGA, 720p, 1080p
- Emulators available for each resolution
- http://msdn.microsoft.com/en-us/library/windows/apps/xaml/hh465349.aspx
- http://msdn.microsoft.com/en-us/library/windows/apps/jj206974(v=vs.105).aspx

Full Screen Size (Effective Pixel Resolution)	Device
384x640	4.5" device 15:9 aspect ratio
400x711	4.7" device 16:9 aspect ratio
450x800	5.5" device 16:9 aspect ratio
491x873	6" device 16:9 aspect ratio

Multiple Resolutions

What is your device screen width & height?

http://blogs.windows.com/buildingapps/2013/11/22/taking-advantage-of-large-screen-windows-phones/

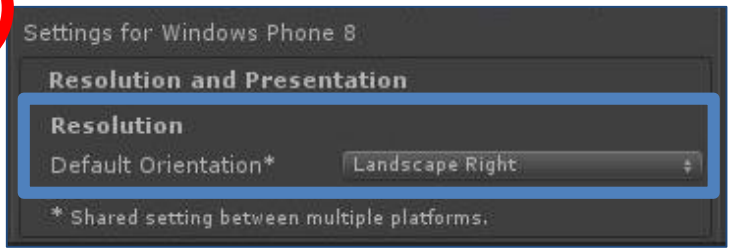
```
string GetBasicScreenInfo()
{
  var width = App.Current.Host.Content.ActualWidth;
  var height = App.Current.Host.Content.ActualHeight;
  var scaleFactor = (double)App.Current.Host.Content.ScaleFactor / 100d;
  return String.Format("{0} x {1}; {2:0.0} scale factor", width, height, scaleFactor);
}

string GetExtendedScreenInfo()
{
  object temp;
  if (!DeviceExtendedProperties.TryGetValue("PhysicalScreenResolution", out temp)) return "not available";

  var resolution = (Size)temp;
  if (!DeviceExtendedProperties.TryGetValue("RawDpiX", out temp) || (double)temp == 0d) return "not available";

  var dpi = (double)temp;
  var screenDiagonal = Math.Sqrt(Math.Pow(resolution.Width / dpi, 2) + Math.Pow(resolution.Height / dpi, 2));
  var width = App.Current.Host.Content.ActualWidth;

  return String.Format("{0} x {1}; {2:0.0#} raw scale; {3:0.0}"",
    resolution.Width, resolution.Height, resolution.Width / width, screenDiagonal);
}
```

Orientation

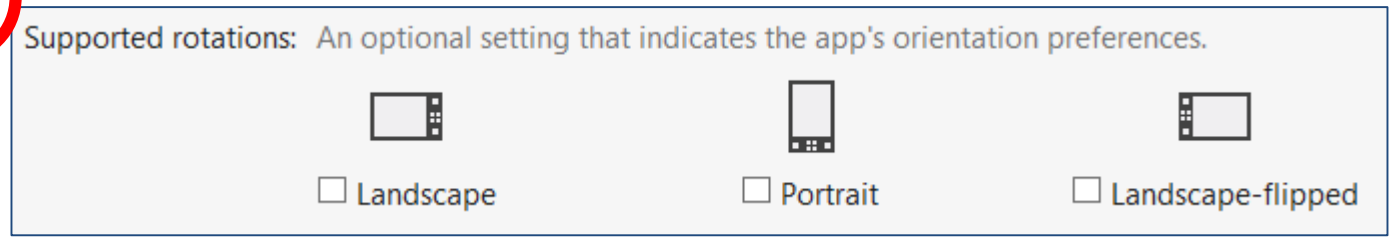
1. Windows Phone 8 Unity Player Settings
2. Package.appmanifest:
3. MainPage.xaml
4. Unity API

```
Screen.orientation =
ScreenOrientation.AutoRotation;
```

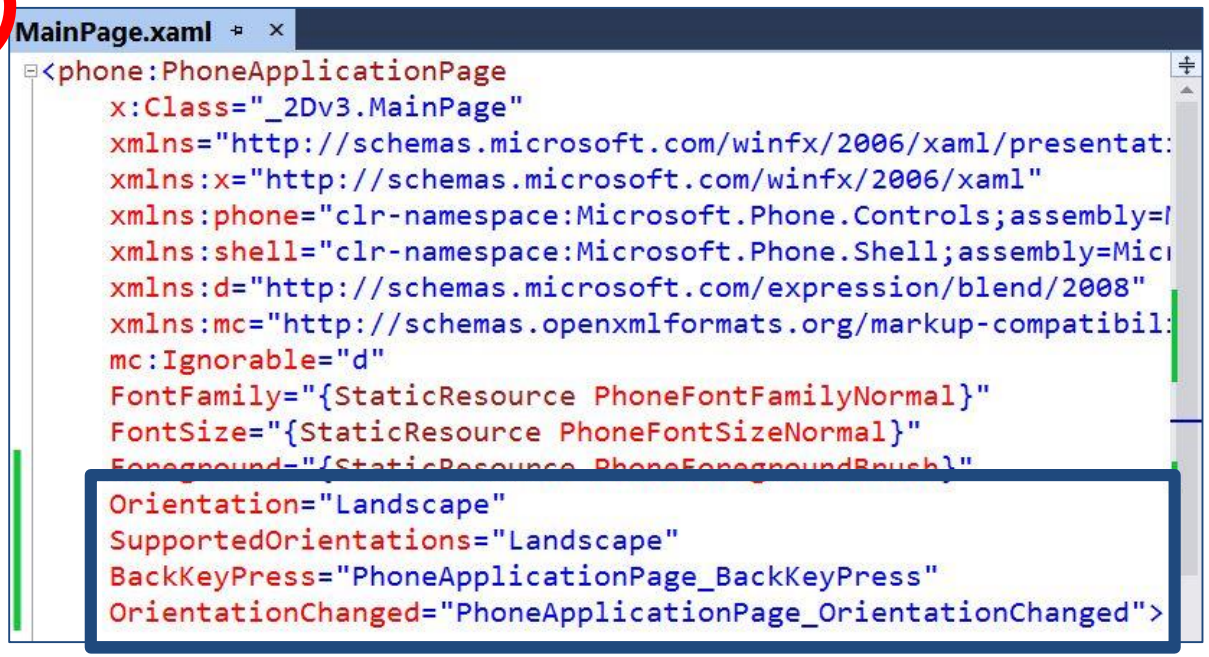
1



2



3



Memory Limits

Device	Memory	Windows Phone 8.0 * Native Min / Silverlight Min / Maximum	Windows Phone 8.1
Lower memory devices WVGA	512 MB	150 MB / 150 MB / 180 MB	185 MB
720p devices	1 GB	150 MB / 300 MB / 380 MB	390 MB
Newer devices: 4.5” WXGA	2 GB	150 MB / 450 MB / 570 MB	825 MB

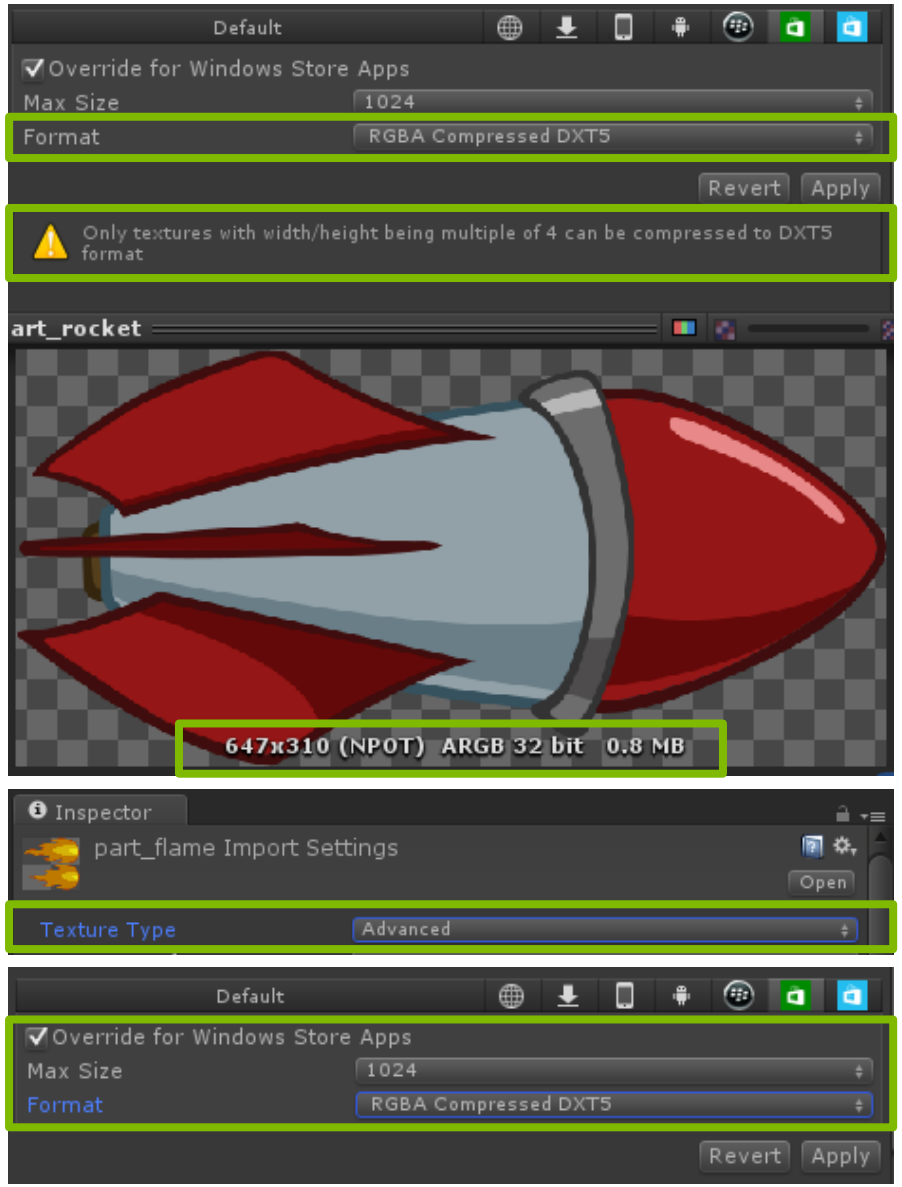
Memory-related manifest entries		
Capabilities	ID_FUNC_EXTENDED_MEM	WP8.0: Use the maximum memory limit
Requirements	ID_REQ_MEMORY_300	WP8.0: Restriction app for only devices greater than 512 MB
Prerequisites	MinDeviceMemory	WP8.1: Define minimum device

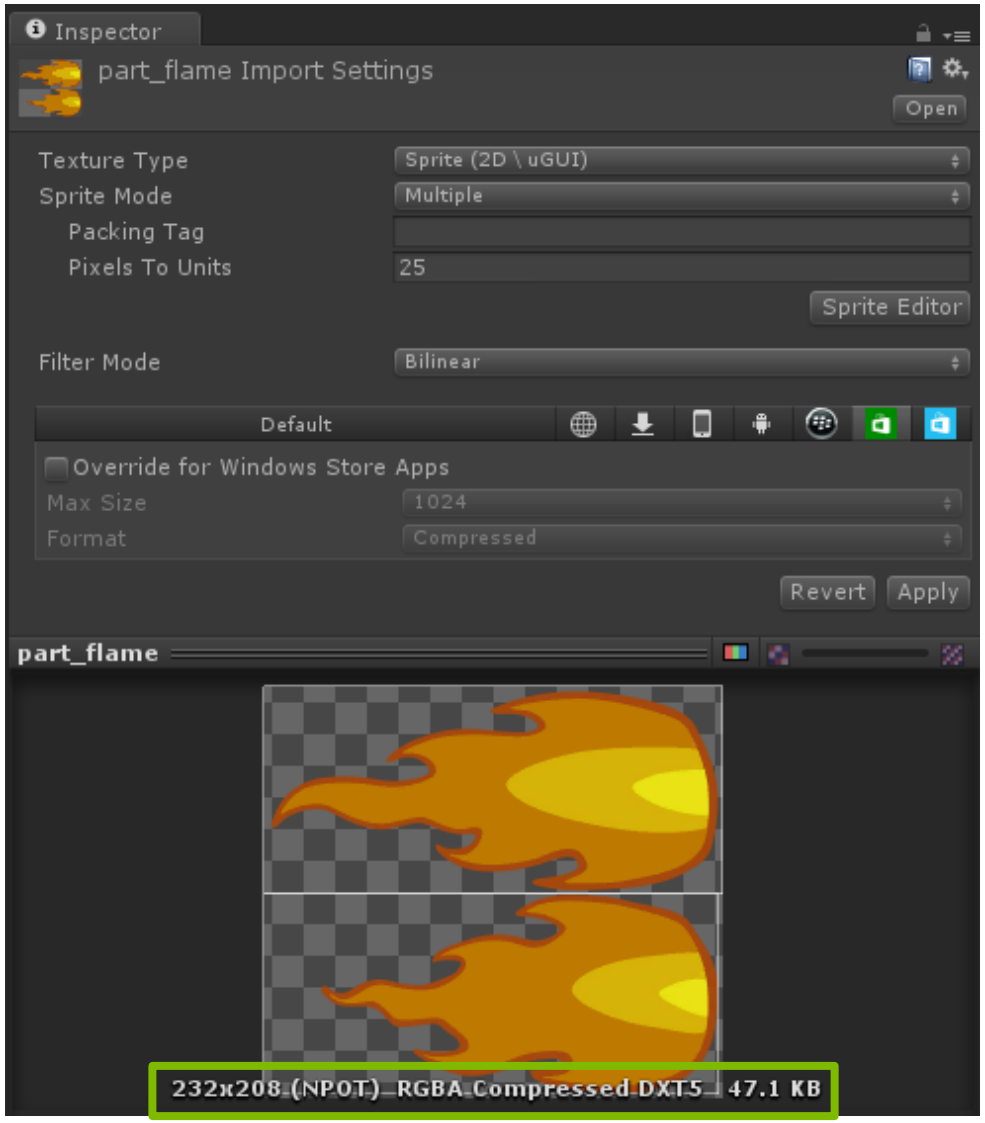
App memory limits for Windows Phone 8
http://msdn.microsoft.com/en-us/library/windowsphone/develop/jj681682(v=vs.105).aspx/html

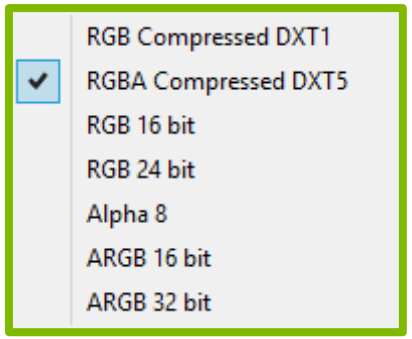
Memory Limits

```
// WP8.0.
ulong committedLimit = Windows.Phone.System.Memory.MemoryManager.ProcessCommittedLimit;
ulong committedBytes = Windows.Phone.System.Memory.MemoryManager.ProcessCommittedBytes;
// WP8.1.
ulong usageLimit = Windows.System.MemoryManager.AppMemoryUsageLimit;
ulong currentUsage = Windows.System.MemoryManager.AppMemoryUsage;
MemoryManager.AppMemoryUsageIncreased += OnAppMemoryUsageIncreased;
MemoryManager.AppMemoryUsageDecreased += OnAppMemoryUsageDecreased;

private void OnAppMemoryUsageIncreased(object sender, object e)
{   switch (MemoryManager.AppMemoryUsageLevel)
    {
        case AppMemoryUsageLevel.High:
            break;
        case AppMemoryUsageLevel.Medium:
            break;
        case AppMemoryUsageLevel.Low:
            break;
    }
}
```

Memory: Texture Compression







http://docs.unity3d.com/Manual/class-TextureImporter.html

Memory Resources

How to disable features in apps for lower-memory phones for WP8

http://msdn.microsoft.com/en-US/library/windows/apps/hh855083(v=vs.105).aspx/html

Managing Resource Constraints on Windows Phone

http://channel9.msdn.com/Events/Build/2014/3-542

Developing apps for lower-memory phones for Windows Phone 8

http://msdn.microsoft.com/en-us/library/windows/apps/hh855081(v=vs.105).aspx

Make sure to profile your phone app's memory usage

http://forum.unity3d.com/threads/202952-tips-and-tricks-make-sure-to-profile-your-phone-app-s-memory-usage!

Back Button

Windows Phone Technical Certification Requirements

http://msdn.microsoft.com/en-us/library/windows/apps/hh184840(v=vs.105).aspx

MainPage.xaml.cs

```
private void PhoneApplicationPage_BackKeyPress(object s, CancelEventArgs e)
{
    e.Cancel = UnityApp.BackButtonPressed();
    // TO DO。。。
}
```

Unity Scripts

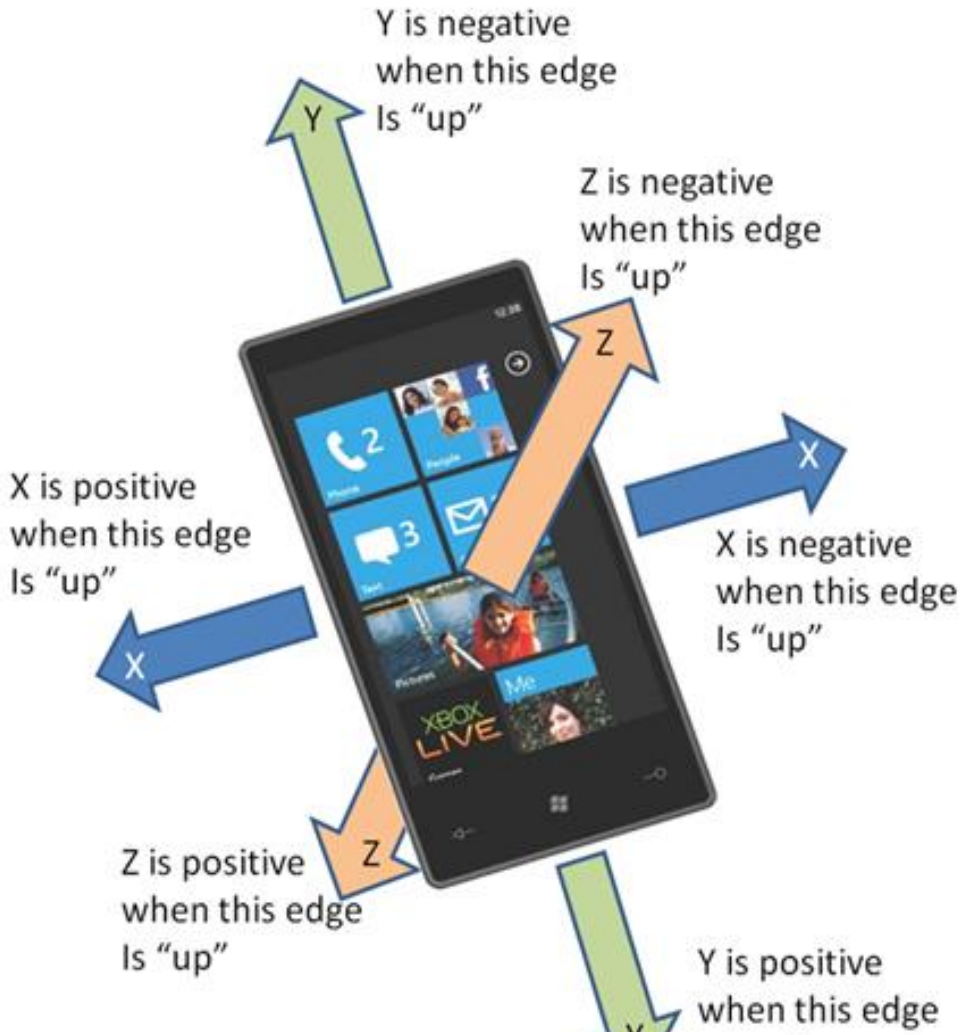
```
if (Input.GetKeyDown(KeyCode.Escape)) {
{
    Debug.Log("ESCAPE Input key down");
    Application.Quit();
}
```

Player Control

Unity Scripts

Accelerometer

```
if(Application.platform == RuntimePlatform.WP8Player)
{
    MoveAmount = (PlayerSpeed  * Input.acceleration.x )
        * Time.deltaTime;
    transform.Translate(Vector3.right * MoveAmount);
    MoveAmount = (PlayerSpeed * ((-Input.acceleration.z
        + 0.5f)*2)) * Time.deltaTime;
    transform.Translate(Vector3.forward * MoveAmount);
}
```



Touch

```
if (Application.platform == RuntimePlatform.WP8Player)
{
    if (grounded && Input.touchCount == 1 && Input.GetTouch(0).phase == TouchPhase.Ended)
        jump = true;
}
```

DEMO

Hardware Modifications

Hardware Modifications

- Multiple Resolutions
- Orientation
- Memory
- Texture Compression
- Back Button
- Accelerometer
- Touch

and more…

MVA Microsoft Virtual Academy

04 | Understand API Changes

Microsoft

.NET Framework

Mono .NET (Unity 3.5 .NET)

https://github.com/mono/mono/tree/master/mcs/class/System/System.Net

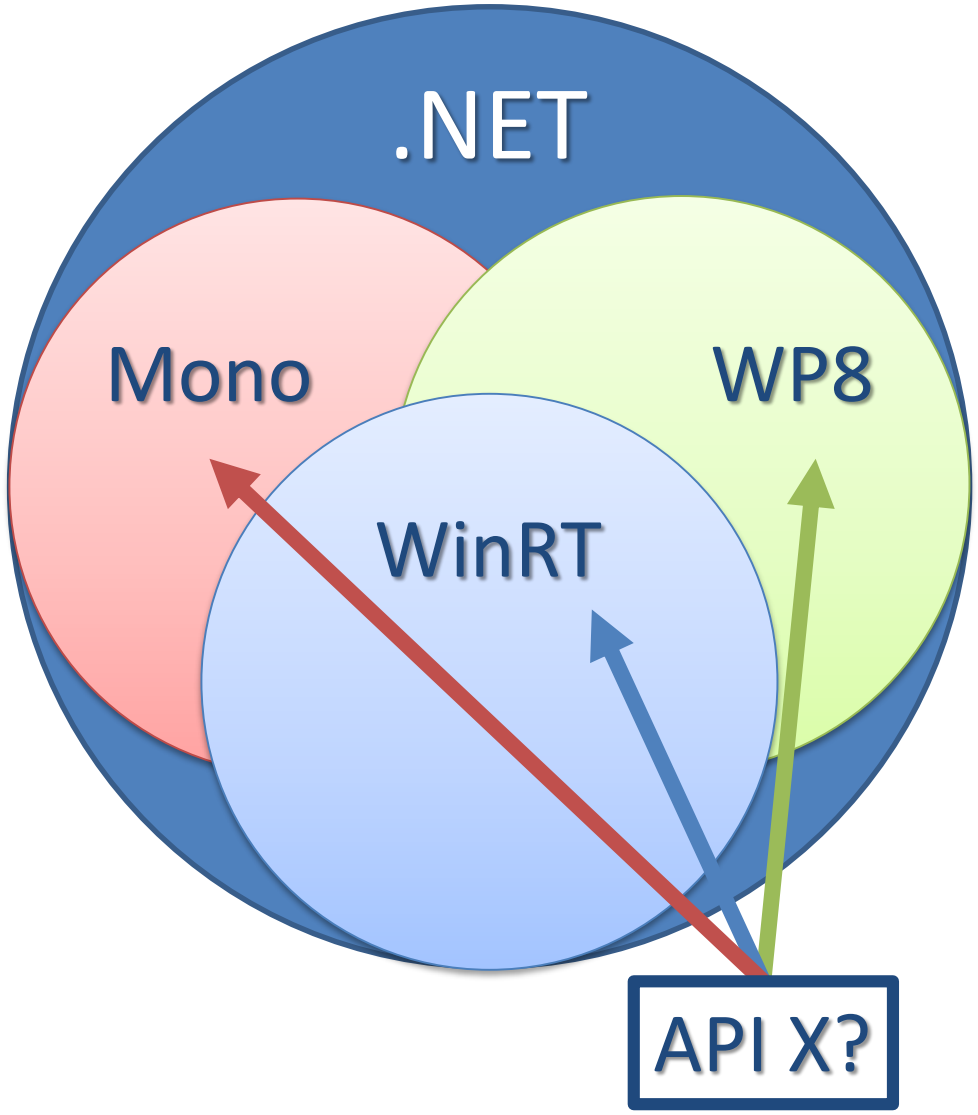
.NET API for Windows Phone 8.0

http://msdn.microsoft.com/en-us/library/windowsphone/develop/jj207211(v=vs.105).aspx

.NET API for Windows Runtime

(Windows 8.1 & Windows Phone 8.1)

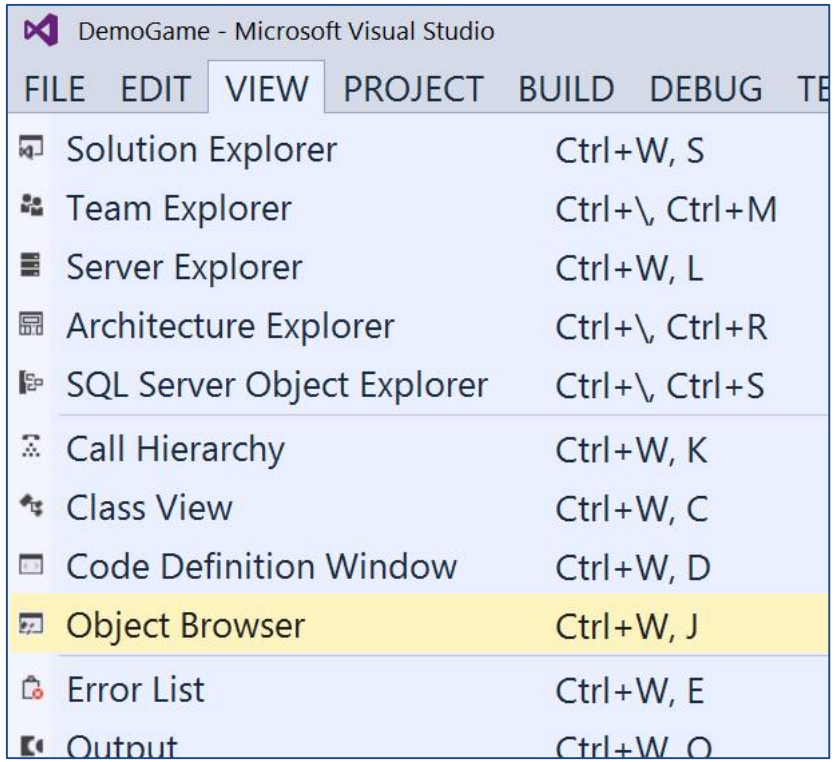
http://msdn.microsoft.com/en-us/library/windows/apps/br211369.aspx

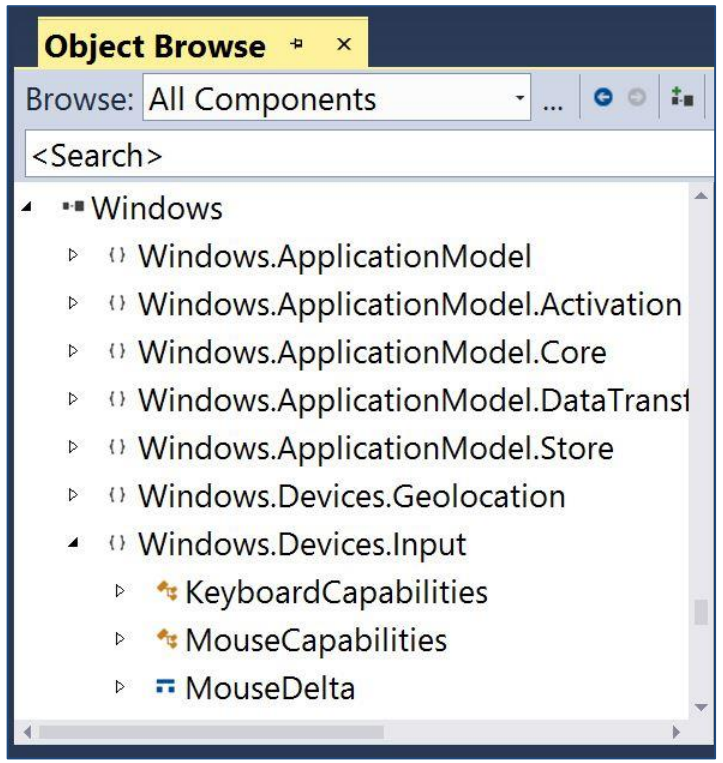


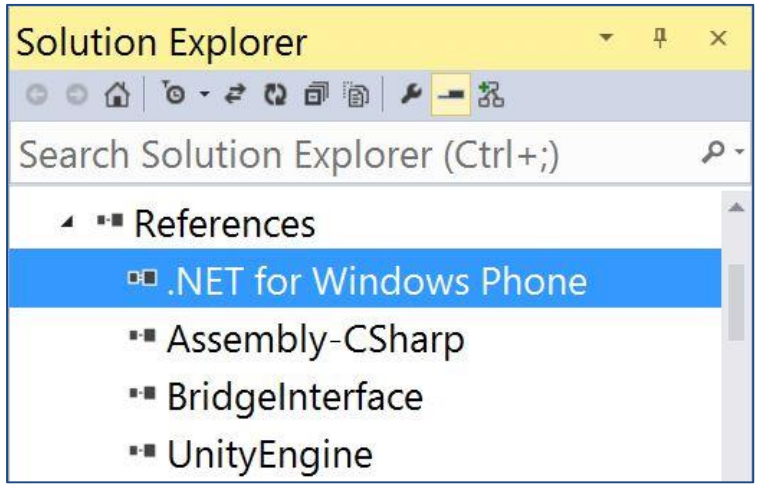
.NET Framework

How do you know if an API is supported?

- Visual Studio IDE ➔ View ➔ Object Browser
- Or open Solution Explorer, and double click on any References







Platform #defines

#defines	Supported Platforms
UNITY_WP8	Windows Phone 8.0 apps
UNITY_WP_8_1	Windows Phone 8.1 and Universal 8.1 apps on Windows Phone devices
UNITY_METRO	Windows Phone 8.1, Windows Store 8.0, Windows Store 8.1 and Universal 8.1 apps
UNITY_METRO_8_0	Windows Store 8.0 apps
UNITY_METRO_8_1	Windows Store 8.1 and Universal 8.1 apps running on Windows devices
UNITY_WINRT	Both Windows Phone and Windows Store apps, regardless of version
UNITY_WINRT_8_0	Windows Phone 8.0 and Windows Store 8.0 apps;
UNITY_WINRT_8_1	Windows Phone 8.1, Windows Store 8.1 and Universal apps for Windows and Windows Phone devices
NETFX_CORE	Windows Store 8.0, Windows Store 8.1, Windows Phone 8.1 and Universal 8.1 scripts that are compiled using Microsoft C# compiler.

Resolve API Needs

- Case 1: Unity Plugins

  Example: Elissa's GitHub Unity Plugins:

  https://github.com/elissatong/UnityPortingLabs

  Example: 3rd Party GitHub Platformer Plugins:

  https://github.com/windowsgamessamples/UnityPorting

- Case 2: Direct Communication: Action (void delegate)

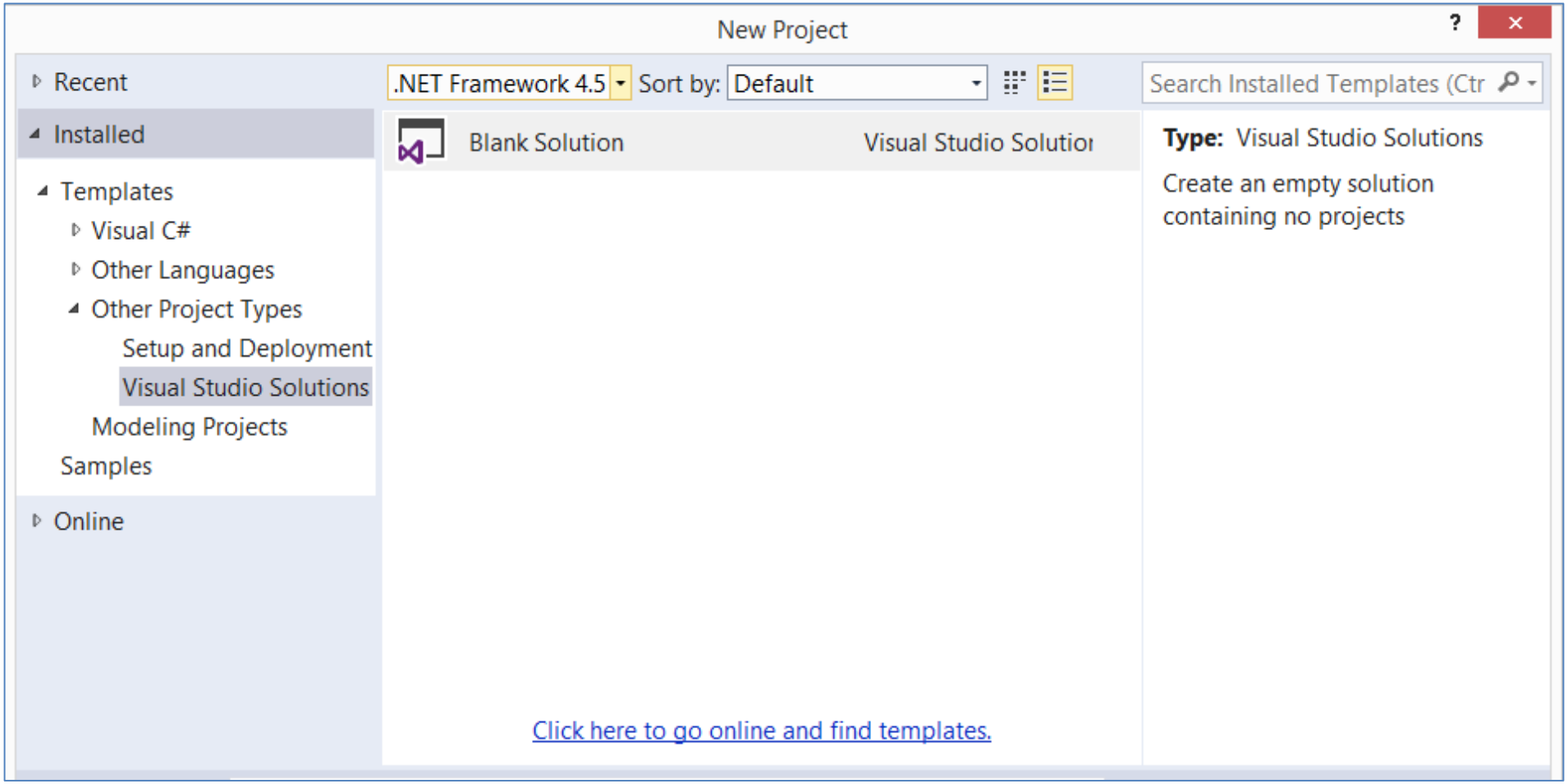
  Example: WindowsGateway.cs

- Case 3: Global instances

  Example: GameManager.cs

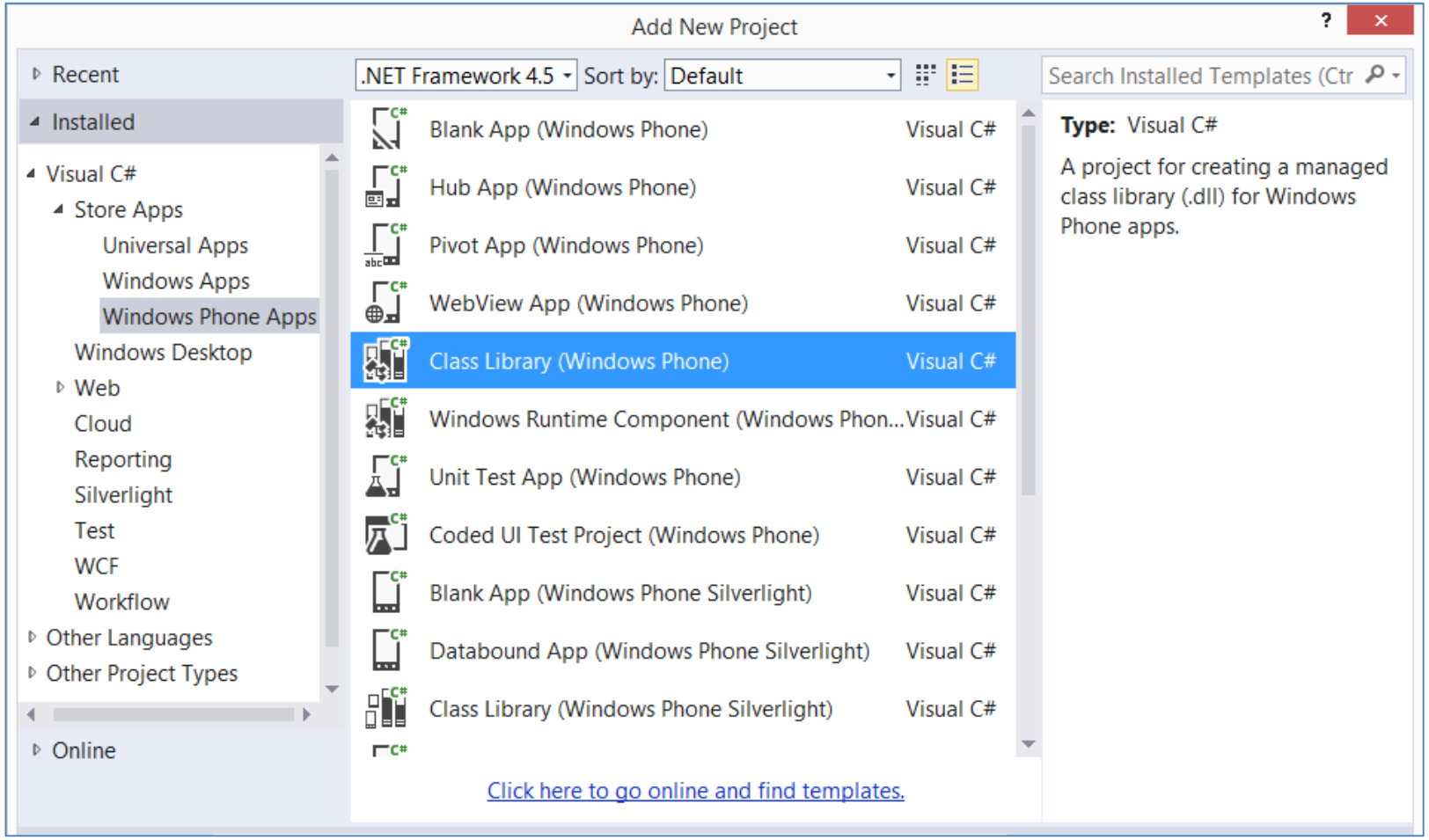
Case 1: Unity Plugins

Step 1: Create a Visual Studio Blank Solution



Unity Plugins

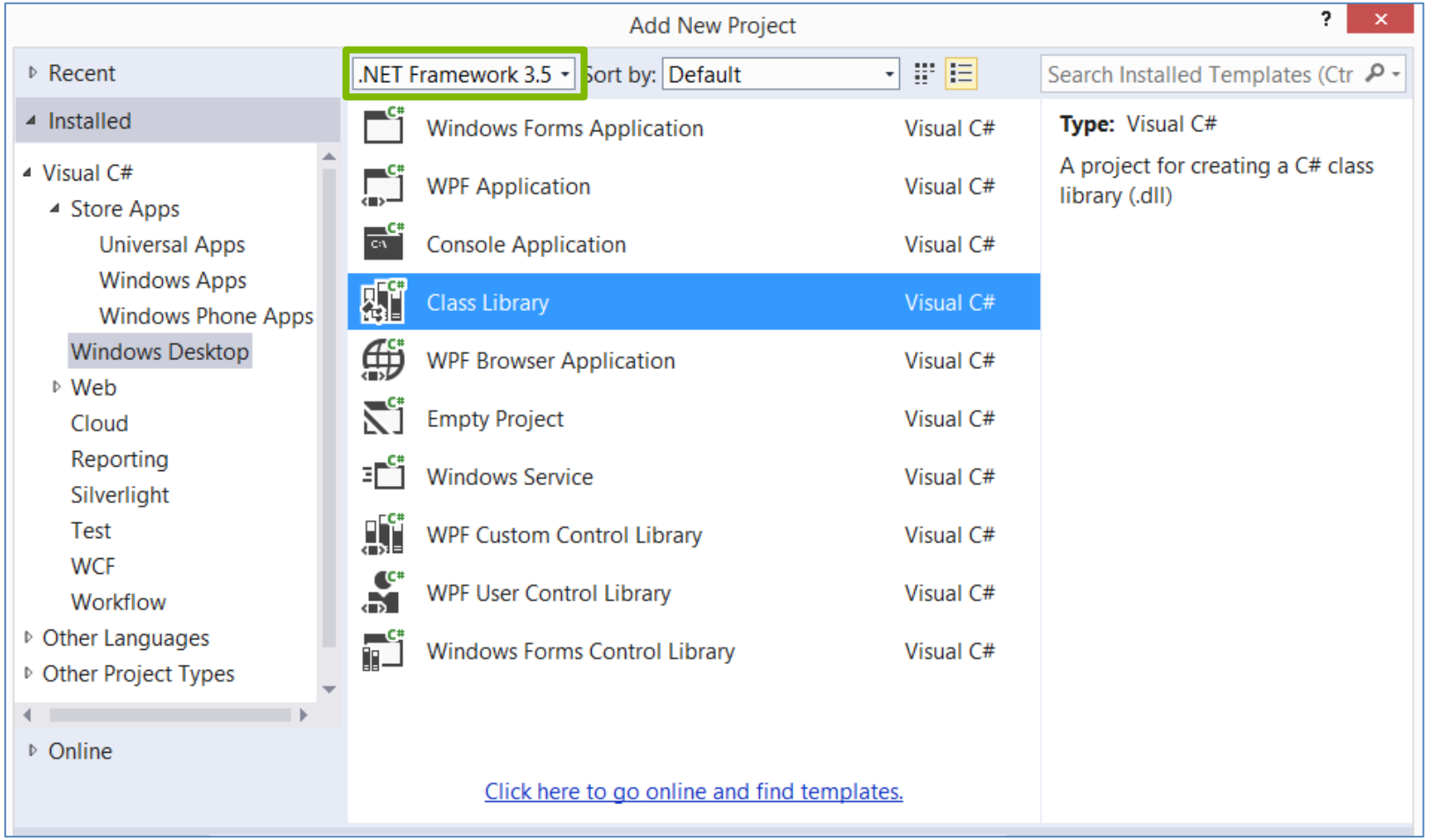
Step 2: Add **New Project** to Solution, select **Class Library (Windows Phone)**



Unity Plugins

Step 3: Add **New Project** to Solution, select **Windows Desktop**: **Class Library**

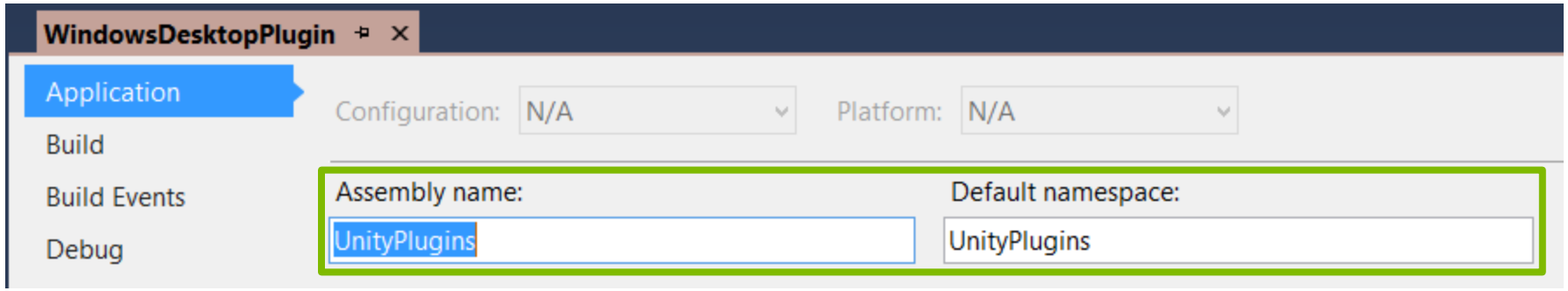
Unity uses .NET 3.5
Hence, need to select
.NET Framework 3.5

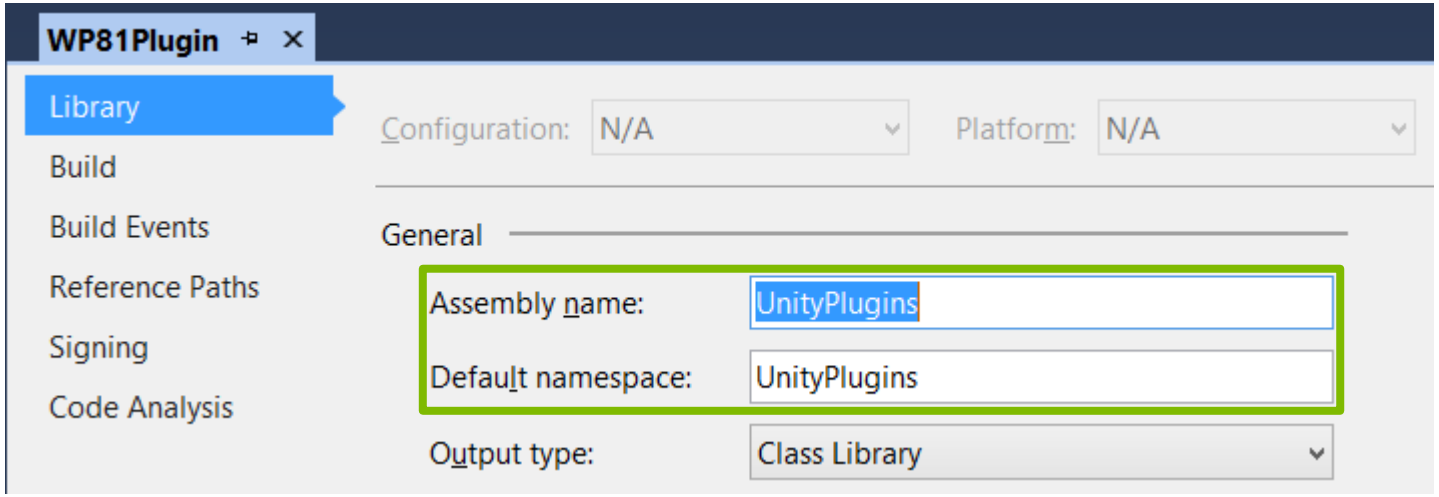


Unity Plugins

Step 4: In each Class Library project, open **Properties**.

Change **Assembly name** and **Default namespace** to use same name.





Unity Plugins

Step 5: Open up Class1.cs, change Namespace name to match.

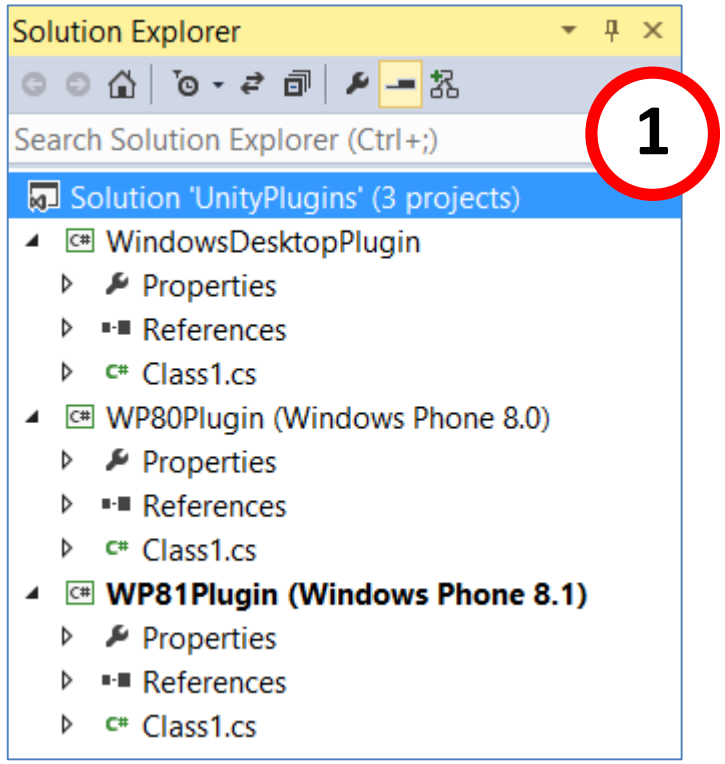
Make sure all clases, functions, and property names match.

```
namespace UnityPlugins
{
    0 references
    public class Class1
    {
        0 references
        public static string GetMemoryUsageLimit
        {
            get
            {
                return "Not supported";
            }
        }

        0 references
        public static string GetMemoryCurrentUsage
        {
            get
            {
                return "Not supported";
            }
        }
    }
}
```

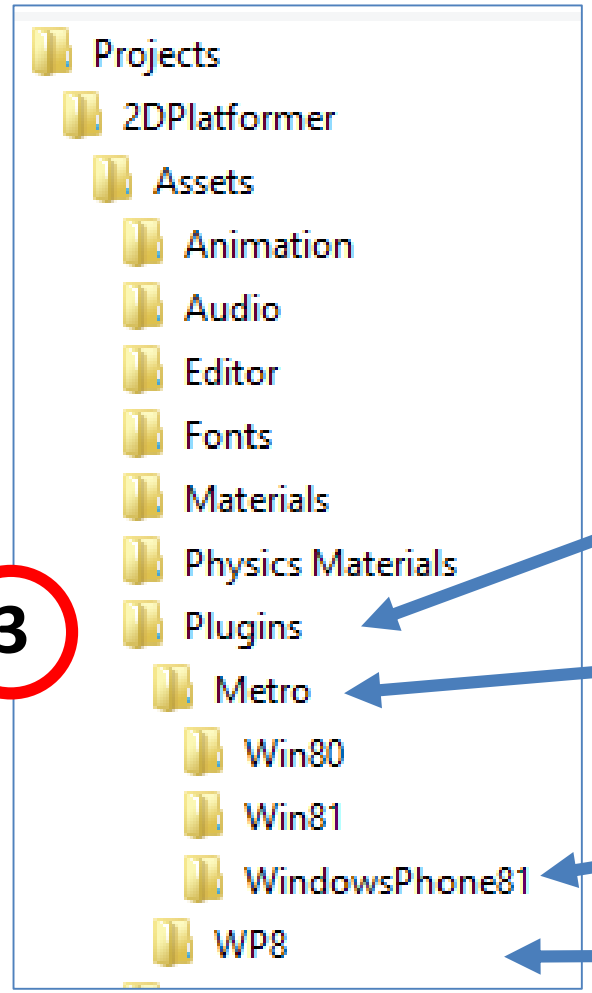
```
namespace UnityPlugins
{
    0 references
    public class Class1
    {
        0 references
        public static string GetMemoryUsageLimit
        {
            get
            {
                // WP8.1.
                ulong usageLimit = Windows.System.MemoryManager.AppMemoryUsageLimit;
                ulong bytesToMB = usageLimit / (1024 * 1024);
                return bytesToMB.ToString() + " MB";
            }
        }

        0 references
        public static string GetMemoryCurrentUsage
        {
            get
            {
                // WP8.1.
                ulong currentUsage = Windows.System.MemoryManager.AppMemoryUsage;
                ulong bytesToMB = currentUsage / (1024 * 1024);
                return bytesToMB.ToString() + " MB";
            }
        }
    }
}
```

Unity Plugins

Step 6: Build Solution, copy UnityPlugins.dll to Unity's Plugins folder.

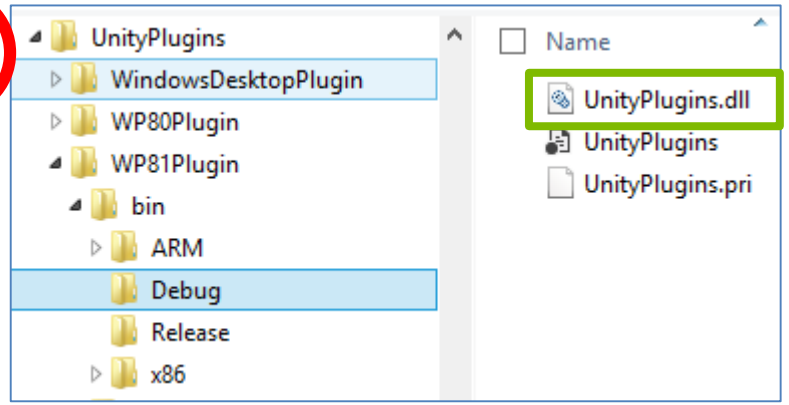


1



3

2



Unity Plugins folder

For Universal Apps plugins:
Windows Store 8.0/8.1 &
Windows Phone 8.1

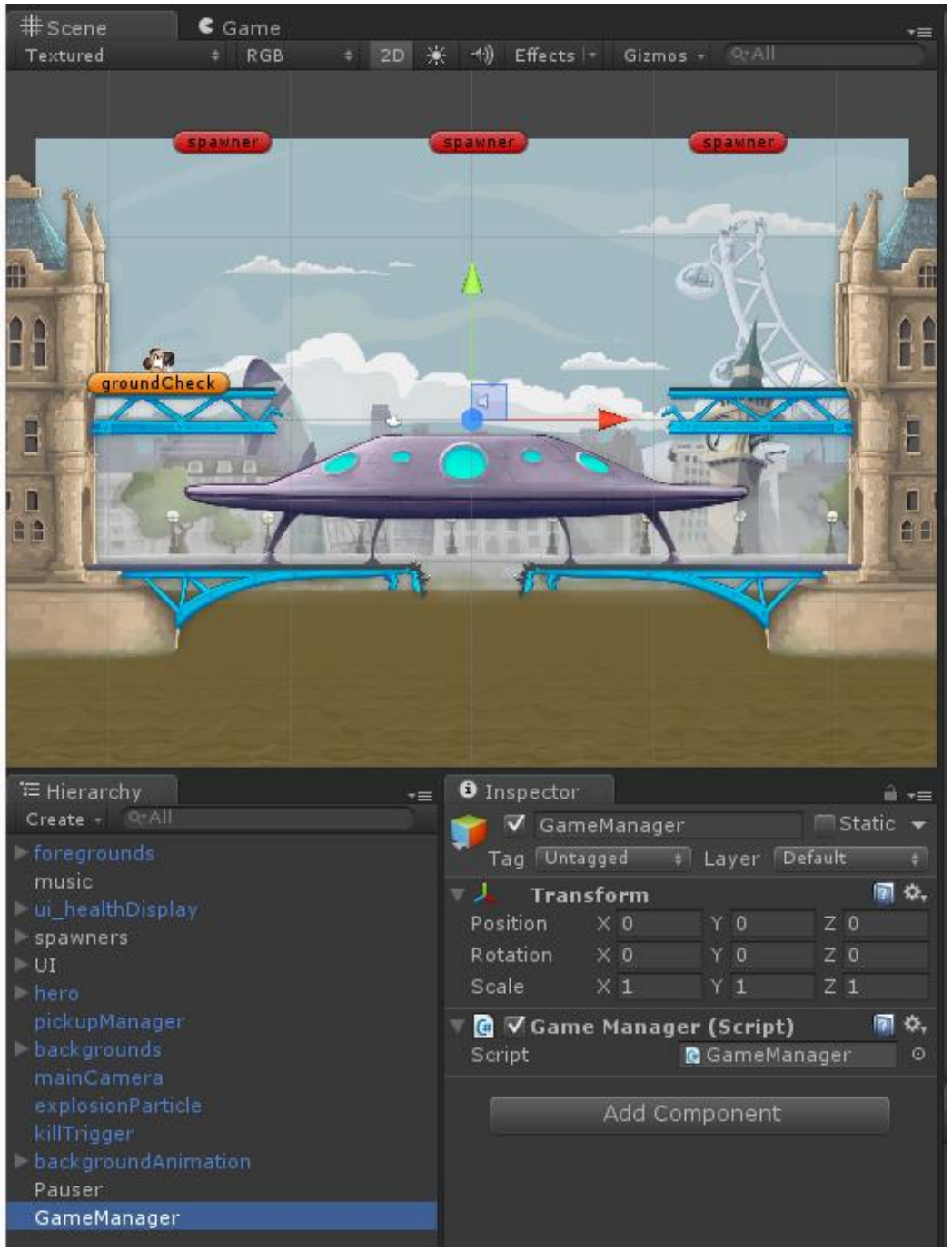
Windows Phone 8.1

Windows Phone 8.0

Unity Plugins

Step 7: To test out the new plugin:

- Add a new GameObject to a Scene
- Add a new script to GameObject
- Build for Universal Apps

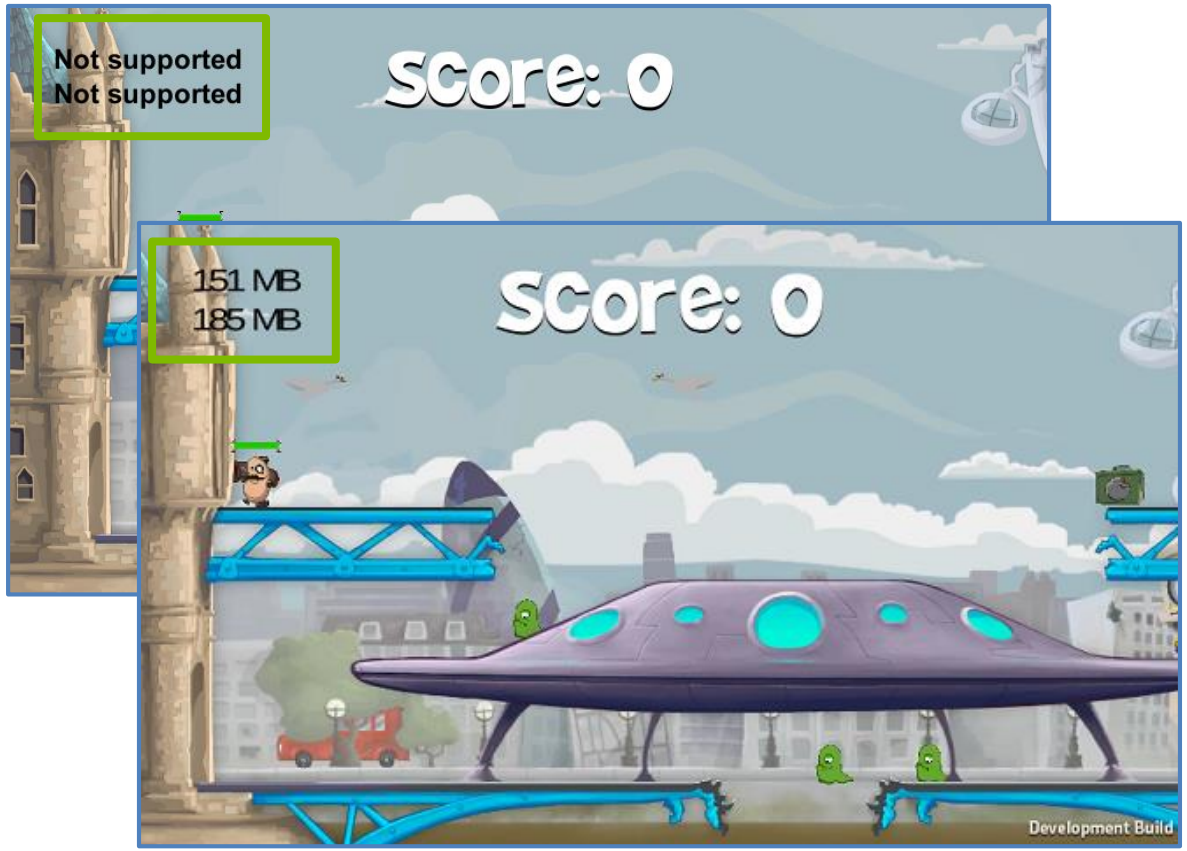


Unity Plugins

Step 7: Open Visual Studio project, open up the Unity script, add code from UnityPlugins.

```
public class GameManager : MonoBehaviour
{
   private GUIStyle mGUIStyleLabel;
   private void SetGuiLabelStyles()
   {
      mGUIStyleLabel = new GUIStyle();
      mGUIStyleLabel.fontSize = 24;
      mGUIStyleLabel.fontStyle = FontStyle.Bold;
      mGUIStyleLabel.alignment = TextAnchor.MiddleLeft;
   }
   void OnGUI()
   {
      GUI.Label(new Rect(40, 20, 250, 50),
         UnityPlugins.Class1.GetMemoryCurrentUsage, mGUIStyleLabel);
      GUI.Label(new Rect(40, 50, 250, 50),
         UnityPlugins.Class1.GetMemoryUsageLimit, mGUIStyleLabel);
   }
   void Start()
   {
      SetGuiLabelStyles();
   }
```

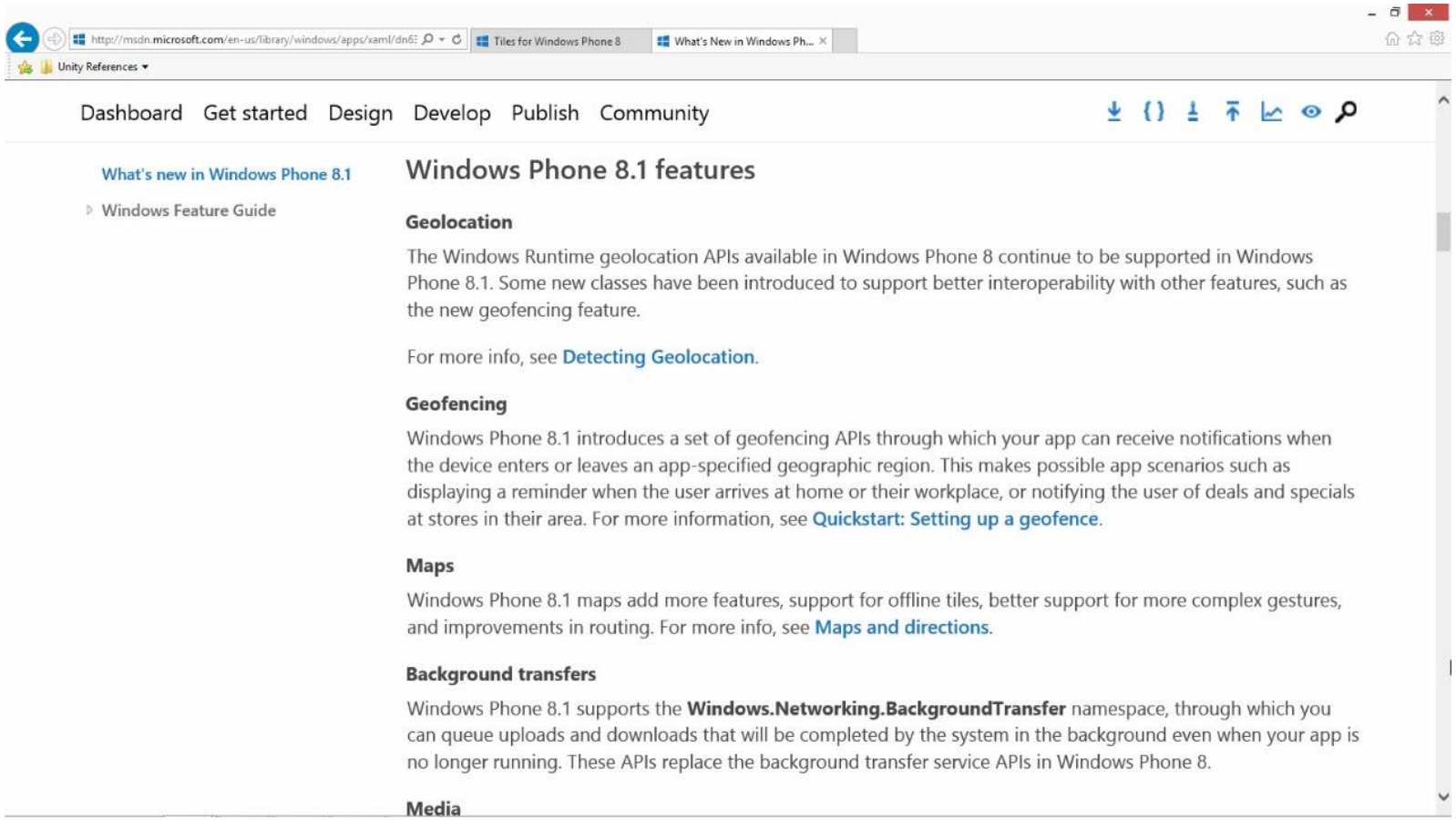
Unity Editor using Windows Desktop plugin



Windows Phone 8.1 Emulator

When to use Unity Plugins

- Missing API errors
  - XMLDocument
  - Reflection
  - File IO
  - Networking
- Windows Phone features
  - IAP
  - Trial Mode
  - Maps
  - Speech
  - Live Tiles



Plugins/API References

Windows Phone Plugins step by step guide (using C#)

http://docs.unity3d.com/Manual/wp8-plugins-guide-csharp.html

Windows Phone Plugins step by step guide (using C++)

http://docs.unity3d.com/Manual/wp8-plugins-guide-cplusplus.html

Unity Script Reference: Windows File API

http://docs.unity3d.com/ScriptReference/Windows.File.html

Case 2: Direct Communication

For direct communication between Unity & Windows Phone

Step 1: Create Unity script, WindowsGateway.cs

```
/// Windows specific and interop between Unity and Windows Store or Windows Phone 8
using UnityEngine; using System.Collections; using System;
public static class WindowsGateway
{
    static WindowsGateway()
    {
        UnityLoaded = delegate { };
        OnClickPlay = delegate { };
        OnScoreUpdate = delegate { };
        OnClickBuy = delegate { };
    }
    public static Action UnityLoaded;
    public static Action OnClickPlay;
    public static Action OnScoreUpdate;
    public static Action OnClickBuy;
}
```

Direct Communication

Step 2: In Unity scripts, make the calls to the functions as needed.

MainManager.cs:

```
void OnGUI()
{
    if (GUI.Button(new Rect(x, yP, BUTTON_SIZE, BUTTON_SIZE), "", mGUIStyle))
    {
        WindowsGateway.OnClickPlay();
        Application.LoadLevel("Level");
    }
}
```

Direct Communication

Step 3: In Windows Phone solution, create events & register events to the Action delegate.

MainPage.xaml.cs:

```
public MainPage(SplashScreen splashScreen)
{  this.InitializeComponent();
    splash = splashScreen;
    GetSplashBackgroundColor();
    OnResize();
    onResizeHandler = new WindowSizeChangedEventHandler((o, e) => OnResize());
    Window.Current.SizeChanged += onResizeHandler;
    WindowsGateway.OnClickPlay = OnClickPlay;
}

private void OnClickPlay()
{      FlurryAnalytics.LogEvent(Constants.EVENT_PLAY_GAME);       }
```

Case 3: Global Instance

To access Unity variables & states

Step 1: In a Unity script, add a static instance.

```
private static readonly GameManager instance = new GameManager();

private GameManager() { }

public static GameManager Instance
{
    get
    {
        return instance;
    }
}

public bool IsMainMenu = true;
public static bool IsPaused = false;
```

Case 3: Global Instance

Step 2: In Windows Phone solution, just reference the Unity static instance!

```
public MainPage(SplashScreen splashScreen)
{
    this.InitializeComponent();
    splash = splashScreen;
    GetSplashBackgroundColor();
    OnResize();
    onResizeHandler = new WindowSizeChangedEventHandler((o, e) => OnResize());
    Window.Current.SizeChanged += onResizeHandler;

    bool isMainMenu = GameManager.Instance.IsMainMenu;
    bool isGamePaused = GameManager.IsPaused;
    if (isGamePaused)
    {
        // Do something
    }
    elseif (isMainMenu) {// Do something}
}
```

DEMO

Understand API Changes

Understanding API Changes

.NET Framework

Mono .NET

.NET for Windows Phone

Missing APIs

Unity Plugins

Direct Communication

Global Instance

and more…

Additional Resources

http://unity3d.com/pages/windows/porting

REACH MILLIONS OF PLAYERS. JUST LIKE THAT.

It's so easy to bring your existing Unity titles to Windows Store and Windows Phone. Use the resources on this page to port your content quickly and efficiently to a whole new global audience.

START WITH THE BASICS

Our Getting Started guides cover everything you need to know: configurations, SDKs, general considerations and first steps

Getting started on Windows Phone (663 KB)

Getting started on Windows Store (720 KB)

THEN GET THE PORTING TIPS

Once you get the basics, read our porting tips that pinpoint what to consider when adjusting your code to target Windows Store and Windows Phone

Tools

Developer Tools
http://developer.windowsphone.com/en-us/getstarted/downloads
Windows Phone Toolkit
http://phone.codeplex.com/
Coding4Fun Toolkit
http://coding4fun.codeplex.com/
Windows Phone Power Tools
http://wptools.codeplex.com/

Resources

Multi-resolution apps for Windows Phone 8
http://msdn.microsoft.com/en-us/library/windowsphone/develop/jj206974(v=vs.105).aspx
Introduction to multiple resolution support
http://developer.nokia.com/community/wiki/Introduction_to_multiple-resolution_support_on_Windows_Phone_8_apps
Command Line Prompts
https://docs.unity3d.com/Documentation/Manual/CommandLineArguments.html

Sharing

http://msdn.microsoft.com/zh-cn/jj923044
http://weibowp8sdk.codeplex.com

Learning Resources

Unite 2014: Seattle
http://unity3d.com/unite/unite2014/keynote
2014 Building Your First Windows Game with Unity
http://channel9.msdn.com/Events/Build/2014/2-503
Unity3D WP8 Examples:
https://docs.unity3d.com/Documentation/Manual/wp8-examples.html
Nokia Community
http://developer.nokia.com/community/wiki/Getting_Started_with_Unity_3D_on_Windows_Phone
Microsoft Virtual Academy: Porting Unity Games to Windows 8.1 & Windows Phone 8
http://www.microsoftvirtualacademy.com/training-courses/porting-unity-games-to-windows-8-1-windows-phone
Porting Unity Games
http://unity3d.com/pages/windows/porting
Code Kwondo: Closer Look at Unity
http://channel9.msdn.com/Events/Developer-Movement/UnityAndGaming/Part1
Building Windows Games with Unity
http://channel9.msdn.com/Events/Windows-Camp/Building-Windows-Games-with-Unity

MVA Microsoft Virtual Academy

Thank you

Elissa Tong
elistong@microsoft.com
https://github.com/elissatong/UnityPortingLabs

Microsoft

Microsoft