

 Windows Phone

XNA Games  
on  
Windows Phone



7

Rob Miles & Andy Wigley  
Windows Phone 7 Jump Start  
Microsoft Corporation

---

---

---

---

---

---

---

Agenda

Windows Phone as an XNA platform

Creating XNA games on Windows Phone

Managing Display Orientation

Monitoring Performance

Optimization tips for XNA games on Windows Phone

2

Windows Phone Jump Start

---

---

---

---

---

---

---

 Windows Phone

Windows Phone as  
an XNA Platform

---

---

---

---


---

---

---

## XNA on Windows Phone

- Windows Phone is a great platform for XNA
- Performance is impressive, especially in 3D
- Games are easy to write, test and deploy
- There are some very interesting input options



4

Windows Phone Jump Start

---

---

---

---

---

---

---

## Key XNA features

- Managed code on Windows Phone is a better fit to the hardware than the Xbox 360
  - Only worry about optimizing when you notice a performance problem
- A number of standard screen dimensions
  - But also hardware scaling so you can upsize low resolution, fast moving, displays
- Integration of touch and accelerometer inputs

5

Windows Phone Jump Start

---

---

---

---

---

---

---

## Key XNA features

- Hardware supports shaders
  - But you can't write your own just yet
  - Use the five built in ones
- Full support for Content Management
  - Write your own importers to run on the phone
- Xbox Live Integration
  - Support for Avatars and Achievements

6

Windows Phone Jump Start

---

---


---

---

---

---

---



# Creating XNA Games on Windows Phone

---

---

---

---

---

---

---

## Creating XNA games

- The game creation process is exactly as for Windows PC or Xbox 360
  - Create the games using Visual Studio 2010 and deploy to your target device
  - Can create multiple versions of projects for the different platforms
- Most XNA 3.1 games will migrate with no problems

---

---

---

---

---

---

---

## XNA 4.0 from 3.n

- If you are an experienced XNA developer your code should be easy to port
  - There have been some breaking changes to the way that low level graphical primitives are managed (see Sean Hargreaves blog)
- However, if you use 3D models, or work in 2D you will not notice many changes

---

---

---

---

---

---

---

## WP7 vs Zune HD

- The Zune HD uses version 3.1 of XNA
- It has a touch screen and an accelerometer
- It does not support 3D
- There is no Zune HD Marketplace



10

Windows Phone Jump Start

---

---

---

---

---

---

---

## XNA Overview

- All XNA games do three things:
  - Load content when they start
  - Update the game world as quickly as possible
  - Draw the game world as quickly as possible
- Each of these behaviours maps onto a method in the game class created by Visual Studio when an XNA game project is created

11

Windows Phone Jump Start

---

---

---

---

---

---

---

## Drawing Cheese in XNA

### Demo



Demo 1: Simple sprite drawing

12

Windows Phone Jump Start

---

---

---

---

---

---

---

## Update Rate

```
public CheeseLanderGame()
{
    graphics = new GraphicsDeviceManager(this);
    Content.RootDirectory = "Content";

    // Frame rate is 30 fps by default for Windows Phone.
    TargetElapsedTime = TimeSpan.FromTicks(333333);
}
```

- The Draw and Update methods are called for you by the XNA framework when your game runs
  - On an Xbox or Windows PC update is called 60 times a second
- On Windows Phone this is reduced to 30
  - This is to reduce power consumption

13

Windows Phone Jump Start

---

---

---

---

---

---

---

## XNA Sprites

- A simple sprite is made up of a texture and a position
- The XNA coordinate system puts the origin at the top left hand corner
- Increasing X moves towards the right
- Increasing Y moves down the screen
- The position of the sprites is updated in the Update method

14

Windows Phone Jump Start

---

---

---

---

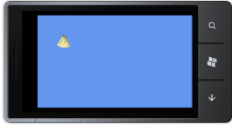
---

---

---

## Moving Cheese in XNA

### Demo



Demo 2: Using Update to move a sprite

15

Windows Phone Jump Start

---

---

---

---

---

---

---

## Object Positioning

- Floating point position values allow for slow and smooth movement
  - Convert to integers to position the drawing
- All movement values should be calculated so that the game runs at the same speed on all platforms
- Use the display width and height to manage this
  - Also might need to perform this calculation after orientation changes (see later)

16

Windows Phone Jump Start

---

---

---

---

---

---

---

## Object Collisions

```
if (cheeseDrawPos.Intersects(breadDrawPos))
{
    // May have won
    if (cheeseDrawPos.Top < breadDrawPos.Top &&
        cheeseDrawPos.Left >= breadDrawPos.Left &&
        cheeseDrawPos.Right <= breadDrawPos.Right &&
        Math.Abs(cheeseXSpeed) < cheeseMaxXLandSpeed &&
        Math.Abs(cheeseYSpeed) < cheeseMaxYLandSpeed)
    {
        // cheese is properly landed - player wins
        gameWon();
    }
}
```

- For a 2D game you can detect when two rectangles intersect and deal with collisions

17

Windows Phone Jump Start

---

---

---

---

---

---

---



## Orientation

---

---

---

---

---

---

---

## Orientation



- Windows Phone games can be played in many orientations
- By default the game is configured for landscape orientation

19

Windows Phone Jump Start

---

---

---

---

---

---

---

## Forcing Game Orientation

```
public Game1()
{
    graphics = new GraphicsDeviceManager(this);
    Content.RootDirectory = "Content";
    graphics.PreferredBackBufferWidth = 480;
    graphics.PreferredBackBufferHeight = 800;

    // Frame rate is 30 fps by default for Windows Phone.
    TargetElapsedTime = TimeSpan.FromTicks(3333333);
}
```

- You can force a particular orientation by setting the size of the back buffer
- This works by using the scaler

20

Windows Phone Jump Start

---

---

---

---

---

---

---

## The Magical Scaler

- The scaler uses hardware, so your game is not slowed down by it
  - It interpolates to make the scaling look good
- It scales from 240x240 to 800x480 (or 480x800)
- It will add a letterbox (black bars) if the chosen aspect ratio doesn't match the hardware
- Viewport properties and touch input positions in your program always match the scaled screen

21

Windows Phone Jump Start

---

---

---

---

---

---

---

## Selecting Orientations

```
public Game1()
{
    graphics = new GraphicsDeviceManager(this);
    Content.RootDirectory = "Content";

    graphics.SupportedOrientations = DisplayOrientation.Portrait |
                                    DisplayOrientation.LandscapeLeft |
                                    DisplayOrientation.LandscapeRight;

    // Frame rate is 30 fps by default for Windows Phone.
    TargetElapsedTime = TimeSpan.FromTicks(333333);
}
```

- Your game can indicate which orientations it can support
- The game above can support anything!

22

Windows Phone Jump Start

---

---

---

---

---

---

---

---

## Detecting Changes

```
public Game1()
{
    graphics = new GraphicsDeviceManager(this);
    Content.RootDirectory = "Content";

    graphics.SupportedOrientations = DisplayOrientation.Portrait |
                                    DisplayOrientation.LandscapeLeft |
                                    DisplayOrientation.LandscapeRight;

    this.Window.OrientationChanged +=
        new EventHandler<EventArgs>(Window_OrientationChanged);

    // Frame rate is 30 fps by default for Windows Phone.
    TargetElapsedTime = TimeSpan.FromTicks(333333);
}
```

- There is an event you can bind to if you need to detect orientation changes

23

Windows Phone Jump Start

---

---

---

---

---

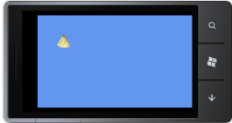
---

---

---

## Changing Orientations in XNA

### Demo



Demo 3: Orientation Changes

24

Windows Phone Jump Start

---

---

---

---

---

---

---

---





# Monitoring Performance

---

---

---

---

---

---

---

# Measuring Performance

```
public StarlightGame()
{
    graphics = new GraphicsDeviceManager(this);
    Content.RootDirectory = "Content";

    graphics.PreferredBackBufferWidth = 480;
    graphics.PreferredBackBufferHeight = 800;

    // Frame rate is 30 fps by default for Windows Phone.
    TargetElapsedTime = TimeSpan.FromTicks(333333);

    //Components.Add(new FrameRateCounter(this));
}
```

- An XNA game contains components that have their own LoadContent, Draw and Update
- I have created one that displays a frame counter

26

Windows Phone Jump Start

---

---

---

---

---

---

---

# Starlight Game Performance

# Demo

Demo 4: Starlight Framecounter



27

Windows Phone Jump Start

---

---

---

---

---

---

---

# Review

XNA on Windows Phone does not support 3D

Your game must call the Update method 60 times a second

XNA games must run at 320 x 260 resolution

Windows Phone only supports games in landscape orientation

XNA games can detect when the phone orientation changes

28

Windows Phone Jump Start

---

---

---

---

---

---

---

# Review

XNA on Windows Phone does not support 3D

Your game must call the Update method 60 times a second

XNA games must run at 320 x 260 resolution

Windows Phone only supports games in landscape orientation

XNA games can detect when the phone orientation changes

29

Windows Phone Jump Start

---

---

---

---

---

---

---

# Review

XNA on Windows Phone does not support 3D

Your game must call the Update method 60 times a second

XNA games must run at 320 x 260 resolution

Windows Phone only supports games in landscape orientation

XNA games can detect when the phone orientation changes

30

Windows Phone Jump Start

---

---

---

---

---

---

---

# Review

XNA on Windows Phone does not support 3D

Your game must call the Update method 60 times a second

XNA games must run at 320 x 260 resolution

Windows Phone only supports games in landscape orientation

XNA games can detect when the phone orientation changes

31

Windows Phone Jump Start

---

---

---

---

---

---

---

# Review

XNA on Windows Phone does not support 3D

Your game must call the Update method 60 times a second

XNA games must run at 320 x 260 resolution

Windows Phone only supports games in landscape orientation

XNA games can detect when the phone orientation changes

32

Windows Phone Jump Start

---

---

---

---

---

---

---

# Review

XNA on Windows Phone does not support 3D

Your game must call the Update method 60 times a second

XNA games must run at 320 x 260 resolution

Windows Phone only supports games in landscape orientation

XNA games can detect when the phone orientation changes

33

Windows Phone Jump Start

---

---

---

---

---

---

---

# Coming Up Next...

Using the accelerometer in an XNA game

Using the touchscreen in an XNA game

Advanced XNA sound playback

Using the Guide for text input

Controlling media playback with XNA

34

Windows Phone Jump Start

---

---


---

---

---

---

---



Windows  
Phone

© 2010 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.

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.

---

---

---

---

---

---

---