Developing Windows Phone Applications

Agenda

Pre-requisites

- **Windows Phone SDK**
  - Free Download
  - IDE : Visual Studio 2010/2012 with Emulator
- **Development Languages**
  - C#
  - Visual Basic.NET
  - F#
- **Development Machine**
  - Windows 7 or above
- **Frameworks**
  - .NET 4.0
  - Silverlight 4
  - XNA

Hardware capabilities

- **Standard resolution for WP7**
  - 800 x 480
  - 480 x 320
- **Same touch input**
- **Consistent processor/GPU**
- **Same available RAM**
- **Optional Keyboard**
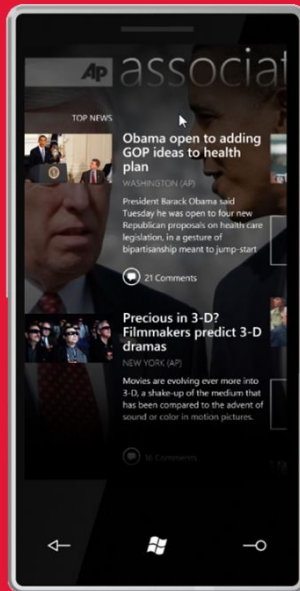
800 or 480

480 or 320

4

Introduction

## Options for Developers

Microsoft® Silverlight™

- Powered 2008/2010 Olympics
- Modern XAML, event-driven application UI framework
- Rapid creation of visually stunning apps
- Metro-themed UI controls
- HTML/JavaScript
- 500,000 developers spanning Windows and web

xna

- XBOX Gai
- High performance game framework
- Rapid creation of multi-screen 2D and 3D games
- Rich content pipeline
- Mature, robust, widely adopted technology spanning Xbox 360, Windows, and Zune

Developer Frameworks
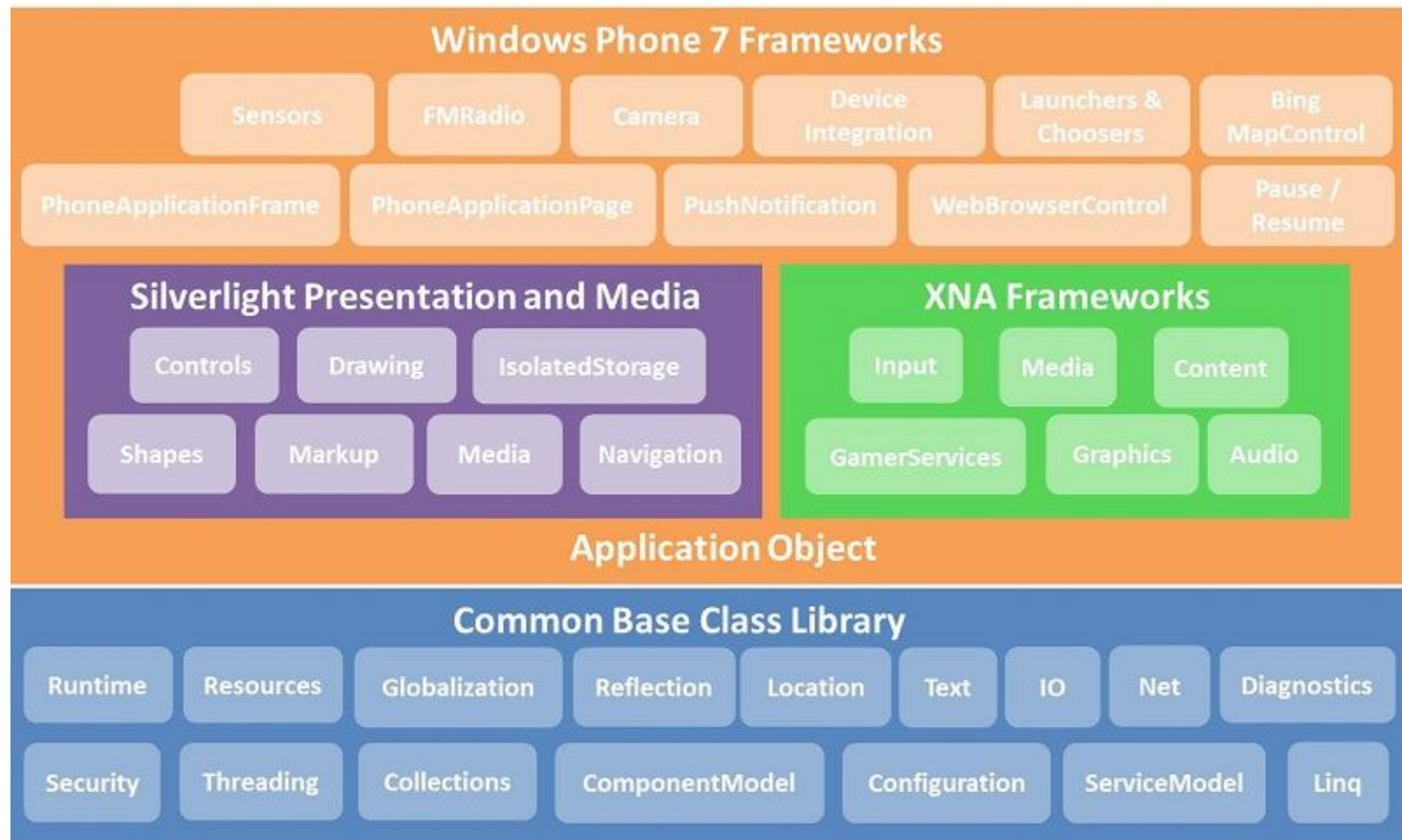
Launchers and Choosers

- 15 Launchers – "Fire and Forget" action
  - EmailComposeTask
  - PhoneCallTask
  - MediaPlayerLauncher
- 10 Choosers – "Open File Dialog" action
  - EmailAddressChooser
  - PhotoChooser

Code Sample – Making a Phone call

```
var phoneCallTask = new PhoneCallTask()
        {
                DisplayName = "John Smith",
                PhoneNumber = "555 555 1212"
        };

phoneCallTask.Show();
```

Silverlight Tools Example

- Hello Phone Demo

Application marketplace

- **Enhanced Discovery**
  - Easy to find great applications, games
  - Fun to browse the marketplace
- **Simple,Reliable Acquisition**
  - Try before you buy
  - Buying and downloading content is a breeze
  - Payment flexibility (MO billing and credit card)
  - Easy application updates

Application Development Lifecycle
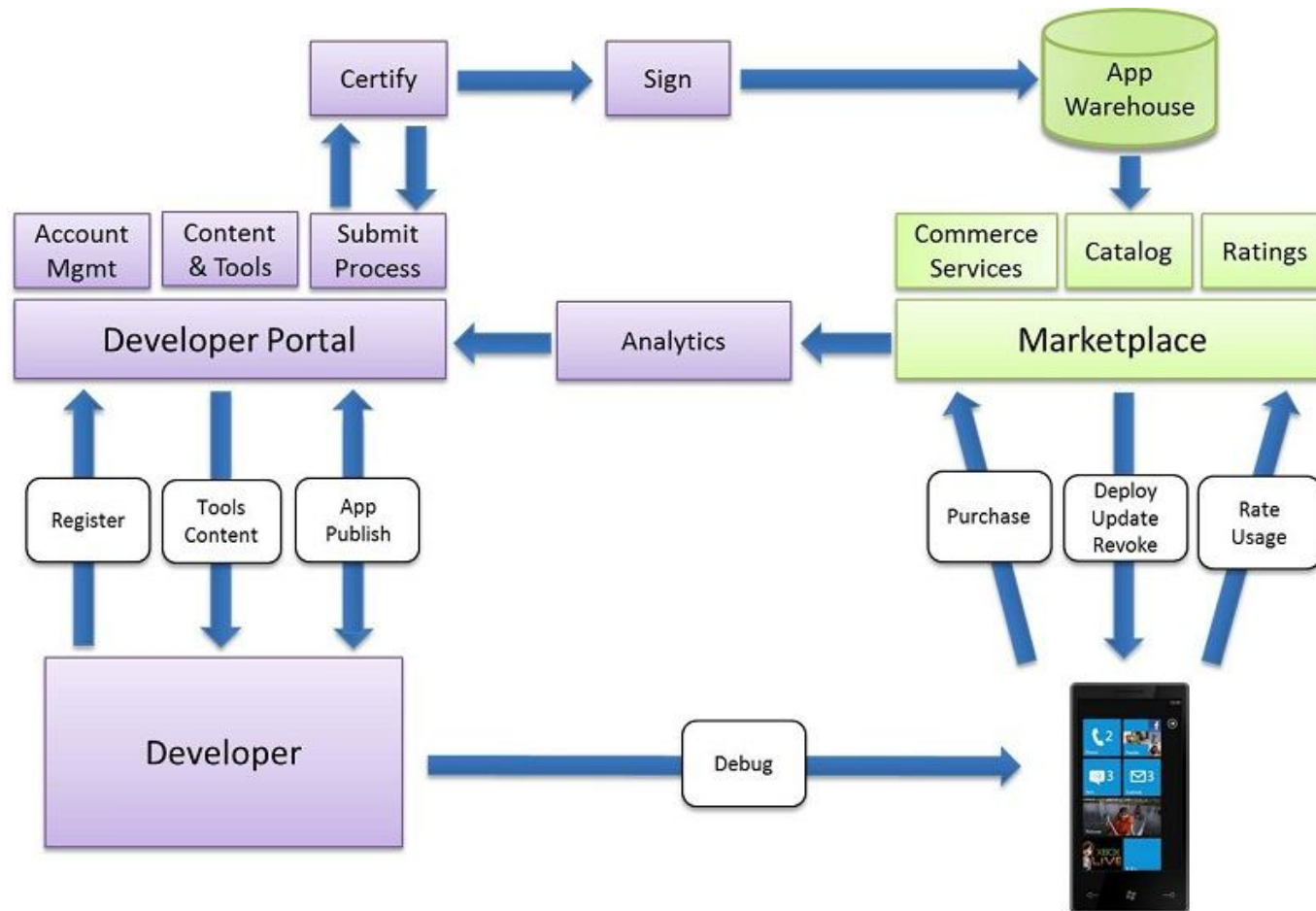
Aspects of a Mobile application

Phone Services

Navigation and Structure

Input

Output

Storage

Aspects of a Silverlight WP7 application

- Phone Services
  - Launchers and Choosers
- Navigation and Structure
  - Start Screen Tiles
  - Application pages
  - Application Bar and Menu
  - Navigation Services
  - Navigation Controls
- Input
  - Data Input Controls
  - Multitouch Controls
- Output
  - List Controls
  - Content Controls
- Storage
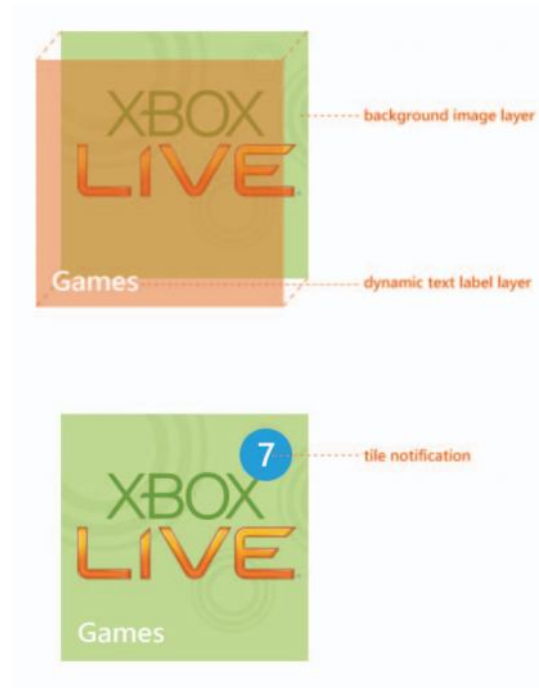 - Isolated Storage
  - Cloud

Phone Services

- Making phone calls
- Accessing network
- Taking and Sharing pictures
- Listening to Music
- Messaging, Sending Email, SMS, Social Networks
- Sensors, GPS, Accelerometer, compass and gyroscope
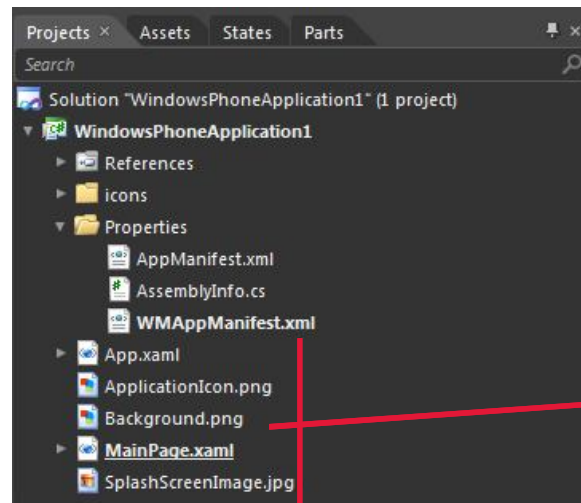
Navigation Structure

Start Screen tiles

- Provide access to
  - Applications
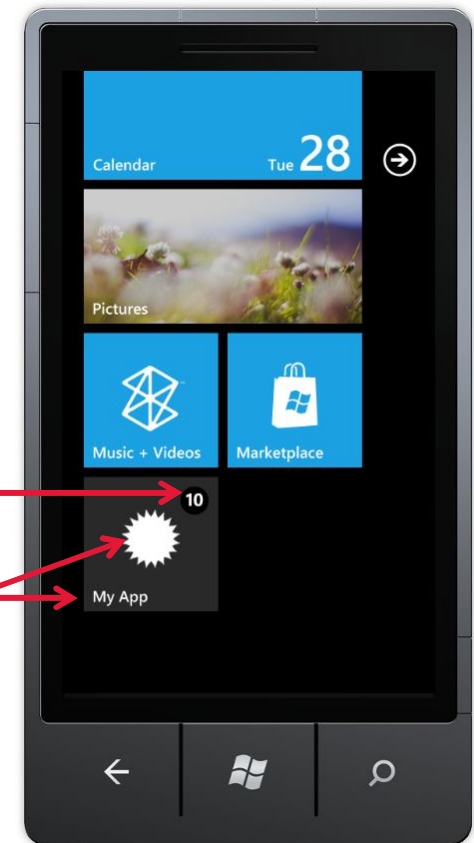 - Contacts
  - Web Pages
- Live Tiles

Navigation and Structure

Creating a Tile for the application
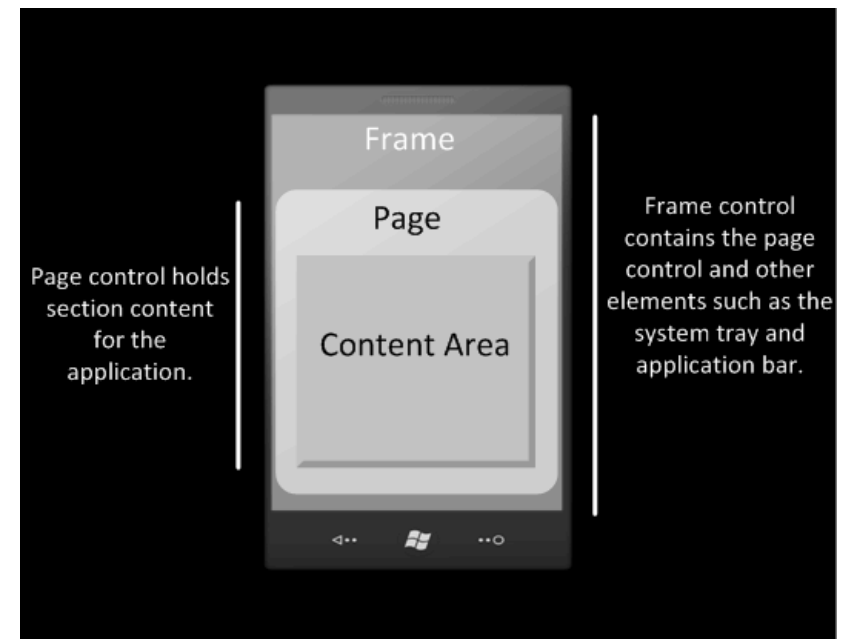
Navigation and Structure:

Pages and Frames

- Frame Control
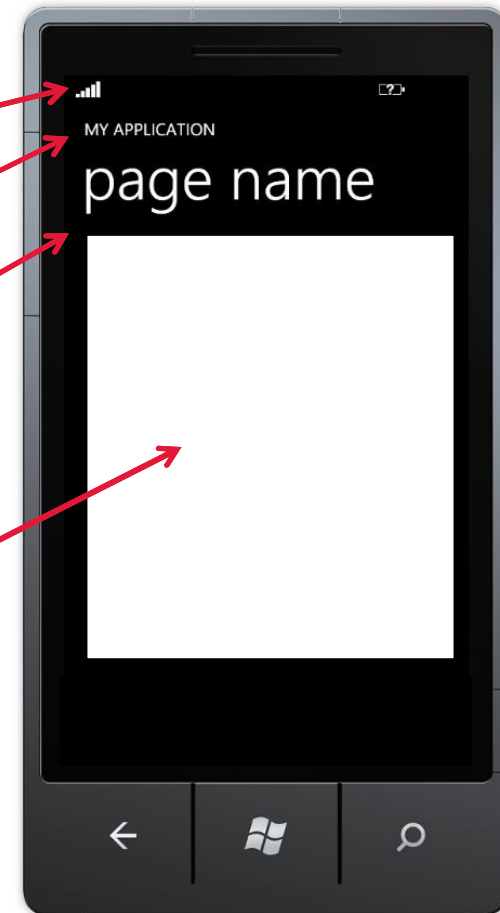 - One per application
  - Top level container control
- Page Control
  - Appear as separate screens to the user
  - 1 or more per application
  - Contained in the frame control

Navigation and Structure

Application Pages

- System Tray

- Application Title

- Page Name

- Content Area

MY APPLICATION

page name

Navigation and Structure

Application Bar

- Icon Buttons

- Menu Items
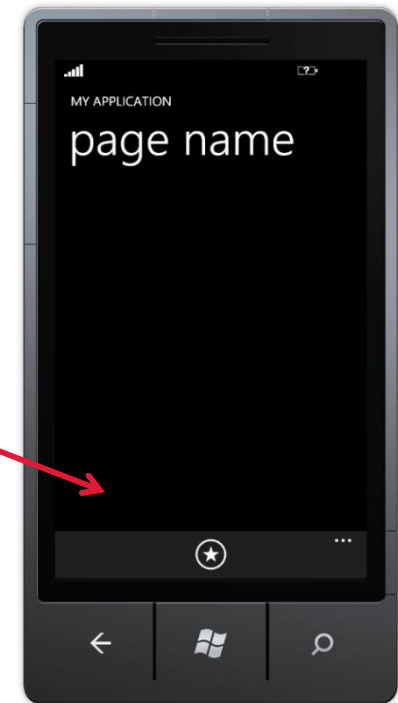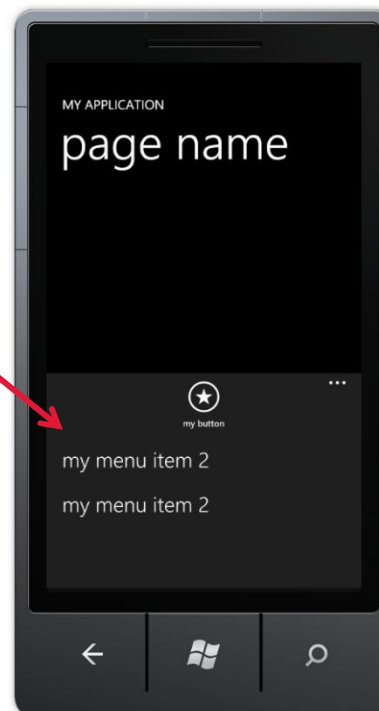
Navigation and Structure

Navigation Services

- Spoke and Hub
- System keeps track of back-stack
- Backstack can be manipulated from code
- 4 methods of navigation
  - HyperLinkButton
  - NavigationService.Navigate(Uri, UriKind)
  - Back Button – navigates using the back-stack
  - NavigationService.GoBack() – same as back button

Navigation Model Demo

Navigation and Structure

Navigation Controls

- Pivot Control - Filtering
- Panorama Control – Providing Bigger Screen Experience

Navigation and Structure

Pivot Control

- Provides a way to switch between views
- Good for filtering large datasets
- Views are positioned horizontally – swipe left and right
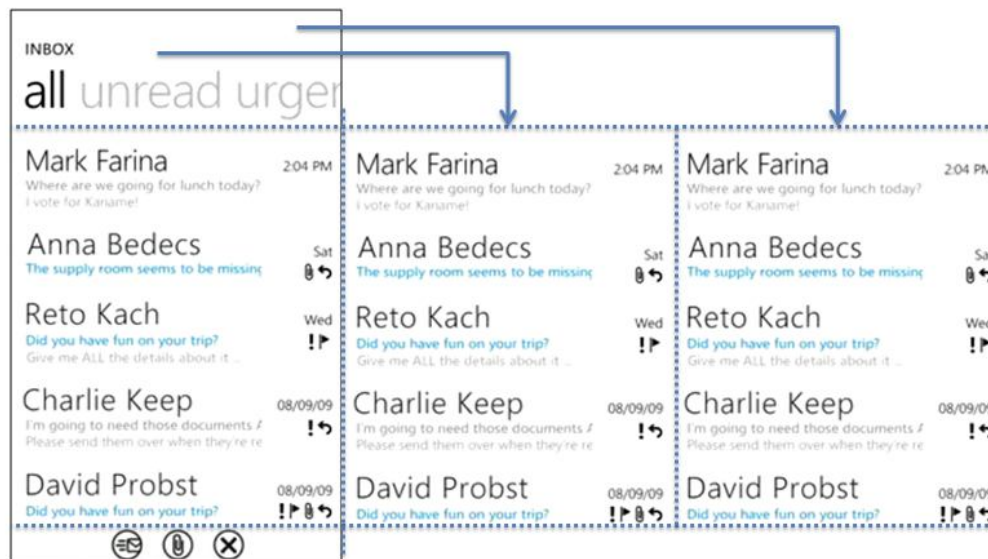
Pivot Demo

Navigation:

Panorama Control

- Long horizontal canvas
- Extends beyond the confines of the screen
- Layered animations
- Rich background image

Panorama Demo

Input Controls

- Button
- TextBox
- DropDown
- RadioButton
- CheckBox
- Slider

Input

TextBox and Keyboard (SIP)

- Software Input Panel
- TextBox Control has optional input scope
- InputScope optimizes the SIP based on type of data expected

```
<TextBox InputScope="TelephoneNumber"/>
```

SIP optimized for phone number entry
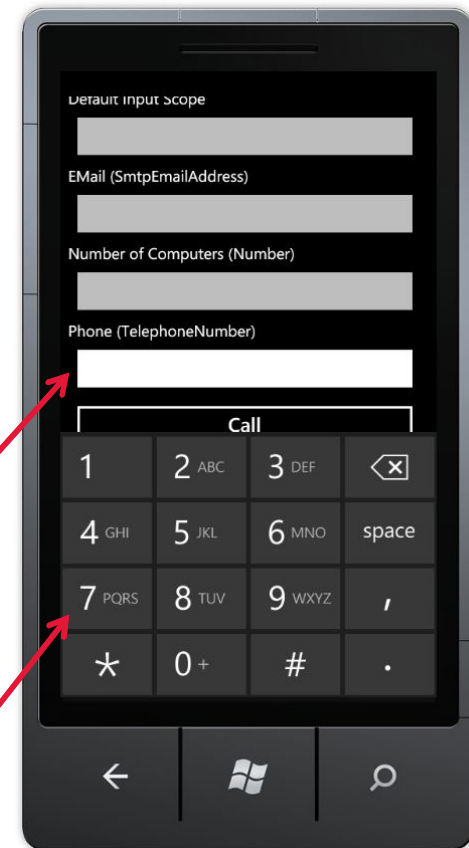
SIP Demo

Input

Gestures

- All Windows Phone 7 devices support capacitive touch
- 4 touch points
- Support 6 gestures
  - Tap
  - Double Tap
  - Pan
  - Flick
  - Pinch and Stretch
  - Touch and Hold

Data Output

Controls

- ListBox
- Textarea
- WebBrowser
- ProgressBar

Storage Options

- Isolated Storage
- Local Database
- Cloud Storage

Data Storage Options

Isolated Storage

- Provides safe application-specific data storage
- 2 storage mechanisms
  - Settings : Simple
  - File based : More complex
- Some usage examples
  - Application State
  - User Preferences and settings
  - Cached Data
- Limited only by available space

Isolated Storage Demo

Data Storage

Local Database

- Ideal for disconnected scenarios
- SQL CE (Compact Edition)
- LINQ to SQL
- 512 MB maximum size
- Supports encryption of entire database
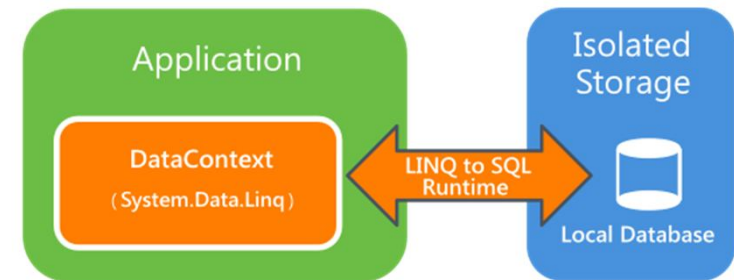
Data Storage

Cloud

- Store data in the cloud using a web service
- Allows you to have data follow the user to other devices
- Enables social media applications – data shared between users
- Data is automatically backed up for the user

Multimedia Integration

Audio

Video

HMTL and Images

Maps

Integrating Media

- Three methods
  - Media Player Launcher
  - Media Element Control
  - Background Audio Player

Media Launcher Demo

Media Element Demo

HTML Delivery Options

- Locally stored content
- Web Server
- Programmatically created

Web Browser Control Demo

Mapping applications

- Road Maps
- Aerial photography
- Attractions
- Driving directions
- GPS

Bing Maps Demo

Local Database

- Relational Database
  - Based on SQL CE
  - Tables, Columns, Indexes and Associations
- DataContext - proxy
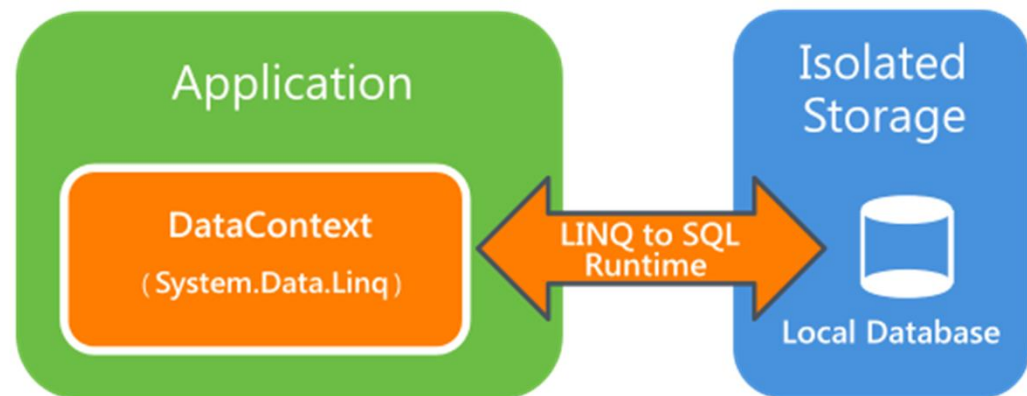
Local Database

- ORM
  - LINQ to SQL
  - Transact SQL not supported
- Not shared between applications

```
// Define query to gather all of the to-do items.
var toDoItemsInDB = from ToDoItem todo in toDoDB.ToDoItems
                    select todo;

// Execute query and place results into a collection.
ToDoItems = new ObservableCollection<ToDoItem>(toDoItemsInDB);
```

Local Database (contd.)

- Modifying the database once deployed

```
using (ToDoDataContext db = new ToDoDataContext(("isostore:/ToDo.sdf")))
{
        //Create the database schema updater
        DatabaseSchemaUpdater dbUpdate = db.CreateDatabaseSchemaUpdater();

        //Get database version
        int dbVersion = dbUpdate.DatabaseSchemaVersion;

        //Update database as applicable
        if (dbVersion < 5)
        {   //Copy data from existing database to new database
            MigrateDatabaseToLatestVersion();
        }
        else if (dbVersion == 5)
        {   //Add column to existing database to match the data context
            dbUpdate.AddColumn<ToDoItem>("TaskURL");
            dbUpdate.DatabaseSchemaVersion = 6;
            dbUpdate.Execute();
        }
}
```

Local Database Demo

Scheduled Actions

Background Agents

- One background agent per application
- 2 Types
  - Periodic : short execution duration : 25 secs
  - Resource Intensive : longer duration : 10 mins
- Can be registered as either type or both
- Memory usage cap – 6 MB
- Reschedule required every 2 weeks
- Frequency:
  - Periodic : every 30 min
  - Resource Intensive: must meet certain criteria
    - External Power
    - Wireless or USB connection, non-cellular
    - Battery 90%+
    - No active phone call

Scheduled Actions

Background Agents

```csharp
protected override void OnInvoke(ScheduledTask task)
{
    //TODO: Add code to perform your task in background
    string toastMessage = "";

    // If your application uses both PeriodicTask and ResourceIntensiveTask
    // you can branch your application code here. Otherwise, you don't need to.
    if (task is PeriodicTask)
    {
        // Execute periodic task actions here.
        toastMessage = "Periodic task running.";
    }
    else
    {
        // Execute resource-intensive task actions here.
        toastMessage = "Resource-intensive task running.";
    }

    // Launch a toast to show that the agent is running.
    // The toast will not be shown if the foreground application is running.
    ShellToast toast = new ShellToast();
    toast.Title = "Background Agent Sample";
    toast.Content = toastMessage;
    toast.Show();

    // If debugging is enabled, launch the agent again in one minute.
#if DEBUG_AGENT
    ScheduledActionService.LaunchForTest(task.Name, TimeSpan.FromSeconds(60));
#endif

    // Call NotifyComplete to let the system know the agent is done working.
    NotifyComplete();
}
```
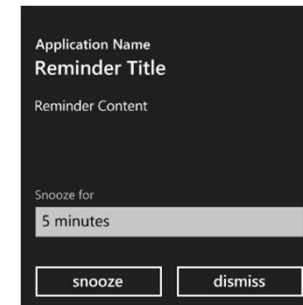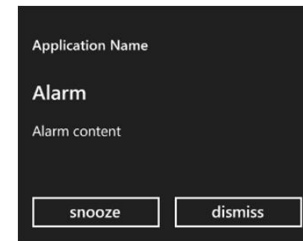
Scheduled Notifications

- A dialog box that appears at a specified time
- 2 Types of Scheduled Notifications
  - Alarm
    - Always displays Alarm as title
    - Application can set alarm sound
    - Tap outside of buttons – app launched
  - Reminder
    - Title is set by application
    - Default notification sound
    - Tap outside of buttons – optional deep link
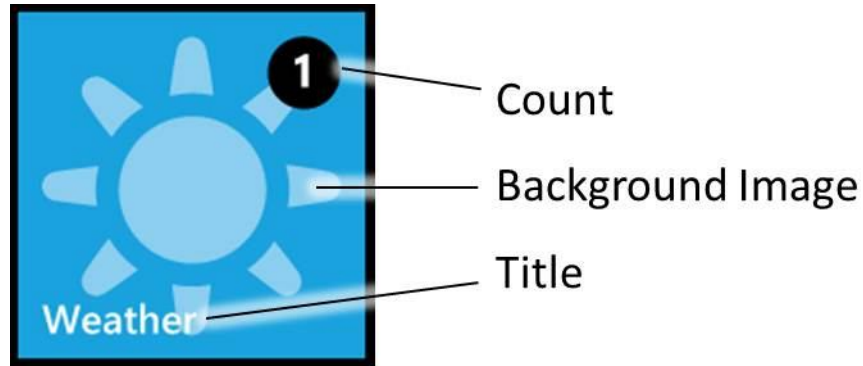
Alarm and Reminder Demo

Live Tile Enhancements

- Two types of Tiles
- Application
  - Created by user pinning application to Start
  - Cannot be deleted by application; only uninstall or unpinning
  - Initial properties set in application manifest
- Secondary
  - Created programmatically from user interaction
  - Can be deleted programmatically, unpinning or app uninstall
  - Deep link into specific application section

Live Tile Enhancements

- Now with optional "back" – flips between front and back sides

Front

Count
Background Image
Title
Weather

Back

It is your turn. — Back Content
Back Background Image
Back Title
Fun & Games
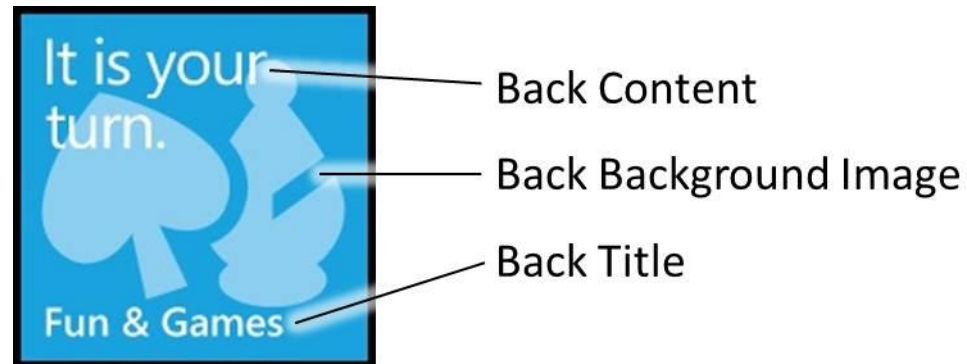
Application & Secondary Tiles Demo

Questions?

Thank You

Disclaimer

Tech Mahindra Limited, herein referred to as TechM provide a wide array of presentations and reports, with the contributions of various professionals. These presentations and reports are for informational purposes and private circulation only and do not constitute an offer to buy or sell any securities mentioned therein. They do not purport to be a complete description of the markets conditions or developments referred to in the material. While utmost care has been taken in preparing the above, we claim no responsibility for their accuracy. We shall not be liable for any direct or indirect losses arising from the use thereof and the viewers are requested to use the information contained herein at their own risk. These presentations and reports should not be reproduced, re-circulated, published in any media, website or otherwise, in any form or manner, in part or as a whole, without the express consent in writing of TechM or its subsidiaries. Any unauthorized use, disclosure or public dissemination of information contained herein is prohibited. Unless specifically noted, TechM is not responsible for the content of these presentations and/or the opinions of the presenters. Individual situations and local practices and standards may vary, so viewers and others utilizing information contained within a presentation are free to adopt differing standards and approaches as they see fit. You may not repackage or sell the presentation. Products and names mentioned in materials or presentations are the property of their respective owners and the mention of them does not constitute an endorsement by TechM. Information contained in a presentation hosted or promoted by TechM is provided "as is" without warranty of any kind, either expressed or implied, including any warranty of merchantability or fitness for a particular purpose. TechM assumes no liability or responsibility for the contents of a presentation or the opinions expressed by the presenters. All expressions of opinion are subject to change without notice.