Using Networks with Windows Phone

Rob S. Miles | Microsoft MVP | University of Hull, UK
Andy Wigley | Microsoft MVP | Appa Mundi

Session 9.0

1

Course Schedule

- Session 1 – Tuesday, August 23, 2011
  - Building Windows Phone Apps with Visual Studio 2010
  - Silverlight on Windows Phone—Introduction
  - Silverlight on Windows Phone—Advanced
  - Using Expression to Build Windows Phone Interfaces
  - Windows Phone Fast Application Switching
  - Windows Phone Multi-tasking & Background Tasks
  - Using Windows Phone Resources (Bing Maps, Camera, etc.)

- Session 2 – Wednesday, August 24, 2011
  - Application Data Storage on Windows Phone
  - **Using Networks with Windows Phone**
  - Tiles & Notifications on Windows Phone
  - XNA for Windows Phone
  - Selling a Windows Phone Application

NOKIA

Microsoft

2

Agenda

- Networking for Windows Phone

- WebClient

- HttpWebRequest

- Sockets

- OData

- Wire Serialization

NOKIA

Microsoft®

Networking on Windows Phone

Networking on Windows Phone

- Support for networking features
  - Windows Communication Foundation (WCF)
  - HttpWebRequest
  - WebClient (no cross-domain policy file required)
  - Sockets

- Not supported in this version
  - Custom WCF Bindings
  - WCF RIA Services
  - NTLM authentication

NOKIA

Microsoft®

WebClient

- Simple API for HTTP requests

- In Windows Phone 7.0, ran on the UI thread
  - Block UI operations
  - Avoid!

- In Windows Phone 7.5
  - Completely reworked
  - Now operates on originating thread

NOKIA

Microsoft®

Simple Http Operations – WebClient

```
WebClient client = new WebClient();
public MainPage() {
    InitializeComponent();
    client.DownloadProgressChanged += client_DownloadProgressChanged;
    client.DownloadStringCompleted += client_DownloadStringCompleted;
    client.OpenReadCompleted += client_OpenReadCompleted;
}
private void WebClientButton_Click(object sender, RoutedEventArgs e) {
    client.DownloadStringAsync(
        new Uri("http://localhost/ServicesApplication/rssdump.xml"), "sample rss");
}
void client_DownloadProgressChanged(object sender, DownloadProgressChangedEventArgs e){
    if (e.UserState as string == "sample rss"){
        this.DownloadProgress.Value = e.ProgressPercentage;
    }
}
void client_DownloadStringCompleted(object sender, DownloadStringCompletedEventArgs e){
    this.DownloadedText.Text = e.Result;
}
```

NOKIA   *Microsoft*

Demo

Simple HTTP Networking with WebClient

Windows Phone

8

More Control - HttpWebRequest

```
private void HttpWebRequestButton_Click(object sender, RoutedEventArgs e){
    var req = HttpWebRequest.Create(
        new Uri("http://localhost/ServicesApplication/rssdump.xml")) as HttpWebRequest;
    req.BeginGetResponse(HttpWebRequestButton_Callback, req);
}

private void HttpWebRequestButton_Callback(IAsyncResult result){
    var req = result.AsyncState as HttpWebRequest;
    var resp = req.EndGetResponse(result);
    var strm = resp.GetResponseStream();
    var reader = new StreamReader(strm);

    this.Dispatcher.BeginInvoke(() =>{
            this.DownloadedText.Text = reader.ReadToEnd();
            this.TextViewer.Visibility = System.Windows.Visibility.Visible;
        });
}
```
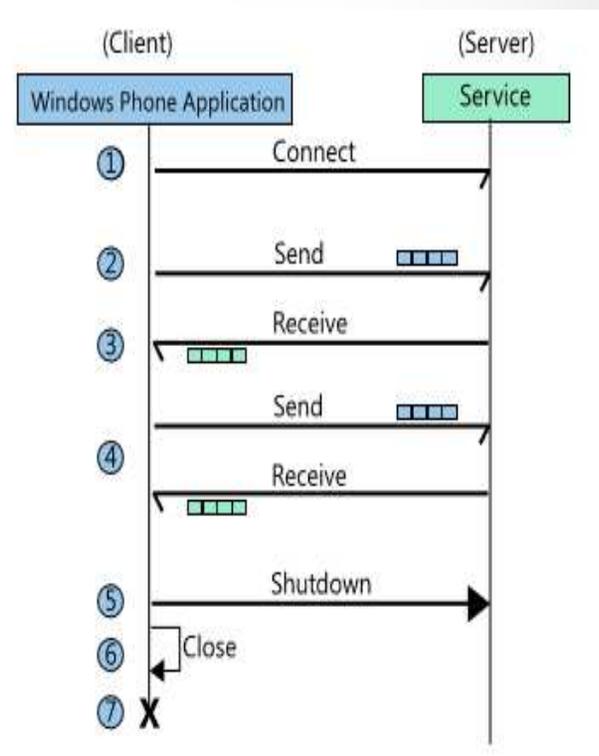
NOKIA

Microsoft®

Demo

HttpWebRequest

Windows Phone

10

Sockets

Sockets Support – Windows Phone 7.1

- TCP
  - Connection-oriented
  - Reliable Communication

- UDP Unicast, UDP Multicast
  - Connectionless
  - Not Reliable

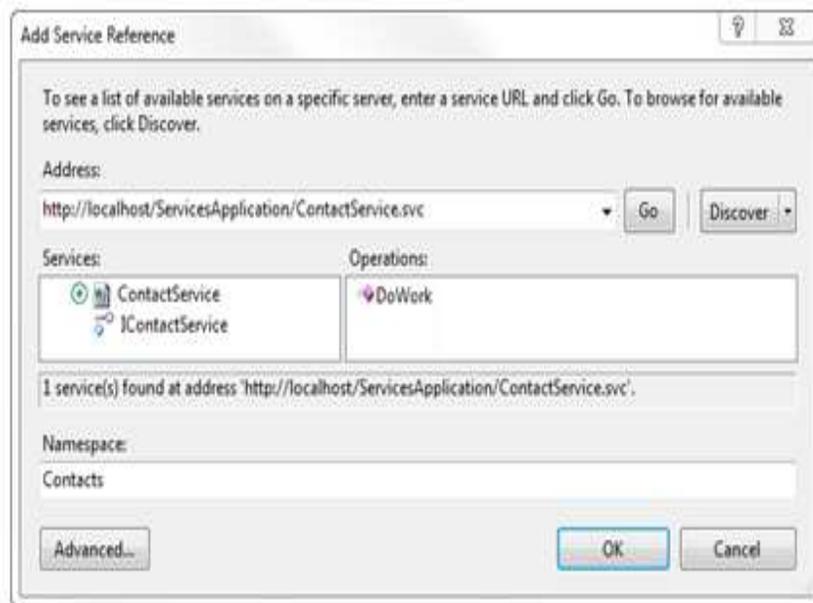- IPV4 supported e.g. 172.36.254.14
  - No IPV6 support

NOKIA

Microsoft

Windows Phone

Demo

Sockets

13

WCF/ASMX Services

- Can 'Add Reference' from Windows Phone projects to automatically generate proxy classes
  - ASMX should 'just work'
  - WCF requires that you use basicHttpBinding

14 **NOKIA**

Microsoft

RESTful Web Services

Building them

- Rather than building "walled gardens," data should be published in a way that allows it to reach the broadest range of mobile clients

- Old-style ASMX SOAP 1.1 Web Services using ASP.NET or Windows Communication Foundation (WCF) require clients to implement SOAP protocol

- With Windows Phone 7 and Silverlight, we use WCF with BasicHttpBinding both on-premise and as a Web Role in Windows Azure to publish our data from local and cloud-based data sources like SQL Azure

- Recommend using lightweight REST + JSON Web Services that are better optimized for high-latency, slow, intermittent wireless data connections

15 NOKIA

Microsoft®

WCF Data Services: OData

- WCF Data Services provide an extensible tool for publishing data using a REST-based interface
  - Publishes and consumes data using the OData web protocol (http://www.odata.org)
  - Formatted in XML or JSON

- OData Client Library (DataServicesClient) now included in Windows Phone OS 7.1 SDK

```xml
<?xml version="1.0" encoding="utf-8" standalone="yes" ?>
<service xml:base=
    "http://odata.netflix.com/Catalog/ ">
 <workspace>
  <atom:title>Default</atom:title>
  <collection href="Genres">
    <atom:title>Genres</atom:title>
  </collection>
  <collection href="Titles">
    <atom:title>Titles</atom:title>
  </collection>
...
  </workspace>
 </service>
```

NOKIA

Microsoft

Generate Client Proxy

- In most cases, Add Service Reference will just work

- Alternatively, open a command prompt as administrator and navigate to **%windir%\Microsoft.NET\Framework\v4.0.30128**

- Run this command
  DataSvcutil.exe /uri:http://odata.netflix.com/Catalog/
  /DataServiceCollection /Version:2.0/out:netflixClientTypes

- Add generated file to your project

NOKIA

Microsoft®

Fetching Data

```
public class NetflixViewModel : INotifyPropertyChanged {
    private DataServiceCollection<Title> _titles;
    private NetflixCatalog context;

    public NetflixViewModel()      {
        context = new NetflixCatalog(new Uri("http://odata.netflix.com/Catalog/"));
        _titles = new DataServiceCollection<Title>();
    }

    public void GetTitles(int numberOfTitles)     {
      _titles.Clear();
      _titles.LoadAsync(context.Titles.Take(numberOfTitles));
    }

    void Titles_LoadCompleted(object sender, LoadCompletedEventArgs e)          {
        Titles.Clear();
        IEnumerable<Title> titles = e.QueryOperationResponse as IEnumerable<Title>;
        foreach (Title title in titles)
        {
            Titles.Add(title);
        }
    }
}
```

NOKIA

Microsoft

Demo

OData Services

19

Network Awareness

Making Decisions based on Data Connections

- Mobile apps shouldn't diminish the user experience by trying to send or receive data in the absence of network connectivity

- Mobile apps should be intelligent about performing heavy data transfers or lightweight remote method calls only when the appropriate data connection is available

- With Windows Phone 7 and Silverlight, we use the NetworkInterfaceType object to detect network type and speed and the NetworkChange object to fire events when the network state changes

NOKIA

Microsoft®

NetworkInformation in 7.1

- All in `Microsoft.Phone.Net.NetworkInformation` namespace

- Determine the Network Operator:
    - `DeviceNetworkInformation.CellularMobileOperator`

- Determine the Network Capabilities:
    - `DeviceNetworkInformation.IsNetworkAvailable`
    - `DeviceNetworkInformation.IsCellularDataEnabled`
    - `DeviceNetworkInformation.IsCellularDataRoamingEnabled`
    - `DeviceNetworkInformation.IsWiFiEnabled`

- Get Connection Information about a socket:
    - `NetworkInterfaceInfo netInterfaceInfo =`
        `socket.GetCurrentNetworkInterface();`

- Set preferred network interface for a socket
    - `socket.SetNetworkPreference(`
        `NetworkSelectionCharacteristics.Cellular);`

NOKIA

Microsoft

Wire Serialization

- Mobile devices are often connected to poor quality network connections

- Best chance of success in network data transfers achieved by
    - Keep data volumes as small as possible
    - Use the most compact data serialization available
    - Avoid large data transfers

- Avoid transferring redundant data

- Design your protocol to only transfer precisely the data you need and no more
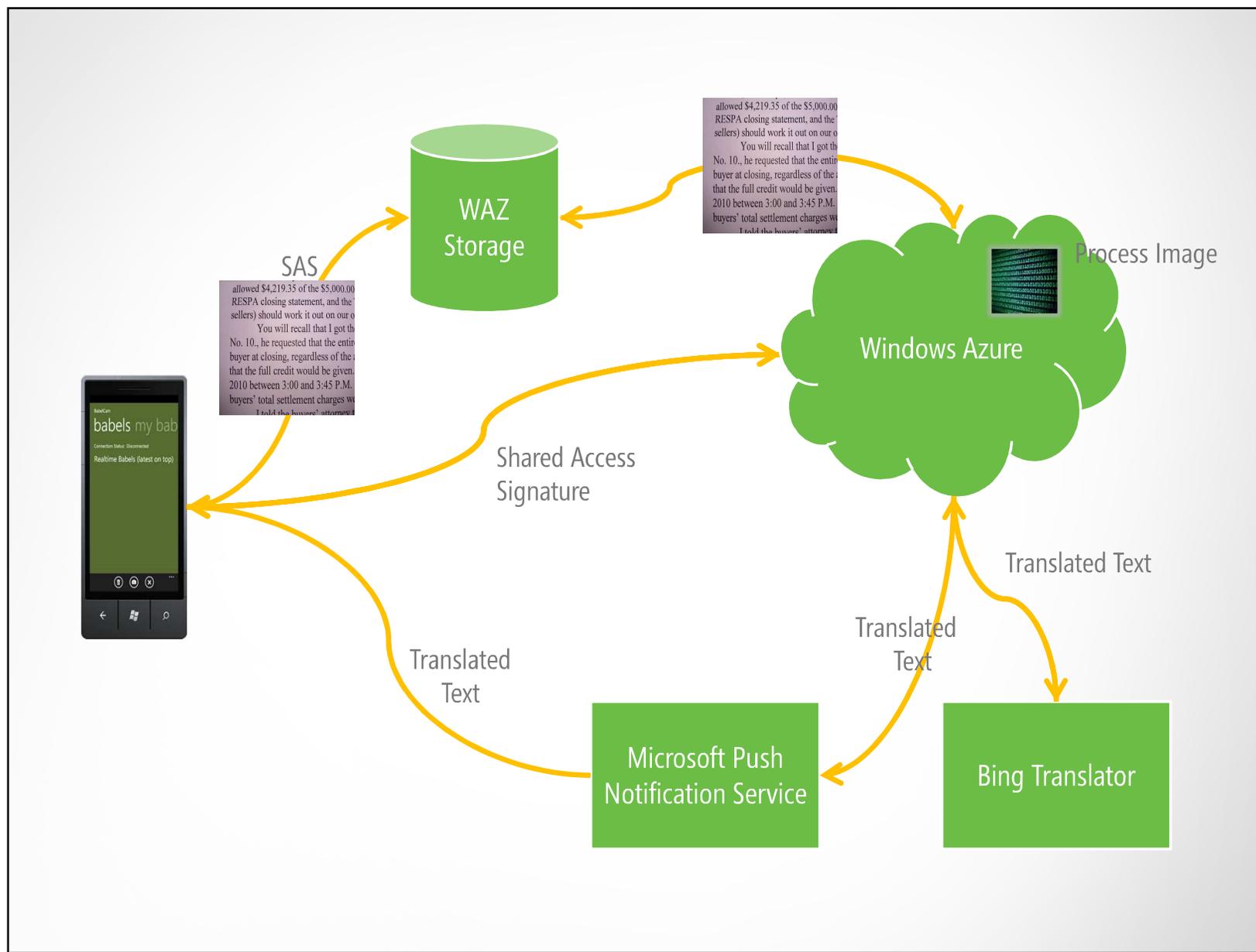
NOKIA

Microsoft®

Demo

Wire Serialization

23

Wire Serialization Affects Payroll Size

- Simple test case: Download 30 data records

- Each record just 12 fields

- Measured bytes to transfer

Wire Serialization Format	Size in Bytes
ASMX SOAP – DataSet (XML)	39670
ODATA XML	73786
ODATA JSON	34030
REST + JSON	15540
REST + JSON GZip	8680

NOKIA

Microsoft®

Windows Azure and Windows Phone

Windows Azure Toolkits for Devices

- Make it easier for phone developers to use Windows Azure
  - WP7 – http://watoolkitwp7.codeplex.com
  - iOS - https://github.com/microsoft-dpe
  - Android (coming soon)

- Toolkits include:
  - Native libraries (e.g. .NET, Objective-C)
  - Samples
  - Project templates
  - Documentation

NOKIA

Summary

- Windows Phone has a sockets API to support connection-oriented and connectionless TCP/IP and UDP/IP networking

- Support for ASMX, WCF and REST Web Services

- DataServicesClient for OData service access out of the box in 7.1 SDK

- Consider JSON serialization for maximum data transfer efficiency

- Windows Azure Toolkit for Windows Phone eases interaction with Windows Azure

NOKIA

Microsoft®