Windows Phone

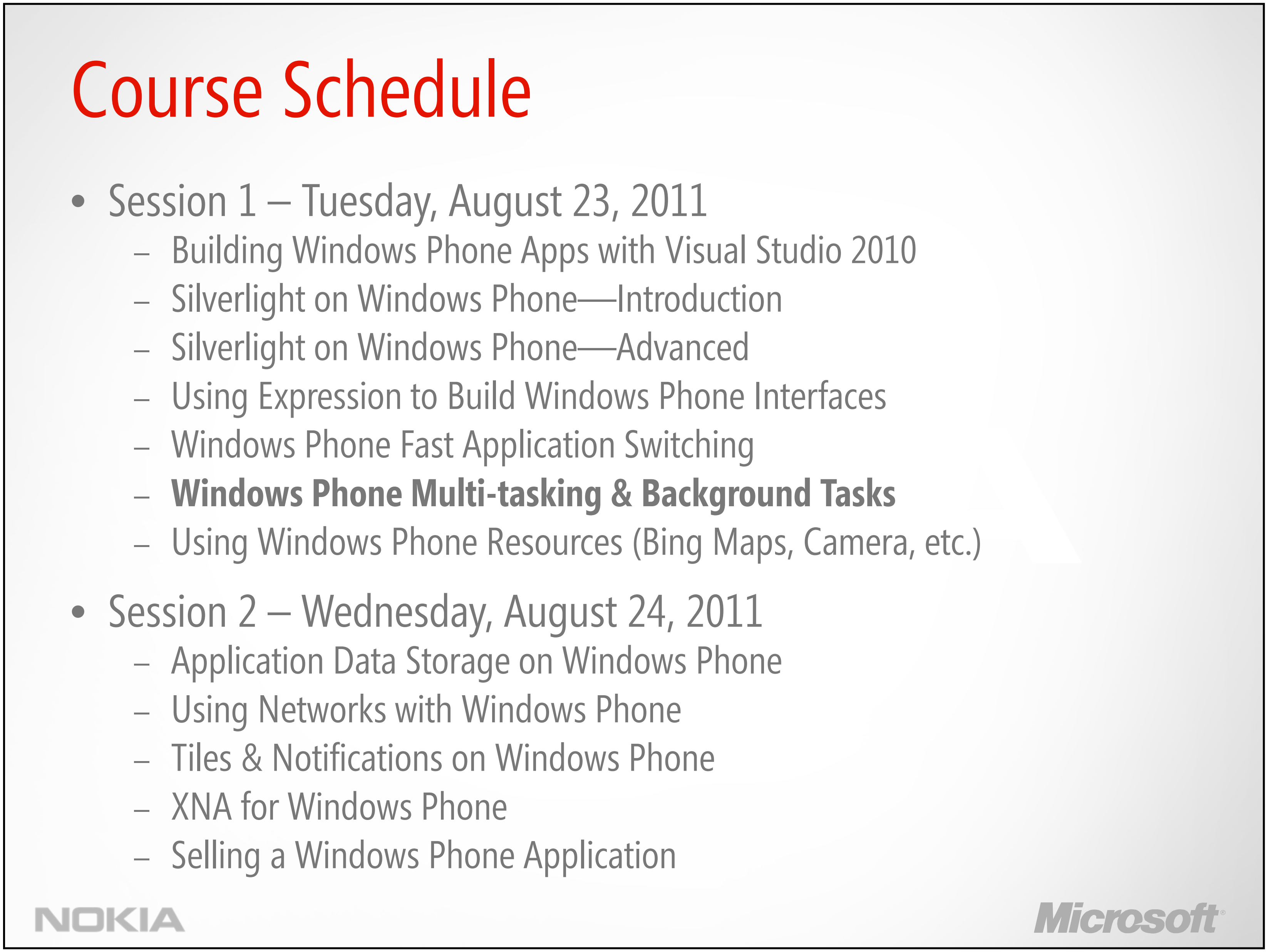
Windows Phone Background Tasks

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

Session 6.0

NOKIA
2
Phone
People
3
Messaging
9
Linked Inbox

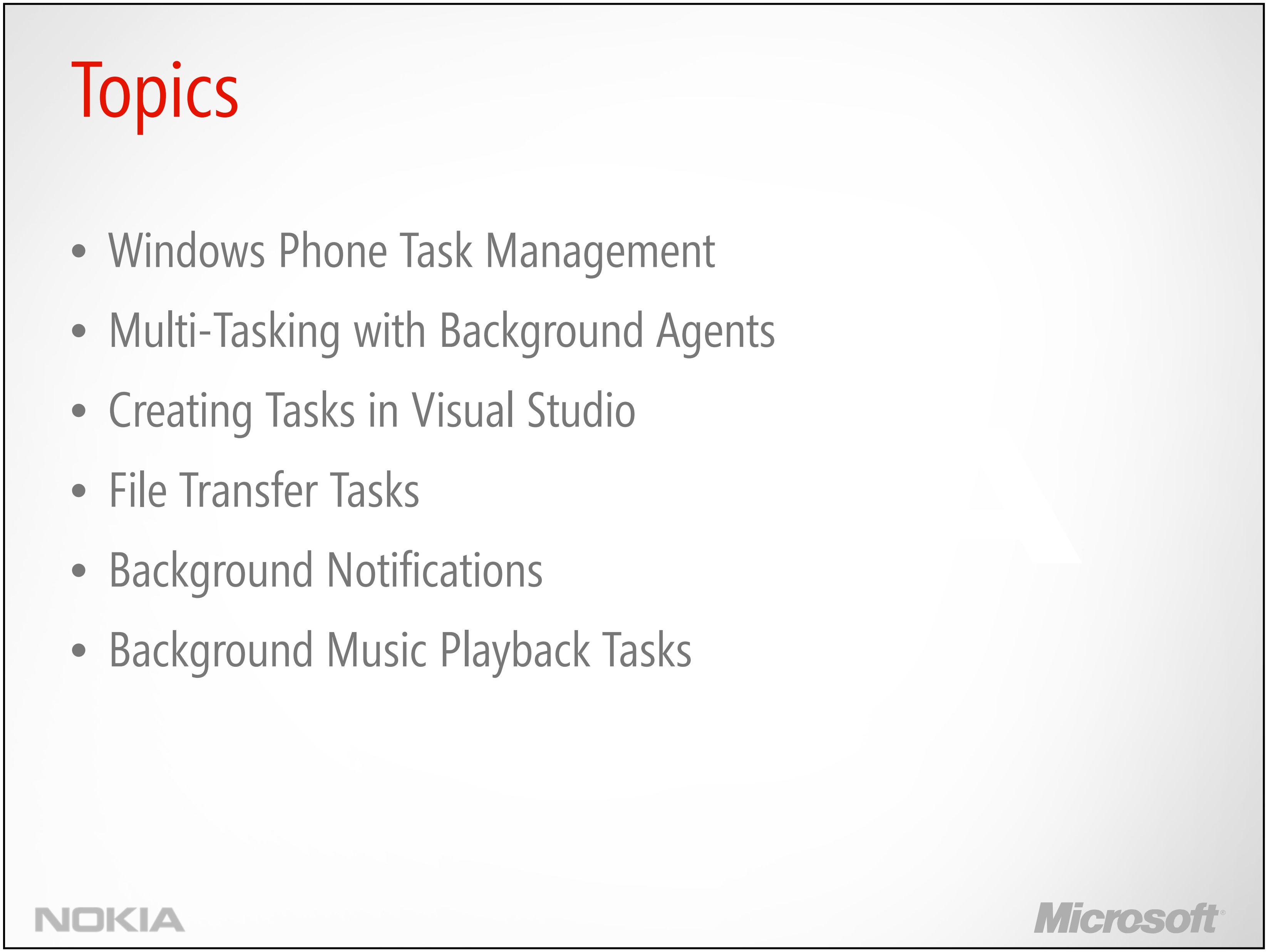
©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current 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.



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

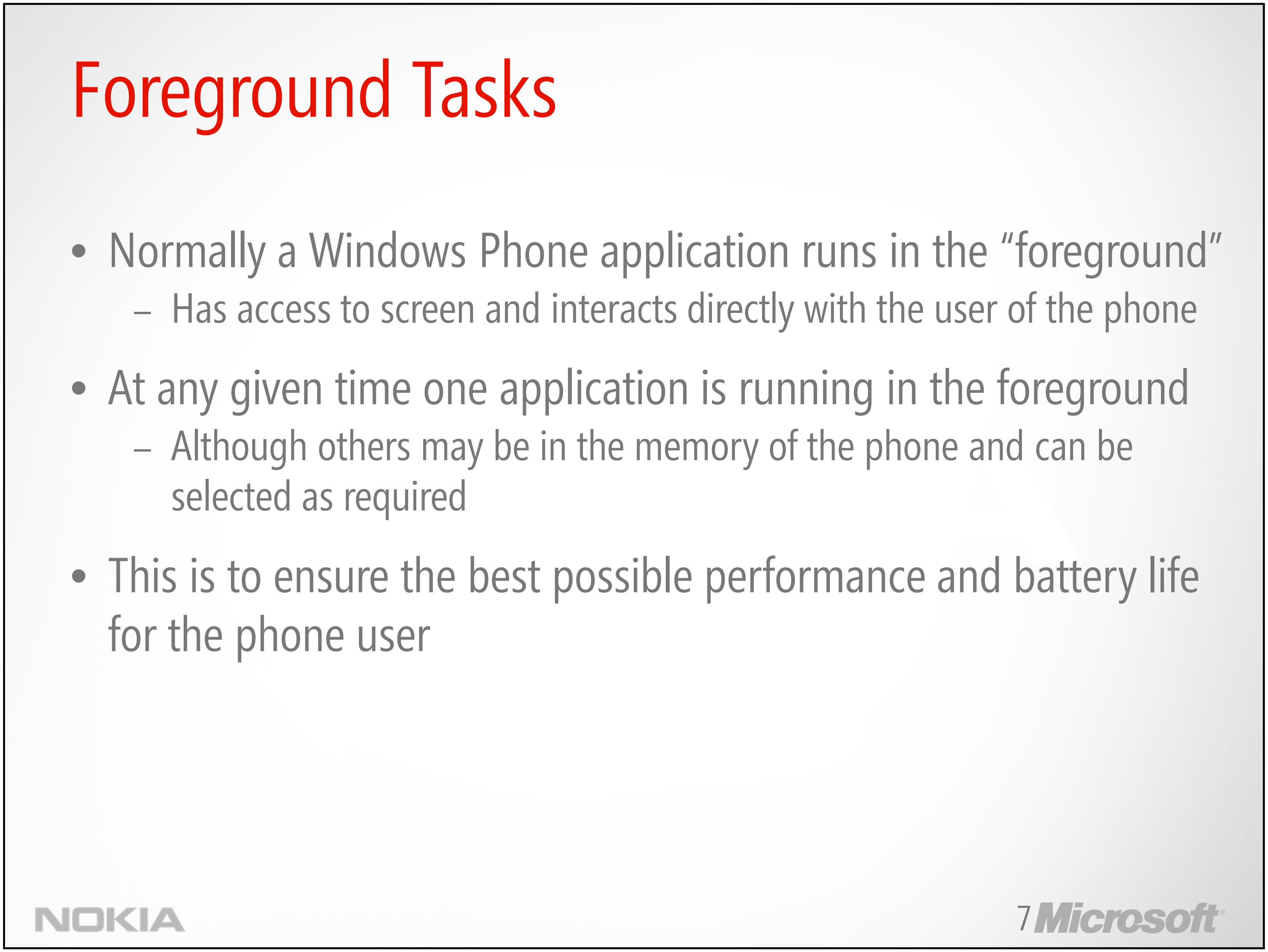
©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current 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.



Topics

- Windows Phone Task Management
- Multi-Tasking with Background Agents
- Creating Tasks in Visual Studio
- File Transfer Tasks
- Background Notifications
- Background Music Playback Tasks

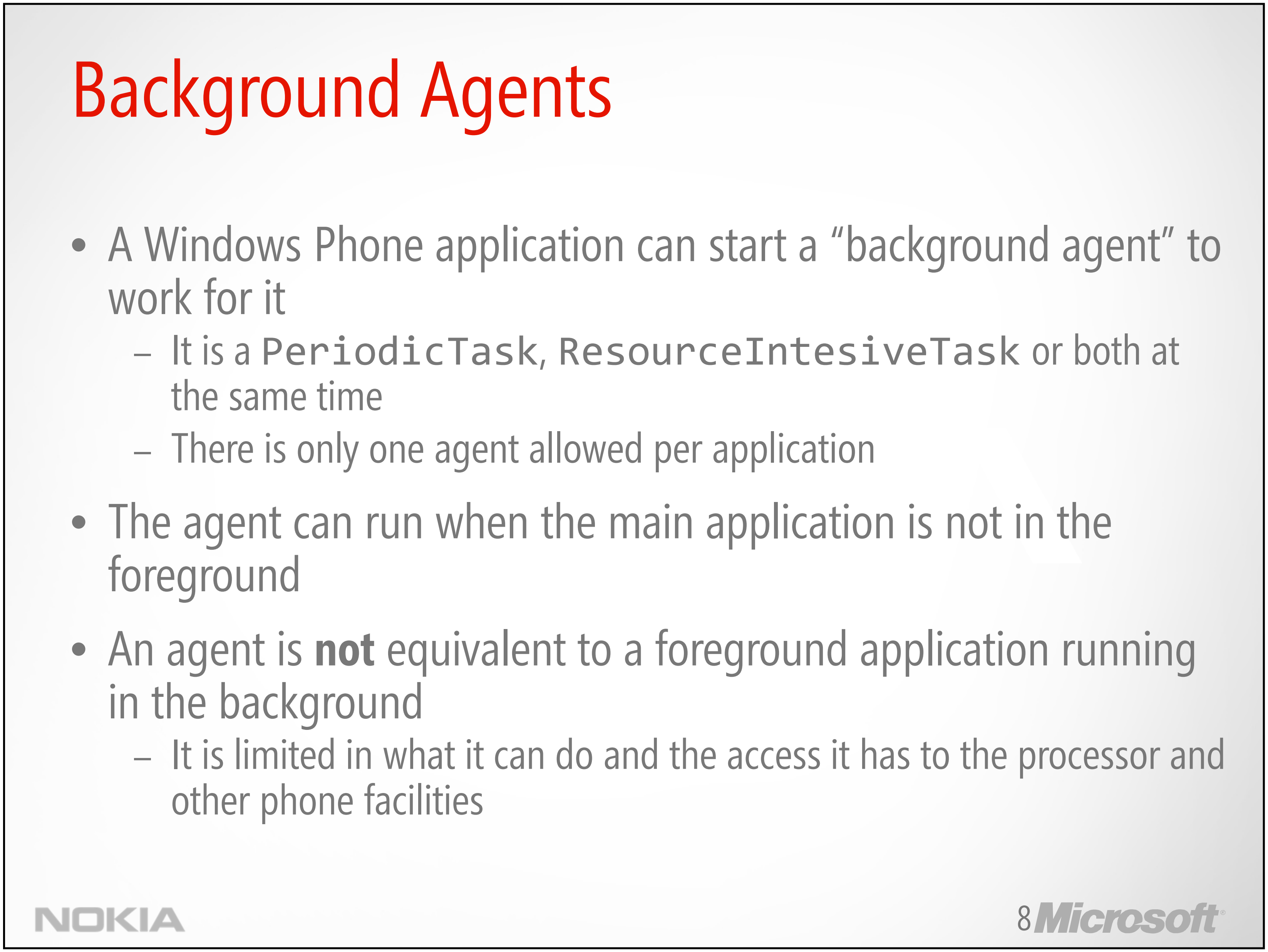
©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current 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.



Foreground Tasks

- Normally a Windows Phone application runs in the “foreground”
  - Has access to screen and interacts directly with the user of the phone
- At any given time one application is running in the foreground
  - Although others may be in the memory of the phone and can be selected as required
- This is to ensure the best possible performance and battery life for the phone user

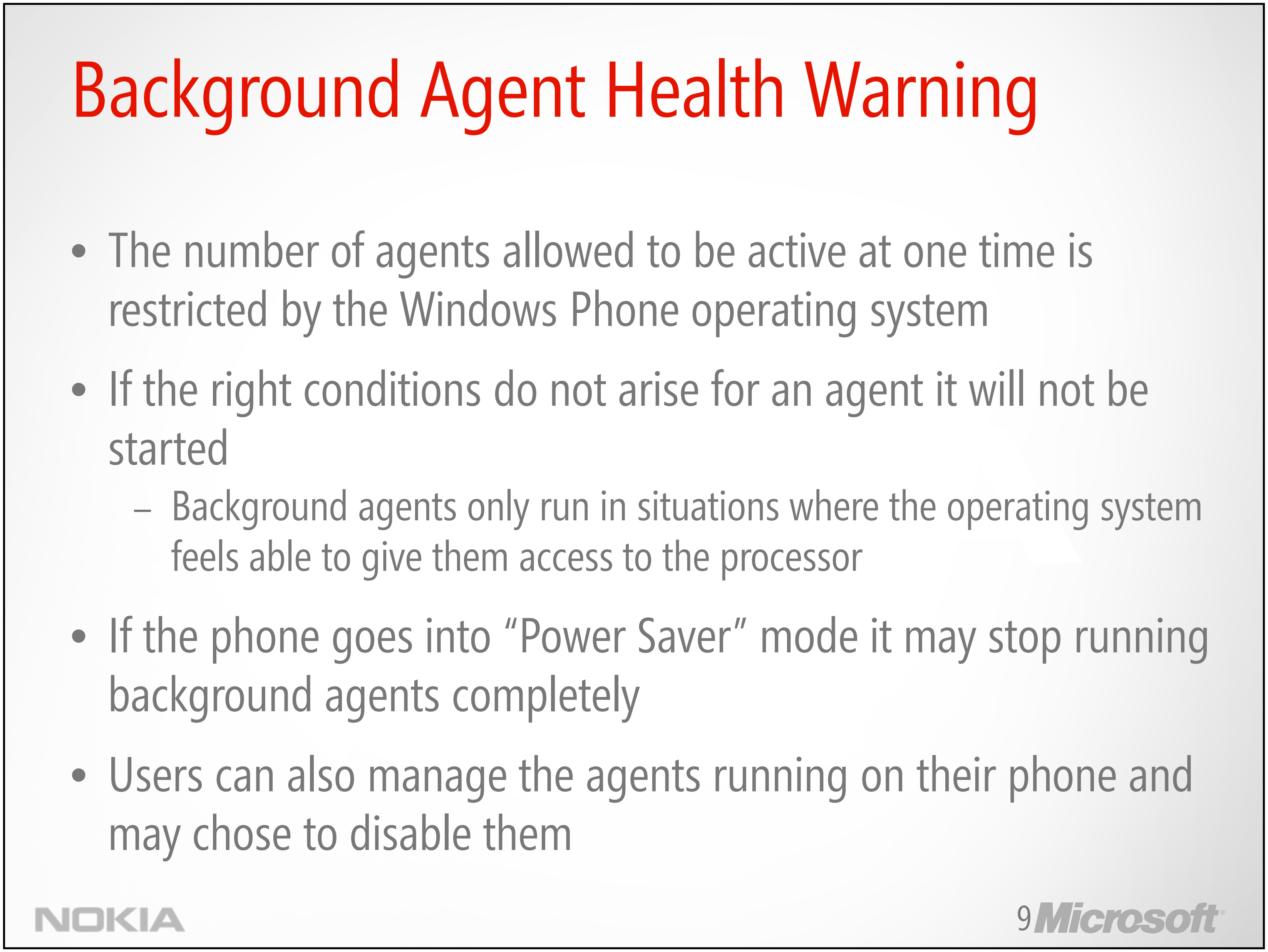
©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current 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.



Background Agents

- A Windows Phone application can start a “background agent” to work for it
  - It is a `PeriodicTask`, `ResourceIntesiveTask` or both at the same time
  - There is only one agent allowed per application
- The agent can run when the main application is not in the foreground
- An agent is **not** equivalent to a foreground application running in the background
  - It is limited in what it can do and the access it has to the processor and other phone facilities

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.

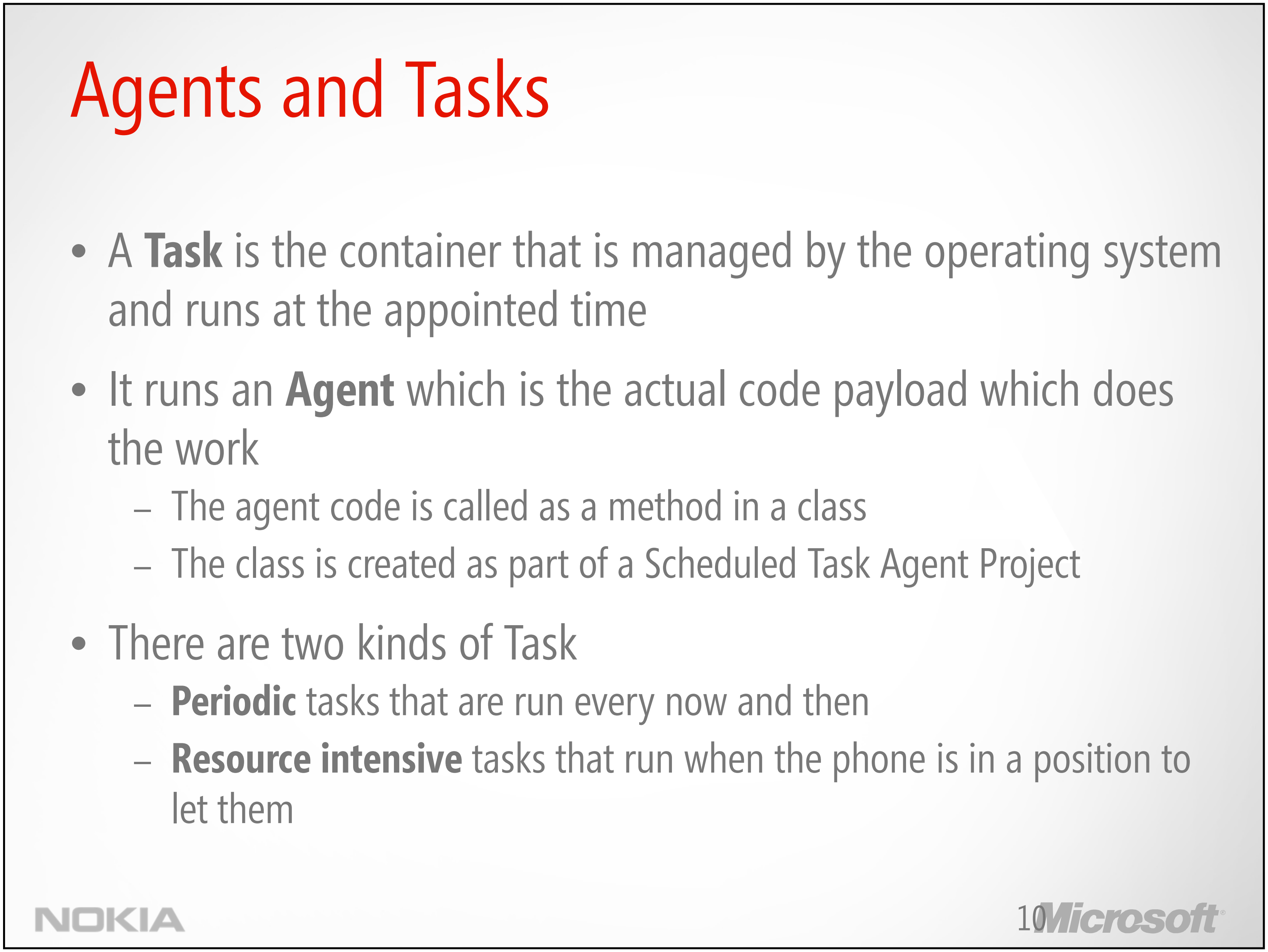


Background Agent Health Warning

- The number of agents allowed to be active at one time is restricted by the Windows Phone operating system
- If the right conditions do not arise for an agent it will not be started
  - Background agents only run in situations where the operating system feels able to give them access to the processor
- If the phone goes into “Power Saver” mode it may stop running background agents completely
- Users can also manage the agents running on their phone and may chose to disable them

9

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.

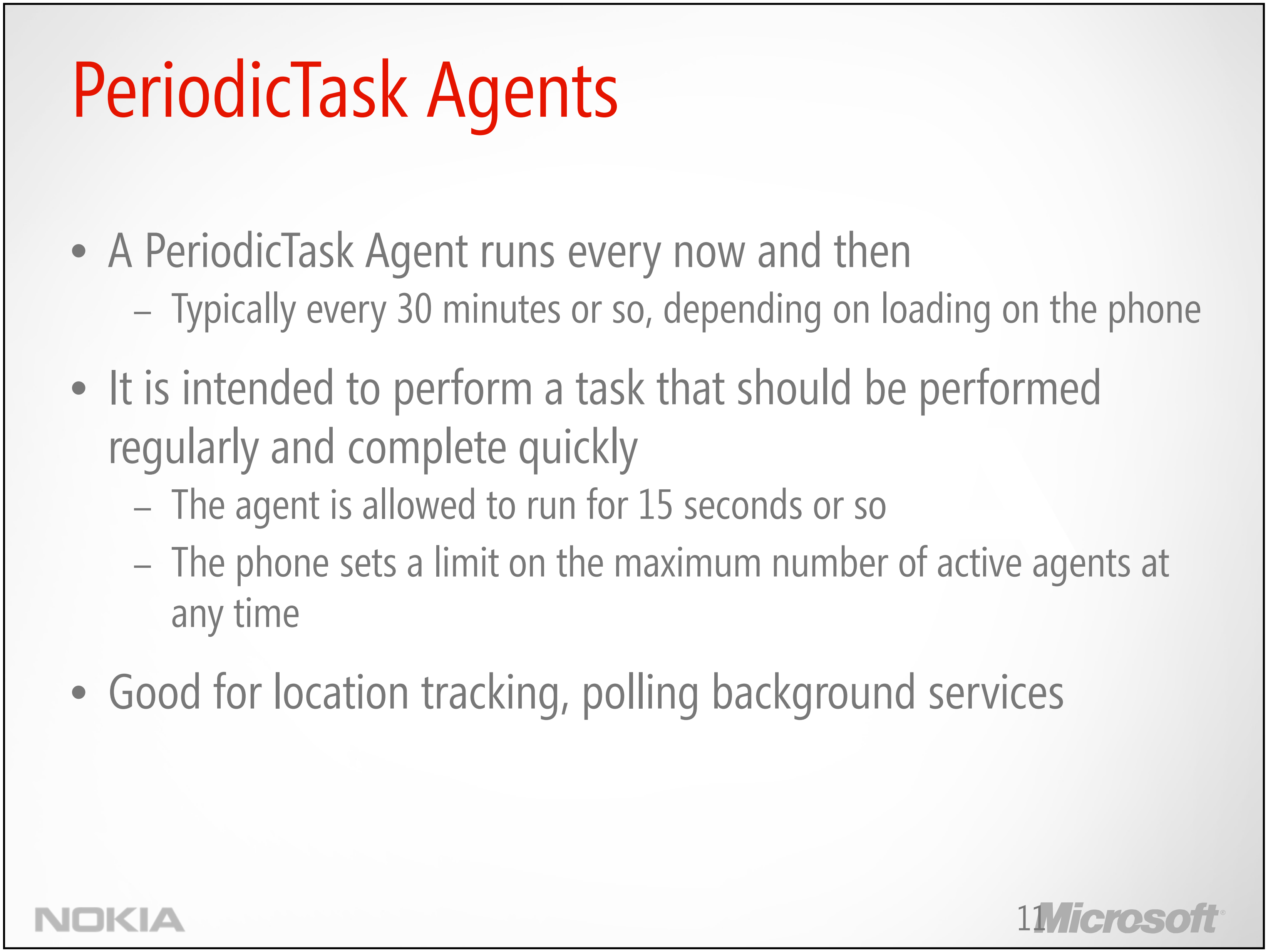


Agents and Tasks

- A **Task** is the container that is managed by the operating system and runs at the appointed time
- It runs an **Agent** which is the actual code payload which does the work
  - The agent code is called as a method in a class
  - The class is created as part of a Scheduled Task Agent Project
- There are two kinds of Task
  - **Periodic** tasks that are run every now and then
  - **Resource intensive** tasks that run when the phone is in a position to let them

10

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.

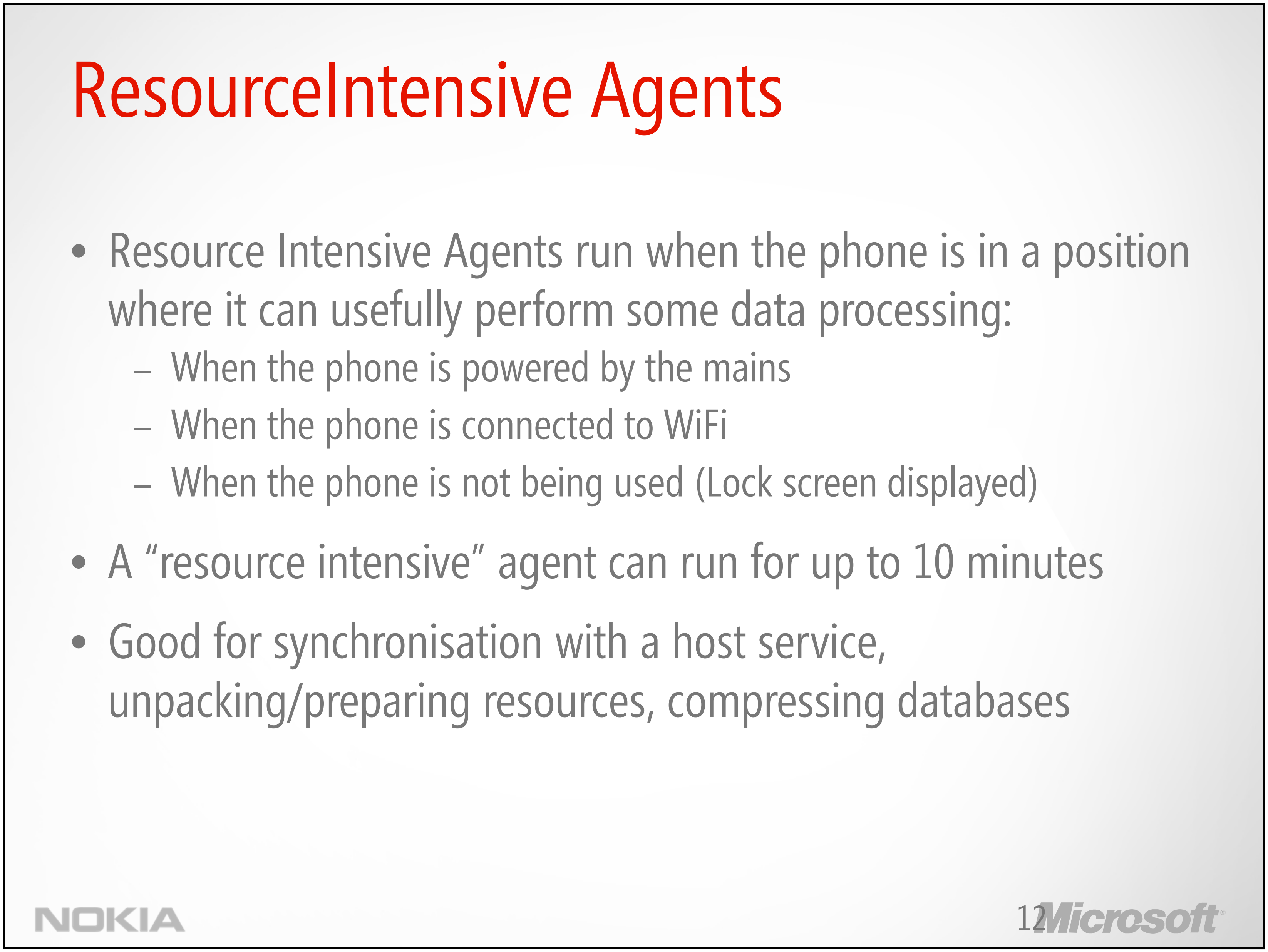


PeriodicTask Agents

- A PeriodicTask Agent runs every now and then
  - Typically every 30 minutes or so, depending on loading on the phone
- It is intended to perform a task that should be performed regularly and complete quickly
  - The agent is allowed to run for 15 seconds or so
  - The phone sets a limit on the maximum number of active agents at any time
- Good for location tracking, polling background services

11

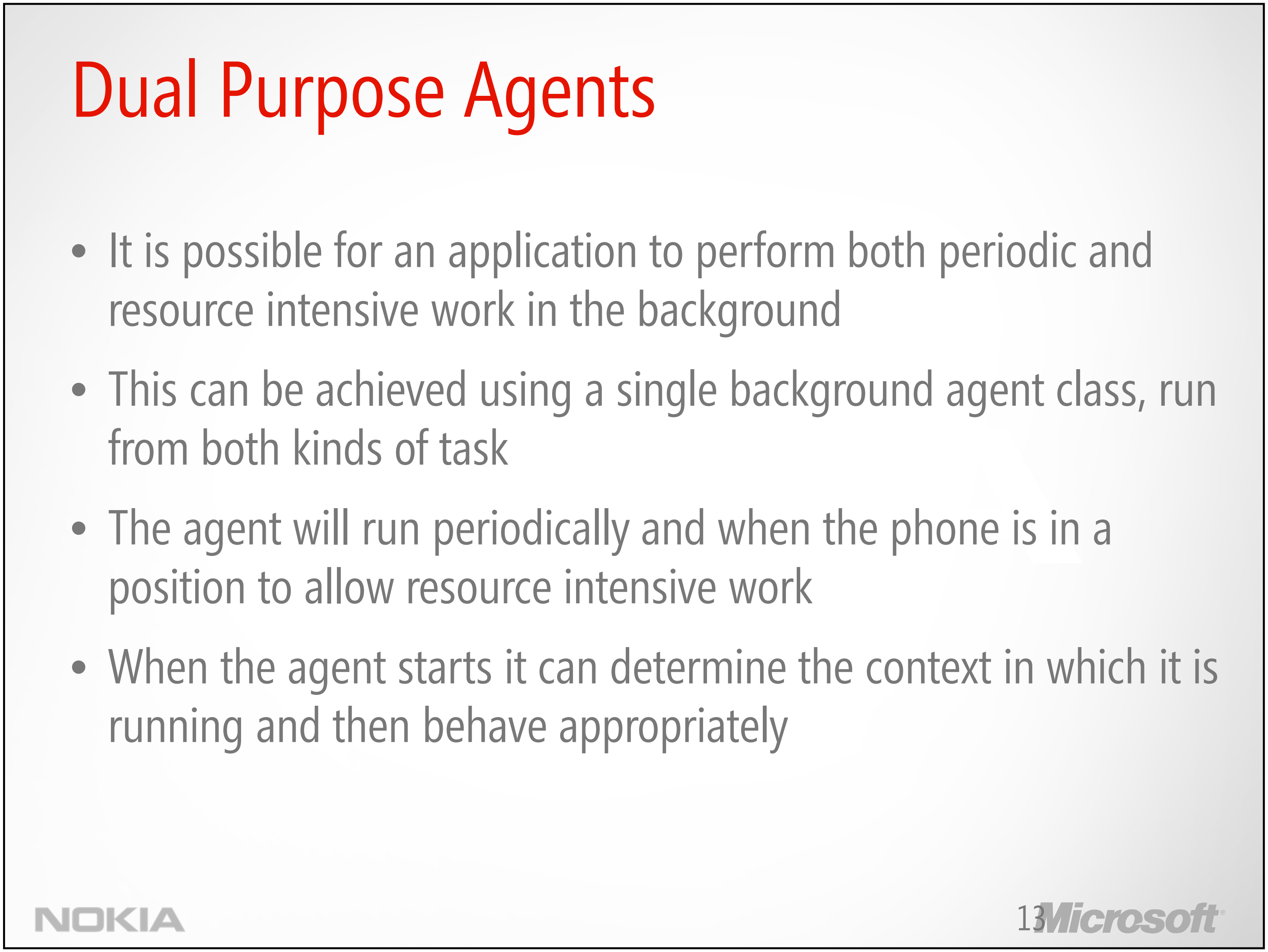
©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current 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.



ResourceIntensive Agents

- Resource Intensive Agents run when the phone is in a position where it can usefully perform some data processing:
  - When the phone is powered by the mains
  - When the phone is connected to WiFi
  - When the phone is not being used (Lock screen displayed)
- A “resource intensive” agent can run for up to 10 minutes
- Good for synchronisation with a host service, unpacking/preparing resources, compressing databases

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.



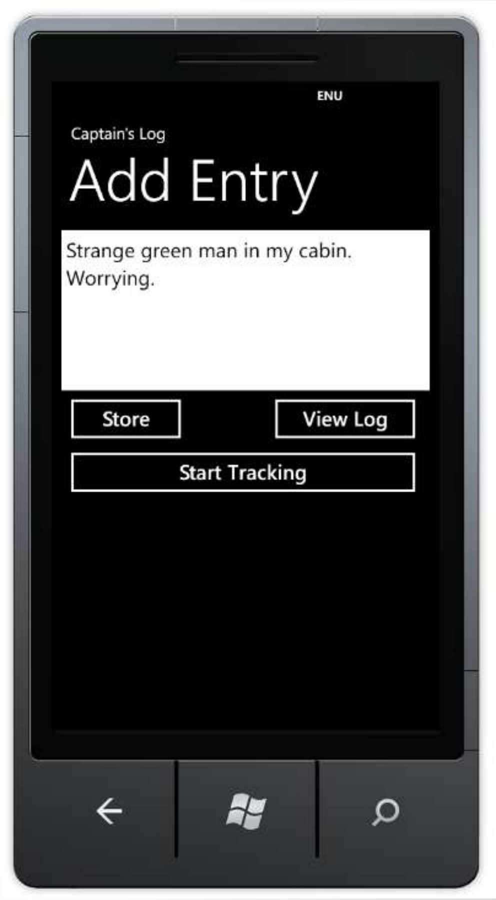
Dual Purpose Agents

- It is possible for an application to perform both periodic and resource intensive work in the background
- This can be achieved using a single background agent class, run from both kinds of task
- The agent will run periodically and when the phone is in a position to allow resource intensive work
- When the agent starts it can determine the context in which it is running and then behave appropriately

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.

Captain’s Log Location Tracker

- The “Captains Log” program is a simple logging application
- Users can type in log entries which are timestamped and stored in isolated storage
- We are going to add a location tracking feature using a background agent to regularly store the location of the phone in the log
- The agent will update the position even when the log program is not active

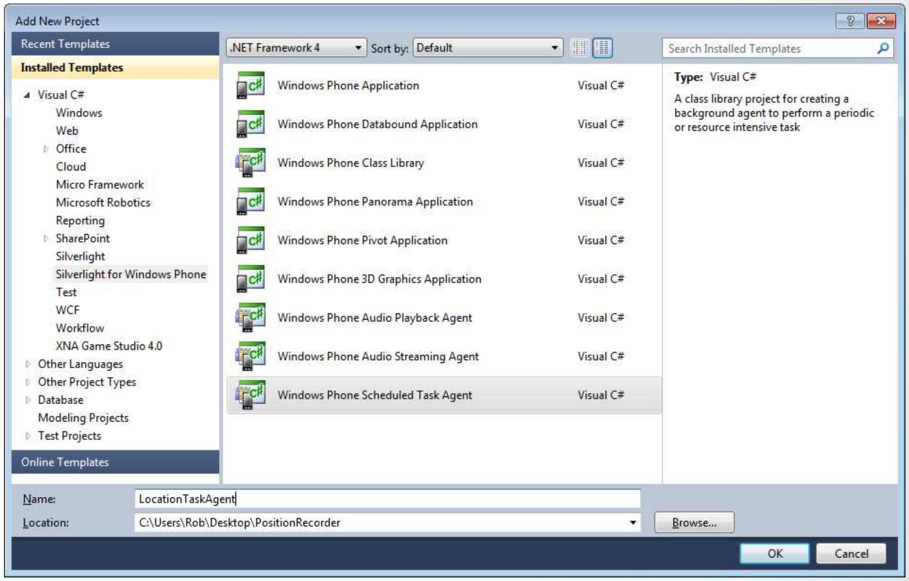


Strange green man in my cabin. Worrying.

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.

Creating a Background Agent

- A Background Agent is added to the application solution as a "Scheduled Task"
- There is a Visual Studio template just for this
- This agent will contain the code that runs when the agent is active



Windows Phone Application	Visual C#
Windows Phone Databound Application	Visual C#
Windows Phone Class Library	Visual C#
Windows Phone Panorama Application	Visual C#
Windows Phone Pivot Application	Visual C#
Windows Phone 3D Graphics Application	Visual C#
Windows Phone Audio Playback Agent	Visual C#
Windows Phone Audio Streaming Agent	Visual C#
Windows Phone Scheduled Task Agent	Visual C#

Type: Visual C#

A class library project for creating a background agent to perform a periodic or resource intensive task

Name: LocationTaskAgent

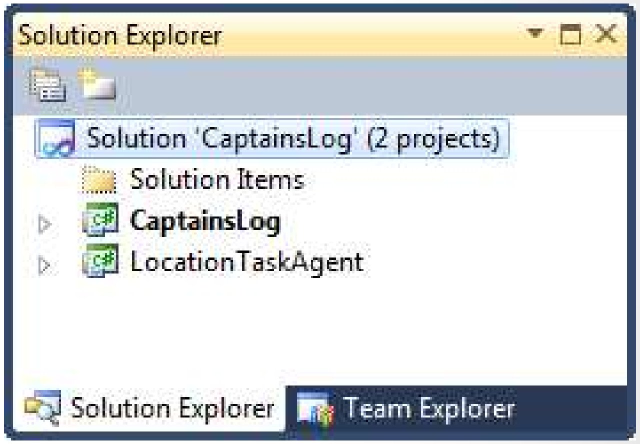
Location: C:\Users\Rob\Desktop\PositionRecorder

15

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.

The Captains Log Solution File

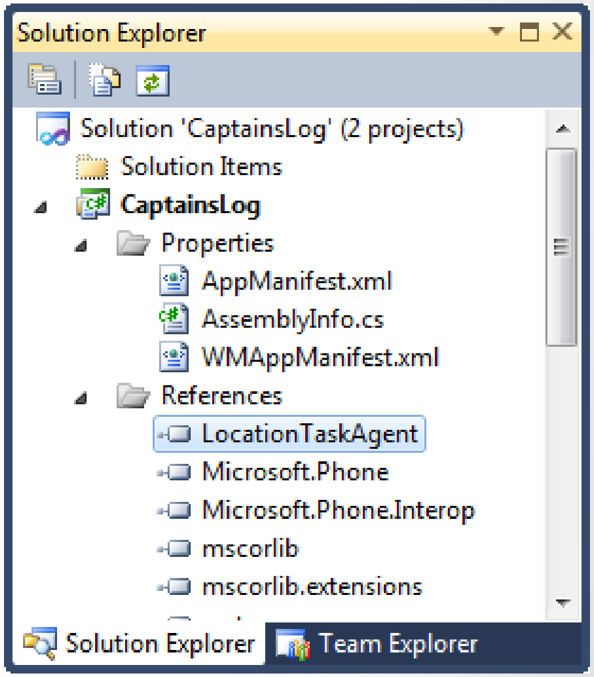
- The solution file contains two projects:
  - **CaptainsLog**: the Windows Phone Silverlight project which is the main application
  - **LocationTaskAgent**: the background agent to perform the tracking
- Solutions can contain many types of different projects
- When the solution is built all the assembly file outputs will be combined and sent to the phone



©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.

Connecting the Agent Project

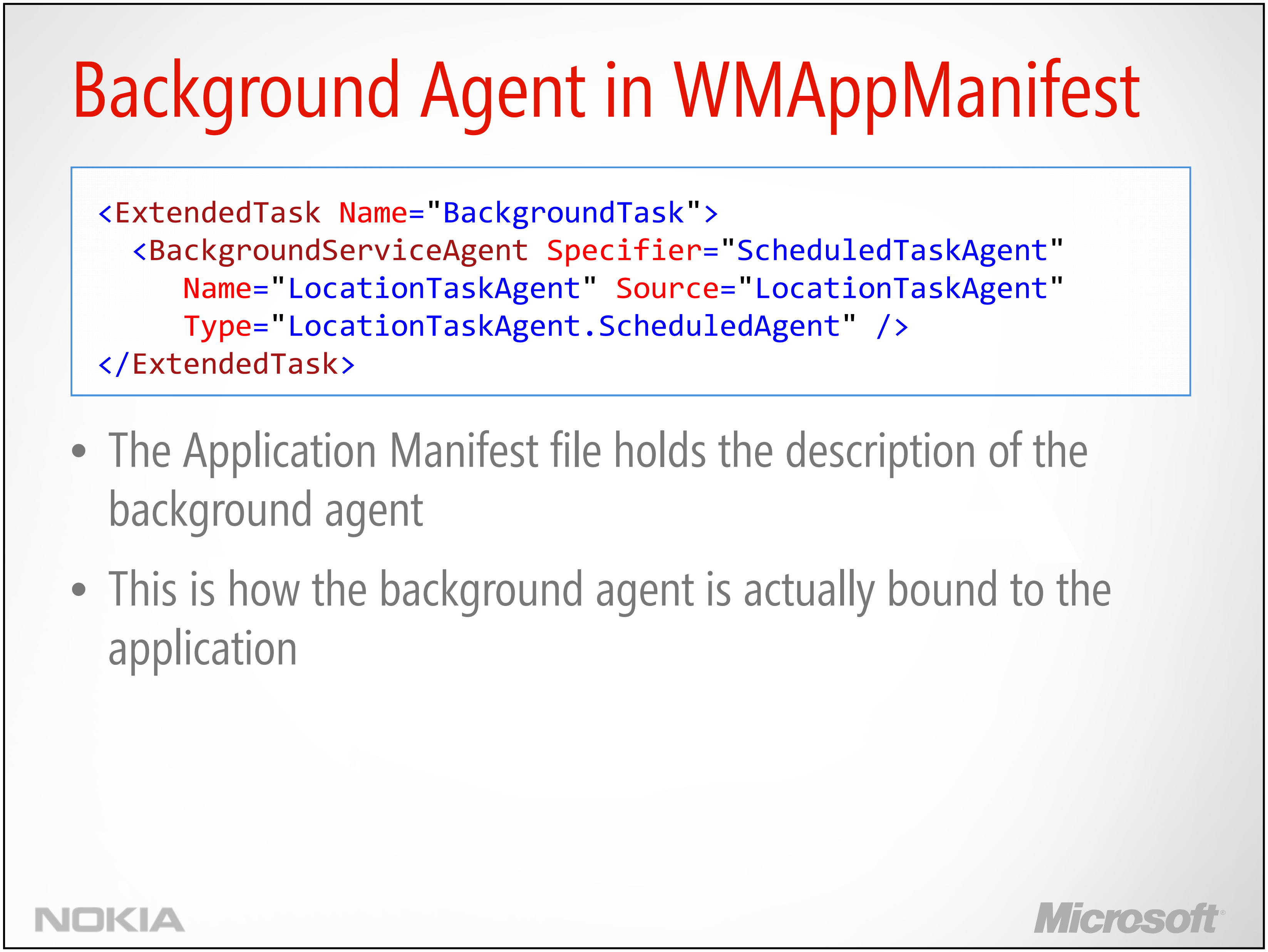
- The **CaptainsLog** project contains a reference to the output of the **LocationTaskAgent** project
- We have to explicitly link these two projects by adding a reference to the **LocationTaskAgent** output to the **CaptainsLog** project



Solution Explorer

```
Solution 'CaptainsLog' (2 projects)
  Solution Items
◢ CaptainsLog
  ◢ Properties
      AppManifest.xml
      AssemblyInfo.cs
      WMAppManifest.xml
  ◢ References
      LocationTaskAgent
      Microsoft.Phone
      Microsoft.Phone.Interop
      mscorlib
      mscorlib.extensions
```

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.

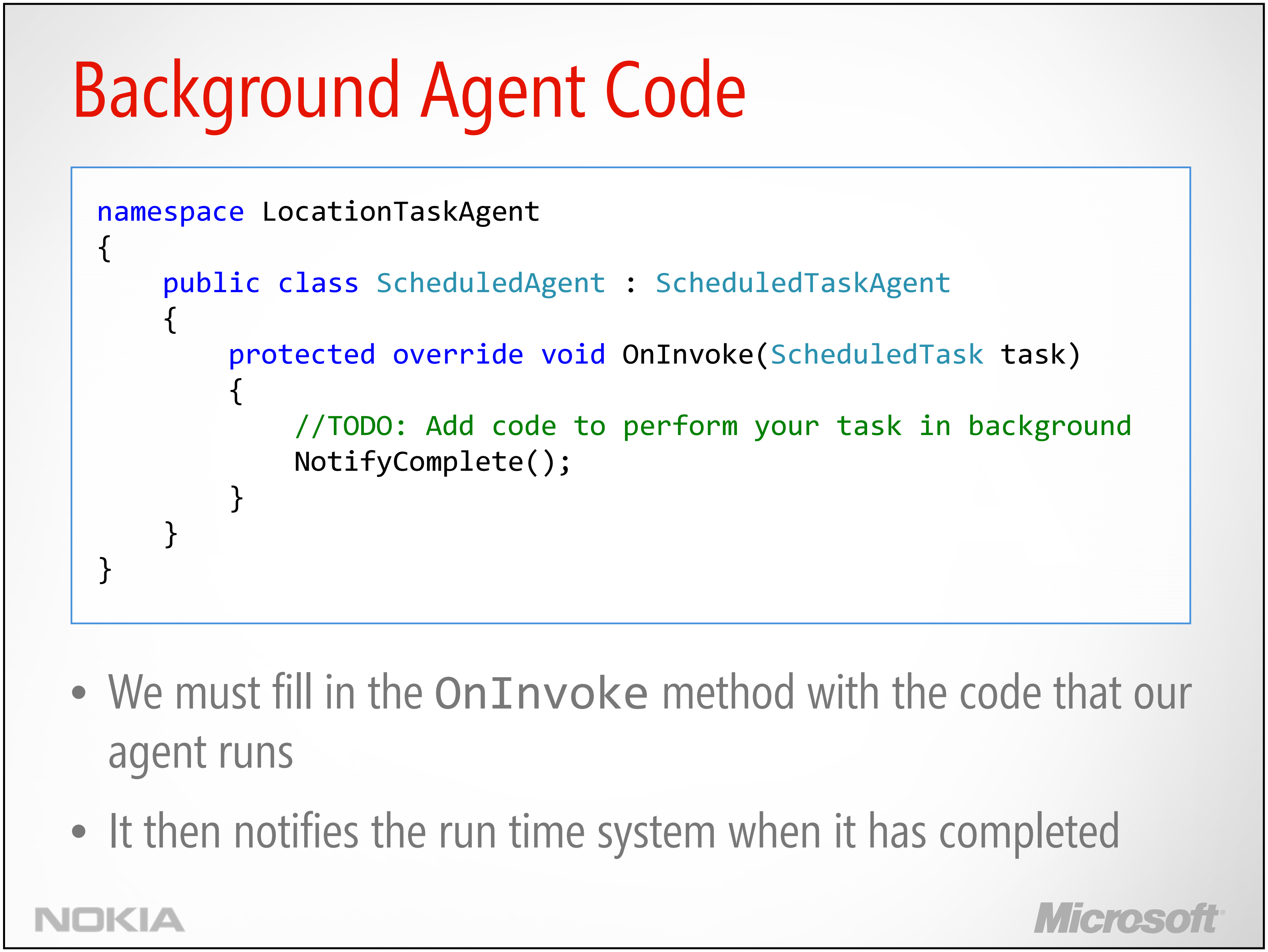


Background Agent in WMAppManifest

```xml
<ExtendedTask Name="BackgroundTask">
  <BackgroundServiceAgent Specifier="ScheduledTaskAgent"
     Name="LocationTaskAgent" Source="LocationTaskAgent"
     Type="LocationTaskAgent.ScheduledAgent" />
</ExtendedTask>
```

- The Application Manifest file holds the description of the background agent
- This is how the background agent is actually bound to the application

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.

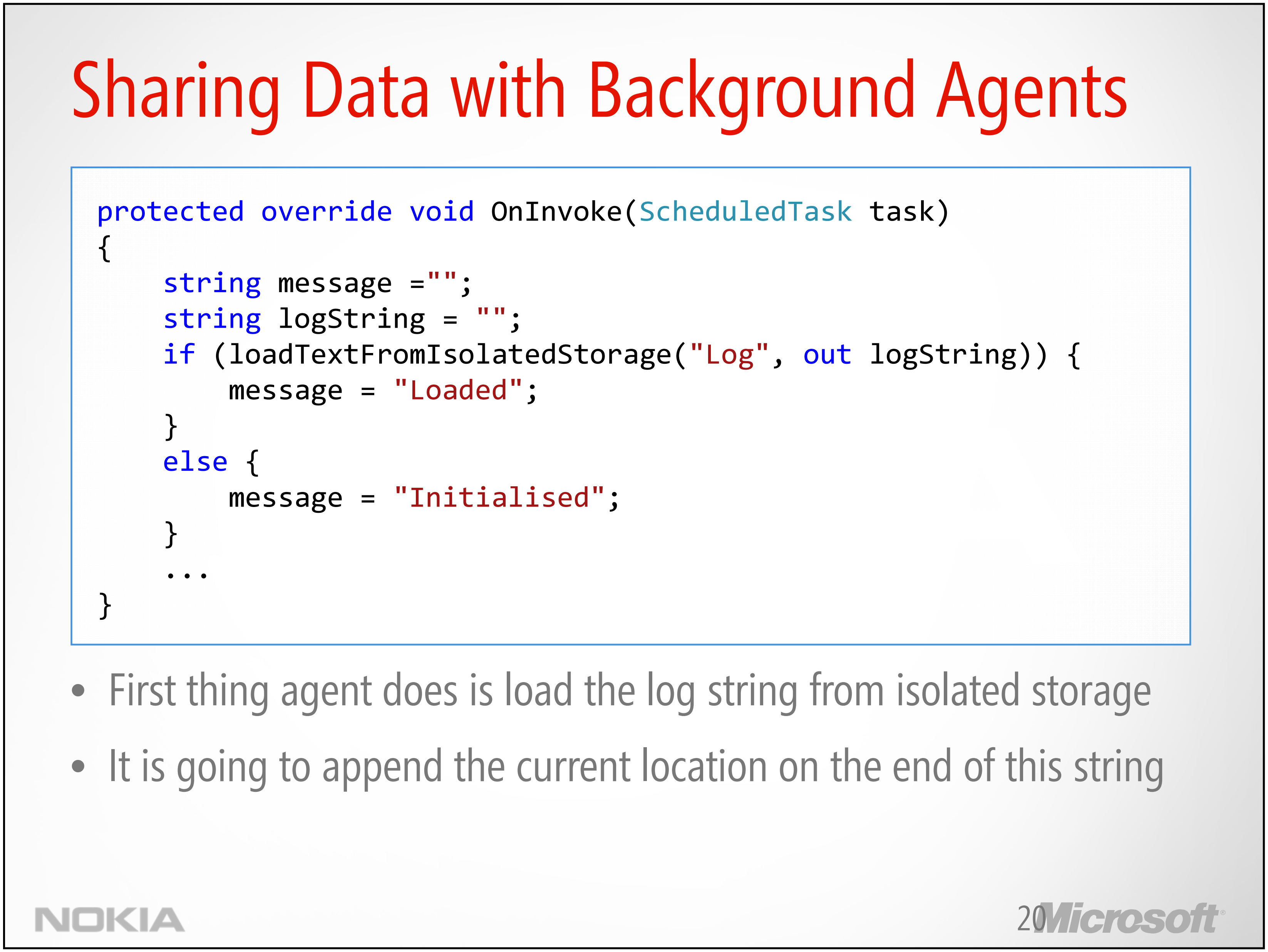


Background Agent Code

```
namespace LocationTaskAgent
{
    public class ScheduledAgent : ScheduledTaskAgent
    {
        protected override void OnInvoke(ScheduledTask task)
        {
            //TODO: Add code to perform your task in background
            NotifyComplete();
        }
    }
}
```

- We must fill in the `OnInvoke` method with the code that our agent runs
- It then notifies the run time system when it has completed

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.

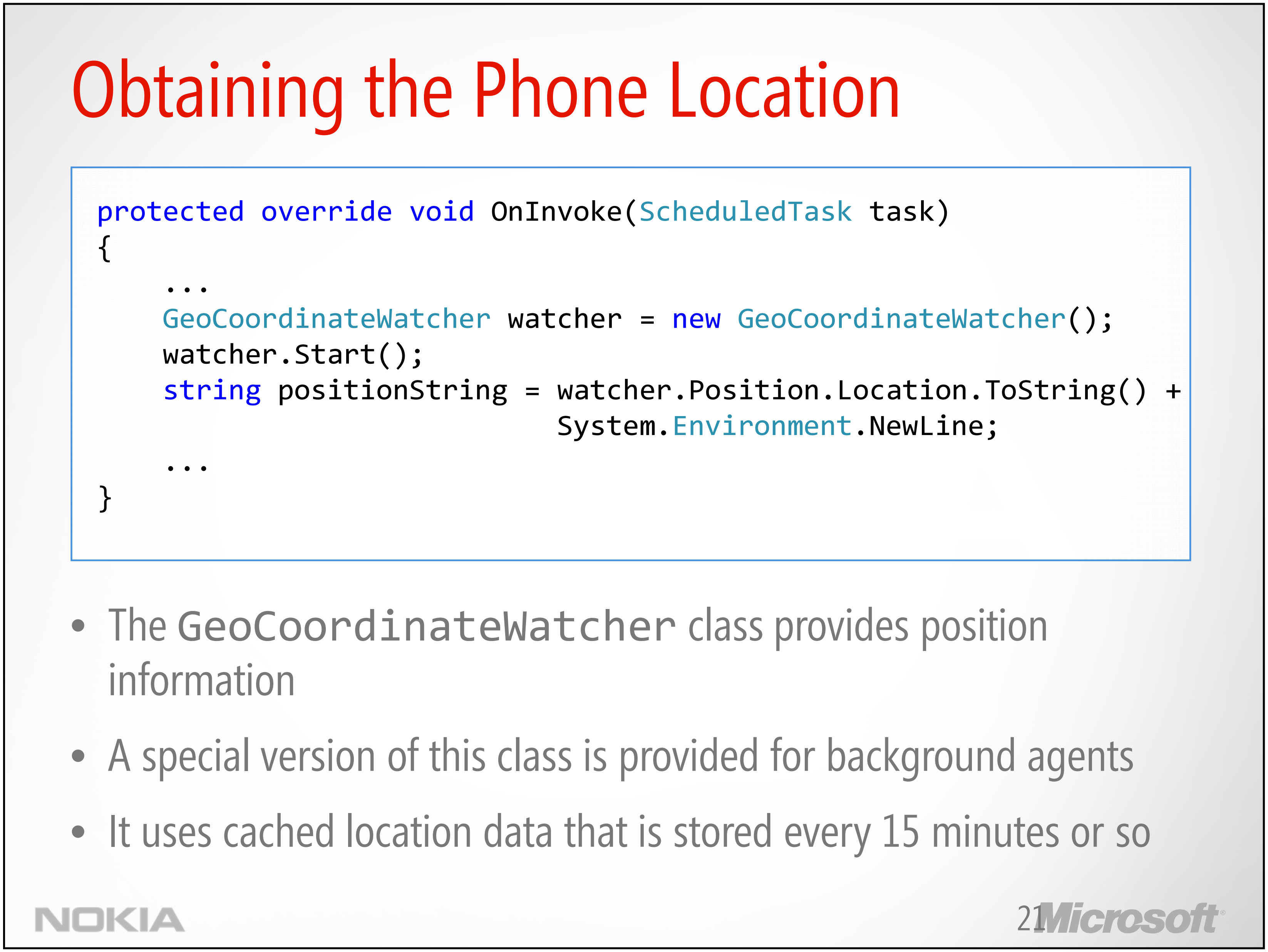


Sharing Data with Background Agents

```
protected override void OnInvoke(ScheduledTask task)
{
    string message ="";
    string logString = "";
    if (loadTextFromIsolatedStorage("Log", out logString)) {
        message = "Loaded";
    }
    else {
        message = "Initialised";
    }
    ...
}
```

- First thing agent does is load the log string from isolated storage
- It is going to append the current location on the end of this string

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.

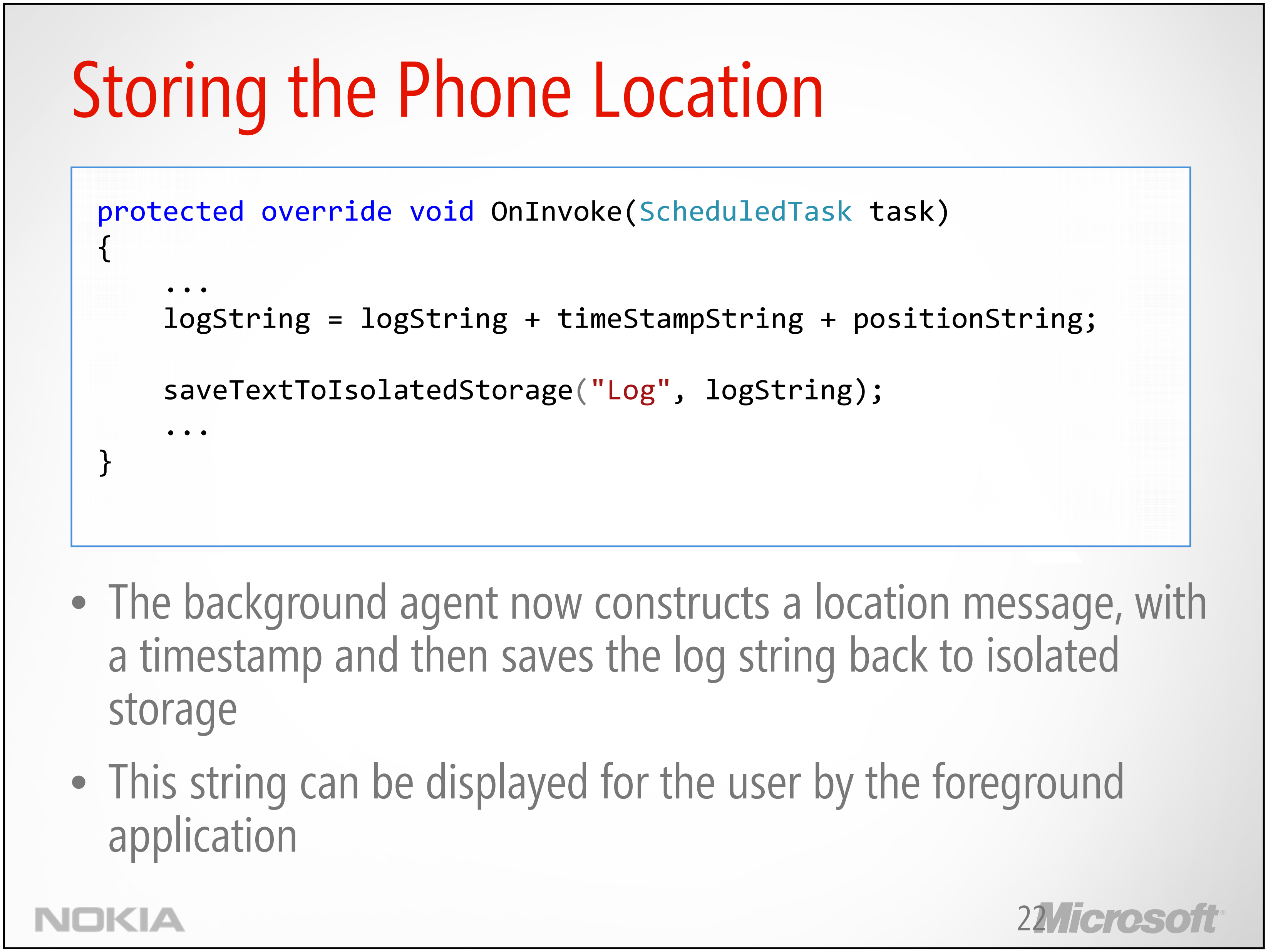


Obtaining the Phone Location

```
protected override void OnInvoke(ScheduledTask task)
{
    ...
    GeoCoordinateWatcher watcher = new GeoCoordinateWatcher();
    watcher.Start();
    string positionString = watcher.Position.Location.ToString() +
                            System.Environment.NewLine;
    ...
}
```

- The `GeoCoordinateWatcher` class provides position information
- A special version of this class is provided for background agents
- It uses cached location data that is stored every 15 minutes or so

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.

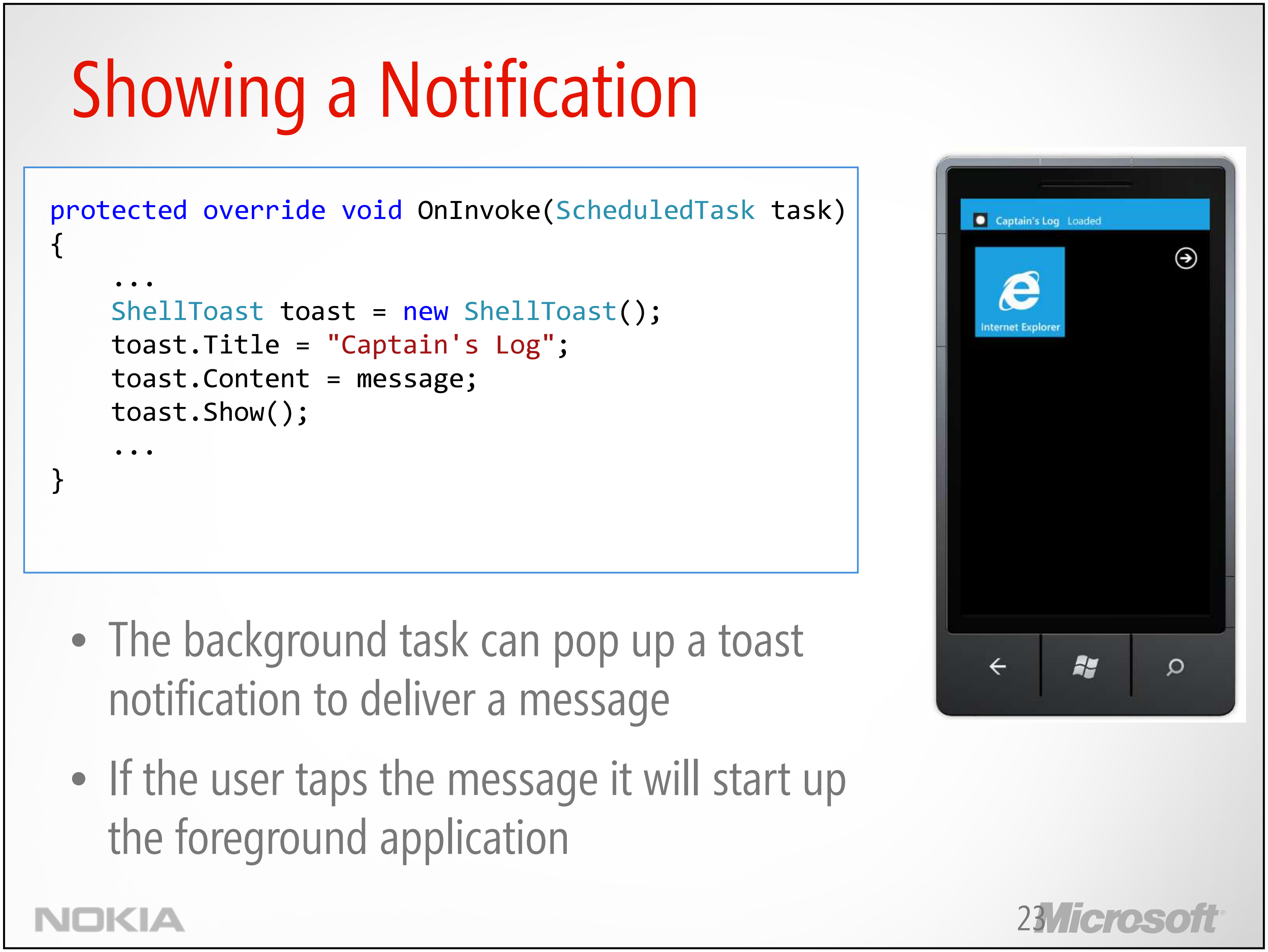


Storing the Phone Location

```
protected override void OnInvoke(ScheduledTask task)
{
    ...
    logString = logString + timeStampString + positionString;

    saveTextToIsolatedStorage("Log", logString);
    ...
}
```

- The background agent now constructs a location message, with a timestamp and then saves the log string back to isolated storage
- This string can be displayed for the user by the foreground application

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.



Showing a Notification

```
protected override void OnInvoke(ScheduledTask task)
{
    ...
    ShellToast toast = new ShellToast();
    toast.Title = "Captain's Log";
    toast.Content = message;
    toast.Show();
    ...
}
```

- The background task can pop up a toast notification to deliver a message
- If the user taps the message it will start up the foreground application

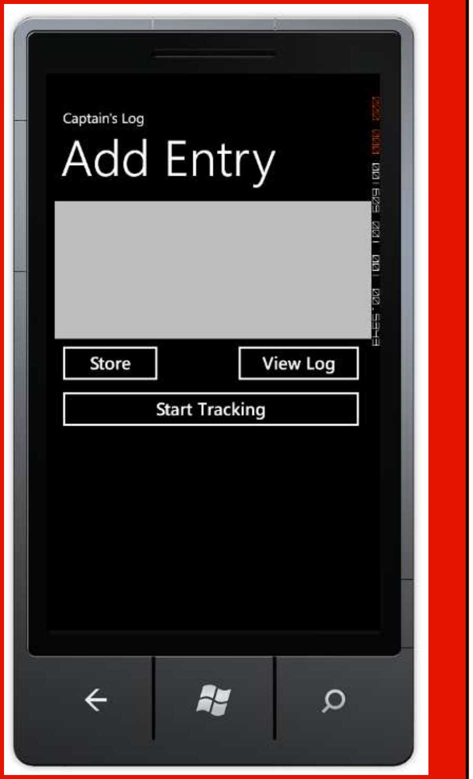
23

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.

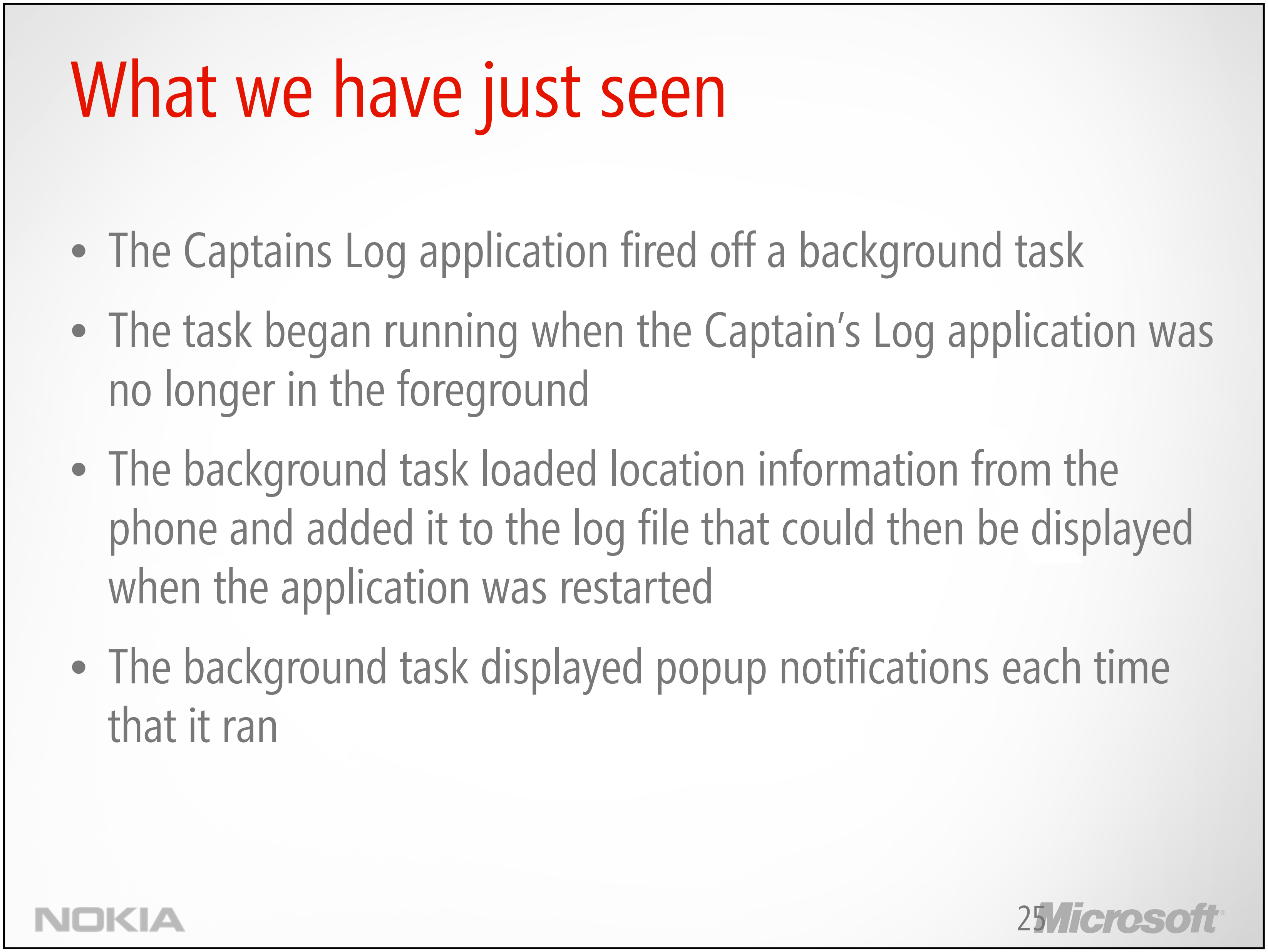
Windows Phone

Demo

Demo1: Location Logging



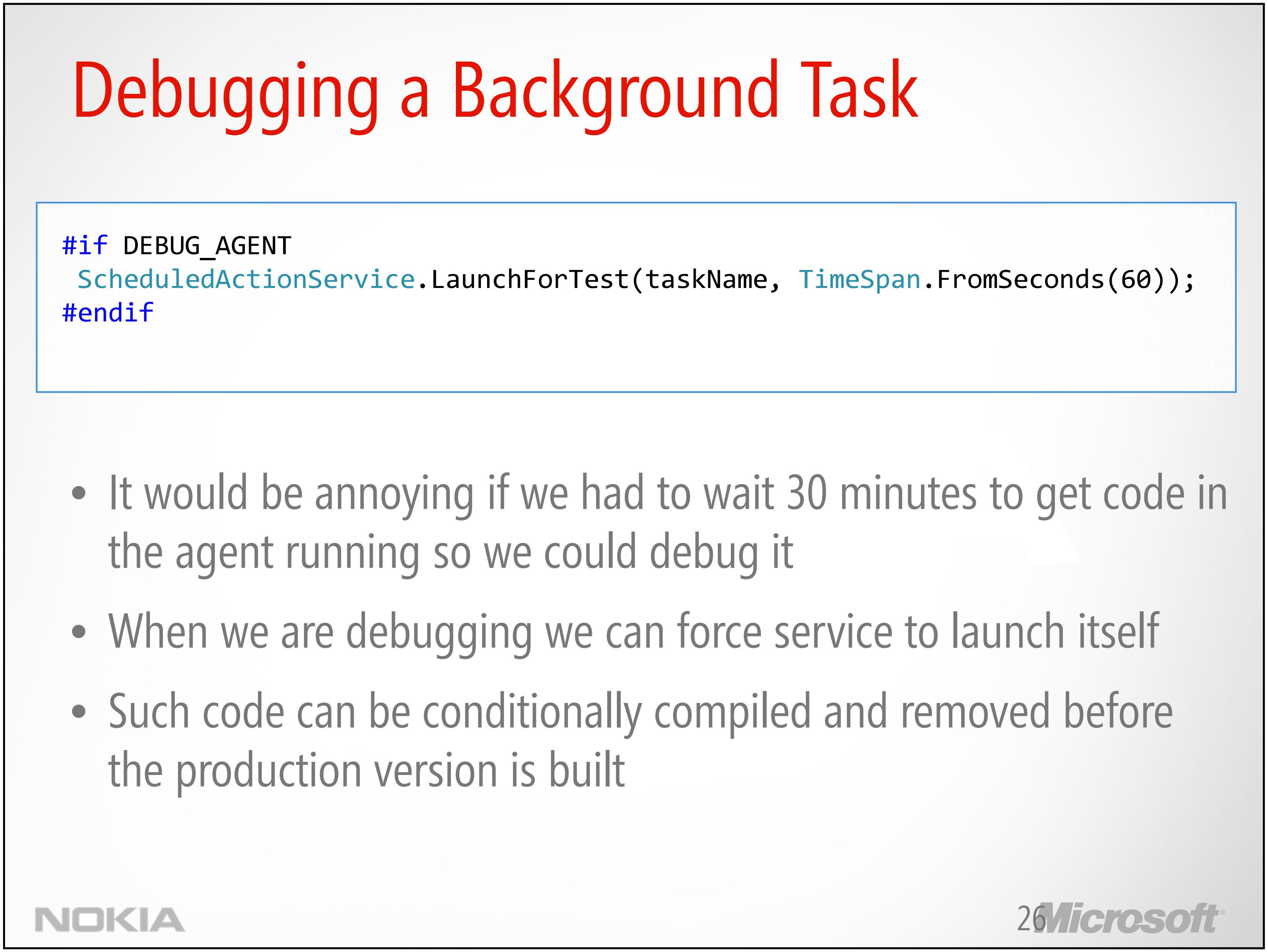
©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current 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.



What we have just seen

- The Captains Log application fired off a background task
- The task began running when the Captain's Log application was no longer in the foreground
- The background task loaded location information from the phone and added it to the log file that could then be displayed when the application was restarted
- The background task displayed popup notifications each time that it ran

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.

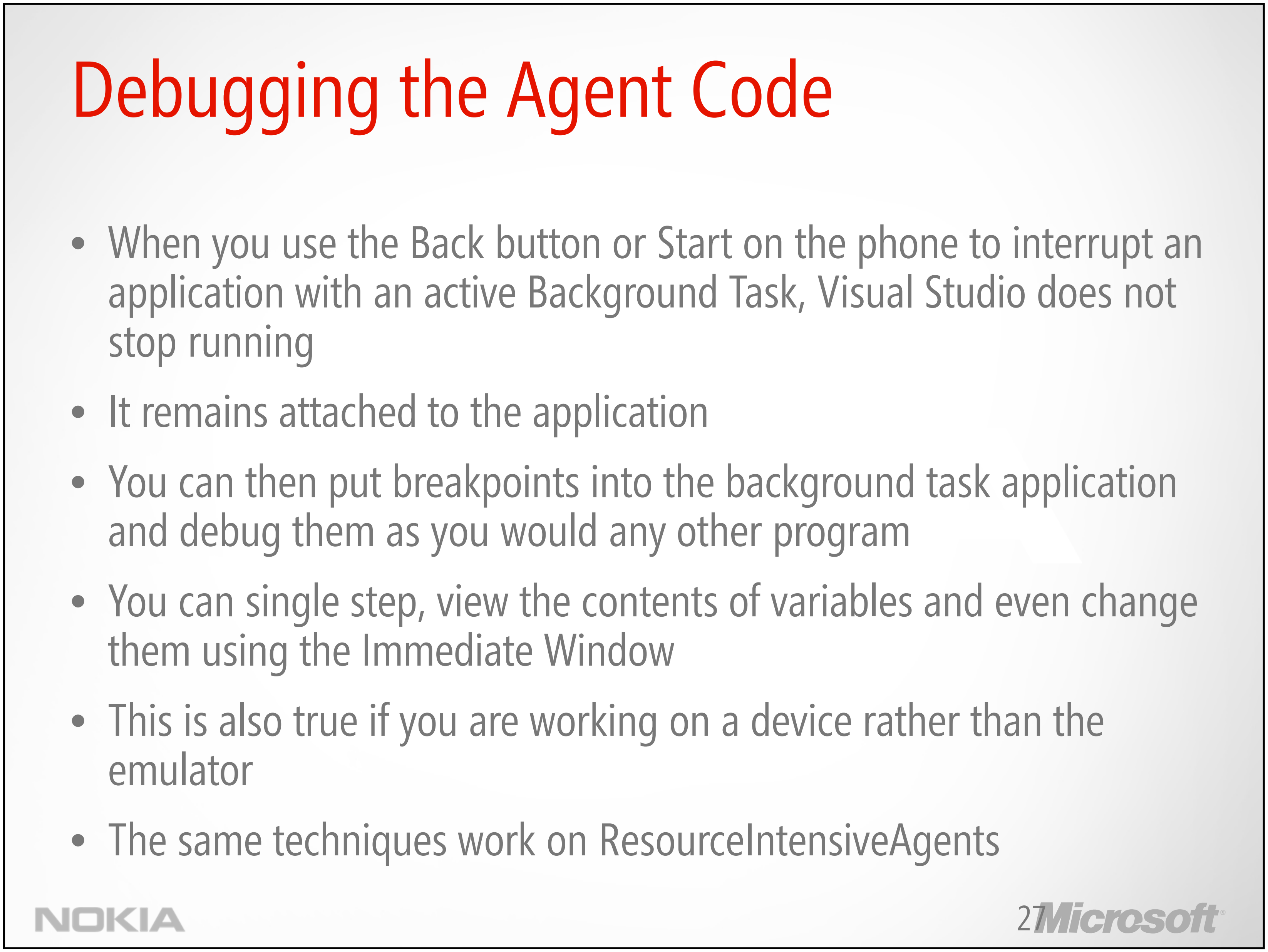


Debugging a Background Task

```
#if DEBUG_AGENT
 ScheduledActionService.LaunchForTest(taskName, TimeSpan.FromSeconds(60));
#endif
```

- It would be annoying if we had to wait 30 minutes to get code in the agent running so we could debug it
- When we are debugging we can force service to launch itself
- Such code can be conditionally compiled and removed before the production version is built

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.



Debugging the Agent Code

- When you use the Back button or Start on the phone to interrupt an application with an active Background Task, Visual Studio does not stop running
- It remains attached to the application
- You can then put breakpoints into the background task application and debug them as you would any other program
- You can single step, view the contents of variables and even change them using the Immediate Window
- This is also true if you are working on a device rather than the emulator
- The same techniques work on ResourceIntensiveAgents

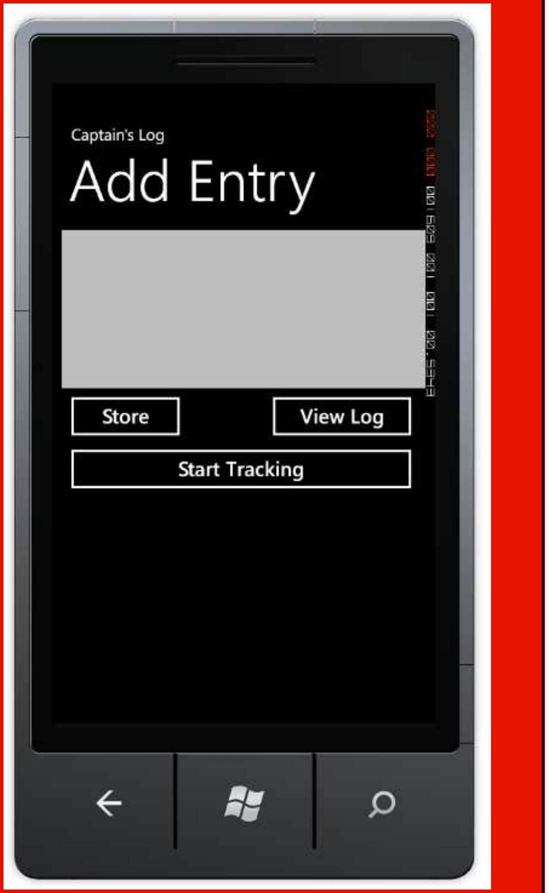
27

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.

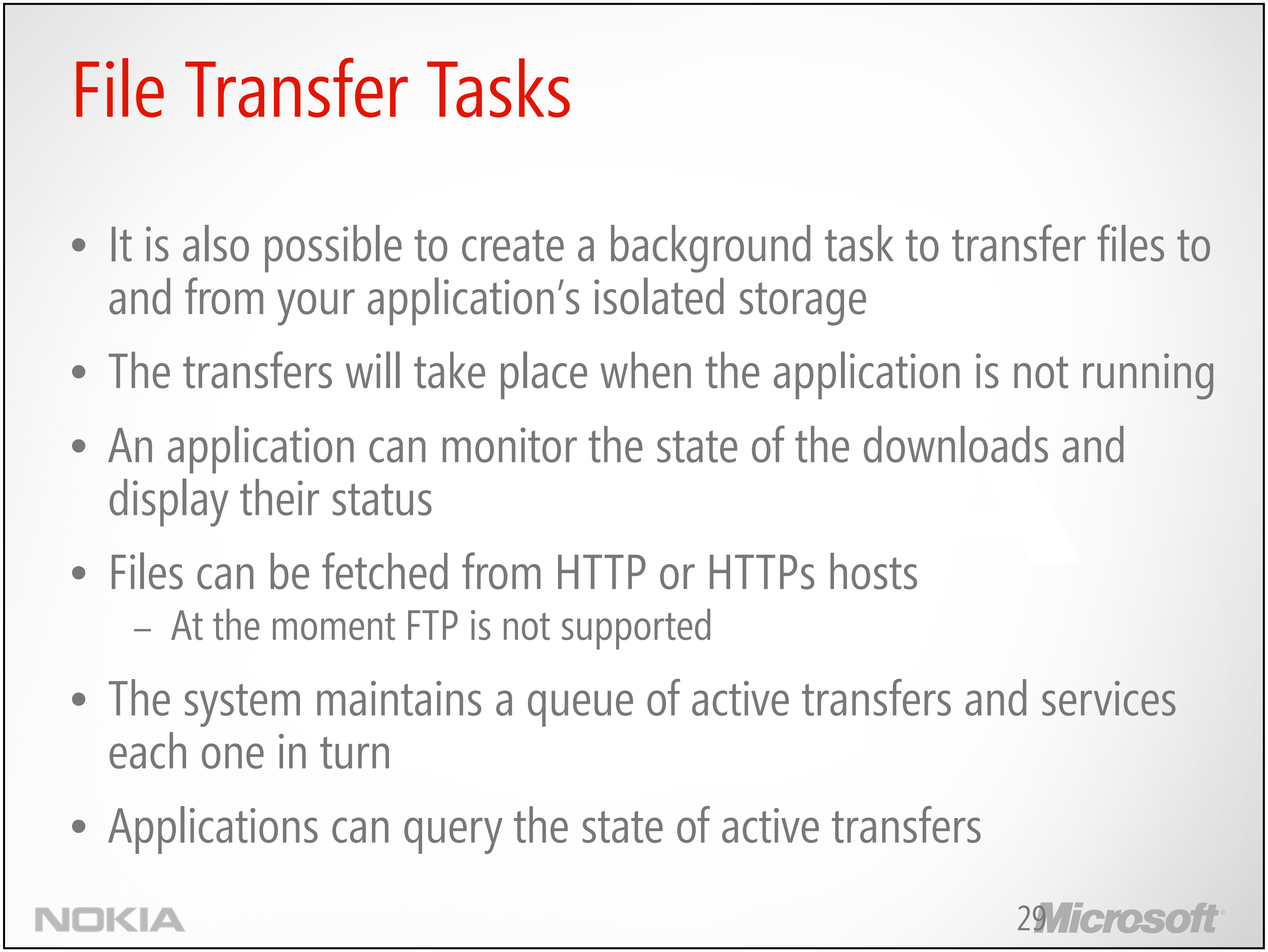
Windows Phone

Demo

Demo2: Debugging Tasks



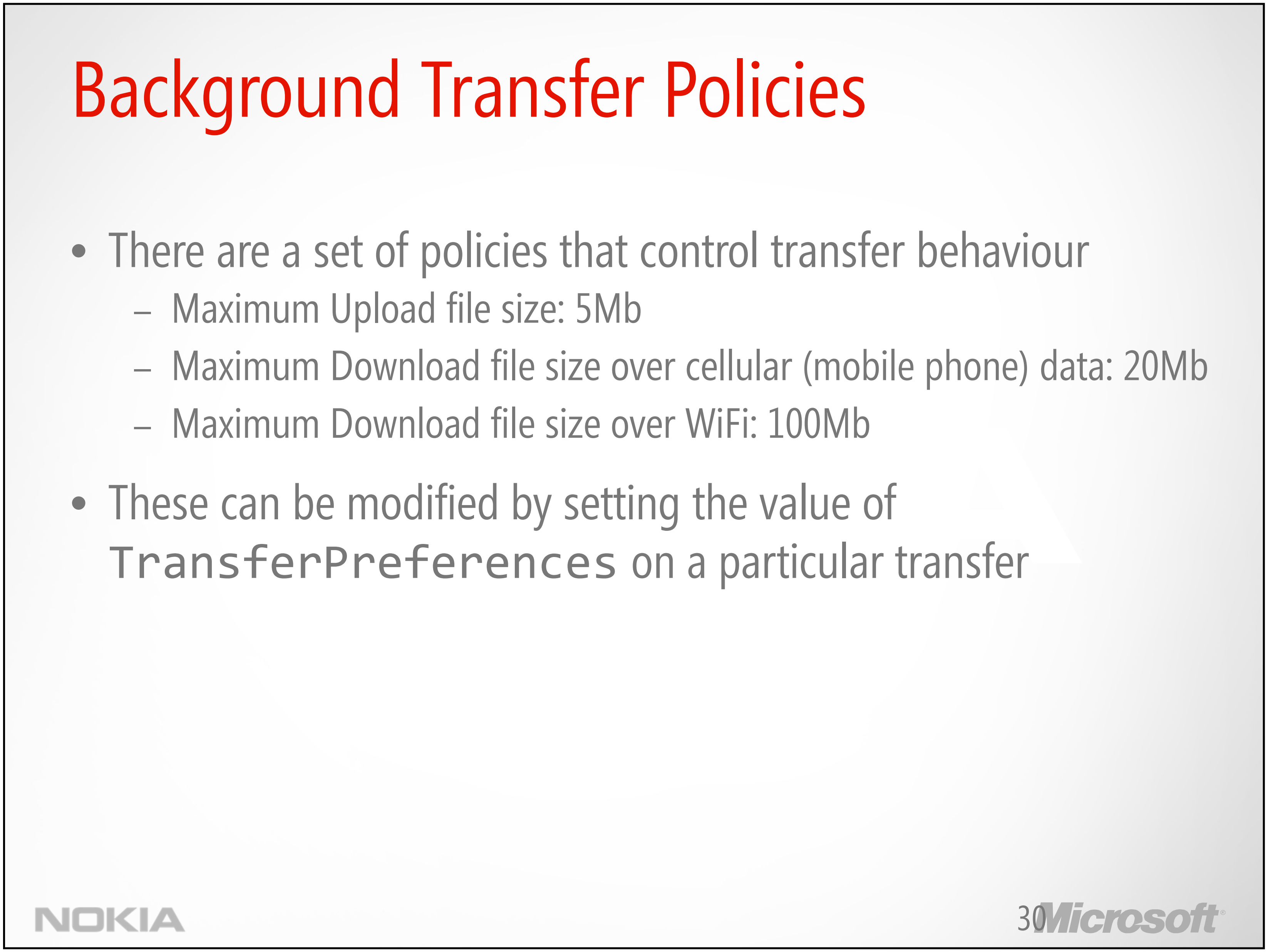
©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current 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.



File Transfer Tasks

- It is also possible to create a background task to transfer files to and from your application's isolated storage
- The transfers will take place when the application is not running
- An application can monitor the state of the downloads and display their status
- Files can be fetched from HTTP or HTTPs hosts
  - At the moment FTP is not supported
- The system maintains a queue of active transfers and services each one in turn
- Applications can query the state of active transfers

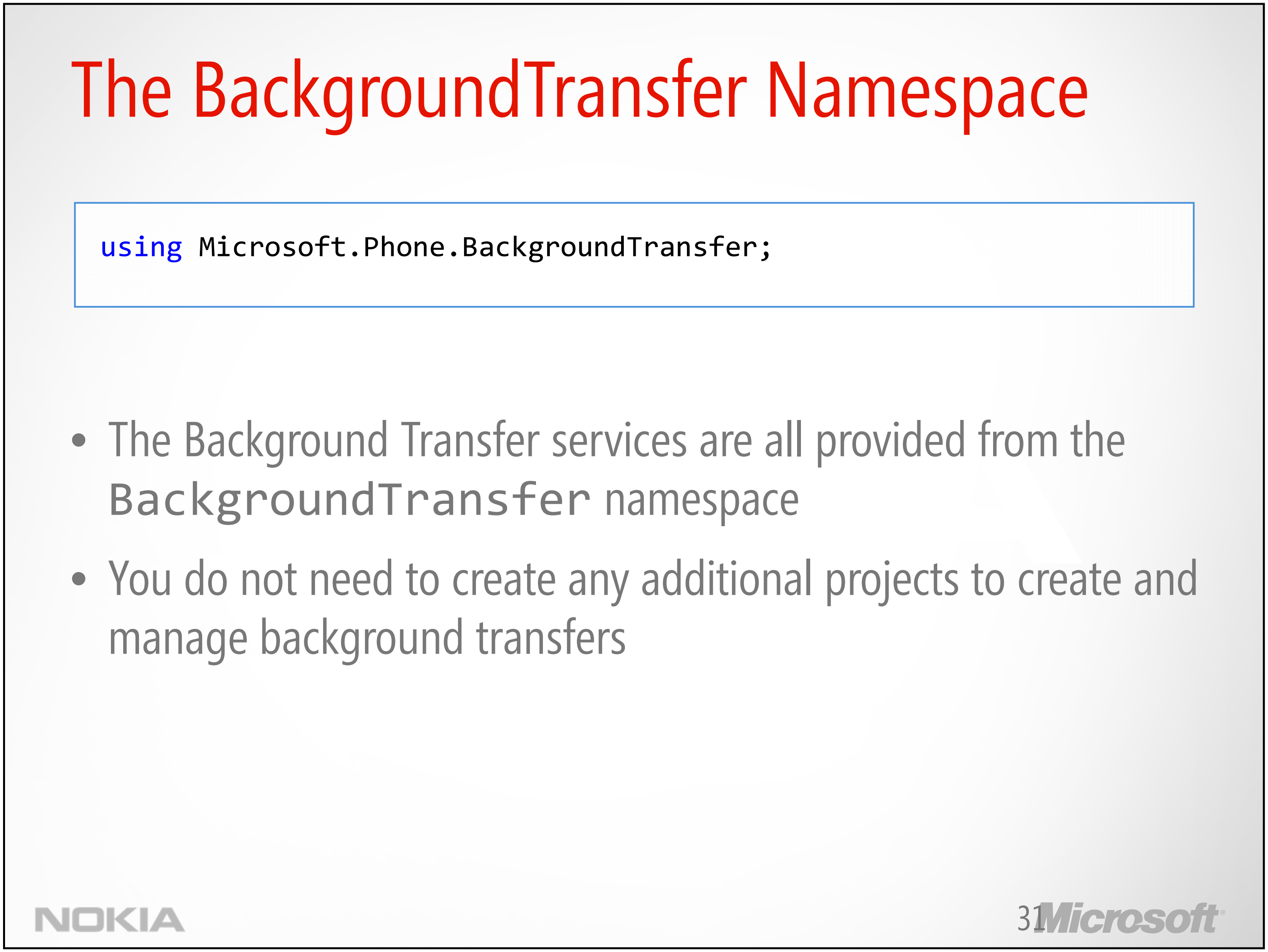
©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current 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.



Background Transfer Policies

- There are a set of policies that control transfer behaviour
  - Maximum Upload file size: 5Mb
  - Maximum Download file size over cellular (mobile phone) data: 20Mb
  - Maximum Download file size over WiFi: 100Mb
- These can be modified by setting the value of `TransferPreferences` on a particular transfer

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.



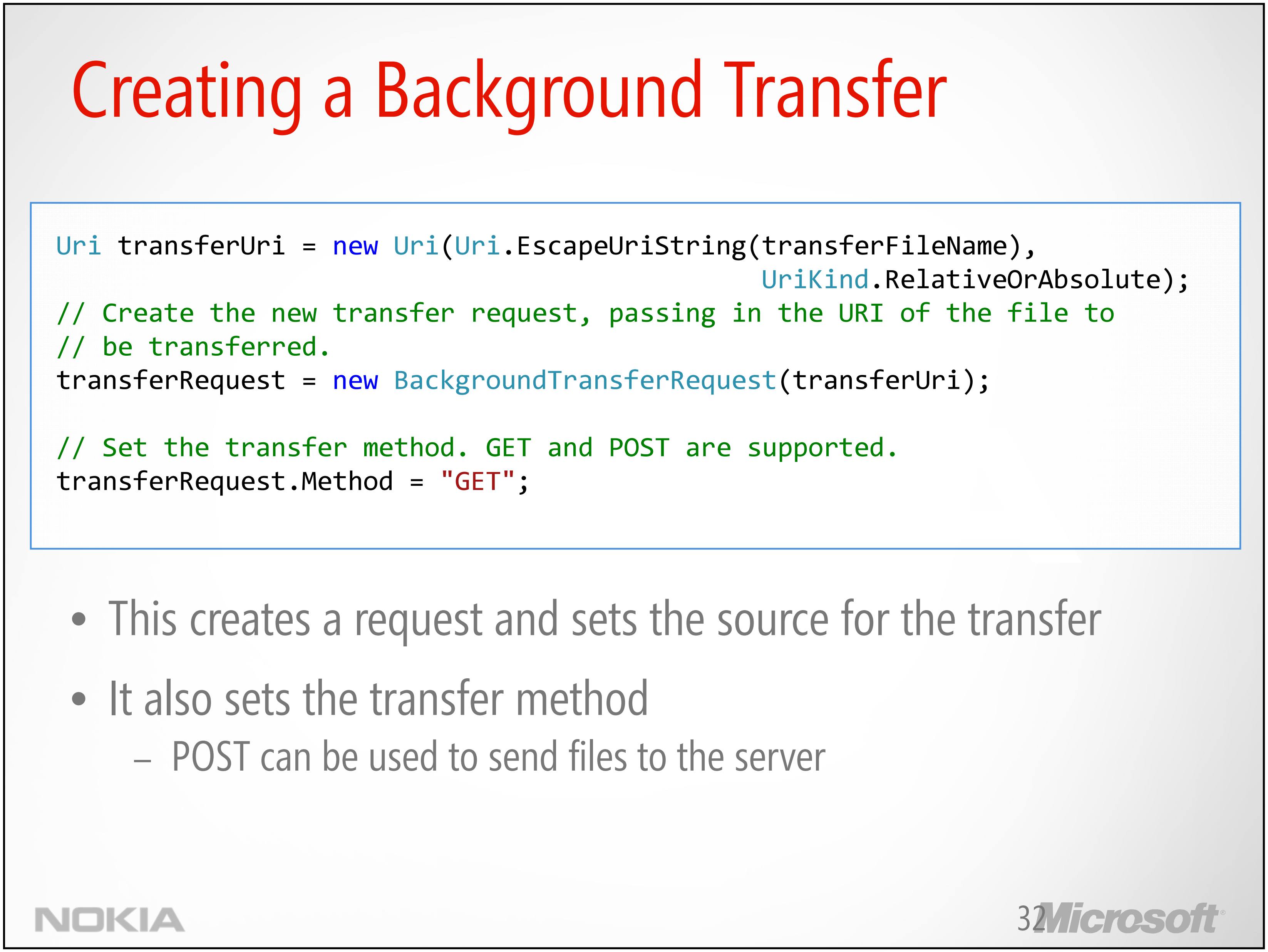
The BackgroundTransfer Namespace

```
using Microsoft.Phone.BackgroundTransfer;
```

- The Background Transfer services are all provided from the `BackgroundTransfer` namespace
- You do not need to create any additional projects to create and manage background transfers

31

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.

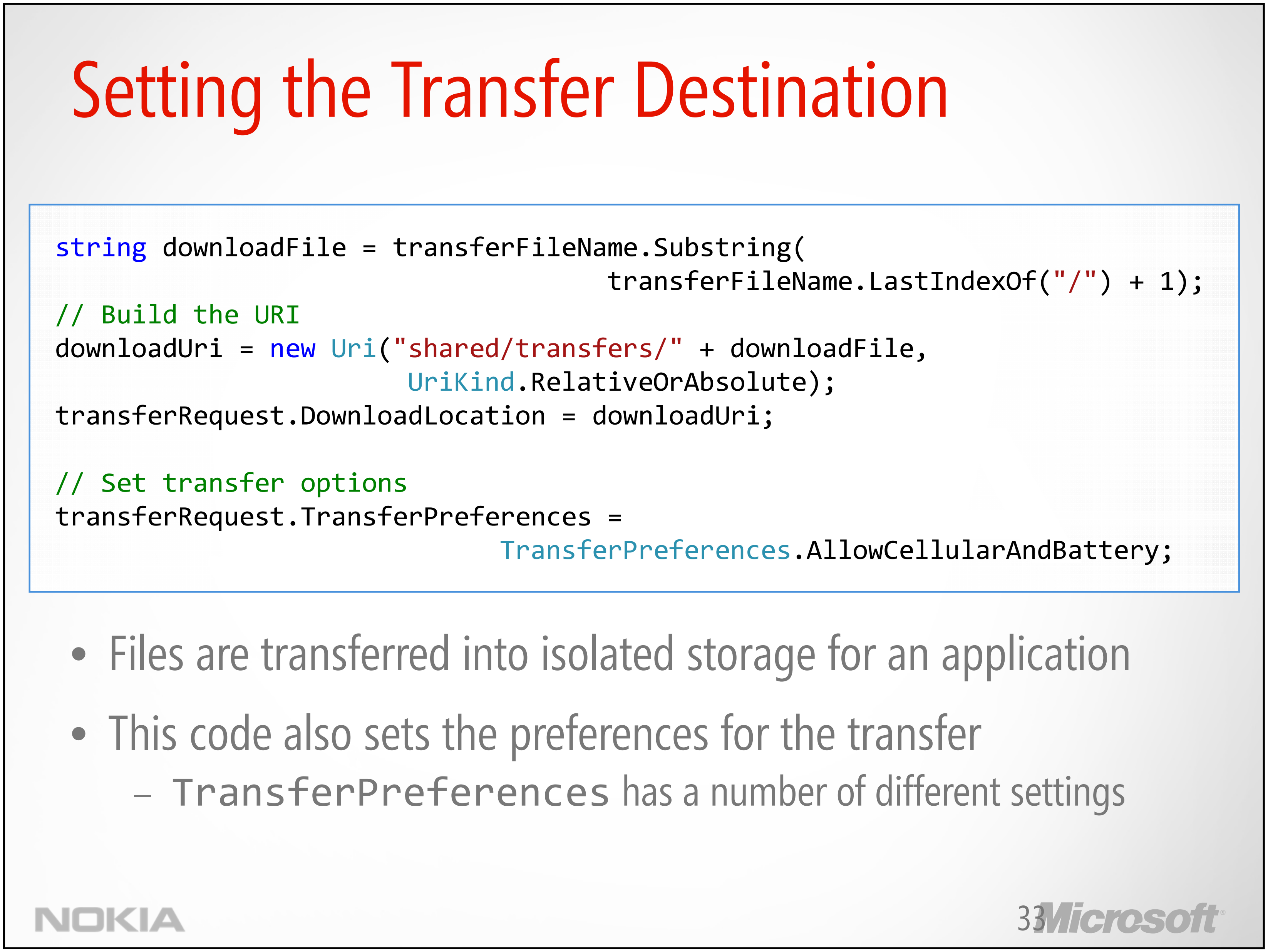


Creating a Background Transfer

```
Uri transferUri = new Uri(Uri.EscapeUriString(transferFileName),
                                              UriKind.RelativeOrAbsolute);
// Create the new transfer request, passing in the URI of the file to
// be transferred.
transferRequest = new BackgroundTransferRequest(transferUri);

// Set the transfer method. GET and POST are supported.
transferRequest.Method = "GET";
```

- This creates a request and sets the source for the transfer
- It also sets the transfer method
  - POST can be used to send files to the server

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.

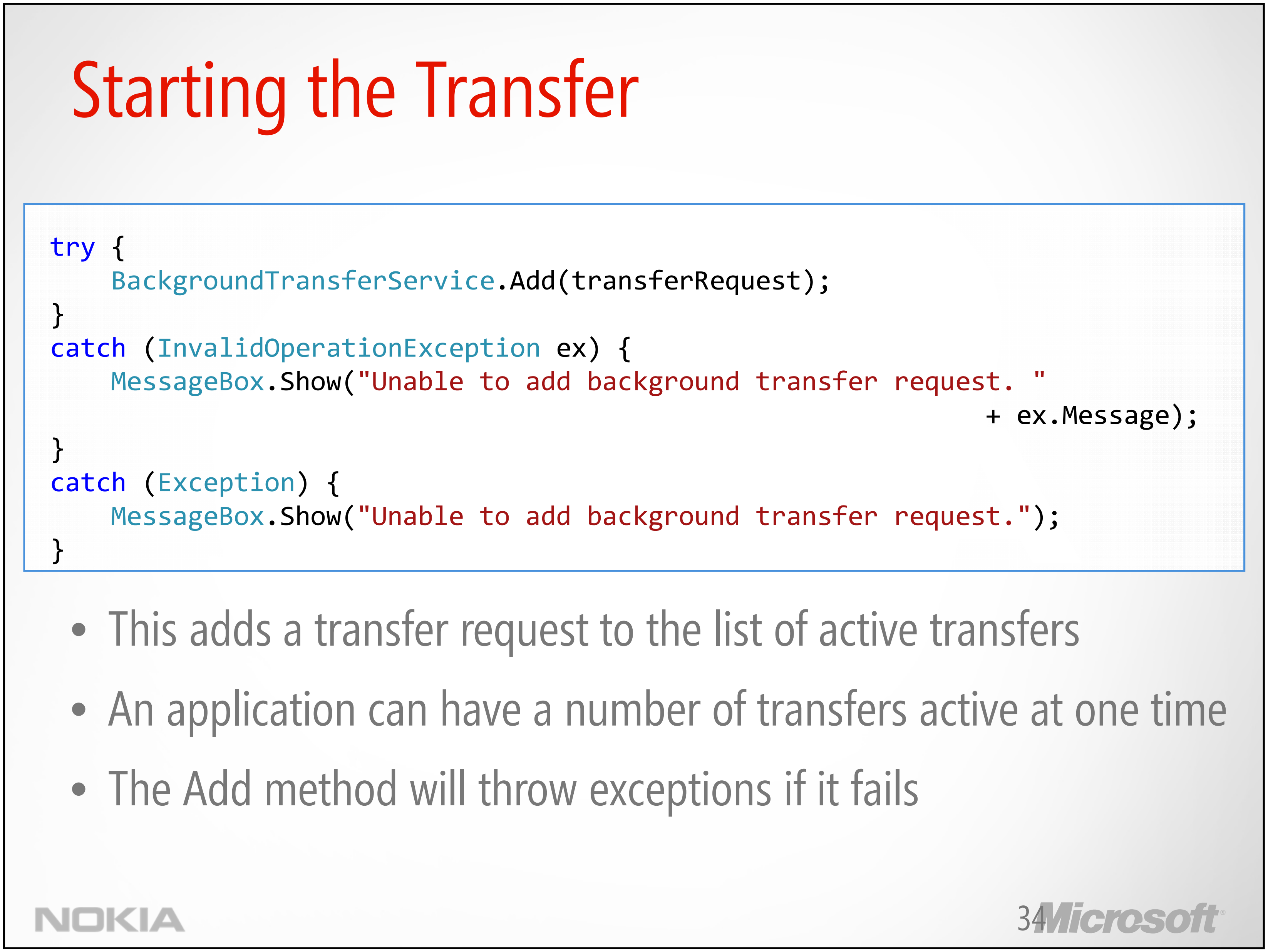


Setting the Transfer Destination

```
string downloadFile = transferFileName.Substring(
                                    transferFileName.LastIndexOf("/") + 1);
// Build the URI
downloadUri = new Uri("shared/transfers/" + downloadFile,
                       UriKind.RelativeOrAbsolute);
transferRequest.DownloadLocation = downloadUri;

// Set transfer options
transferRequest.TransferPreferences =
                             TransferPreferences.AllowCellularAndBattery;
```

- Files are transferred into isolated storage for an application
- This code also sets the preferences for the transfer
  - `TransferPreferences` has a number of different settings

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.

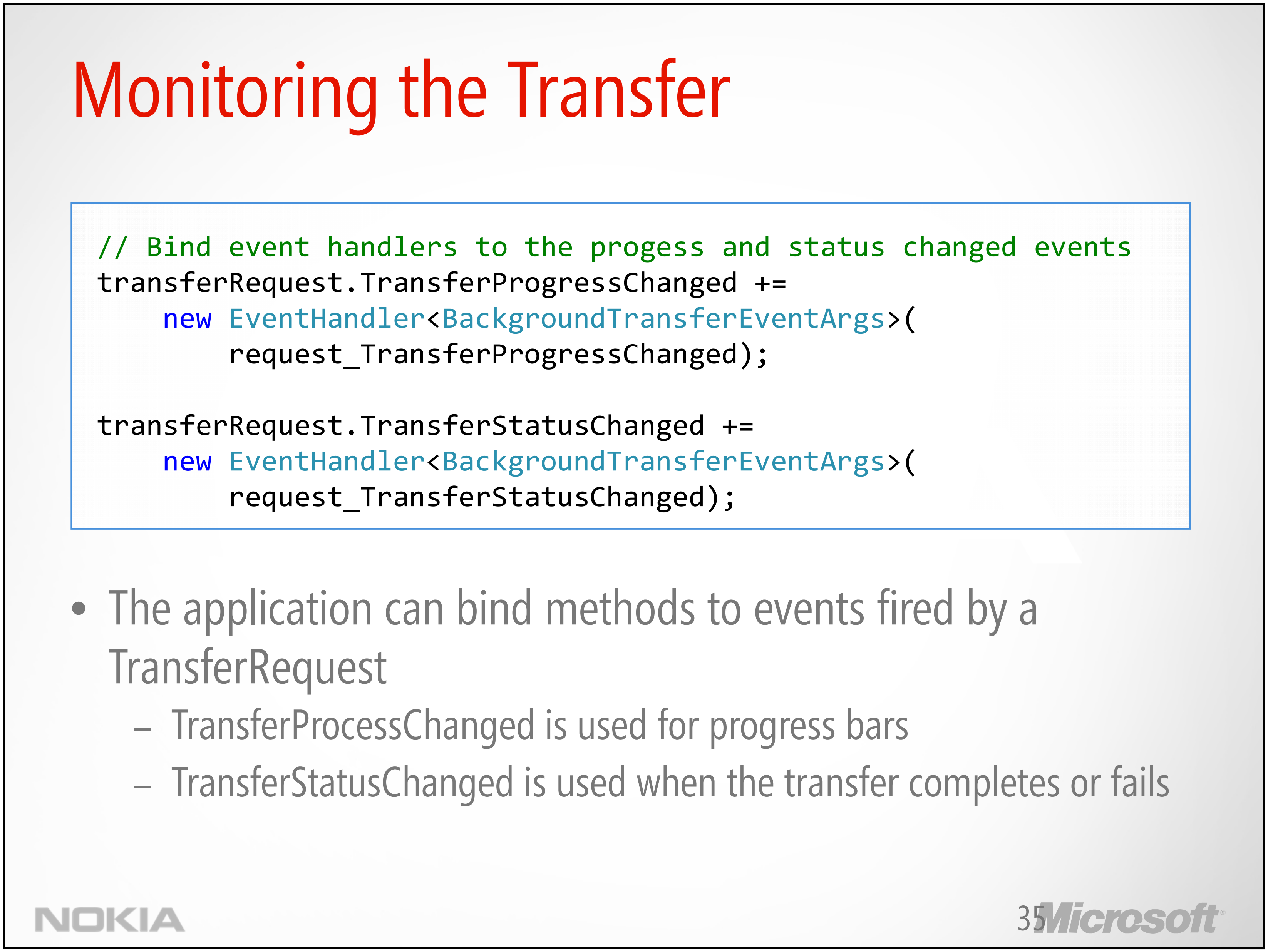


Starting the Transfer

```
try {
    BackgroundTransferService.Add(transferRequest);
}
catch (InvalidOperationException ex) {
    MessageBox.Show("Unable to add background transfer request. "
                                                            + ex.Message);
}
catch (Exception) {
    MessageBox.Show("Unable to add background transfer request.");
}
```

- This adds a transfer request to the list of active transfers
- An application can have a number of transfers active at one time
- The Add method will throw exceptions if it fails

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.



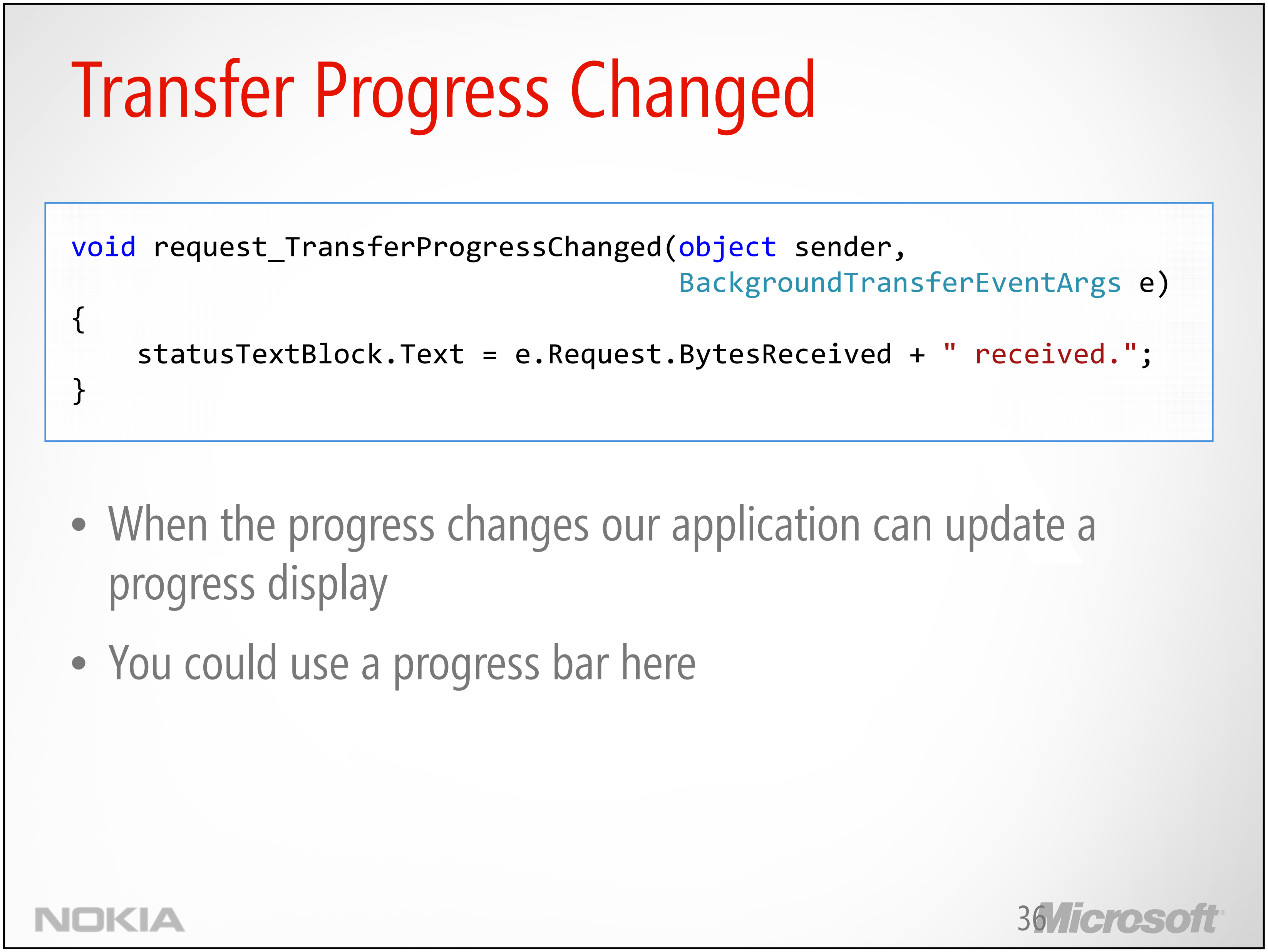
Monitoring the Transfer

```
// Bind event handlers to the progess and status changed events
transferRequest.TransferProgressChanged +=
    new EventHandler<BackgroundTransferEventArgs>(
        request_TransferProgressChanged);

transferRequest.TransferStatusChanged +=
    new EventHandler<BackgroundTransferEventArgs>(
        request_TransferStatusChanged);
```

- The application can bind methods to events fired by a TransferRequest
  - TransferProcessChanged is used for progress bars
  - TransferStatusChanged is used when the transfer completes or fails

35

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.



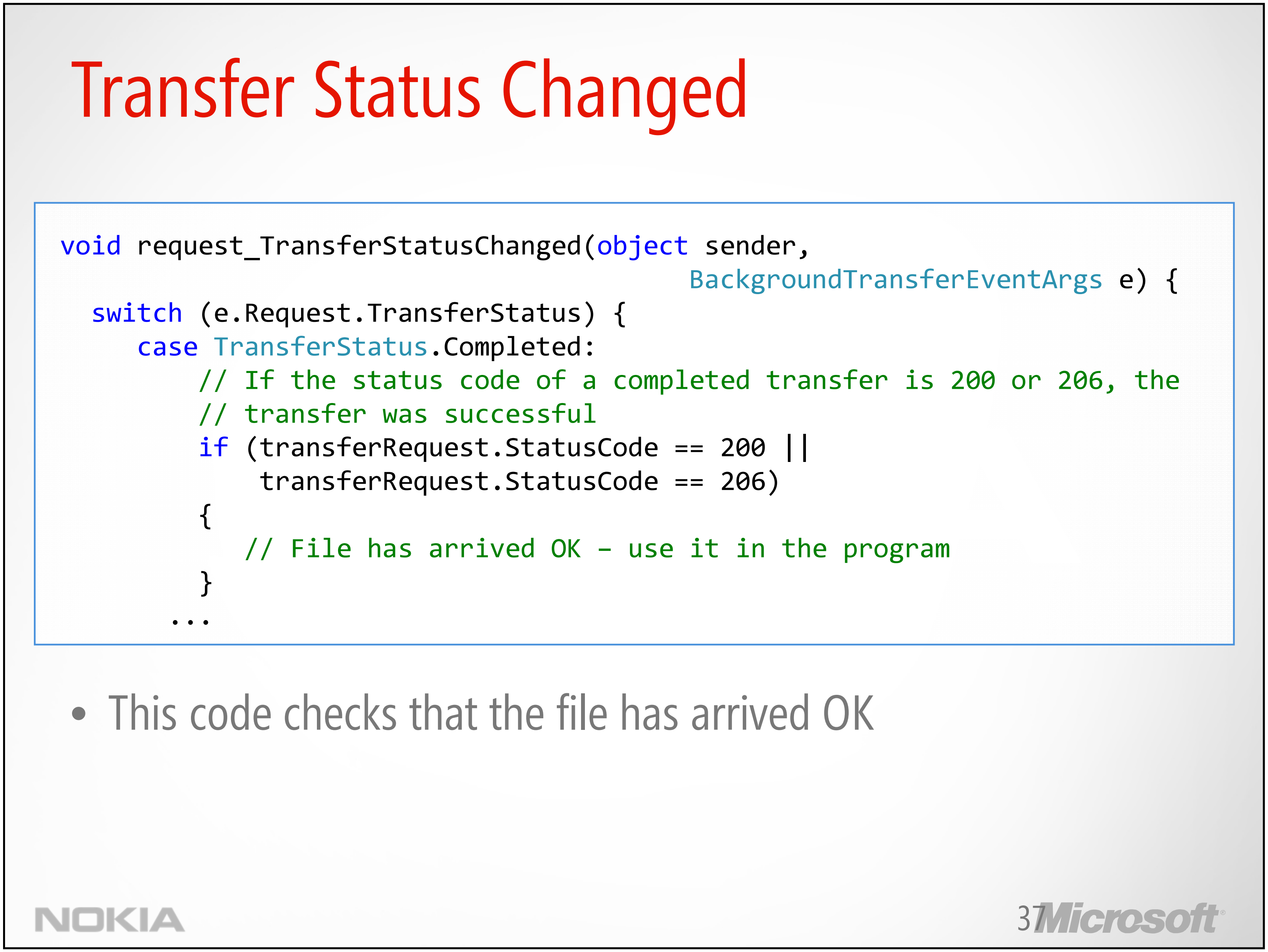
Transfer Progress Changed

```
void request_TransferProgressChanged(object sender,
                                     BackgroundTransferEventArgs e)
{
    statusTextBlock.Text = e.Request.BytesReceived + " received.";
}
```

- When the progress changes our application can update a progress display
- You could use a progress bar here

36

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.



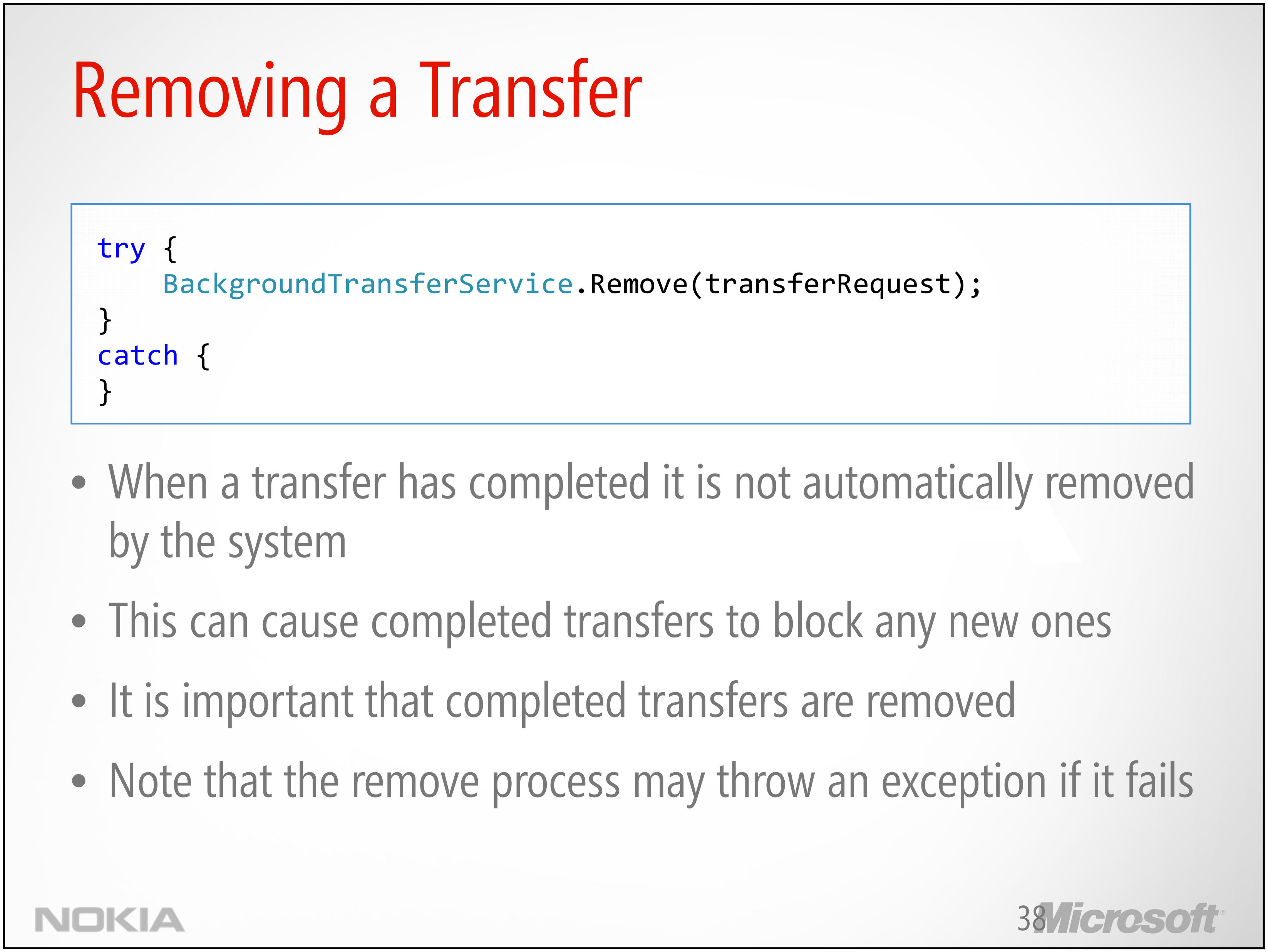
Transfer Status Changed

```
void request_TransferStatusChanged(object sender,
                                        BackgroundTransferEventArgs e) {
  switch (e.Request.TransferStatus) {
     case TransferStatus.Completed:
         // If the status code of a completed transfer is 200 or 206, the
         // transfer was successful
         if (transferRequest.StatusCode == 200 ||
             transferRequest.StatusCode == 206)
         {
            // File has arrived OK – use it in the program
         }
       ...
```

- This code checks that the file has arrived OK

37

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.



Removing a Transfer

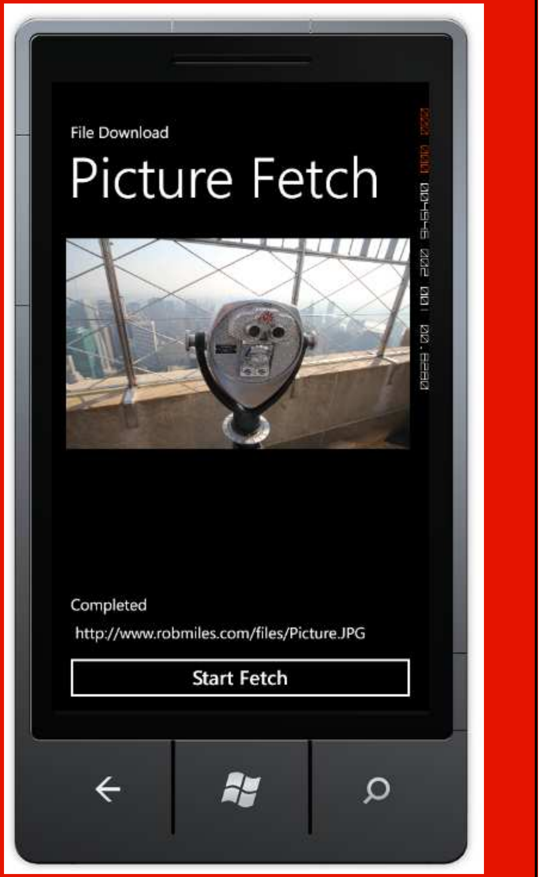
```
try {
    BackgroundTransferService.Remove(transferRequest);
}
catch {
}
```

- When a transfer has completed it is not automatically removed by the system
- This can cause completed transfers to block any new ones
- It is important that completed transfers are removed
- Note that the remove process may throw an exception if it fails

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.

Demo

Demo3: Picture Fetch



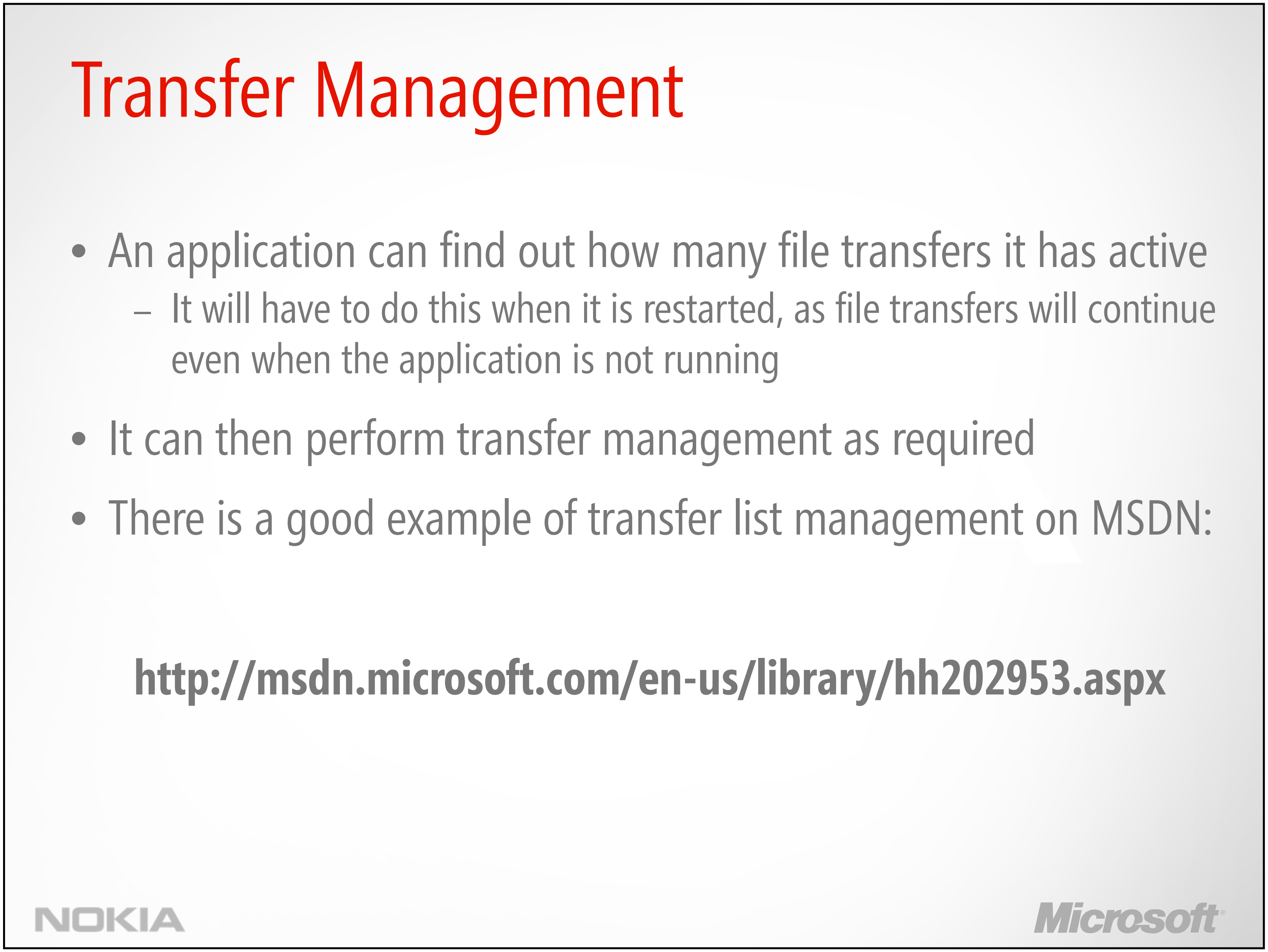
File Download

Picture Fetch

Completed

http://www.robmiles.com/files/Picture.JPG

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.

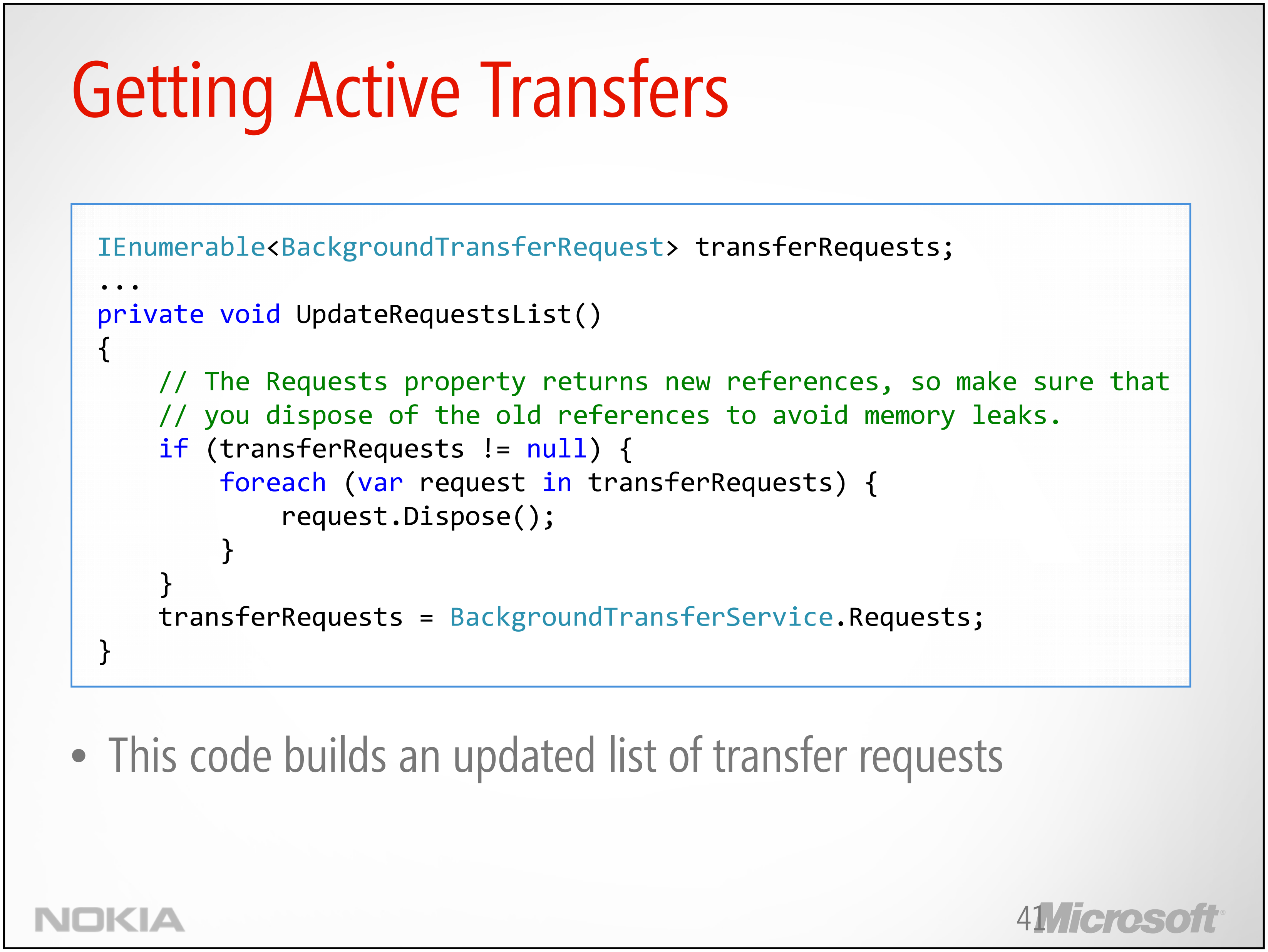


Transfer Management

- An application can find out how many file transfers it has active
  - It will have to do this when it is restarted, as file transfers will continue even when the application is not running
- It can then perform transfer management as required
- There is a good example of transfer list management on MSDN:

http://msdn.microsoft.com/en-us/library/hh202953.aspx

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.



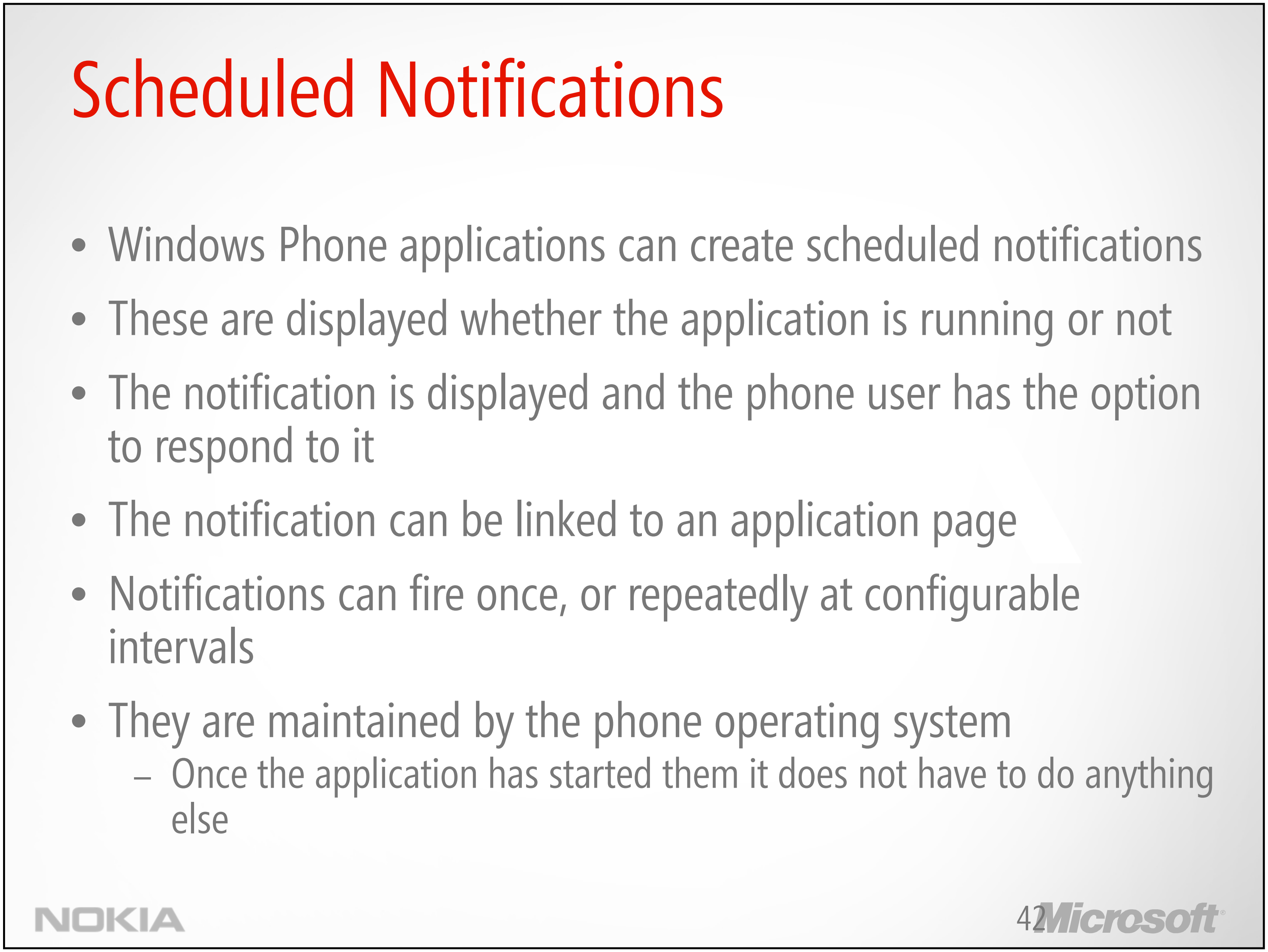
Getting Active Transfers

```
IEnumerable<BackgroundTransferRequest> transferRequests;
...
private void UpdateRequestsList()
{
    // The Requests property returns new references, so make sure that
    // you dispose of the old references to avoid memory leaks.
    if (transferRequests != null) {
        foreach (var request in transferRequests) {
            request.Dispose();
        }
    }
    transferRequests = BackgroundTransferService.Requests;
}
```

- This code builds an updated list of transfer requests

41

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.



Scheduled Notifications

- Windows Phone applications can create scheduled notifications
- These are displayed whether the application is running or not
- The notification is displayed and the phone user has the option to respond to it
- The notification can be linked to an application page
- Notifications can fire once, or repeatedly at configurable intervals
- They are maintained by the phone operating system
  - Once the application has started them it does not have to do anything else

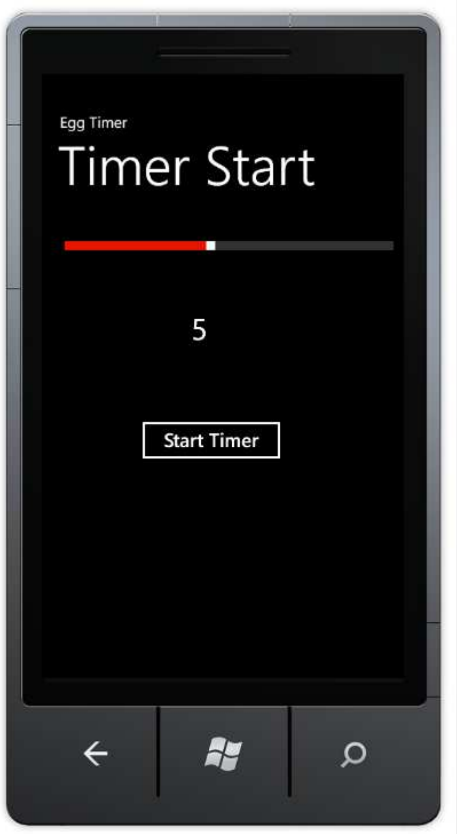
42

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.

Published: 8/22/2011

The "Egg Timer" Application

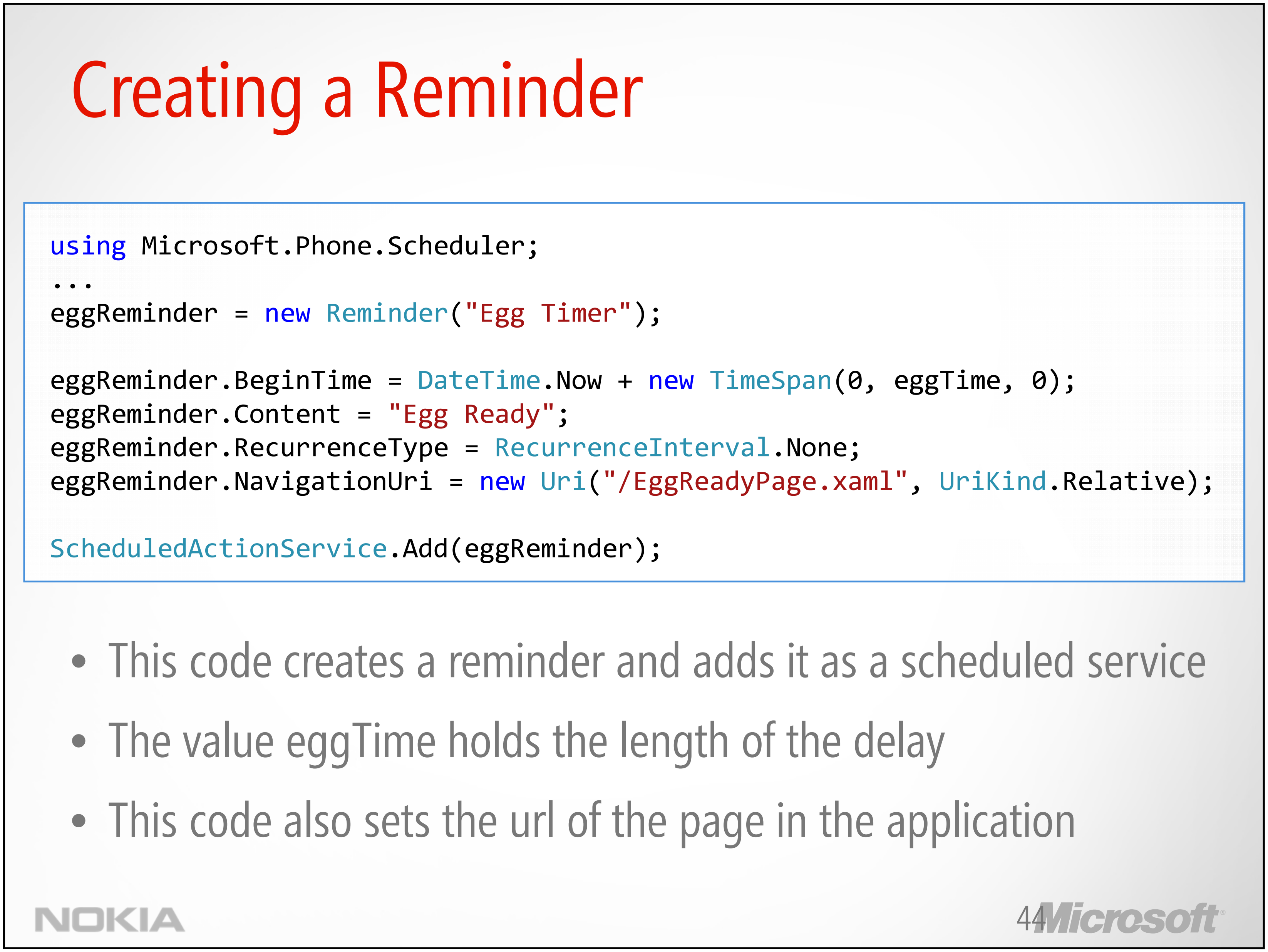
- This is a simple two page Silverlight Application
- The user sets the time using the slider and then presses the Start Timer button to create a notification
- When the notification fires the "Egg Ready" page is displayed if the user clicks through to the application



NOKIA

43 Microsoft

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.



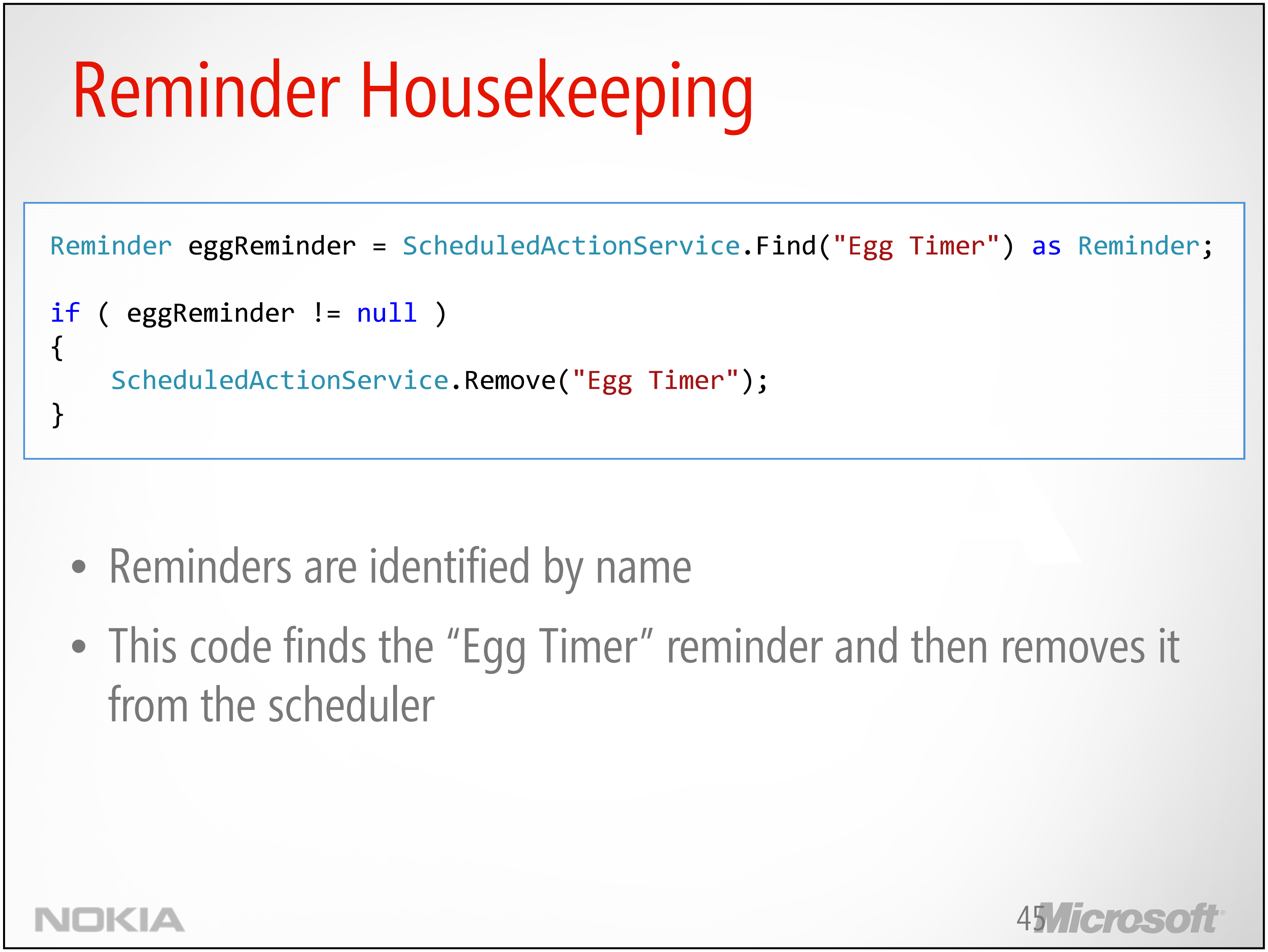
Creating a Reminder

```
using Microsoft.Phone.Scheduler;
...
eggReminder = new Reminder("Egg Timer");

eggReminder.BeginTime = DateTime.Now + new TimeSpan(0, eggTime, 0);
eggReminder.Content = "Egg Ready";
eggReminder.RecurrenceType = RecurrenceInterval.None;
eggReminder.NavigationUri = new Uri("/EggReadyPage.xaml", UriKind.Relative);

ScheduledActionService.Add(eggReminder);
```

- This code creates a reminder and adds it as a scheduled service
- The value eggTime holds the length of the delay
- This code also sets the url of the page in the application

44

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.



Reminder Housekeeping

```
Reminder eggReminder = ScheduledActionService.Find("Egg Timer") as Reminder;

if ( eggReminder != null )
{
    ScheduledActionService.Remove("Egg Timer");
}
```

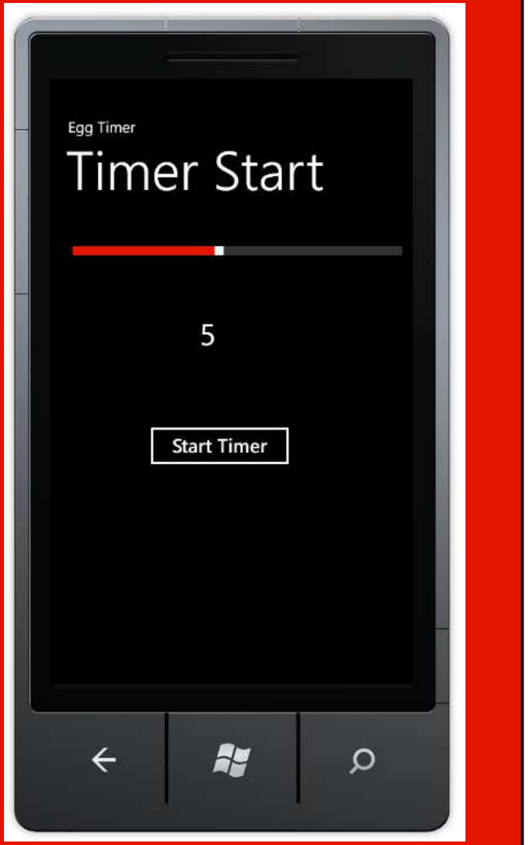
- Reminders are identified by name
- This code finds the “Egg Timer” reminder and then removes it from the scheduler

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.

Windows Phone

Demo

Demo4: Egg Timer


Egg Timer

Timer Start

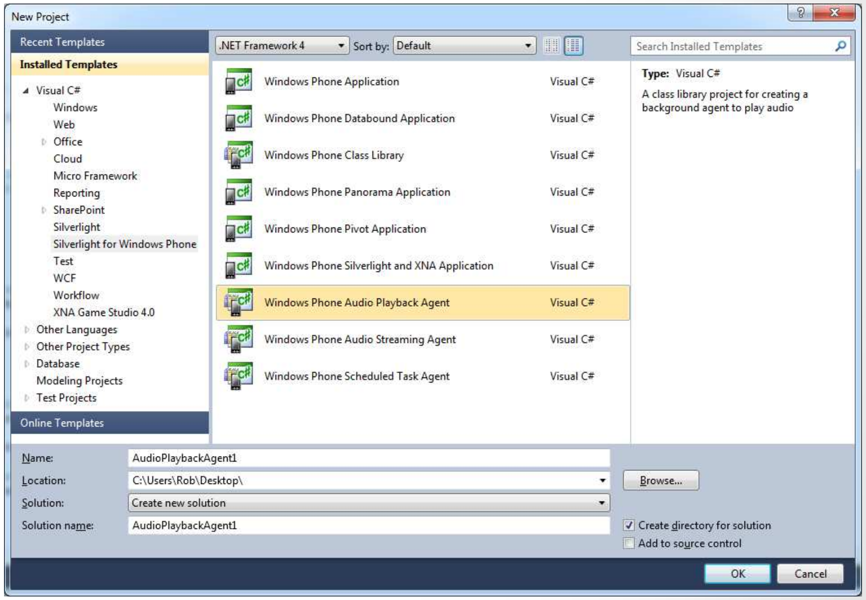
5

Start Timer

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.

Audio Playback Agents

- Also possible to create an Audio Playback Agent that will manage an application controlled playlist
- The mechanism is the same as for other background tasks
- The audio can be streamed or held in the application isolated storage



.NET Framework 4 Sort by: Default

Windows Phone Application	Visual C#
Windows Phone Databound Application	Visual C#
Windows Phone Class Library	Visual C#
Windows Phone Panorama Application	Visual C#
Windows Phone Pivot Application	Visual C#
Windows Phone Silverlight and XNA Application	Visual C#
Windows Phone Audio Playback Agent	Visual C#
Windows Phone Audio Streaming Agent	Visual C#
Windows Phone Scheduled Task Agent	Visual C#

Type: Visual C#

A class library project for creating a background agent to play audio

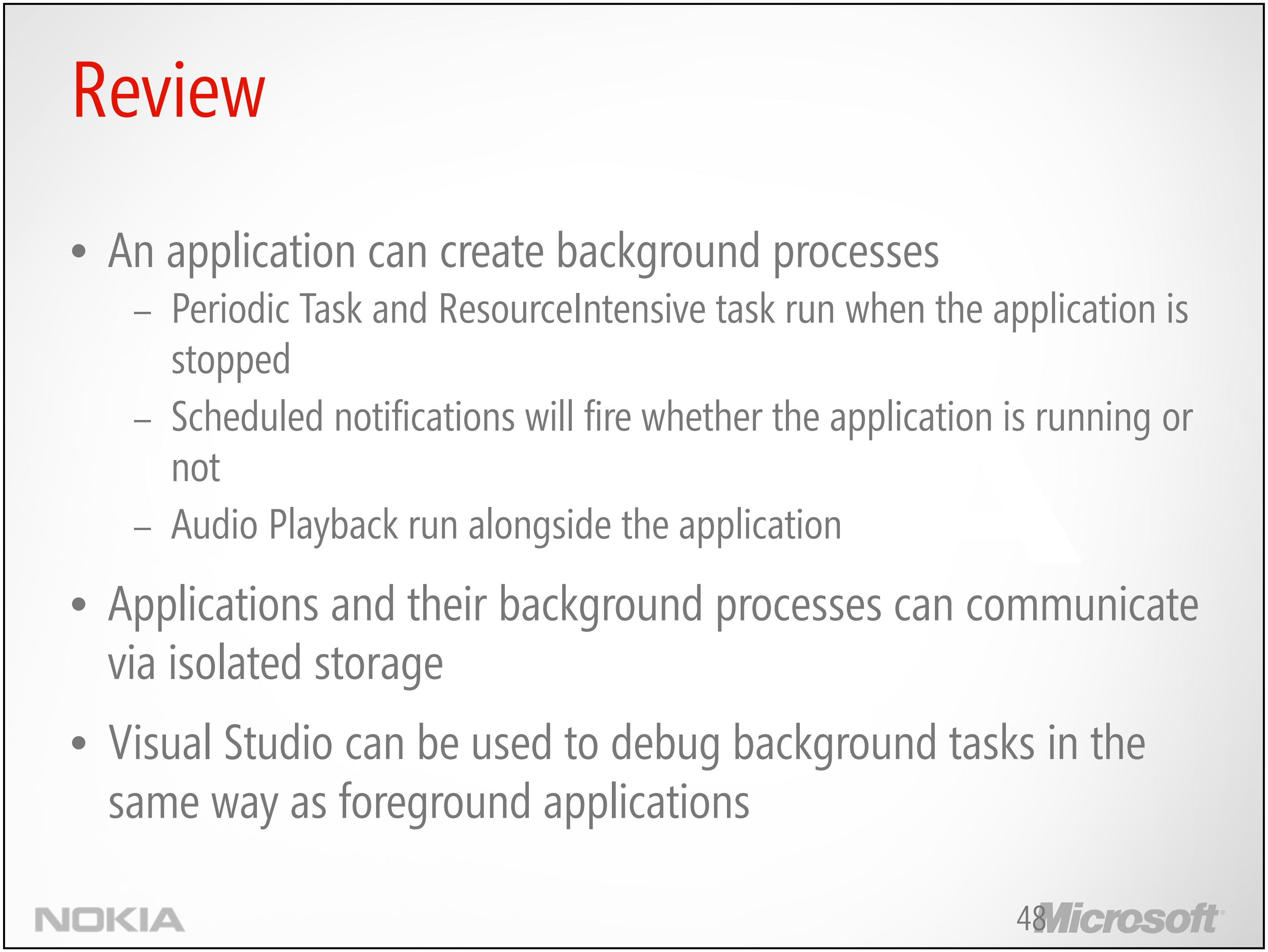
Name: AudioPlaybackAgent1
Location: C:\Users\Rob\Desktop\
Solution: Create new solution
Solution name: AudioPlaybackAgent1

☑ Create directory for solution
☐ Add to source control

NOKIA

47 Microsoft

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.



Review

- An application can create background processes
  - Periodic Task and ResourceIntensive task run when the application is stopped
  - Scheduled notifications will fire whether the application is running or not
  - Audio Playback run alongside the application
- Applications and their background processes can communicate via isolated storage
- Visual Studio can be used to debug background tasks in the same way as foreground applications

48

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.

Windows® Phone

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.

© 2011 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.

Microsoft®

©2011 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista, Windows Azure, Hyper-V 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.