



Windows



Introduzione alle Windows Store Universal App

Daniele Bochicchio

Network Manager @ ASPItalia.com

Chief Digital Officer @icubedsrl

Microsoft Regional Director

Microsoft MVP per ASP.NET

daniele@aspitalia.com - Twitter @dbochicchio

<http://blogs.aspitalia.com/daniele/>

Microsoft



Agenda

- Windows + Windows Phone = Universal App
- Shared Project
- XAML vs WinJS
- Novità di Visual Studio
- Migrazione di app esistenti

Universal app = tablet/desktop + smartphone

- Alla base di tutto c'è WinRT
 - Convergenza di app model, tool e store
- Si pubblica una sola app sullo store
 - Condividono il canale delle push, gli acquisti, etc
- Si sviluppano usando WinRT
 - Un unico framework per semplificarci la vita
- WinRT = Windows RunTime

Cos'è un'Universal App?

- Un'app WinRT è potenzialmente già un'Universal App
- Un'Universal App è un'app WinRT
- In futuro ci sarà il supporto anche per Xbox One

Supporto per le Universal App

- Tablet + desktop:
 - Windows 8.1 (Update 1)
- Smartphone
 - Windows 8.1 (Developer Preview, disponibile per il mercato da giugno 2014)
- Xbox One
 - Ogni Xbox One è un dev kit, ma non ci sono date di rilascio
- Per sviluppare ci serve Visual Studio 2013 Update 2
 - <http://aspit.co/auk>

Universal App = magia

- Un'Universal App è *magia* in fase di compilazione
- Il template aggiunge uno Shared Project in cui inserire il codice condiviso
- I file in questo progetto vengono automaticamente copiati nei progetti da cui è referenziato
- Consente di tenere condivisi qualsiasi tipo di risorsa
 - XAML/WinJS
 - Codice
- Supporta C#, C++ e JavaScript

Condividere il codice

- La condivisione è possibile grazie alle direttive di compilazione

```
#if WINDOWS_APP  
` codice per Windows  
#endif
```

```
#if WINDOWS_PHONE_APP  
` codice per Windows Phone  
#endif
```



- Oppure usando le partial class

Shared Projects vs Portable Class Library

- Shared Project = condivisione di sorgente
 - C'è il supporto per C#, C++ e JavaScript
 - Incluso e compilato localmente al progetto
- Portable Class Library = one binary
 - È limitato al .NET Framework
 - Pertanto c'è il supporto per C# e VB
 - Compila una volta per ogni piattaforma e condivide il compilato

Quanto posso condividere?

- View
 - XAML: il Common XAML Framework consente una buona portabilità, grazie anche ai nuovi controlli
 - WinJS: controlli e librerie specifiche
- Codice: grazie alle direttive di compilazione, quasi tutto
 - 90% delle API portate anche su Windows Phone
 - mancano WCF, CameraCaptureTask, Lock Screen, Ringtone provider, VOIP, Lenses e altro
- Migliorie alle Portable Class Library se volete condividere il codice anche con piattaforme precedenti (WP7/8)

Quanto posso condividere?

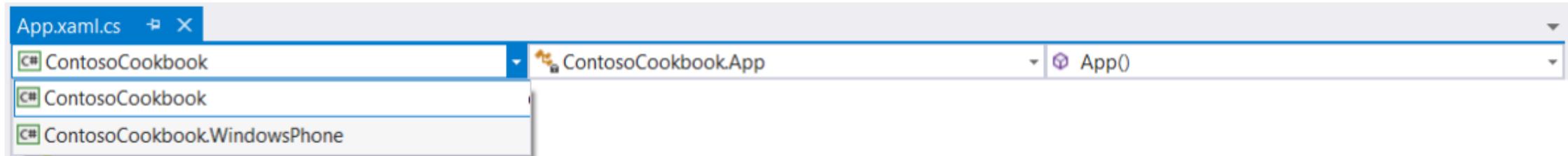
- A tendere ci sarà il 100% di convergenza
- Oggi per alcuni tipi di app per Windows Phone bisogna continuare ad usare Silverlight
 - Ma ci sono alcuni scenari non supportati, come i background player
- Un'Universal App condivide impostazioni e data
 - Se salvate i dati nel roaming storage, saranno sincronizzati con il backup
 - Scenario: installo l'app sul telefono, mi trovo tutto configurato come sul tablet

Demo

- La nostra prima Universal App

Visual Studio: navbar

- Contesto dalla navbar



- Consente di cambiare al volo il contesto del codice e del designer

Visual Studio: Intellisense

- L'Intellisense avvisa di mancanze specifiche e/o differenze tra le piattaforme

The screenshot shows a code editor in Visual Studio displaying C# code. A tooltip is open over the text 'Windows.Phone.UI.Input.HardwareButtons.', which is underlined with a red squiggle, indicating a potential issue or missing reference. The tooltip contains the following information:

- BackPressed
- CameraHalfPressed
- CameraPressed
- CameraReleased
- Equals
- ReferenceEquals

Event Handler: EventHandler<Windows.Phone.UI.Input.BackPressedEventArgs> HardwareButtons.BackPressed

Description: Occurs when the user presses the hardware Back button.

Notes: Windows Phone 8.1 - Available
Windows 8.1 - Not available

You can use the navigation bar to switch context

```
Window.Current.CoreWindow.Dispatcher.AcceleratorKeyActivated +=  
    CoreDispatcher_AcceleratorKeyActivated;  
Window.Current.CoreWindow.PointerPressed +=  
    this.CoreWindow_PointerPressed;  
}  
#if WINDOWS_PHONE_APP  
    Windows.Phone.UI.Input.HardwareButtons.  
#endif  
};
```

Visual Studio: Intellisense

- L'Intellisense avvisa di mancanze specifiche e/o differenze tra piattaforme

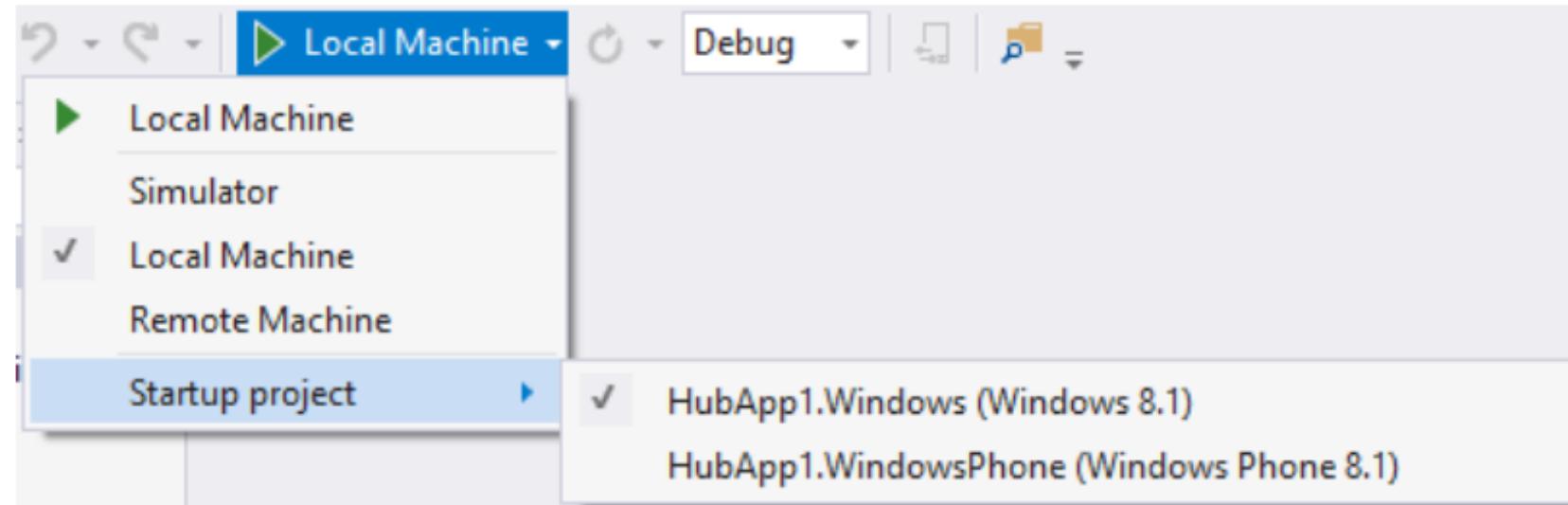
The screenshot shows a code editor in Visual Studio displaying C# code for a Windows Phone application. The code includes event handlers for the AcceleratorKeyActivated and PointerPressed events, and a conditional block for the WINDOWS_PHONE_APP macro. A tooltip from Intellisense is open over the `Windows.Phone.UI.Input.HardwareButtons.` part of the code, listing several methods: BackPressed, CameraHalfPressed, CameraPressed, CameraReleased, Equals, and ReferenceEquals. The `BackPressed` method is highlighted. The tooltip also provides documentation for the `BackPressed` event, stating it occurs when the user presses the hardware Back button, is available on Windows Phone 8.1, and is not available on Windows 8.1. It also includes a note about switching contexts using the navigation bar.

```
Window.Current.CoreWindow.Dispatcher.AcceleratorKeyActivated +=  
    CoreDispatcher_AcceleratorKeyActivated;  
Window.Current.CoreWindow.PointerPressed +=  
    this.CoreWindow_PointerPressed;  
}  
#if WINDOWS_PHONE_APP  
    Windows.Phone.UI.Input.HardwareButtons.  
#endif  
};
```

⚡ BackPressed ⚠ EventHandler<Windows.Phone.UI.Input.BackPressedEventArgs> HardwareButtonsBackPressed
⚡ CameraHalfPressed ⚠ Occurs when the user presses the hardware Back button.
⚡ CameraPressed ⚠
⚡ CameraReleased ⚠
❼ Equals ⚠
❼ ReferenceEquals ⚠
Windows Phone 8.1 - Available
Windows 8.1 - Not available
You can use the navigation bar to switch context

Visual Studio: startup project

- Facilmente modificabile dalla barra Run, per semplificare il debugging



- Visualizzato in caso di presenza di più di un progetto

Quanto posso migrare?

- Se avete un'app per Windows Store, siete sulla buona strada
 - Tasto destro dentro il progetto in Visual Studio, poi *Add Windows Phone 8.1 app*
- Ma probabilmente partire da un'app Silverlight...
- E quindi dovete rifare l'app
 - Avrete a *costo-quasi-zero* anche quella per tablet
- Lo XAML si migra facilmente
 - Al netto di alcuni differenti controlli
 - Ad esempio, Panorama = Hub, Pivot = FlipView

Quanto posso migrare?

- Per il codice: dipende da cosa c'è all'interno
 - Se è codice che accede all'hardware o è specifico, occorrerà probabilmente rifarlo
- Se avete usato M-V-VM per la vostra app siete a buon punto
 - Avete già pensato la vostra app in maniera agnosta rispetto alla View
- Ci sono delle zone grigie
 - Niente WCF, CameraCaptureTask, LockScreen, RingTone, etc

Demo

- Migrare un'app esistente

One app, multiple screens

- Bisogna iniziare a pensare ai layout in maniera fluida
- Windows ha infinite risoluzioni, Windows Phone 3 form factor (16:9, 16:10, 15:9)
- XAML: layout con dimensioni in percentuale
 - In alcuni casi può aiutarci il controllo `ViewBox`
- WinJS: i CSS e le media queries consentono di adattare il layout alle differenti risoluzioni

Alcuni trucchi per le View

- Usate il più possibile gli stili
 - Consentono di diversificare la resa grafica in base al tipo di piattaforma
- Se parte del layout cambia, incapsulatelo in uno User Control
 - Potrete così variare il risultato e riutilizzarlo in parti della vostra app
- Tenete semplice lo XAML/markup
- Capite i diversi use case dell'UX di tablet e telefono
 - E non abbiate paura di diversificare il comportamento

Bonus content: Live Tile

- Le Live Tile sono il sistema più semplice per convogliare
- Le Universal App consentono di avere un solo formato per le Tile
- Windows Phone non supporta le large (310x310)
- Stesso sistema di push lato server
 - Meno investimenti in codice
- Questo concetto si estende anche alle tile associate ad un sito in IE per Windows Phone 8.1



Bonus Demo

- Live Tile

Recap

- Universal App è per eliminare l'effort di avere 2 app differenti per tablet e smartphone
- Le Universal App offrono una serie di API per l'accesso alle funzionalità del sistema operativo attraverso WinRT
- Serve Visual Studio 2013 Update 2 per sviluppare
- Ci sono tante opportunità per condividere codice: non perdete le prossime sessioni!



The information contained in this document represents the current view of Microsoft Corp. on the issues discussed as of the date of publication. 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 presented after the date of publication. This guide is for informational purposes only. **MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS DOCUMENT.** Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form, by any means (electronic, mechanical, photocopying, recording or otherwise), or for any purpose, without the express written permission of Microsoft. Microsoft may have patents, patent applications, trademarks, copyrights or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights or other intellectual property. The example companies, organizations, products, domain names, email addresses, logos, people, places, and events depicted herein are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

© 2013 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.**