



WPF(.NET3.5)

vs Silverlight 2.0 Beta1

*+Beta2?*

中 博俊@わんくま同盟



## アジェンダ

- Silverlight2.0までの  
おさらい
- 細かい違いを見よう
- まとめ

# Silverlight2.0までのおさらい



## Silverlightとは

- 1.0が2007/09/04リリース
  - <http://www.microsoft.com/presspass/press/2007/sep07/09-04SilverlightPR.msp>
- Flashキラー？
- XAML(サブセット)で見た目を作成する
- OSがWindowsとMacintosh
- ブラウザがIE, Safari, Firefox, Mozilla

## Silverlightとは

- Silverlight Mobile (Nokia) が発表されました。
  - <http://www.nokia.com/A4136001?newsid=1197788>
  - 日本上陸は？

## 2.0の特徴

- MiniCLRが搭載されている
- Xamlのデータモデルが結構変更されている。
- 日本語対応

2.0って？

- 2.0は今年中リリース予定？

- <http://blogs.msdn.com/ashish/archive/2008/04/03/silverlight-roadmap-questions.aspx>

- We are targeting late Summer

- Here is a rough timeline:

Silverlight 2.0 Beta 1 (Q1CY08 with limited (non commercial) Go-Live)

Silverlight 2.0 Beta 2 (Q2CY08 with Go-Live) ←イマココ

Silverlight 2.0 RTM (Summer 2008) – Exact timing TBD

- Silverlight v.next – We are working on a v.Next plan and have nothing to announce at this time

- Silverlight for mobile – No date available

CY = Calendar Year = 暦日

Q = Quarter = 4半期

H = Half = 半期



## 2.0 Beta1を試す環境

- Silverlight 2.0 SDK Beta1

- <http://www.microsoft.com/downloads/details.aspx?FamilyId=4E03409A-77F3-413F-B108-1243C243C4FE&displaylang=en>

- Microsoft® Silverlight™ 2 Software Development Kit Beta 1 Documentation

- <http://www.microsoft.com/downloads/details.aspx?FamilyId=1840CAB5-196C-4264-B55D-562242A72625&displaylang=en>

- Tools Beta1

- <http://www.microsoft.com/downloads/details.aspx?displaylang=ja&FamilyID=e0bae58e-9c0b-4090-a1db-f134d9f095fd>

- Blend 2.5 March 2008 Preview

- <http://www.microsoft.com/downloads/details.aspx?FamilyId=32A3E916-E681-4955-BC9F-CFBA49273C7C&displaylang=en>





## 2.0 Beta2を試す環境

- Microsoft® Silverlight™ 2 Software Development Kit Beta 2  
– <http://www.microsoft.com/downloads/details.aspx?FamilyId=4E03409A-77F3-413F-B108-1243C243C4FE&displaylang=en>
- Microsoft® Silverlight™ 2 Software Development Kit Beta 2 Documentation  
– <https://www.microsoft.com/downloads/details.aspx?FamilyID=bce7684a-507b-4fc6-bc99-6933cd690cab&DisplayLang=en>
- Microsoft Silverlight Tools Beta 2 for Visual Studio 2008  
– <https://www.microsoft.com/downloads/details.aspx?FamilyID=50a9ec01-267b-4521-b7d7-c0dba8866434&DisplayLang=en>
- Microsoft Expression Blend 2.5 June 2008 Preview  
– <https://www.microsoft.com/downloads/details.aspx?FamilyID=32a3e916-e681-4955-bc9f-cfba49273c7c&DisplayLang=en>

*Beta2*

細かい違いをみてみよう



## 起動ルーチン

- WPFのデフォルトはStartupUri

```
<Application x:Class="WpfApplication"
  xmlns="http://schemas.microsoft.com/winfx/2006/xaml"
  xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
  StartupUri="Window1.xaml">
  <Application.Resources>
```

WPF

- ほかにApplication.Startイベントを利用する。

## 起動ルーチン

- SilverlightのデフォルトはStartupイベント

```
public App()  
{  
    this.Startup += this.Application_Startup;  
    this.Exit += this.Application_Exit;  
    this.UnhandledException += this.Application_UnhandledException;  
  
    InitializeComponent();  
}  
  
private void Application_Startup(object sender, StartupEventArgs e)  
{  
    // メインコントロールを読み込みます。  
    this.RootVisual = new Page();  
}
```

- RootVisualにPageを入れています。

**Silverlight**



# Application

# WPF

- Application Class
  - Application Members
  - Application Constructor
- Application Methods
  - GetResourceStream Method
    - GetResourceStream Method
    - GetResourceStream Method
  - LoadComponent Method
- Application Properties
  - Current Property
  - Host Property
  - Resources Property
  - RootVisual Property
- Application Events
  - Exit Event
  - Startup Event
  - UnhandledException Event

- Application メソッド
  - FindResource メソッド
  - GetContentStream メソッド
  - GetCookie メソッド
  - GetRemoteStream メソッド
  - GetResourceStream メソッド
  - LoadComponent メソッド
  - OnActivated メソッド
  - OnDeactivated メソッド
  - OnExit メソッド
  - OnFragmentNavigation メソッド
  - OnLoadCompleted メソッド
  - OnNavigated メソッド
  - OnNavigating メソッド
  - OnNavigationFailed メソッド
  - OnNavigationProgress メソッド
  - OnNavigationStopped メソッド
  - OnSessionEnding メソッド
  - OnStartup メソッド
  - Run メソッド
  - SetCookie メソッド
  - Shutdown メソッド
  - TryFindResource メソッド

- Application プロパティ
  - Current プロパティ
  - MainWindow プロパティ
  - Properties プロパティ
  - ResourceAssembly プロパティ
  - Resources プロパティ
  - ShutdownMode プロパティ
  - StartupUri プロパティ
  - Windows プロパティ
- Application イベント
  - Activated イベント
  - Deactivated イベント
  - DispatcherUnhandledException イベント
  - Exit イベント
  - FragmentNavigation イベント
  - LoadCompleted イベント
  - Navigated イベント
  - Navigating イベント
  - NavigationFailed イベント
  - NavigationProgress イベント
  - NavigationStopped イベント
  - SessionEnding イベント
  - Startup イベント

# Silverlight

- 必要最小限に絞り込まれています



## Application

- 終了させる方法は？

```
Application.Current.MainWindow.Close();
```

# WPF

- たぶんSilverlightはない

# Silverlight

- Flashっぽいアプリケーションなら頭から再生を用意しよう。
- 業務系ならあっさり<TextBlock Text=“終了しました”>でいいかもしれない。
- Application.Exitイベントが走るのはブラウザが閉じられるか、別ページに行ったとき
- Webのお約束で、実行を期待しないこと。

## 最初の画面、画面遷移

- WPFはWindow作成

```
<Window x:Class="WpfApplication1.Window1"
  xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
  xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
  Title="Window1" Height="300" Width="300">
  <Grid>

  </Grid>
</Window>
```

# WPF

- SilverlightはすべてUserControl

```
<UserControl x:Class="SilverlightApplication1.Page"
  xmlns="http://schemas.microsoft.com/client/2007"
  xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
  Width="400" Height="300">
  <Grid x:Name="LayoutRoot" Background="White">

  </Grid>
</UserControl>
```

# Silverlight



## 最初の画面、画面遷移

- 画面遷移などは？

```
private void clk(object sender, RoutedEventArgs e)
{
    Hide();
    new Window2().ShowDialog();
    Show();
}
```

WPF

- 今までのWindowsFormと特に変わらない



## 最初の画面、画面遷移

- Windowがない
- 画面遷移などは？

```
private void clk(object sender, RoutedEventArgs e)
{
    var visual = (Page)Application.Current.RootVisual;
    visual.LayoutRoot.Children.Clear();
    visual.LayoutRoot.Children.Add(new Page2());
}
```

```
<UserControl x:Class="SilverlightApplication1.Page"
    xmlns="http://schemas.microsoft.com/client/2007"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    Width="400" Height="300">
    <Grid x:Name="LayoutRoot" Background="White">
        <Button Click="clk" />
    </Grid>
</UserControl>
```

**Silverlight**



最初の画面、画面遷移

DEMO1



わんくま同盟 東京勉強会#21

## メッセージボックス

# • いつものメッセージボックス WPF

```
if ( MessageBox.Show("登録しますか?", "確認",  
    MessageBoxButton.YesNo, MessageBoxImage.Question )  
    == MessageBoxResult.Yes )  
{  
    //処理  
}
```

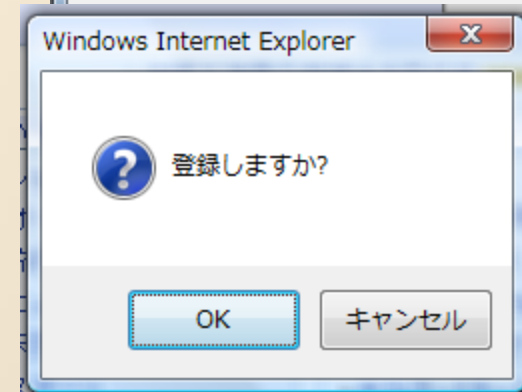
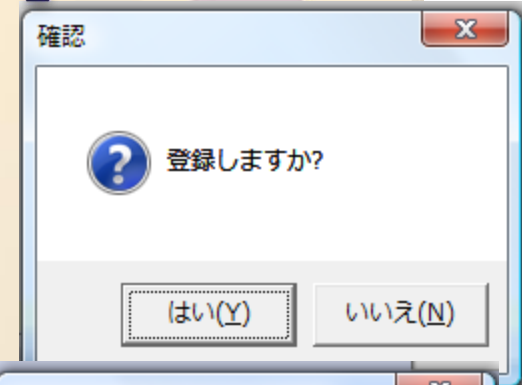
## • Javascript



```
javascript:confirm('登録しますか?')
```

## • Silverlightはホストページのconfirmをつかう

```
private void clk(object sender, RoutedEventArgs e)  
{  
    HtmlPage.Window.Confirm("登録しますか?");  
}
```



# Silverlight











メッセージボックス

DEMO2



わんくま同盟 東京勉強会#21

## System.Windows.Browser 名前空間

	Class
	<a href="#">BrowserInformation</a>
	<a href="#">HtmlDocument</a>
	<a href="#">HtmlElement</a>
	<a href="#">HtmlElementCollection</a>
	<a href="#">HtmlEventArgs</a>
	<a href="#">HtmlObject</a>
	<a href="#">HtmlPage</a>
	<a href="#">HtmlWindow</a>
	<a href="#">HttpUtility</a>
	<a href="#">ScriptableMemberAttribute</a>
	<a href="#">ScriptableTypeAttribute</a>
	<a href="#">ScriptObject</a>

一通りのDOMコントロールはできるようです。

## Panel要素

# WPF

# Silverlight

[Windows.Controls.Panel](#)

[Windows.Controls.Canvas](#)

[Windows.Controls.DockPanel](#)

[Windows.Controls.Grid](#)

[Windows.Controls.StackPanel](#)

[Windows.Controls.VirtualizingPanel](#)

[Windows.Controls.WrapPanel](#)

[Windows.Controls.Primitives.TabPanel](#)

[Windows.Controls.Primitives.ToolBarOverflowPanel](#)

[Windows.Controls.Primitives.UniformGrid](#)

[System.Windows.Controls.Panel](#)

[System.Windows.Controls.Canvas](#)

[System.Windows.Controls.DataGridCellsPresenter](#)

[System.Windows.Controls.DataGridColumnHeadersPresenter](#)

[System.Windows.Controls.DataGridDetailsPresenter](#)

[System.Windows.Controls.DataGridRowsPresenter](#)

[System.Windows.Controls.Grid](#)

[System.Windows.Controls.Primitives.TabPanel](#)

[System.Windows.Controls.StackPanel](#)

- Grid, StackPanelが2.0で増えた。
- WrapPanelがないのがちょっと残念
  - (Gridで代用)



# WPF

## Binding

# Silverlight

- Binding プロパティ
  - AsyncState プロパティ
  - BindsDirectlyToSource プロパティ
  - Converter プロパティ
  - ConverterCulture プロパティ
  - ConverterParameter プロパティ
  - ElementName プロパティ
  - IsAsync プロパティ
  - Mode プロパティ
  - NotifyOnSourceUpdated プロパティ
  - NotifyOnTargetUpdated プロパティ
  - NotifyOnValidationError プロパティ
  - Path プロパティ
  - RelativeSource プロパティ
  - Source プロパティ
  - UpdateSourceExceptionFilter プロパティ
  - UpdateSourceTrigger プロパティ
  - ValidatesOnDataErrors プロパティ
  - ValidatesOnExceptions プロパティ

- System.Windows.Data Namespace
  - Binding Class
    - Binding Members
    - Binding Constructor
    - Binding Methods
  - Binding Properties
    - Converter Property
    - ConverterCulture Property
    - ConverterParameter Property
    - Mode Property
    - Source Property
  - BindingMode Enumeration

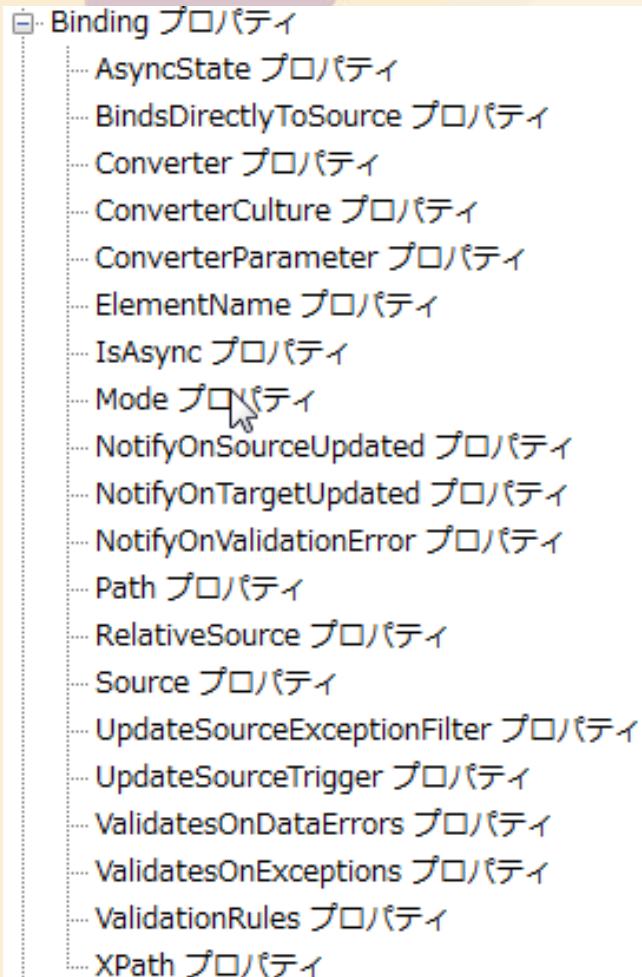
- Binding Class
  - Binding Members
  - Binding Constructor
  - Binding Methods
  - Binding Properties
    - Converter Property
    - ConverterCulture Property
    - ConverterParameter Property
    - Mode Property
    - NotifyOnValidationError
    - Path Property
    - Source Property
    - ValidatesOnExceptions

```
<TextBox Text="{Binding Path=text1}" />  
<TextBox Text="{Binding Path=text2}" Grid  
<TextBox Text="{Binding Path=text3}" Grid
```

```
<TextBox Text="{Binding text1}" />  
<TextBox Text="{Binding text2}" Grid  
<TextBox Text="{Binding text3}" Grid  
<Button Click="clk" Grid.Row="2" />
```



# Binding



Binding プロパティ

- AsyncState プロパティ
- BindsDirectlyToSource プロパティ
- Converter プロパティ
- ConverterCulture プロパティ
- ConverterParameter プロパティ
- ElementName プロパティ
- IsAsync プロパティ
- Mode プロパティ
- NotifyOnSourceUpdated プロパティ
- NotifyOnTargetUpdated プロパティ
- NotifyOnValidationError プロパティ
- Path プロパティ
- RelativeSource プロパティ
- Source プロパティ
- UpdateSourceExceptionFilter プロパティ
- UpdateSourceTrigger プロパティ
- ValidatesOnDataErrors プロパティ
- ValidatesOnExceptions プロパティ
- ValidationRules プロパティ
- XPath プロパティ

- ポイントになりそうな欠け
  - ElementName
    - 別の要素の値へのバインド
  - RelativeSource
    - VisualTreeをたどってのバインド
  - XPath
    - XMLへの直接のバインド



Binding

DEMO3



わんくま同盟 東京勉強会#21

# WPF

## Binding

# Silverlight

[BindingMode](#) 値のいずれか。  
既定値は [Default](#) で、これは  
対象となる依存関係プロパ  
ティの既定のバインディング  
モード値を返します。

### Binding.Mode Property

[Binding Class](#) [Example](#) [See Also](#) [Send Feedback](#)

#### Property Value

Type: [System.Windows.Data.BindingMode](#)

One of the [BindingMode](#) values. The default is [BindingMode.OneWay](#).

メンバ名
TwoWay
OneWay
OneTime
OneWayToSource
Default

反映させるコントロール  
には

```
{Binding text1,  
Mode=TwoWay}
```

と明示的に書かねばいけ  
ない。

### Members

Member name	D
OneWay	U ci p
OneTime	U ci
TwoWay	U e ta



## UIElement.IsEnabledと、プロパティの継承

- <http://d.hatena.ne.jp/Yamaki/20080409/1207721518>
- Silverlight2では依存プロパティ値を継承させる方法がない
- <http://d.hatena.ne.jp/Yamaki/20080610/1213085261>
- Beta2で解禁

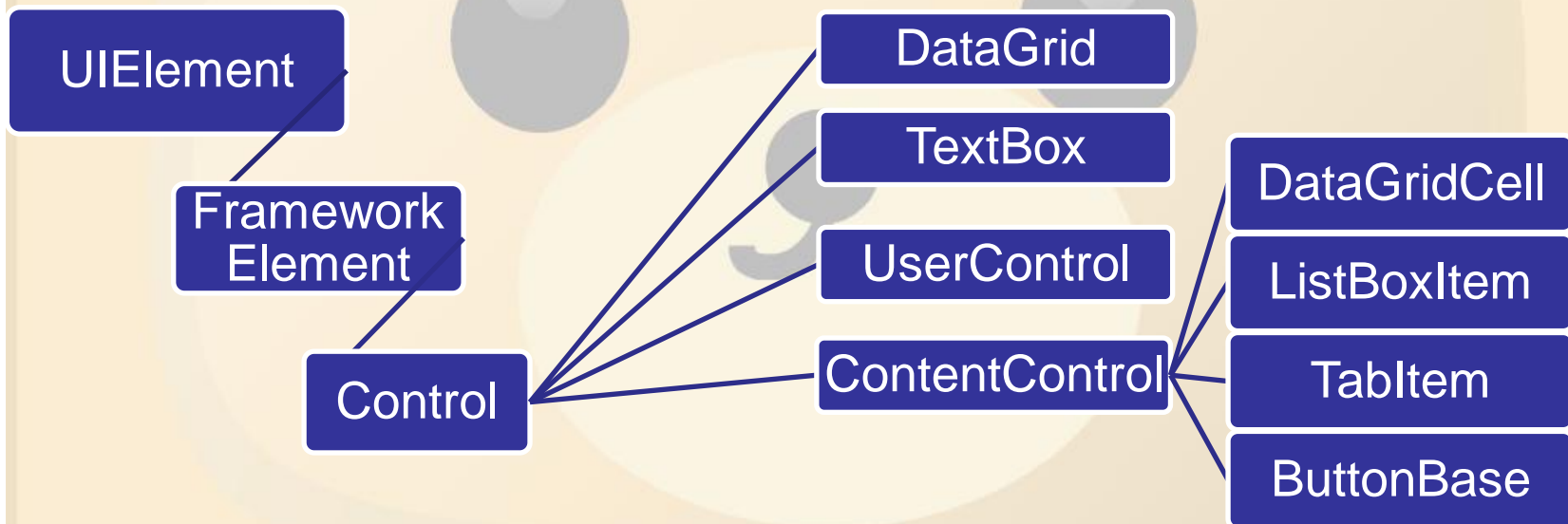
*Beta1*

*+Beta2*

## UIElement.IsEnabledと、プロパティの継承

+Beta2

- UIElement.IsEnabledがない
- ContentControl.IsEnabledがある
- →パネルの範囲を無効にするとかができない。



# まとめ



- WPF知ってればSilverlightは全く怖くない
- はまるポイント(できること、できないことをきっちりとは)は確実に抑えよう
- そのためにはSilverlightアプリもWPFアプリも作っておくことが必要。
- RTWがめっちゃめっちゃたのしみです。

Enjoy WPF & Silverlight

