

DEV  intersection
europe 2015

<anglebrackets/>
europe 2015

WPF Data Binding In Depth

Brian Noyes

CTO & Co-founder, Solliance
brian.noyes@solliance.net, @briannoyes



Tweet this talk

#DEVintersectionEU@briannoyes

DEV  intersection
europe 2015

<anglebrackets/>
europe 2015

Thank You Sponsors!

 GitHub

 DevExpress

 JetBrains

 SDN
SOFTWARE DEVELOPMENT NETWORK
www.sdn.nl

About Brian Noyes

www.solliance.net

CTO and Co-founder, Solliance
www.solliance.net

Microsoft Regional Director

Microsoft MVP

Microsoft
Regional Director



Pluralsight author
www.pluralsight.com

pluralsight

Web API Insider, Windows Azure Insider,
Window Store App Insider, C#/VB Insider

 brian.noyes@solliance.net

 [@briannoyes](https://twitter.com/briannoyes)

 <http://briannoyes.net>

Agenda

- Data Binding Overview
- Data Sources
- Bindings
- Data Templates
- Validation



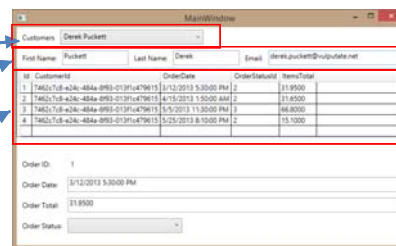
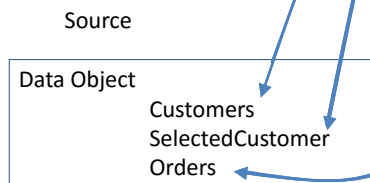
Data Binding Overview

- One of the most powerful XAML platform features
- Promotes loose coupling
 - View \neq Model
 - View \neq Data access
 - View \neq Interaction logic
- Reduces code / maintenance effort
 - Automatic data flow to-from screen
 - Per property push-pull logic
 - Natural associations between screen elements and data objects
- Facilitates Developer-Designer workflow
 - Lets designers work on XAML in isolation
- Key enabler for Model-View-ViewModel (MVVM) pattern
 - Bindings are the non-sticky bond between a view and its view model
 - Provide two way data flow and communication



Sources and Targets

Data binding flows data between a source object property and target element property



Target

Target properties must be DependencyProperties

Data Binding Building Blocks

DataContext

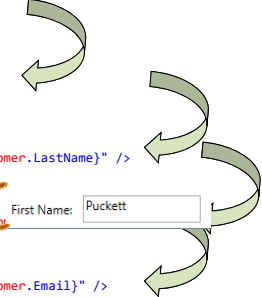
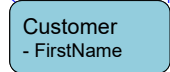
Bindings

XAML Parsing

Change Notifications

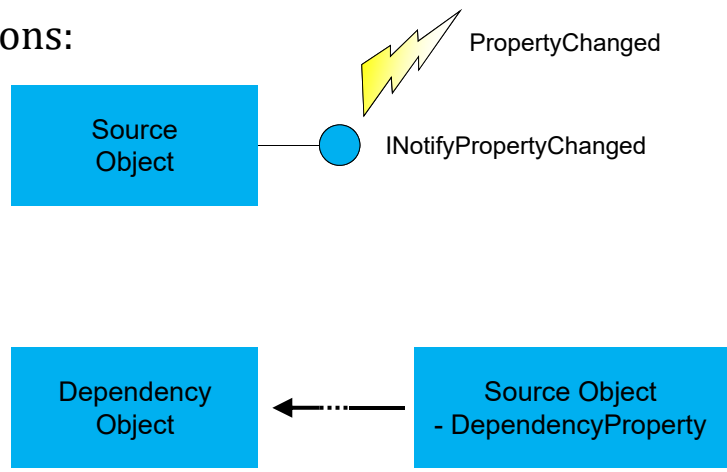
```

<Grid Grid.Row="1">
  <Grid.ColumnDefinitions>
    <ColumnDefinition Width="Auto" />
    <ColumnDefinition Width="*" />
    <ColumnDefinition Width="Auto" />
    <ColumnDefinition Width="*" />
    <ColumnDefinition Width="Auto" />
    <ColumnDefinition Width="2*" />
  </Grid.ColumnDefinitions>
  <Label Content="First Name:" />
  <TextBox Grid.Column="1"
    Text="{Binding SelectedCustomer.LastName}" />
  <Label Content="Last Name:"
    Grid.Column="2" />
  <TextBox Grid.Column="3"
    Text="{Binding SelectedCustomer.FirstName}" />
  <Grid.Column="4" />
  <TextBox Grid.Column="5"
    Text="{Binding SelectedCustomer.Email}" />
</Grid>
  
```



Supporting Property Change Notifications

- Two options:



Other Important Interfaces

- ICollectionView – sorting, filtering, paging, currency
- IEditableObject – rollback changes
- INotifyDataErrorInfo / IDataErrorInfo – validation support



Binding Sources

- The source object is determined by the binding
- Bindings use the element DataContext by default
- Can directly indicate the source object one of three other ways:
 - RelativeSource
 - Obtain object reference from a type / position in the element hierarchy relative to the current element
 - ElementName
 - Obtain object reference for some other named element within the visual tree
 - Source
 - Obtain object reference from a Resource object



Binding Property Paths

- The Path property on a Binding determines where on the source object to obtain the bound value
- Can pass Path to binding through parameterized constructor or property
 - {Binding Name} <==> {Binding Path=Name}
- Simple path: property name on source object
 - Path=Name
- Complex path: walk object graph
 - Path=Customer.Orders[1].OrderDetails[2].Product.Name
- Whole source object
 - {Binding} <==> {Binding Path=,}



Data Templates

```
<ContentControl Grid.Row="1" Content="{Binding CurrentViewModel}"/>
```

DataContext=CurrentViewModel →

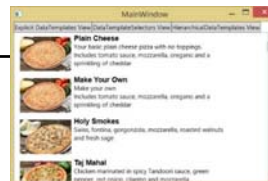
```
<DataTemplate DataType="{x:Type local:HierarchicalDataTemplatesViewModel}">  
<local:HierarchicalDataTemplatesView/>  
</DataTemplate>
```



DataTemplates Overview

```
<ListBox x:Name="ProductsListBox" ItemsSource="{Binding Products}"
        ItemTemplate="{StaticResource ProductTemplate}" />
```

```
<DataTemplate x:Key="ProductTemplate">
  <Grid>
    <Grid.ColumnDefinitions>
      <ColumnDefinition Width="Auto"/>
      <ColumnDefinition Width="Auto"/>
    </Grid.ColumnDefinitions>
    <Image x:Name="ProductImage" Source="{Binding Image}" Margin="5" Width="100"/>
    <Grid Grid.Column="1" Height="{Binding ElementName=ProductImage, Path=Height}">
      <Grid.RowDefinitions>
        <RowDefinition Height="Auto"/>
        <RowDefinition Height="Auto"/>
      </Grid.RowDefinitions>
      <TextBlock Text="{Binding Name}" FontFamily="Arial" FontSize="14" FontWeight="Bold"/>
      <TextBlock Grid.Row="1" Text="{Binding Description}" TextWrapping="Wrap" Width="300"/>
    </Grid>
  </Grid>
</DataTemplate>
```






Validation

- Exceptions
- INotifyDataErrorInfo / IDataErrorInfo
- ValidationRules
- DataAnnotations



Resources

 brian.noyes@solliance.net
 [@briannoyes](https://www.facebook.com/briannoyes)
 <http://briannoyes.net>

- Pluralsight courses:
 - WPF Data Binding in Depth – Brian Noyes
<http://pluralsight.com/training/courses/TableOfContents?courseName=wpf-data-binding-in-depth>
 - XAML Jumpstart
<http://pluralsight.com/training/courses/TableOfContents?courseName=xaml-jumpstart>
- Books
 - WPF Unleashed: <http://www.amazon.com/Windows-Presentation-Foundation-Unleashed-WPF/dp/0672328917>
- Validation blog post:
 - <http://blog.pluralsight.com/async-validation-wpf-prism>



DEV  intersection
europe 2015

<anglebrackets/>
europe 2015

Connect with me

Brian Noyes
brian.noyes@solliance.net
[@briannoyes](https://www.facebook.com/briannoyes)
<http://briannoyes.net>
<http://www.solliance.net>

My Connect

On the page for this session, you'll find

- Ask a question to the speaker(s)
- Slides
- Resources
- Feedback form