Unleashing the Full Power of Bindings

Brian Noyes
CTO, Solliance (www.solliance.net)
brian.noyes@solliance.net, @briannoyes

pluralsight
hardcore developer training

Outline

UpdateSource Trigger

Fallback Values

Converters

FormatString

Async Bindings

MultiBinding PriorityBinding

Binding Mode and Direction

- **Bindings can flow data in two directions**
  - Source to target
  - Target to source
- **Default is set by the DependencyProperty that is the target**
  - FrameworkPropertyMetadata
- **In WPF, primary properties of editable controls are two-way by default**
  - Read-only controls (i.e. Label) and other properties will be one-way by default
- **Set through Mode property on Binding**
  - OneWay
  - TwoWay
  - OneWayToSource
  - OneTime
  - Default

Two-Way Data Binding Triggers

- **By default, modified values in a bound control get pushed to source object on focus change**
  - Tab out of field
- **Can get it immediately on bound property change**
- **Binding.UpdateSourceTrigger**
  - Default
  - LostFocus
  - PropertyChanged
- **Default and LostFocus mean the same thing for most controls**
  - DataGrid
    - Default – Row lost focus
    - LostFocus – Cell lost focus
- **PropertyChanged**
  - Bound target property value has changed

Converters

- **Custom code to sit between source and target objects**
- **Can transform data value flowing in each direction**
  - Change data value
  - Change data type
- **IValueConverter**
  - Convert – from source to target
  - ConvertBack – from target to source
    - Only matters for two-way data binding
- **Create class that implements IValueConverter**
- **Create instance in Resources**
- **Set Converter property on Binding through {StaticResource}**

Async Binding

- **Bindings call property get/set synchronously by default**
  - On UI thread
- **Source objects can raise PropertyChanged on separate thread**
  - WPF only
  - Binding calls get block and updates on UI thread
- **Bindings can call get/set blocks on background thread**
  - IsAsync = true on Binding
- **Bindings can delay calling set block**
  - Delay property on Binding (ms)
  - Used to avoid calling downstream logic too soon for rapidly changing values
    - Typically combined with UpdateSourceTrigger = PropertyChanged

Fallback Values

- **TargetNullValue**
  - Alternate value to use when source property can be found but it's value is null
- **FallbackValue**
  - Used when the source property cannot be found

StringFormat

- **Like an automatic converter from source property to target property**
  - Any type -> ToString()
- **Provide a standard format string in .NET**
  - Examples: d – short date, f2 – float with two decimal places
- **Not recommended in combination with UpdateSourceTrigger=PropertyChanged**
  - Each keystroke in a TextBox causes control to re-render and re-format

MultiBinding / PriorityBinding

- **MultiBinding**
  - Can contain a collection of Bindings
  - Combine with an IMultiValueConverter
  - Come up with a single value from the values of each of the child bindings
- **PriorityBinding**
  - Can contain a collection of Bindings
  - Evaluated in order declared, highest priority first
  - Value produced by binding is highest priority available value
- **Generally don't use either**
  - Too much "logic" in the view - coupling and difficult to debug
  - Control what value is available in a bound source object property behind the scenes, then just bind to that normal

Summary

UpdateSourceTrigger
controls when data flows

Multibinding /
PriorityBinding for advanced

Converters transform
bound value or type

StringFormat lets you
"pretty print" values
into bound controls

Bindings can execute async

Can have fallback values
for bindings