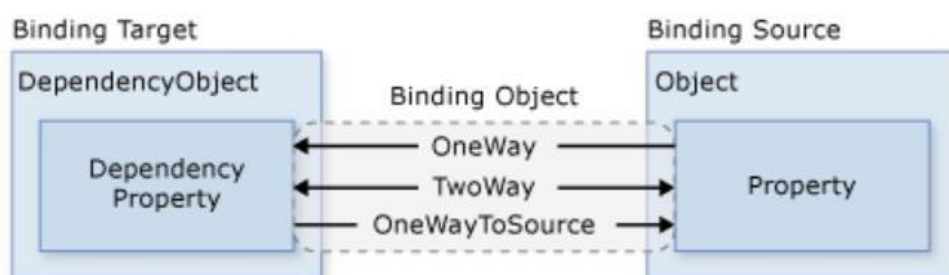


## Tema 1.4.7. Data Binding

Un binding es un “enlace” mediante el cual un control recibe y/o envía información. La información viaja de “origen” a “destino” y/o viceversa.

Concretamente, cuando se realiza un data binding se está creando una **dependencia** entre el valor de una **propiedad** llamada **target** y el valor de otra **propiedad** llamada **source**. Ejemplo: `MiElementoInterfaz.Valor = MiAplicacion.Dato`



Para realizar el data binding se utiliza la **clase Binding**. En ella se establece el tipo de relación entre las propiedades target y source. Se utiliza **desde XAML** mediante:

- Notación **{Binding ...}**
- Notación explícita **<Binding...>...</Binding>**.
- Clase **Binding** (System.Windows.Data).
- **Propiedades** principales de la clase:
  - **ElementName**: obtiene o establece el nombre del elemento que se va a utilizar como objeto de origen de enlace.
  - **Mode**: valor que indica la dirección del flujo de datos en el enlace. Normalmente, la propiedad target recibe el valor de la propiedad source (**OneWay**). Opciones: o **OneTime** o **OneWay** o **OneWayToSource** o **TwoWay**
  - **Path**: ruta de acceso a la propiedad del origen de enlace.
  - **Source**: objeto que se va a utilizar como origen de enlace.

Por lo tanto, el Enlace de Datos le permite vincular las propiedades de dependencia de un control a varias fuentes, lo que elimina la necesidad de crear código de plomería para algunos escenarios. Una de las fuentes de enlace más simples es una propiedad a otra, denominada control WPF.

Para demostrar las uniones necesitamos vamos a realizar un ejemplo:

Crea un nuevo proyecto de aplicación WPF en Visual Studio y nombra al proyecto como “ControlBinding”. Una vez preparado el proyecto, reemplace el XAML de la ventana principal con lo siguiente:

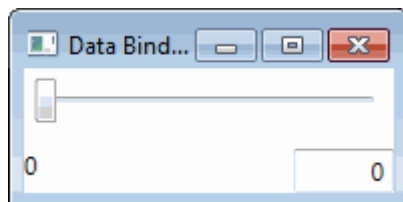
```
Height="100"
Width="200">
<Grid>
  <Grid.RowDefinitions>
    <RowDefinition/>
    <RowDefinition Height="Auto"/>
  </Grid.RowDefinitions>

  <Slider Name="ValueSlider" Margin="5" />

  <TextBlock Grid.Row="1" HorizontalAlignment="Left" Text="0" />

  <TextBox Grid.Row="1"
    HorizontalAlignment="Right"
    Width="50"
    TextAlignment="Right"
    Text="0" />
</Grid>
</Window>
```

La ventana generada por XAML incluye un [Slider](#) , un [TextBlock](#) y un [TextBox](#) . Usaremos el Enlace de Datos para vincular estos controles de modo que el valor en el control deslizante aparezca en los otros dos controles y para que la actualización del control deslizante o del cuadro de texto afecte a los otros controles.



Slider + TextBlock + TextBox

## Agregaremos ahora un Enlace de Datos/Data Binding con “sintaxis de elemento de propiedad”/ Property Element Syntax

Para nuestro primer Enlace de Datos, vincularemos la propiedad “**Valor/Value**” del Control deslizante a la **propiedad Texto** del **Bloque de texto**. Usaremos la *sintaxis del elemento de propiedad* para establecer la propiedad Text de TextBlock en un objeto **Binding** . Este es el objeto que define el enlace, especificando el control de fuente y la ruta a la propiedad a la que deseamos enlazar.

Cuando se vincula a un control específico, la fuente se establece utilizando el atributo **ElementName** del enlace. El valor de este atributo debe coincidir con el nombre del control de fuente. Para especificar qué propiedad vincular, utiliza el

atributo **Ruta** . Hay muchas otras opciones para configurar un enlace, pero para este ejemplo no son necesarias.

Para establecer el enlace de la propiedad Text de TextBlock, reemplaza el XAML del control con lo siguiente:

```
<TextBlock Grid.Row="1" HorizontalAlignment="Left">
    <TextBlock.Text>
        <Binding ElementName="ValueSlider"
            Path="Value"/>
    </TextBlock.Text>
</TextBlock>
```

El XAML actualizado elimina la propiedad Text de la etiqueta XAML principal de TextBlock y la reemplaza con la sintaxis del elemento de propiedad. En lugar de proporcionar texto sin formato en la propiedad Texto, utilizamos un enlace con los detalles de origen adecuados. Ten en cuenta que **ElementName** se establece en el nombre del control de origen y **Path** se establece en **Value**, lo que indica que la propiedad Value del control deslizable debe estar vinculada.

Aunque los tipos de datos de las propiedades enlazadas no coincidan, no es necesario agregar ninguna información adicional para realizar la conversión. WPF puedes convertir automáticamente entre el número de punto flotante contenido en Valor y la cadena en la propiedad Texto.

Ejecuta el programa y mueva el control deslizable para ver los resultados. Deberías ver que el texto del TextBlock se actualiza automáticamente a medida que deslizas el control, mostrando el valor seleccionado.

## Sintaxis de ruta

La parte "Path/Ruta" del enlace en el ejemplo anterior es muy básica. Simplemente nombra la propiedad **fuente**, que es una propiedad de nivel superior del control de fuente. En escenarios más complejos, es posible que desee vincularse a propiedades que se encuentran más profundamente en el modelo de objetos, a [indexadores](#) o a propiedades adjuntas.

Para vincular a propiedades más profundas, puedes especificar una ruta que contenga puntos o puntos. Por ejemplo, si desea vincular la longitud de una propiedad STRING (cadena de caracteres), puedes utilizar el siguiente enlace:

```
<Binding ElementName="MyText"
    Path="Text.Length"/>
```

Si quieres vincular algo una propiedad indexada, puedes especificar el índice entre paréntesis. Para propiedades de indexador con múltiples índices, separe los índices con

comas. Si el indexador utiliza una cadena/string para identificar el elemento deseado, proporcione la cadena entre comillas. Por ejemplo, el siguiente enlace extrae el primer carácter de una propiedad de cadena:

```
<Binding ElementName="MyText"  
        Path="Text[0]" />
```

Si quieres vincular a una propiedad adjunta, especificala entre paréntesis. Por ejemplo, si vas a vincular el número de la fila/Row de un control que aparece en un [Grid](#), puedes usar la siguiente sintaxis:

```
<Binding ElementName="MyText"  
        Path="(Grid.Row)" />
```

## Expresiones vinculantes

La etiqueta Binding puedes simplificarla en muchos enlaces utilizando una expresión de enlace. Las expresiones de enlace se proporcionan a un atributo como una cadena. Incluyen la misma información que los enlaces anteriores, especificada como una serie de pares clave-valor separados por comas. Por ejemplo, nuestro enlace a la propiedad Valor del control deslizante usaría la siguiente expresión:

**{Binding ElementName = ValueSlider, Path = Valor}**

Podemos demostrar esto usando la propiedad Texto del TextBox. Vincularemos esto a la misma fuente que TextBlock, permitiendo que el valor del control deslizante se muestre y actualice automáticamente. Reemplace el XAML del TextBox con lo siguiente:

```
<TextBox Grid.Row="1"  
        HorizontalAlignment="Right"  
        Width="50"  
        TextAlignment="Right"  
        Text="{Binding ElementName=ValueSlider, Path=Value}" />
```

Ejecuta el programa nuevamente para ver los resultados. Debería descubrir que operar el control deslizante hace que se actualice el texto en los controles TextBlock y TextBox. Además, si escribes un valor entre uno y diez en el cuadro de texto, luego presiona la tecla tab para que el control pierda el foco, el control deslizante y el bloque de texto se actualizan automáticamente.