Property Bindings in Vision

Property to Property Binding

A property binding is a simple type of binding. It binds one component’s property to another. When that property changes, the new value is pushed into the property that the binding is set up on. In the following example, we’ll bind the value of a Numeric Text Field to a Slider component.

Why aren’t all properties listed?
You may notice that the list of properties available to bind to is smaller than the list of all properties. While nearly all properties can be bound, only some properties can be bound to. Only properties that fire a `propertyChangeEvent` may be bound to.

For this example, start by placing a Numeric Text Field component and a Slider component on window.

1. Select the Numeric Text Field.
2. Click the Binding icon next to the Value (Integer) property.
3. Select the Property Binding Type. Choose the Slider’s Value property.
4. Click OK. Put the Designer in **Preview Mode**.

5. Move the slider. You'll see that the value from the Slider component appears in the Numeric Text field.

This can be useful to provide visual feedback to what a user is doing. The operator would input something, and they would see another component adjust to match the setting they just changed. Notice though, how if I were to change the value of the Numeric Text Field, the Slider will not update. Bindings are one direction only by default.

### Bidirectional Property Bindings

Property Bindings have the ability to become Bidirectional, meaning instead of having the binding go one way only, it will work both ways, even with just the one binding. Take the previous example with the Numeric Text Field and Slider again. When changing the value of the Slider, the Numeric Text Field would update, but updating the value of the Numeric Text Field would not update the Slider. If we reopen the binding on the Value (Integer) property of the Numeric Text Field, we can see in the bottom left corner a checkbox for **Bidirectional**.
Check the **Bidirectional** option and then save the binding. It will now be a bidirectional binding.