

XAML = XML + {markup extensions}

Razor = HTML + @code

Dictionary<TKey, TValue>

Key	Value
MyBackgroundBrush	LinearGradientBrush

HashSet<TValue>

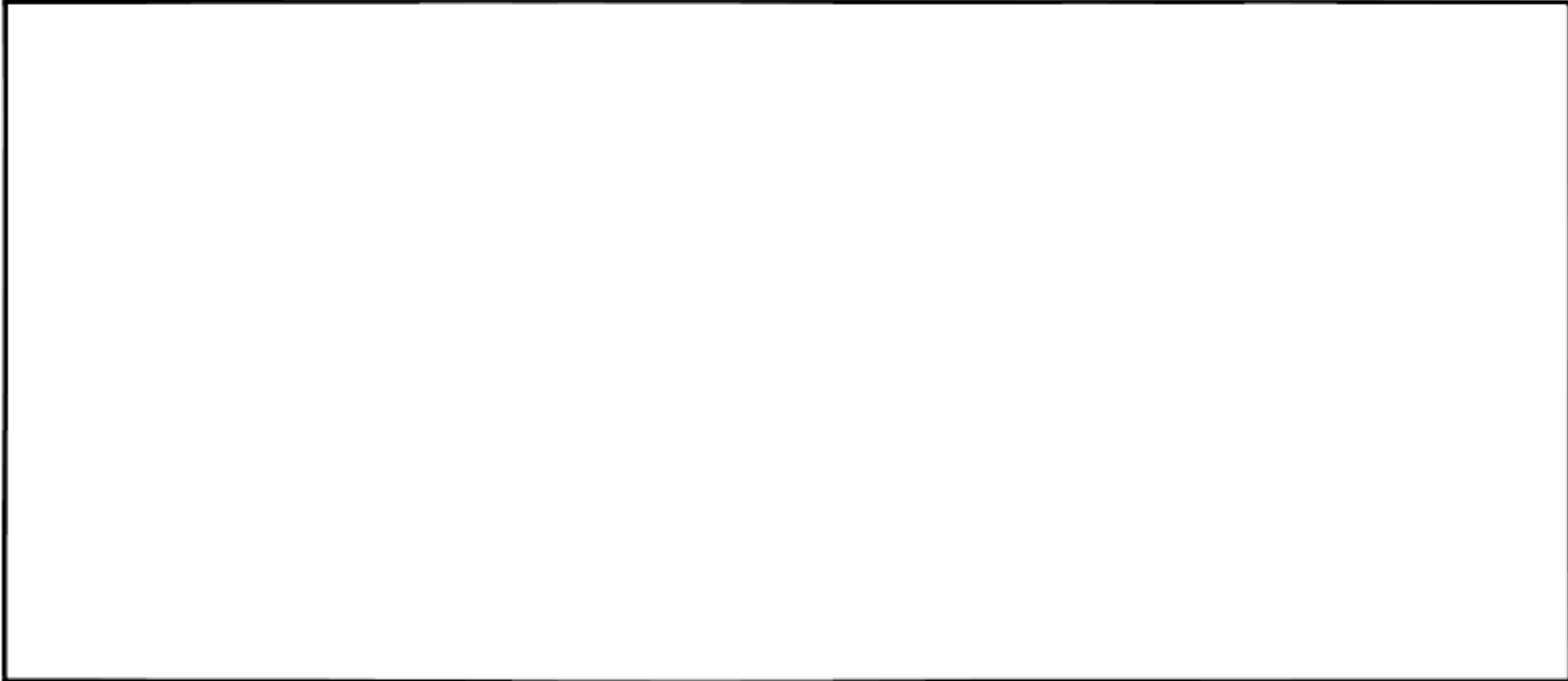
{hash}	Value
	LinearGradientBrush

List<TValue>

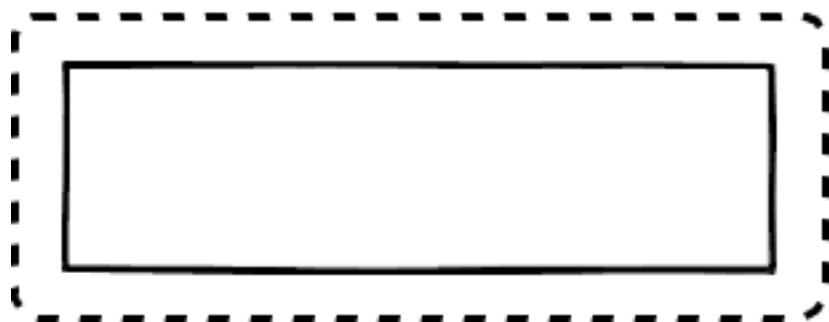
Value
LinearGradientBrush
LinearGradientBrush
LinearGradientBrush

Frame

Page



Margin=5



Padding=5



# Attached Property

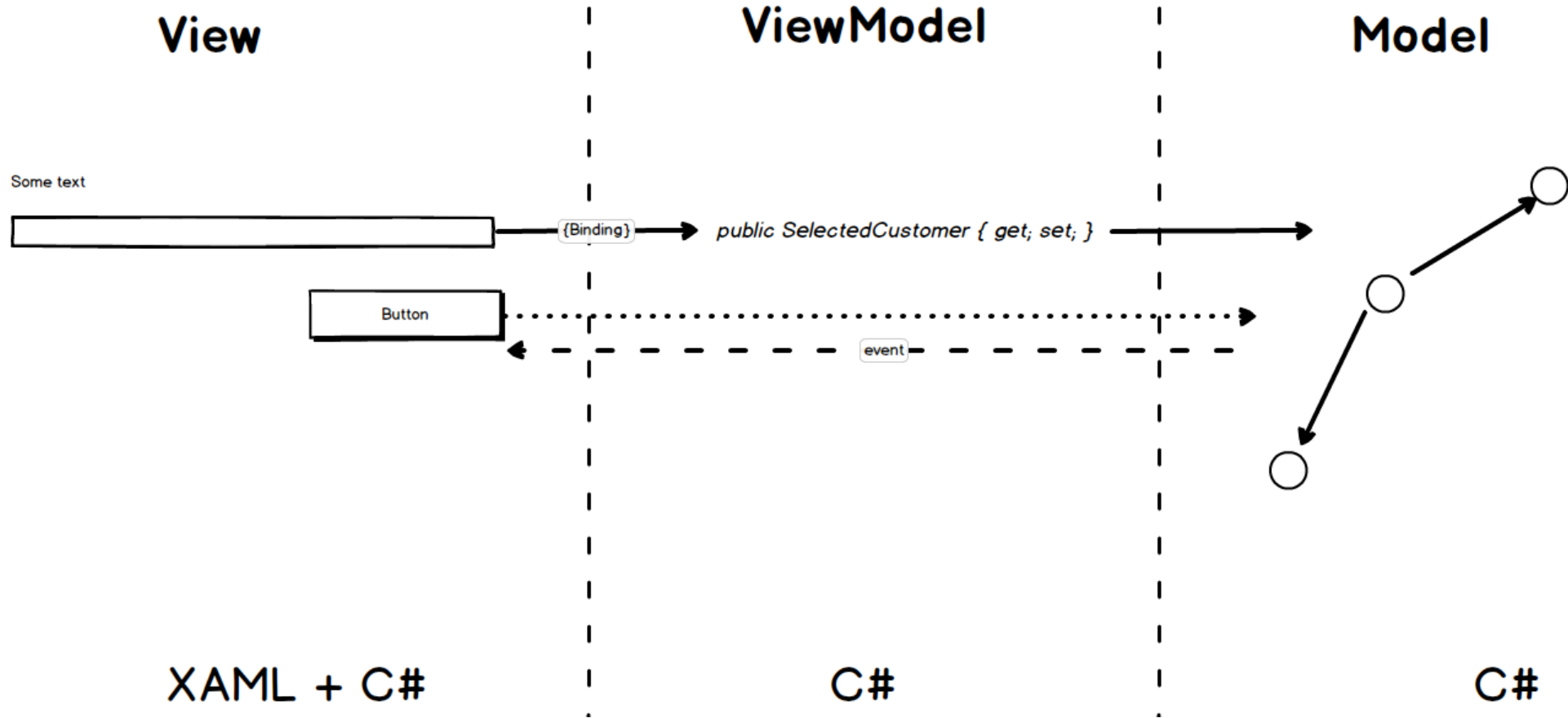
Panel

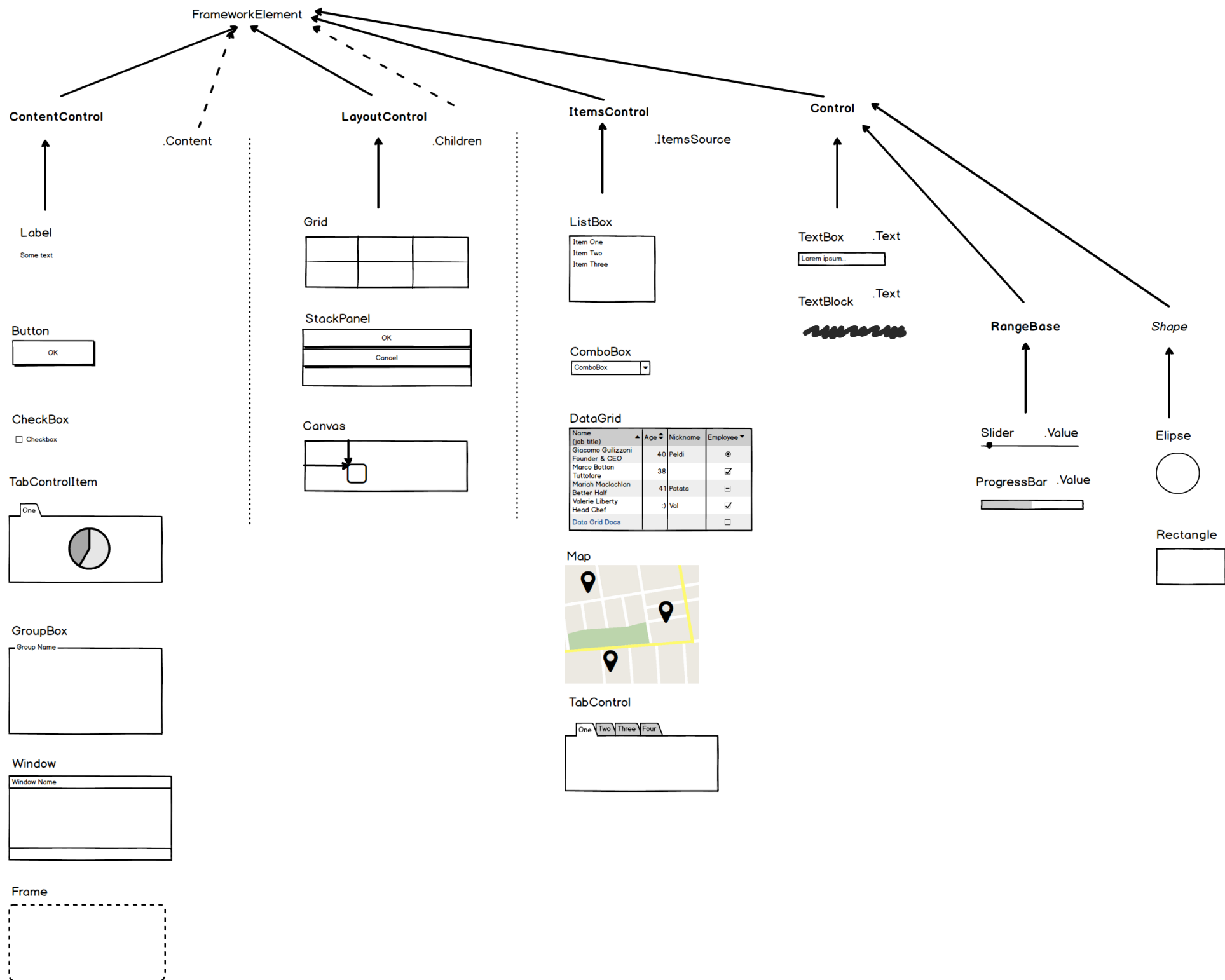


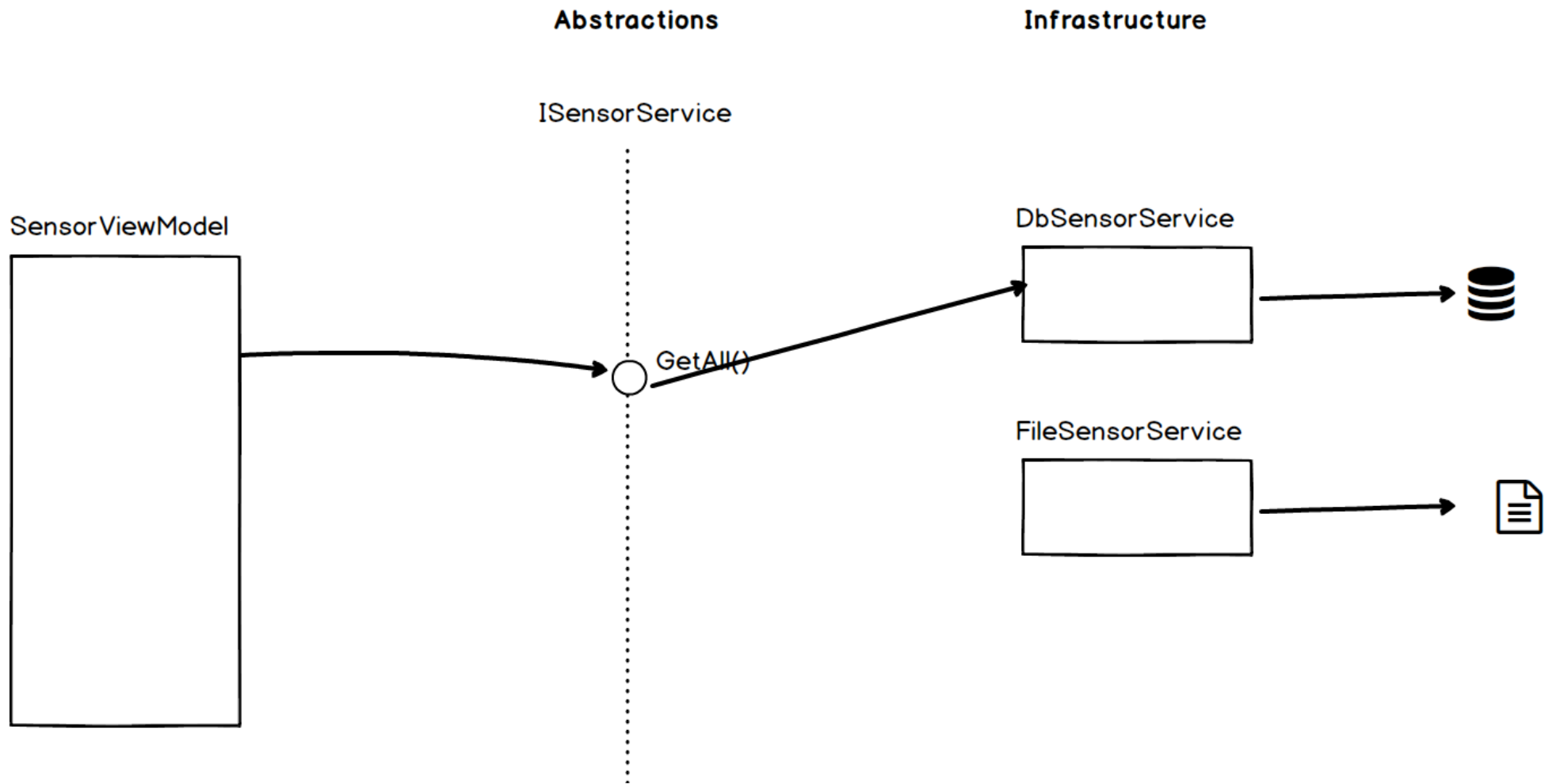
A diagram illustrating an attached property. It consists of a large dashed rectangular box. Inside this box, centered, is a smaller rounded rectangular box with a solid black border. Inside this inner box, the text "Panel.Property" is written.

Panel.Property

# MVVM









Type	TargetType	Cykl zycia
SensorsViewModel	SensorsViewModel	Transient
<i>ISensorService</i>	FakeSensorService	Transient
ShellViewModel	ShellViewModel	Singleton



Some

Some



Some

Some



Some

Some



Some

Some



Some

Some



Some

Some



Some

Some



Some

Some



Some

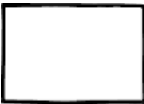
Some



Some

Some

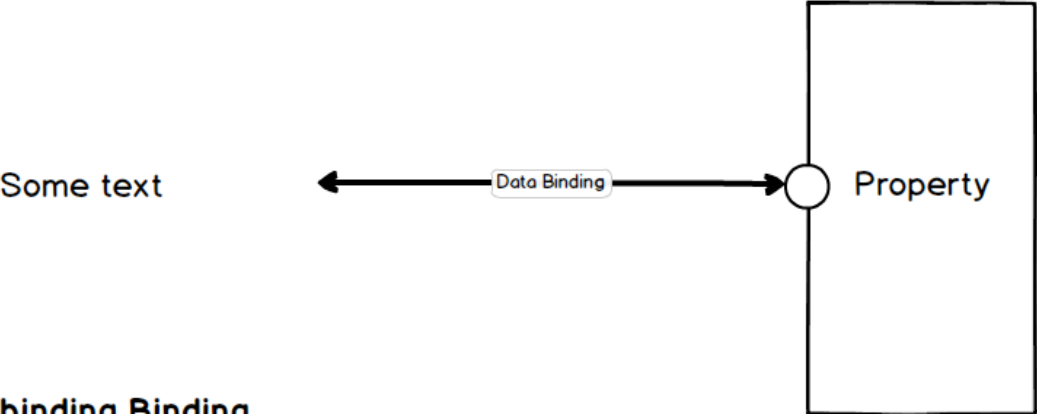




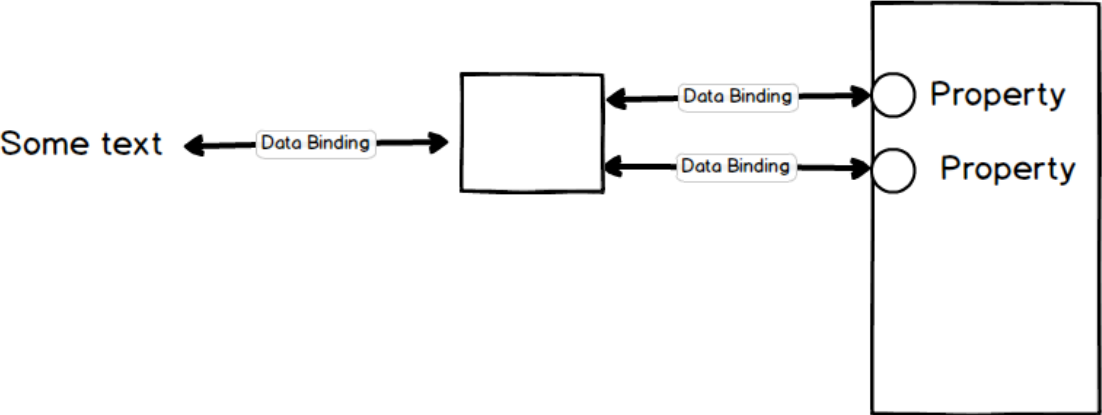
**Element Binding**



**Data Binding**

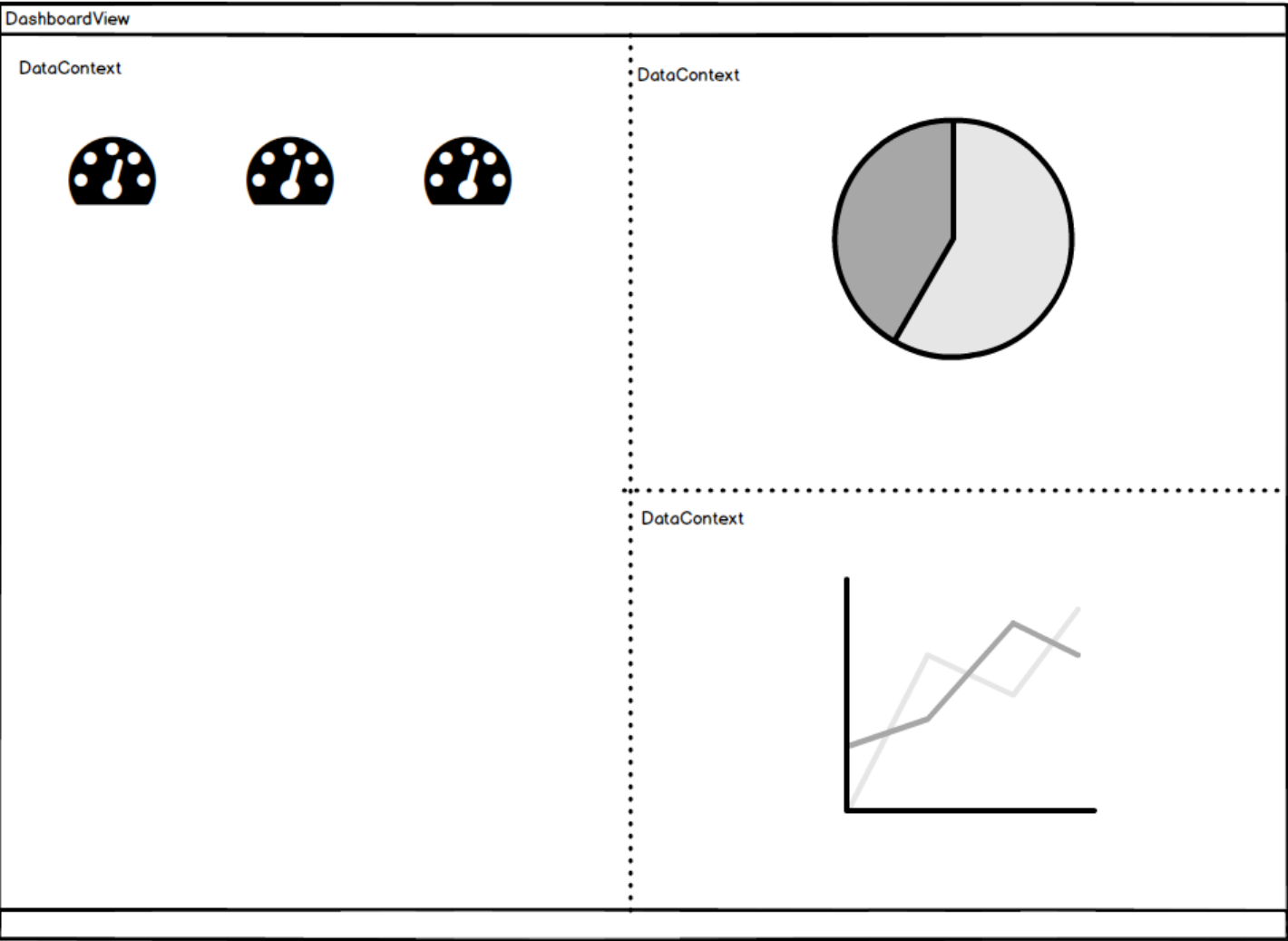


**Multibinding Binding**

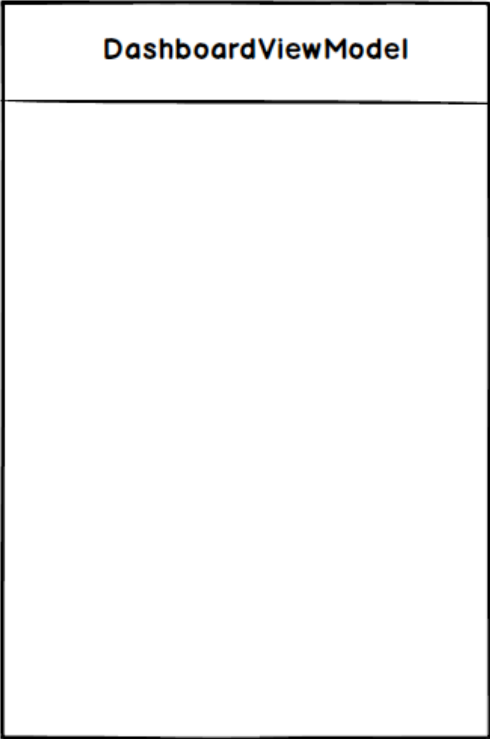


ViewModelA

ViewModelChart

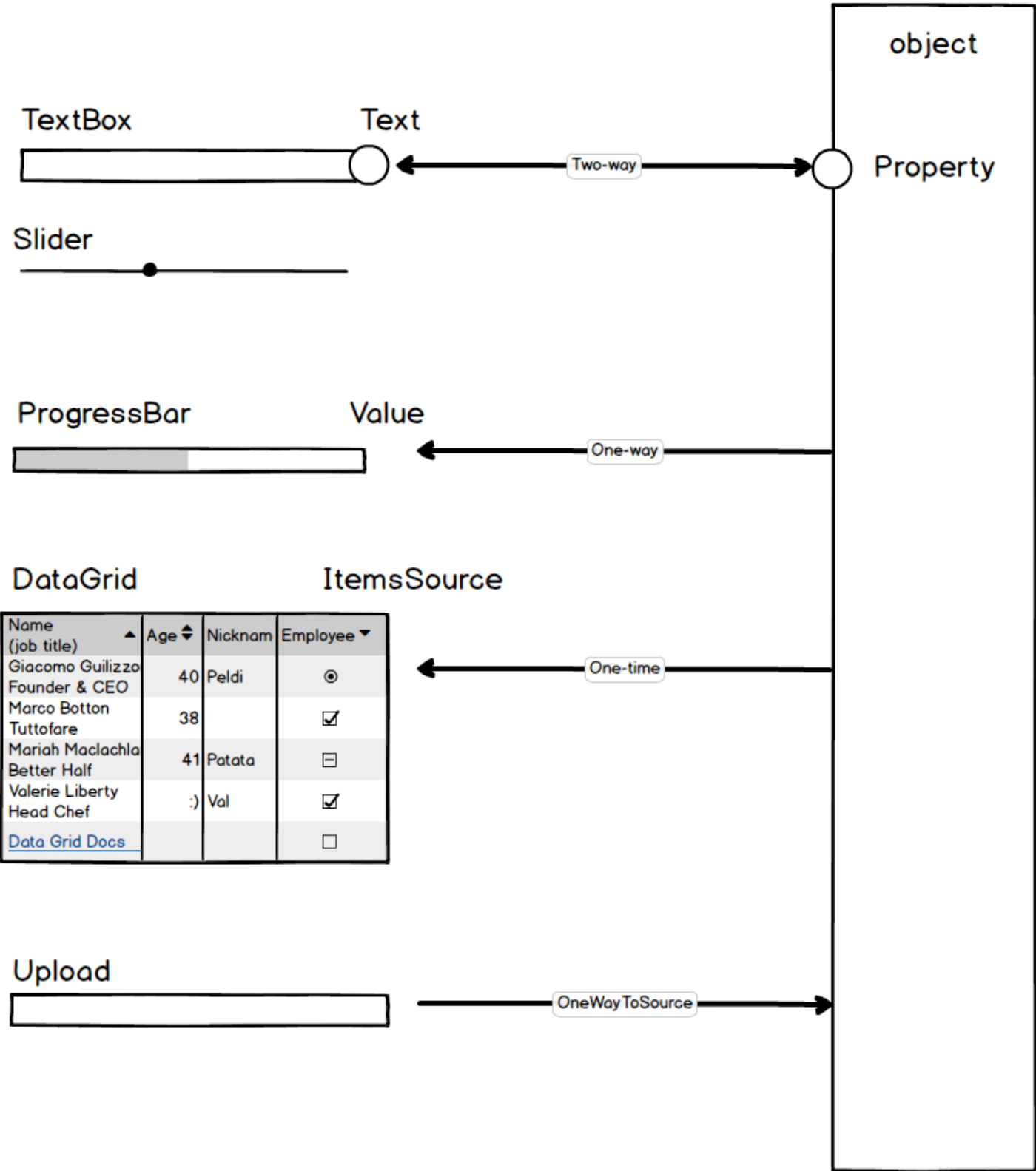


DashboardViewModel





Binding Mode



# Command Pattern

```
class Printer {  
  
    void Print(string content)  
    {  
        ....  
    }
```

```
    void Clean()  
    {  
        ...  
    }
```

```
interface ICommand  
{  
  
    void Execute();  
  
}
```

```
class PrintCommand : ICommand  
{  
  
    public PrintCommand(string content)  
    {  
  
    }  
  
    public void Execute()  
    {  
        ....  
    }
```

```
class CleanCommand : ICommand  
{  
  
    public void Execute()  
    {  
        ....  
    }  
  
}
```

```
Printer printer = new Printer();
```

```
Queue<ICommand> commands = new Queue<ICommand>();  
commands.Add(new PrintCommand("Hello World!"));  
commands.Add(new PrintCommand("Hello World!"));  
commands.Add(new CleanCommand());
```

ContentControl

Content={Binding SelectedViewModel}



## Trigger

Property

Control.IsEnabled

## DataTrigger

Binding

sensor.IsOverLimit

## EventTrigger

EventName

Control.MouseOver

# Partial Class

