

The background of the slide is a dark, moody photograph of a mountain range. A prominent, rugged mountain peak rises in the center, its slopes partially covered in snow. The sky above is filled with heavy, dark clouds, creating a somber and dramatic atmosphere. In the foreground, there's a large expanse of dark, possibly frozen or shadowed ground.

MVU Who?

Mark Allibone
Rey Technology

@mallibone



Hello my name is Mark

 Mark Allibone  

 Rey Technology

 X  @mallibone

 <https://mallibone.com>

 <https://nullpointers.io>





MVU

A large yellow diamond-shaped sign with a thick red border. The letters "MVU" are written in red inside the sign.

@mallibone



.NET MAUI 101



@mallibone



@mallibone

.NET MAUI Architecture

UI Framework

.NET MAUI XAML

Blazor .NET MAUI

MauiReactor

Comet

Framework

.NET MAUI

Platform

iOS

Android

Windows

macOS

Tizen

Linux?



Model View Update - MVU



@mallibone

The background image is a wide-angle aerial photograph of a city at night, likely Dubai, showing a dense cluster of skyscrapers and a complex network of elevated highways and interchanges. The city lights create a glowing, golden-yellow texture against the dark sky.

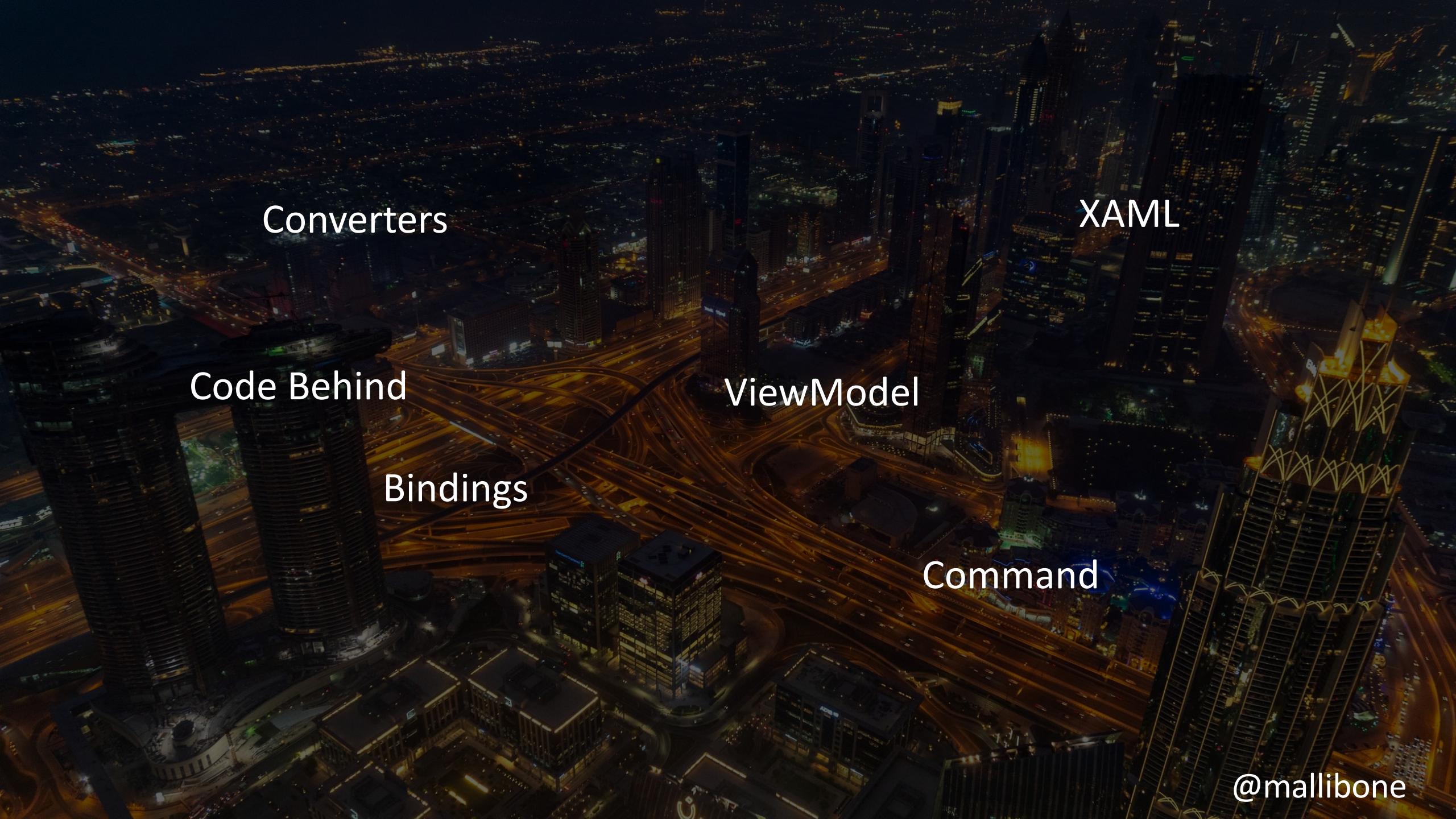
MVU the functional MV* Pattern

@mallibone



Why not just use MVVM?!



The background of the slide is a nighttime aerial photograph of a dense urban landscape, likely Dubai, featuring numerous skyscrapers and a complex network of elevated highways and overpasses. The city lights create a glowing, golden-yellow glow against the dark sky.

Converters

XAML

Code Behind

ViewModel

Bindings

Command



XAML vs Coded UI



CommunityToolkit.Maui.Markup

3.3.0

 Prefix Reserved[.NET 7.0](#)[.NET CLI](#) [Package Manager](#) [PackageReference](#) [Paket CLI](#) [Script & Interactive](#) [Cake](#)

> dotnet add package CommunityToolkit.Maui.Markup --version 3.3.0

Downloads

[Full stats →](#)Total **154.1K**Current version **563**Per day average **200**

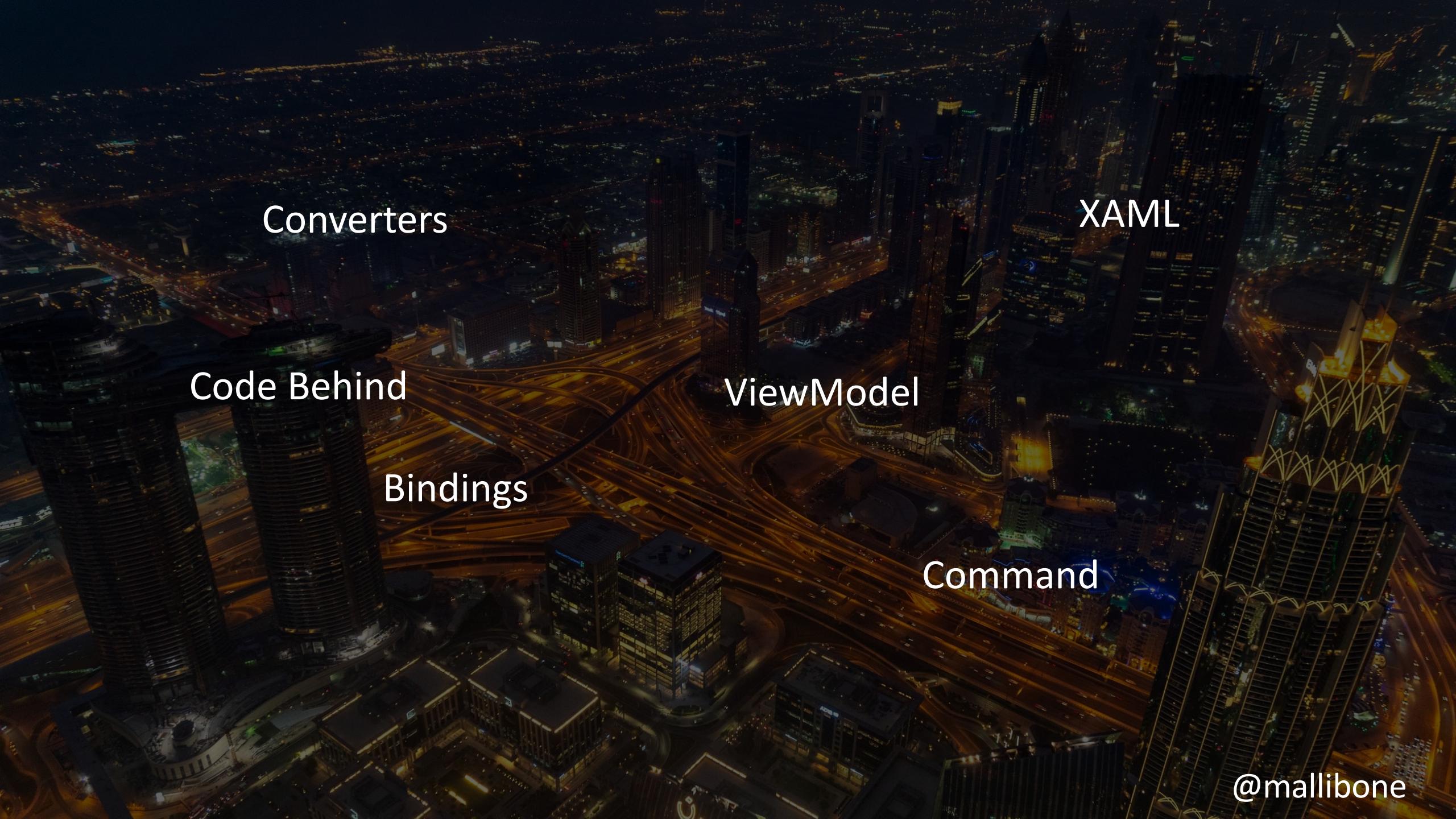
About

 [Last updated 3 days ago](#) [Project website](#) [Source repository](#) [MIT license](#) [Download package \(100.43 KB\)](#) [Download symbols \(15.6 KB\)](#) [Open in NuGet Package Explorer](#) [Open in FuGet Package Explorer](#) [Report package](#)

Owners

[Contact owners →](#)[Microsoft.Toolkit](#)[dotnetfoundation](#)[xamarin](#) [xamarin.forms](#)[net](#) [maui](#) [netmaui](#)[toolkit](#) [kit](#)

@mallibone

The background of the slide is a nighttime aerial photograph of a dense urban landscape, likely Dubai, featuring numerous skyscrapers and a complex network of elevated highways and overpasses. The city lights create a glowing, golden-yellow glow against the dark sky.

Converters

XAML

Code Behind

ViewModel

Bindings

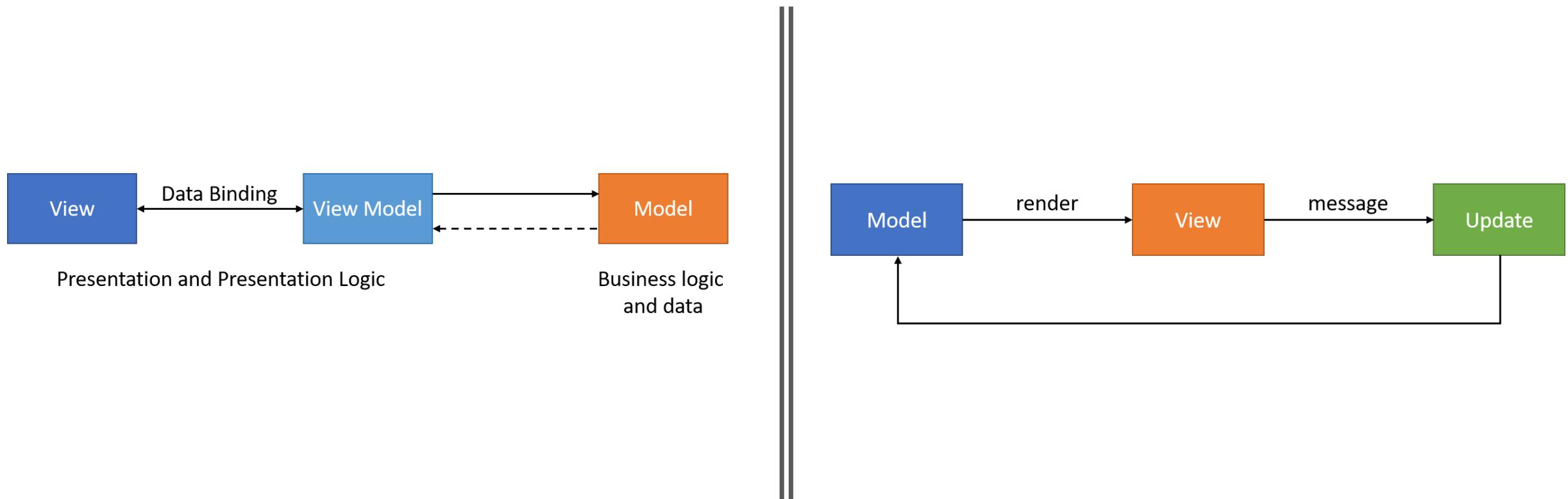
Command



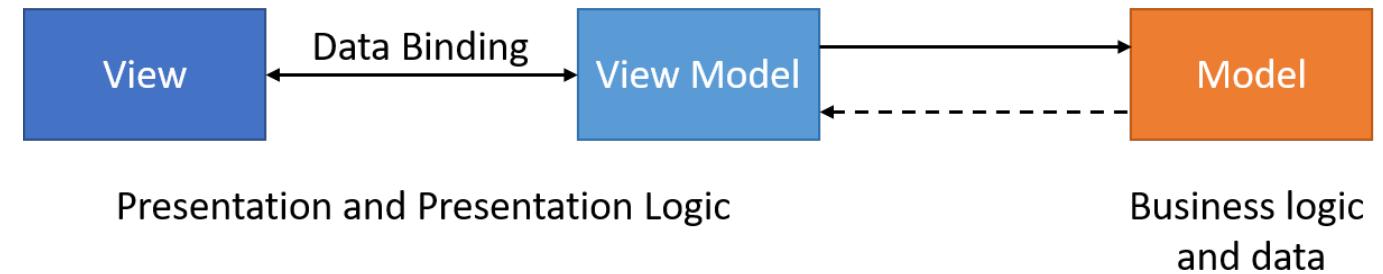
Where did I put that?



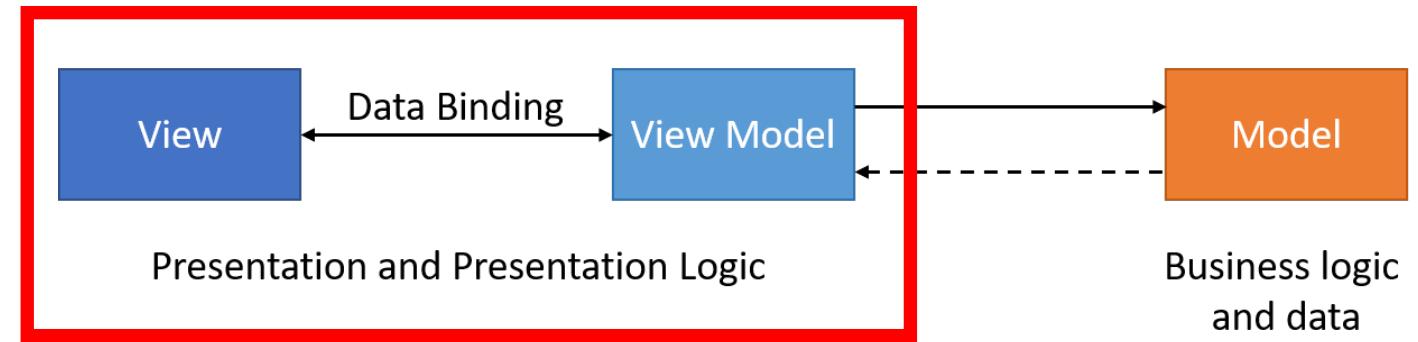
MVVM vs MVU



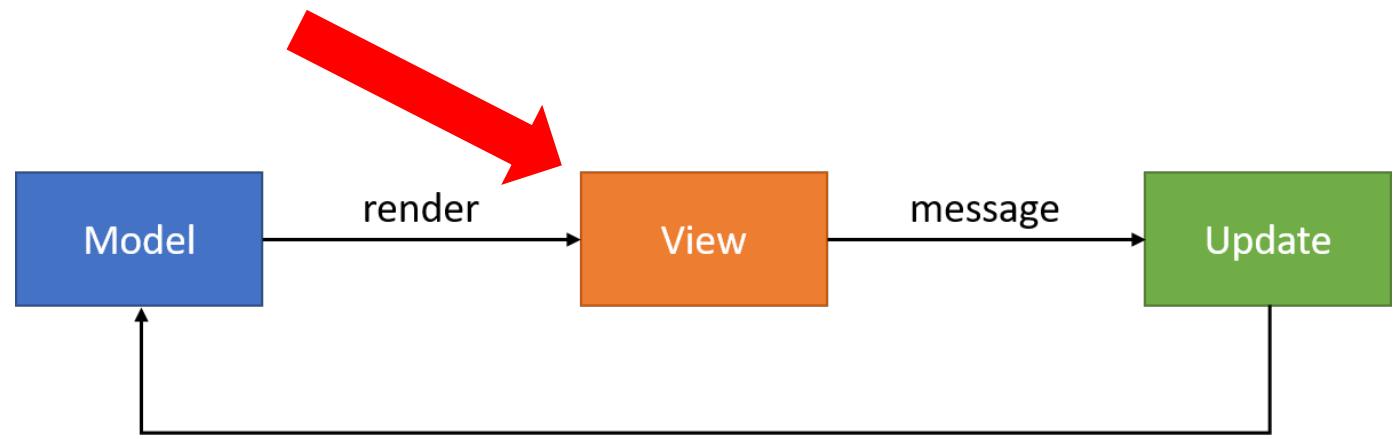
Model View View Model



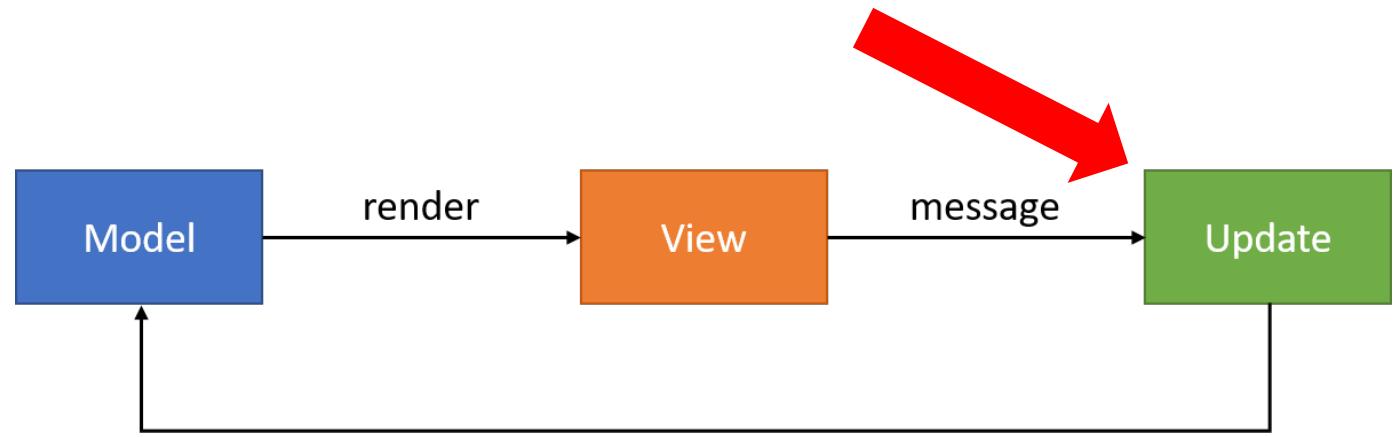
Model View View Model



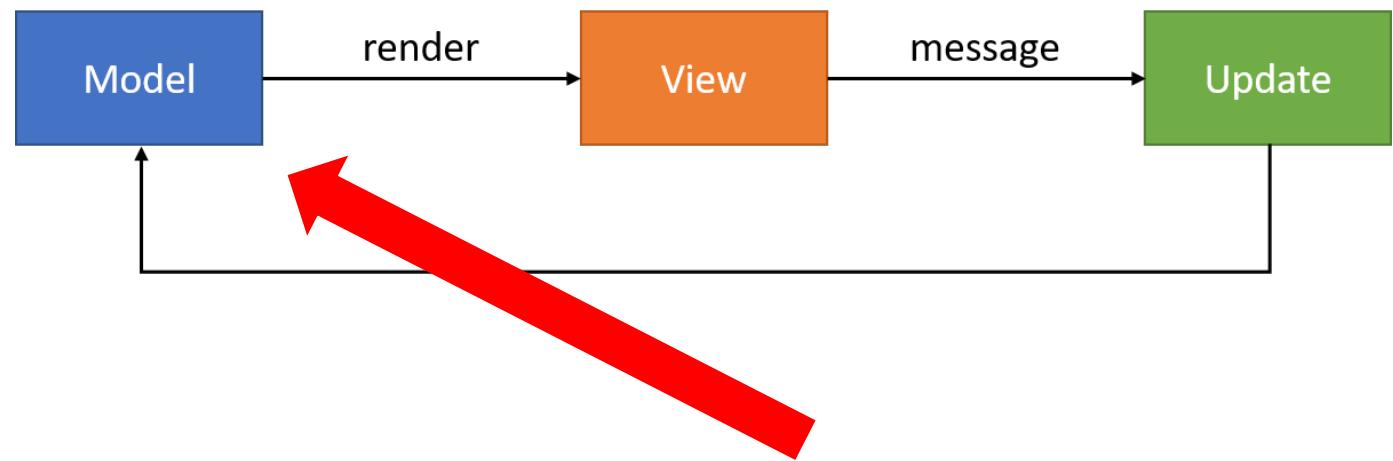
Model View Update



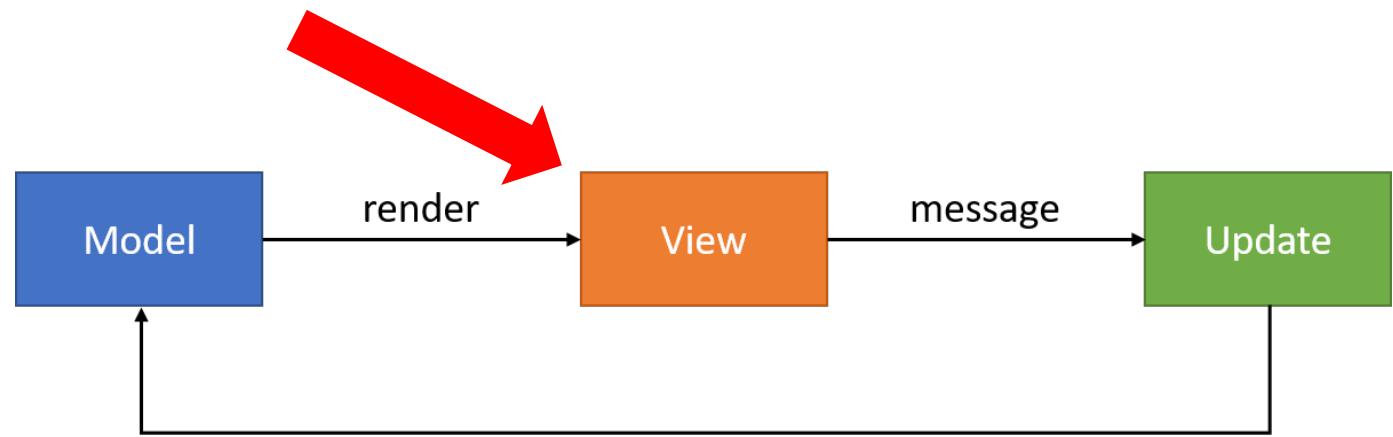
Model View Update



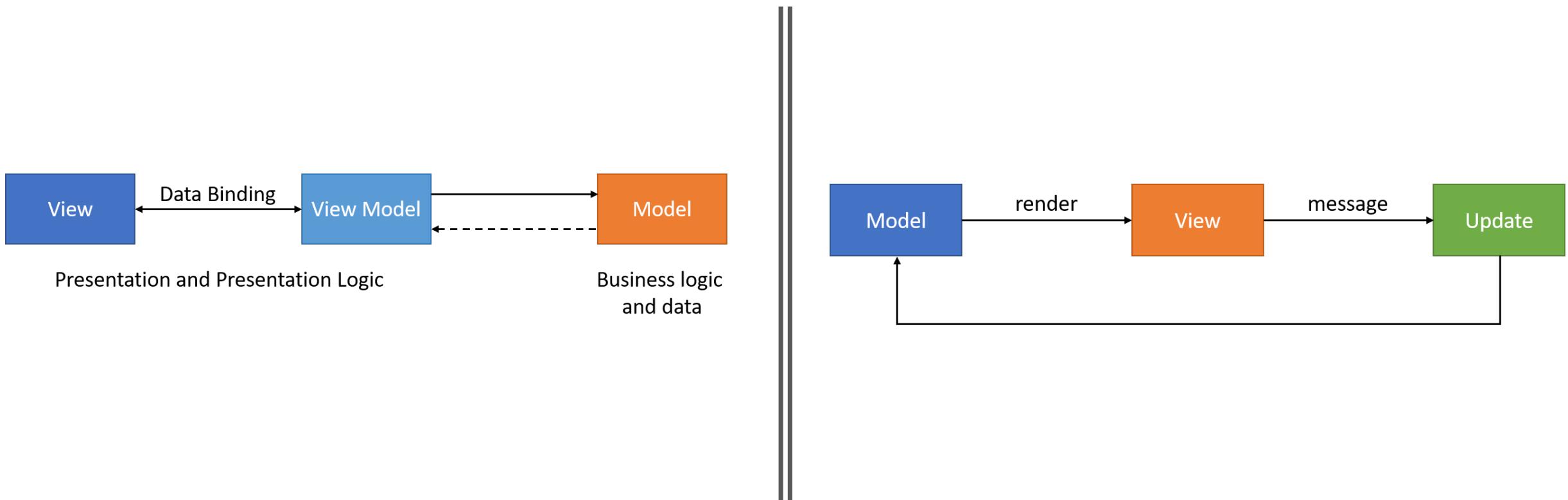
Model View Update



Model View Update



MVVM vs MVU



Visual Studio Installer

Modifying — Visual Studio Enterprise 2022 — 17.3.2

Workloads Individual components Language packs Installation locations

Web & Cloud (4)

- ASP.NET and web development** Build web applications using ASP.NET Core, ASP.NET, HTML/Javascript, and Containers including Docker supp...
- Python development** Editing, debugging, interactive development and source control for Python.
- Azure development** Azure SDKs, tools, and projects for developing cloud apps and creating resources using .NET and .NET Framework...
- Node.js development** Build scalable network applications using Node.js, an asynchronous event-driven JavaScript runtime.

Desktop & Mobile (5)

- .NET Multi-platform App UI development** Build Android, iOS, Windows, and Mac apps from a single codebase using C# with .NET MAUI.
- Desktop development with C++** Build modern C++ apps for Windows using tools of your choice, including MSVC, Clang, CMake, or MSBuild.
- .NET desktop development** Build WPF, Windows Forms, and console applications using C#, Visual Basic, and F# with .NET and .NET Frame...
- Universal Windows Platform development** Create applications for the Universal Windows Platform with C#, VB, or optionally C++.

Installation details

- ▶ **Visual Studio core editor**
- ▶ **ASP.NET and web development**
- ▶ **Data storage and processing**
- ▶ **.NET Multi-platform App UI development**
 - ▼ Included
 - ✓ .NET MAUI
 - ✓ .NET Framework 4.7.2 development tools
 - ✓ C# and Visual Basic
 - ✓ Development tools for .NET
 - ✓ MSBuild
 - ✓ NuGet package manager
 - ▼ Optional
 - ✓ IntelliCode
 - Xamarin
- ▶ **Individual components**

Total space required 0 B

Install while downloading ▾ Close

INSTALLED VERSION 17.3.2 TO 17.4.1457

@mallbone

Visual Studio Installer

Modifying — Visual Studio Enterprise 2022 — 17.3.2

Workloads Individual components Language packs Installation locations

Web & Cloud (4)



ASP.NET and web development

Build web applications using ASP.NET Core, ASP.NET, HTML/JavaScript, and Containers including Docker supp...



Azure development

Azure SDKs, tools, and projects for developing cloud apps and creating resources using .NET and .NET Framework....



Python development

Editing, debugging, interactive development and source control for Python.



Node.js development

Build scalable network applications using Node.js, an asynchronous event-driven JavaScript runtime.



Desktop & Mobile (5)



.NET Multi-platform App UI development

Build Android, iOS, Windows, and Mac apps from a single codebase using C# with .NET MAUI.



.NET desktop development

Build WPF, Windows Forms, and console applications using C#, Visual Basic, and F# with .NET and .NET Frame...



Desktop development with C++

Build modern C++ apps for Windows using tools of your choice, including MSVC, Clang, CMake, or MSBuild.



Universal Windows Platform development

Create applications for the Universal Windows Platform with C#, VB, or optionally C++.



Location

C:\Program Files\Microsoft Visual Studio\2022\Enterprise

By continuing, you agree to the [license](#) for the Visual Studio edition you selected. We also offer the ability to download other software with Visual Studio. This software is licensed separately, as set out in the [3rd Party Notices](#) or in its accompanying license. By continuing, you also agree to those licenses.

Total space required

Install while downloading



Installation details

- ▶ Visual Studio core editor
- ▶ ASP.NET and web development
- ▶ Data storage and processing
- ▶ .NET Multi-platform App UI development
 - ▼ Included
 - ✓ .NET MAUI
 - ✓ .NET Framework 4.7.2 development tools
 - ✓ C# and Visual Basic
 - ✓ Development tools for .NET
 - ✓ MSBuild
 - ✓ NuGet package manager
 - ▼ Optional
 - IntelliCode
 - Xamarin
- ▶ Individual components



@mallibone



Choosing the right Tool

- Comet
- MauiReactor



Choosing the right Framework

- Comet
- MauiReactor



Setting up MauiReactor from CLI

1. Install MauiReactor templates 

```
dotnet new install Reactor.Maui.TemplatePack
```



2. Install MauiReactor hot reload console command 

```
dotnet tool install -g Reactor.Maui.HotReload
```



Setting up MauiReactor from CLI

1. Install MauiReactor templates 

```
dotnet new install Reactor.Maui.TemplatePack
```



2. Install MauiReactor hot reload console command 

```
dotnet tool install -g Reactor.Maui.HotReload
```



Create a new project

Recent project templates

Console App

F#

Console App

C#

MauiReactor

X

Clear all

All languages

All platforms

Blazor



.NET MAUI Blazor App

New

A project for creating a .NET MAUI application for iOS, Android, Mac Catalyst, WinUI, and Tizen using Blazor

C#

Android

Blazor

iOS

Mac Catalyst

macOS

MAUI

Tizen

Windows

Other results based on your search



MauiReactor based app (Adolfo Marinucci)

New

A project for creating a .NET MAUI application based on the MauiReactor library

C#

MAUI



MauiReactor based app with App Shell (Adolfo Marinucci - James Alickolli)

New

A project for creating a .NET MAUI application based on the MauiReactor library

C#

MAUI



.NET MAUI App

New

A project for creating a .NET MAUI application for iOS, Android, Mac Catalyst, WinUI and Tizen

C#

Android

iOS

Mac Catalyst

macOS

MAUI

Tizen

Windows

Next

@mallbone

MainPage.cs — RideTheComet

RideTheComet > MainPage.cs > {} RideTheComet > RideTheComet.MainPage > Body()

```
namespace RideTheComet;
public class MainPage : View
{
    [State]
    private readonly CometRide _comet = new();
    [Body]
    private new View Body()
    {
        => new VStack {
            new Text(<=> $"({_comet.Rides}) rides take
                .Frame(width:300)
                .LineBreakMode(LineBreakMode.CharacterW
            new Button("Ride the Comet! ↴", ()=>{
                _comet.Rides++;
            })
            .Frame(height:44)
            .Margin(8)
            .Color(Colors.White)
            .Background(Colors.Green)
            .RoundedBorder(color:Colors.Blue)
            .Shadow(Colors.Grey,4,2,2),
            new Button("Ride the Comet! ↵", ()=>{
                _comet.Rides+=2;
            })
            .Frame(height:44)
            .Margin(8)
            .Color(Colors.White)
            .Background(Colors.Green)
            .RoundedBorder(color:Colors.Blue)
            .Shadow(Colors.Grey,4,2,2),
        };
    }
    public class CometRide : BindingObject
    {
        public int Rides
        {
            get => GetProperty<int>();
            set => SetProperty(value);
        }
        public string CometTrain => "↗".Repeat(Rides);
    }
}
```

Do you wanna find out?

RideTheComet | net6.0-android | Debug Select a Device 0 ▲ 0 ① 4 Live Share RideTheComet.sln -- NORMAL -- RideTheComet.csproj Ln 25, Col 36 Tab Size: 4 UTF-8 CRLF C# 🔍



Do you wanna find out?

Hello MauiReactor

Templates are available in VS and the CLI

It's just C#

Hot Reload 🔥

The screenshot shows the Microsoft Visual Studio IDE interface. The top menu bar includes File, Edit, View, Git, Project, Build, Debug, Test, Analyze, Tools, Extensions, Window, Help, Search, and Project1. The toolbar contains various icons for file operations like Open, Save, Print, and Build. The main code editor window displays `MainPage.cs` (Project1.net7.0-android) with the following C# code:

```
8  namespace Project1.Pages
9  {
10     1 reference
11     internal class MainPageState
12     {
13         3 references
14         public int Counter { get; set; }
15     }
16     1 reference
17     internal class MainPage : Component<MainPageState>
18     {
19         0 references
20         public override VisualNode Render()
21         {
22             return new ContentPage
23             {
24                 new ScrollView
25                 {
26                     new VerticalStackLayout
27                     {
28                         new Image("dotnet_bot.png")
29                             .HeightRequest(200)
30                             .HCenter()
31                             .Set(Microsoft.Maui.Controls.SemanticProperties.DescriptionProperty, "Cute dot net bot waving hi to you!"),
32
33                         new Label("Hello, World!")
34                             .FontSize(32)
35                             .HCenter(),
36
37                         new Label("Welcome to MauiReactor: MAUI with superpowers!")
38                             .FontSize(18)
39                             .HCenter(),
40
41                         new Button(State.Counter == 0 ? "Click me" : $"Clicked {State.Counter} times!")
42                             .OnClicked(()=>SetState(s => s.Counter ++))
43                             .HCenter()
44                     }
45                     .VCenter()
46                     .Spacing(25)
47                     .Padding(30, 0)
48                 };
49             };
49 }

```

The Solution Explorer on the right shows the project structure for 'Project1' with files like `MainPage.cs`, `Platforms`, `Resources`, `GlobalUsings.cs`, and `MauiProgram.cs`. The Properties and Git Changes tabs are also visible.

The image shows a screenshot of the Visual Studio IDE. The title bar reads "File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help Search Project1". The main window displays the code for "MainPage.cs" under the project "Project1 (net7.0-android)". The code defines an internal class `MainPageState` which contains a public property `Counter`. This class is then used as a component in the `MainPage` class, which returns a `ContentPage` with various UI elements like an image, labels, and a button.

```
8 namespace Project1.Pages
9 {
10    internal class MainPageState
11    {
12        public int Counter { get; set; }
13    }
14}
15 internal class MainPage : Component<MainPageState>
16 {
17     public override VisualNode Render()
18     {
19         return new ContentPage
20         {
21             new ScrollView
22             {
23                 new VerticalStackLayout
24                 {
25                     new Image("dotnet_bot.png")
26                         .HeightRequest(200)
27                         .HCenter()
28                         .Set(Microsoft.Maui.Controls.SemanticProperties.DescriptionProperty, "Cute dot net bot waving hi to you!"),
29
30                     new Label("Hello, World!")
31                         .FontSize(32)
32                         .HCenter(),
33
34                     new Label("Welcome to MauiReactor: MAUI with superpowers!")
35                         .FontSize(18)
36                         .HCenter(),
37
38                     new Button(State.Counter == 0 ? "Click me" : $"Clicked {State.Counter} times!")
39                         .OnClicked(()=>SetState(s => s.Counter ++))
40                         .HCenter()
41                 }
42                 .VCenter()
43                 .Spacing(25)
44             }
45         }
46     }
47 }
```

Model

View

```
9
10    1 reference
11    internal class MainPageState
12    {
13        3 references
14        public int Counter { get; set; }
15    }
16
17    1 reference
18    internal class MainPage : Component<MainPageState>
19    {
20        0 references
21        public override VisualNode Render()
22        {
23            return new ContentPage
24            {
25                new ScrollView
26                {
27                    new VerticalStackLayout
28                    {
29                        new Image("dotnet_bot.png")
30                            .HeightRequest(200)
31                            .HCenter()
32                            .Set(Microsoft.Maui.Controls.SemanticProperties.DescriptionProperty, "Cute dot net bot waving hi to you!"),
33
34                        new Label("Hello, World!")
35                            .FontSize(32)
36                            .HCenter(),
37
38                        new Label("Welcome to MauiReactor: MAUI with superpowers!")
39                            .FontSize(18)
40                            .HCenter(),
41
42                        new Button(State.Counter == 0 ? "Click me" : $"Clicked {State.Counter} times!")
43                            .OnClicked(()=>SetState(s => s.Counter ++))
44                            .HCenter()
45                    }
46                    .VCenter()
47                    .Spacing(25)
48                    .Padding(30, 0)
49                };
            };
        }
    };
}
```

Update

```
 9
10 1 reference
11 internal class MainPageState
12 {
13     3 references
14     public int Counter { get; set; }
15 }
16
17 1 reference
18 internal class MainPage : Component<MainPageState>
19 {
20     0 references
21     public override VisualNode Render()
22     {
23         return new ContentPage
24         {
25             new ScrollView
26             {
27                 new VerticalStackLayout
28                 {
29                     new Image("dotnet_bot.png")
30                         .HeightRequest(200)
31                         .HCenter()
32                         .Set(Microsoft.Maui.Controls.SemanticProperties.DescriptionProperty, "Cute dot net bot waving hi to you!"),
33
34                     new Label("Hello, World!")
35                         .FontSize(32)
36                         .HCenter(),
37
38                     new Label("Welcome to MauiReactor: MAUI with superpowers!")
39                         .FontSize(18)
40                         .HCenter(),
41
42                     new Button {Content = "Click me", Clicked = "Clicked {State.Counter} times!"}
43                         .OnClicked(()=>SetState(s => s.Counter ++))
44                         .VCenter()
45
46                 };
47             };
48         };
49     }
```

An aerial night photograph of a dense urban landscape, likely Dubai, featuring a complex network of elevated highways and numerous skyscrapers illuminated against a dark sky.

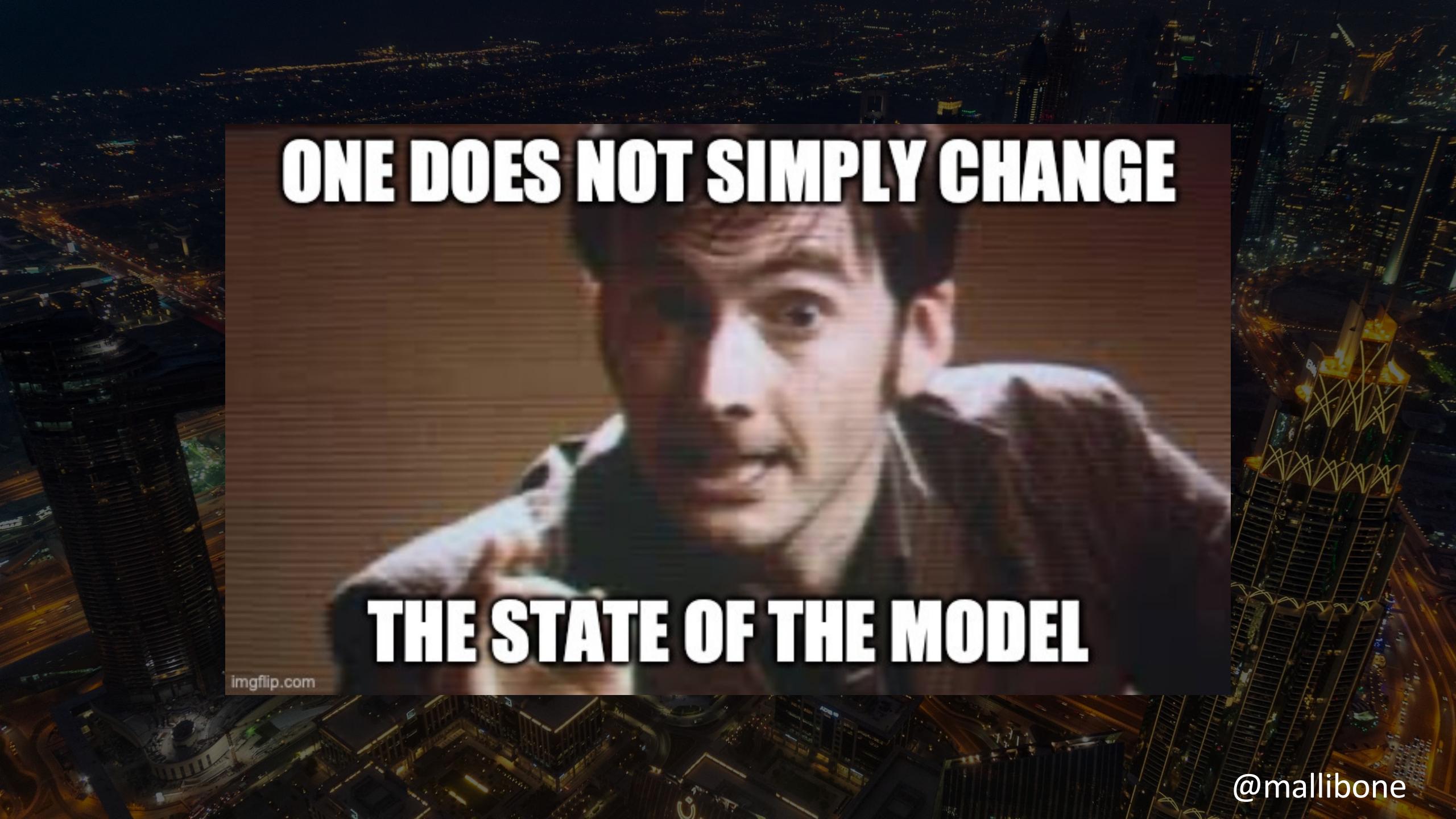
MVU

@mallibone

An aerial night photograph of a dense urban landscape, likely Dubai, featuring a complex network of elevated highways and numerous skyscrapers illuminated against a dark sky.

MVUish

@mallibone



ONE DOES NOT SIMPLY CHANGE

THE STATE OF THE MODEL

imgflip.com

@mallibone



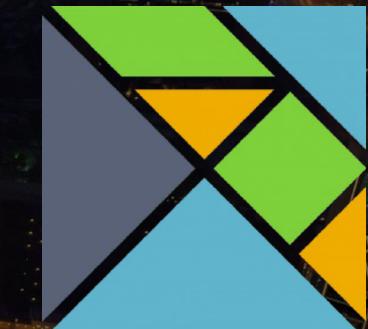
To the origins
of MVU

@mallibone

This Photo by Unknown Author is licensed under CC BY-SA

Elm

Functional Web Frontend



@mallibone



Functional Programming

- No Nulls
- Declarative
- Immutable
- Composition/Monads/Functors...



F# Code Ahead



Fabulous



@mallibone

HelloFabulousWorld.fs — 00_Demo1

EXPLORER OPEN EDITORS

00_DEMO1 HelloFabulousWorld.fs HelloFabulousWorld.fsproj HelloFabulousWorld.Android HelloFabulousWorld.iOS .gitignore HelloFabulousWorld.sln

```
20 | let step of int
21 |   | TimerToggled of bool
22 |   | TimedTick
23 |
24 | let initModel = { Count = 0; Step = 1; TimerOn=false }
25 |
26 | let init () = initModel, Cmd.none
27 |
28 | let timerCmd =
29 |   | async { do! Async.Sleep 200
30 |           | return TimedTick }
31 |   |> Cmd.ofAsyncMsg
32 |
33 | let update msg model =
34 |   match msg with
35 |   | Increment -> { model with Count = model.Count + model.Step }, Cmd.none
36 |   | Decrement -> { model with Count = model.Count - model.Step }, Cmd.none
37 |   | Reset -> init ()
38 |   | SetStep n -> { model with Step = n }, Cmd.none
39 |   | TimerToggled on -> { model with TimerOn = on }, (if on then timerCmd else Cmd.none)
40 |   | TimedTick ->
41 |     if model.TimerOn then
42 |       { model with Count = model.Count + model.Step }, timerCmd
43 |     else
44 |       model, Cmd.none
45 |
46 | let view (model: Model) dispatch =
47 |   View.ContentPage(
48 |     content = View.StackLayout(padding = Thickness 20.0, verticalOptions = LayoutOptions.Center,
49 |       children = [
50 |         View.Label(text = sprintf "%d" model.Count, horizontalOptions = LayoutOptions.Center, width=200.0, horizontalTextAlignment=TextAlignment.Center)
51 |         View.Button(text = "Increment", command = (fun () -> dispatch Increment), horizontalOptions = LayoutOptions.Center)
52 |         View.Button(text = "Decrement", command = (fun () -> dispatch Decrement), horizontalOptions = LayoutOptions.Center)
53 |         View.Label(text = "Timer", horizontalOptions = LayoutOptions.Center)
54 |       ]
55 |     )
56 |   )
```

TERMINAL JUPYTER PROBLEMS OUTPUT DEBUG CONSOLE AZURE

Comet for .NET Mobile

postaiSearch* Mobile .NET Projects... Live Share -- NORMAL -- HelloFabulousWorld.fsproj Project loading failed Ln 46, Col 38 Spaces: 4 UTF-8 with BOM CRLF F# @mallibone

@mallibone

View

```
20 let init () = initModel, Cmd.none
19
18 let timerCmd =
17   async { do! Async.Sleep 200
16     return TimedTick }
15   ▷ Cmd.ofAsyncMsg
14
13 let update msg model =
12   match msg with
11   | Increment → { model with Count = model.Count + model.Step }, Cmd.none
10   | Decrement → { model with Count = model.Count - model.Step }, Cmd.none
9   | Reset → init ()
8   | SetStep n → { model with Step = n }, Cmd.none
7   | TimerToggled on → { model with TimerOn = on }, (if on then timerCmd else Cmd.none)
6   | TimedTick →
5     if model.TimerOn then
4       { model with Count = model.Count + model.Step }, timerCmd
3     else
2       model, Cmd.none
1
46 let view (model: Model) dispatch =
1   View.ContentPage(
2     content = View.StackLayout(padding = Thickness 20.0, verticalOptions = LayoutOptions.Center,
3     children = [
4       View.Label(text = sprintf "%d" model.Count, horizontalOptions = LayoutOptions.Center, width=200.0, horizontalTextAlignment=TextAlignment.Center)
5       View.Button(text = "Increment", command = (fun () → dispatch Increment), horizontalOptions = LayoutOptions.Center)
6       View.Button(text = "Decrement", command = (fun () → dispatch Decrement), horizontalOptions = LayoutOptions.Center)
7 ])
```

TERMINAL

JUPYTER

PROBLEMS

OUTPUT

DEBUG CONSOLE

AZURE

Comet

Update

```
20 let init () = initModel, Cmd.none
19
18 let timerCmd =
17   |> Async { do! Async.Sleep 200
16     |> return TimedTick }
15   ▷ Cmd.ofAsyncMsg
14
13 let update msg model =
12   match msg with
11   | Increment → { model with Count = model.Count + model.Step }, Cmd.none
10   | Decrement → { model with Count = model.Count - model.Step }, Cmd.none
9   | Reset → init ()
8   | SetStep n → { model with Step = n }, Cmd.none
7   | TimerToggled on → { model with TimerOn = on }, (if on then timerCmd else Cmd.none)
6   | TimedTick →
5     if model.TimerOn then
4       { model with Count = model.Count + model.Step }, timerCmd
3     else
2       model, Cmd.none
1
46 let view (model: Model) dispatch =
1   View.ContentPage(
2     content = View.StackLayout(padding = Thickness 20.0, verticalOptions = LayoutOptions.Center,
3     children = [
4       View.Label(text = sprintf "%d" model.Count, horizontalOptions = LayoutOptions.Center, width=200.0, horizontalTextAlignment=TextAlignment.Center)
5       View.Button(text = "Increment", command = (fun () → dispatch Increment), horizontalOptions = LayoutOptions.Center)
6       View.Button(text = "Decrement", command = (fun () → dispatch Decrement), horizontalOptions = LayoutOptions.Center)
7       View.Label(text = "Timer", horizontalOptions = LayoutOptions.Center)
```

```
20 let init () = initModel, Cmd.none
19
18 let timerCmd =
17   async { do! Async.Sleep 200
16     return TimedTick }
15 ▷ Cmd.ofAsyncMsg
14
13 let update msg model =
12   match msg with
11   | Increment → { model with Count = model.Count + model.Step }, Cmd.none
10   | Decrement → { model with Count = model.Count - model.Step }, Cmd.none
9   | Reset → init ()
8   | SetStep n → { model with Step = n }, Cmd.none
7   | TimerToggled on → { model with TimerOn = on }, (if on then timerCmd else Cmd.none)
6   | TimedTick →
5     if model.TimerOn then
4       { model with Count = model.Count + model.Step }, timerCmd
3     else
2       model, Cmd.none
1
46 let view (model: Model) dispatch =
1   View.ContentPage(
2     content = View.StackLayout(padding = Thickness 20.0, verticalOptions = LayoutOptions.Center,
3       children = [
4         View.Label(text = sprintf "%d" model.Count, horizontalOptions = LayoutOptions.Center, width=200.0, horizontalTextAlignment=TextAlignment.Center)
5         View.Button(text = "Increment", command = (fun () → dispatch Increment), horizontalOptions = LayoutOptions.Center)
6         View.Button(text = "Decrement", command = (fun () → dispatch Decrement), horizontalOptions = LayoutOptions.Center)
7         View.Label(text = "Timer", horizontalOptions = LayoutOptions.Center)
```

Increment

```
20 let init () = initModel, Cmd.none
19
18 let timerCmd =
17   async { do! Async.Sleep 200
16     return TimedTick }
15 ▷ Cmd.ofAsyncMsg
14
13 let update msg model =
12   match msg with
11   | Increment → { model with Count = model.Count + model.Step }, Cmd.none
10   | Decrement → { model with Count = model.Count - model.Step }, Cmd.none
9   | Reset → init ()
8   | SetStep n → { model with Step = n }, Cmd.none
7   | TimerToggle on → { model with TimerOn = on }, (if on then timerCmd else Cmd.none)
6   | TimedTick
5     if model.TimerOn then
4       { model with Count = model.Count + model.Step }, timerCmd
3     else
2       model, Cmd.none
1
46 let view (model: Model) dispatch =
1   View.ContentPage(
2     content = View.StackLayout(padding = Thickness 20.0, verticalOptions = LayoutOptions.Center,
3     children =
4       View.Label(text = sprintf "%d" model.Count, horizontalOptions = LayoutOptions.Center, width=200.0, horizontalTextAlignment=TextAlignment.Center)
5       View.Button(text = "Increment", command = (fun () → dispatch Increment), horizontalOptions = LayoutOptions.Center)
6       View.Button(text = "Decrement", command = (fun () → dispatch Decrement), horizontalOptions = LayoutOptions.Center)
7       View.Label(text = "Timer", horizontalOptions = LayoutOptions.Center)
```

Increment

```
20 let init () = initModel, Cmd.none
19
18 let timerCmd =
17   async { do! Async.Sleep 200
16     return TimedTick }
15 ▷ Cmd.ofAsyncMsg
14
13 let update msg model =
12   match msg with
11   | Increment → { model with Count = model.Count + model.Step }, Cmd.none
10   | Decrement → { model with Count = model.Count - model.Step }, Cmd.none
9   | Reset → init ()
8   | SetStep n → { model with Step = n }, Cmd.none
7   | TimerToggled on → { model with TimerOn = on }, (if on then timerCmd else Cmd.none)
6   | TimedTick →
5     if model.TimerOn then
4       { model with Count = model.Count + model.Step }, timerCmd
3     else
2       model, Cmd.none
1
46 let view (model: Model) dispatch =
1   View.ContentPage(
2     content = View.StackLayout(padding = Thickness 20.0, verticalOptions = LayoutOptions.Center,
3     children =
4       View.Label(text = sprintf "%d" model.Count, horizontalOptions = LayoutOptions.Center, width=200.0, horizontalTextAlignment=TextAlignment.Center)
5       View.Button(text = "Increment", command = (fun () → dispatch Increment), horizontalOptions = LayoutOptions.Center)
6       View.Button(text = "Decrement", command = (fun () → dispatch Decrement), horizontalOptions = LayoutOptions.Center)
7       View.Label(text = "Timer", horizontalOptions = LayoutOptions.Center)
```

Increment

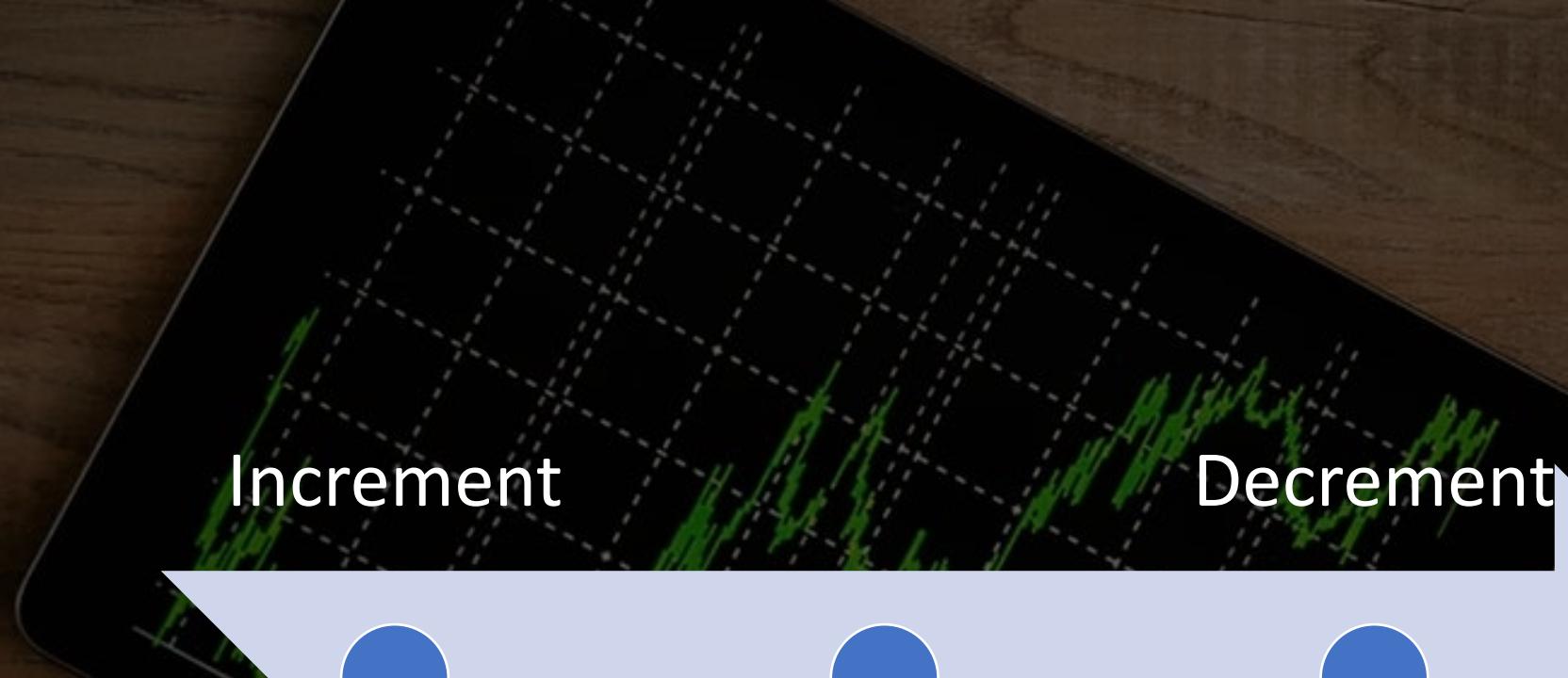
```
20 let init () = initModel, Cmd.none
19
18 let timerCmd =
17   async { do! Async.Sleep 200
16     return TimedTick }
15 ▷ Cmd.ofAsyncMsg
14
13 let update msg model =
12   match msg with
11   | Increment → { model with Count = model.Count + model.Step }, Cmd.none
10   | Decrement → { model with Count = model.Count - model.Step }, Cmd.none
9   | Reset → init ()
8   | SetStep n → { model with Step = n }, Cmd.none
7   | TimerToggled on → { model with TimerOn = on }, (if on then timerCmd else Cmd.none)
6   | TimedTick →
5     if model.TimerOn then
4       { model with Count = model.Count + model.Step }, timerCmd
3     else
2       model, Cmd.none
1
46 let view (model: Model) dispatch =
1   View.ContentPage(
2     content = View.StackLayout(padding = Thickness 20.0, verticalOptions = LayoutOptions.Center,
3     children = [
4       View.Label(text = sprintf "%d" model.Count, horizontalOptions = LayoutOptions.Center, width=200.0, horizontalTextAlignment=TextAlignment.Center)
5       View.Button(text = "Increment", command = (fun () → dispatch Increment), horizontalOptions = LayoutOptions.Center)
6       View.Button(text = "Decrement", command = (fun () → dispatch Decrement), horizontalOptions = LayoutOptions.Center)
7       View.Label(text = "Timer", horizontalOptions = LayoutOptions.Center)
```

Increment

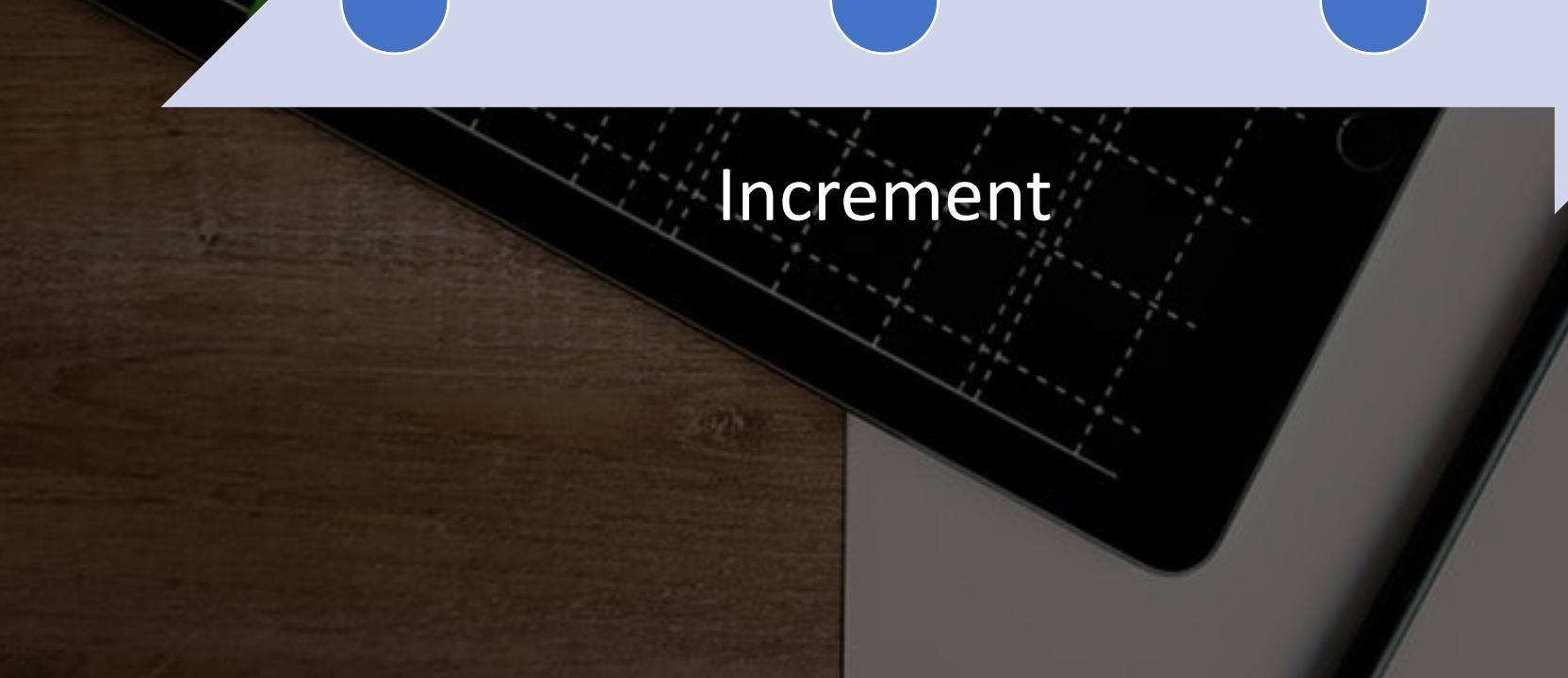
An aerial night photograph of a dense urban landscape, likely Dubai, featuring a complex network of elevated highways and numerous skyscrapers illuminated against a dark sky.

Who cares?!

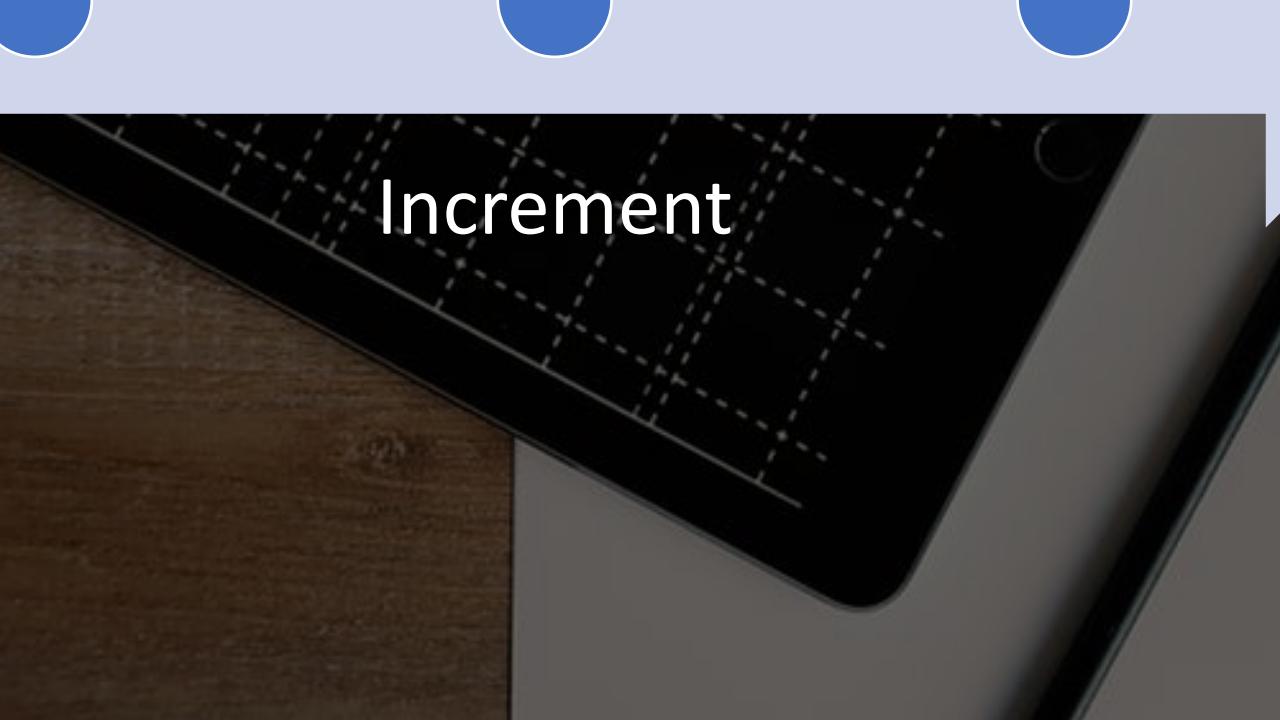
@mallibone



Increment

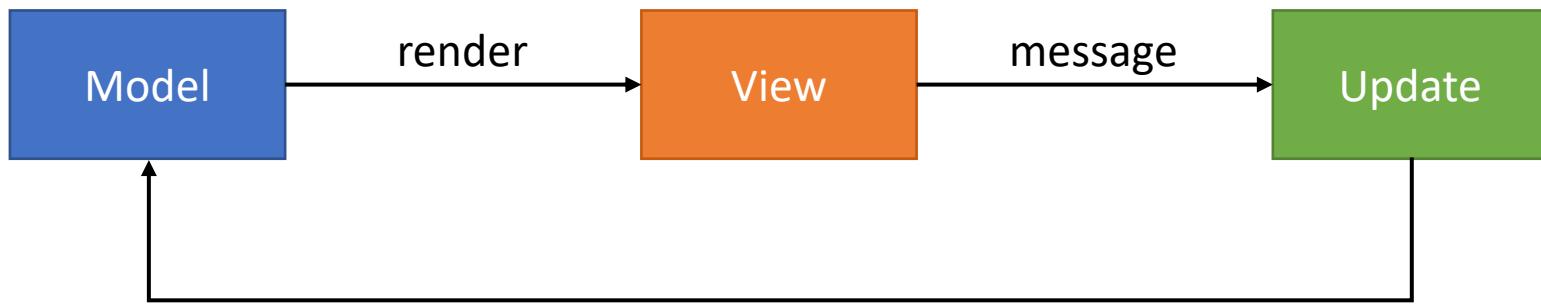


Decrement

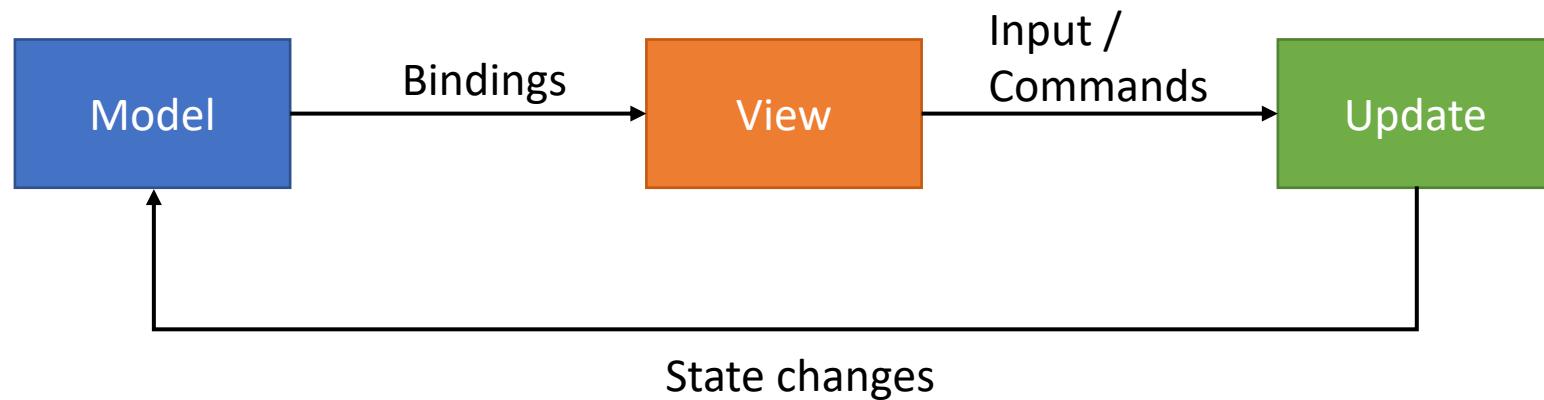


Increment

Model View Update



Model View Update



MauiReactor

AND THIS CONCLUDES



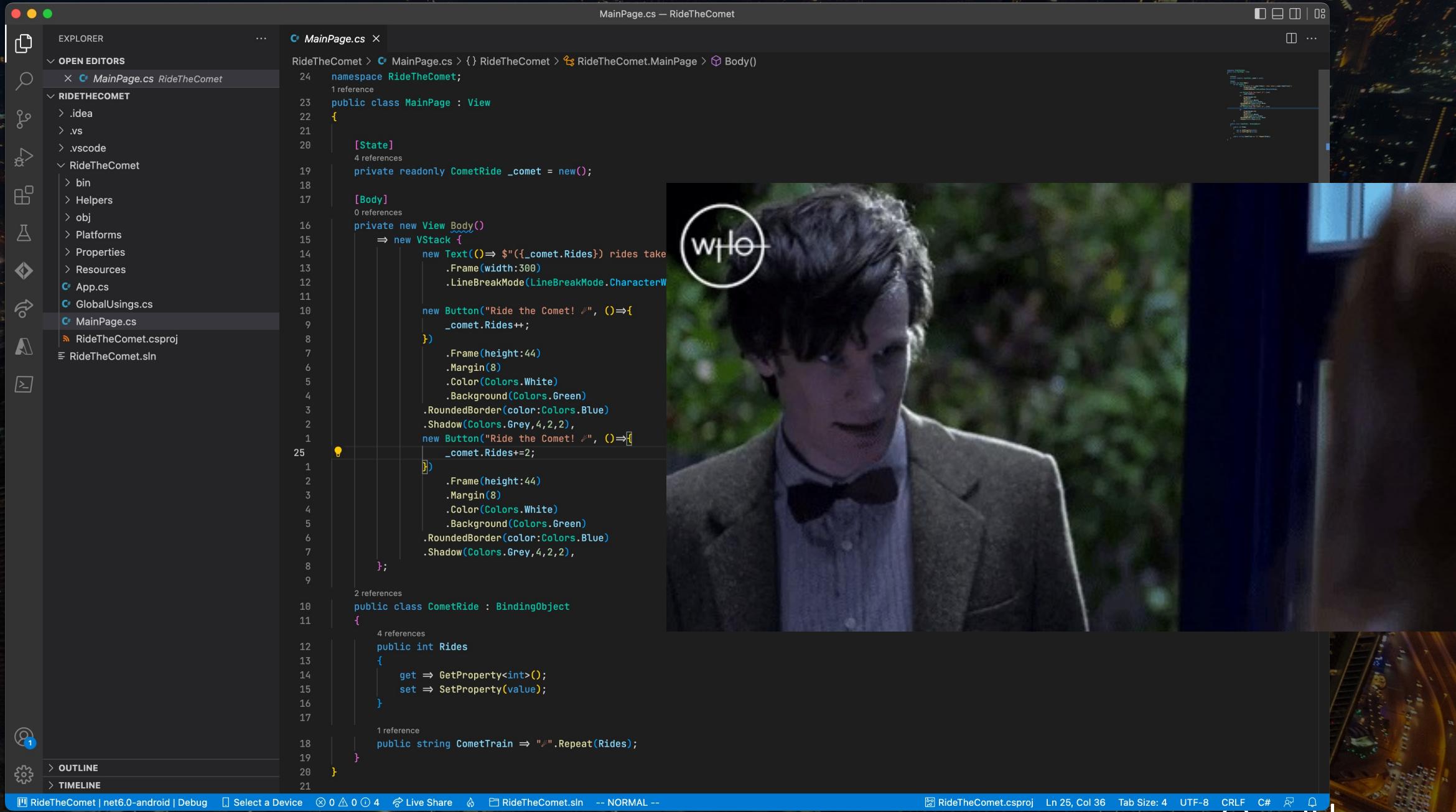
**THE MVU
ARCHITECTURE 101**

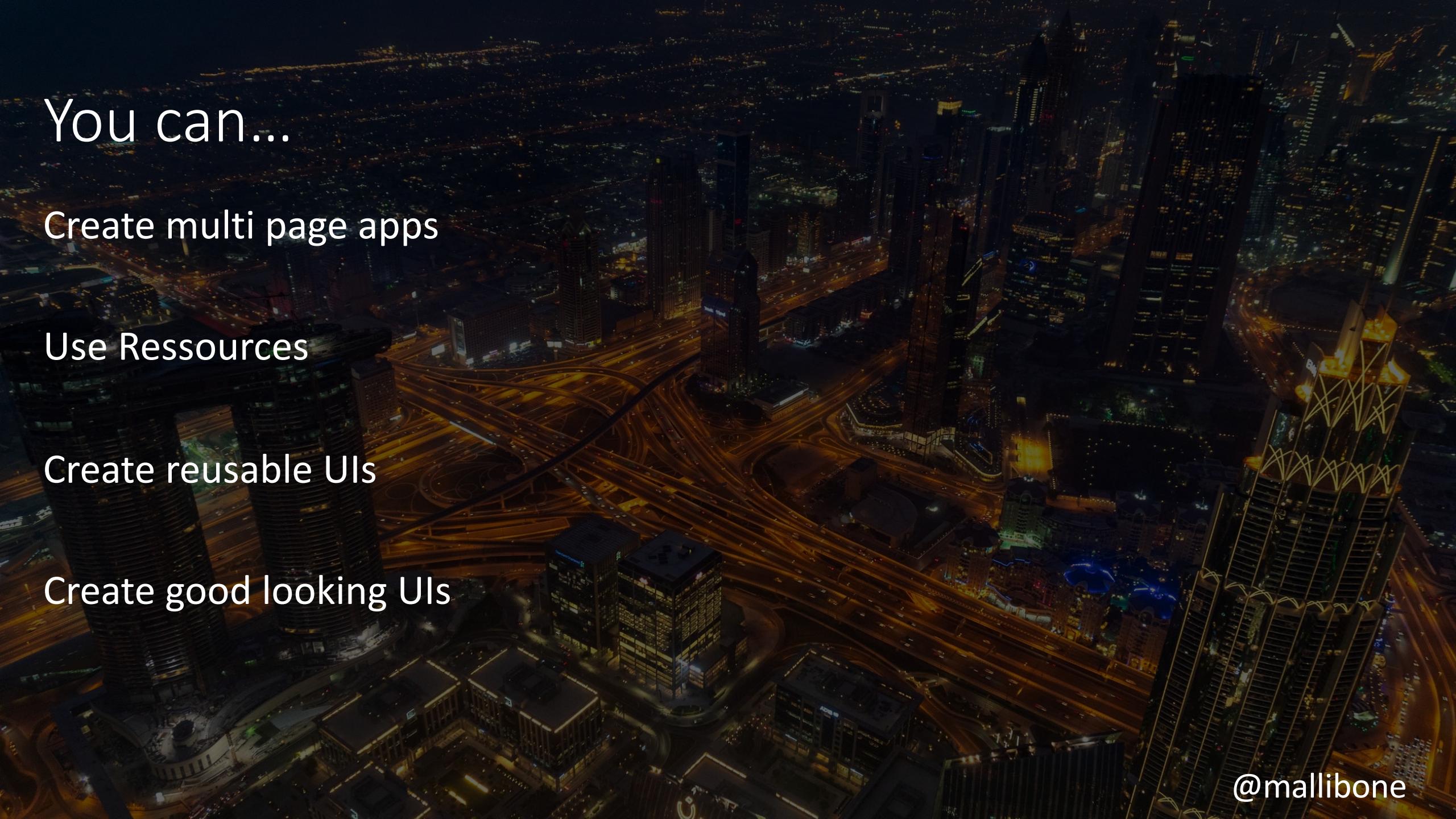
FASCINATING

NOW MAKE IT PRETTY

imgflip.com

@mallibone



The background image is a wide-angle, aerial night photograph of a major city's skyline. The city is densely packed with skyscrapers of various heights, many of which are brightly lit with blue, white, and yellow lights. A complex network of elevated highways and overpasses cuts through the city, with numerous cars and lights visible on the roads. The overall atmosphere is one of a modern, bustling urban center.

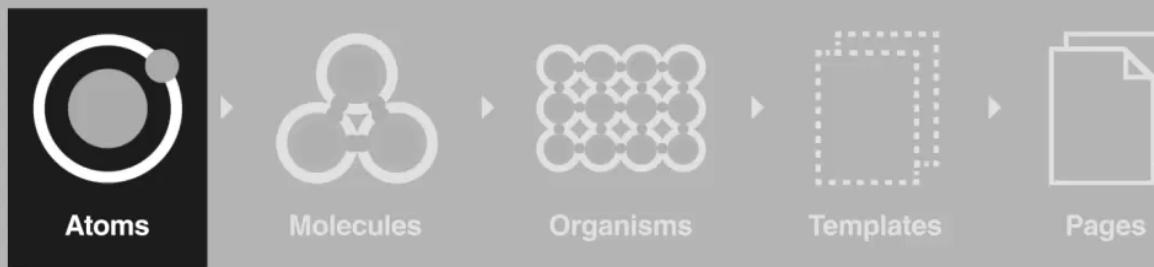
You can...

Create multi page apps

Use Ressources

Create reusable UIs

Create good looking UIs

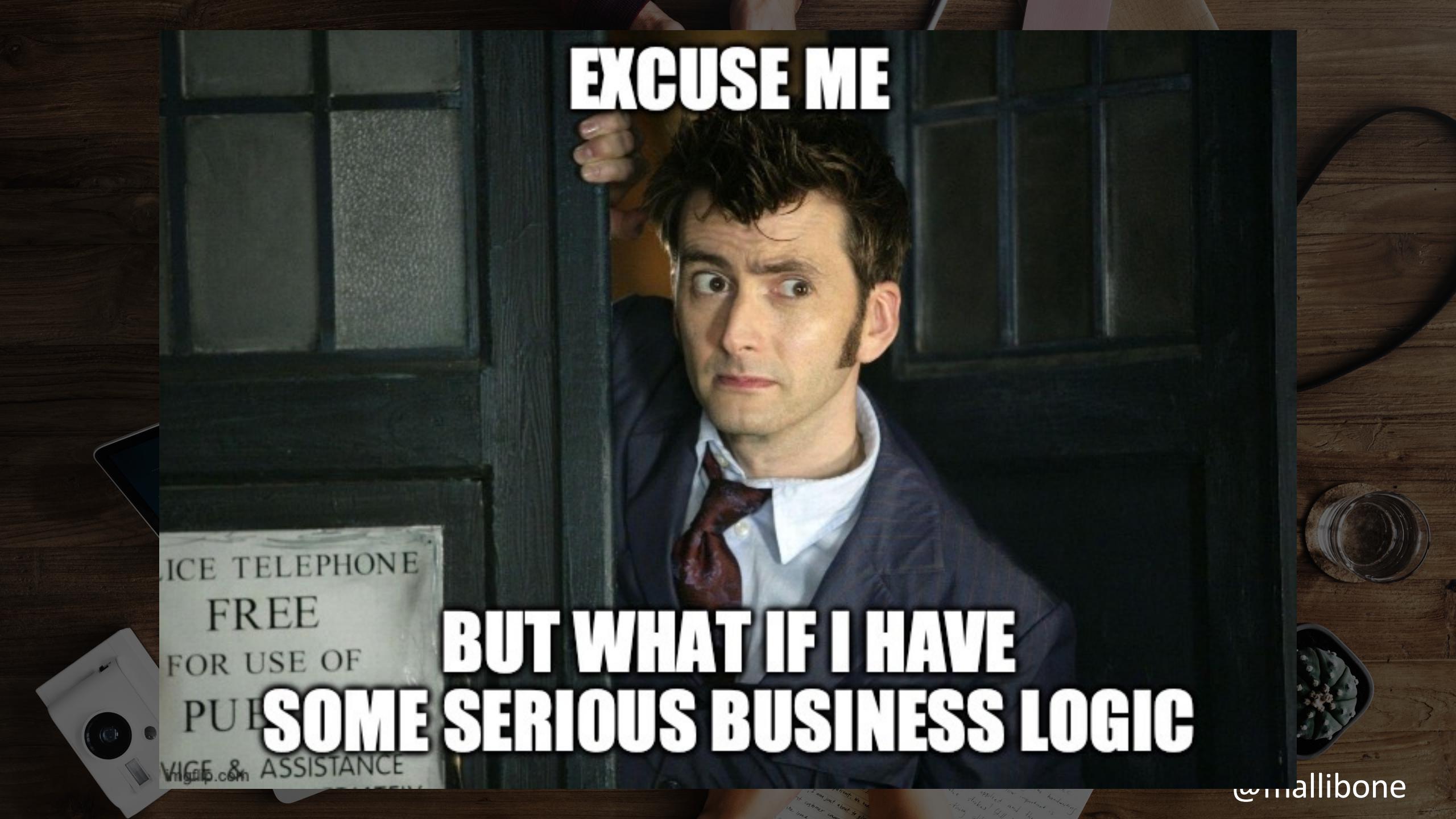




Hot Reload ❤️ Coded UI



@mallbone



EXCUSE ME

**BUT WHAT IF I HAVE
SOME SERIOUS BUSINESS LOGIC**

ICE TELEPHONE
FREE
FOR USE OF
PUB
VICE & ASSISTANCE

@mhallibone

MainPage.cs — RideTheComet

EXPLORER OPEN EDITORS MainPage.cs RideTheComet RIDETHECOMET .idea .vs .vscode RideTheComet bin Helpers obj Platforms Properties Resources App.cs GlobalUsings.cs MainPage.cs RideTheComet.csproj RideTheComet.sln

```
 MainPage.cs
24 namespace RideTheComet;
25     1 reference
26     public class MainPage : View
27     {
28
29         [State]
30         4 references
31         private readonly CometRide _comet = new();
32
33         [Body]
34         0 references
35         private new View Body()
36             => new VStack {
37                 new Text(_comet.Rides) rides taken:{_comet.CometTrain}
38                     .Frame(width:300)
39                     .LineBreakMode(LineBreakMode.CharacterWrap),
40
41                 new Button("Ride the Comet! ✈", ()=>
42                     _comet.Rides++);
43             }
44             .Frame(height:44)
45             .Margin(8)
46             .Color(Colors.White)
47             .Background(Colors.Green)
48             .RoundedBorder(color:Colors.Blue)
49             .Shadow(Colors.Grey,4,2,2),
50             new Button("Ride the Comet! ✈", ()=>
51                 _comet.Rides+=2;
52             )
53             .Frame(height:44)
54             .Margin(8)
55             .Color(Colors.White)
56             .Background(Colors.Green)
57             .RoundedBorder(color:Colors.Blue)
58             .Shadow(Colors.Grey,4,2,2),
59         };
60
61         2 references
62         public class CometRide : BindingObject
63         {
64             4 references
65             public int Rides
66             {
67                 get => GetProperty<int>();
68                 set => SetProperty(value);
69             }
70
71             1 reference
72             public string CometTrain => "✈".Repeat(Rides);
73         }
74     }
75 }
```

RideTheComet | net6.0-android | Debug Select a Device ① 0 △ 0 ② 4 Live Share RideTheComet.sln -- NORMAL -- RideTheComet.csproj Ln 25, Col 36 Tab Size: 4 UTF-8 CRLF C# @mailibone

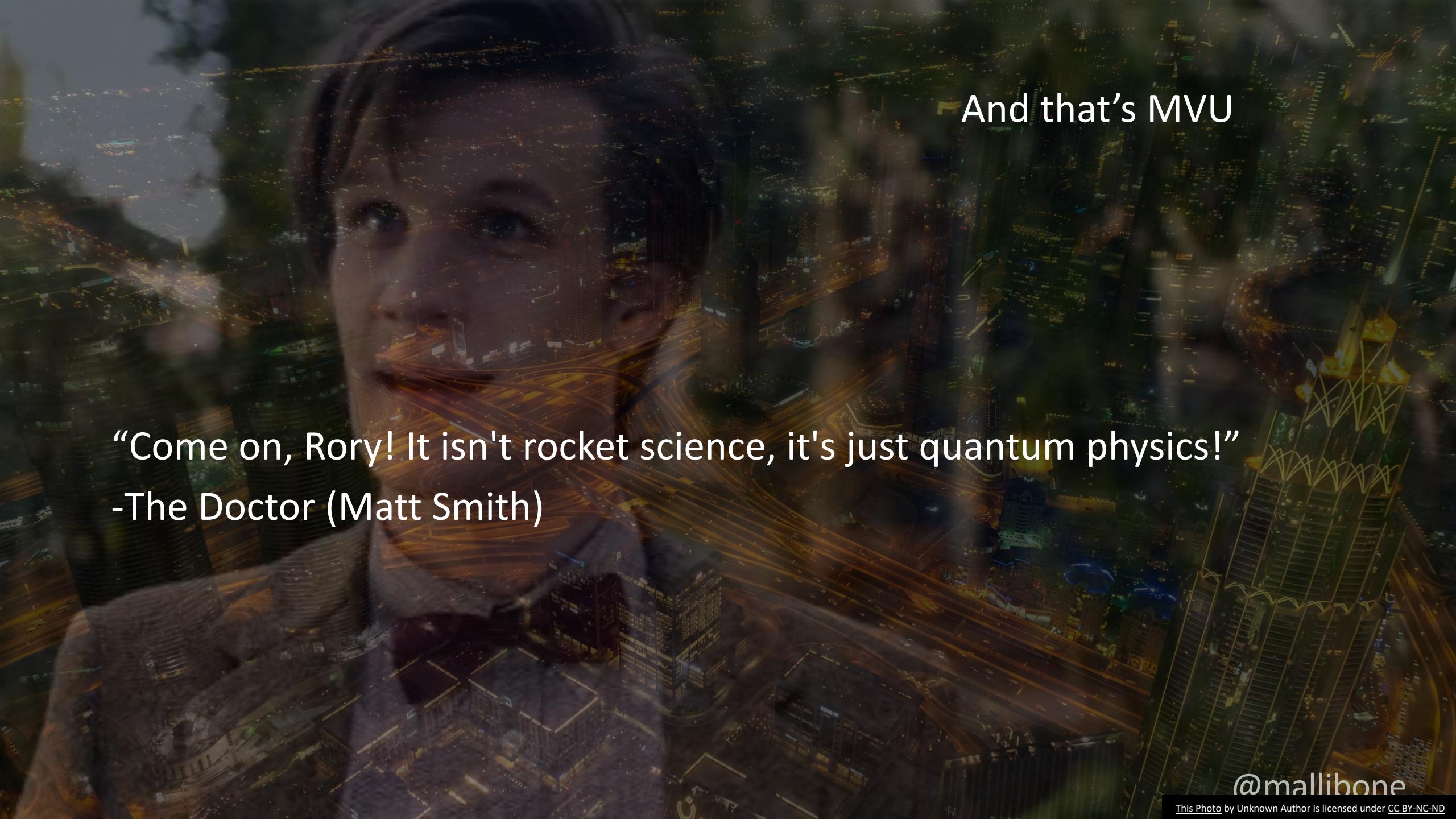
BLOBs and MVU

Dependency Injection via ServiceLocator

Reuse Business Logic across Views

Extract Services/Libs into separate Library (🔥 Reload)

Go for BLOBs

A wide-angle, aerial night photograph of a dense urban landscape. The city is filled with numerous skyscrapers of varying heights, their windows glowing with lights. A complex network of elevated highways and overpasses cuts through the city, with streaks of light from moving vehicles creating a sense of motion. The overall atmosphere is dark and futuristic.

And that's MVU

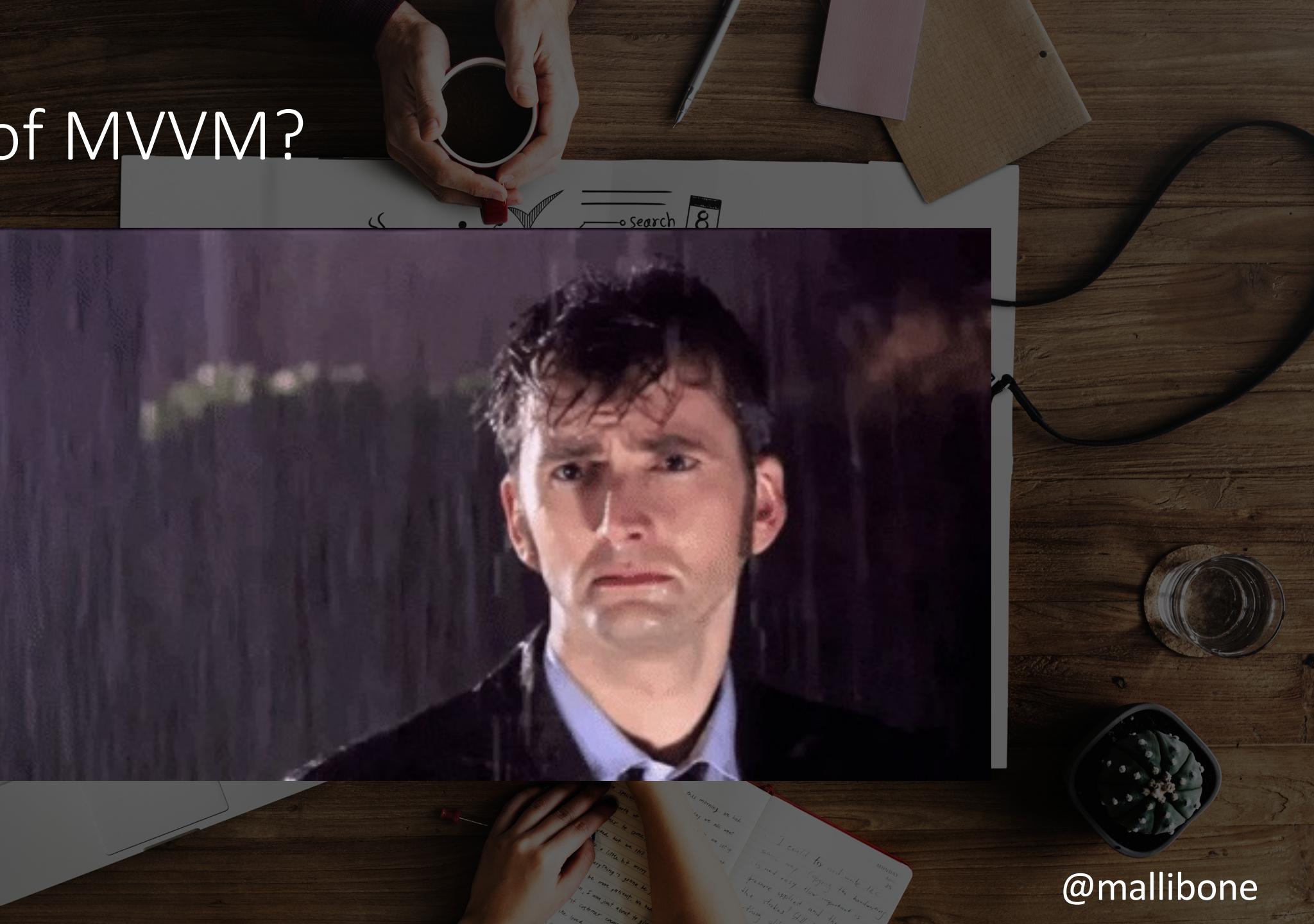
“Come on, Rory! It isn't rocket science, it's just quantum physics!”

-The Doctor (Matt Smith)

@mallibone

This Photo by Unknown Author is licensed under CC BY-NC-ND

The end of MVVM?



@mallibone

Takeaways



Takeaways

- Comet/MauiReactor provide MVU for .NET MAUI C# devs
- Less Ceremony
- Use all the libraries you know and love when developing Model View Update
- Hot Reload dev experience





Thank you for your time!



Mark Allibone



Rey Technology



@mallibone



<https://mallibone.com>



<https://nullpointers.io>



<https://adospace.gitbook.io/mauireactor/>

