



Building cross-device experiences with
Uno Platform and Elmish in F#

Elmish.Uno

DOT
NEXT

- Azure архитектор
- DevOps автоматизатор
- Фанат:
F#, XAML, Blend,
функционального
программирования,
Формулы 1
- MCSD: Windows 8 C#
- Соучредитель:
The Secret Circle Solutions
- Преподаватель:
Компьютерной Академии ШАГ
- Contributor
Dial Tools for Visual Studio

Андрей Чебукин
@XperiAndri





Building cross-device experiences with
Uno Platform and Elmish in F#

Elmish.Uno

DOT
NEXT

Ты .NET-чик, любишь XAML и
хочешь писать под mobile?

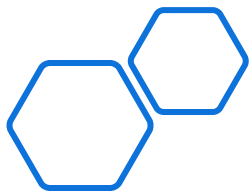
И чтобы

- Без изучения новых платформ
- Без странных концепций
- Так чтобы переиспользовать
- И чтобы минимум баг

я покажу

Сейчас полюбишь

Ещё не любишь Blend?



Повестка

- Uno Platform
- Elmish.Uno
- С чего начать – лучший TTFNW
- Архитектурные приёмы из реальных проектов

Мечта, часть 1



DOT
NEXT



Blend



Visual Studio interface showing the design and code view of a Xamarin.Forms application.

Top Bar: FILE, EDIT, VIEW, GIT, PROJECT, BUILD, DEBUG, TEAM, TOOLS, EXTENSIONS, WINDOW, HELP. Search (Ctrl+Q).

Left Panel:

- Solution Explorer:** Solution 'DotNext' (2 of 2 projects)
 - MotionPath
 - VisualStates (Universal Windows)**
 - Properties
 - References
 - Assets
 - App.xaml
 - MainPage.xaml
 - Package.appxmanifest
- Assets:** [Empty]
- States:** [Empty]
- Objects and Timeline:** (No Storyboard open)
 - [Page]
 - BottomAppBar
 - grid
 - button** [EventTriggerBehavior]

Design View: MainPage.xaml. 5" Phone (1920 x 1080) 300% scale. Effective: 640 x 360. A button is visible on the screen.

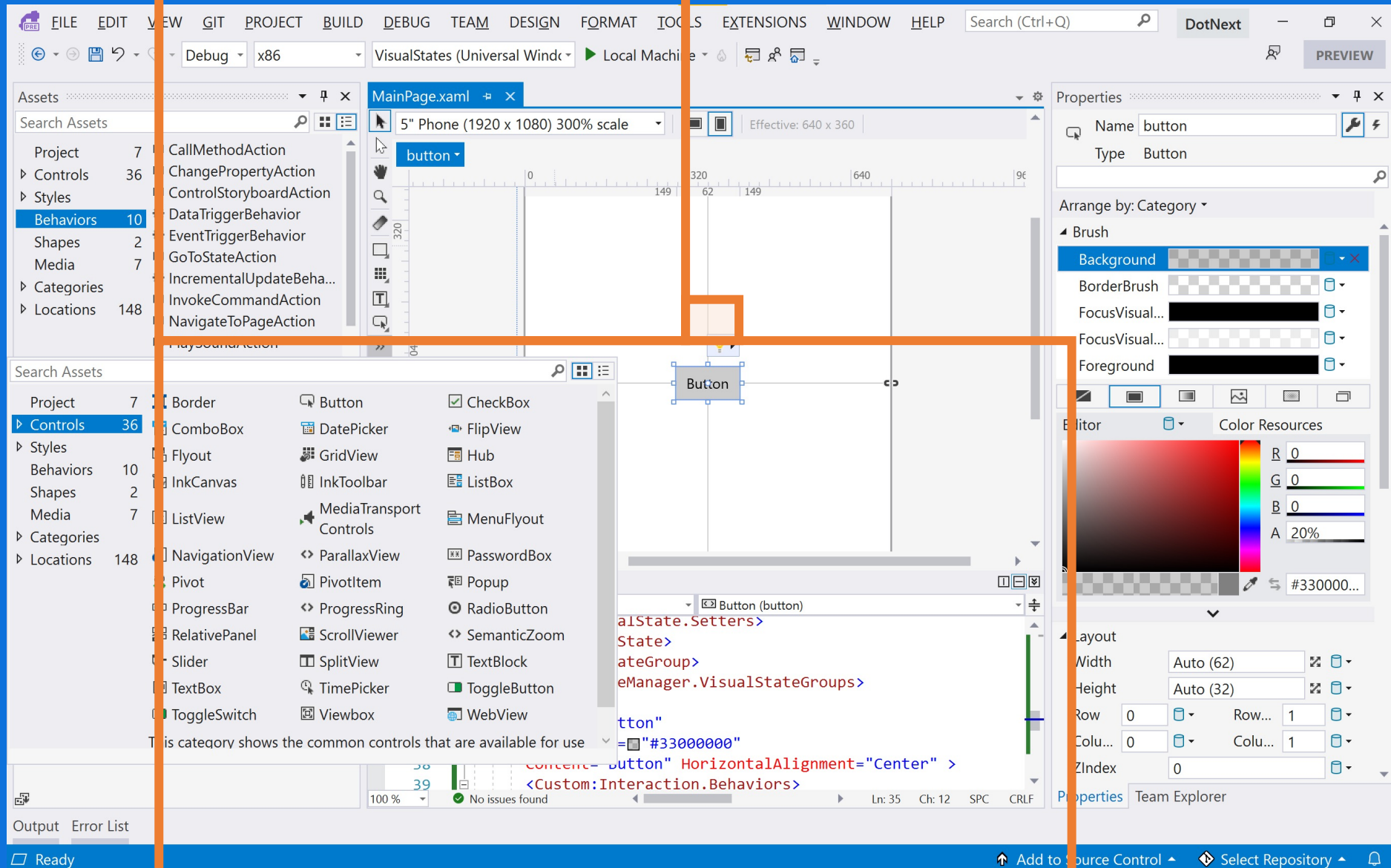
Code View:

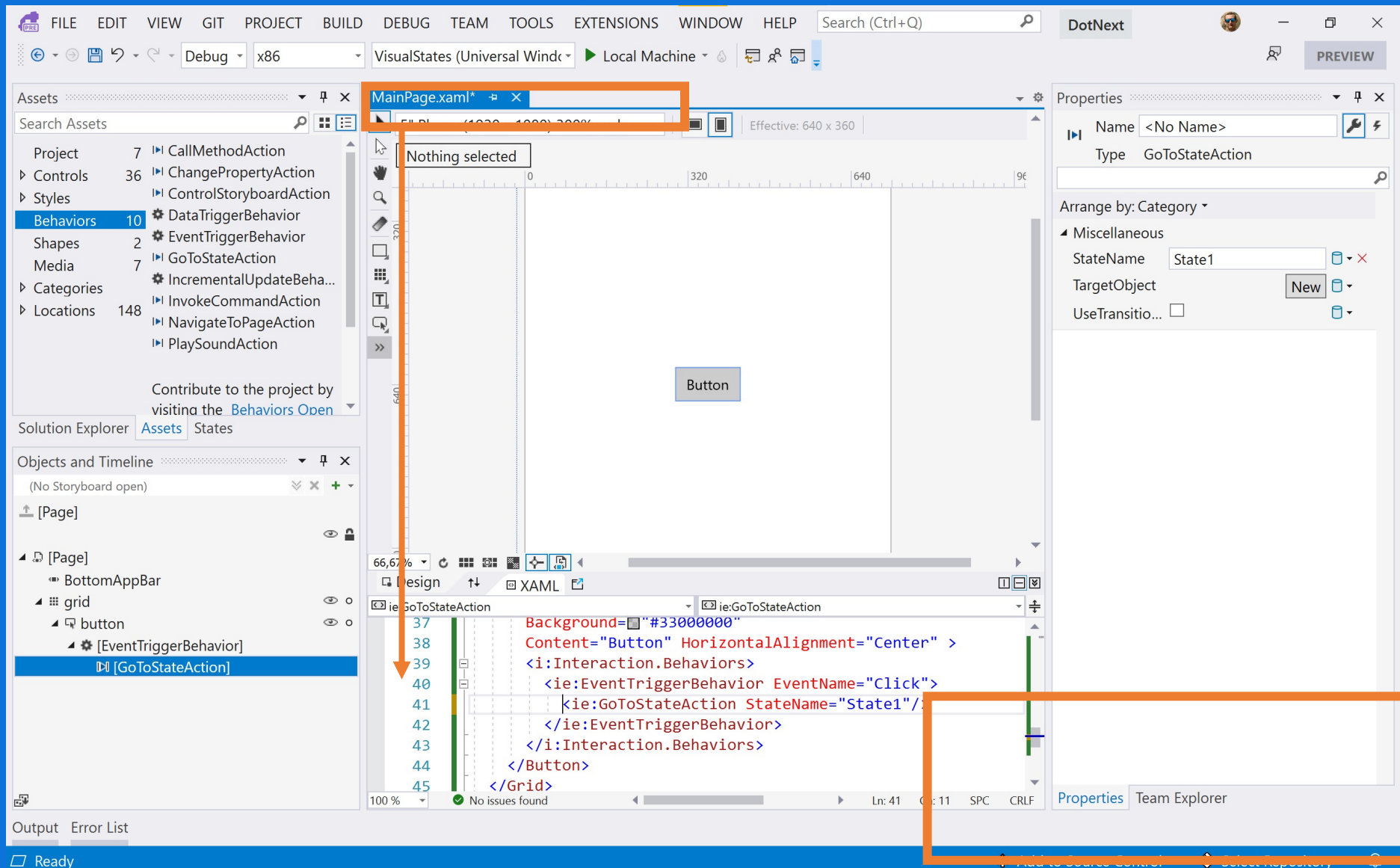
```
</VisualState.Setters>
</VisualState>
</VisualStateGroup>
</VisualStateManager.VisualStateGroups>
<Button
  x:Name="button"
  Background="#33000000"
  Content="Button" HorizontalAlignment="Center" >
  <Custom:Interaction.Behaviors>
```

Right Panel:

- Properties:** Name: button, Type: Button.
- Brush:** Background: [Checkerboard], BorderBrush: [Checkerboard], FocusVisual...: [Black], FocusVisual...: [Checkerboard], Foreground: [Black].
- Editor:** Color Resources. R: 0, G: 0, B: 0, A: 20%. #33000000.
- Layout:** Width: Auto (62), Height: Auto (32), Row: 0, Colu...: 0, Row...: 1, Colu...: 1, ZIndex: 0.

Bottom Bar: Output, Error List, Item(s) Saved, Add to Source Control, Select Repository.





Visual Studio Code interface showing the design of a Xamarin.Forms application with Visual States.

States Panel: Shows the hierarchy of visual states. The **ScreenWidthVisualStateGroup** is expanded, showing **Narrow** (selected) and **Wide** states. The **Default transition** is set to **Narrow**.

Design View: Displays a visual representation of the UI on a 5" Phone (1920 x 1080) at 300% scale. The effective resolution is 640 x 360. A button is visible in the center.

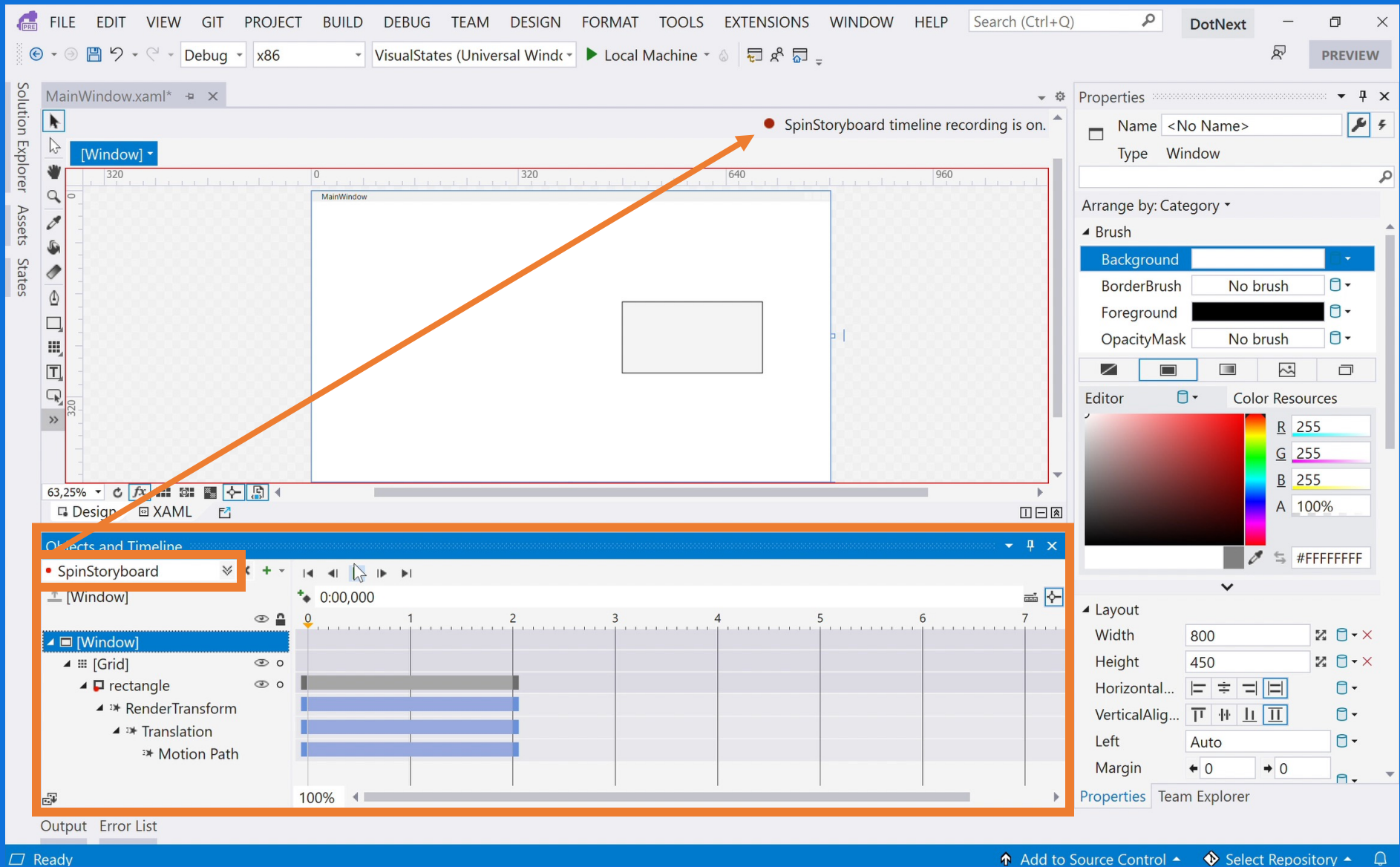
Properties Panel: Shows the properties of the selected element (Storyboard). The **Theme Animations** list includes:

- FadeInThemeAnimation
- FadeOutThemeAnimation
- PopInThemeAnimation
- PopOutThemeAnimation
- RepositionThemeAnimation
- PointerUpThemeAnimation
- PointerDownThemeAnimation
- SplitOpenThemeAnimation
- SplitCloseThemeAnimation
- SwipeBackThemeAnimation
- SwipeHintThemeAnimation
- DragItemThemeAnimation
- DropTargetItemThemeAnimation
- DragOverThemeAnimation

Objects and Timeline Panel: Shows the visual tree. The **Narrow** state is selected, showing the **button** and its **Background** property.

XAML View: Displays the XAML code for the **VisualState (Narrow)** state. The code sets the background of the button to red when the state is active.

```
11 <Grid x:Name="grid" Background="Transparent">
12 <VisualStateManager.VisualStateGroups>
13 <VisualStateGroup x:Name="ScreenWidthVisualStateGroup">
14 <VisualState x:Name="Narrow">
15 <VisualState.Setters>
16 <Setter Target="button.(Control.Background)">
17 <Setter.Value>
18 <SolidColorBrush Color="Red" />
19 </Setter.Value>
20 </VisualState.Setters>
21 </VisualState>
22 </VisualStateGroup>
23 </VisualStateManager.VisualStateGroups>
24 </Grid>
```



Visual Studio interface showing the design of a button in a Universal Windows app.

Top Bar: FILE, EDIT, VIEW, GIT, PROJECT, BUILD, DEBUG, DESIGN, FORMAT, TOOLS, EXTENSIONS, WINDOW, HELP. Search (Ctrl+Q). DotNext.

Left Panel:

- Solution Explorer:** Solution 'DotNext' (2 of 2 projects)
 - MotionPath
 - VisualStates (Universal Windows)**
 - Properties
 - References
 - Assets
 - App.xaml
 - MainPage.xaml
 - Package.appxmanifest
- Assets:** Assets, States
- Objects and Timeline:** (No Storyboard open)
 - [Page]
 - [Page]
 - BottomAppBar
 - grid
 - button**

Center Canvas: MainPage.xaml. 5" Phone (1920 x 1080) 300% scale. Effective: 640 x 360. A button is centered on the canvas.

Right Panel:

- Properties:**
 - Name: button
 - Type: Button
 - Arrange by: Category
 - OpacityTran... (New)
 - RenderTran... (0, 0)
 - RequiresPoi... (Never)
 - Rotation (0)
 - RotationAxis (Vector3) (New)
 - RotationTra... (New)
 - Scale (Vector3) (New)
 - ScaleTransit... (New)
 - Shadow (New)
 - Style (Default)
 - TabFocusN...
 - Template (ControlTemplate)
 - Transform3D (Edit...)
 - Transform... (Matrix4x4) (New)
 - Transitions (Collection) (...)
 - Translation (Vector3) (New)
 - Translation... (New)
 - UseSystemF... (checked)
 - XamlRoot (New)
 - XYFocusDo... (New)
 - XYFocusDo... (Auto)
- Team Explorer:** Properties, Team Explorer

Bottom Bar: Output, Error List. Ready. Add to Source Control. Select Repository.

Это что за хрень?

Windows 10

Windows 11

LINUX



platform.uno/discord

Uno Platform





Manage Extensions

▸ Installed

▴ Online

▴ Visual Studio Marketplace

Search Results

▸ Controls

▸ Templates

▸ Tools

▸ Updates

▸ Roaming Extension Manager

Sort by: Relevance



Uno Platform Solution Templates

A solution template for Uno Platform enabled projects.



Silverlight for Visual Studio

Unofficial support for Silverlight projects in Visual Studio 2017 and 2019



Arduino IDE for Visual Studio

A fully compatible Arduino Edit, Build and Deployment tool. All Arduino IDE versions are supported. This is an admin install for all users. (option...

uno

Created by: nventive

Version: 1.43.1.826

Downloads: 2261

Pricing Category: Free

Rating: ★★★★★ (7 Votes)

[More Information](#)

[Report Extension to Microsoft](#)

Scheduled For Install:

None

Scheduled For Update:

None

Scheduled For Uninstall:

None

1

[Change your settings for Extensions](#)

Close

Create a new project

Recent project templates

A list of your recently accessed templates will be displayed here.

Search for templates (Alt+S)

Clear all

All languagesWindowsUno Platform



Multi-Platform Library (Uno Platform)

New

Template for creating a multi-platform library with the Uno Platform that targets UWP, WebAssembly, Android, iOS, macOS and Linux

C#iOSAndroidWindowsmacOSWeb
MobileUWPWebAssemblyUno Platform



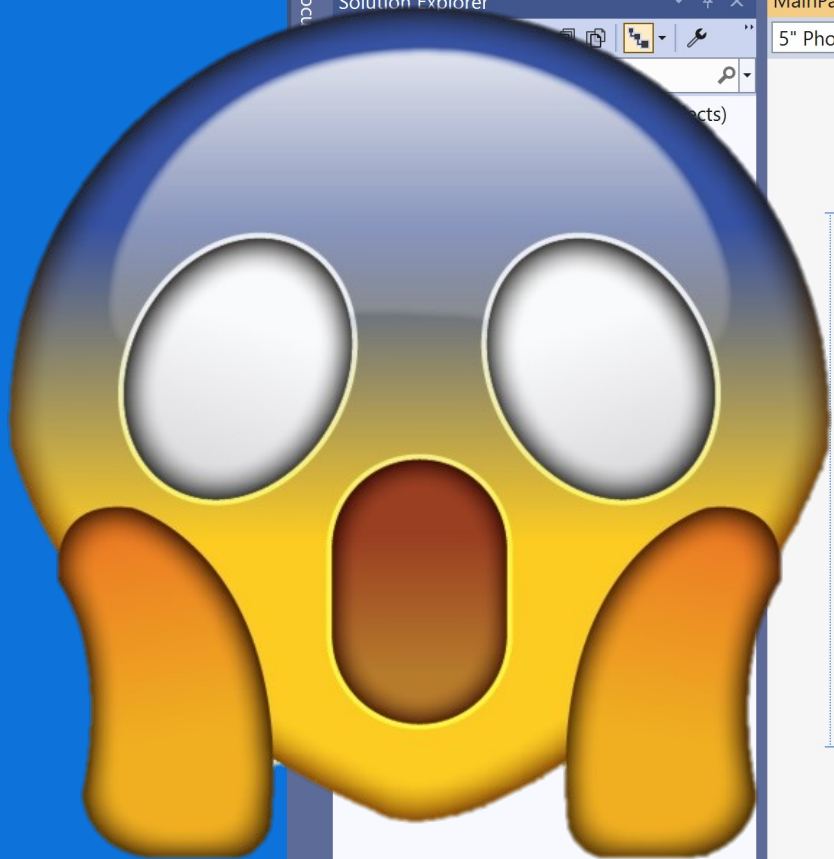
Multi-Platform App (Uno Platform)

New

Template for creating a multi-platform C# & XAML application targeting UWP, WebAssembly, Android, iOS, macOS and Linux

C#iOSAndroidWindowsmacOSTizenWeb
MobileUWPWebAssemblyUno Platform

Next



Visual Studio Code interface showing a Xamarin.Forms project named 'UnoMvvm.UWP'.

Menu Bar: FILE, EDIT, VIEW, GIT, PROJECT, BUILD, DEBUG, CODERUSH, DESIGN, FORMAT, ARCH, TEST, ANALYZE, TOOLS, EXT, WIN, ?

Toolbar: Includes icons for file operations, debugging, and a 'PREVIEW' button.

Solution Explorer: Shows the project structure with 'MainPage.xaml' selected.

MainPage.xaml Design View: Displays a visual representation of the page on a '5" Phone (1920 x 1080) 300% scale' with the text 'Hello, world!'.

MainPage.xaml Source View: Shows the XAML code for the page:

```
1 <Page
2   x:Class="UnoMvvm.MainPage"
3   xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
4   xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
5   xmlns:local="using:UnoMvvm"
6   xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
7   xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility"
8   mc:Ignorable="d">
9
10  <Grid Background="{ThemeResource ApplicationPageBackgroundTheme"
11        <TextBlock Text="Hello, world!" Margin="20" FontSize="30" />
12  </Grid>
13 </Page>
14
```

Notifications / Toolbox / Properties: Panel on the right side of the interface.

Status Bar: Shows 'Ready', '40,19%', '100 %', 'No issues found', and 'Ln: 1 Ch: 1 MIXED CRLF'.

Bottom Panel: Includes tabs for 'Package Manager Console', 'Output', 'Breakpoints', 'Exception Settings', 'Bookmarks', 'Test Runs', 'Error List', and 'Task Runner Explorer'.

Footer: Includes 'Add to Source Control' and 'Select Repository' buttons.

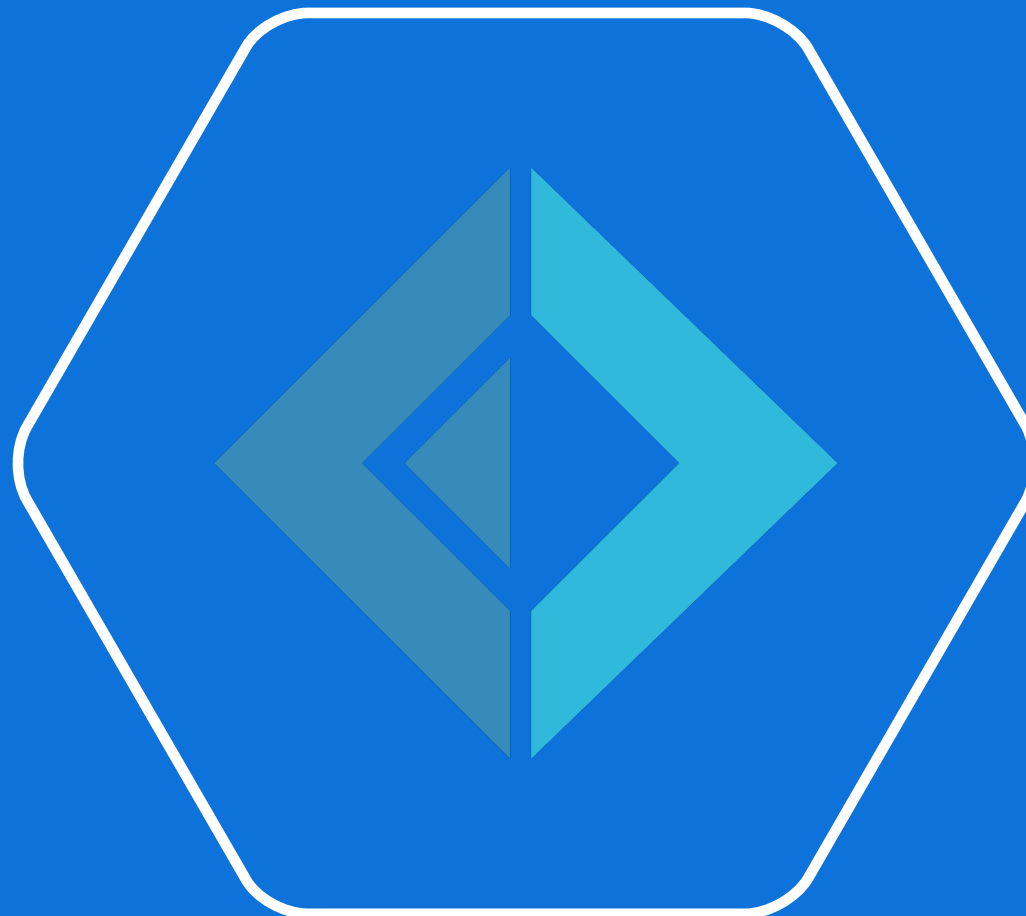
Мечта, часть 2

A decorative graphic consisting of three white hexagons on a blue background. One hexagon is solid white, while the other two are white outlines. They are arranged in a cluster at the top center of the slide.

DOT
NEXT



F#



F# может сломать вам мозг...

Как описать факториал без явной рекурсии и без использования переменных?

```
let rec fact n = if n=1 then 1 else n*fact(n-1)
```

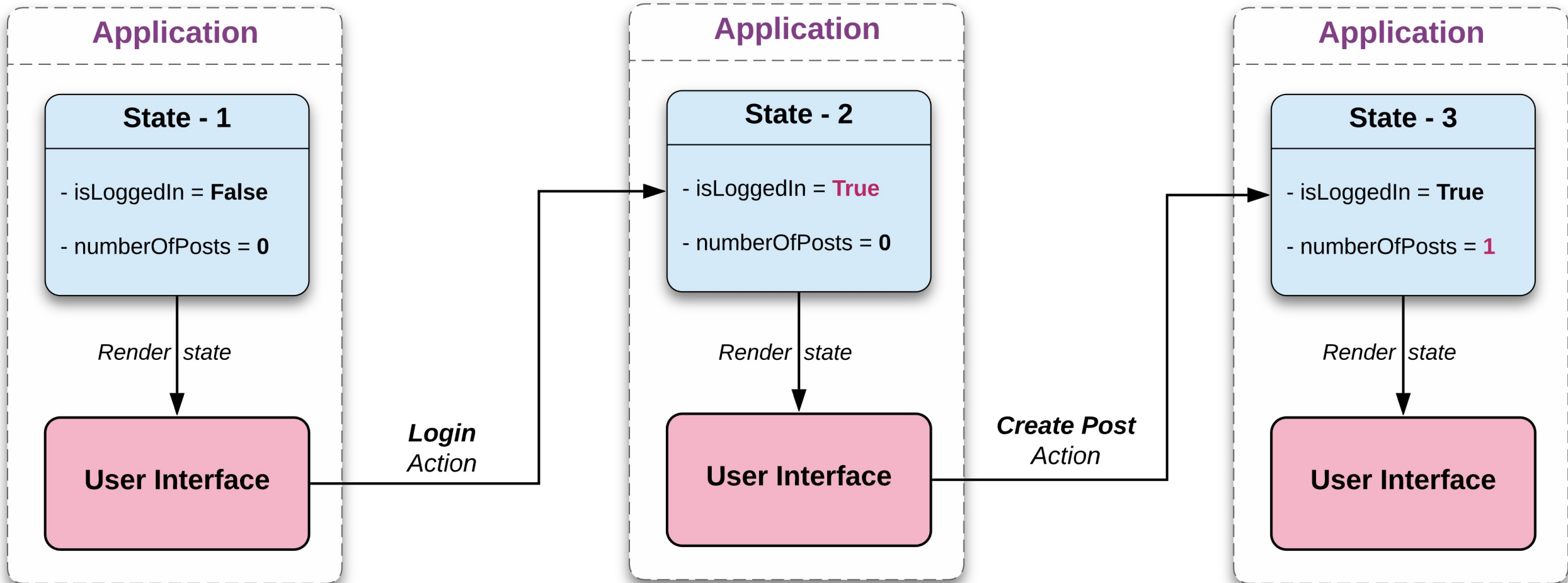
```
let fact = y ((<<)(cond ((=)0)(k 1))((<<)(s(*))((>>)(c(-)1))))
```



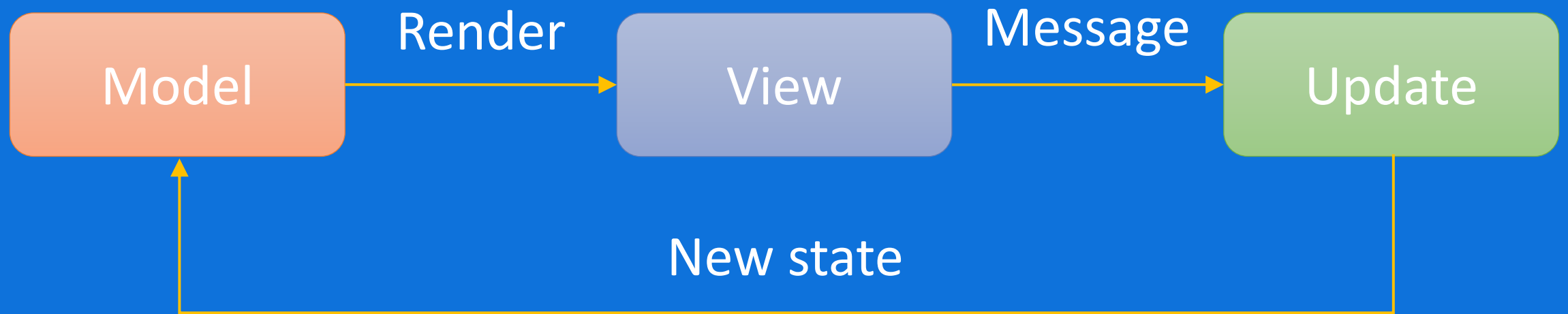


Elm





Model-View-Update



Пример кода на Elm

```
type alias Model = Int
```

```
init : Model
```

```
init = 0
```

```
type Msg
```

```
  = Increment
```

```
  | Decrement
```

```
update : Msg -> Model -> Model
```

```
update msg model =
```

```
  case msg of
```

```
    Increment -> model + 1
```

```
    Decrement -> model - 1
```

```
view : Model -> Html
```

```
view model =
```

```
  div []
```

```
    [ button [ onClick Decrement ] [ text "-" ]
```

```
      , div [] [ text (String.fromInt model) ]
```

```
      , button [ onClick Increment ] [ text "+" ]
```

```
    ]
```

```
main =
```

```
  Browser.sandbox { init = init, update = update, view = view }
```

MAUI

```
readonly State<int> count = 0;
```

```
[Body]
```

```
View body() => new StackLayout
```

```
{
```

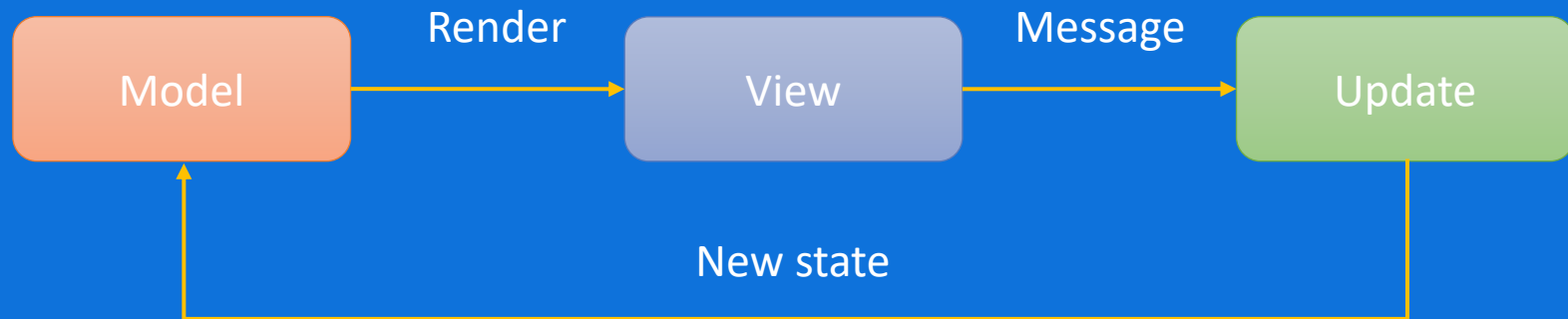
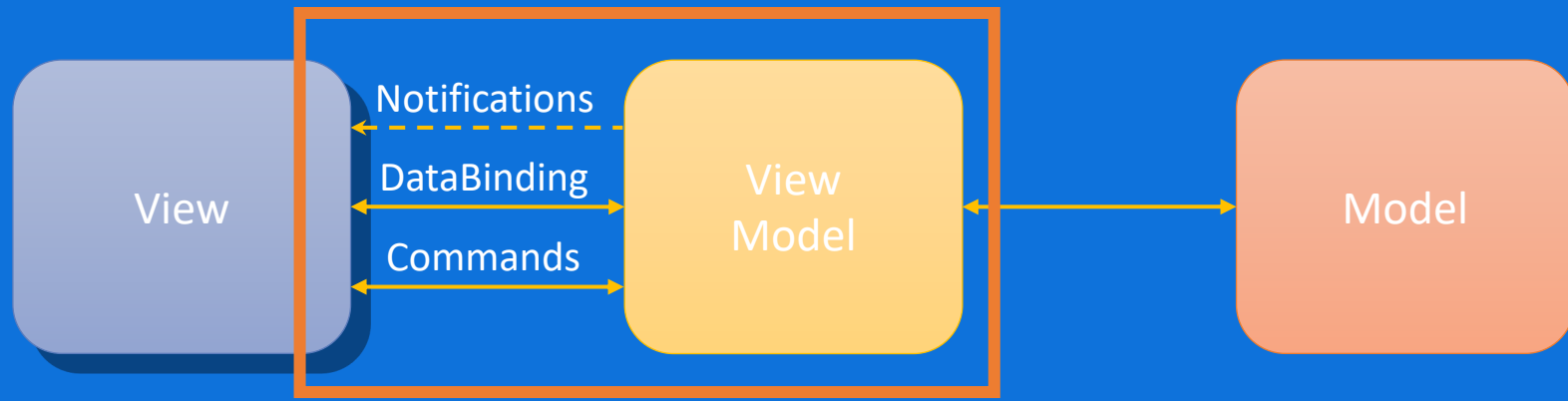
```
    new Button(() => "-", () => count.Value --)
```

```
    new Label($"{count.Value}"),
```

```
    new Button(() => "+", () => count.Value ++)
```

```
};
```

MVVM VS. MVU



Elm — это, как бы, не .NET...

Elmish



Но Elmish — это, как бы, Web...

Таки есть Fabulous (Xamarin/MAUI)

```
module App =
```

```
/// The messages dispatched by the view
```

```
type Msg =  
    | Increment  
    | Decrement
```

```
/// The model from which the view is generated
```

```
type Model =  
    { Count: int }
```

```
/// Returns the initial state
```

```
let init() = { Count = 0 }
```

```
/// The function to update the view
```

```
let update (msg: Msg) (model: Model) =  
    match msg with  
    | Increment -> { model with Count = model.Count + 1 }  
    | Decrement -> { model with Count = model.Count - 1 }
```

```
/// The view function giving updated content for the page
```

```
let view (model: Model) dispatch =
```

```
    View.ContentPage(  
        View.StackLayout([
```

```
            if model.Count < 0 then
```

```
                View.Label("Count is negative")
```

```
                View.Button(text = "-",  
                    command = fun () -> dispatch Decrement
```

```
                View.Label($"{{mode.Count}}")
```

```
                View.Button(text = "+",  
                    command = fun () -> dispatch Increment
```

```
            )
```

```
        ])
```

```
    )
```

```
type App () as app =
```

```
    inherit Application ()
```

```
let runner =
```

```
    // use Program.mkProgram if your app contains commands
```

```
    Program.mkSimple App.init App.update App.view
```

```
    |> Program.withConsoleTrace
```

```
    |> XamarinFormsProgram.run app
```

Почему не Fabulous?

- Сложно интегрировать сторонние компоненты
- Мудрёный способ достичь виртуализации интерфейса
- Лично мне писать интерфейс кодом не заходит
- Xamarin.Forms — отвратительная поделка из «говна и палок»



УСІ

ЗОБРАЖЕННЯ

ВІДЕО

НОВИНИ

77,200 Результати

Elmish.Uno - Static UWP views for elmish programs running ...

<https://www.findbestopensource.com/product/unoplatform-elmish-uno> ▼

Elmish.Uno - Static UWP views for **elmish** programs running with the **Uno Platform** 22 This library uses **Elmish**, an Elm architecture implemented in F#, to build WPF applications. **Elmish** was originally written for Fable applications, however it was trimmed and packaged for .NET as well. Create an F# Console Application (you can create a Windows ...

GitHub - LuisFX/Uno.Elmish: Static UWP views for elmish ...

<https://github.com/LuisFX/Uno.Elmish> ▼

Static UWP views for **elmish** programs running with the **Uno Platform** - GitHub - LuisFX/Uno.Elmish: Static UWP views for **elmish** programs running with the **Uno Platform**

Elmish.Uno/README.md at uno · unoplatform/Elmish.Uno

<https://github.com/unoplatform/Elmish.Uno/blob/uno/README.md> ▼

Static UWP views for **elmish** programs running with the **Uno Platform** - Elmish.Uno/README.md at uno · unoplatform/Elmish.Uno

elmish wpf - Can Model View Update be used with Uno ...

<https://stackoverflow.com/questions/68126402/can...> ▼

25/06/2021 · However, **Elmish.Uno** is troubled: A: 3 years since last maintainance B: from the page and docs, it is unclear to me how this project works with **Uno**, and if it is cross **platform**. Are there alternatives to **Elmish.Uno**, which also provide Model View Update?







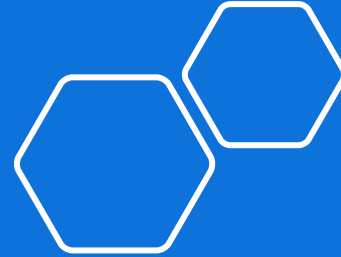
Elmish.WPF



Elmish + Uno = Elmish.Uno 🐱💻

Разберёмся как это работает

Примеры это
хорошо...



DOT
NEXT

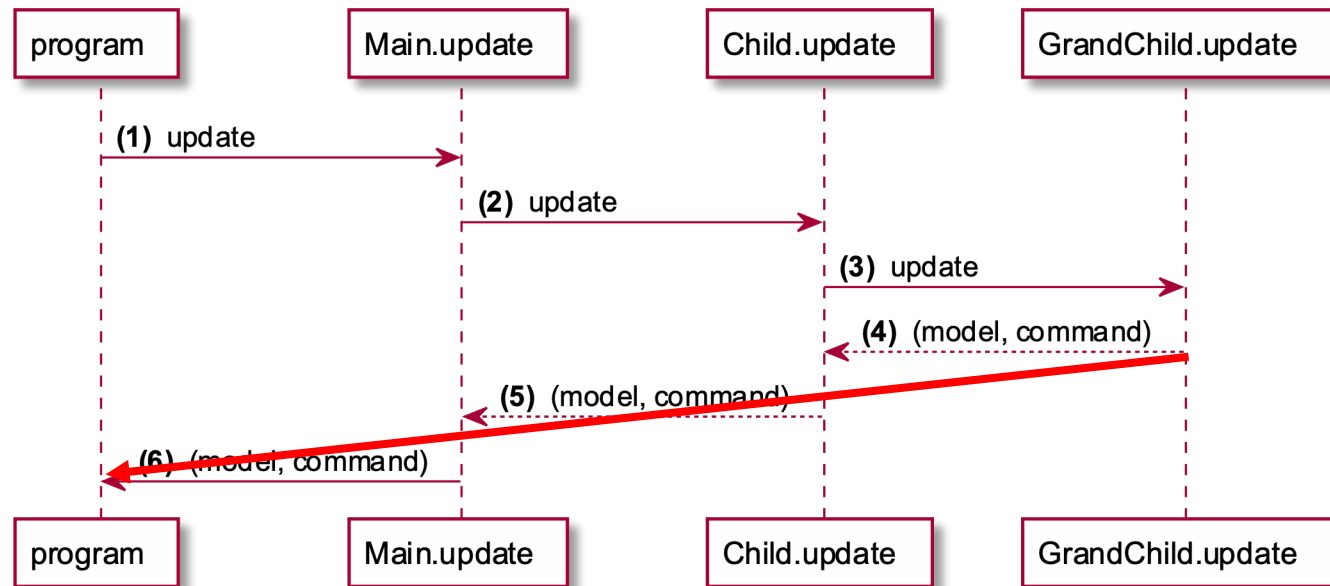
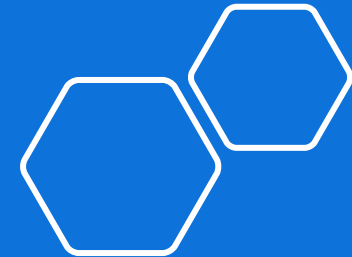


Чего не хватает многим
OSS проектам?

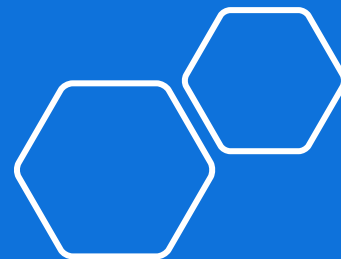
- dotnet new
 - i "Elmish.Uno.ProjectTemplates.Dotnet::1.0.0-ci-*
 - nuget-source "https://www.myget.org/F/elmish_uno/api/v3/index.json"
- dotnet new elmish-unoapp

Шаблон

- Полноценный проект с .editorconfig, анализаторами и базовой архитектурой
- Контейнер внедрения зависимостей как в ASP.NET Core/MAUI
- Конфигурация
- Настроенный работающий Linker
- Вывод оповещений
- Чистый logout



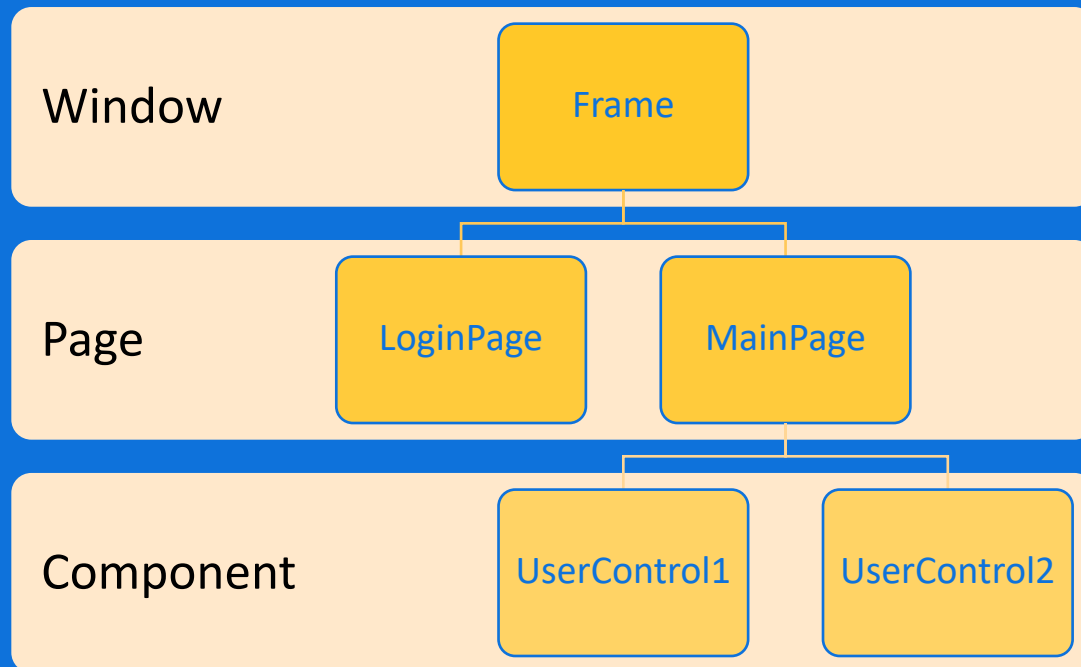
Опыт реальных проектов



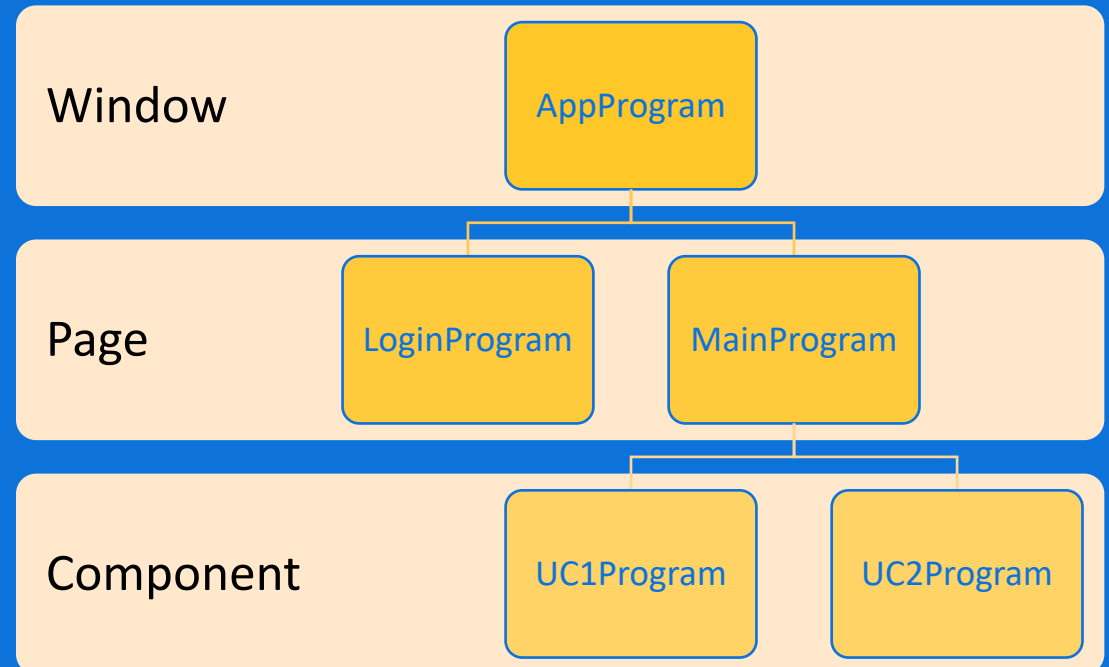
DOT
NEXT

Классический функциональный подход

UI

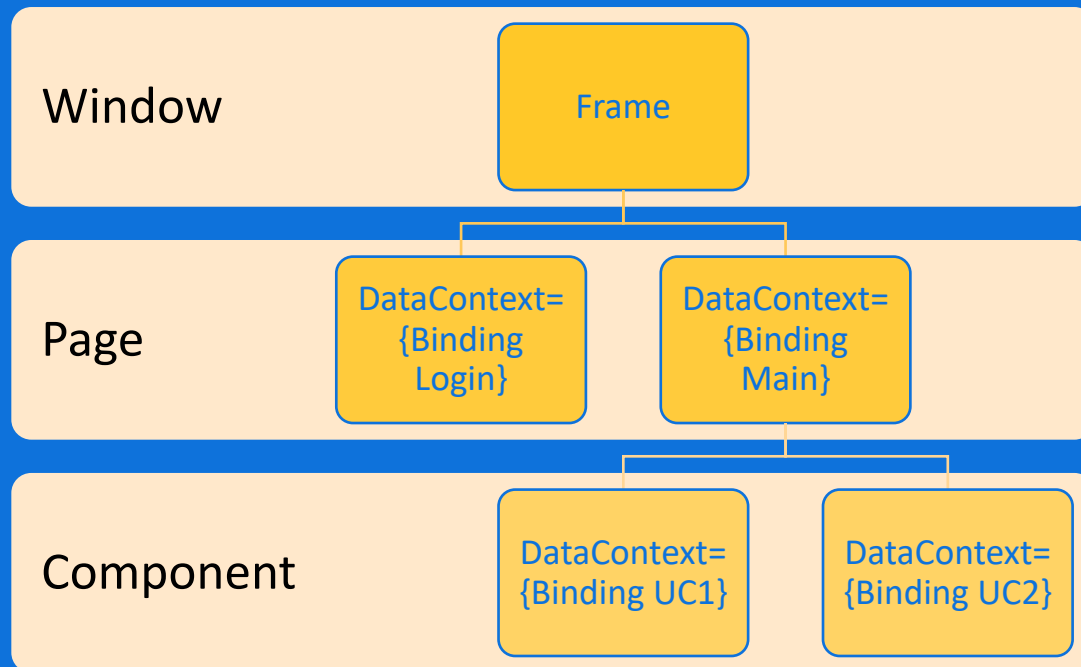


Code

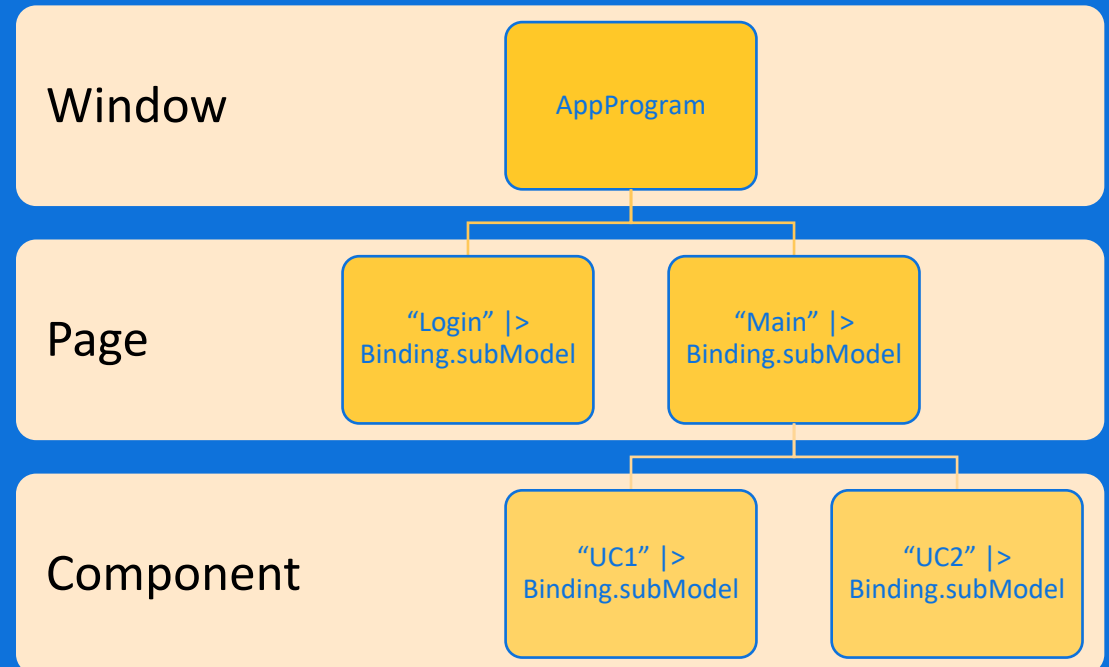


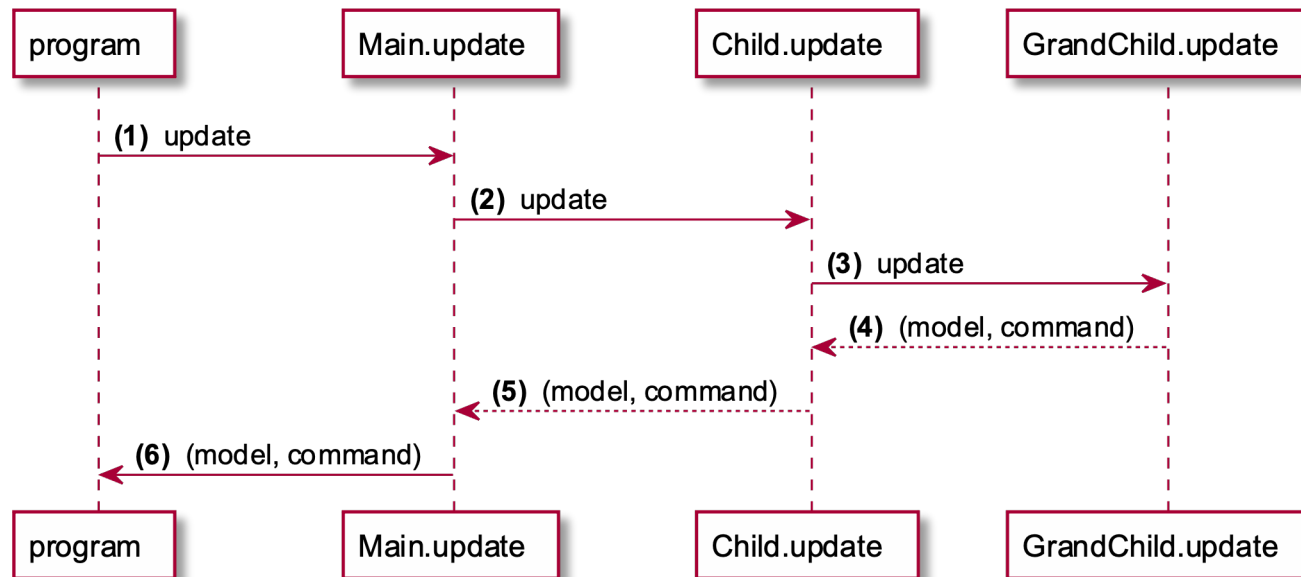
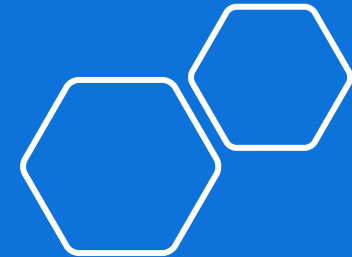
Классический функциональный подход

UI



Code





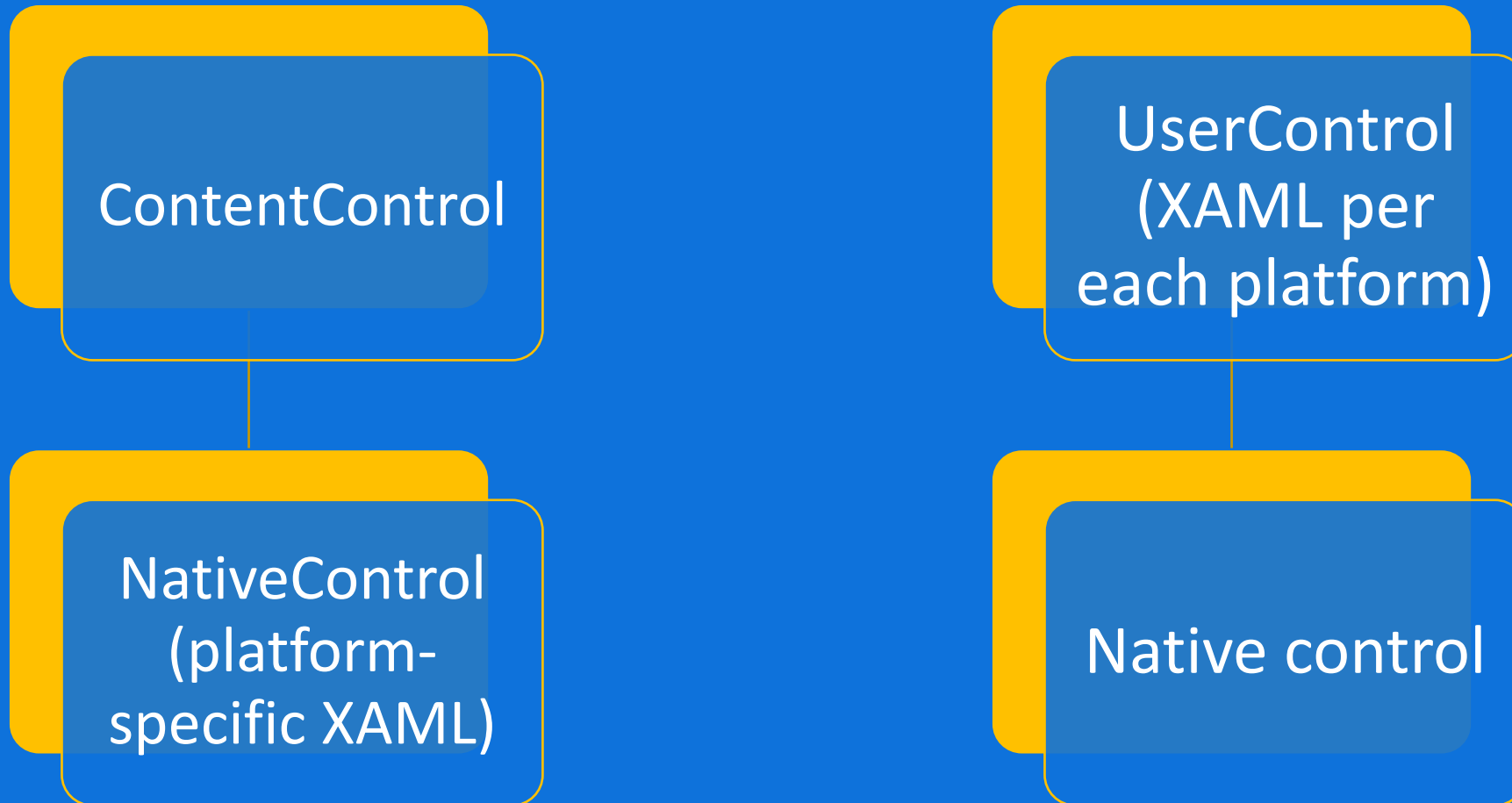
Смешанный вариант



Смешанный вариант



Рендеринг любого компонента



Проблема невалидных состояний

```
<VisualStateGroup x:Name="RIdInputStatesGroup">
  <VisualState x:Name="RId">
    <VisualState.Setters>
      <Setter
        Value="Collapsed"
        Target="vinTextBox.(UIElement.Visibility)"/>
    </VisualState.Setters>
  </VisualState>
  <VisualState x:Name="ExistingCarInfo">
    <VisualState.Setters>
      <Setter
        Value="True"
        Target="vinTextBox.(TextBox.IsReadOnly)"/>
    </VisualState.Setters>
  </VisualState>
  <VisualState x:Name="NewCarInfo" />
  <VisualState x:Name="ConfirmedCarInfo">
    <VisualState.Setters>
      <Setter
        Value="True"
        Target="vinTextBox.(TextBox.IsReadOnly)"/>
    </VisualState.Setters>
  </VisualState>
</VisualStateGroup>
```

```
type Model =
  | RId of
    RId: string
    * IsLoading: bool
  | ExistingCarInfo of
    Id: string
    * Vin: string
    * RId: string
    * Make: string
    * Model: string
    * Year: string
  | NewCarInfo of
    Vin: string
    * RId: string
    * Make: string
    * Model: string
    * Year: string
  | ConfirmedCarInfo of
    Id: string
    * Vin: string
    * RId: string
    * Make: string
    * Model: string
    * Year: string
```

СВЯЗЬ

```
[<AutoOpen>]
module State =
open Microsoft.FSharp.Reflection

let getUnionCaseName (x: 'a) =
    match FSharpValue.GetUnionFields(x, typeof<'a>) with
    | case, _ -> case.Name

"State" |> Binding.oneWay getUnionCaseName
```

<i:Interaction.Behaviors>

<ic:DataTriggerBehavior

Binding="{Binding StateName, ElementName=GoToCarState}"

ComparisonCondition="NotEqual">

<ic:GoToStateAction

x:Name="GoToCarState"

StateName="{Binding State}"

TargetObject="{x:Bind Path=page}" />

</ic:DataTriggerBehavior>

</i:Interaction.Behaviors>

Как из кода Elmish вызвать код в UI?

```
type PickPrinterEventArgs(selectedPrinter : string) =  
    inherit EventArgs()  
    member _.SelectedPrinter = selectedPrinter  
  
type PickPrinterEventHandler =  
    EventHandler<PickPrinterEventArgs>  
  
type PickPrinterEvent =  
    Event<PickPrinterEventHandler, PickPrinterEventArgs>  
  
type EventHolder (event : PickPrinterEvent) =  
  
    [<CLIEvent>]  
    member val PickPrinterEvent = event.Publish  
  
type Model = {  
    PickPrinterEvent : PickPrinterEvent  
    Events : EventHolder }  
  
"Events" |> Binding.oneWay (fun m -> m.Events)  
m.PickPrinterEvent.Trigger(p, PickPrinterEventArgs("test"))
```

```
public async void ShowPrinterSelectorWindow(  
    object holder,  
    PickPrinterEventArgs args)  
{  
    #if __IOS__  
        var SelectedPrinterUrl = args.SelectedPrinter;  
        var uiPrinterPickerController =
```

<i:Interaction.Behaviors>

<ic:EventTriggerBehavior

EventName="PickPrinterEvent"

SourceObject="{Binding Events}">

<ic:CallMethodAction

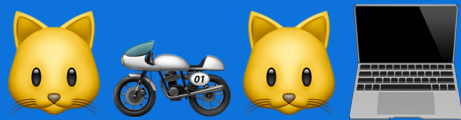
MethodName="ShowPrinterSelectorWindow"

TargetObject="{Binding ElementName=page}" />

</ic:EventTriggerBehavior>

</i:Interaction.Behaviors>

XAML + F# + MVU =



- Вся мощь XAML дизайнера и Blend
- Знакомая модель программирования UWP
- Краткость и защита от ошибок F#
- Односторонний цикл Model-View-Update
- Состояние в модели + visual state в XAML
- Все мыслимые платформы
- Лучший шаблоны промышленного уровня



DOT NEXT

Попробуй!

- `dotnet new`
-i "Elmish.Uno.ProjectTemplates.Dotnet::1.0.0-ci-*"
--nuget-source "https://www.myget.org/F/elmish_uno/api/v3/index.json"
- `dotnet new elmish-unoapp`

Спасибо за внимание



The Secret Circle Solutions

secretcircle.solutions

www.fb.com/scrtcrcl/

GitHub

github.com/xperiandri/

Social

www.fb.com/xperiandri

www.vk.com/xperiandri

www.twitter.com/xperiandri

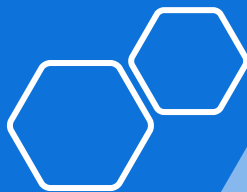
YouTube

[www.youtube.com/user/
andriicsharp](https://www.youtube.com/user/andriicsharp)

Uno Platform

platform.uno/discord

Требуется помощь



- FSharp.Collections.Immutable
- Логотип
- Страница с документацией
- Миграция на Elmish.WPF 4