

Code-Behind

张典

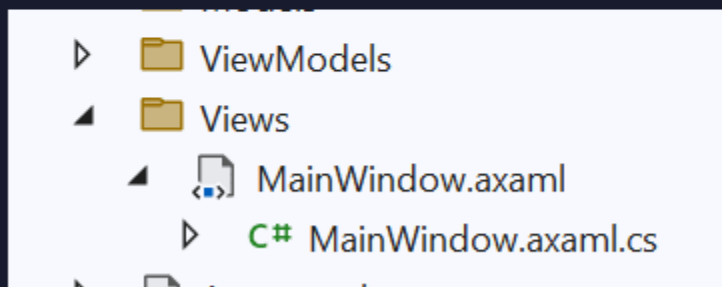
An ordinary Avalonia Developer | IRIHI staff

Code-Behind

Code-behind

除了XAML文件外，大多数Avalonia控件还具有常常用C#编写的 *code-behind* 文件。代码后台文件通常具有 `.axaml.cs` 文件扩展名，并且通常在IDE中显示在XAML文件的下一级。

例如，在Visual Studio的解决方案资源管理器中，您可以看到一个 `MainWindow.axaml` 文件以及它的code-behind文件 `MainWindow.axaml.cs`：



code-behind文件包含一个与XAML文件同名的类。例如：

Code-Behind

代码后置

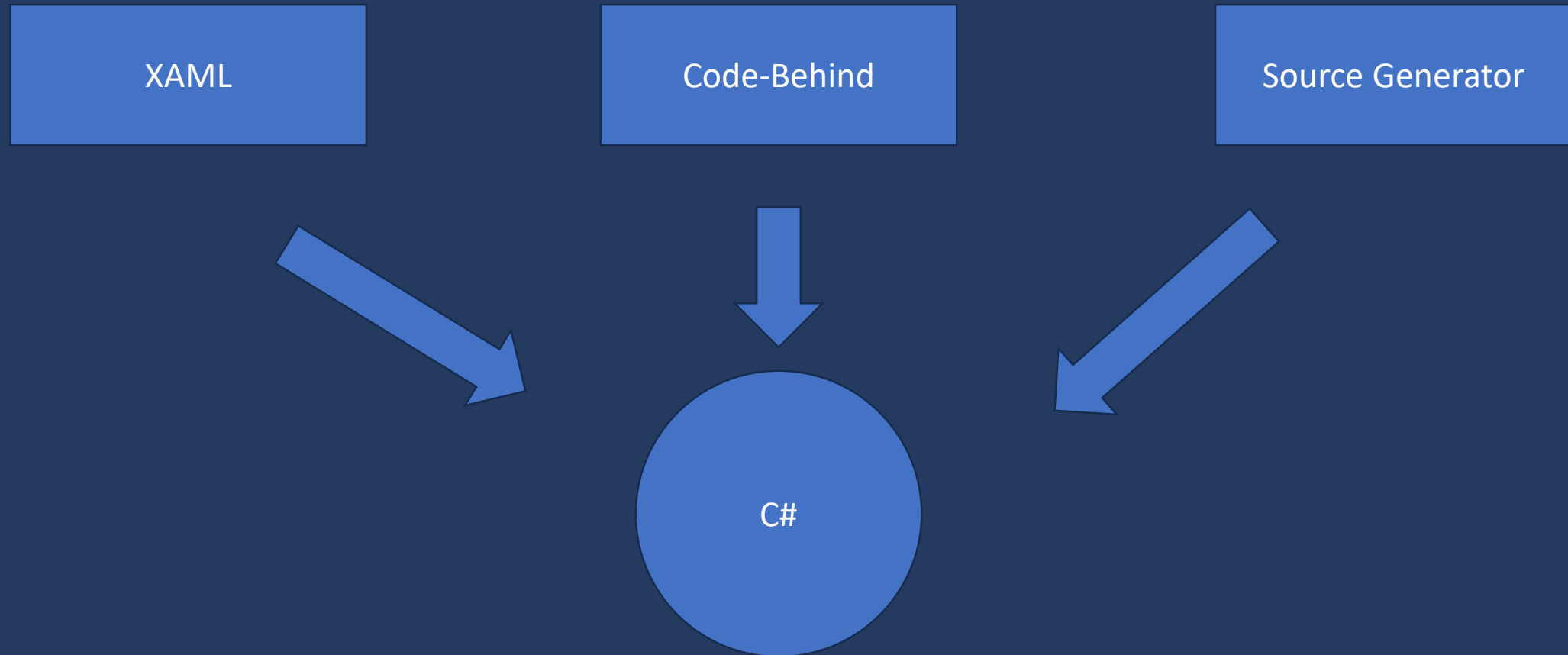
Code-Behind

```
using Avalonia.Controls;

namespace AvaloniaApplication1.Views;

☒ 1 用法   👤 Zhang Dian
public partial class MainWindow : Window
{
    ☒ 1 用法   ◇   👤 Zhang Dian
    public MainWindow()
    {
        InitializeComponent();
    }
}
```

Code-Behind



Code-Behind

```
<Window xmlns="https://github.com/avaloniaui"
        xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
        xmlns:vm="using:AvaloniaApplication1.ViewModels"
        xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
        xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
        mc:Ignorable="d" d:DesignWidth="800" d:DesignHeight="450"
        x:Class="AvaloniaApplication1.Views.MainWindow"
        x:DataType="vm:MainWindowViewModel"
        Icon="/Assets/avalonia-logo.ico"
        Title="AvaloniaApplication1">
```

Code-Behind

```
using Avalonia.Controls;

namespace AvaloniaApplication1.Views;

1 用法  Zhang Dian
public partial class MainWindow : Window
{
    1 用法  Zhang Dian
    public MainWindow()
    {
        InitializeComponent();
    }
}
```

Code-Behind

```
public void InitializeComponent(bool loadXaml = true, bool attachDevTools = true)
{
    if (loadXaml)
    {
        AvaloniaXamlLoader.Load(this);
    }

#if DEBUG
    if (attachDevTools)
    {
        this.AttachDevTools();
    }
#endif
}
```


Code-Behind

```
<Window xmlns="https://github.com/avaloniaui"
  xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
  xmlns:vm="using:AvaloniaApplication1.ViewModels"
  xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
  xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
  mc:Ignorable="d" d:DesignWidth="800" d:DesignHeight="450"
  x:Class="AvaloniaApplication1.Views.MainWindow"
  x:DataType="vm:MainWindowViewModel"
  Icon="/Assets/avalonia-logo.ico"
  Title="AvaloniaApplication1">
```

XAML

```
using Avalonia.Controls;

namespace AvaloniaApplication1.Views;

public partial class MainWindow : Window
{
    public MainWindow()
    {
        InitializeComponent();
    }
}
```

Code-Behind

```
public void InitializeComponent(bool loadXaml = true, bool attachDevTools = true)
{
    if (loadXaml)
    {
        AvaloniaXamlLoader.Load(this);
    }

    #if DEBUG
    if (attachDevTools)
    {
        this.AttachDevTools();
    }
    #endif
}
```

Source Generator

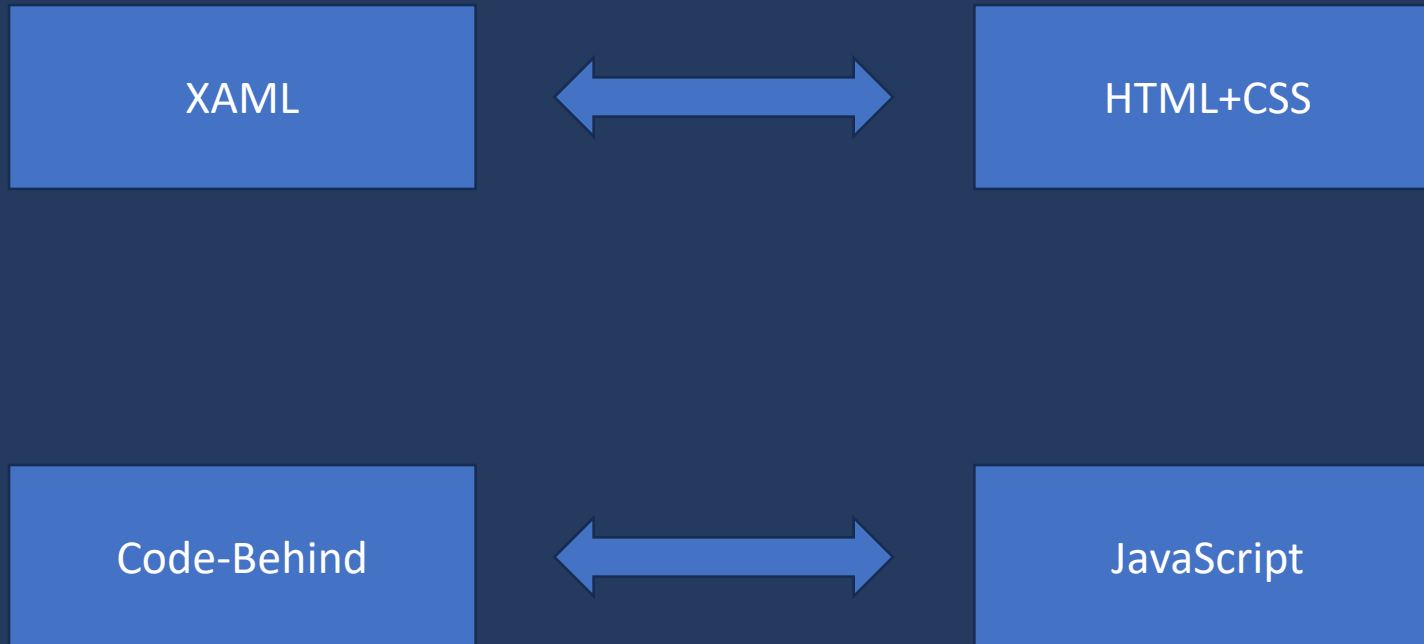
```
public class MainWindow : Window
{
    [CompilerGenerated]
    private static Action<object> !XamlILPopulateOverride;

    public MainWindow()
    {
        base..ctor();
        this.InitializeComponent(true, true);
    }

    [GeneratedCode("Avalonia.Generators.NameGenerator.InitializeComponentCodeGenerator", "11.3.0.0")]
    [ExcludeFromCodeCoverage]
    public void InitializeComponent(bool loadXaml = true, bool attachDevTools = true)
    {
        if (!loadXaml)
            MainWindow.!XamlILPopulateTrampoline(this);
        if (!attachDevTools)
            return;
    }
}
```

C#

Code-Behind

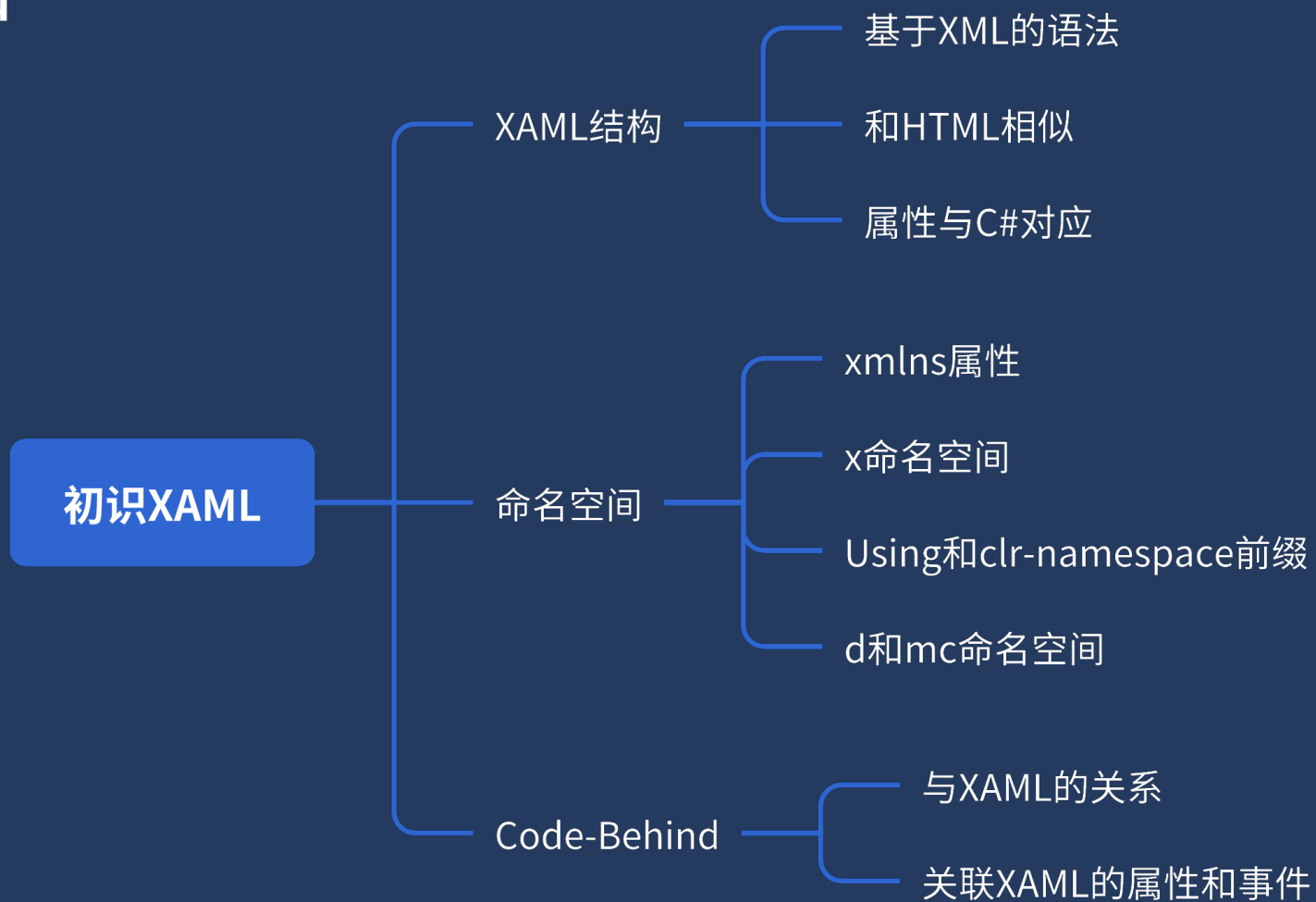


Code-Behind

MVVM

Model-View-ViewModel

章节小结



Thank you.