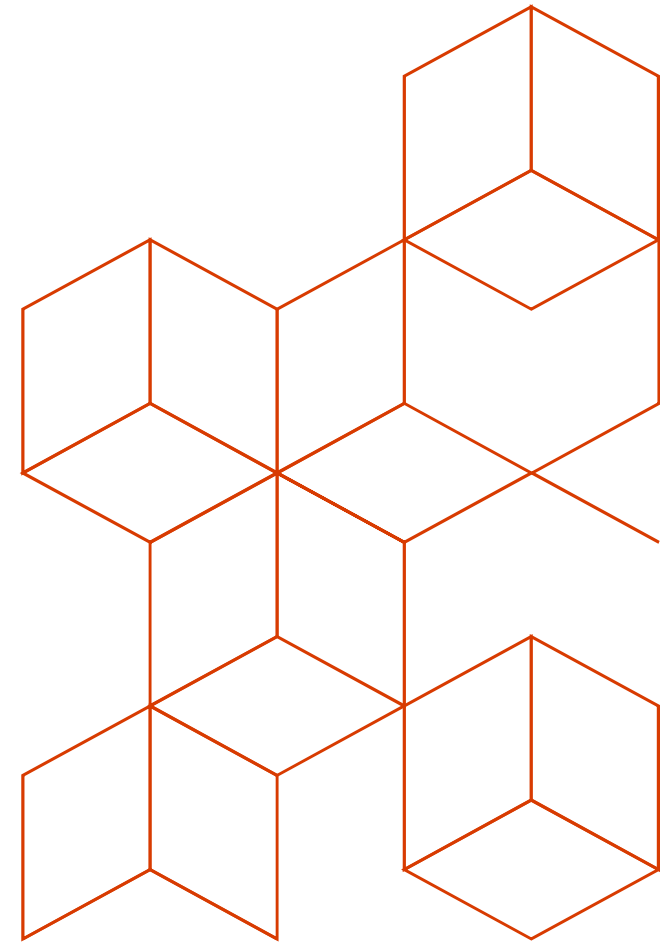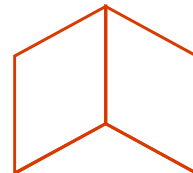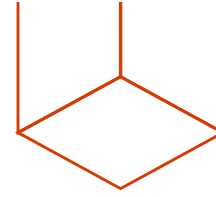# XAML Islands

Vinoth

r.vinoth@live.com | @VinothRajendran

Technical specialist - Bosch

Agenda

Windows 10 features

Journey of Modern App

XAML Islands

Demo

"Will my LOB desktop Apps just run in Windows 10?"

"…my goal is to reduce App packaging and deployment costs…"

"…Upgrading framework for my solution is expensive… there are many components involved…"

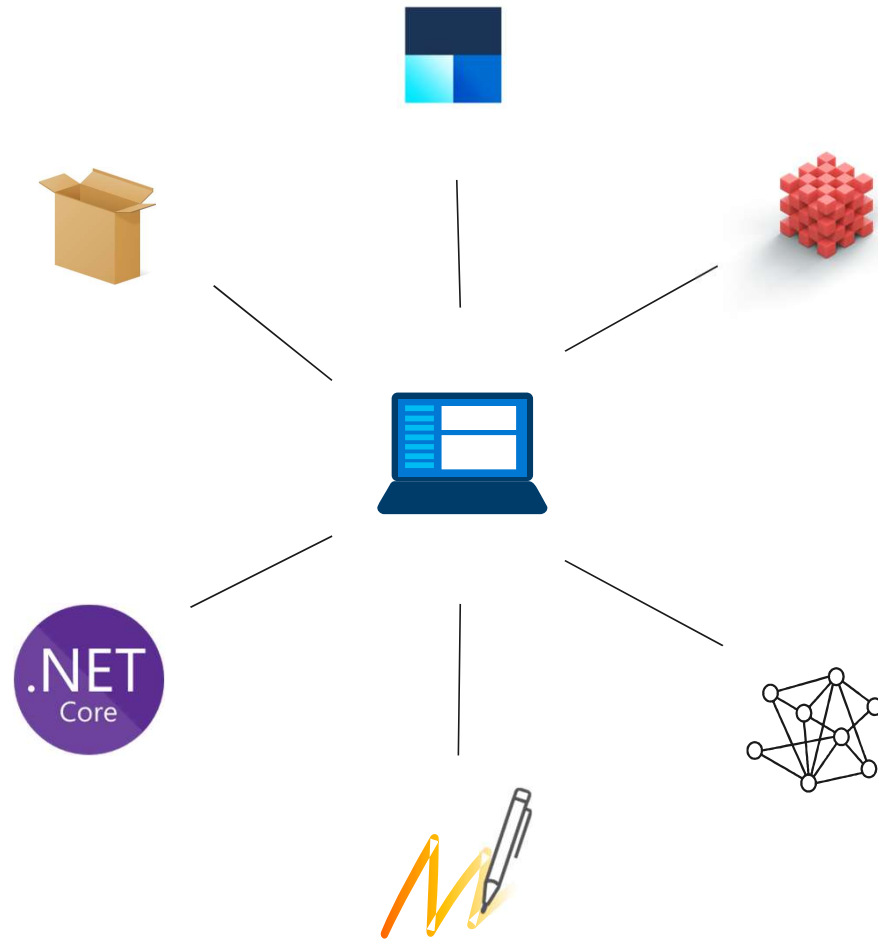"…we have many LOB Apps that rely on WinForms and WPF….what should I do with them?"

"…My users need Apps that work best with keyboard and mouse…"

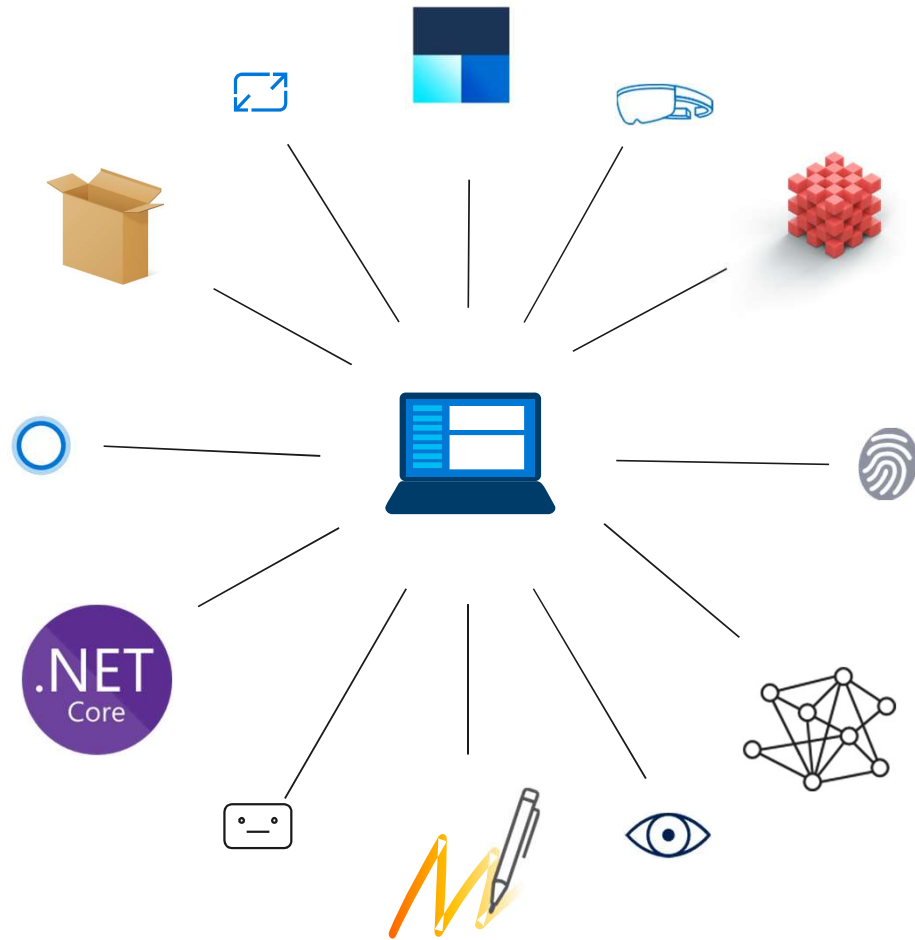"…I want to enhance my App by adding windows 10 but don't want to rewrite my App…"

Windows
Enterprise un
Unka power ed un
App container
App containers
Desktop Apps
Extension App
Learning

And more...

Journey of Modern App

Modern App Platform

Win32 Apps`

Desktop bridge
Win32 Apps to the Microsoft Store

Windows 10 APIs
Call Windows 10 APIs into the Win32 apps

XAML Islands
Windows 10 Controls in Native apps

WPF | **WinForms** | **C++**

2.4M Developers building Desktop apps
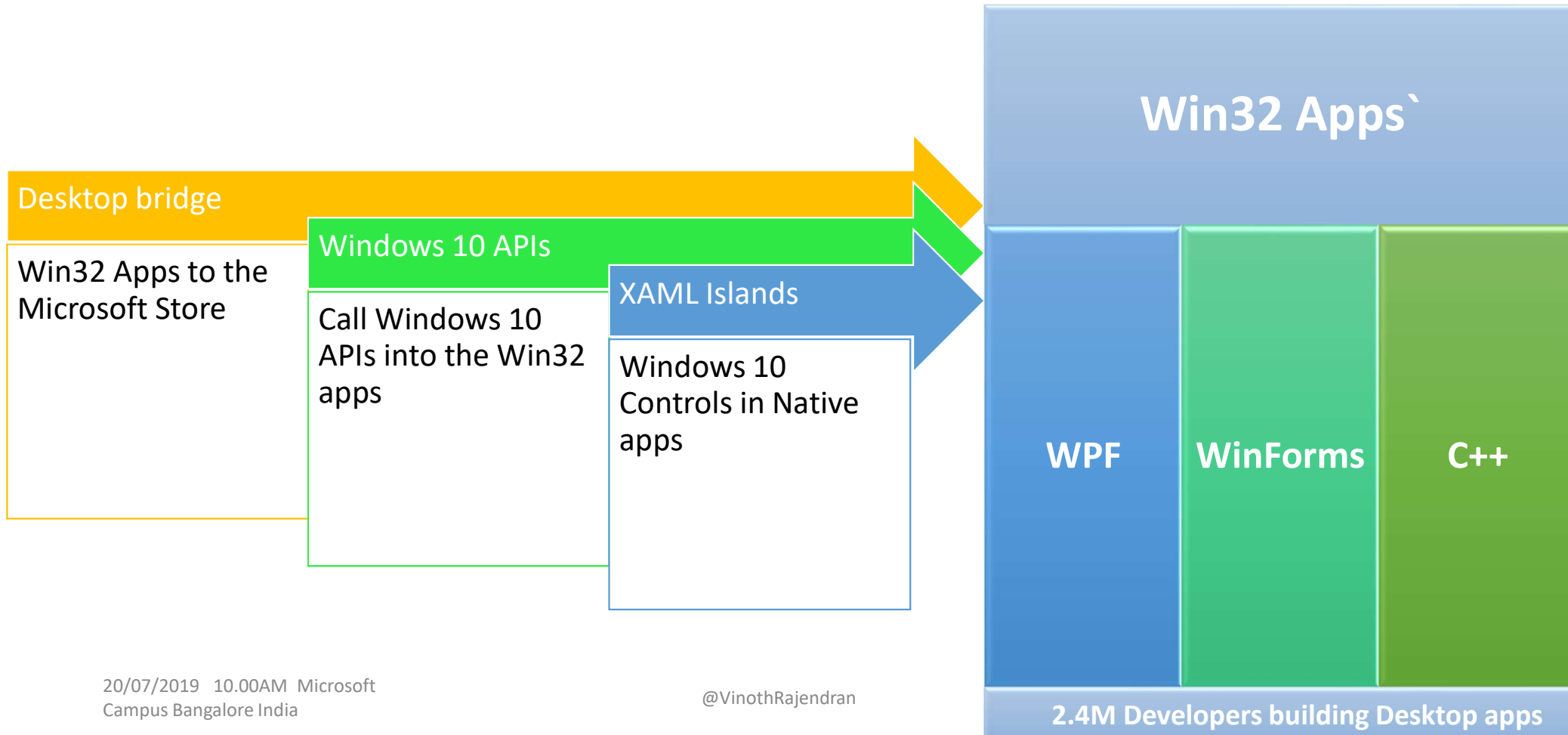
XAML Islands

XAML Islands

- Enable .NET and native Win32 applications to host UWP Controls
- New Controls & Fluent Design System
- XAML Islands V1 has released on Windows 10 -> 1903 (May update)
- XAML Islands V2 will be release in First Half 2020

XAML Islands

- Enable .NET and native Win32 applications to host UWP Controls
- New Controls & Fluent Design System
- XAML Islands V1 has released on Windows 10 -> 1903 (May update)
- XAML Islands V2 will be release in First Half 2020

XAML Islands Requirements

| 1903 | 10.0.18362 | IDE | 3.0 | 4.7.2 | 6.0 |

XAML Islands

- Enable .NET and native Win32 applications to host UWP Controls
- New Controls & Fluent Design System
- XAML Islands V1 has released on Windows 10 -> 1903 (May update)
- XAML Islands V2 will be release in First Half 2020

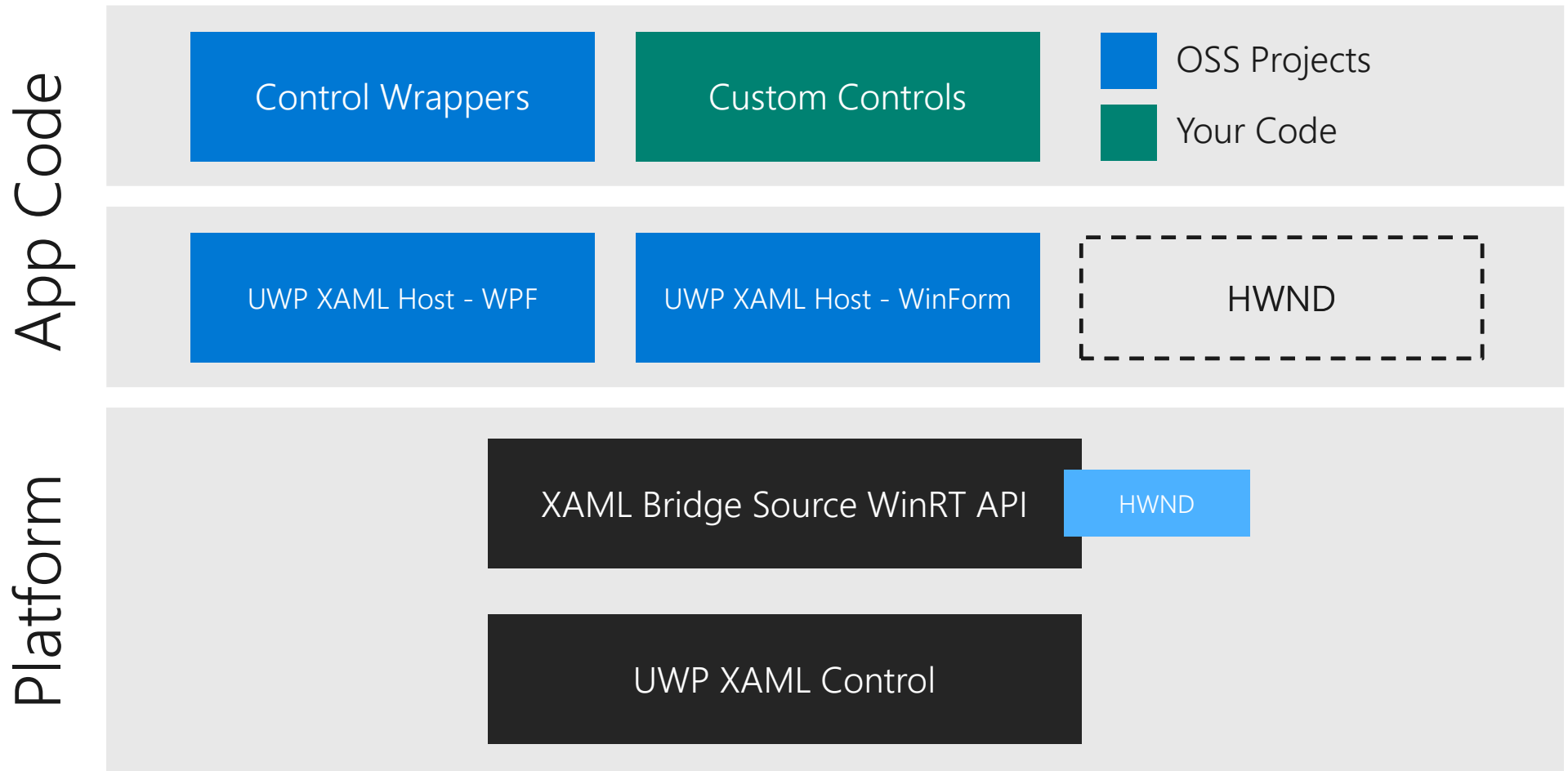# XAML Islands Requirements

| 1903 | 10.0.18362 | IDE | 3.0 | 4.7.2 | 6.0 |

Optional Package

The architecture

App Code

| Control Wrappers | Custom Controls | OSS Projects |
| | | Your Code |

| UWP XAML Host - WPF | UWP XAML Host - WinForm | HWND |

Platform

XAML Bridge Source WinRT API | HWND

UWP XAML Control

UWP XAML hosting API
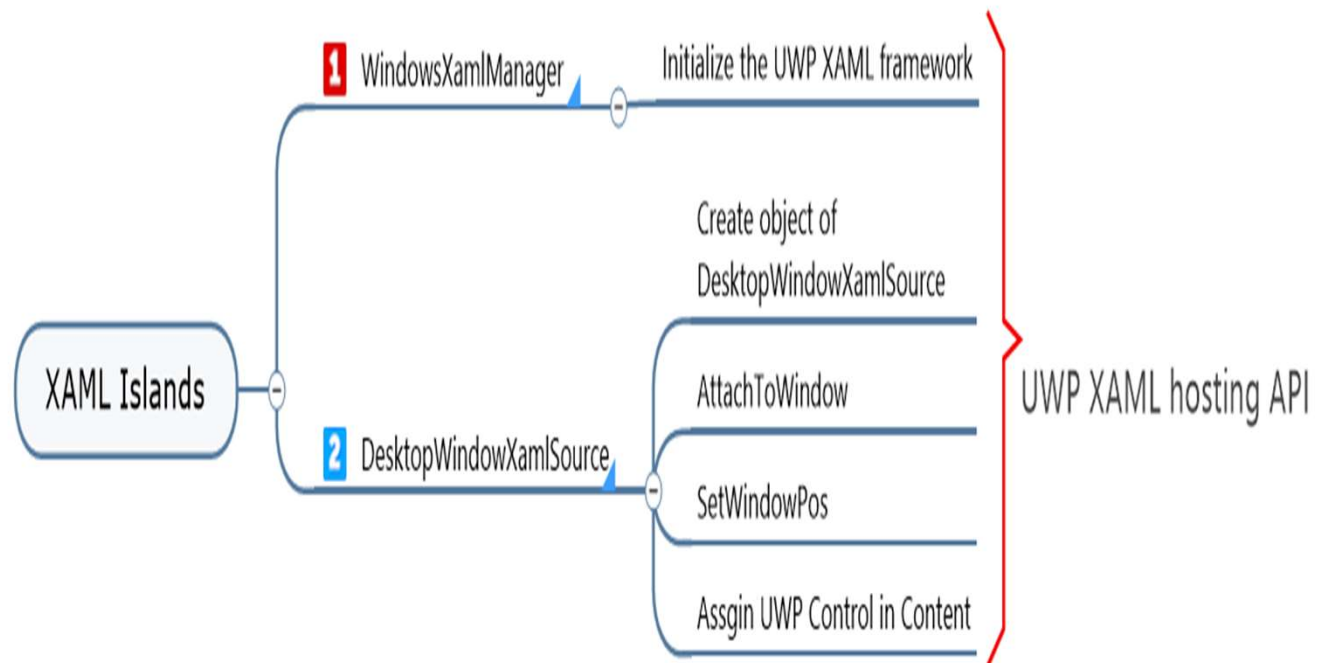
@VinothRajendran

Windows Community Support

UWP XAML hosting API
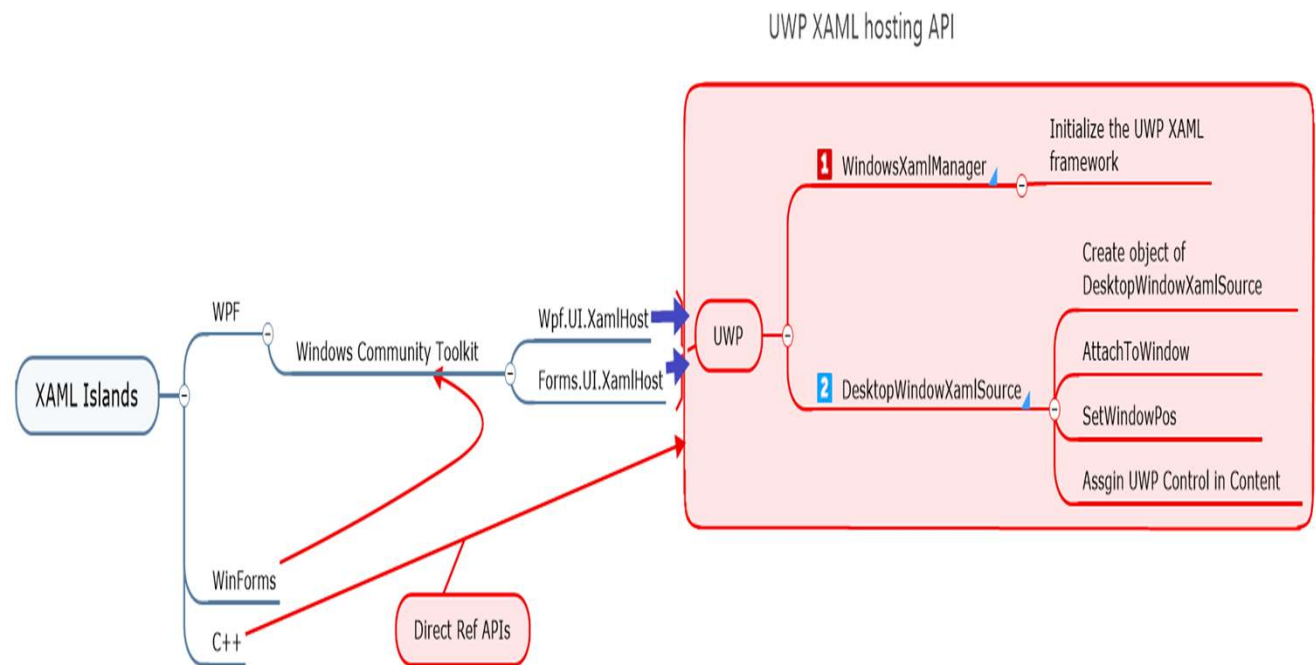
@VinothRajendran

Windows Community Toolkit wrappers – 1st party control

Your WPF/WinForms app

WPF/WinForms Control Wrappers

Xaml Controls Wrappers
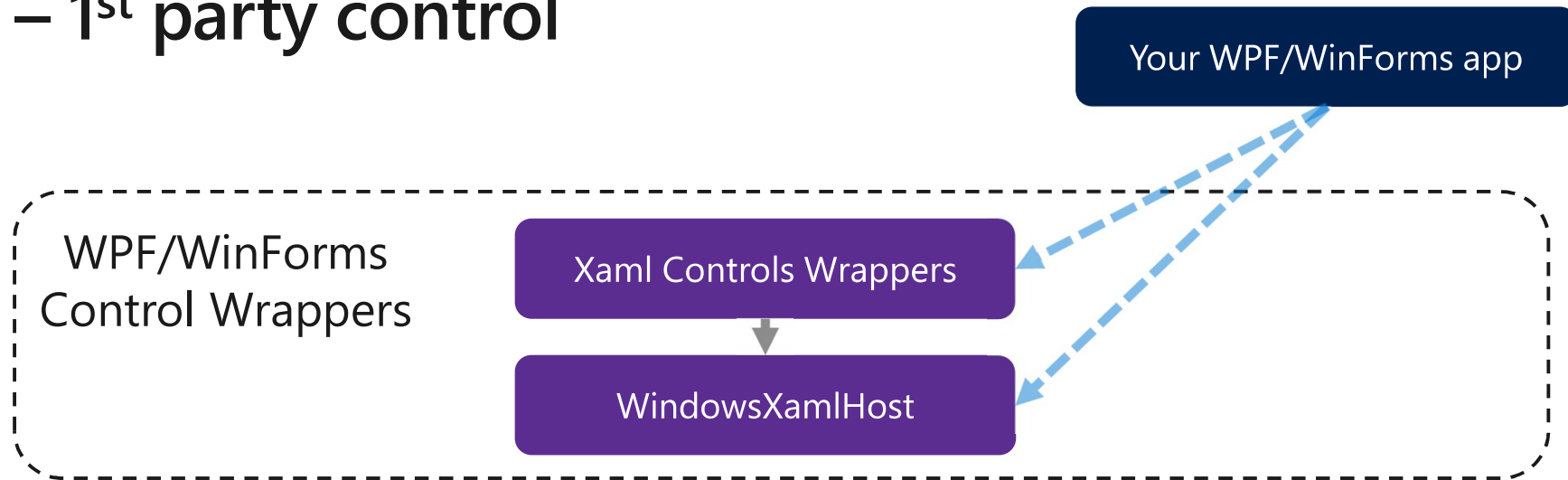
WindowsXamlHost

- No WinRT concepts
- NuGet Packages for Windows Forms and WPF
- Designer Support

XAML Control Wrappers

WindowsXamlHost
- Custom Xaml content

WebView (EdgeHtml)
- Html 5 support
- Most requested feature in UserVoice
- No using DWXS API
- Works on April 2018 Update

Map Control
- Tilted & Rotated views
- Offline Support
- 3D data
- Streetside
- Traffic

MediaPlayerElement
- Closed Captioning
- Smooth Streaming
- Adapted bitrates
- Additional codecs support
- Better power utilization

SwapChainPanel
- Native DirectX11 support
- Native DirectX12 support
- Multithreading support
- High DPI
- HDR rendering

Ink Canvas
- Super-wet Ink
- Ink Toolbar
- Text recognition
- Shape recognition
- Store and retrieve ink

Graph
- Microsoft Graph
- Microsoft 365
  - Azure AAD
  - Office 365
  - Enterprise Mobility
  - Windows 10

Legend:
- Shipped in Preview (green)
- In development (blue)

Windows Community Toolkit – WindowsXamlHost
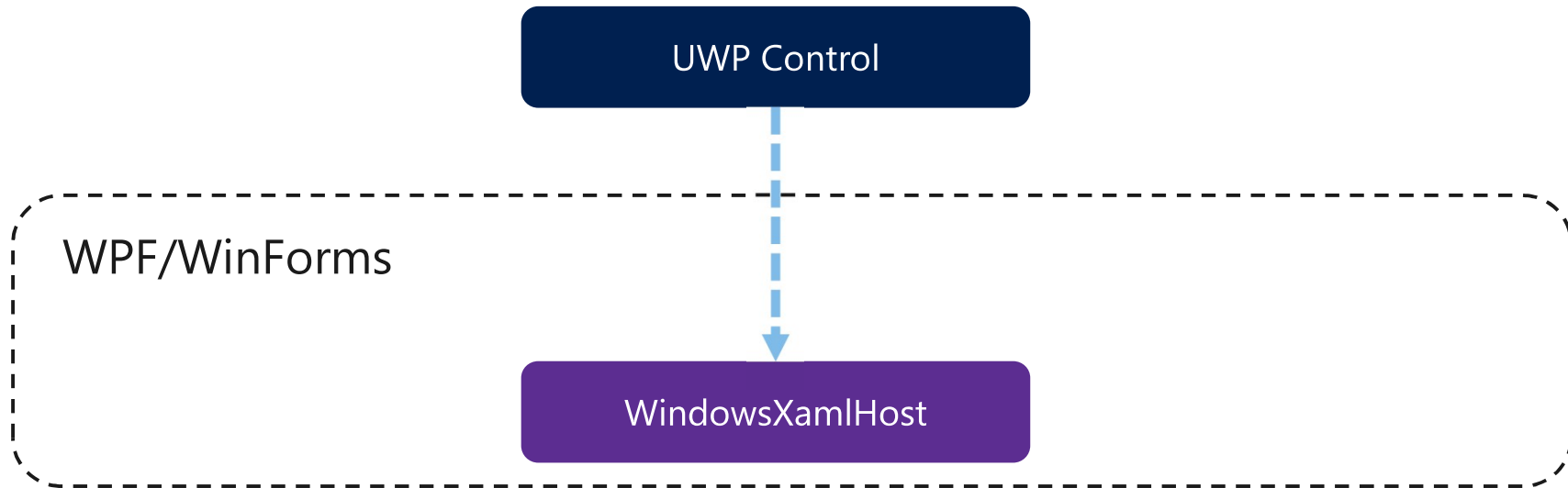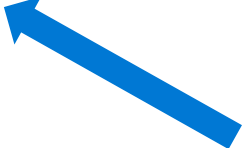
- Host for any UWP Control
- Dedicated NuGet package:
  - WPF: https://www.nuget.org/packages/Microsoft.Toolkit.Wpf.UI.XamlHost/
  - Windows Forms: https://www.nuget.org/packages/Microsoft.Toolkit.Forms.UI.XamlHost/
- If you are using 1st party wrapped controls, the package is already installed since it's a dependency

Using the control

```
<xamlhost:WindowsXamlHost
InitialTypeName="Windows.UI.Xaml.Controls.NavigationView"
ChildChanged="WindowsXamlHost_ChildChanged" />
```

The full signature of the control you want to render

You subscribe to this event to handle interaction with the control and to customize it

Using the control

```csharp
private void WindowsXamlHost_ChildChanged(object sender, EventArgs e)
{
    WindowsXamlHost windowsXamlHost = (WindowsXamlHost)sender;

    var navView = (Windows.UI.Xaml.Controls.NavigationView)windowsXamlHost.Child;

    if (navView != null)
    {
        navView.ItemInvoked += (obj, args) =>
        {

        };

        navView.MenuItems.Add(new NavigationViewItem() { Content = "Home" });
        navView.MenuItems.Add(new NavigationViewItem() { Content = "People" });
    }
}
```

This property has a reference to the hosted control

Event handling

Properties customization

DEMO

@VinothRajendran

WPF – GUI