

Windows Presentation Foundation

WPF

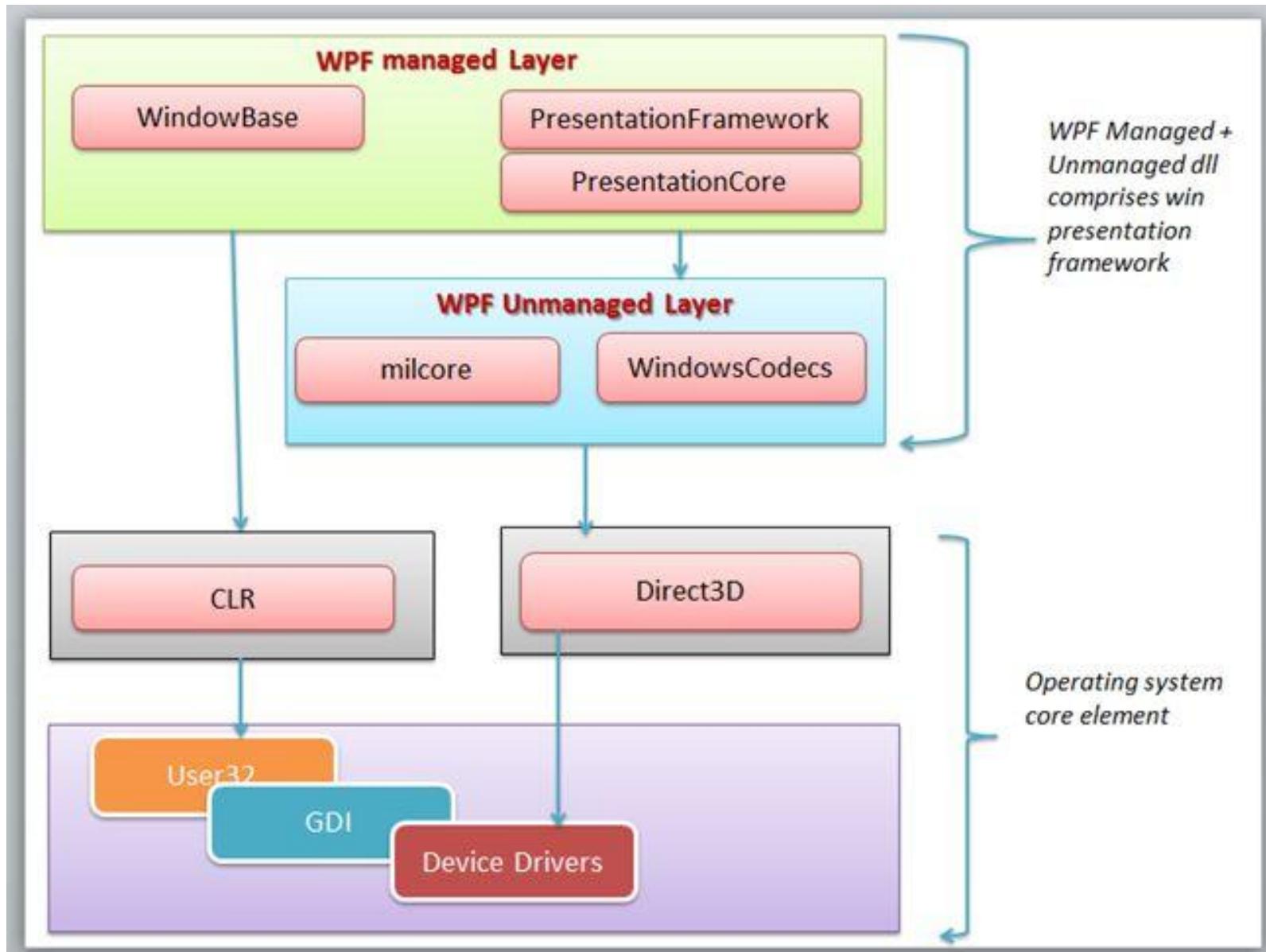
Overview

- WPF stands for Windows Presentation Foundation.
- It is a powerful framework for building Windows applications.
- WPF was first introduced in .NET framework 3.0 version, and then so many other features were added in the subsequent .NET framework versions

WPF Architecture

- Before WPF, the other user interface frameworks offered by Microsoft such as MFC and Windows forms, were just wrappers.
- WPF is more than just a wrapper.
- It is a part of the .NET framework.
- It contains a mixture of managed and unmanaged code.

WPF Architecture



WPF – Advantages

- In the earlier GUI frameworks, there was no real separation between how an application looks like and how it behaved. Both GUI and behavior was created in the same language.
- In WPF, UI elements are designed in XAML while behaviors can be implemented in procedural languages such C# and VB.Net.

WPF – Features

Feature	Description
Control inside a Control	Allows to define a control inside another control as a content.
Data binding	Mechanism to display and interact with data between UI elements and data object on user interface.
Media services	Provides an integrated system for building user interfaces with common media elements like images, audio, and video.
Templates	In WPF you can define the look of an element directly with a Template
Animations	Building interactivity and movement on user Interface
Alternative input	Supports multi-touch input on Windows 7 and above.
Direct3D	Allows to display more complex graphics and custom themes

Hello World Example

XAML Overview

- One of the first things you will encounter while working with WPF is XAML.
- XAML stands for Extensible Application Markup Language.
- It's a simple and declarative language based on XML.

Basic Syntax

```
<Window x:Class = "Resources.MainWindow"
    xmlns = "http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x = "http://schemas.microsoft.com/winfx/2006/xaml"
    Title = "MainWindow" Height = "350" Width = "525">

    <Grid>

    </Grid>

</Window>
```

Basic Syntax

Information	Description
<Window	It is the opening object element or container of the root.
x:Class = "Resources.MainWindow"	It is a partial class declaration which connects the markup to the partial class code defined behind.
xmlns = "http://schemas.microsoft.com/winfx/2006/xaml/presentation" ↴	Maps the default XAML namespace for WPF client/framework
xmlns:x = "http://schemas.microsoft.com/winfx/2006/xaml" ↴	XAML namespace for XAML language which maps it to x: prefix
>	End of object element of the root
<Grid> </Grid>	It is starting and closing tags of an empty grid object.
</Window>	Closing the object element

WPF Controls

- Windows Presentation Foundation (WPF) allows developers to easily build and create visually enriched UI based applications.
- The classical UI elements or controls in other UI frameworks are also enhanced in WPF applications.
- All of the standard WPF controls can be found in the Toolbox which is a part of the `System.Windows.Controls`.
- These controls can also be created in XAML markup language.

WPF Controls

Sr. No.	Controls & Description
1	Button  A control that responds to user input
2	Calendar  Represents a control that enables a user to select a date by using a visual calendar display.
3	CheckBox  A control that a user can select or clear.
4	ComboBox  A drop-down list of items a user can select from.
5	ContextMenu  Gets or sets the context menu element that should appear whenever the context menu is requested through user interface (UI) from within this element.

WPF - Layouts

Sr. No.	Panels & Description
1	<p>Stack Panel ↗</p> <p>Stack panel is a simple and useful layout panel in XAML. In stack panel, child elements can be arranged in a single line, either horizontally or vertically, based on the orientation property.</p>
2	<p>Wrap Panel ↗</p> <p>In WrapPanel, child elements are positioned in sequential order, from left to right or from top to bottom based on the orientation property.</p>
3	<p>Dock Panel ↗</p> <p>DockPanel defines an area to arrange child elements relative to each other, either horizontally or vertically. With DockPanel you can easily dock child elements to top, bottom, right, left and center using the Dock property.</p>
4	<p>Canvas Panel ↗</p> <p>Canvas panel is the basic layout panel in which the child elements can be positioned explicitly using coordinates that are relative to the Canvas any side such as left, right, top and bottom.</p>
5	<p>Grid Panel ↗</p> <p>A Grid Panel provides a flexible area which consists of rows and columns. In a Grid, child elements can be arranged in tabular form.</p>