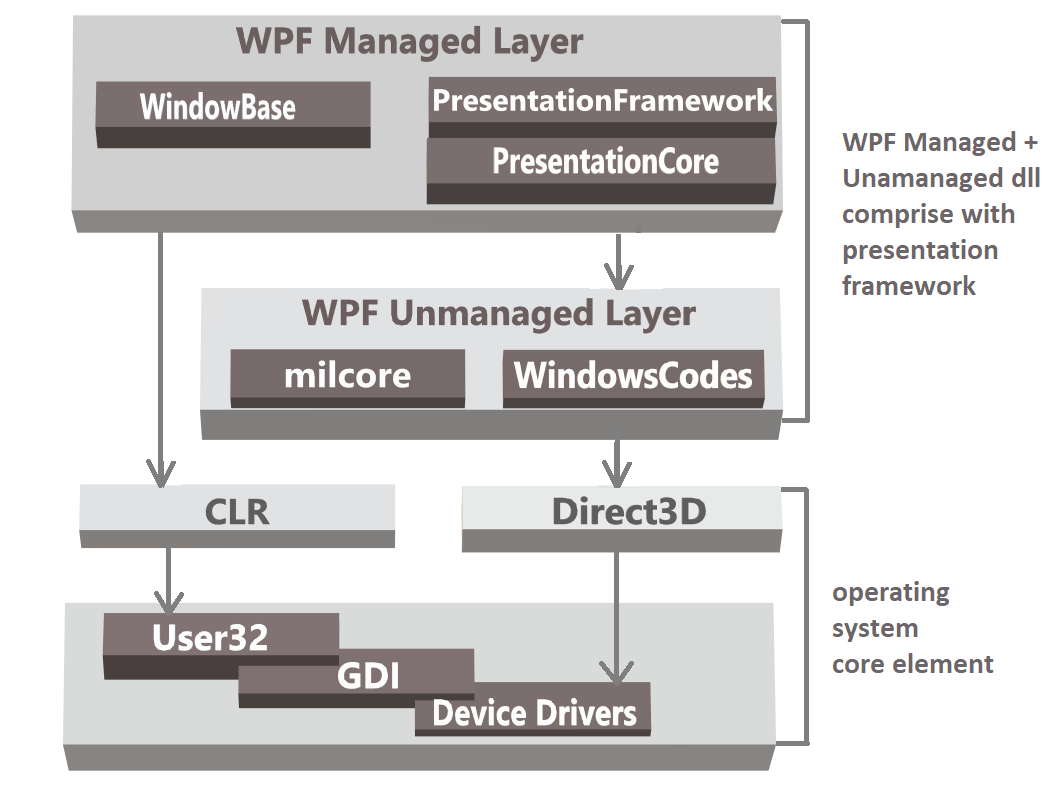
**WPF in c#.NET**

WPF stands for Windows Presentation Foundation. WPF support graphics, animation, multimedia and documents to develop high-end graphic rich desktop application without any Application Programming Interfaces(APIs). WPF allow developer and designer to share and combine their work on single platform. WPF use XAML language which is based on XML(Extensible Markup Language). XAML allow designer to easily and quickly define and describe UI element.

**Why It Is Used?**

With Windows forms application it is not possible to use Advance graphics and window form will always be inefficient in comparison to WPF.

**Architecture of WPF**

****

WPF contain both managed and unmanaged component. Managed components are managed and executed by CLR(Common Language Runtime). As managed component are executed by CLR they provide type-safety, garbage collection, security and cross- language integration.

**PresentationFramework**

Presentation framework provide different classes that we can use for presentation and appearance of control, layout and animations.

**PresentationCore**

Presentation Core component provide some of the most common class as text manipulation animation, 2D, brushes, etc. and these classes provide some of the basic functionality as event, message, properties.

**WindowBase**

This component provide functionality of dependency, thread and type c convert.

**milcore**

milecore is unmanaged component of WPF which make animation, render3D scene, 2D and 3D graphics possible.

**Feature of WPF**

* Support for touch-enabled application.
* Control’s appearance is enhanced by visual state manager(VSM)
* Support for XAML browse application(XBAP’s)
* XAML as the Markup: XAML goes in some ways beyond XML because it is simple to understand.
* It provide dependency properties such as data binding, styling, triggers, and animation.
* Styles & Triggers are very useful in CSS
* Control templates: Separation of appearance and behavior
* Integration of 2D and 3D

**Disadvantage**

* It has less controls as compare to web forms
* Not Compatible with older version

**Types of WPF Application**

* + Windows Application(Standalone application)
  + XAML Browser Application(XBAPs)

**Windows Application(Standalone application)**

* Windows Application is similar to Windows Form Application
* We can open multiple windows at any given time, and there is no built-in sense of navigation or history
* WPF windows are objects of the **Window** class, which have manyAllo properties which help into working with windows.
* AllowsTransparency
* Height
* Width
* Icon
* Name
* Title
* When We Use windows application?
  + We use windows application for a user experience that closely resembles a traditional Windows Form application.
  + In menu driven, multiwindow application that combines the rich functionality of a desktop with the rich UI.

**XBAPs Application**

* XAML Browser Application are application that are hosted on web server and run on web browser.
* HTML is used to script XBAP application.
* XBAPs application are similar to Navigation Applications, but they are designed to run in Internet Explorer.
* XABPs are not installed on computer, so if user is offline then user cannot access XBAPs.
* In earlier version, XBAPs was provided with partial-trust environment however now it is provided with full trust permission.

When we Use XBAPs application?

* + When you want to deploy the application to a web server, which will start from the hyperlink. It will make it easily accessible to a large-scale audience.
  + If your application does not require access to system resources, XBAP might be good choice.

Happy learning and wish you had a great day!!!