

XAML - OVERVIEW

http://www.tutorialspoint.com/xaml/xaml_overview.htm

Copyright © tutorialspoint.com

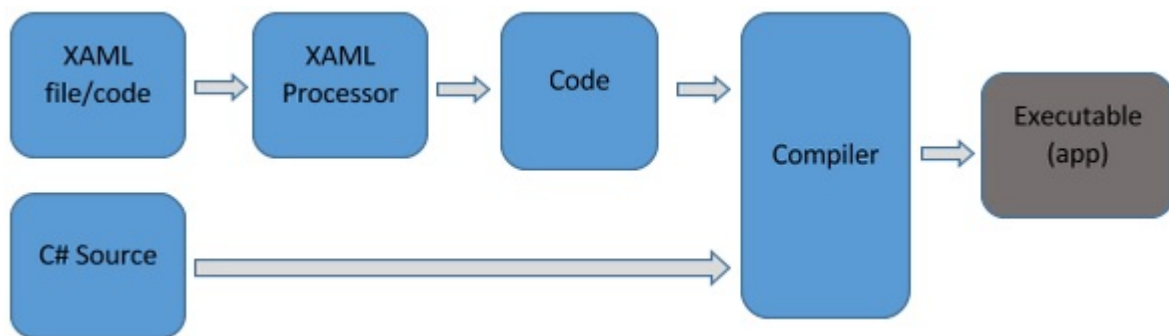
XAML stands for Extensible Application Markup Language. It's a simple and declarative language based on XML.

- In XAML, it's very easy to create, initialize, and set properties of an object with hierarchical relations.
- It is mainly used for designing GUIs.
- It can be used for other purposes as well, e.g., to declare workflow in Workflow Foundation.

XAML can be used in different platforms such as WPF *WindowsPresentationFoundation*, Silverlight, Mobile Development, and Windows Store App. It can be used across different .Net framework and CLR *commonlanguageruntime* versions.

How XAML Works

XAML is a **declarative** language in the sense it defines the **WHAT** and **HOW** you want to do. XAML processor is responsible for the **HOW** part to find out. Let's have a look at the following schema. It sums up the XAML side of things –



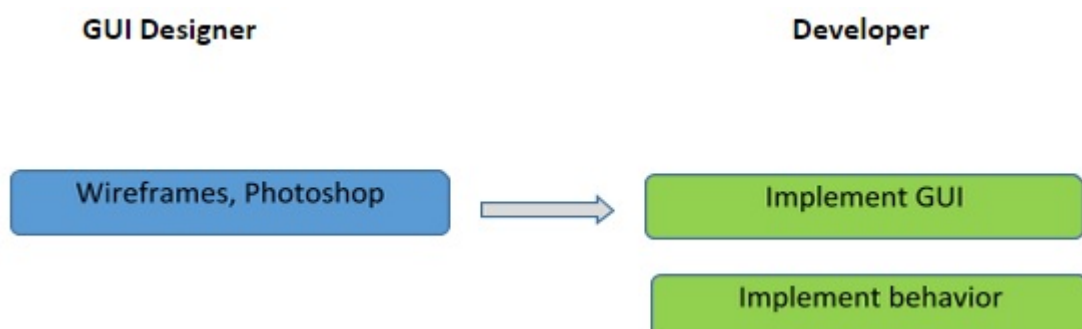
The figure illustrates the following actions –

- The XAML file is interpreted by a platform-specific XAML processor.
- The XAML processor transforms the XAML to internal code that describes the UI element.
- The internal code and the C# code are linked together through partial classes definitions and then the .NET compiler builds the app.

Advantages of XAML

One of the longstanding problems that all of us face with GUI design can be solved by using XAML. It can be used to design UI elements in Windows Forms applications.

In the earlier GUI frameworks, there was no real separation between how an application looks like and how it behaves. Both the GUI and its behavior were created in the same language, e.g. C# or VB.net, which would require more effort from the developer to implement both the UI and the behavior associated with it.



With XAML, it is very easy to separate the behavior from the designer code. Hence, the XAML programmer and the designer can work in parallel. XAML codes are very easy to read and understand.

