*XAML* (*Extensible Application Markup Language*, *расширяемый язык разметки приложений*) – это язык разметки, предлагающий основанный на XML синтаксис для представления логического дерева объектов .NET.

Существует несколько подмножеств XAML:

* *WPF XAML* включает элементы, описывающие содержимое WPF вроде векторной графики и элементов управления;
* *XPS XAML* ‑ часть WPF XAML, определяющая XML-представление форматированных электронных документов. Эта часть опубликована как отдельный стандарт *XML Paper Specification* (*XPS*);
* *Silverlight XAML* ‑ подмножество WPF XAML, предназначенное для Silverlight-приложений. Silverlight ‑ это межплатформенный браузерный подключаемый модуль, позволяющий создавать веб-содержимое с двумерной графикой, анимацией, аудио и видео;
* *WF XAML* включает элементы, описывающие содержимое Windows Workflow Foundation.

Документ XAML записан в формате XML. Это означает, что имена элементов XAML чувствительны к регистру, нужна правильная вложенность элементов, некоторые символы требуют особого обозначения (например, &amp; ‑ это символ &).

*Объектные элементы XAML* описывают объект некоторого типа .NET (класса или структуры). Имя объектного элемента совпадает с именем типа. Необходимо, чтобы тип обладал открытым конструктором без параметров. Типы .NET обычно вложены в пространства имён. В XAML пространство имён .NET ставится в соответствие пространству имён XML. При необходимости указывается сборка, содержащая пространство имён:

xmlns:*префикс*="clr-namespace:*ПространствоИмён*;assembly=*ИмяСборки*"

Описание объекта подразумевает задание значений его свойств. В XAML для этого применяются атрибуты, элементы свойств и содержимое элемента. При использовании атрибутов указывается имя свойства и значение свойства в виде строки. Анализатор XAML применяет для преобразования строки в значение свойства специальные *конвертеры типов*. Набор стандартных конвертеров достаточно богат. При необходимости можно разработать собственный конвертер, используя базовый класс TypeConverter. *Элемент свойства* – это дочерний элемент объектного элемента, имеющий вид <*TypeName.Property*>. Содержимое элемента свойства рассматривается как значение свойства.

Механизм *расширений разметки* (*markup extensions*) позволяет вычислять значение свойства при выполнении приложения. Встретив в XAML расширение разметки, анализатор генерирует код, который создаёт объект расширения разметки и вызывает особый метод объекта для получения значения.

Анализатор XAML генерирует код, выполняющий по документу XAML создание и настройку объектов. *Действия* с объектами описываются в отдельном *классе кода*. Чтобы сослаться на объект в коде, объект должен иметь имя. Для этого в пространстве имён анализатора XAML определён атрибут Name. Чтобы связать класс кода с документом XAML используется атрибут Class из пространства имён анализатора XAML. Этот атрибут применяется только к корневому элементу и содержит имя класса, являющегося наследником класса корневого элемента:

<Window x:Class="WpfApplication1.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml">

<Button x:Name="okButton" Content="Click me!" />

</Window>

В заключение заметим, что WPF XAML описывает *логическое дерево элементов*. Наряду с этим термином используется понятие *визуального дерева элементов*. Визуальное дерево составляют отображаемые объекты. Некоторые одиночные логические объекты распадаются на несколько визуальных составляющих, так как стоятся из нескольких визуальных примитивов. Например, любое окно Window включает визуальный примитив Border, который содержит примитив AdornerDecorator c объектами ContentPresenter и AdornerLayer.