

1.0 Introduction to MS.Net

1.1 History of .Net:

Sometime in the July 2000, Microsoft announced a completely new software development framework for Windows called .NET in the Professional Developer Conference (PDC). Microsoft also released PDC version of the software for the developers to test.

Microsoft started development on the .NET Framework in under the name of Next Generation Windows Services (NGWS). By late 2000, the first beta versions of .NET 1.0 were released.

1.2 .Net SDK's:

IDE	FRAMEWORK
VS.NET 2002	1.0
VS.NET 2003	1.1
VS.NET 2005	2.0
VS.NET 2008	2.0, 3.0, 3.5
VS.NET 2010	2.0, 3.0, 3.5, 4.0
VS.NET 2012	2.0, 3.0, 3.5, 4.5
VS.NET 2013	2.0, 3.0, 3.5, 4.5,4.5.1,4.5.2
VS.NET 2015	2.0, 3.0, 3.5, 4.5,4.6

Note: When single IDE supports multiple fx, this is known as “Multi Target Support”

1.3 Flavors of .Net:

MS.Net has various environments for different requirements. We have stated here some of major application environments.

Console Application: These refer to traditional DOS kind of applications like batch scripts. This environment is used to develop application having core functionality and need not require any GUI. It is require character based command. These type of applications are lightweight hence runs fast.

E.g., ping utility in windows.

a) **Windows Form based /Windows/Desktop Applications:**

These refer to traditional rich client applications. It depends on underlying O.S. for visual style or theme. It has rich GUI. Generally, these types of applications used to implement large functionality in user-friendly manners. e.g. MS-Word, NotePad, Tally etc.

b) **ASP.NET /Web applications:**

MS.Net has made drastic changes in web application than windows application, the way web page design and program.MS.Net has increased productivity of web programmer by using ASP.Net.It is very user friendly environment for new users.ASP.Net change the face of web sites the way it looks and functions.

c) **WPF Application:**

WPF is a powerful new infrastructure based on DirectX, the hardware-accelerated graphics API that's commonly used in cutting-edge computer This means that you can use rich graphical effects without incurring the performance overhead that you'd suffer with Windows Forms. In fact, you even get advanced features such as support for video files and 3-D content.

1.4 Features of .Net:

a) **Rich Functionality out of the box:**

MS.Net framework provides rich set of libraries to achieve basic, intermediate and advanced functionalities. You need not to go for third party libraries oftenly.SDK also has more than single framework with IDE, which gives us backward and forward backward compatibility.

b) **Easy Development of Web Application:**

Before ASP.Net Classic ASP was used to develop web application but there was certain drawback of classic ASP that It was quite cumbersome

job to design web application using HTML tags.ASP.Net Provides various server side and client side controls in toolbox to drag and drop on pages. Classic ASP pages interpret and ASP.Net pages compile rather than interpretation, it gives performance to web pages.

c) **OOP's Support :**

MS.Net has more than 60 framework compatible languages and all are OOP.This approach gives programmer code reusability, robustness and security.OOP makes programmer task more comfortable.

d) **Multi-Language Support:**

Before Visual Studio.Net 2008 there was a single framework with a IDE but history was changed with VS.Net 2008 onwards, It has more than single framework with single IDE, this is known as “Multi-Target Support”.

e) **Multi-Device Support:**

Application built on .Net framework can execute on Desktops, Tablet, Hand held and Note book Devices. These devices need only MS.Net framework on which it was built.

f) **Ease of Deployment:**

After the development now, finally we need to deploy application to release it for customers.MS.Net gives us easy deployment processes like XCopy, SetUp wizard. We need not to do more while deploy applications.

g) **Security:**

Windows always criticize for security but in .Net there are many way to implement security. Framework gives libraries like System.Security it has many types like cryptography, Windows policy etc.

h) **Automatic Memory Management:**

MS.Net does not require garbage collection, CLR collects garbage on behalf of programmer, and it means CLR releases memory of object when it goes out of scope. We can also do it forcefully using System.GC library.

i) No more DLL hell:

Vb 6.0 always requires DLL registration while working with activeX control.

In .Net normally CLR does not require any type of registration. .Net does not require API viewer as well.

j) Strong XML Support:

XML (Extensible Markup Language) comes from SGML (Standard Generalized Markup Language).XML is core of MS.Net as xml has

basically three forms in .net platform.

- a) Data Presentation (in ADO.Net).
- b) Interface Designing (in WPF).
- c) Application Configurations (in Windows, ASP.Net and WPF).

1.5 About .Net Framework:

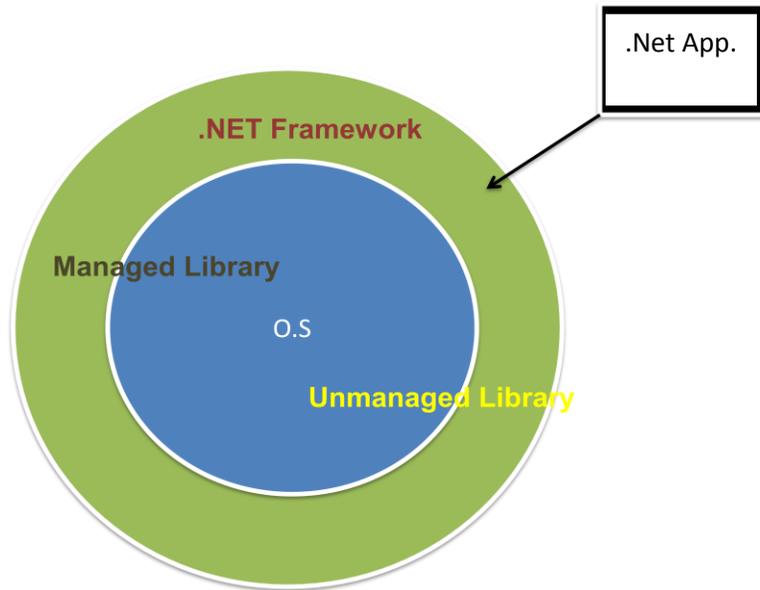
Net framework is actually wrapper around Operating system. Software developed with .NET does not worry about operating system task like file handling and memory allocation. .NET makes internet applications as easy as Desktop applications. Enables high-level software integration through the use of Web services.

It includes a large library of coded solutions to common programming problems and a virtual machine that manages the execution of programs written specifically for the framework.

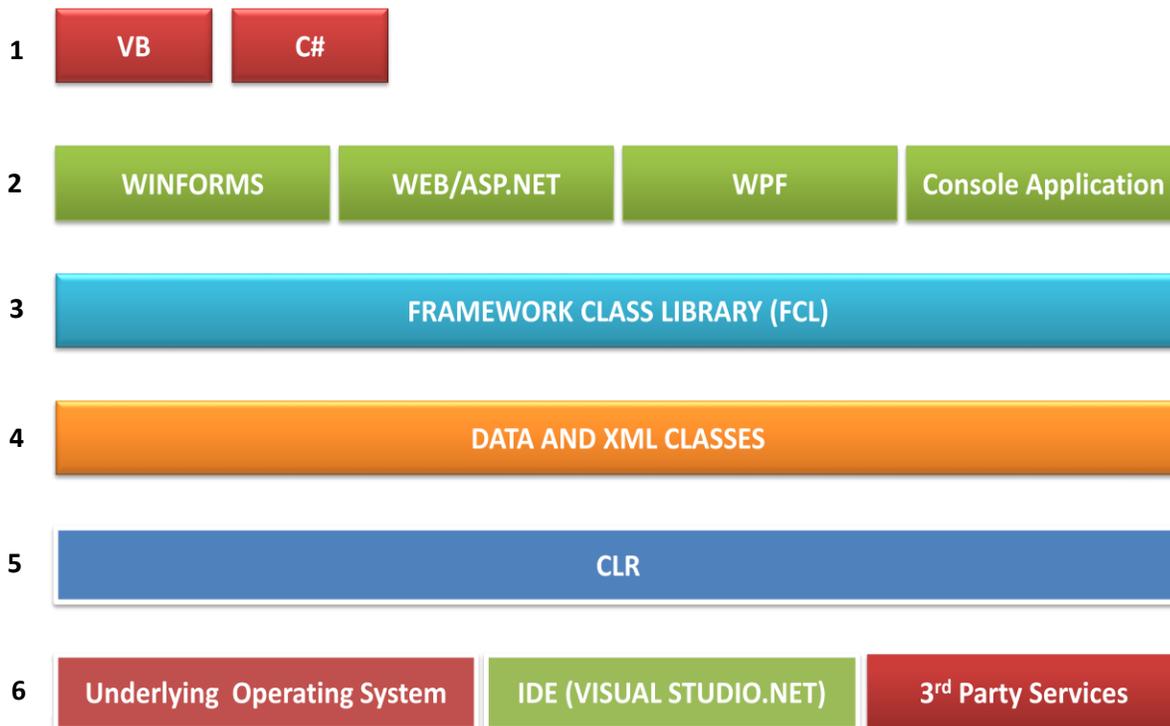
- NET Framework is designed to fulfill the following

Objectives:

- Consistent object-oriented programming environment
- Code-execution environment



1.6 Architecture of .Net Framework:



1. First Layer includes different .Net Framework compatible languages like C#.Net, VB.Net

- a. **Visual Basic.Net:**

In .net Framework all languages used Same Class Library (FCL).

Therefore, which language you are using is not matter.VB. net is newly implemented in .net Framework but this matched to your Visual Basic .

There is basic Difference between vb and vb.net that vb is Object Based and Vb.net is Object Oriented. That is why vb.net taking advantage over vb.

Hence, vb.net supports code reusability as per OOP's Feature and vb.net has all the features as per OOP's.

In Vb.net Classes are extended. Due to rich classes and Easy to use, vb.net is becoming more popular.

- b. **Visual C#.Net:**

C# (pronounced as 'C Sharp') is the new language introduced with .Net for developers. The first language that has been designed from ground up with internet in mind. It is modern language that Combines the best features of many commonly used languages: VB, JAVA and C++.

2. **Environment:**

Second layer consist of different application environment .Net.

We had discussed about this environment in 1.3.

3. **FCL (Framework Class Library):**

Framework Class Library contains Namespaces.Namespace allow you to hierarchically organize classes in logical groups based on what they do and where they originate. Namespaces organizes classes.

Some Examples are

Data:-OleDb, SqlClient

IO: - Stream Writer, Stream Reader

Diagnostics:-Process.

4. Data and XML Classes:

This is also part of FCL but we have state it separately due to its specific purpose. Data Classes are used to implementing database technologies like to save records, to retrieve records, filter records and to manipulate records.ADO.net deals with accessing databases. ADO.Net provides access to data such as Microsoft SQL Server, OLEDB and XML.Ado.Net based on disconnected architecture.

DATA classes: SqlClient- for SQL Server,Oledb- for Ms-Access.

ODBC –for Oracle.

XML Classes: .Net provides Six Classes for accessing XML.

System.XML- Contains major XML Classes for reading and writing XML.

a) XmlReader:

b) XmlWriter:

c) XmlNode:

d) XmlDocument:

e) XmlDocumentFragment:

f) XmlDataDocument:

5. CLR (Common Language Runtime):

Run time is an environment in which programs are executable.

CLR is runtime provide for .Net application.

e.g. To execute a program written in Vb6 the machine must have a VB runtime (msvbvm.dll) installed, Java Programs required JVM since different languages require different runtime the developer life become more difficult. To avoid such problems .NET introduced a single common Language Runtime that all .NET languages share.

Each assembly you build can be either an executable application or a DLL containing a set of types for use by an executable application. Of course, the CLR is responsible for managing the execution of code contained within these assemblies. This means that the .NET Framework must be installed on the host machine. Microsoft has created a redistribution package that you can freely ship to install the .NET Framework on your customers' machines. Some versions of Windows ship with the .NET Framework already installed. We will see how source code converts into native machine code by CLR using VES- section 1.7.

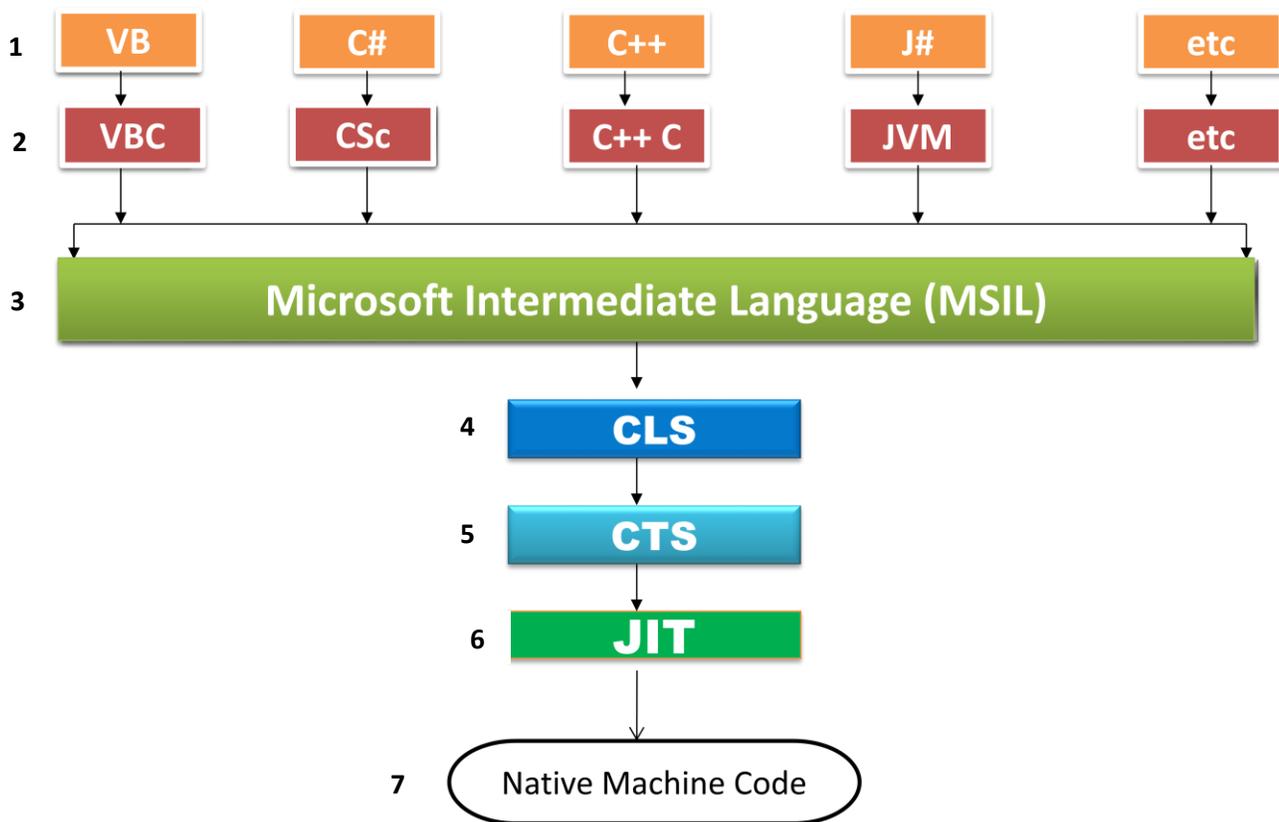
6. Underlying O.S, IDE and Third Party Services:

Last layer of .Net framework consist of operating system on which .net runtime is installed.

IDE (Integrated Development Environment) is editor for developers to design, Program and debug applications.

Third Party Services are that service which comes from outside .net environment into application. E.g., ActiveX Controls.

1.7 Virtual Execution System (VES):



1. **Languages:** First layer of CLR consist different .Net framework compatible languages. Each language has their own syntax.
2. **Compilers:** Second layer of CLR consist of different compilers for different languages, here compiler does not compile code into machine language it is only checks and build source code.

3. **MSIL** (Microsoft Intermediate Language):

.NET-programming language does not compile into executable code, instead it compiles into an intermediate code called Microsoft Intermediate Language (MSIL). IL is CPU independent language. The IL code is send to the CLR that converts the code to machine language using Just –In-Time Compiler.

4. **CLS** (Common Language Specification) :

CLS introduced to make it easier for language developers to adapt their languages to make them compatible with .NET. The CTS and CLS are the foundation for interoperation. The CLI defines a subset of features that are considered compatible across language boundaries. This subset is called Common Language Specification (CTS).

5. **CTS** (Common Type System):

Language interoperability under .Net is possible only when all the languages Share a common data type system. For this, common type system (CTS) is introduced. CTS ensures that an **int** in C# is same as an **int** in VC++.Under CTS All the classes are mapped to the structure defined in Base Class Library.

The features of CTS.

1. Primitive types: Integer and String, etc.
2. Complex Types: structures, Classes, Enumerations, etc

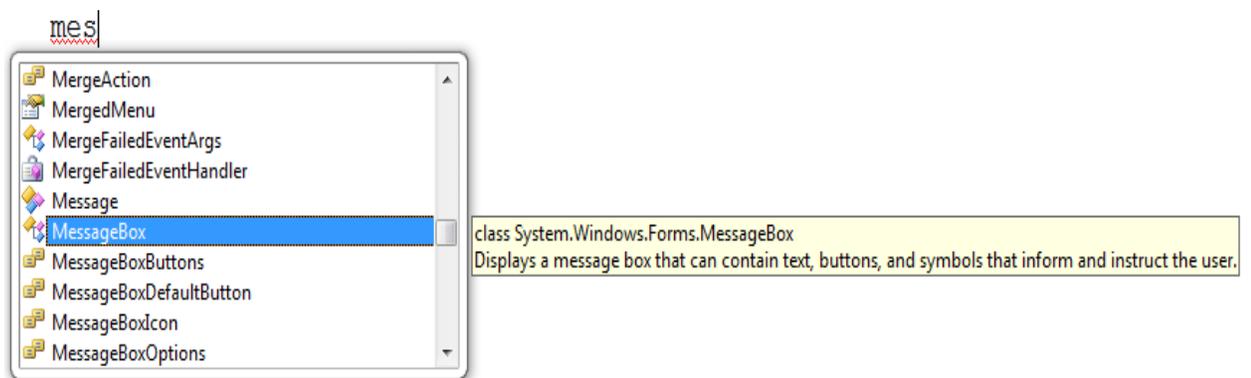
6. **JIT** (Just-In-Time Compiler):

JIT compilers play a major roles in the .Net platform because all .NET

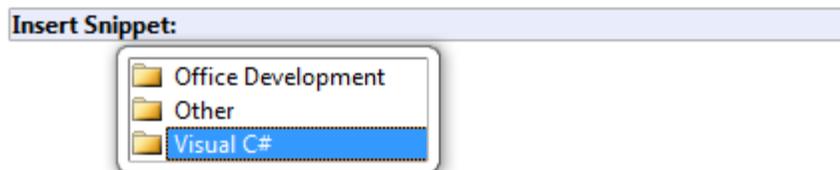
Portable Executable (PE) files contain IL and metadata, not native code. The JIT compilers convert IL to native code so that it can execute on the target operating system. For each method that has successfully been verified for type Safety, a JIT compiler in the CLR will compile the method and convert it into Managed native code. Managed native code is required because the execution support components will manage and execute only managed code on the target platform.

1.8 Features of Visual Studio for .Net Professional:

1. **Shared IDE:** VS.Net has various environments with multiple languages.
2. **VS 2010 Multi-Targeting Support:** Visual Studio 2008, 2010 and 2012 has multiple .Net framework with single IDE for backward and forward compatible.
3. **Intellisense:** VS has strong documentation support with intellisense to avoid error prone coding. Intellisense is a special type of popup window, which explores .net framework in-built type, and user defined types.



4. **Strong Debugging Support:** VS.Net has strong debugging support. It provides break point to suspend debugger and resume it step by step. VS.Net also provides watch window for modification of values of objects at runtime.
5. **Code Snippet:** VS.net provides a facility to access syntax templates for different block of language. E.g. for loop, for..each Loop etc.



```
foreach (var item in collection)
{
}
}
```

6. **Solution Explorer Window:** This window keeps together all the files of application. We can load project using solution file located inside project directory. We can find it in View→Solution Explorer.
7. **Properties Window:** This window contains properties for Application and controls. We can find it in View→Properties.
8. **Object Browser:** This window is used to browse libraries of framework and browse namespace of our project as well. We can find it View→Object Browser.

1.9 .NET Compatible Programming Languages:

Visual Basic.Net	RPG
C#	Component Pascal
APL	Mercury
Fortran	Scheme
Pascal	Curriculum
C++	Mondrian
Haskell	SmallTalk
Perl	Eiffel
J#	Oberon
Python	Standard ML
COBOL	Forth
Microsoft JScript	VC++