Govt. Polytechnic Dehradun

Branch- Information Technology, Sem-6th

Subject-.Net with C#

**.NET FRAMEWORK** - .NET is a framework to develop software applications. It is designed and developed by Microsoft and the first beta version released in 2000.

It is used to develop applications for web, Windows, phone. Moreover, it provides a broad range of functionalities and support.

This framework contains a large number of class libraries known as Framework Class Library (FCL). The software programs written in .NET are executed in the execution environment, which is called CLR (Common Language Runtime). These are the core and essential parts of the .NET framework.

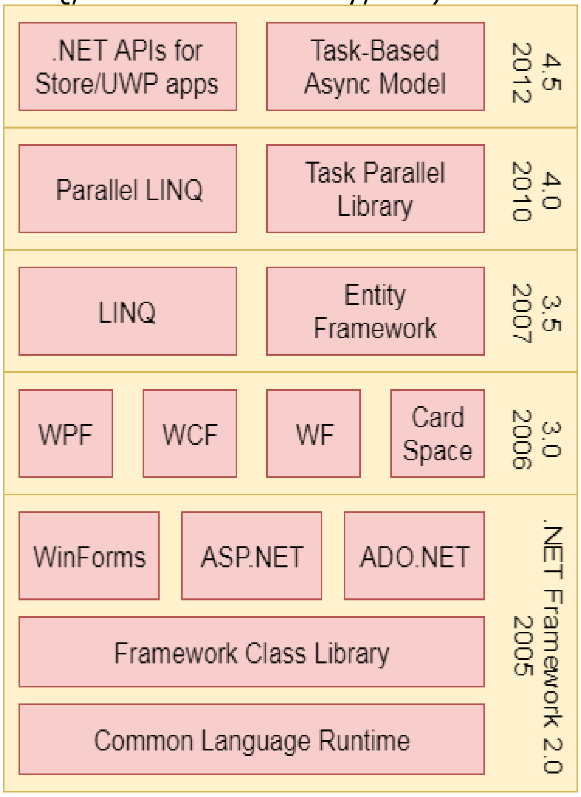
This framework provides various services like memory management, networking, security, memory management, and type-safety.

The .Net Framework supports more than 60 programming languages such as C#, F#, VB.NET, J#, VC++, JScript.NET, APL, COBOL, Perl, Oberon, ML, Pascal, Eiffel, Smalltalk, Python, Cobra, ADA, etc.

Following is the .NET framework Stack that shows the modules and components of the Framework.

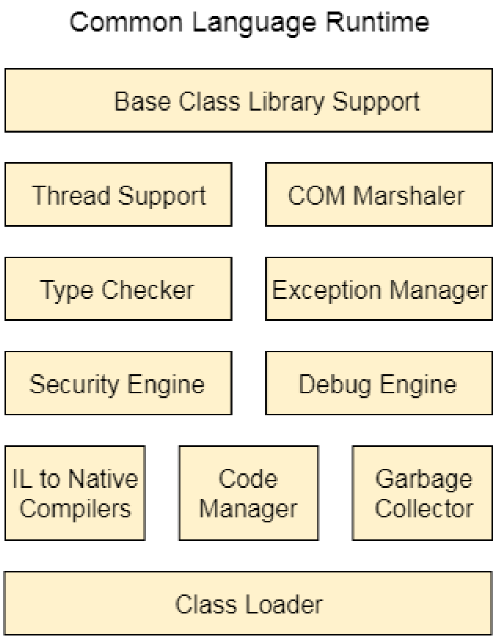
The .NET Framework is composed of four main components:

1. Common Language Runtime (CLR)
2. Framework Class Library (FCL),
3. Core Languages (WinForms, ASP.NET, and ADO.NET), and
4. Other Modules (WCF, WPF, WF, Card Space, LINQ, Entity Framework, Parallel LINQ, Task Parallel Library, etc.)



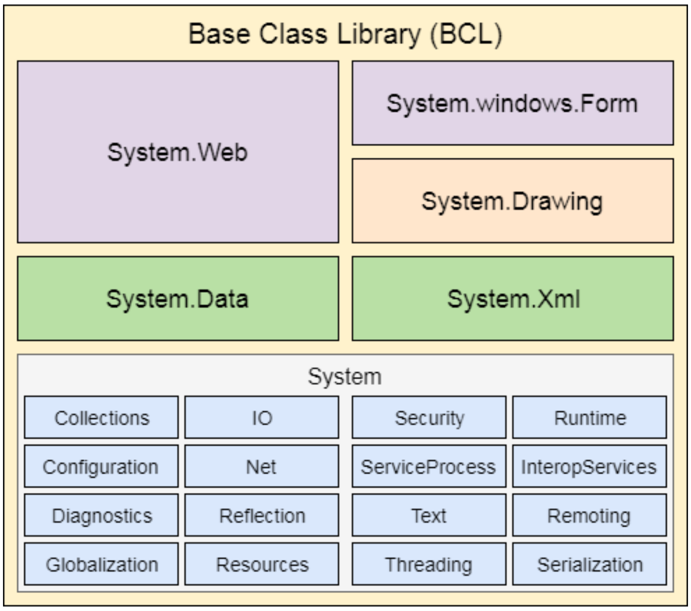
## CLR (Common Language Runtime)

It is a program execution engine that loads and executes the program. It converts the program into native code. It acts as an interface between the framework and operating system. It does exception handling, memory management, and garbage collection. Moreover, it provides security, type-safety, interoperability, and portablility. A list of CLR components are given below:



## FCL (Framework Class Library)

It is a standard library that is a collection of thousands of classes and used to build an application. The BCL (Base Class Library) is the core of the FCL and provides basic functionalities.



## Features of .NET Framework

- It is a platform neutral framework.
- It is a layer between the operating system and the programming language.

- It supports many programming languages, including VB.NET, C# etc.

- .NET provides a common set of class libraries, which can be accessed from any .NET based programming language. There will not be separate set of classes and libraries for each language. If you know anyone .NET language, you can write code in any .NET language.

- In future versions of Windows, .NET will be freely distributed as part of operating system and users will never have to install .NET separately.

## Application Development and Execution

### Developing application:

Since Microsoft .NET is a Multilanguage platform then any.NET based language can be chosen to develop applications. Comfort ability of application programmers, specific requirement of applications may be the major factors in selection of language.

### Choosing a Compiler

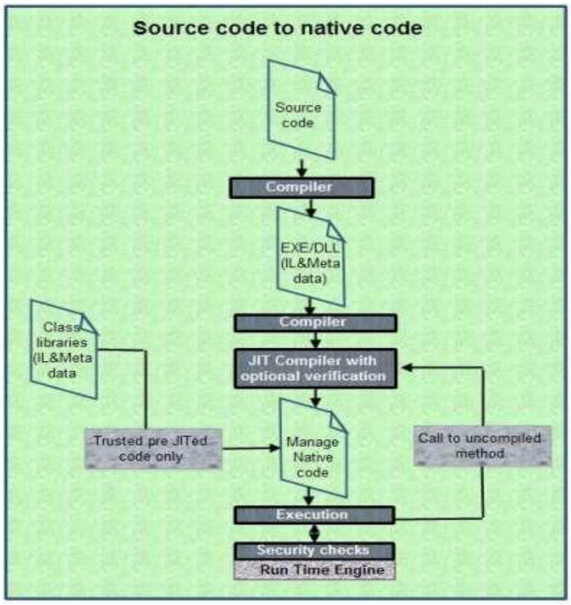
According to the language we can choose its run time aware compiler for .NET platform. Because it is a Multilanguage execution environment, the runtime supports a wide variety of data types and language features.

### Compiling to MSIL

When compiling your source code, the compiler translates it into an intermediate code represented in Microsoft intermediate language (MSIL). Before code Can be run, MSIL code must be converted to CPU-specific code, usually by a just-in-time (JIT) compiler. When a compiler produces MSIL, it also produces metadata. Metadata includes following information.

- Description of the types in your code, including the definition of each type,
- The signatures of each type's members,
- The members that your code references,
- Other data that the runtime uses at execution time.

The MSIL and metadata are contained in a portable executable (PE) file that is based on and extends the published Microsoft PE and common object file format (COFF) used historically for executable content. This file format, which accommodates MSIL or native code as well as metadata, enables the operating system to recognize common language runtime images.


Source code to native code
Source code
Compiler
EXE/DLL (IL&Meta data)
Compiler
Class libraries (IL&Meta data
JIT Compiler with optional verification
Trusted pre JITed code only
Manage Native code
Call to uncompiled method
Execution
Security checks
Run Time Engine

The presence of metadata in the file along with the MSIL enables your code to describe itself. This composite file serves as a self describing unit to the .NET Framework Runtime is called Assembly. The runtime locates and extracts the metadata from the file as needed during execution.

### JIT Compilation to Native code:

The MSIL code is compiled into native code by component of CLR named JIT Compiler.

JIT compiler intelligently guesses and compiles the intermediate code on piece by piece basis. This piece may be a method or a set of methods. Before a method can be run, it must be compiled to processor-specific code. Each method for which Microsoft intermediate language (MSIL) has been generated is just-in-time-compiled (JIT-compiled) when it is called for the first time, and then run. The next time the method is run, the existing JIT -compiled native code is run. The process of JIT -compiling and then executing the code is repeated until execution is complete.

### Type Safety Checking

As part of compiling MSIL to native code, code must pass a verification process unless an administrator has established a security policy that allows code to bypass verification. Verification examines MSIL and metadata to find out whether the code is type safe, which means that it only accesses the memory locations it is authorized to access. Additionally, verification inspects code to determine whether the MSIL has been correctly generated, because incorrect MSIL can lead to a violation of the type safety rules .

The runtime relies on the fact that the following statements are true for code that is verifiably type safe:

- A reference to a type is strictly compatible with the type being referenced.
- Only appropriately defined operations are invoked on an object.
- Identities are what they claim to be.

If type-safe code is required by security policy and the code does not pass verification, an exception is thrown when the code is run.

### Code execution under CLR

The common language runtime is responsible for providing following low-level execution services, such as garbage collection, exception handling, security services, and runtime type safety checking. Because of the common language runtime's role in managing execution, programs that target the .NET Framework are sometimes called "managed" applications.

## WinForms

Windows Forms is a smart client technology for the .NET Framework, a set of managed libraries that simplify common application tasks such as reading and writing to the file system.

## ASP.NET

ASP.NET is a web framework designed and developed by Microsoft. It is used to develop websites, web applications, and web services. It provides a fantastic integration of HTML, CSS, and JavaScript. It was first released in January 2002.

## ADO.NET

ADO.NET is a module of .Net Framework, which is used to establish a connection between application and data sources. Data sources can be such as SQL Server and XML. ADO .NET consists of classes that can be used to connect, retrieve, insert, and delete data.

## WPF (Windows Presentation Foundation)

Windows Presentation Foundation (WPF) is a graphical subsystem by Microsoft for rendering user interfaces in Windows-based applications. WPF, previously known as "Avalon", was initially released as part of .NET Framework 3.0 in 2006. WPF uses DirectX.

## WCF (Windows Communication Foundation)

It is a framework for building service-oriented applications. Using WCF, you can send data as asynchronous messages from one service endpoint to another.

## WF (Workflow Foundation)

Windows Workflow Foundation (WF) is a Microsoft technology that provides an API, an in-process workflow engine, and a rehostable designer to implement long-running processes as workflows within .NET applications.

## LINQ (Language Integrated Query)

It is a query language, introduced in .NET 3.5 framework. It is used to make the query for data sources with C# or Visual Basics programming languages.

## Entity Framework

It is an ORM based open source framework which is used to work with a database using .NET objects. It eliminates a lot of developers effort to handle the database. It is Microsoft's recommended technology to deal with the database.

## Parallel LINQ

Parallel LINQ or PLINQ is a parallel implementation of LINQ to objects. It combines the simplicity and readability of LINQ and provides the power of parallel programming.

It can improve and provide fast speed to execute the LINQ query by using all available computer capabilities.

Apart from the above features and libraries, .NET includes other APIs and Model to improve and enhance the .NET framework.

In 2015, Task parallel and Task parallel libraries were added. In .NET 4.5, a task-based asynchronous model was added.