1 .NET Framework Architecture

1.1 History of .Net

- Sometime in the July 2000, Microsoft announced a whole new software development framework for Windows called .NET in the Professional Developer Conference (PDC).
- 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 (Software Development Kit)

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 2012,13,14	2.0, 3.0, 3.5, 4.0,4.5
VS.NET 2015,2017	2.0,3.0.3.5,4.0,4.5,4.6
VS.NET 2019	2.0,3.0.3.5,4.0,4.5,4.6,4.7,4.8

1.3 Flavors of .NET

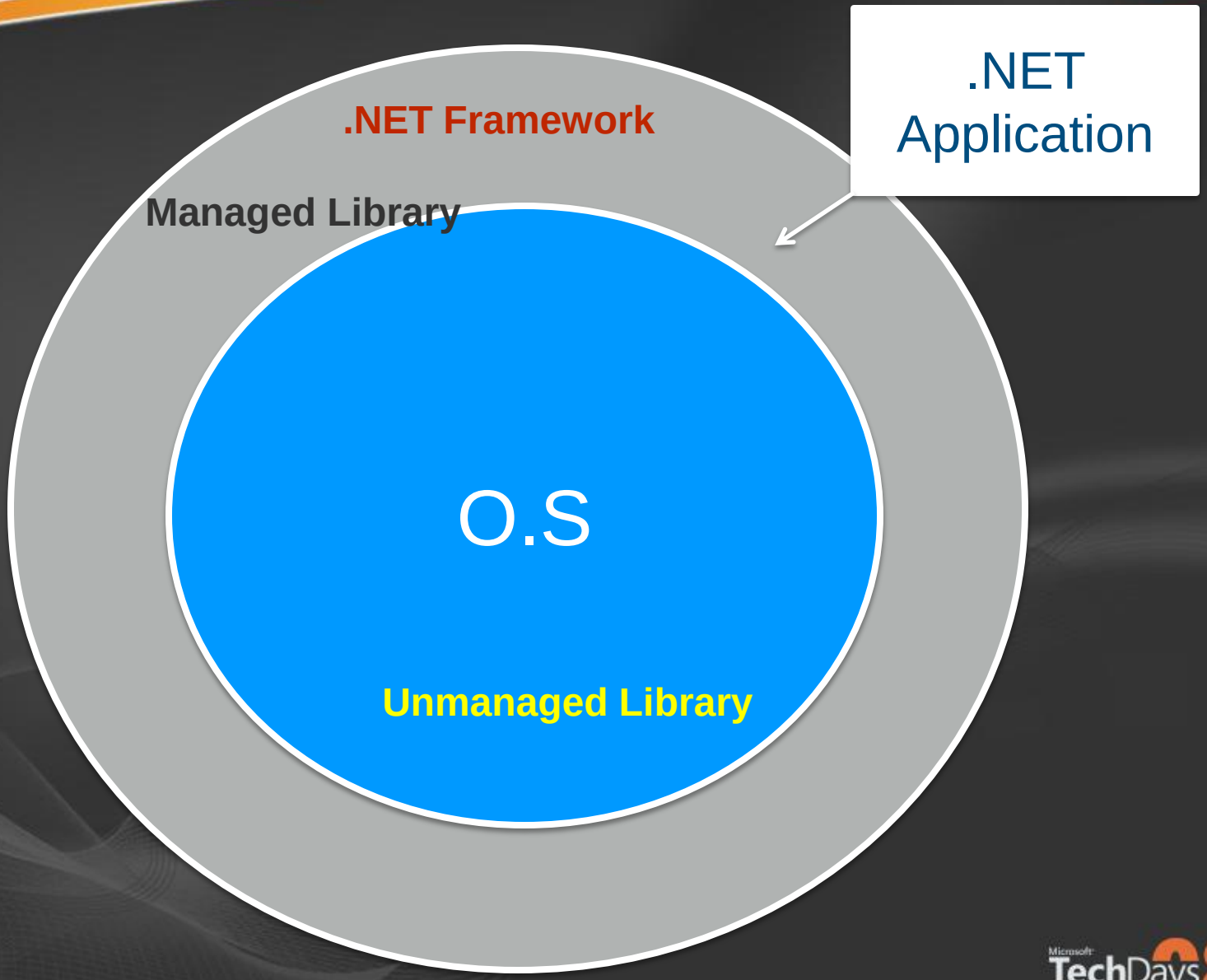
Console applications: These refer to traditional DOS kind of applications like batch scripts. It has CUI(character based User Interface).

Windows Form based applications: These refer to traditional rich client applications. It has GUI (Graphical User Interface)

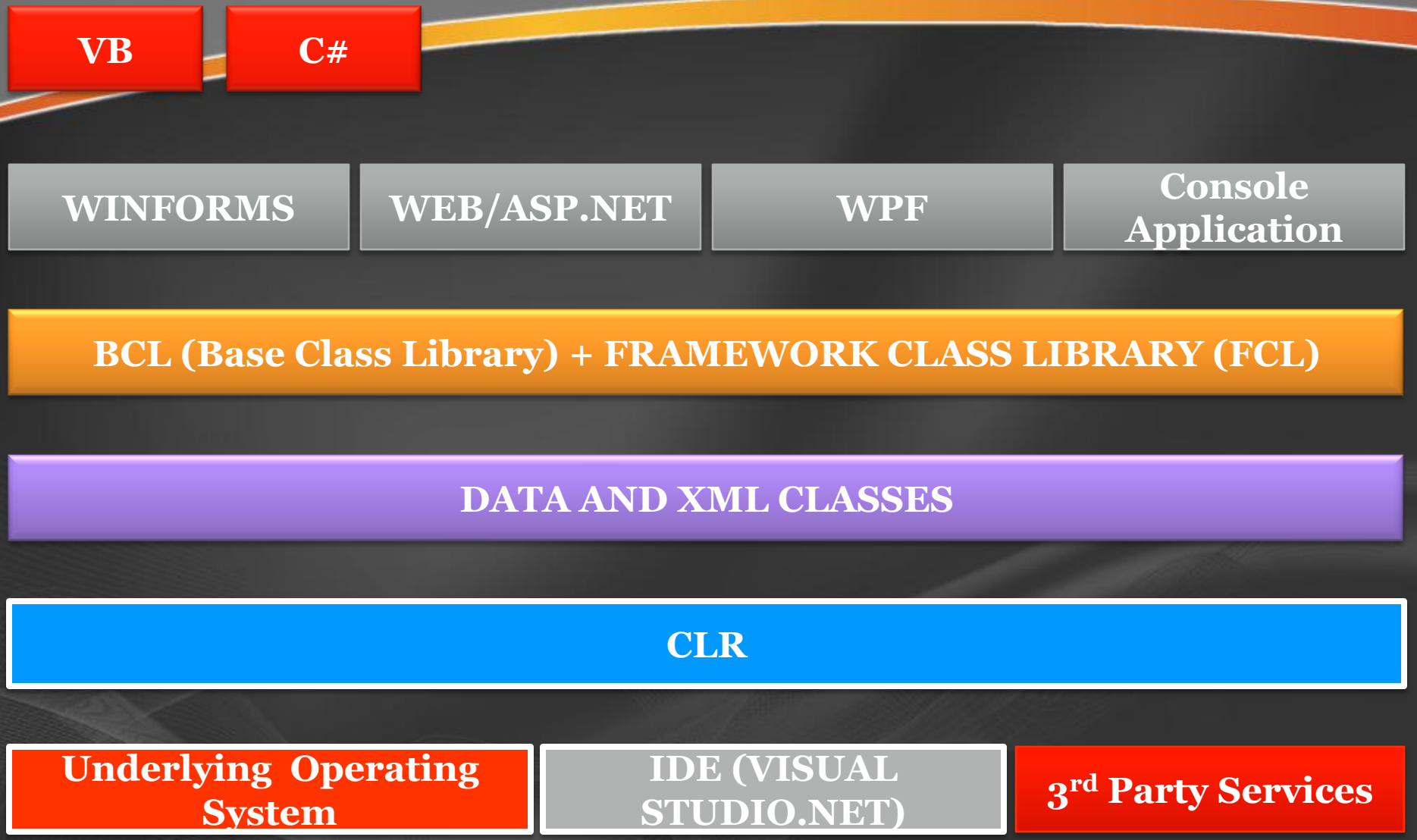
ASP.NET/ Web applications: These include dynamic and data driven browser based applications.

WPF Application (Windows Presentation Foundation): 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.

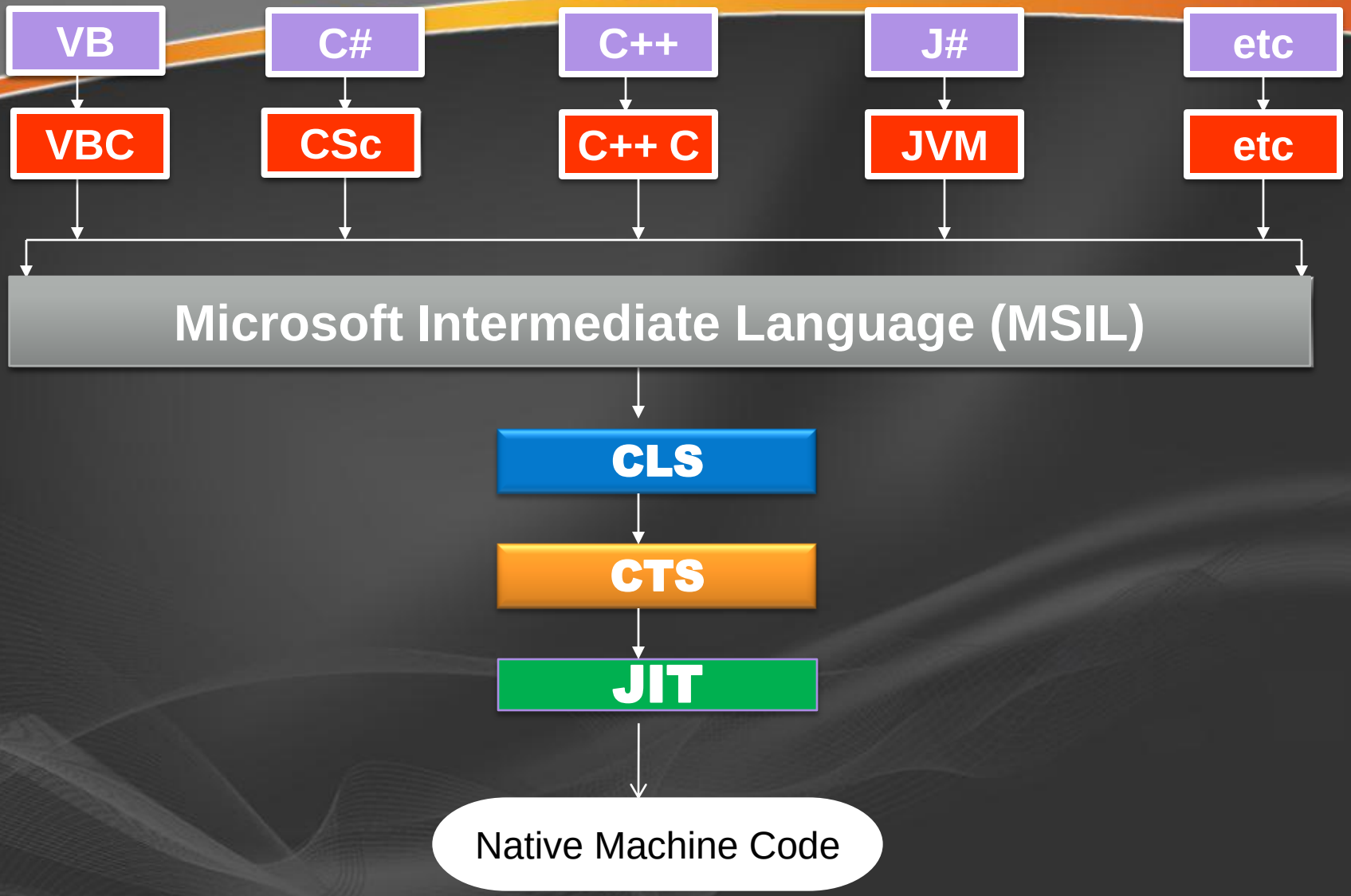
1.4 Communication between .NET framework & O.S



1.5 Architecture of .Net Framework:



1.6 Virtual Execution System (VES)



What is CLR and Role of CLR?

▪.NET CLR is a run-time environment that manages and executes the code written in any .NET programming language.

•It converts code into native code which further can be executed by the CPU.

•**Base Class Library Support**
It is a class library that provides support of classes to the .NET application

•**Type Checker**
It checks types used in the application and verifies that they match to the standards provided by the CLR.

•**Code Manager**
It manages code at execution run-time.

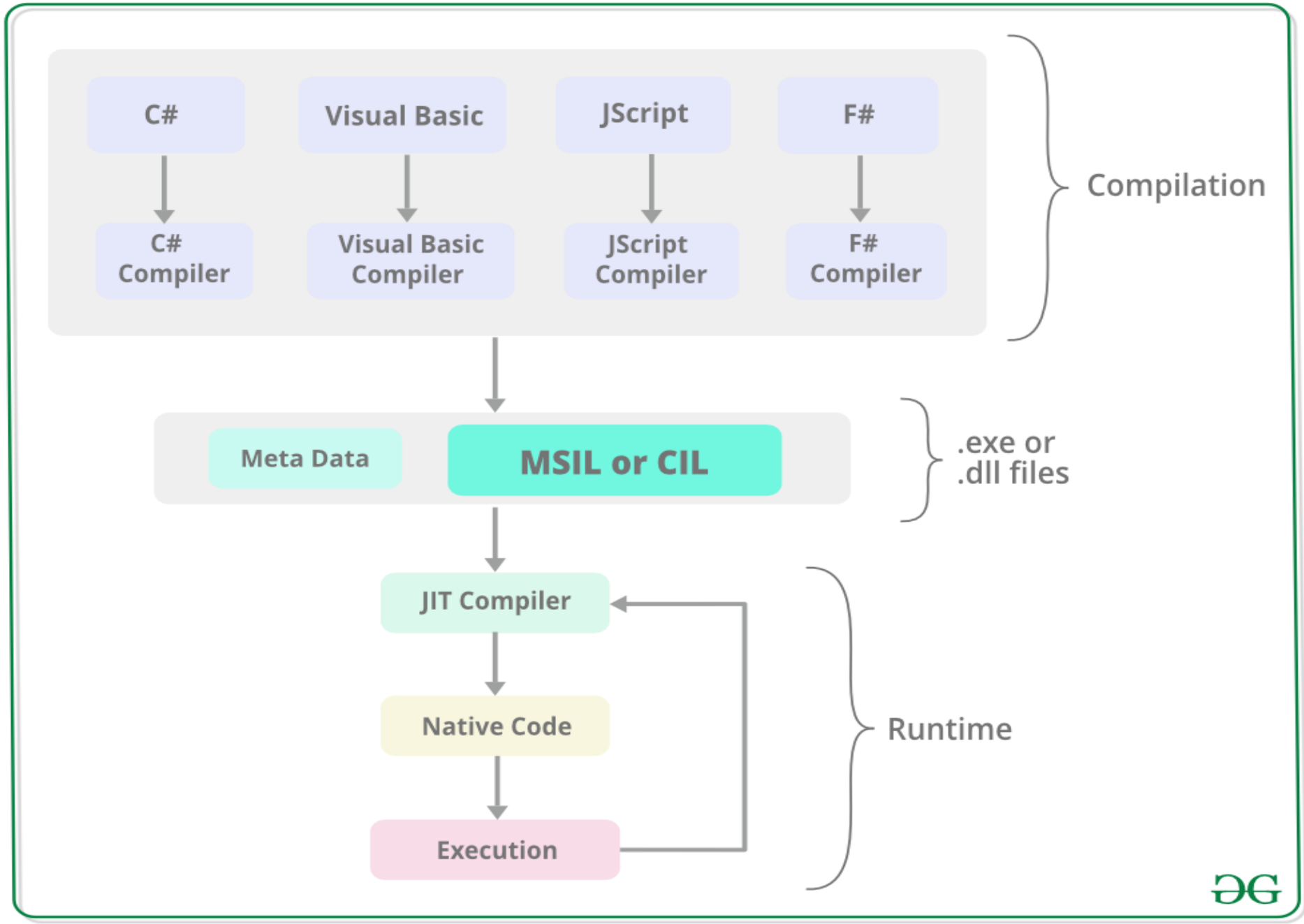
•**Garbage Collector**
It releases the unused memory and allocates it to a new application.

•**Exception Handler**
It handles the exception at runtime to avoid application failure.

Microsoft Intermediate Language or Common Intermediate Language (MSIL.CIL)

- is a set of instructions that are platform independent and are generated by the language-specific compiler from the source code.

- The MSIL is platform independent and consequently, it can be executed on any of the Common Language Infrastructure supported environments such as the Windows *.NET* runtime.


C#
Visual Basic
JScript
F#
C# Compiler
Visual Basic Compiler
JScript Compiler
F# Compiler
Compilation
Meta Data
MSIL or CIL
.exe or .dll files
JIT Compiler
Native Code
Execution
Runtime

CLS and CTS

- CLS stands for Common Language Specification and it is a subset of CTS. It defines a set of rules and restrictions that every language must follow which runs under the .NET framework.
- The languages which follow these set of rules are said to be CLS Compliant.
- **For Example**
- For each. Net lang we have to use the new keyword to create an instance, index of array of each. Net lang should start with 0.

- Common Type System (CTS) describes the datatypes that can be used by managed code. CTS defines how these types are declared, used and managed in the runtime.

- It facilitates cross-language integration, type safety, and high-performance code execution.

Just-In-Time (JIT):Types of JIT

Just-in-time (JIT) is a term used to describe an action such as compilation or object activation only at the time when it becomes necessary. This term is associated mostly with software compilation. JIT compilation is mainly designed for high-speed code execution and support for multiple platforms.

- Pre-JIT: Compiles the entire source code during compilation and is used at the time of deployment.

- Econo-JIT: Compiles methods that are called during run time.
- Normal-JIT: Compiles only the methods called during run time (at the instant of their first call) and stores the compiled code in cache to be used in subsequent calls.

MS.Net Framework 4.0	MS.Net Framework 4.5
Dynamic binding C# dynamic is a keyword that is used to make a property or a method dynamic. When we make dynamic type, compiler does not check it at compile-time.	**Bundling and Minification**
Named/optional arguments	**Strongly Typed Data Controls**
Generic covariant and contra variance.	**Model Binding** - Isolating the Web Form from the Model The Model binding feature in ASP.NET 4.5 enables you to develop Webforms that are independent of the Model that populates the view.

Embedded interop. types	**Value Providers** •ASP.NET4.5 provides many Value Providers that can be used to filter data. These are: Querystring Session Cookie Control Value
	Support for improved paging in ASP.NET 4.5 GridView control
	Enhanced support for asynchronous programming: Using System.Threading.Task
	Support for HTML5 form types

	Windows Presentation Foundation (WPF) Features in 4.5
	Support for arrays that are larger than 2 gigabytes (GB) on 64-bit platforms
	Asynchronous file operations
	Portable Class Libraries

1.7 Features of .NET

- ✓Rich Functionality out of the box:
- ✓Easy development of web applications:
- ✓OOPs Support
- ✓Multi -Language Support:
- ✓Multi-Device Support:

- ✓Ease of deployment and configuration
- ✓Security:
- ✓Automatic memory management
- ✓No more DLL Hell:
- ✓Strong XML support

1.8 Features of Visual Studio for .NET

- Shared IDE
- VS.Net Multi-Targeting Support
- Intellisense
- Strong Debugging Support
- Code Snippet

1.9 .NET Compitible Programming Languages

- Visual Basic.Net
- C#
- APL
- 4Fortran
- Pascal
- C++
- Haskell
- Perl
- J#
- Python
- COBOL
- Microsoft JScript
- RPG
- Component Pascal
- Mercury
- Scheme
- Curriculum
- Mondrian
- SmallTalk
- Eiffel
- Oberon
- Standard ML
- Forth
- VC++

FAQ's on .Net Basics

1. Which of the following components of the .NET framework provide an extensible set of classes that can be used by any .NET compliant programming language?

A. .NET class libraries

B. Common Language Runtime

C. Common Language Infrastructure

D. Component Object Model

E. Common Type System

2. Which of the following .NET components can be used to remove unused references from the managed heap?

A. Common Language Infrastructure

B.CLR

C. Garbage Collector

D. Class Loader

E. CTS

3. Code that targets the Common Language Runtime is known as

A. Unmanaged

B. Distributed

C. Legacy

D. Managed Code

E. Native Code

4. Which of the following is the root of the .NET type hierarchy?

A. System.Object

B. System.Type

C. System.Base

D. System.Parent

E. System.Root

5. Which of the following jobs are done by Common Language Runtime?

It provides core services such as memory management, thread management, and remoting.

It enforces strict type safety.

It provides Code Access Security.

It provides Garbage Collection Services.

A. Only 1 and 2

B. Only 3, 4

C. Only 1, 3 and 4

D. Only 2, 3 and 4

E. All of the above

6. Which of the following components of the .NET framework provide an extensible set of classes that can be used by any .NET compliant programming language?

A. .NET class libraries

B. Common Language Runtime

C. Common Language Infrastructure

D. Component Object Model

E. Common Type System

7. Provide us Type Safety and Language interoperations in CLR?

A. CTS

B. CLS

C. MSIL

8. Which of Following is not type of JIT?

A. Econo JIT

B. Pre-JIT

C. Normal-JIT

D. None of the Above

9. MSIL stands for?

A. Microsoft Internal Language

B. Microsoft Intermediate Language

C. Microsoft Intermediate Linker

10. Which is the following tool used to show meta data?

A. ILSAM

B. ILDLL

C. ILDASM