

Objectives

- The CLR and .NET Framework: Understand the motivation behind the .NET platform , Common Language Infrastructure (CLI).
- Common Type System (CTS) , Common Language Specification (CLS)
- Common Language Runtime (CLR),
- Assembly, metadata, namespace, type distinction
- Contrast single-file and multi-file assemblies
- Platform independent .NET(.NET core)



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Motivation Behind the .NET

- Consistent object-oriented programming environment.
- To provide a code-execution environment that minimizes software deployment and versioning conflicts.
- To provide a code-execution environment that promotes safe execution of code, including code created by an unknown or semi-trusted third party.
- To make the developer experience consistent across widely varying types of applications, such as Windows-based applications and Web-based applications.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Motivation Behind the .NET

- To build all communication on industry standards to ensure that code based on the .NET Framework can integrate with any other code.

The .NET Framework provides a comprehensive programming model for building all kinds of applications on Windows, from mobile to web to desktop.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



.NET and .NET Framework

- .NET is a developer platform made up of tools, programming languages, and libraries for building many different types of applications.
- There are various implementations of .NET. Each implementation allows .NET code to execute in different places—Linux, macOS, Windows, iOS, Android, and many more.
- **.NET Framework** is the original implementation of .NET. It supports running websites, services, desktop apps, and more on Windows.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



.NET and .NET Framework

- **.NET** is a cross-platform implementation for running websites, services, and console apps on Windows, Linux, and macOS. .NET is open source on GitHub. .NET was previously called .NET Core.
- **Xamarin/Mono** is a .NET implementation for running apps on all the major mobile operating systems, including iOS and Android.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



.NET Framework

- Microsoft .NET Framework is a complex technology that provides the infrastructure for building, running, and managing next generation applications.
- In a layered representation, the .NET Framework is a layer positioned between the Microsoft Windows operating system and your applications.
- .NET is a platform but also is defined as a technology because it is composed of several parts such as libraries, executable tools, and relationships and integrates with the operating system.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



.NET Framework

- Can be described as Development platform or Execution environment which comprises tools and technologies, to develop distributed applications and distributed web services.
- Microsoft started development on the .NET Framework in the late 1990s originally under the name of Next Generation Windows Services.
- It consists of two major components: the **Common Language Runtime (CLR)**, which provides memory management and other system services, and an extensive **Class Library**, which includes tested, reusable code for all major areas of application development



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



CLI

- The **Common Language Infrastructure (CLI)** is an open specification developed by Microsoft and standardized by ISO and ECMA
- It describes the executable code and runtime environment that form the core of the Microsoft .NET Framework and the free and open source implementations Mono and Portable.NET.
- The specification defines an environment that allows multiple high-level languages to be used on different computer platforms without being rewritten for specific architectures.
- The CLI specification describes the following four aspects:-

CTS(Common Type System), Meta data

CLS (Common Language specification), VES(Virtual Execution

System)

- **CTS** : A set of data types and operations that are shared by all CTS-compliant programming languages.
- **METADATA:** It refers to certain data structures embedded within the Common Intermediate Language code that describes the high-level structure of the code.
- Metadata describes all classes and class members that are defined in the assembly, and the classes and class members that the current assembly will call from another assembly.
- The metadata for a method contains the complete description of the method, including the class (and the assembly that contains the class), the return type and all of the method parameters.

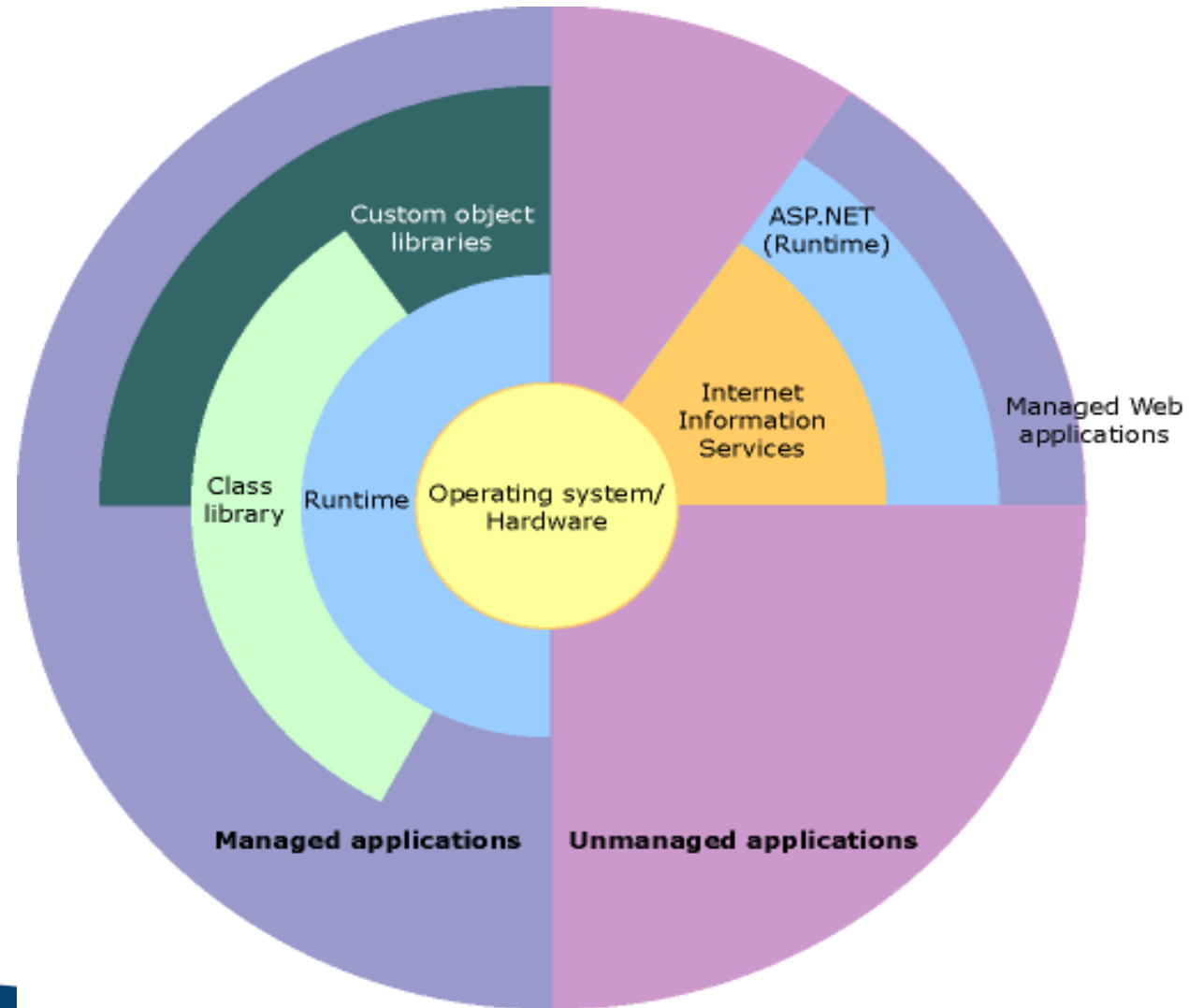
- **Common Language Specification (CLS)** A set of base rules to which any language targeting the CLI should conform in order to interoperate with other CLS-compliant languages. The CLS rules define a subset of the Common Type System.
- **VES :** The VES loads and executes CLI-compatible programs, using the metadata to combine separately generated pieces of code at runtime



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013





**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Features

- **Memory management.** In many programming languages, programmers are responsible for allocating and releasing memory and for handling object lifetimes. In .NET Framework applications, the CLR provides these services on behalf of the application.
- **A common type system.** In traditional programming languages, basic types are defined by the compiler, which complicates cross-language interoperability. In the .NET Framework, basic types are defined by the .NET Framework type system and are common to all languages that target the .NET Framework.
- **An extensive class library.** Instead of having to write vast amounts of code to handle common low-level programming operations, programmers can use a readily accessible library of types and their members from the .NET Framework Class Library.
- **Development frameworks and technologies.** The .NET Framework includes libraries for specific areas of application development, such as ASP.NET for web applications, ADO.NET for data access, and Windows Communication Foundation for service-oriented applications.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



- **Language interoperability.** Language compilers that target the .NET Framework emit an intermediate code named Common Intermediate Language (CIL), which, in turn, is compiled at run time by the common language runtime. With this feature, routines written in one language are accessible to other languages, and programmers can focus on creating applications in their preferred language or languages.
- **Version compatibility.** With rare exceptions, applications that are developed by using a particular version of the .NET Framework can run without modification on a later version.
- **Side-by-side execution.** The .NET Framework helps resolve version conflicts by allowing multiple versions of the common language runtime to exist on the same computer. This means that multiple versions of applications can also coexist, and that an application can run on the version of the .NET Framework with which it was built.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Versions

- Version 1.0 was the first release and was included in the Visual Studio .NET . – add on support for ODBC and Oracle
- Version 1.1 was released after the first release and Microsoft changed the Visual Studio to Visual studio .NET 2003 with the enhanced features. It was the default framework in Windows Server 2003 – Internet protocol V6 support
- Version 2.0 was released with the Visual Studio 2005 in the year 2005 – Generics
- 3.0 was released with the same visual studio of the previous one and was integrated with Windows Vista and Windows Server 2008 – WPF,WCF and WF
- 3.5 – AJAX - Visual Studio 2008
- 4.0 – Visual Studio 2010
- 4.5 – Visual Studio 2012
- 4.5.1 , 4.5.2 - Visual Studio 2013
- 4.6,4.6.1,4.6.2 – Visual studio 2015
- 4.7, 4.7.1, 4.7.2 - Visual studio 2017
- 4.8 ,4.8.1 - Visual Studio 2019 Visual studio 2022

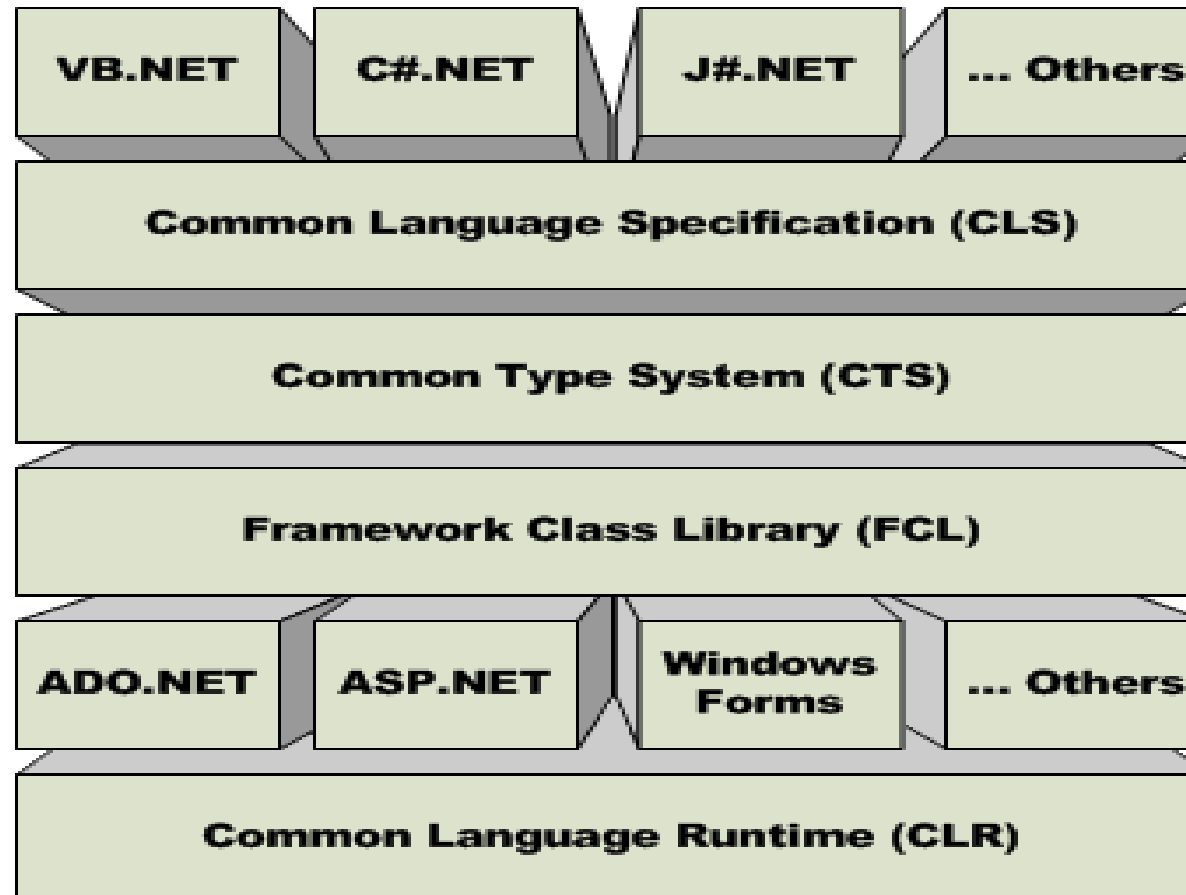


**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



.NET Architecture



© Dan Mahbutt



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



The .NET Framework has two main components:

The .NET Framework has two main components: the common language runtime and the .NET Framework class library.

- The **common language runtime** is the foundation of the .NET Framework
- Runtime as an agent that manages code at execution time, providing core services such as memory management, thread management, and remoting, while also enforcing strict type safety and other forms of code accuracy that promote security and robustness.
- In fact, the concept of **code management** is a fundamental principle of the runtime.
- Code that targets the runtime is known as managed code, while code that does not target the runtime is known as unmanaged code.
- The **class library**, the other main component of the .NET Framework, is a comprehensive, applications ranging from traditional command-line or graphical user interface (GUI) applications to applications based on the latest innovations provided by ASP.NET, such as Web Forms and XML Web services.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



CLR

- The common language runtime manages memory, thread execution, code execution, code safety verification, compilation, and other system services.
- It is an execution environment for program code defined by CLI.
- It lies between operating systems and applications written in .NET languages.
- The runtime enforces code access security
- The runtime also enforces code robustness by implementing a strict type-and-code-verification infrastructure called the common type system (CTS). The CTS ensures that all managed code is self-describing.
- Automatic memory management : CLR provides the garbage collection. The objects whose lifetime is managed by the garbage collection are called managed data.
- The runtime is designed to enhance performance.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



- Platform independence: When you compile a program developed in a language that targets the CLR, the compiler translates the code into an intermediate language ie CPU-independent. The code can be executed from any platform that supports the .NET CLR.
- Security management : It is achieved through the code access Security model. In this, CLR enforces restrictions on managed code through the use of objects called permissions. It specifies what the code can access instead of specifying who can access the resources.
- Language interoperability: It is the ability of an application to interact with another application written in a different programming language. It helps maximize code reuse.

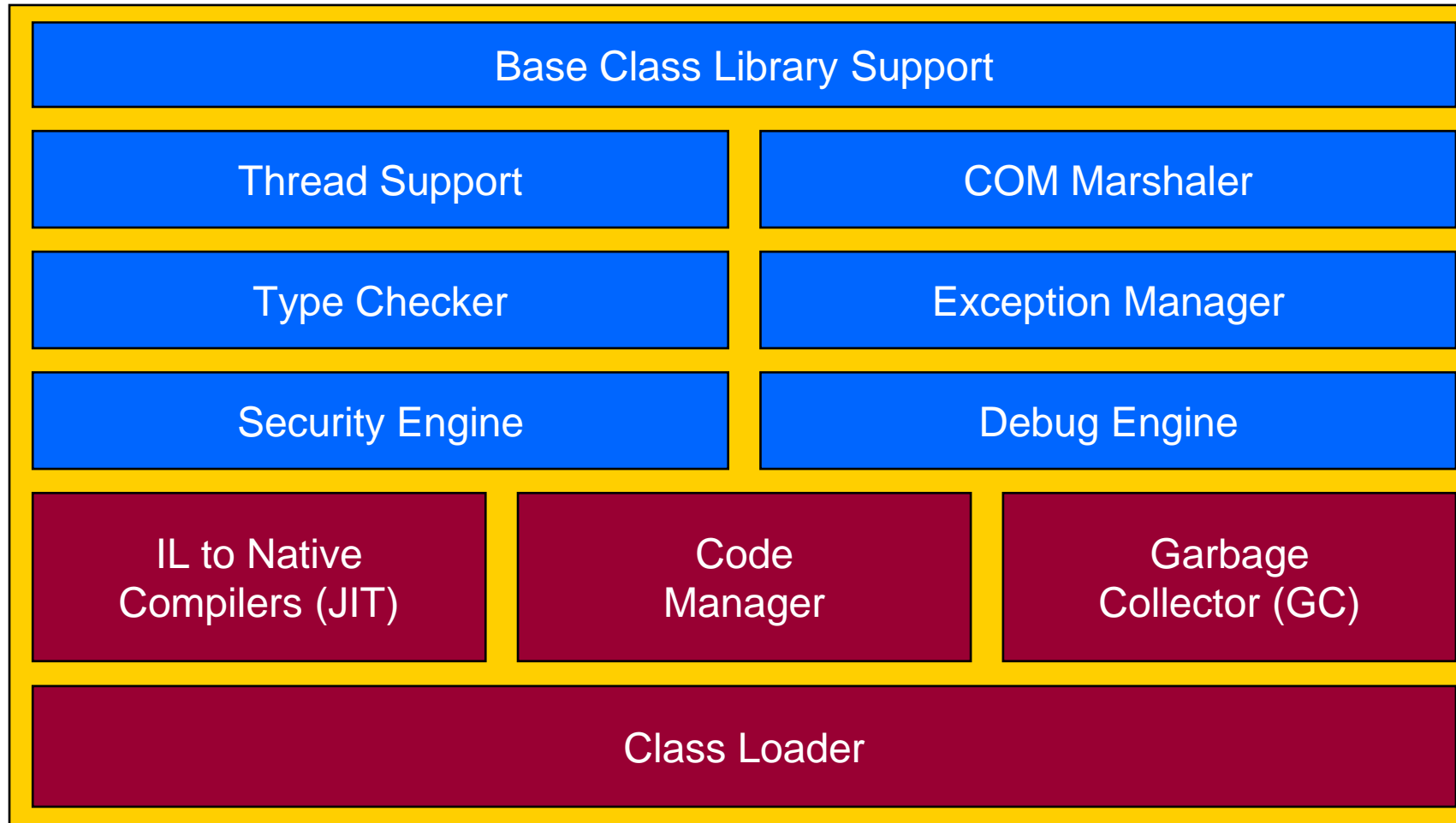


**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



CLR Architecture



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



CLR Architecture

Different components :

- ***Class Loader*** : C# compiler of the framework generates assembly after compiling and this consists of CIL code and Meta data. The class loader is loading the assembly into the run time.
- ***Code Manager***: This component is managing the above code during execution. This takes the responsibility of allocating memory to the objects also.
- ***Garbage collector***: This provides automatic garbage collection of the object when the object is no longer in use. To achieve this, Garbage collector is performing the periodical checks in the heap from where the object gets memory.
- ***Security Checker***: One of the important components of CLR is security checker. This engine restricts the access to system resources such as hard disk and enforcing the restriction on the MSIL code.
- ***Debug Engine*** : Debug means finding and removing the bug from the programs, The application written in any supported framework languages are debugged by this engine



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



- ***Type Checker:*** This ensures the datatype checking of the variable. This is also checking the valid operations on the corresponding datatype. It means that integer value must be assigned to integer datatype and valid operations are allowed in that type. Otherwise it raises the exception.
- ***Thread support:*** Multithreading is a very important feature of any programming language. Threads are playing an very important role in developing the application in this framework The application can contain one or more threads. These threads are managed by the CLR.
- ***Exception Manager :*** The Net framework follows the structured exception handling in all its compliant languages. The exception might rise from managed code as well as unmanaged code. This manager provides the support to handle these type of Exceptions.
- ***Base class library support*** – It provides the types that the applications need at run time.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



CTS(Common Type System)

- The runtime also enforces code robustness by implementing a strict type-and-code-verification infrastructure called the common type system (CTS).
- The CTS ensures that all managed code is self-describing
- Type refers the datatypes supported by the programming languages.
- .NET supports the various programming languages. To ensure a free interaction, CTS has been introduced.
- The specification of CTS defines about the definitions of the type and the values supported by that.
- CTS is supporting the object oriented concepts thus all types represented in this are objects and also provides support for sharing of common data types.
- The system allows you to do the language integration.
- This represents that code in one language can be inherited by the code in another language.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



- The CTS also defines the rules that ensures that the data types of objects written in various languages are able to interact with each other.
- The CTS also specifies the rules for type visibility and access to the members of a type, i.e. the CTS establishes the rules by which assemblies form scope for a type, and the Common Language Runtime enforces the visibility rules.
- The CTS defines the rules governing type inheritance , virtual methods and object lifetime.
- Languages supported by .NET can implement all or some common data types.
- CTS Supports two different types : Value and Reference

Value type : Primitive or built in data types of the programming language are Value types. It includes user defined types and enumerations.

- They are used to store the value of the Variable. This includes Character, Integer and other data types. These data types are passed to the method by using by value method.
- Instances of value types are either allocated on the stack or allocated inline in a structure.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



- Value types are data types whose objects are represented by the object's actual value. If an instance of a value type is assigned to a variable, that variable is given a fresh copy of the value.

Reference Types :Reference types store a reference to the value's memory address, and are allocated on the heap.

- Reference types can be self-describing types, pointer types, or interface types.
- The type of a reference type can be determined from values of self-describing types.
- Reference types are data types whose objects are represented by a reference (similar to a pointer) to the object's actual value.
- If a reference type is assigned to a variable, that variable references (points to) the original value. No copy is made.

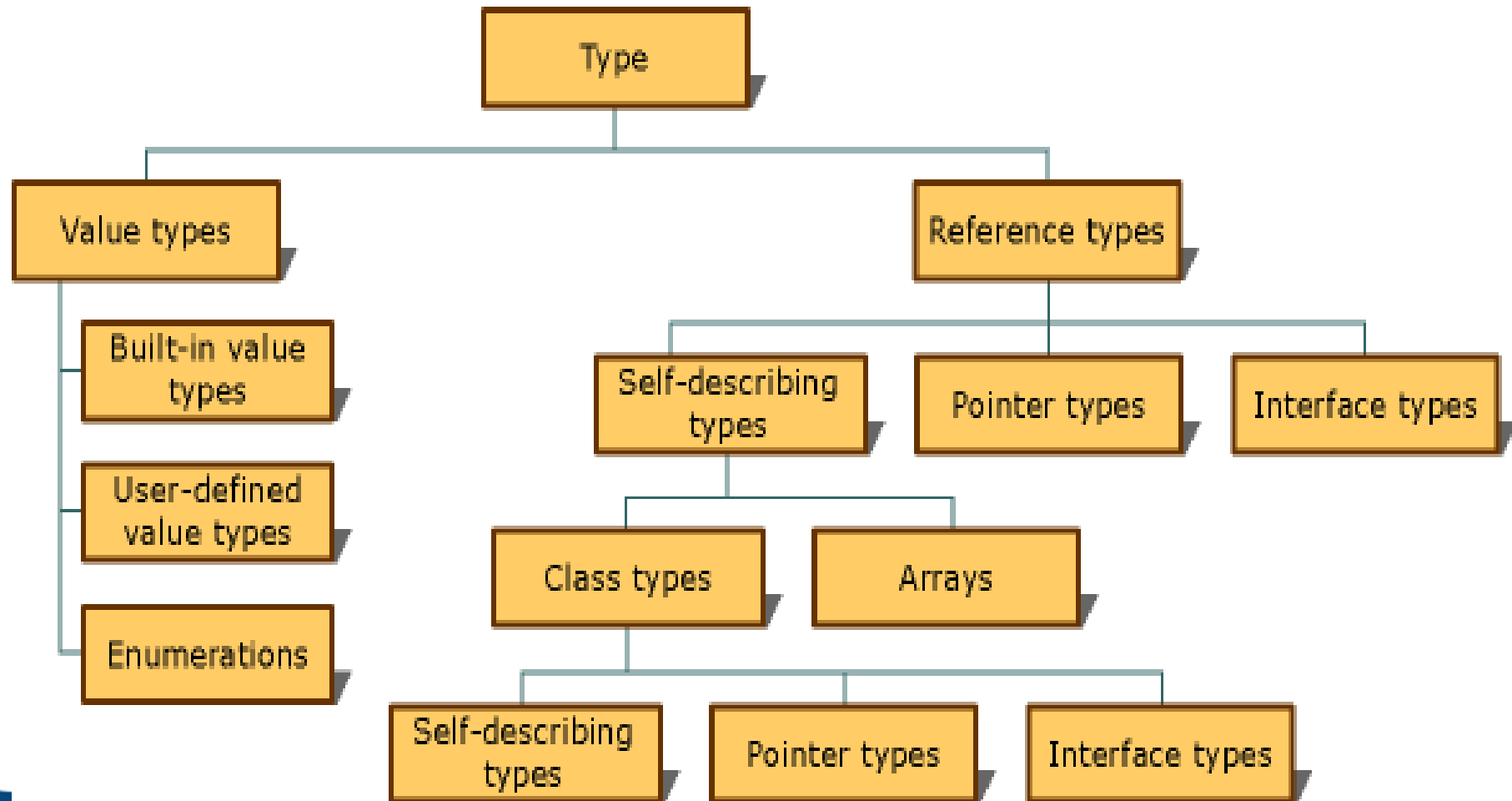


**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



CTS



Class Library

- The .NET Framework class library is a collection of reusable types that tightly integrate with the common language runtime.
- The class library is object oriented, providing types from which your own managed code can derive functionality. This not only makes the .NET Framework types easy to use, but also reduces the time associated with learning new features of the .NET Framework.
- In addition, third-party components can integrate seamlessly with classes in the .NET Framework.
- We can use the same set of classes for performing a specific task in VB as well as VC++
- Using these libraries, you can accomplish different tasks such as database connectivity, string handling, input/output functionality, numerical functions and file handling.
- Library comprises of namespaces and Assemblies.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



- The entire library is split into two parts: Base Class Library(BCL) and the Framework Class Library(FCL).
- Superset of BCL library is called Framework Class library.
- BCL is a standard library which is available to all languages using the framework.
- The classes in **mscorlib.dll** and some of the classes
System.dll and System.core.dll are considered to be a part of the BCL
- The BCL classes are available in both .NET framework as well as its alternative implementations including .NET Compact Framework, Microsoft Silverlight and Mono
- .NET includes the BCL in order to encapsulate a large number of common functions, such as file reading and writing, graphic rendering, database interaction, and XML document manipulation, which makes the programmer's job easier.
- The **Framework Class Library (FCL)** is a standard library and one of two core components of Microsoft NET Framework. The FCL is a collection of thousands of reusable classes (within hundreds of namespaces), interfaces and value types.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Namespaces

- It helps you to create logical groups of related classes and interfaces that can be used by any language targeting the .NET framework.
- It allows us to organize the classes so that they can be easily accessed in other applications.
- It can also be used to avoid name conflicts between classes that have the same names.
- We can access the classes belonging to a namespace by importing namespace into an application.
- Framework uses . (dot) as a delimiter between classes and namespaces.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Assemblies

- An assembly is a single deployable unit that contains all the information about the implementation of classes, structures and interfaces.
- The new way of packaging the executable code in the .NET framework is Assemblies.
- It is a self describing unit and all the programs in .net constructed from these assemblies only.
- The .EXE or .DLL is called Assembly.
- It stores all the information about itself.
- This information is called assembly metadata and includes the name and version number of the assembly, security information, information about the dependencies and a list of the files that constitute the assembly.
- Namespaces are also stored in assemblies.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Assembly Contains the following :

- **Assembly Manifest** – Metadata that describes the assembly and its contents
- **Common Intermediate Code** – Compiled version of Source Code is CIL
- **Type Metadata** – Defines all types, their properties and methods, and most importantly, public types exported from this assembly
- **Resources** – Icons, images, text strings and other resources



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



- Assemblies and metadata provide the CLR with the information required for executing application.
- Assemblies also play an important role in deployment and versioning.
- Assemblies can be .DLL or .EXE but they differ in their content.
- It also consists of assembly metadata known as Assembly Manifest.
- Every assembly has only one entry point.
- Information about the assembly version, identity (name), resources needed by the assembly and the scope of the assembly are contained in the manifest file.
- CLR loads and executes the IL code which is in portable executable file only if assembly contains the manifest.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



- Assemblies can be private or shared.
- Private assembly can be accessible by a single application. On other way, Shared one is shared by multiple application.
- Depends upon the creation, Assemblies are classified into two namely static and dynamic, Static Assembly is created when we compile our code using compiler.
- It is stored in the hard disk in the form portable executable file. The dynamic one will be created in the fly during run time and it occupies only memory.
- The assembly can be stored in a single file or multiple files. Single file means that the CIL code, meta data, resources and Assembly metadata all are in single file only.
- On the other way, they can be separated in different file. That is called multiple file assembly.
- Assembly linker is used to link all the modules and resource files in the case of multiple structures.
- Sharing of the assemblies by multiple application is also possible by adding assemblies to the Global Assembly Cache.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Assembly Manifest

- Every assembly, whether static or dynamic, contains a collection of data that describes how the elements in the assembly relate to each other.
- The assembly manifest contains this assembly metadata.
- An assembly manifest contains all the metadata needed to specify the assembly's version requirements and security identity, and all metadata needed to define the scope of the assembly and resolve references to resources and classes.
- It contains the assembly name, version number, culture, and strong name information that makes up the assembly's identity.
- It also contains list of all files in the assembly, type reference information and list of other assemblies that are referenced by the assembly.
- The assembly manifest can be stored in either a PE file (an .exe or .dll) with intermediate language (IL) code or in a standalone PE file that contains only assembly manifest information.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Metadata

- In the past, a software component (.exe or .dll) that was written in one language could not easily use a software component that was written in another language. COM provided a step towards solving this problem.
- The .NET Framework makes component interoperation even easier by allowing compilers to emit additional declarative information into all modules and assemblies.
- This information, called metadata, helps components to interact seamlessly.
- Metadata is binary information describing your program that is stored either in a common language runtime portable executable (PE) file or in memory.
- Metadata is stored in one section of a .NET Framework portable executable (PE) file, while intermediate language (IL) is stored in another section of the PE file.
- Every type and member that is defined and referenced in a module or assembly is described within metadata.
- When code is executed, the runtime loads metadata into memory and references it to discover information about your code's classes, members, inheritance, and so on.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



- The metadata portion of the file contains a series of table and heap data structures.
- The MSIL portion contains MSIL and metadata tokens that reference the metadata portion of the PE file.

Metadata stores the following information:

- Description of the assembly.
 - Identity (name, version, culture, public key).
 - The types that are exported.
 - Other assemblies that this assembly depends on.
 - Security permissions needed to run.
- Description of types.
 - Name, visibility, base class, and interfaces implemented.
 - Members (methods, fields, properties, events, nested types).
- Attributes.
 - Additional descriptive elements that modify types and members.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Application Types

CLR supports the following type of applications:

- ASP.NET Web applications: These include dynamic and data driven browser based applications.
- Windows Form based applications: These refer to traditional rich client applications.
- Console applications: These refer to traditional DOS kind of applications like batch scripts.
- Component Libraries: This refers to components that typically encapsulate some business logic.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



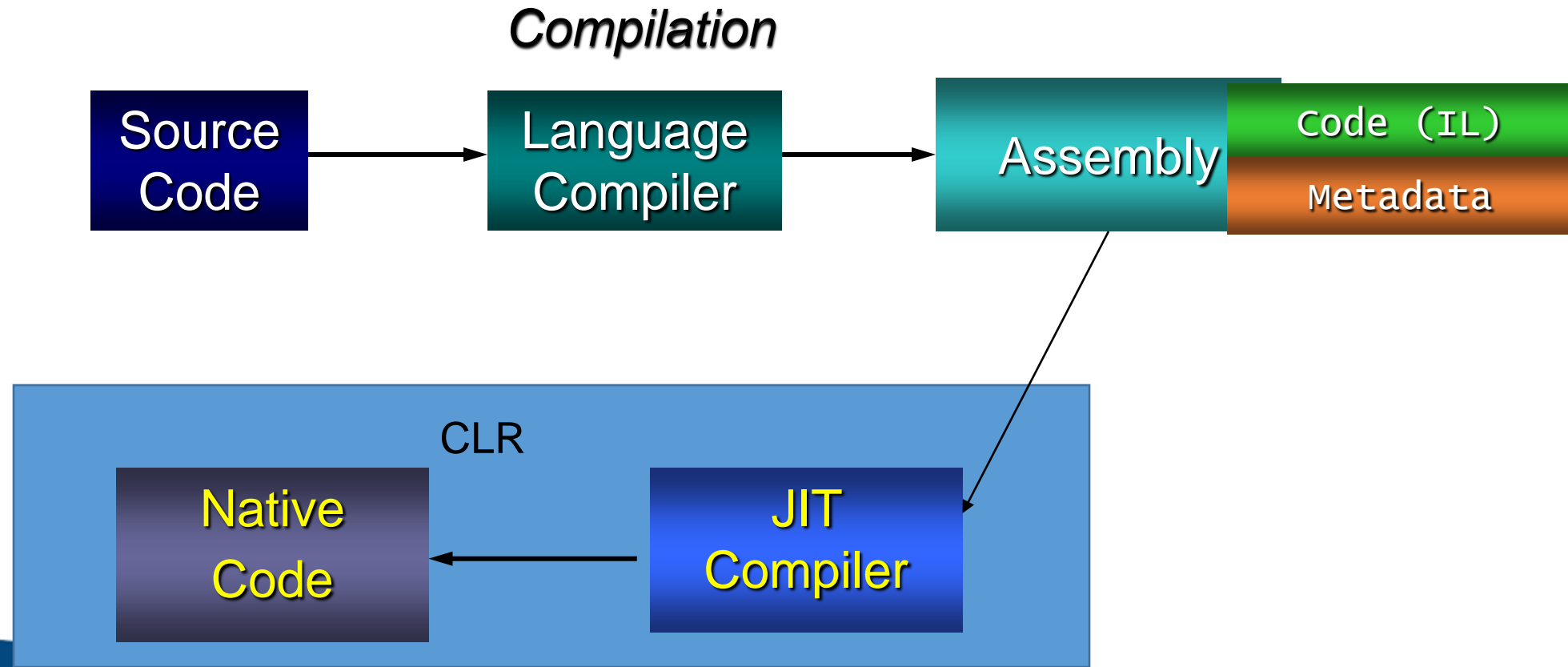
- Windows Custom Controls: As with traditional ActiveX controls, you can develop your own windows controls.
- Web Custom Controls: The concept of custom controls can be extended to web applications allowing code reuse and modularization.
- Web services: They are “web callable” functionality available via industry standards like HTTP, XML and SOAP.
- Windows Services: They refer to applications that run as services in the background. They can be configured to start automatically when the system boots up.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013





Execution



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Managed code Execution Process

- When we compile the code, the code gets translated into IL (Intermediate language). No matter which language has been used to develop.
- IL includes instructions for loading, storing, initializing, and calling methods on objects, as well as instructions for arithmetic and logical operations, control flow, direct memory access, exception handling, and other operations
- The compiler also produces the metadata about the program during the process of compilation.
- Metadata contains the description of the program, the dependencies and the version of the components used in the program.
- The IL and metadata are linked in an assembly.
- The compiler creates .EXE or .DLL file.
- When you execute the .EXE or .DLL file the code and all other relevant information from the base class library is sent to the class loader. This loads the code in the memory.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



- Before the code can be executed, .NET framework needs to convert the IL into native or specific code.
- The JIT compiler translates the code from IL into managed native code.
- JIT compiler compiles only the code that is required during execution instead of compiling the complete IL code.
- When an uncompiled method is invoked during execution, the JIT compiler converts that into native code.
- During JIT compilation, the code is also checked for type safety.
- After that, converted code is sent to the .NET runtime manager.
- .NET runtime manager executes the code. While executing a security check is performed .

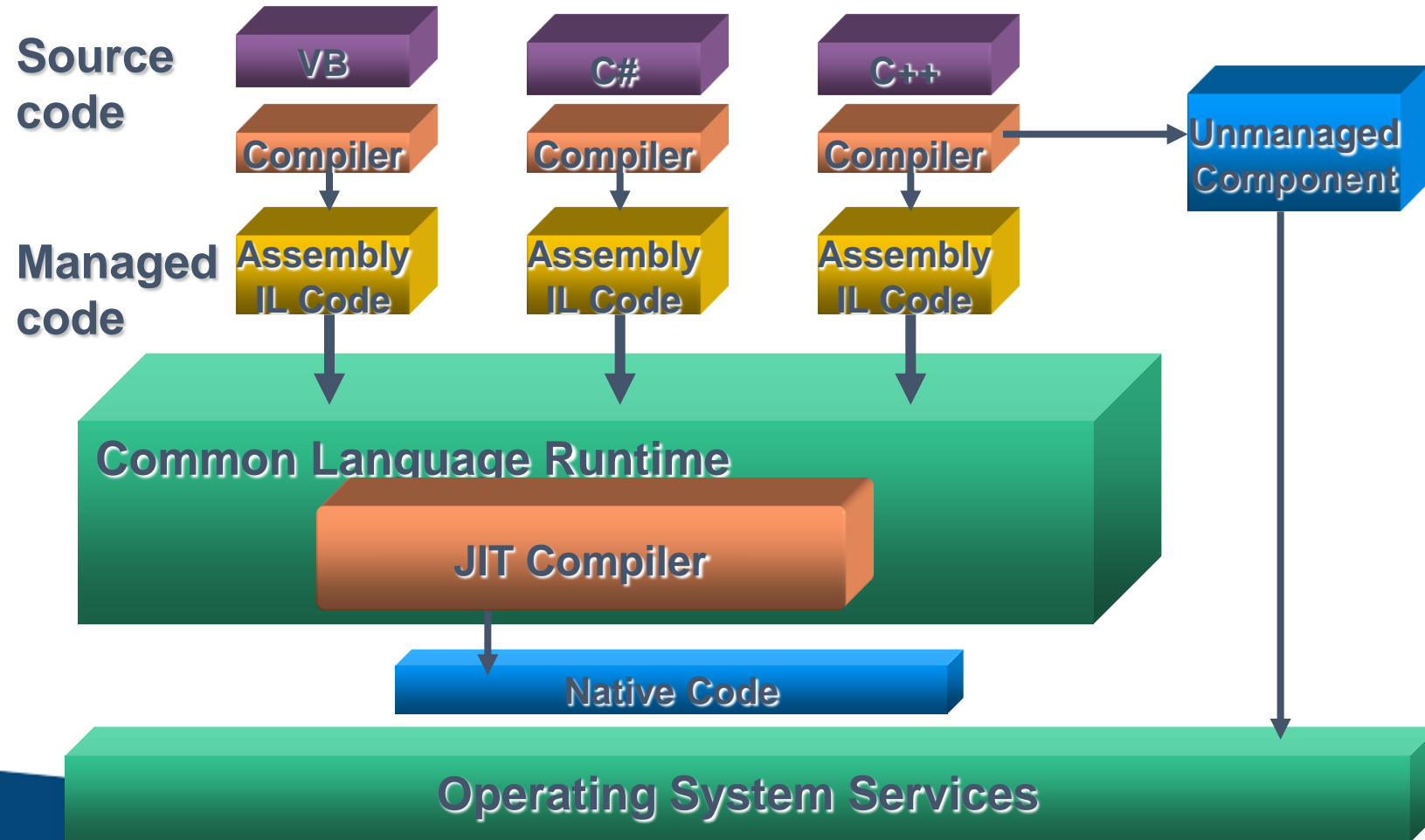


**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



CLR: Execution Model



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



JIT Types

1. Pre JIT

It converts all the code in executable code and it is slow

2. Econo JIT

It will convert the called executable code only. But it will convert code every time when a code is called again

3. Normal JIT

It will only convert the called code and will store in cache so that it will not require converting code again. Normal JIT is fast



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Introduction to .NET Core

- .NET Core is a general-purpose framework that may be used to build software applications for Windows, Linux, and MacOS.
- Cross-platform, open source platform for building many kinds of applications such as Web, Desktop, Mobile, Cloud, Gaming, Internet of Things
- The runtime is designed to support multiple programming languages. C#, F#, and Visual Basic languages are supported by Microsoft.
- .NET Core is Sharable.
- .NET Core is Lightweight.
- The Common Language Runtime (CLR) is the foundation all .NET apps are built on.
- .NET supports both Ahead-Of-Time (AOT) and Just-In-Time (JIT) compilation models.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Module 1 - Questions

- 1) What is .NET framework.
- 2) What do you understand by the term CLR and its basic functionality.
- 3) Differentiate between managed code and unmanaged code.
- 4) Briefly explain the four aspects of CLI.
- 5) Explain the concept of Assemblies
- 6) Mention the different type of applications supported by .NET.
- 7) Explain the architecture of a CLR in detail.
- 8) Write down the various steps involved in managed execution process of a .NET framework.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013

