Agenda

- What is .NET Framework?
- .NET Framework Objectives.
- Components of .NET Framework.

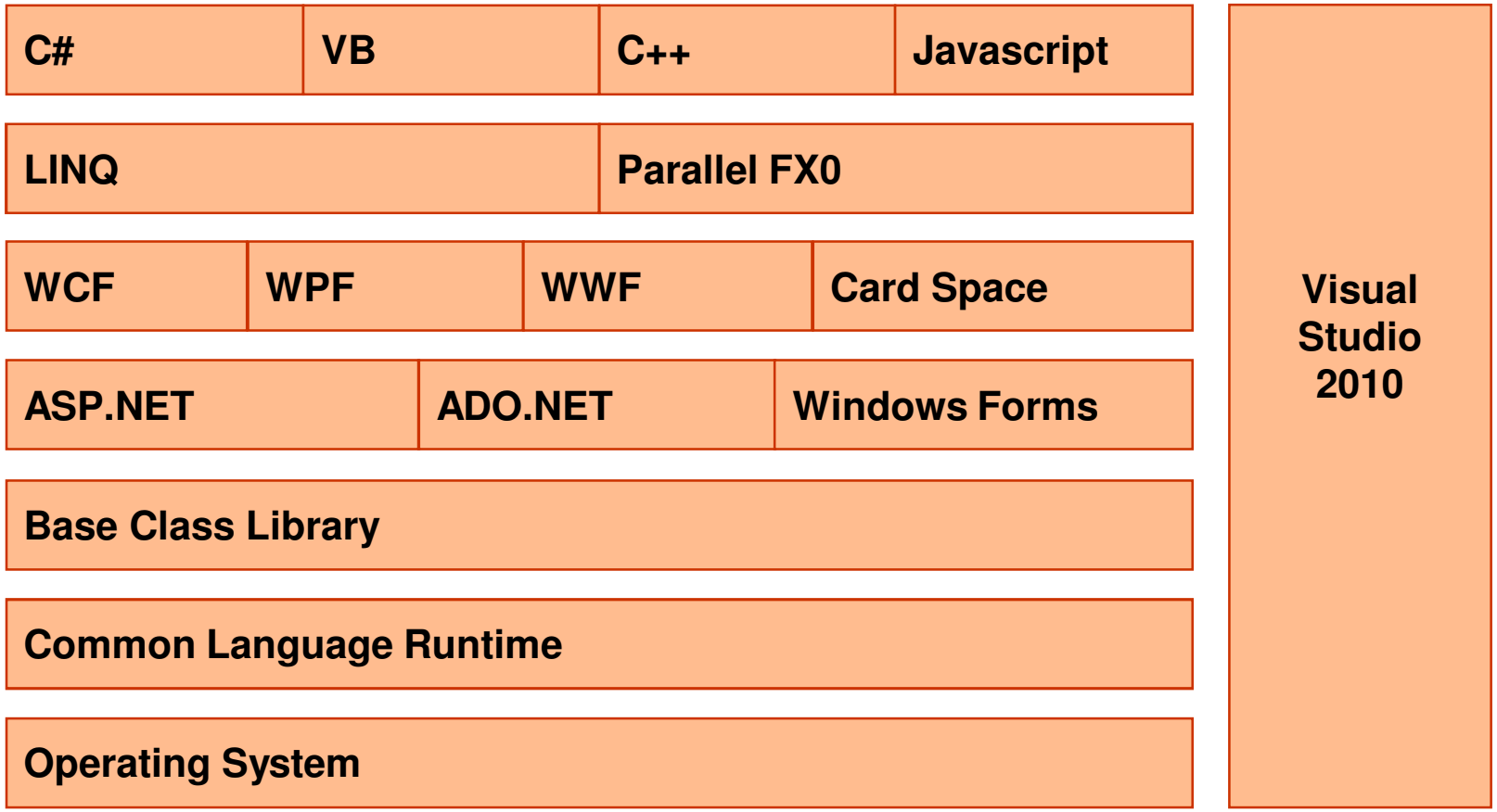
What is .NET Framework?

- .NET Framework is an integral windows component that supports building and running the next generation of applications and Web services.
- The .NET Framework provides a managed execution environment, simplified development and deployment and integration with a wide variety of programming languages.
- Microsoft Visual Studio® 2010 and the Microsoft .NET Framework allow developers to develop traditional windows client applications, distributed components, client server applications and much more.

.NET Framework Objectives

- The .NET Framework is designed to:
  - Provide a consistent object-oriented programming environment.
  - Provide a code-execution environment that minimizes software deployment and versioning conflicts.
  - Provide a code-execution environment that promotes safe execution of code, including code created by unknown or semi-trusted third parties.
  - Provide a code-execution environment that eliminates the performance problems of scripted or interpreted environments.
  - Provide consistency across widely varying types of applications, such as Windows-based applications and Web-based applications.
  - Build all communication to industry standards ensuring that code based on the .NET Framework can integrate with any other code.

.NET Framework



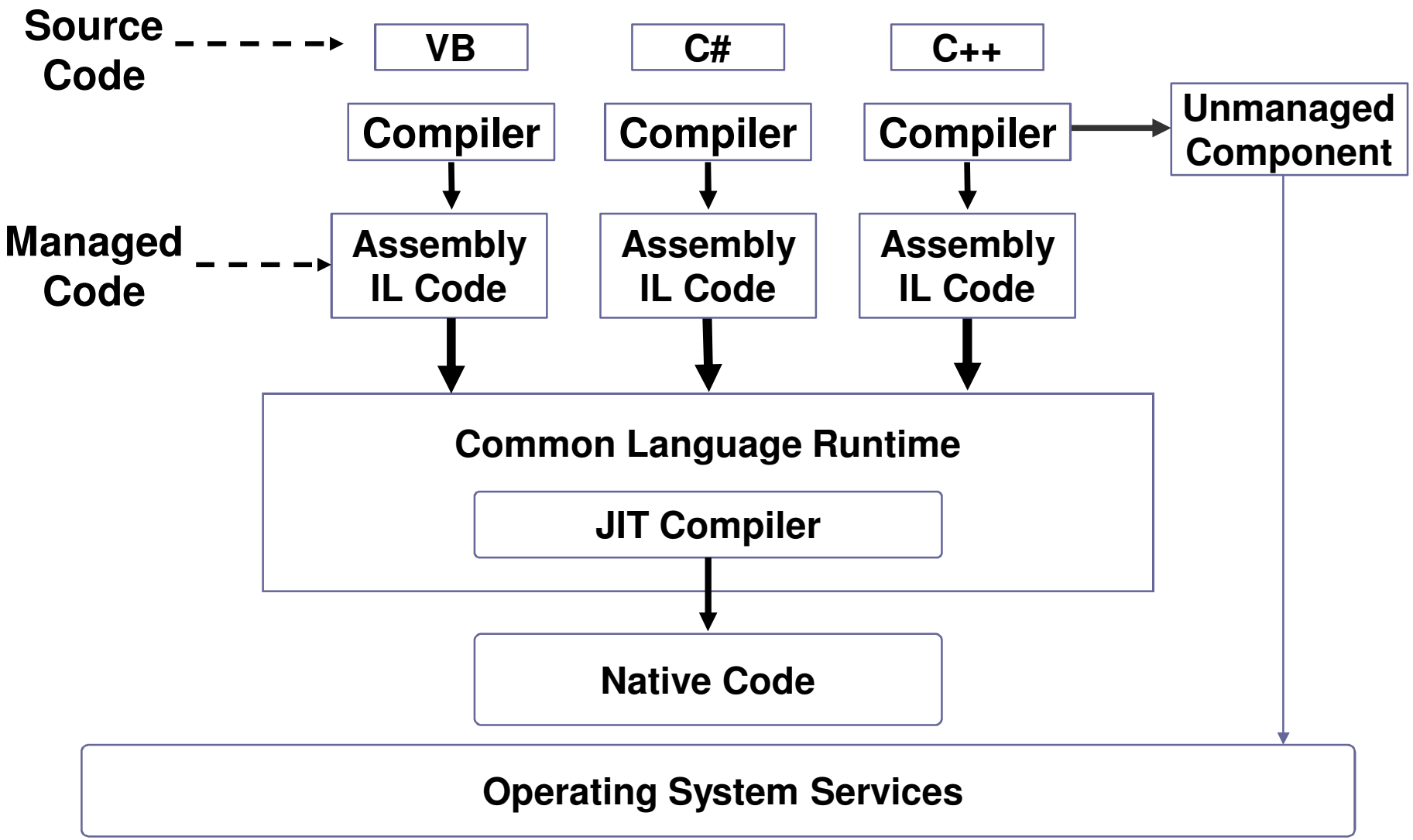
.NET Framework Components

- The .NET Framework components consist of:
  - Common Language Runtime(CLR)
  - .NET Framework Class Library
  - Common Type System(CTS)
  - Common Language Specification(CLS)
  - Assemblies
  - Windows Forms
  - ASP.NET
  - ActiveX Data Objects(ADO.NET)
  - Windows Workflow Foundation(WF)
  - Windows Presentation Foundation(WPF)
  - Windows Communication Foundation(WCF)
  - Windows CardSpace
  - Language Integrated Query (LINQ)
  - Parallel Programming

Common Language Runtime (CLR)

- CLR ensures:
  - A language-neutral development and execution environment that provides services to help "manage" application execution.
  - A common runtime environment for all .NET languages.
  - Use of Common Type System (strict-type & code-verification).
  - Use of metadata for safe execution.
  - Memory allocation and garbage collection.
  - Intermediate Language (IL) to native code compilers.
    - Compiles MSIL code into native executable code.
  - Security and interoperability of the code with other languages.
- Over 15 languages supported today.
  - C#, VB, Jscript, Visual C++ from Microsoft.
  - Perl, Python, Smalltalk, Cobol, Haskell, Mercury, Eiffel, Oberon, Oz, Pascal, APL, CAML, Scheme, etc.
- Enables cross-language interoperability.
  - Common Language Specification describes interoperability requirements.

CLR: Execution Mode



.NET Framework Class Library

- The Class Library is a comprehensive, object-oriented collection of reusable types.
- The Class Library is organized in a hierarchy of namespaces.
- These class types can be used to develop applications that include:
  - Traditional command-line applications.
  - Graphical User Interface (GUI) applications.
  - Applications based on the latest innovations provided by ASP.NET.
    - Web Forms.
    - XML Web services.

Common Type System

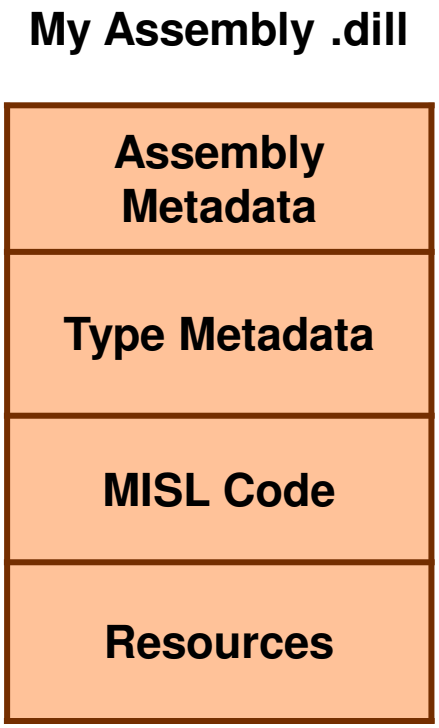
- Defines how types are declared, used, and managed in runtime.
- Is also an important part of the runtime's support for cross-language integration.
- Performs the following functions:
  - Establishes a framework that helps enable cross-language integration, type safety, and high performance code execution.
  - Provides an object-oriented model that supports the complete implementation of many programming languages.
  - Defines rules that languages must follow, which helps ensure that objects written in different languages can interact with each other.

Common Language Specification (CLS)

- The CLS is a set of basic language features needed by many applications.
- The CLS rules define a subset of the Common Type System. All the rules that apply to the Common Type System apply to the CLS, except where stricter rules are defined in the CLS.
- The CLS helps to enhance and ensure language interoperability by defining a set of features that the developers can rely on to be available in a wide variety of languages.
- Components that adhere to the CLS rules and use only the features included in the CLS are said to be CLS-compliant components.
- Most of the members defined by types in the .NET Framework Class Library are CLS-compliant.

Assemblies

- Assemblies are the building blocks of .NET Framework applications.
- They form the fundamental unit of deployment, version control, reuse, activation scoping, and security permissions.
- An assembly is a collection of types and resources that are built to work together and form a logical unit of functionality.
- They provide the CLR with the information it needs to be aware of type implementations. To the runtime, a type does not exist outside the context of an assembly.
- An assembly can be Shared or Private:
  - A Private Assembly is available only to the application for which it is created.
  - A Shared Assembly is available to more than one (multiple) applications. Shared Assemblies are stored in the assembly cache (GAC) using the .NET utility Gacutil.exe or Regasm.exe.



Assemblies – Features (1 of 2)

- Self-Describing:
  - Contains metadata that stores information about the assembly, such as the data type of the variables and methods declared in the assembly.
- Side-by-Side execution:
  - Enables you to install multiple versions of the same assembly in one application.
- Version dependency:
  - An assembly manifest is used to maintain versions of the resources in an assembly. Manifest is a part of the assembly that contains metadata. When you reference an application the version of the referenced assembly is stored in the manifest of the application.

Assemblies – Features (2 of 2)

- Application domain:
  - Enables you to execute multiple applications that are independent of each other. All applications are part of the same process. An error in one application does not affect another application even though they are part of the same process.
- Zero-impact installation:
  - You don’t have to register the assembly with the operating system. You can use xcopy or copy commands to install an assembly.

ASP.NET

- ASP.NET is Web based application development model ,which is used deliver interactive, data-driven web applications over the internet.
- ASP.NET applications using any CLR compliant language such as VB, C#, Visual C++, and J#.
- ASP.NET also consists of large number of controls for assembling, configuring, and manipulating code to create HTML pages.

Windows Forms

- The graphical representation of any window displayed in an application.
- Uses any CLR compliant language such as VB, C#, Visual C++, and J#.
- Consists of a large number of inbuilt controls for developing windows based applications.
- Develops applications easily and quickly.

ADO.NET

- A technology used for working with data and databases.
- Provides access to different data sources like SQLServer, Oracle, MySql etc.
- ADO.NET provides functionality to developers writing managed code similar to the functionality provided to native COM developers by ADO.
- Data sharing consumers can use ADO.NET to connect to the data sources and retrieve, manipulate and update data.

Windows Presentation Foundation (WPF)

- Provides building interfaces that include two or three-dimensional graphics, animations, and web like characteristics.
- Used for standalone and browser hosted applications.
- A next generation Presentation system for building windows client applications.
- WPF is included in the .NET Framework.
- Uses XAML markup to implement the appearance of an application.

Windows Communication Foundation (WCF)

- A service oriented technology for developing applications.
- The service-oriented design results in distributed systems that runs between services and clients.
- WCF is built in .NET Framework.

Windows Workflow Foundation (WWF)

- Provides a programming model for building workflow-based applications in Windows.
- The main feature of WWF is the separation between the business code and actual implementation code.

Language Integrated Query

- Native querying capabilities to .NET languages using a syntax similar to SQL.
- It is purely C# language and is not an SQL statement.
- LINQ is not restricted to accessing only relational databases.

Parallel Programming

- Leveraging multicores or multiple processors is called Parallel Programming.
- This is a subset of the broader concept of multithreading.
- .NET 4.0 enhances support for Parallel Programming by providing new class library types and new diagnostic tools.

C# (C Sharp)

- Microsoft C# (pronounced C Sharp) developed by Microsoft Corporation, USA.
- New programming language that runs on the .NET Framework.
- An evolution of Microsoft C and Microsoft C++.
- C# is simple, modern, type safe, and object oriented.
- C# code is compiled as managed code.
- Combines the best features of Visual Basic, C++, and Java.

C# Features

- Simple
- Modern
- Object-Oriented
- Type-safe
- Versionable
- Compatible
- Inter-operability
- Secure
- Multi-threaded

Visual Studio 2010

- Integrated Development Environment:
  - Visual Basic 2010:
    - Many language enhancements.
    - Inheritance, overloading, free threading.
  - Visual C++:
    - Integration with .NET Framework with managed extensions (classes).
  - C#:
    - New development language.
    - Based on C/C++ with garbage collection/memory managemnt.