

Distribution and Integration Technologies

.NET Architecture

Traditional Architectures

Web applications

Local applications

Distributed applications

Other applications

GUI Services

Network Services

Scripts Web

Data Access

Other Services

Libraries

Execution environment (Posix, Win32, ...)

Operating System

Java Architecture

Web applications

Local applications

Distributed applications

Other applications

Swing

Enterprise
Java Beans

Java Server
Pages

JDBC

Others

Standard Java Packages

Java Virtual Machine (JVM)

Operating System

.NET Architecture

Web applications

Local applications

Distributed applications

Other applications

Windows Forms

Enterprise Services

ASP.NET

ADO.NET

Others

.NET Framework Class Library (FCL)

Common Language Runtime (CLR)

Operating System

Common Language Runtime (CLR)

- ❖ All .NET applications use the CLR
- ❖ The CLR is OO
- ❖ It is independent from high level languages
- ❖ The CLR supports:
 - A common set of data types for all languages (CTS)
 - An intermediate language independent from the native code (CIL)
 - A common format for compiled code files (*assemblies*)
- ❖ All the software developed using the CLR is known as *managed code*

Common Type System (CTS)

❖ All the languages able to generate code for the CLR make use of the CTS (the CLR implements the CTS)

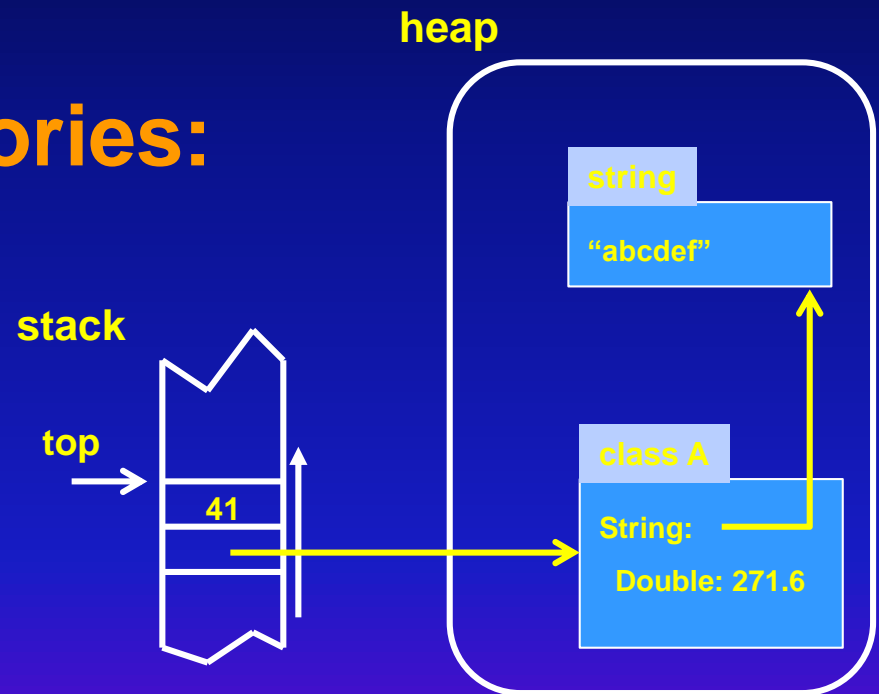
❖ There is 2 type categories:

- **Value**

- Simple types
- Allocated in the *stack*

- **Reference**

- Complex types
- Allocated in the *heap*
- Destroyed automatically by *garbage collection*



The CTS

Object

Class

Interface

Array

String

Delegate

Others

Reference types

ValueType

Byte

Char

Int16

Int32

Int64

Decimal

Boolean

UInt16

UInt32

UInt64

Structure

Others

Single

Double

Enum

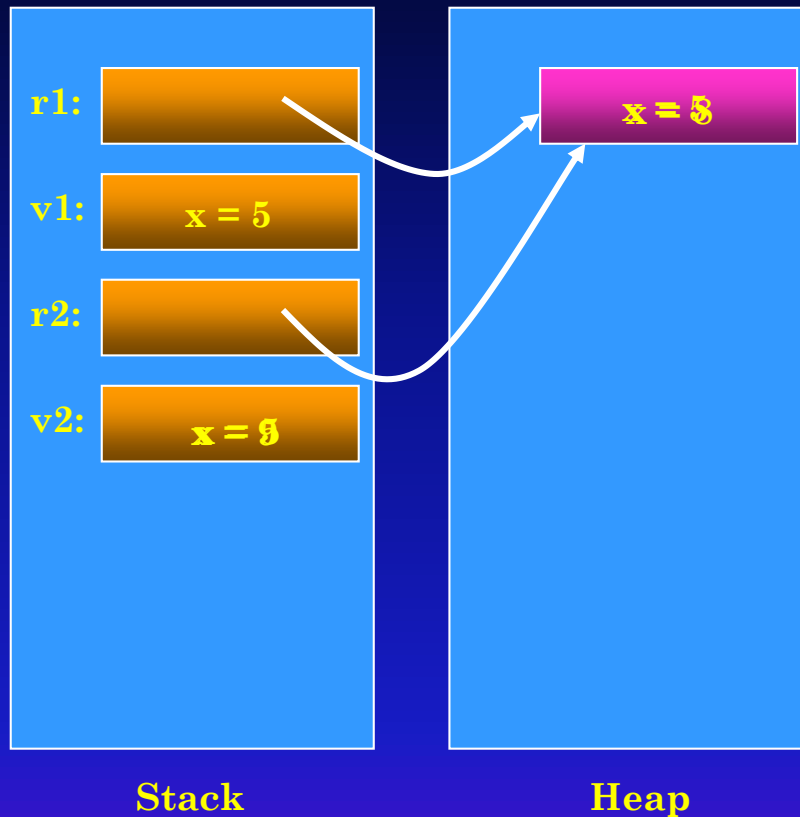
Value types

Value and Reference Types

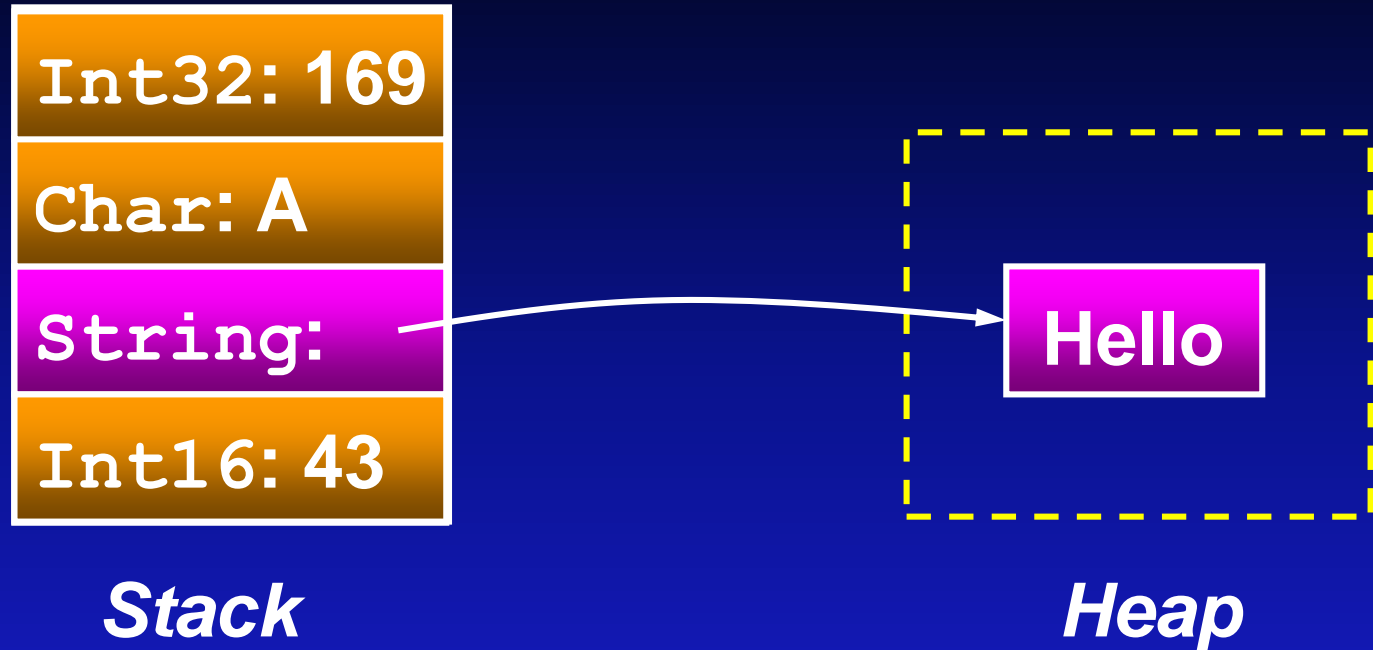
```
class Ref { public int x; }  
struct Val { public int x; }
```

...

```
void Method ( ) {  
  Ref r1 = new Ref();  
  Val v1 = new Val();  
  r1.x = 5;  
  v1.x = 5;  
  ...  
  Ref r2 = r1;  
  Val v2 = v1;  
  r2.x = 8;  
  v2.x = 9;  
  ...  
}
```



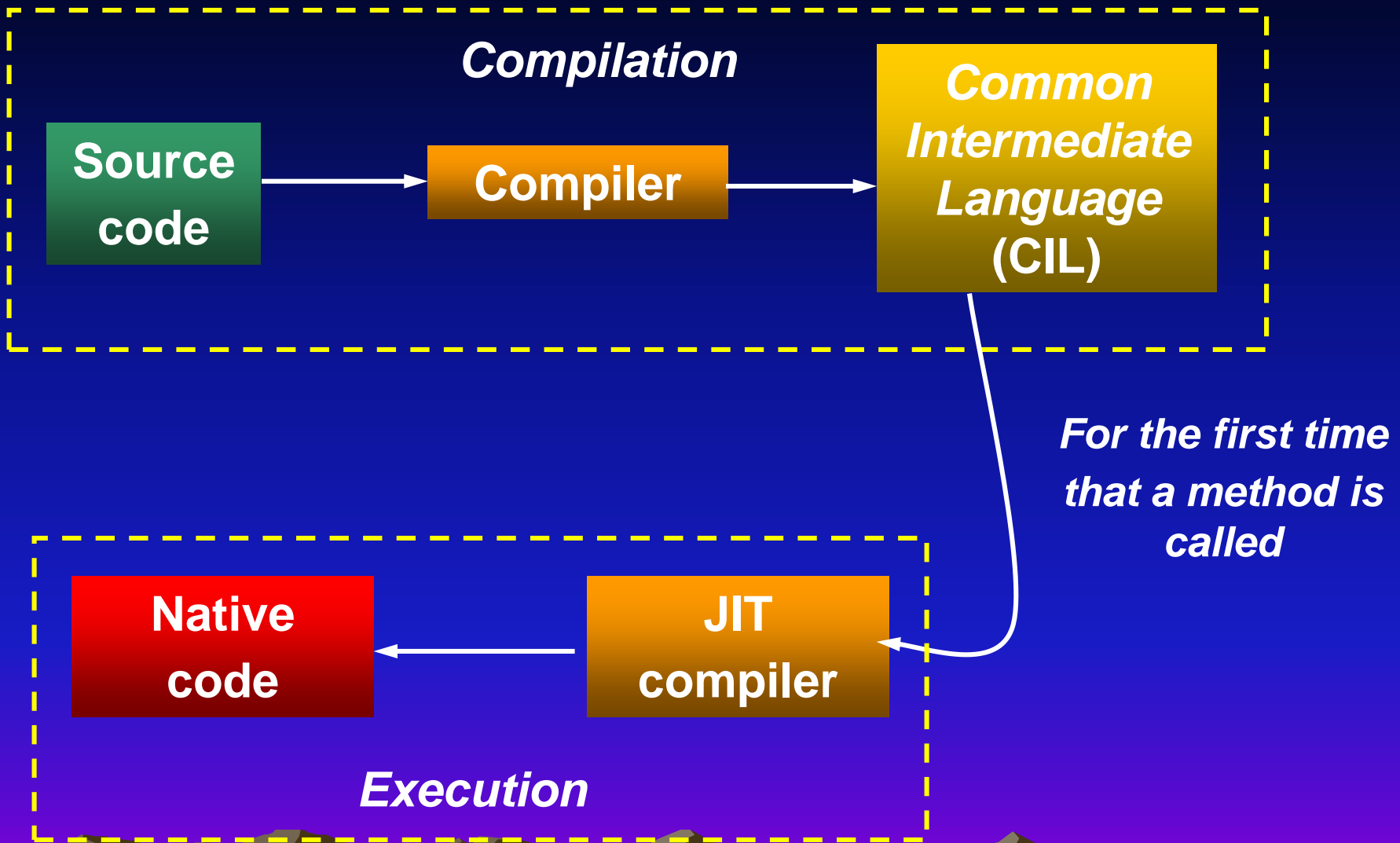
Value and Reference Types



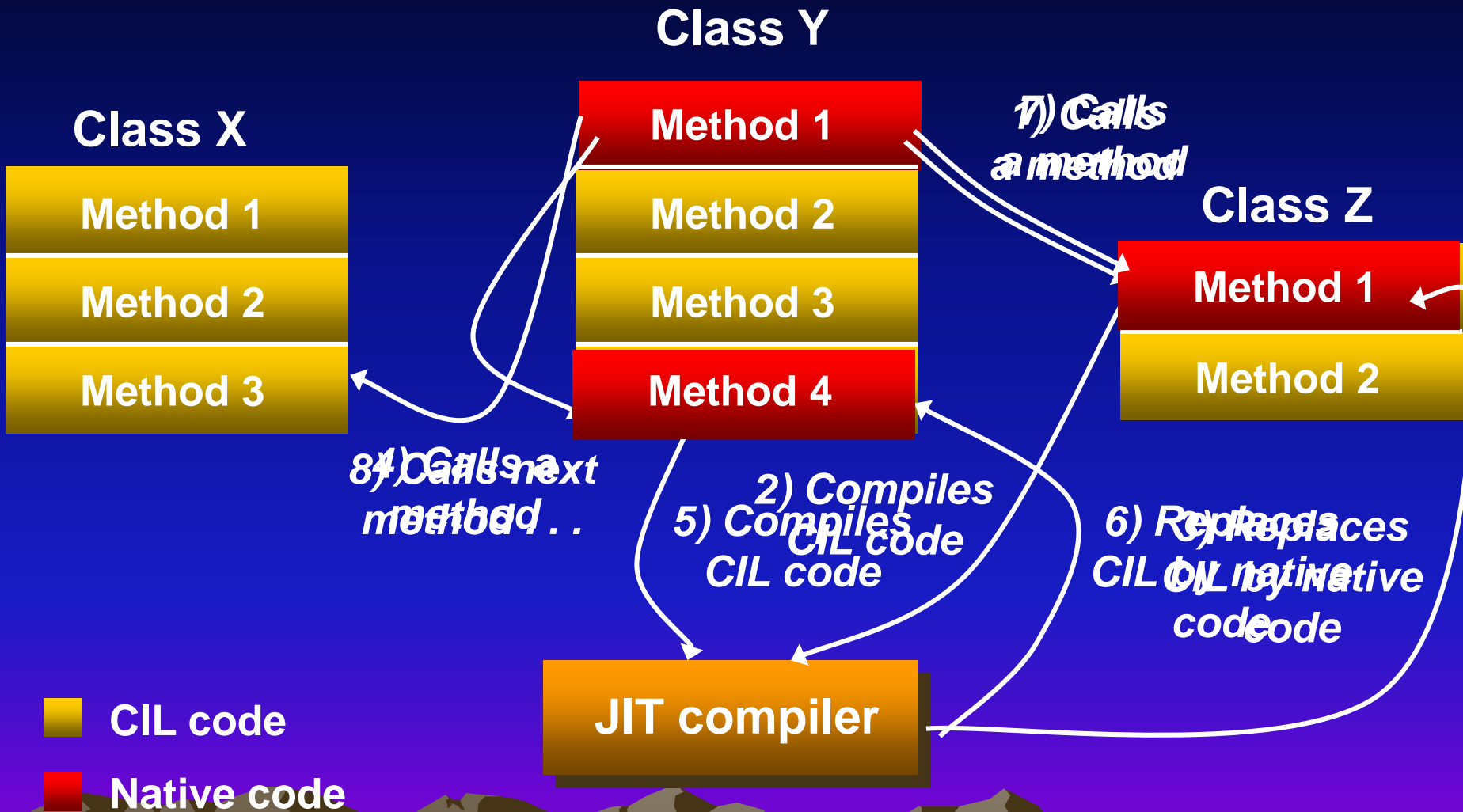
Value type

Reference type

Intermediate Code



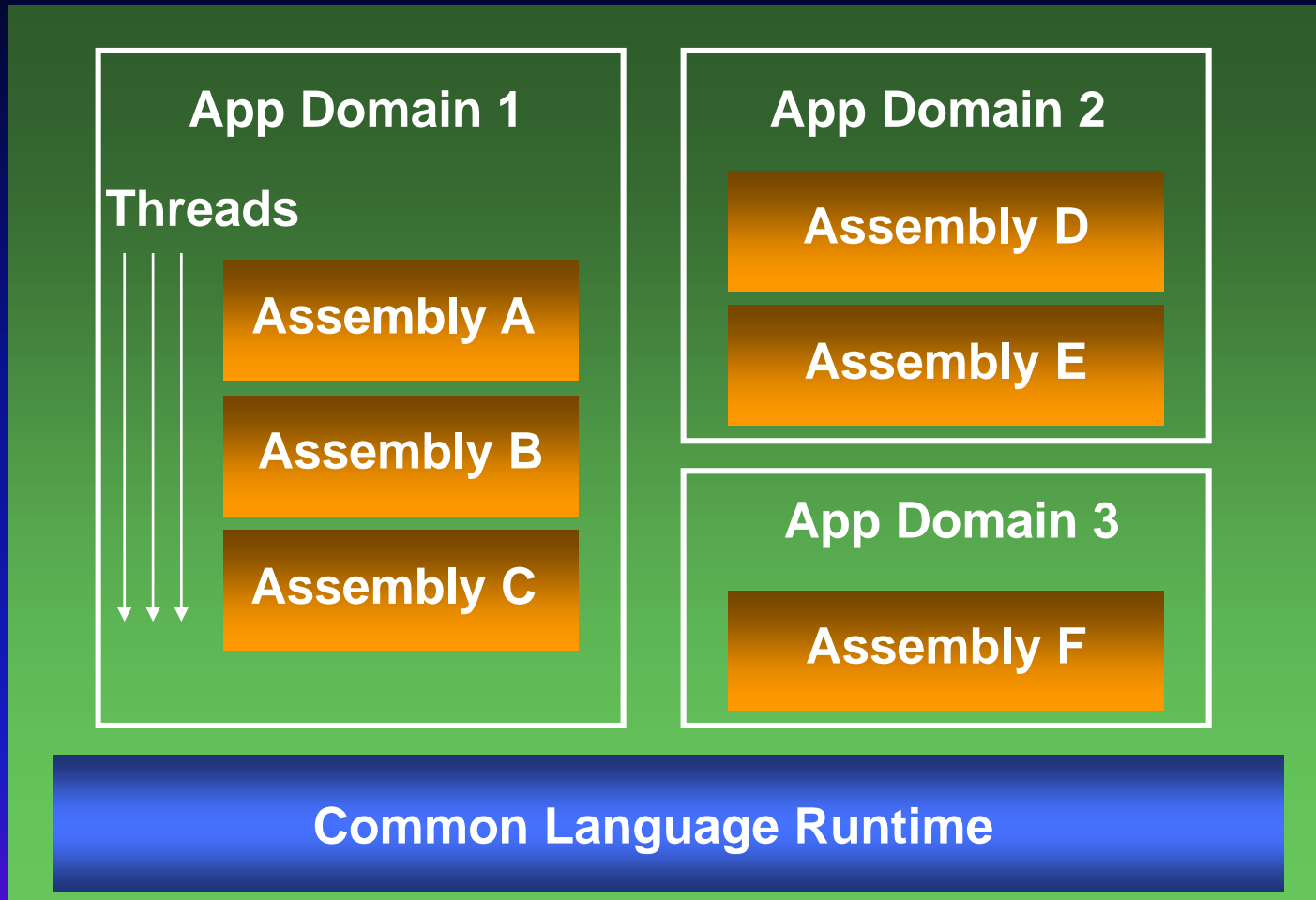
Execution in the CLR



Application domains

- ❖ Several .NET applications can run in the same process (CLR) but in different *domains*
- ❖ The domains are isolated from each other
- ❖ Domains can communicate using some inter process communication techniques
- ❖ Domains can run several *threads*
- ❖ An application can be made multithreaded easily

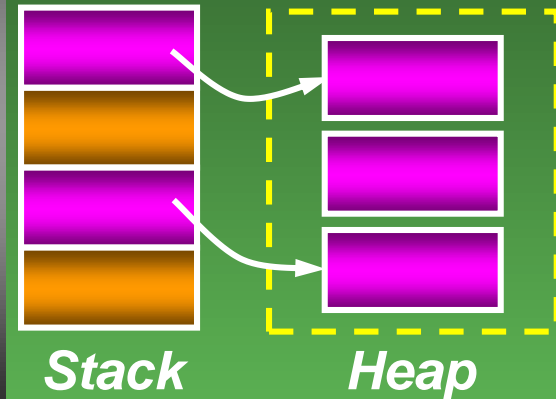
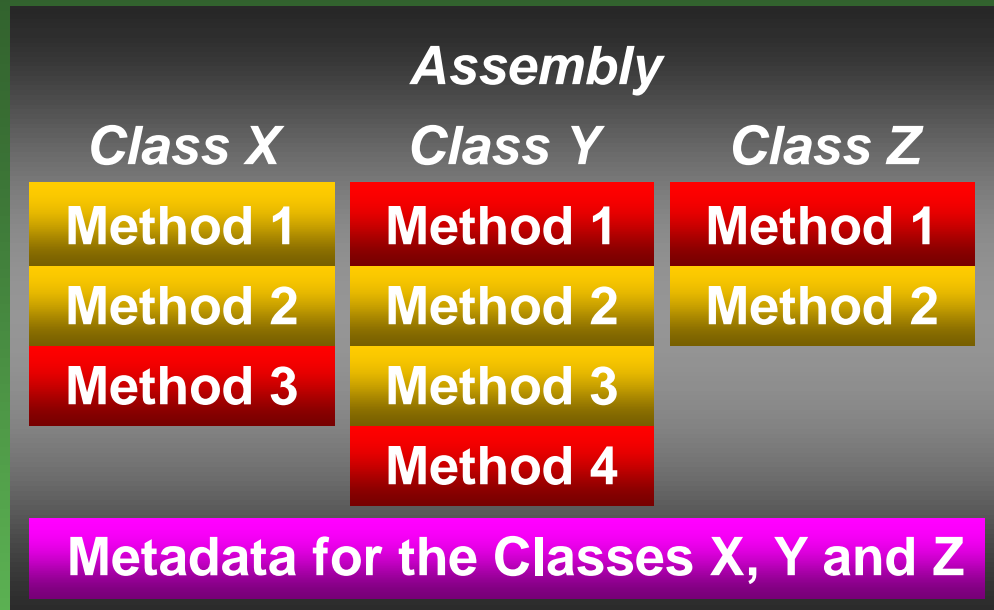
Application domains



An OS process

Application domain and CLR

App Domain



Loader

JIT Compiler

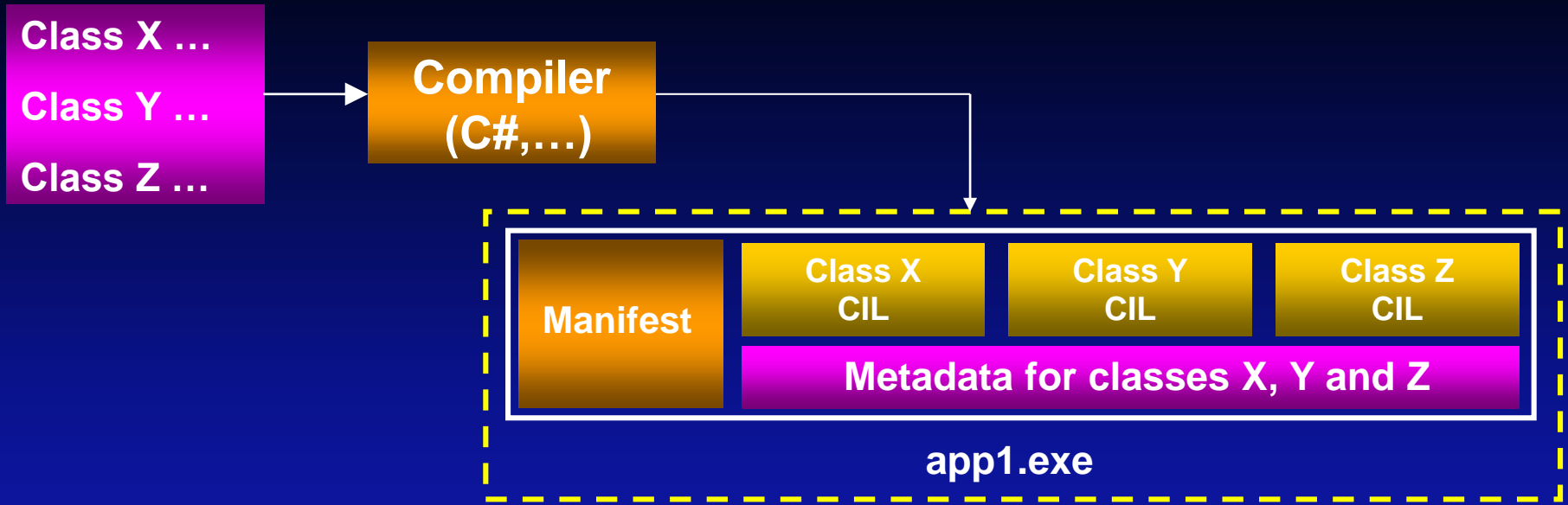
*Garbage
Collector*

Common Language Runtime

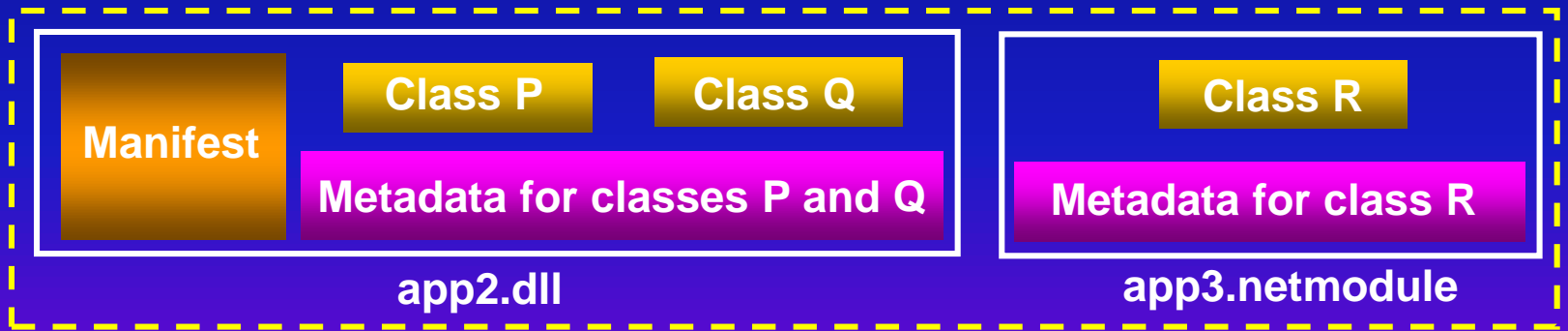
Assemblies

- ❖ When we do a compilation from a high level .NET language we create modules of *managed code*
- ❖ One or more modules can constitute an *assembly* (one or more files)
- ❖ The modules also contain metadata describing types, methods, fields, ...
- ❖ One of the *assembly* modules contains a manifest, describing the *assembly* components (modules)
- ❖ A private *assembly* gets installed by a simple copy
 - There is no registry entries, include files, etc.

Assemblies



Assembly A



Assembly B

Languages

- ❖ Any language capable of supporting a well defined subset of the CLR/CTS (*CLS - Common Language Specification*) can be used as a high level .NET language
 - One of them is the new C# language (follows closely the CTS and CLR specifications)
 - From Microsoft: VB.NET, C++.NET, J#, F# and JScript.NET
 - Others: APL, COBOL, Pascal, Eiffel, Haskell, ML, Oberon, Perl, Python, Scheme, Smalltalk, Fortran, etc.

Framework Class Library (FCL)

- ❖ It is very broad
- ❖ Accessible from any .NET language
- ❖ It's written in C# (most of it)
- ❖ It's organized in several *namespaces*
 - **Base: *namespace System***
- ❖ Contains thousands of classes, structs and enums
- ❖ Many more thousands of methods, properties and events

Ports to other platforms

For Linux / Solaris / Mac OS:

Mono - CLR (with JIT), C# Compiler and other tools and a very substantial part of FCL
Cross platform, open source, .NET development framework

<http://www.mono-project.com>

For iPhone and iPad (iOS):

Xamarin.iOS – Apps with .NET and C# integrated in Xcode and UI designer

<http://xamarin.com/platform>

For Android:

Xamarin.Android - .NET and C# for Android platform integrated with VS

<http://xamarin.com/platform>