# 一、.NET的体系结构

## 1.JIT编译器，提高性能

JIT编辑器并不是编译整个程序，而是只编译运行部分的代码。

## 2.IL，.net编译语言

IL可以指Intermediate Language，同MSIL(Microsoft Intermediate Language),是将.NET代码转化为机器语言的一个中间语言的缩写。

使用IL不仅支持平台无关性，还支持语言的互操作性。

互操作性是指能将包括C#的任何语言编译成中间语言，如VB，VC++

### 主要特征

a)面向对象和使用接口

b)值类型和引用类型之间的显著差异：

引用类型的实例存储在“托管堆”中，而值类型存储在堆栈中（如果值类型在引用类型中声明为字段，它们内联存储在堆中）

c)强数据类型化：

通用类型系统(CTS)、公共语言规范(CLS)

垃圾回收：.net运行库采用了垃圾回收器，其工作原理是：检查对存储在托管堆上的对象的引用，无引用时将其回收。

进程不能直接访问物理内存，因此一个进程不可能访问分配给另一个进程的内存，也就是一个进程出错不会影响另一个进程。

每个应用程序域大致对应一个应用程序，不同的可执行文件都运行在同一个进程中，数据是可以共享的。

d)使用异常来处理错误

e)使用特性(attribute)

## 3.语言的互操作性

语言互操作性的真正含义是：用一种语言编写的类应能直接与用另一种语言编写的类通信。

1. 用一种语言编写的类应能继承用另一种语言编写的类
2. 一个类应能包含另一个类的实例，而不管两个类是使用什么语言编写的
3. 一个对象应能直接调用用其他语言编写的另一个对象的方法
4. 对象（或对象的引用）应能在方法之间传递
5. 在不同的语言之间调用方法时，应能在调试器中交替调试这些方法调用，即调试不同语言编写的源代码

.NET和中间语言已经实现了这个目标，VS IDE提供了这样的工具

公共语言规范和通用类型系统一起确保语言的互操作性

### 1）VB

VB.NET与C#之间的区别很小，主要是使用花括号还是END语句的个人喜好问题

### 2）VC++

VC++.NET要在.NET framework中运行，只需要在代码开头添加以下命令即可

#using <mscorlib.dll>

### 3）VF#

VF#是一门函数语言，而不是面向对象，有助于用来编写复杂的算法。

### 4）COM和COM+

### 5）Windows运行库

这个运行库可以在VB、C#、C++和JS中使用。它的多样性是通过使用语言投影实现的。

## 4.数据类型

### 1）值类型与引用类型

如果要把自己的类型定义为值类型，就应该把它声明为一个结构。

### 2）预定义类型

整型：sbyte、short、int、long、byte、ushort、uint、ulong

所有的整型都能被赋予十进制或者十六进制的值，后者需要0x前缀

若没有显式的声明，则默认为int，当然也可以直接指定值，如long l=1234L，ulong ul=1234UL。建议使用大写的U和L，否则小写l容易与数字1混淆。

浮点型：double和float，指定float值一般后面加F（f）

decimal：指定值加M

char：除了带单引号的字符字面量外，还可用4位十六进制的Unicode值，带有数据类型转换的整数值或者十六进制数，还可用转义序列

string：字符串中允许包含换行符，在代码中一个字符串可以分两行写，显示的时候也会出现换行

## 5.C#的其他基础

goto语句可以跳转到指定行，但不能跳入像for这样的代码块中，也不能跳出try-catch后面的finially。一般不允许使用goto

Enum.Parse可以将字符串转化为枚举，通常三个参数：第一个是typeof获得的枚举类型，第二个是字符串，第三个表示是否不区分大小写