

**Indroducion**

Microsoft.NET es el conjunto de nuevas tecnologías en las que Microsoft ha estado trabajando durante los últimos años con el objetivo de obtener una plataforma sencilla y potente para distribuir el software en forma de servicios que puedan ser suministrados remotamente y que puedan comunicarse y combinarse unos con otros de manera totalmente independiente de la plataforma, lenguaje de programación y modelo de componentes con los que hayan sido desarrollados.



**Características de .NET**

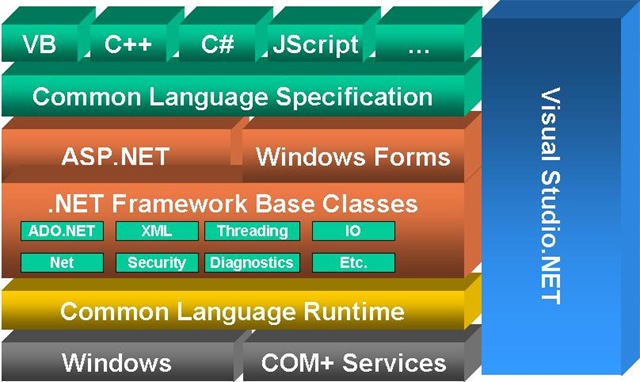
* + .NET es una nueva plataforma para el desarrollo y explotación de aplicaciones “gestionadas” (managed) modernas y orientadas a objetos
  + Las aplicaciones .NET se pueden desarrollar en cualquier lenguaje de programación que se ajusta a .NET
  + .NET soporta una extensa framework de librerías de clases independientes del lenguaje de programación
  + .NET soporta la creación de componentes auto-describibles
  + .NET ofrece integración multi-lenguaje, reutilización de componentes, y herencia entre componentes desarrollados en diferentes lenguajes
  + .NET ofrece un nueva manera de desarrollar aplicaciones de sobremesa usando las clases Windows Forms
  + .NET ofrece una nueva manera de desarrollar aplicaciones basadas en navegador Web a través de ASP.NET
  + Las clases ADO.NET proveen una arquitectura desconectada para acceso a datos a través de Internet
  + .NET soporta la creación de Servicios Web XML independientes de la plataforma, a través de SOAP (Simple Object Access Protocol) y WSDL (Web Services Description Language)
  + .NET ofrece una nueva arquitectura para el desarrollo y explotación de objetos remotos
  + .NET convierte a varias tecnologías y técnicas Windows en obsoletas

**Componentes de .NET**

* + Plataforma .NET
  + .NET Framework SDK
  + Visual Studio .NET
  + Servicios Web (Microsoft .NET myServices)
  + Servidores para empresas(SQL Server.NET...)

Plataforma .NET

* El concepto en el que se basa .NET no es nuevo.
* Java y su entorno de ejecución (JVM) ya utilizan el concepto de encapsulamiento del sistema operativo para permitir la interoperabilidad entre diferentes sistemas operativos.



**Componentes:**

* + Common Language Runtime(CLR): Entorno de ejecución de la plataforma.
  + .NET Framework Base Classes o FCL (Framework Class Library): Añaden funcionalidad.
  + ASP.NET: Versión .Net de ASP. Incluye los servicios Web.
  + Windows Forms

Los servicios del sistema operativo se suministran mediante librerías (DLL) dispuestas por el CLR cuando se instala

.NET Framework

en una plataforma(

Windows

,

McIntosh

,

Linux

, etc.). Dichas librerías son referenciadas en el códigode C# mediante la sentencia

using

seguida del nombre de la librería deseada.Tales declaraciones reciben el nombre de Espacios Calificados o Espacios conNombre (

Namespaces

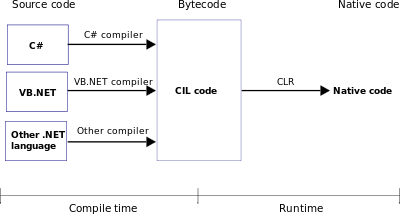
).

**Common Language Runtime**

* Un Runtime no es más que un entorno en el que se ejecutan los programas.
* De esta forma, el CLR es el entorno donde se ejecutarán las aplicaciones .NET que han sido compiladas a un lenguaje común llamado Microsoft Intermediate Language(MSIL).

**Common Language Runtime**

* + El proceso de compilación produce un fichero ejecutable en Windows al que se denomina portable executable (PE)
  + CLR sólo ve IL, por lo que se puede reemplazar el compilador JIT para usar un nuevo lenguaje



**Common Language Runtime**

* Caracteríticas y servicios que ofrece:
  + Ejecución multiplataforma
  + Integración de lenguajes
  + Gestión de memoria(Recolector de basura)
  + Tratamiento de excepciones
  + Soporte multi-hilo
  + Distribución transparente
  + Interoperabilidad con código antiguo

Clases de la plataforma .NET

Espacios de nombres más usados:

* + **System:** Contiene tipos de datos muy usados, datos básicos, tablas, excepciones... Es el raíz.
  + **System.Object:** Soporta toda la jerarquía de las clases de .NET.
  + **System.Collections:** Colecciones de datos de uso común como pilas, colas, listas...
  + **System.Data:** Manipulación de base de datos. Forma la denominada arquitectura ADO.NET.

Visual Studio.NET

* Es un entorno gráfico que permite a los desarrolladores crear, probar y depurar aplicaciones desarrolladas o no para la plataforma .NET.

SharpDevelop es un IDE open source para .NET que representa una buena alternativa a Visual Studio.NET

* Microsoft incorpora en este entorno los siguientes lenguajes de programación:
  + Visual Basic.NET: Versión .NET de Visual Basic 6.0.
  + Visual C++.net: Evolución de Visual C++.
  + Visual J#.NET: Adaptación de Visual J++.
  + Visual C#.NET: Un nuevo lenguaje de programación OO.
* Empresas ajenas han creado compiladores de sus lenguajes para la plataforma .NET:
  + Eiffel, Perl, Python, Haskell, Pascal, Oberon...

Análisis del código

Vamos a analizar el código fuente para irnos familiarizando con los mecanismostípicos de un programa en C#: La primera declaración indica que vamos a utilizar elespacio calificado (o

NameSpace

) llamado

System.

Un espacio calificado onominado es una especie de firma digital con la que podemos hacer referencia auna librería o bien -mediante la declaración

NameSpace

- hacer que sirva deidentificador de todos los módulos que pertenezcan a una aplicación, o unfragmento de aplicación.

System

es el NameSpace básico para todas lasaplicaciones que requieran algún servicio del sistema (que son prácticamentetodas). Note el lector que la declaración del espacio calificado podría haberseeliminado si en el código hubiéramos hecho referencia a la jerarquía completa, estoes, si hubiéramos escrito:

System.Console.WriteLine(“Hola Mundo”);

Más sobre el concepto de Namespace

Esto es así, porque elCLRpone a disposición de cualquier aplicación todos losservicios de sistema operativo a través de un conjunto jerárquico de clases con sólohacer referencia a ellas. No obstante, y para facilitar la escritura del códigoutilizamos el concepto de

Namespace

(

espacio calificado ó espacio con nombre

).Un

namespace

es una forma adecuada de organizar clases y otros tipos de datosen una jerarquía. Podríamos decir que sirven al mismo tiempo para 3 propósitos:organizar (dentro de la jerarquía), para firmar (o marcar de forma única)fragmentos de código como pertenecientes a ese

namespace

, evitando colisionescon otros fragmentos que puedan contener definiciones iguales, y como método deabreviación, ya que una vez declarado el

namespace

podemos referirnos a suscontenidos en el código sin necesidad de repetir toda la secuencia jerárquica, tal ycomo aparece en la línea de código anterior.

Los espacios calificados pueden estar predefinidos, como en este caso, o serdeclarados en el código por el propio usuario. Como nuestra aplicación utiliza unaúnica salida de texto por el dispositivo por defecto (Consola), no es preciso hacerninguna referencia a Windows ni a sus servicios. La consola está representada endicho espacio por el objeto

Console

que dispone de métodos para escribir texto enla consola por defecto

(Write()

y

Writeline())

y también para recoger entradasde datos del mismo dispositivo (

Read()

y

Readline()

) (Piense

Hola Mundo en .NET

// fichero: hola.cs

// compilar: csc hola.cs

using System;

class Hola {

public static void Main() {

Console.WriteLine(“Hola desde C#”);

}

}

* Todo programa en C# contiene al menos una clase
* Main es el punto de entrada del programa
* Console.WriteLine(“Hola desde C#”), visualiza un mensaje, invocando método WriteLine de la clase Console
* El espacio de nombres System pertenece a .NET FCL (Framework Class Library)
* FCL contiene muchas clases útiles, que pueden usarse desde nuestras aplicaciones .NET
* Para compilar el programa necesitamos una copia de .NET SDK, y usar el compilador de C#, csc

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **C# Tipo** | **.Net Framework (System) type** | **Signed?** | **Bytes en Ram** | **Rango** |
| sbyte | System.Sbyte | Yes | 1 | -128 a 127 |
| short | System.Int16 | Yes | 2 | -32768 a 32767 |
| int | System.Int32 | Yes | 4 | -2147483648 a 2147483647 |
| long | System.Int64 | Yes | 8 | -9223372036854775808 a 9223372036854775807 |
| byte | System.Byte | No | 1 | 0 a 255 |
| ushort | System.Uint16 | No | 2 | 0 a 65535 |
| uint | System.UInt32 | No | 4 | 0 a 4294967295 |
| ulong | System.Uint64 | No | 8 | 0 a 18446744073709551615 |
| float | System.Single | Yes | 4 | Aprox. ±1.5 x 10-45 a ±3.4 x 1038 con 7 decimales |
| double | System.Double | Yes | 8 | Aprox. ±5.0 x 10-324 a ±1.7 x 10308 con 15 o 16 decimales |
| decimal | System.Decimal | Yes | 12 | Aprox. ±1.0 x 10-28 a ±7.9 x 1028 con 28 o 29 decimales |
| char | System.Char | N/A | 2 | Cualquier caracter Unicode |
| bool | System.Boolean | N/A | 1 / 2 | true o false |