



Desarrollo de aplicaciones distribuidas con .NET:
Introducción a .NET



Grupo ARCOS

Desarrollo de Aplicaciones Distribuidas

Ingeniería Informática

Universidad Carlos III de Madrid

Contenidos

1. **Introducción a .NET**
2. **Anatomía de .NET:**
 1. Bibliotecas de clases base
 2. CLR
 3. .NET PE + Assembly
3. **Ejemplo de aplicación**
 - ▶ Hola desde...

Contenidos

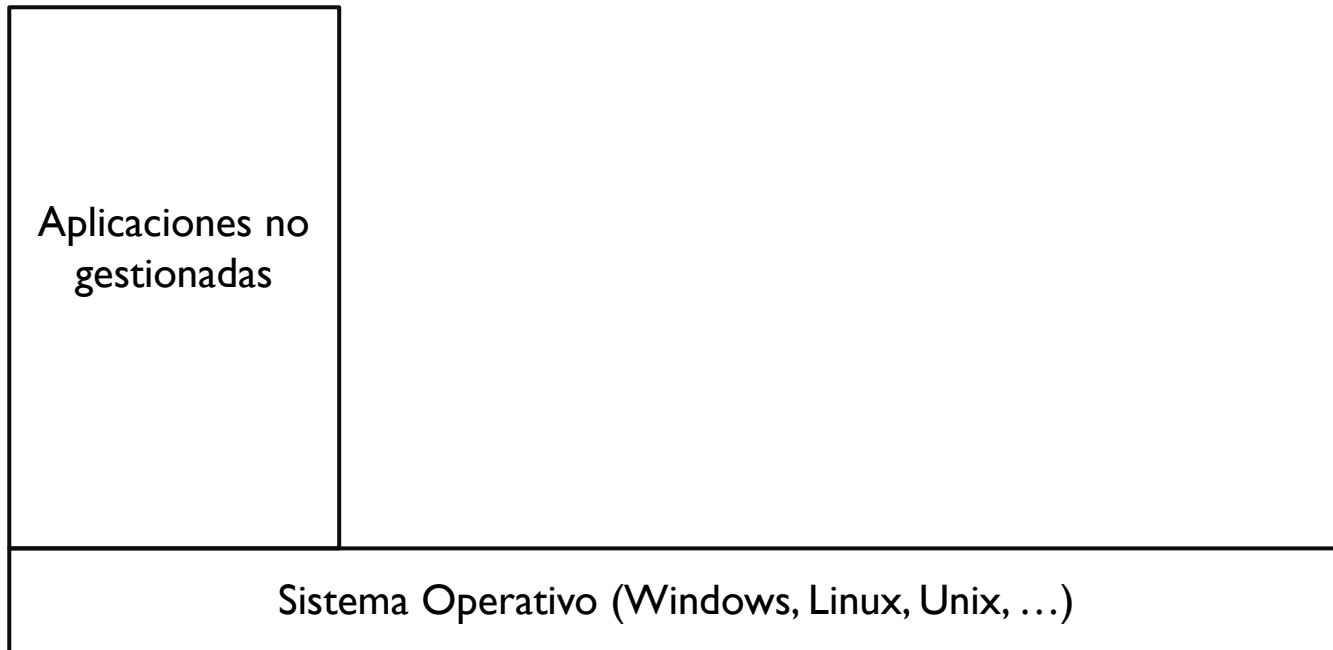
1. **Introducción a .NET**
2. Anatomía de .NET:
 1. Bibliotecas de clases base
 2. CLR
 3. .NET PE + Assembly
3. Ejemplo de aplicación
 - ▶ Hola desde...

¿Qué es .NET?

.NET:
• Qué es
• Objetivos
• Entornos

4

- ▶ Una nueva plataforma software para escritorio y para Web

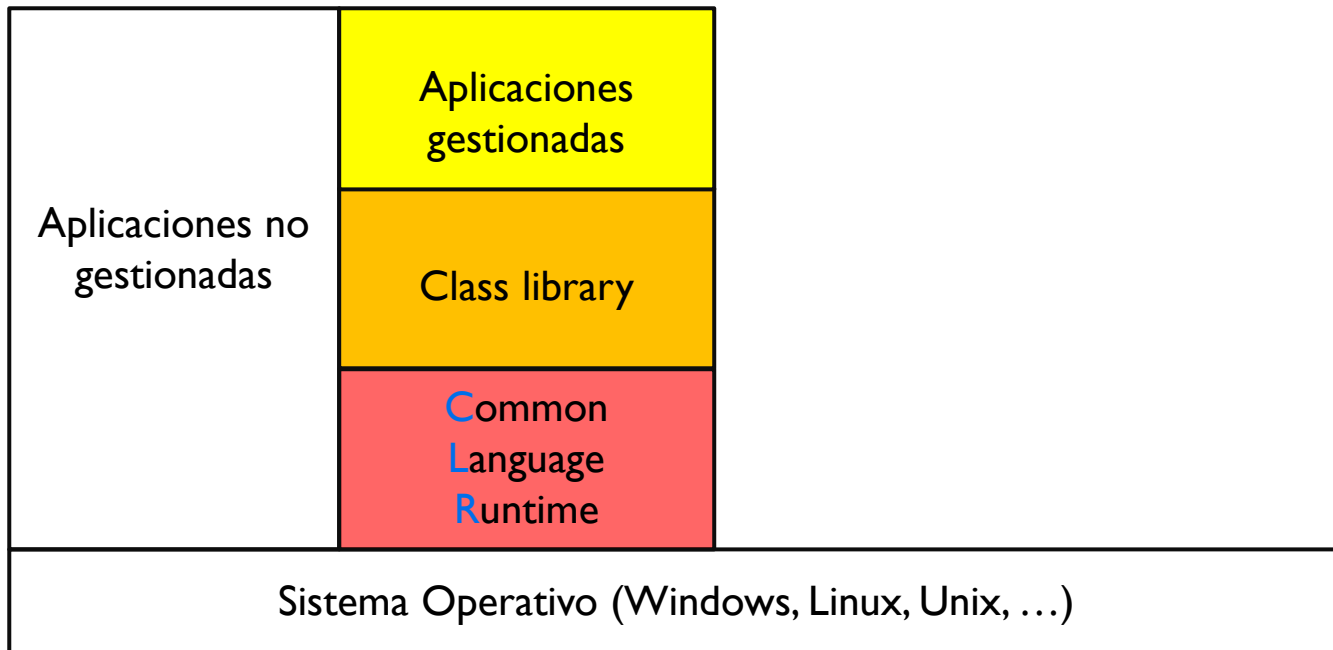


¿Qué es .NET?

.NET:
• Qué es
• Objetivos
• Entornos

5

- ▶ Una nueva plataforma software para escritorio y para Web

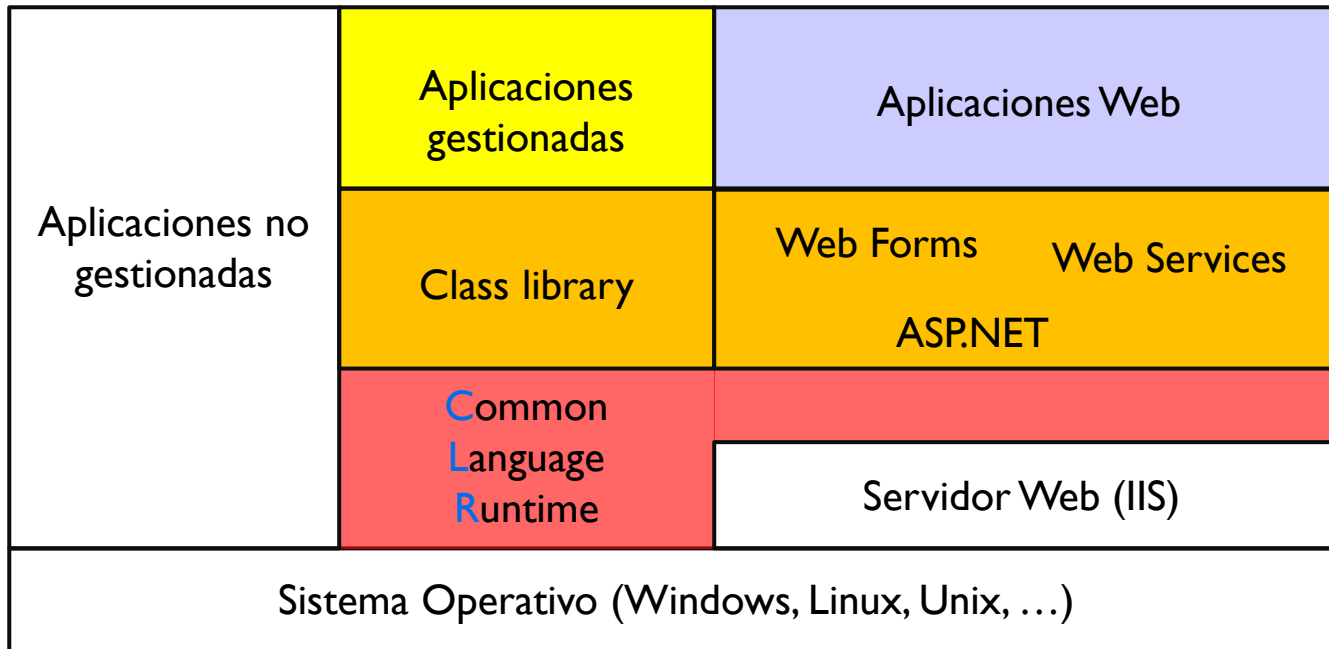


- ▶ **Common Language Runtime:**
interoperabilidad, seguridad, recolección de basura, versionado, etc.
- ▶ **Class library:**
GUI, colecciones, hilos, red, reflexión, XML, etc.

¿Qué es .NET?

.NET:
• Qué es
• Objetivos
• Entornos
6

- ▶ Una nueva plataforma software para escritorio y para Web

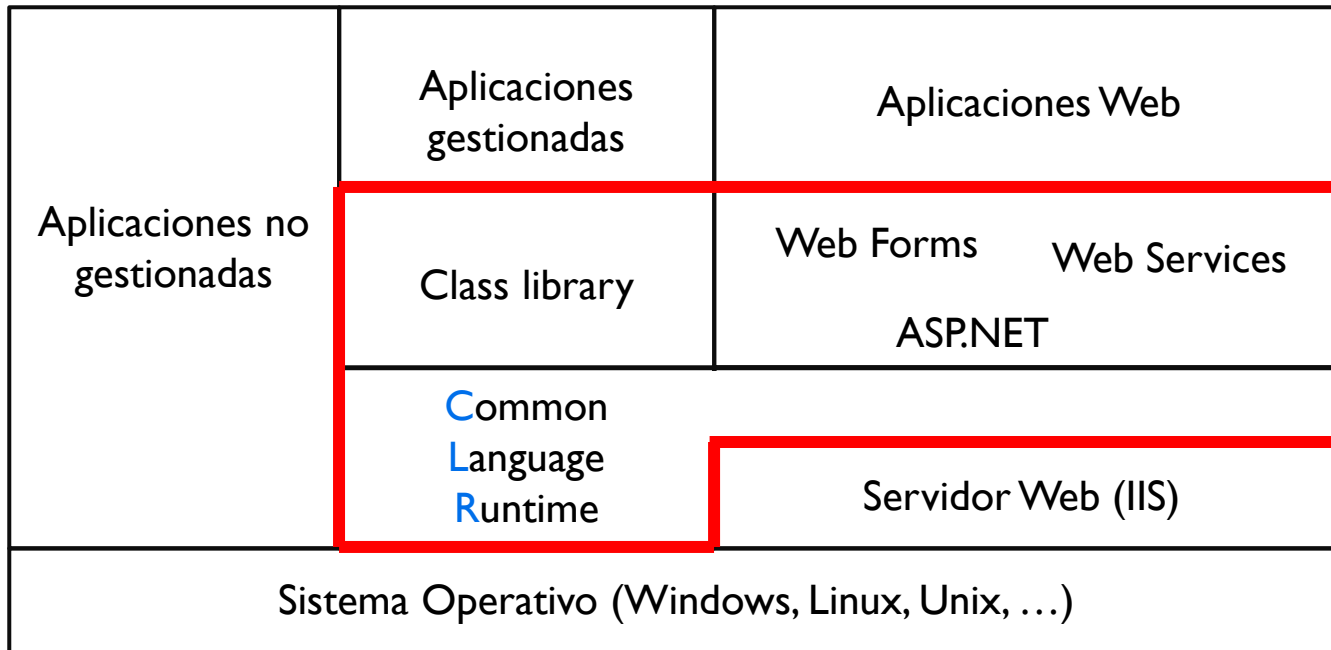


- ▶ **ASP.NET, Web Forms:**
Interfaz gráfica Web (orientada a objetos, basada en eventos e independiente de navegador)
- ▶ **Web Services:**
Servicios distribuidos sobre RPC (SOAP y HTTP).

¿Qué es .NET?

.NET:
• Qué es
• Objetivos
• Entornos
7

- ▶ Una nueva plataforma software para escritorio y para Web



- ▶ **.NET framework:**
 - ▶ **Utilidades:** Visual Studio .NET, ildasm, gacutil, etc.
 - ▶ **Servidores:** SQL server, BizTalk server, Exchange server, etc.
 - ▶ **Servicios:** My Services, Passport service, etc.

Principales objetivos

- ▶ Unificar la programación en distintos entornos:
 - ▶ Escritorio, programación Web y entornos móviles

Antes

Escritorio
<ul style="list-style-type: none">• orientado a objetos• compilado (C, C++, ...)• biblioteca de clases

Web
<ul style="list-style-type: none">• ASP (no OO)• interpretado (PHP,...)• biblioteca específicas

Principales objetivos

- ▶ Unificar la programación en distintos entornos:
 - ▶ Escritorio, programación Web y entornos móviles

Antes

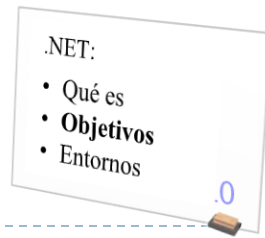
Escritorio
<ul style="list-style-type: none">• orientado a objetos• compilado (C, C++, ...)• biblioteca de clases

Web
<ul style="list-style-type: none">• ASP (no OO)• interpretado (PHP,...)• biblioteca específicas

Con .NET

Escritorio y Web
<ul style="list-style-type: none">• orientado a objetos (ASP.NET)• compilado (C#, C++, VB.NET, Fortran,...)• biblioteca de clases uniforme

Principales objetivos

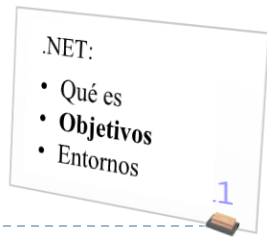


► Apoyo para multitud de lenguajes de programación:

Lenguajes de .NET ofrecidos por Microsoft:

Lenguaje	Link
C#	http://msdn.microsoft.com/vstudio/nextgen/technology/csharpintro.asp
JScript	http://msdn.microsoft.com/workshop/languages/clinic/scripting07142000.asp
Managed C++	http://msdn.microsoft.com/vstudio/nextgen/Technology/managedext.asp
VB.NET	http://msdn.microsoft.com/vstudio/nextgen/technology/language.asp

Principales objetivos

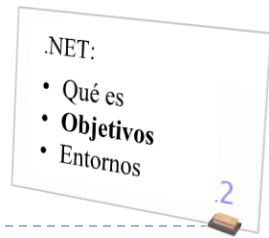


► Apoyo para multitud de lenguajes de programación:

Lenguajes de .NET ofrecidos por otros proveedores:

Lenguaje	Link
APL	http://www.dyadic.com
COBOL	http://www.adtools.com/info/whitepaper/net.html
Component Pascal	http://www2.fit.qut.edu.au/CompSci/PLAS//ComponentPascal
Delta Forth	http://www.dataman.ro/dforth/
Eiffel#	http://www.eiffel.com/doc/manuals/technology/dotnet/eiffelsharp/white_paper.html
Fortran	http://www.lahey.com/dotnet.htm , http://www.salfordsoftware.co.uk/compilers/ftn95/dotnet.shtml
Haskell	http://haskell.cs.yale.edu/ghc
Mercury	http://www.cs.mu.oz.au/research/mercury/dotnet.html
Mondrian	http://www.mondrian-script.org
Oberon	http://www.oberon.ethz.ch/lightning
Perl	http://www.activestate.com/ASP/NET
Python	http://www.activestate.com/ASP/NET
RPG	http://www.asna.com/pr2%5F20%5F01.asp
Scheme	http://rover.cs.nwu.edu/~scheme
Smalltalk	http://www.qks.com
Standard ML	http://www.research.microsoft.com/Projects/SML.NET
TMT Pascal	http://www.tmt.com/net.htm

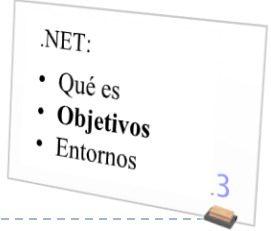
Principales objetivos



- ▶ Interoperabilidad entre lenguajes de programación:
 - ▶ Posible escribir aplicaciones en diferentes lenguajes

Antes

- ▶ Millones de líneas de código en C++, Fortran, Visual Basic, ...
- ▶ Interoperabilidad muy limitada



Principales objetivos

- ▶ Interoperabilidad entre lenguajes de programación:
 - ▶ Posible escribir aplicaciones en diferentes lenguajes

Antes

- ▶ Millones de líneas de código en C++, Fortran, Visual Basic, ...
- ▶ Interoperabilidad muy limitada

Con .NET

- ▶ Compatibilidad binaria entre más de 20 lenguajes (C#, ML, ...)

Clase en VB.NET

```
Public Class A
  Public x As Integer
  Public Sub Foo() ...
End Class
```

Subclase en C#

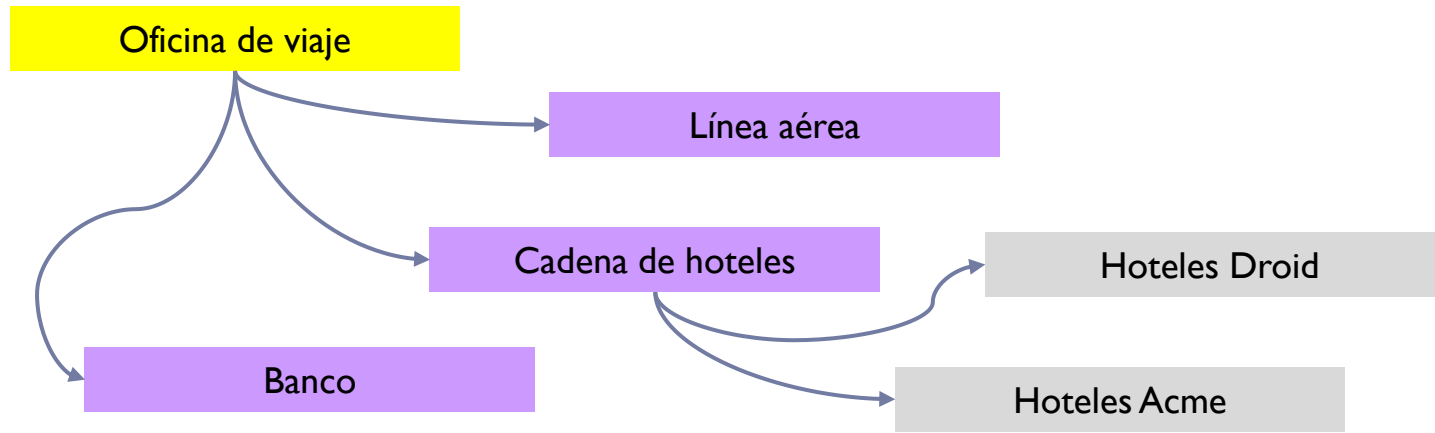
```
class B :A {
  public string s;
  public void Bar() {...}
}
```

Usado en Eiffel

```
class Client feature
  obj: B;
  ...
  create obj;
  obj.Bar;
  ...
end
```

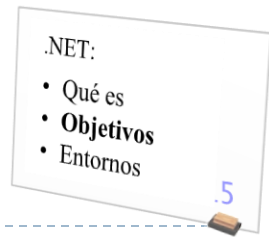
Principales objetivos

- ▶ **Apoyo para desarrollar aplicaciones Web:**
 - ▶ Aplicaciones B2B sin navegador Web



- ▶ **Basadas en estándares:**
 - ▶ HTTP
 - ▶ SOAP
 - ▶ XML

Principales objetivos

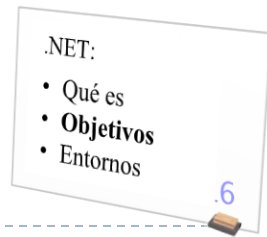


- ▶ **Apoyo para desarrollar aplicaciones Web:**
 - ▶ Páginas Web dinámicas más simples



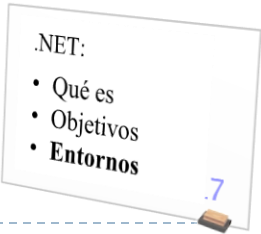
- ▶ **ASP.NET**
 - ▶ Separación clara de HTML y código
 - ▶ Orientado a objetos
 - ▶ Basado en Eventos
 - ▶ Gestión de estado
 - ▶ Interfaz con elementos personalizables
 - ▶ Eficiente
 - ▶ Etc.

Principales objetivos



- ▶ **Facilidades para el despliegue:**
 - ▶ Seguridad
 - ▶ Firma con clave pública
 - ▶ Derecho de acceso basado en códigos
 - ▶ Ejecución *side-by-side*
 - ▶ Fin del infierno de las DLL
 - ▶ Versionado (pueden coexistir diferentes bibliotecas)
 - ▶ Instalación con impacto nulo
 - ▶ No entradas en el registro, desinstalación limpia, etc.
 - ▶ Diferentes clientes
 - ▶ Ordenadores, móviles, sistemas embebidos, etc.

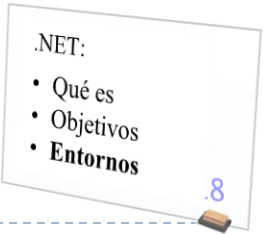
Entornos de desarrollo



- ▶ V 0.x
 - ▶ Presentación de la nueva plataforma



Entornos de desarrollo



- ▶ V 1.0
 - ▶ Además de las ideas originales muchas sugerencias aportadas por los desarrolladores y usuarios han sido añadidas



Entornos de desarrollo

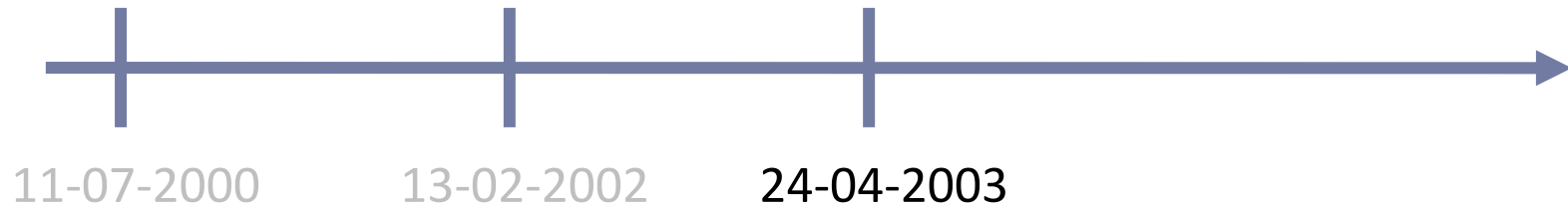
.NET:
• Qué es
• Objetivos
• Entornos
.9

- ▶ V 1.1
 - ▶ Se ofrece como paquete redistribuible (~jre) o como kit de desarrollo de aplicaciones (~jdk)
 - ▶ Fuerte actualización, ofrecida de serie en WS2003
 - ▶ IPv6, soporte nativo para ODBC, oracle, mobile ASP.NET, etc.

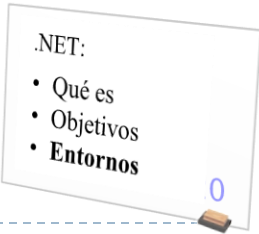
Microsoft®
.net[™]
Pre-beta (alfa)

Microsoft®
.net[™]
Versión 1.0

Microsoft®
.net[™]
Versión 1.1



Entornos de desarrollo



- ▶ V 2.0
 - ▶ Numerosos cambios en el API
 - ▶ Soporte para 64 bits (x64 + IA64)
 - ▶ Soporte para programación genérica en el CLR
 - ▶ Etc.

Microsoft®
.net[™]
Pre-beta (alfa)

Microsoft®
.net[™]
Versión 1.0

Microsoft®
.net[™]
Versión 1.1

Microsoft®
.net[™]
Versión 2.0



Entornos de desarrollo

.NET:
• Qué es
• Objetivos
• Entornos

1

- ▶ V 3.0
 - ▶ WPF (*Windows Presentation Foundation*)
 - ▶ WCF (*Windows Communication Foundation*)
 - ▶ WF (*Windows Workflow Foundation*)
 - ▶ InfoCard (*Windows CarSpace*)



Versión 3.0



06-11-2006

Entornos de desarrollo

.NET:
• Qué es
• Objetivos
• Entornos

2

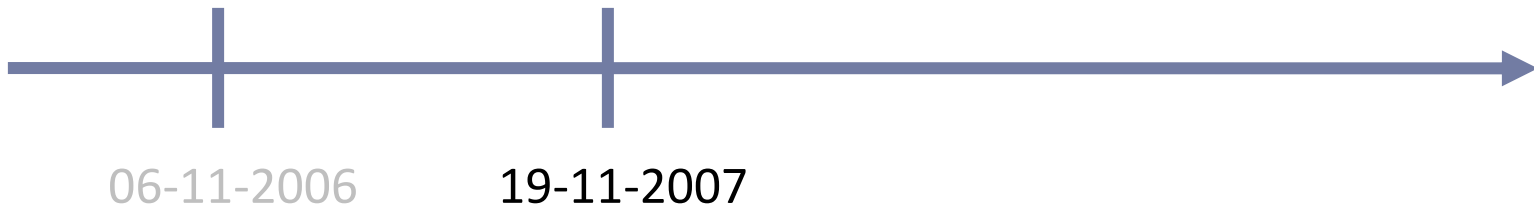
- ▶ V 3.5
 - ▶ LINQ (*Language Integrated Query*)
 - ▶ ASP.NET Ajax
 - ▶ Etc.



Versión 3.0



Versión 3.5



Entornos de desarrollo

.NET:
• Qué es
• Objetivos
• Entornos

3

- ▶ V 4.0 (beta 2)
 - ▶ Extensiones para programación paralela
 - ▶ Memoria transaccional por software
 - ▶ Diversas mejoras en los lenguajes:
 - ▶ lambda, parámetros opcionales, ...
 - ▶ Etc.



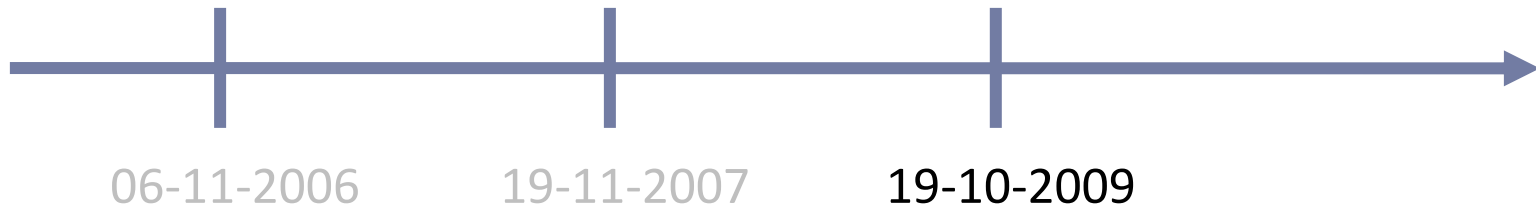
Versión 3.0



Versión 3.5

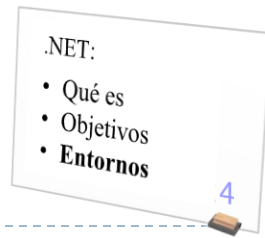


Versión 4.0b2





Entornos de desarrollo:



- ▶ Definido como:
“un sistema de desarrollo .NET multiplataforma de código abierto”
- ▶ Principales características:
 - ▶ **Multi-plataforma:**
 - ▶ Ejecuta en Linux, OS X, BSD y Microsoft Windows, incluyendo x86, x86-64, ARM, s390, PowerPC y muchos más
 - ▶ **Multi-lenguaje:**
 - ▶ Desarrollo en C# 3.0 (incluyendo LINQ), VB 8, Java, Python, Ruby, Eiffel, F#, Oxygene y más
 - ▶ **Compatible binario:**
 - ▶ Construido según la especificación del estándar ECMA de CLI y C#
 - ▶ **Compatible con el API de Microsoft .NET:**
 - ▶ Ejecuta aplicaciones ASP.NET, ADO.NET y Windows.Forms 2.0 sin recompilación
 - ▶ **Código abierto:**
 - ▶ El entorno de desarrollo, compilador y bibliotecas se distribuyen bajo licencias aprobadas por OSI y están disponibles para licencias duales

Entornos de desarrollo

.NET:
• Qué es
• Objetivos
• Entornos

5

- ▶ V 1.0
 - ▶ Herramientas de desarrollo (compilador C#, CLI, etc.)
 - ▶ Bibliotecas que implementan el perfil .NET 1.1
 - ▶ Bibliotecas de terceros (Gtk#, Mono, etc.)
 - ▶ Etc.



Versión 1.0



30-06-2004

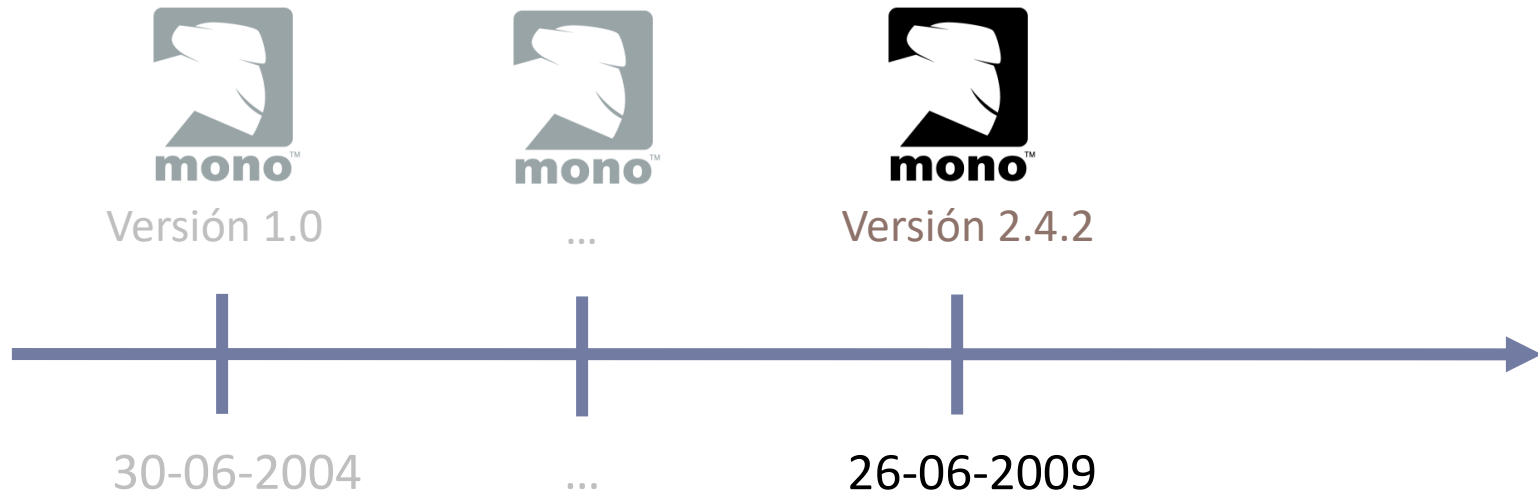
Entornos de desarrollo

.NET:

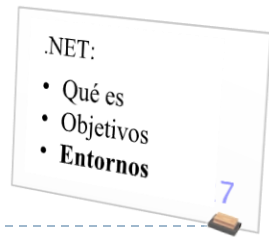
- Qué es
- Objetivos
- **Entornos**

6

- ▶ V 2.4.2
 - ▶ Optimizaciones: uso de instrucciones SIMD, hilos sin espera activa, ...
 - ▶ C# 3.0, LINQ, ADO.NET 2.0, ASP.NET 2.0, Windows.Forms 2.0, ...
 - ▶ Corrección de errores



Entornos preinstalados



- ▶ **.NET Framework V 2.0**
 - ▶ Incluido de serie en Visual Studio 2005 y Microsoft SQL Server 2005
 - ▶ El SDK (~JDK) y el paquete redistribuible (~JRE) pueden obtenerse desde la Web de Microsoft
- ▶ **.NET Framework V 3.0**
 - ▶ Incluido de serie en Windows Vista y Windows Server 2008
 - ▶ C# y VB.NET, usando el CLR de .NET Framework 2.0
- ▶ **.NET Framework V 3.5**
 - ▶ Incluido de serie en Windows 7
 - ▶ C# y VB.NET

Entornos a instalar

- .NET:
- Qué es
- Objetivos
- Entornos

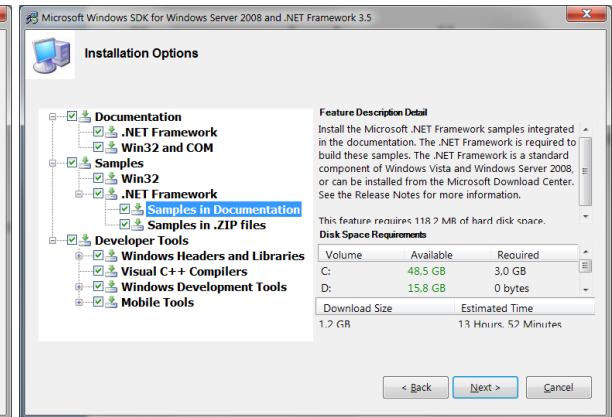
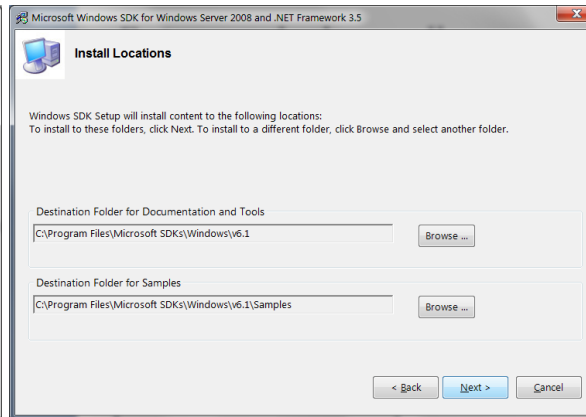
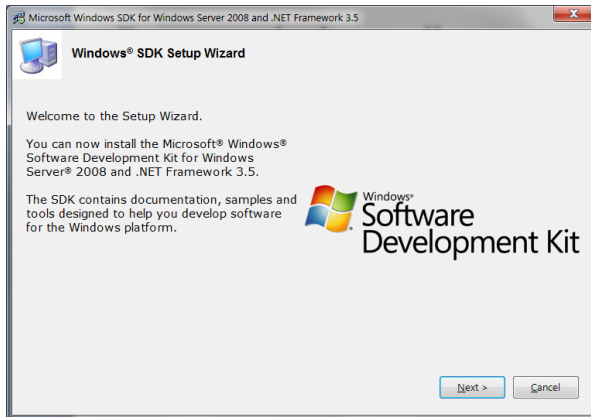
8

▶ .NET Framework V 3.5

- ▶ Descarga desde: <http://www.microsoft.com/downloads/details.aspx?FamilyID=333325FD-AE52-4E35-B531-508D977D32A6>
- ▶ Instala el *Framework* en entornos no disponible por defecto.

▶ Windows SDK para .NET Framework 3.5

- ▶ Descarga desde: <http://www.microsoft.com/downloads/details.aspx?FamilyID=e6e1c3df-a74f-4207-8586-711ebe331cdc>
- ▶ Añade documentación, ejemplos y herramientas para el desarrollo



Contenidos

1. **Introducción a .NET**
2. **Anatomía de .NET:**
 1. Bibliotecas de clases base
 2. CLR
 3. .NET PE + Assembly
3. **Ejemplo de aplicación**
 - ▶ Hola desde...

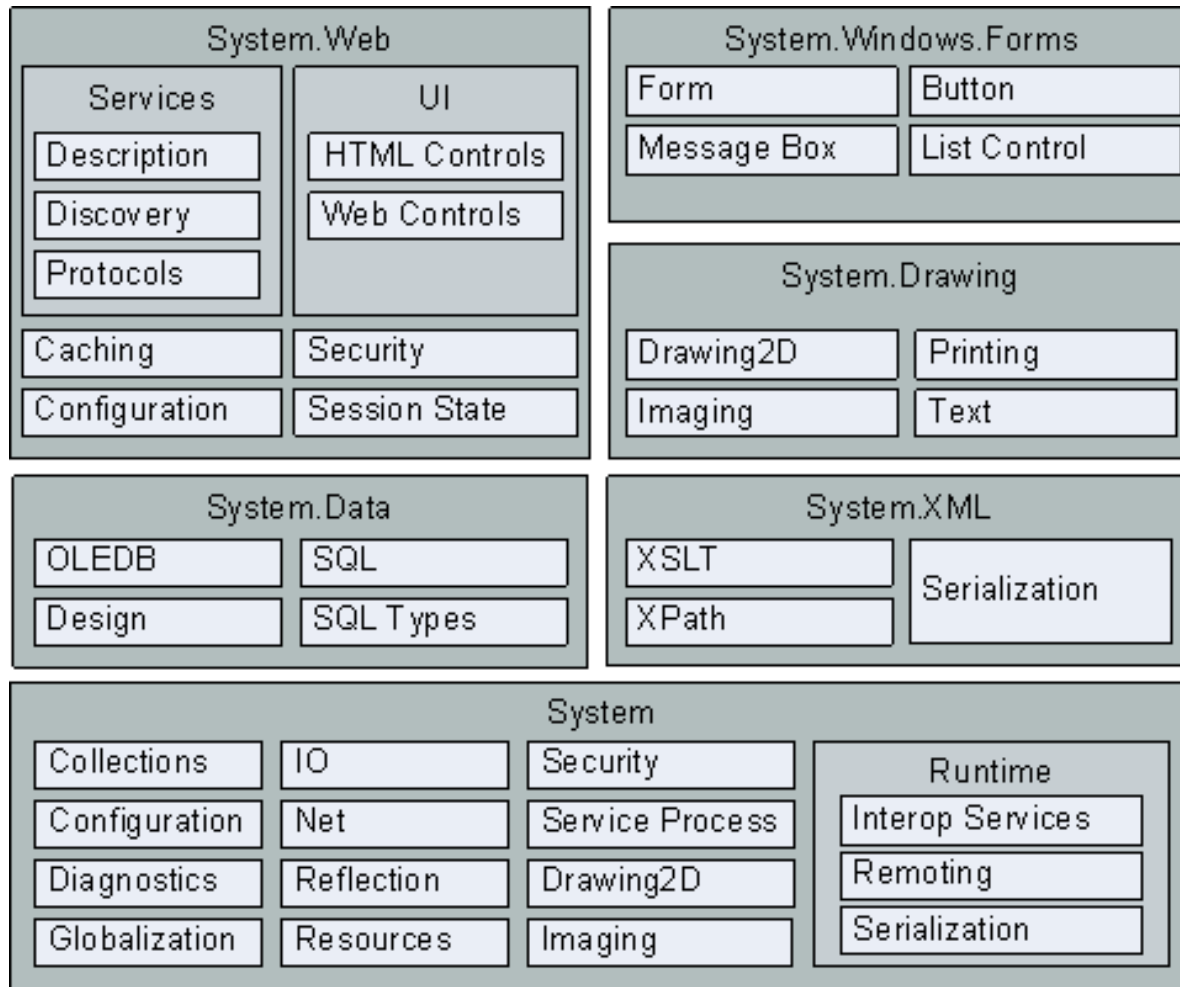
Esqueleto de .NET



Biblioteca de Clases

- .NET:
- Clases base
- CLR
- Assembly

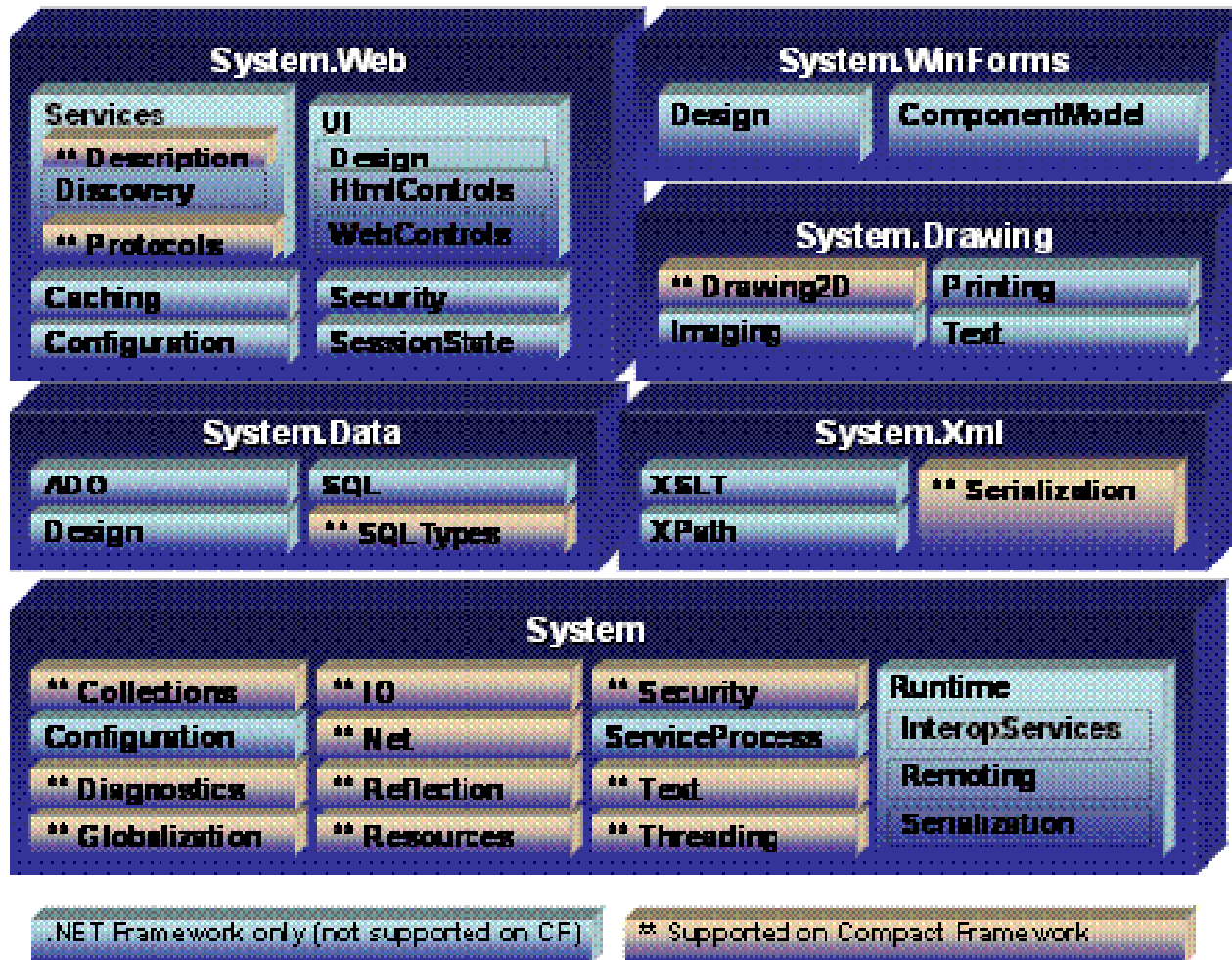
31



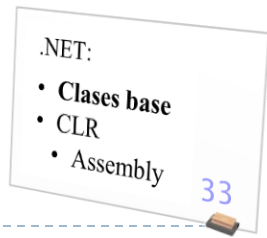
Biblioteca de Clases

- .NET:
- Clases base
 - CLR
 - Assembly

32



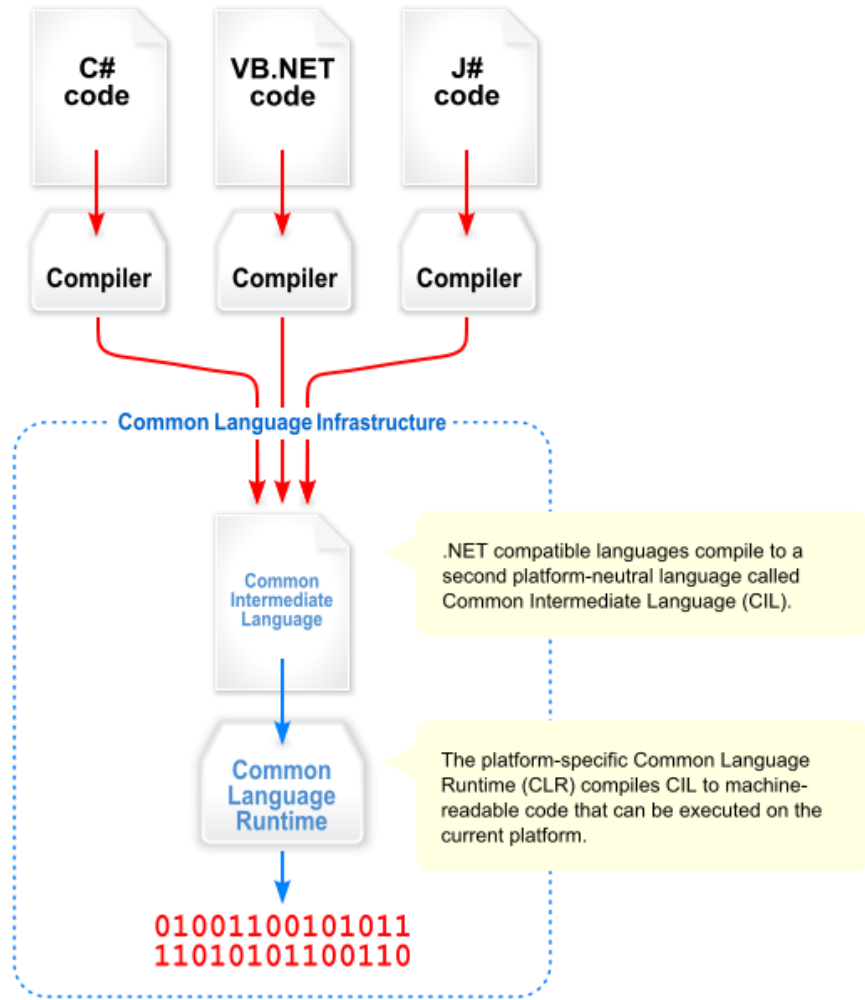
Biblioteca de Clases Base



- ▶ **System**
 - ▶ Se incluye los servicios básicos como String, DateTime, Boolean, así como excepciones, arrays, etc.
- ▶ **System.Collections**
 - ▶ Define mucho de los contenedores o colecciones, como listas, colas, pilas, etc.
- ▶ **System.Diagnostics**
 - ▶ Posibilita generar trazas, usar contadores, etc.
- ▶ **System.Globalization**
 - ▶ Ofrece apoyo para internacionalización.
- ▶ **System.IO**
 - ▶ Permite lectura y escritura de diferentes flujos, como ficheros.
- ▶ **System.Net**
 - ▶ Ofrecen una interfaz para distintos protocolos de red.
- ▶ **System.Reflection**
 - ▶ Ofrece la posibilidad de gestionar las capacidades de reflexión.
- ▶ **System.Runtime**
 - ▶ Permite la configuración del comportamiento del CLR.
- ▶ **System.Security**
 - ▶ Ofrece mecanismos de seguridad.
- ▶ **System.Text**
 - ▶ Apoyo para el uso de expresiones regulares, y formas eficientes de uso de Strings.
- ▶ **System.Threading**
 - ▶ Facilita la programación multihilo.

Uso del CLR

.NET:
• Clases base
• CLR
• Assembly

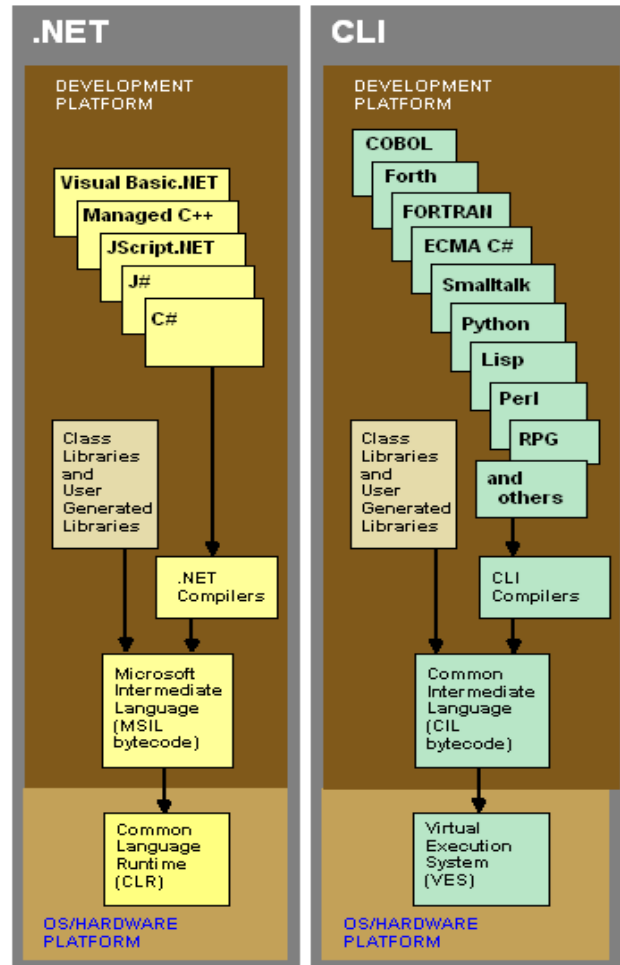


Uso del CLR

.NET:
• Clases base
• CLR
• Assembly

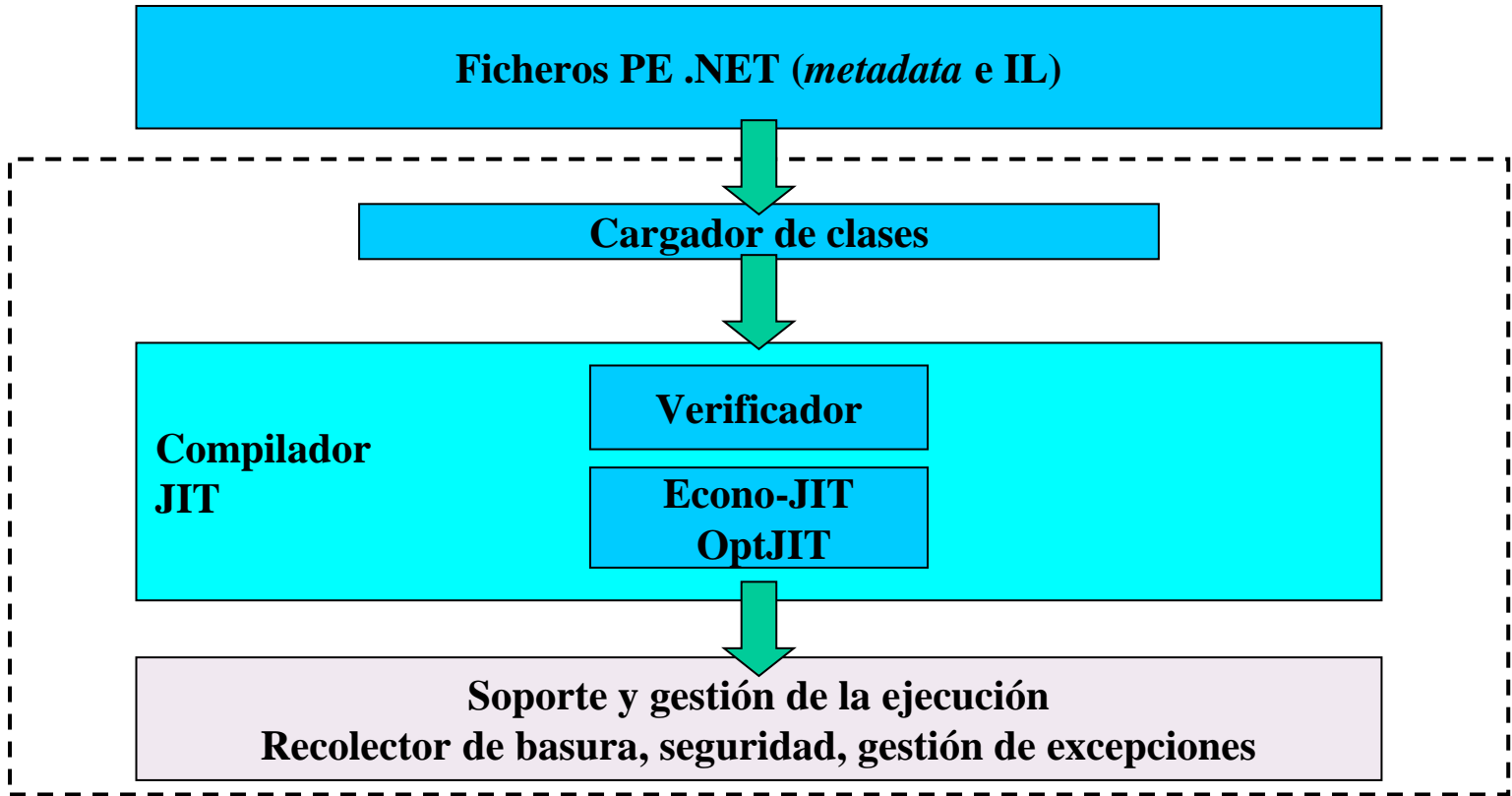
35

From Computer Desktop Encyclopedia
© 2006 The Computer Language Co., Inc.



Estructura del CLR

Herramienta virtual de ejecución CRL



Estructura del CLR

.NET:
• Clases base
• CLR
• Assembly

37



Diagrama interno de un **.NET PE**

.NET:
• Clases base
• CLR
• Assembly

38

- ▶ **.NET Portable Executable file:**
 - ▶ Fichero de formato PE
 - ▶ Derivado formato Microsoft Common Object File Format (COFF)

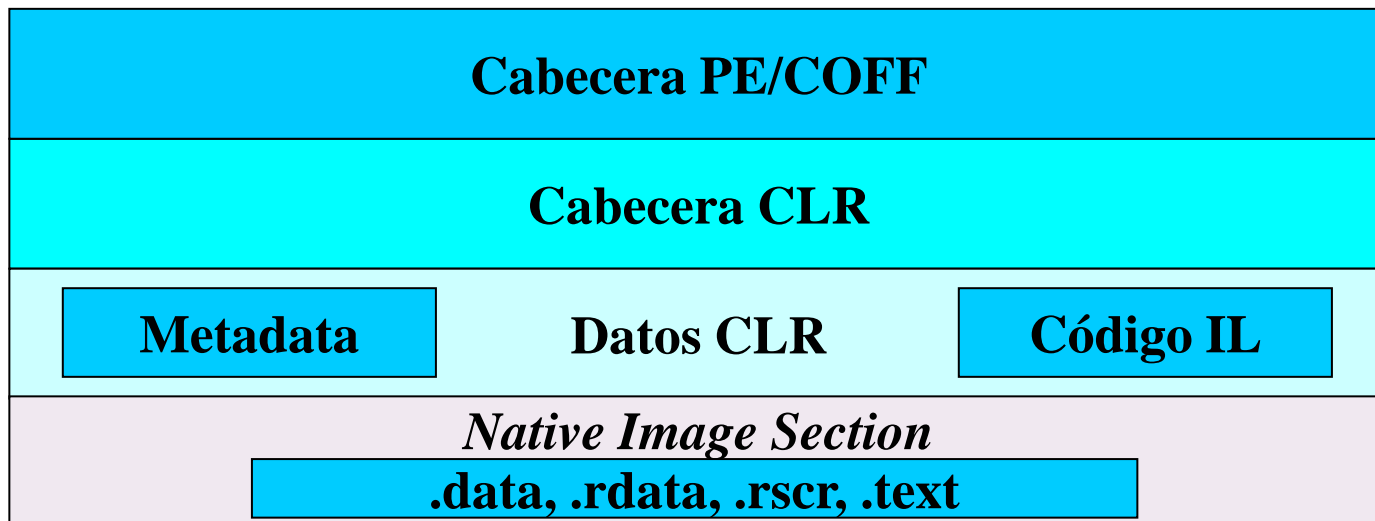
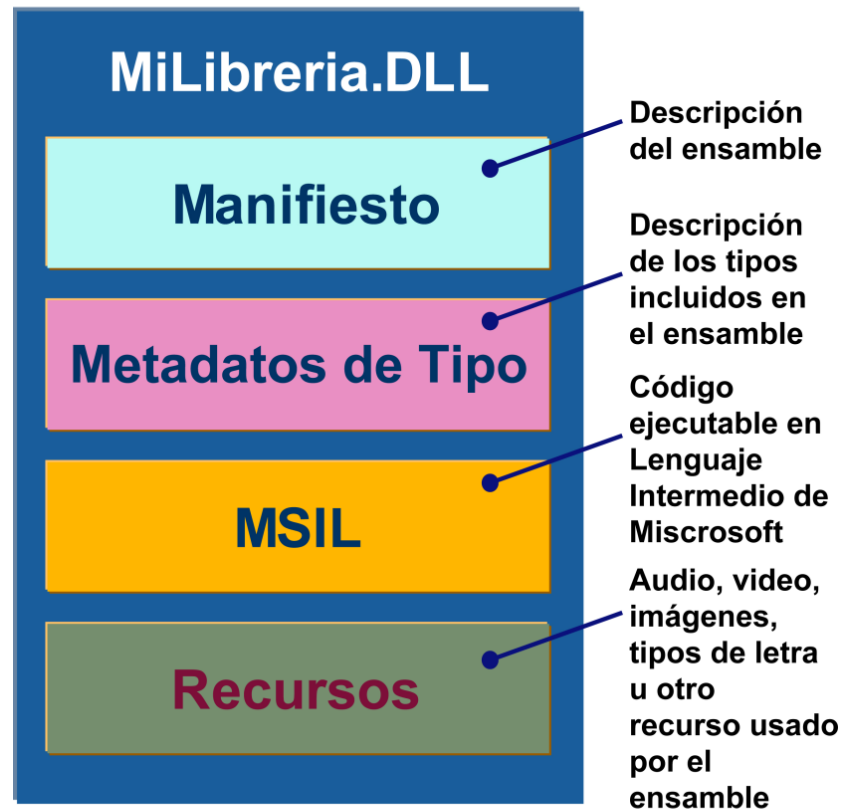


Diagrama interno de un Assembly .NET.

- .NET:
- Clases base
- CLR
- Assembly

39



Un ensamble es una unidad lógica, no física. Puede consistir de uno o más módulos (.dll, .jpg, etc)

Contenidos

1. **Introducción a .NET**
2. **Anatomía de .NET:**
 1. Bibliotecas de clases base
 2. CLR
 3. .NET PE + Assembly
3. **Ejemplo de aplicación**
 - ▶ **Hola desde...**

Entorno para el ejemplo

MS Windows XP/Vista/7 con .Net Framework 3.5

netenv.bat

```
@echo off
```

```
Echo
```

```
Echo Línea de mandatos para el .NET Framework 3.5
```

```
Echo.
```

```
@SET FrameworkDir=C:\WINDOWS\Microsoft.NET\Framework
```

```
@SET FrameworkVersion=v3.5
```

```
@set PATH=%FrameworkDir%\v3.5;%FrameworkDir%\%FrameworkVersion%;%PATH%
```

```
@set LIBPATH=%FrameworkDir%\v3.5;%FrameworkDir%\%FrameworkVersion%;%LIBPATH%
```

```
cmd
```

Ejemplo en C#

MS Windows XP/Vista/7 con .Net Framework 3.5

► Código del ejemplo:

hola.cs

```
using System;

class MainClass
{
    static void Main(string[] args)
    {
        Console.WriteLine ("Hola desde C#");
    }
}
```

Ejemplo en C#

MS Windows XP/Vista/7 con .Net Framework 3.5

► Compilación del ejemplo:

```
C:\Temp> csc hola.cs  
Microsoft (R) Visual C# 2008 Compiler version 3.5.30729.4926  
for Microsoft (R) .NET Framework version 3.5  
Copyright (C) Microsoft Corporation. All rights reserved.
```

► Ejecución del ejemplo:

```
C:\Temp> hola  
Hola desde C#
```

Ejemplo en Visual Basic .NET

MS Windows XP/Vista/7 con .Net Framework 3.5

► Código del ejemplo:

hola.vb

```
Imports System
```

```
Public Module modmain
```

```
    Sub Main( )
```

```
        Console.WriteLine ("Hola desde VB.NET")
```

```
    End Sub
```

```
End Module
```

Ejemplo en Visual Basic .NET

MS Windows XP/Vista/7 con .Net Framework 3.5

► Compilación del ejemplo:

```
C:\Temp> vbc /t:exe /out:hola.exe hola.vb
```

```
Microsoft (R) Visual Basic Compiler version 9.0.30729.715
```

```
Copyright (c) Microsoft Corporation. All rights reserved.
```

► Ejecución del ejemplo:

```
C:\Temp> hola
```

```
Hola desde VB.NET
```

Ejemplo en C++

MS Windows XP/Vista/7 con SDK para .Net Framework 3.5

▶ Código del ejemplo:

hola.cpp

```
#using <mscorlib.dll>
using namespace System;

void main( )
{
    Console::WriteLine("Hola desde C++");
}
```

Tutorial C++: <http://msdn.microsoft.com/en-us/beginner/cc305129.aspx>

Ejemplo en C++

MS Windows XP/Vista/7 con SDK para .Net Framework 3.5

► Compilación del ejemplo:

```
C:\Temp> cmd.exe /V:ON /E:ON /K "C:\Program Files\Microsoft SDKs\
→ Windows\v6.1\bin\setenv.cmd" /Release
C:\Temp> color
C:\Temp> cl hola.cpp /clr /link /entry:main
```

► Ejecución del ejemplo:

```
C:\Temp> hola
Hola desde C++
```

Contenidos

1. **Introducción a .NET**
2. **Anatomía de .NET:**
 1. Bibliotecas de clases base
 2. CLR
 3. .NET PE + Assembly
3. **Ejemplo de aplicación**
 - ▶ Hola desde...
 - ▶ **Generación de DLL e inspección...**

Ejemplo en C#

MS Windows XP/Vista/7 con .Net Framework 3.5

► Código del ejemplo:

hola.cs

```
using System;

class MainClass
{
    static void Main (string[] args)
    {
        Console.WriteLine ("Hola desde C#");
    }
}
```

Ejemplo en C#

MS Windows XP/Vista/7 con .Net Framework 3.5

► Compilación como ejecutable:

```
C:\Temp> csc /out:hola.exe hola.cs
```

```
Microsoft (R) Visual C# 2008 Compiler version 3.5.30729.4926  
for Microsoft (R) .NET Framework version 3.5  
Copyright (C) Microsoft Corporation. All rights reserved.
```

► Compilación como DLL:

```
C:\Temp> csc /out:hola.dll /t:library hola.cs
```

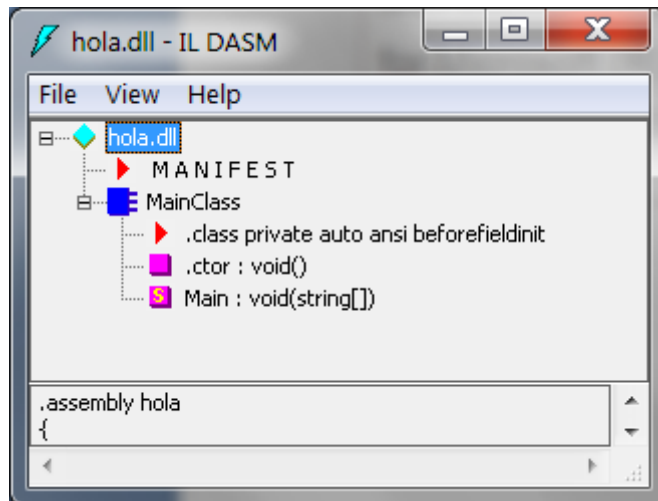
```
Microsoft (R) Visual C# 2008 Compiler version 3.5.30729.4926  
for Microsoft (R) .NET Framework version 3.5  
Copyright (C) Microsoft Corporation. All rights reserved.
```

Ejemplo en C#

MS Windows XP/Vista/7 con .Net Framework 3.5

► Inspección de *assembly*:

```
C:\Temp> ildasm hola.dll
```

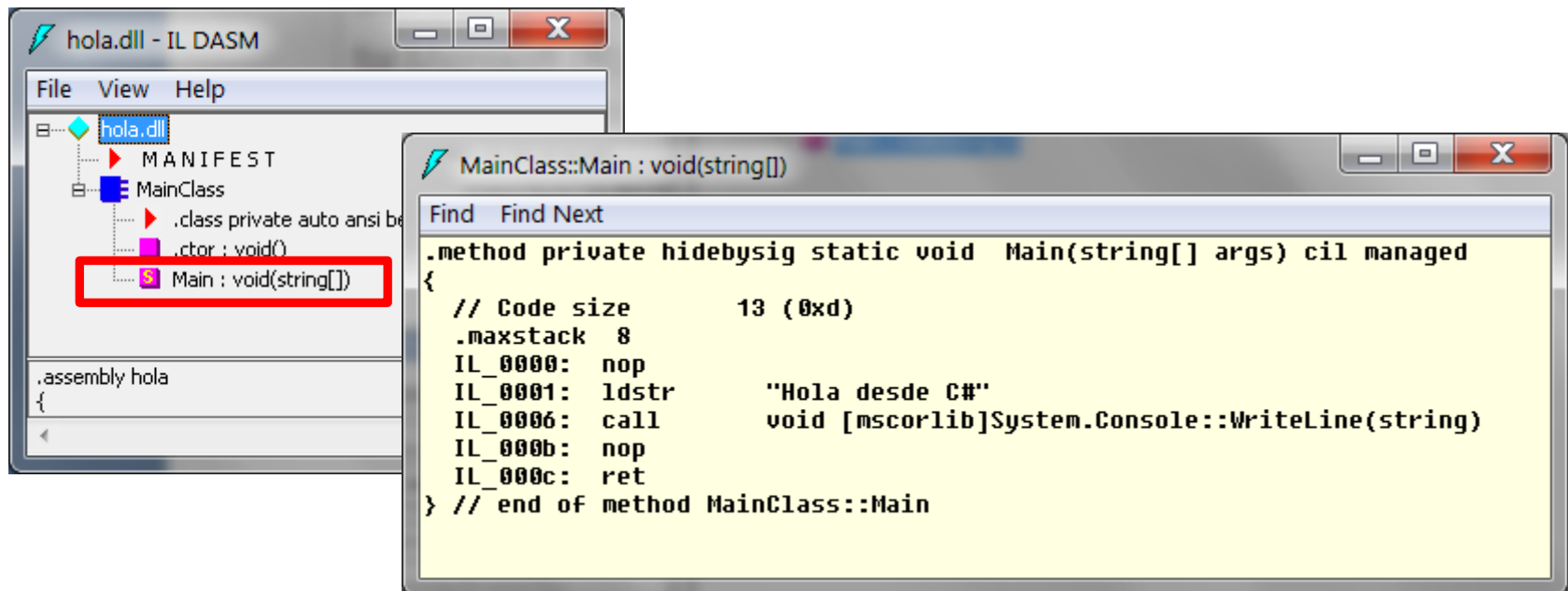


Ejemplo en C#

MS Windows XP/Vista/7 con .Net Framework 3.5

► Inspección de *assembly*:

```
C:\Temp> ildasm hola.dll
```



The screenshot shows the ILDASM tool interface. The left pane displays the assembly structure for hola.dll, with the Main method highlighted in red. The right pane shows the IL code for the Main method:

```
Find Find Next
.method private hidebysig static void Main(string[] args) cil managed
{
    // Code size          13 (0xd)
    .maxstack 8
    IL_0000: nop
    IL_0001: ldstr      "Hola desde C#"
    IL_0006: call        void [mscorlib]System.Console::WriteLine(string)
    IL_000b: nop
    IL_000c: ret
} // end of method MainClass::Main
```



Desarrollo de aplicaciones distribuidas con .NET:
Introducción a .NET



Grupo ARCOS

Desarrollo de Aplicaciones Distribuidas

Ingeniería Informática

Universidad Carlos III de Madrid