

.NET para todos: Bienvenido a MONO



Pablo Barrera González
barrera@gsyc.info
<http://gsyc.info/~barrera>

Grupo de Usuarios de Linux
<http://gul.uc3m.es>

Apaguen sus móviles



Microsoft®
.**net**™

file:///home/pablo/Charlas/Enlaces/Multimedia/Varios/longhorn_beta_logo_grande.gif



Mono



monoTM

- Qué es
- Para qué vale
- Cómo funciona

- Muchos lenguajes
- Muchas bibliotecas, en muchos lenguajes
- Muchas posibilidades, con muchas bibliotecas, en muchos lenguajes
- Muchas cosas que aprender



WARNING

The system is either busy or has become unstable. You can wait and see if it becomes available again, or you can restart your computer.

- * Press any key to return to Windows and wait.
- * Press CTRL+ALT+DEL again to restart your computer. You will lose unsaved information in any programs that are running.

Press any key to continue -



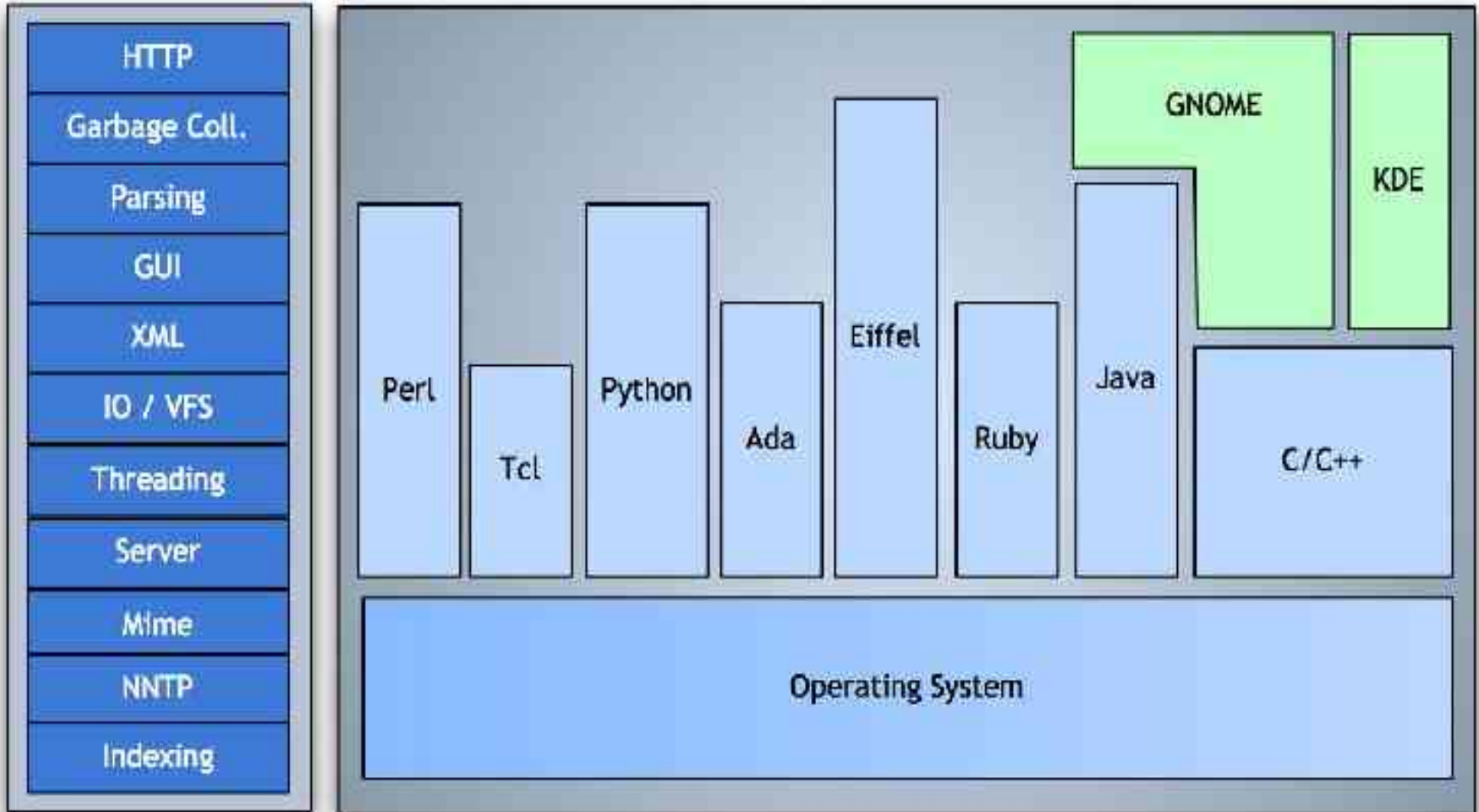
- Buen Funcionamiento
- Prestaciones
- Velocidad de desarrollo
- Segmentation Fault (Core Dumped)



Desarrollar Software con Linux

- Muy buenas herramientas
- Pero son muy viejas
- La plataforma era buena hace 15 años
- Hoy tenemos una plataforma pobre
- Parece estancada
- Falta consistencia: cada API es diferente

Unix: Cada uno por su lado





¿Qué podemos hacer?

- Mejorar la productividad
 - C/C++: no tiene porqué ser la solución
 - Perl, Python, PHP: aplicaciones de tamaño medio
- Objetivos
 - Reducir costes
 - Innovar rápidamente
 - Producir más con menos
- Hoy en día
 - No podemos competir con lo que viene de Microsoft
 - Atraer a los desarrolladores de Windows



Microsoft .NET

- Núcleo de .NET enviado a ECMA
 - Formato de ficheros
 - Opcodes, especificación funcional
 - Lenguaje C#



Puntos a favor de .NET

- Especificación para inter-operabilidad entre lenguajes
- Soporte de lenguajes funcionales e “inseguros”
- C# presenta mejoras sobre Java



Principales ventajas

- Independencia del lenguaje
- Independencia plataforma
- Velocidad (JIT)
- Recolector de basura
- Seguridad (sandbox)
- Soporte para BBDD
- Aplicaciones web
- Servicios web
- Soporte XML



¿Qué es Mono?

- Una implementación libre de .NET
- Implementación libre del framework de .NET
- Impulsado por Ximian, comienza en Julio de 2001
- Objetivos
 - Traer .NET a Unix
 - Sistema de ejecución virtual
 - Biblioteca de clases compatible con .NET
 - JIT para x86



Componentes de Momo

- Un compilador de C#
- CLR (la máquina virtual)
- Clases básicas
- API's (.NET y Linux/Gnome)
- ADO.NET y ASP.NET
- Y más (VB, Java, ...)



Mono hoy en día

- Unos 300 desarrolladores (~30 muy activos)
- Compañías: Novell, Mainsoft, Voelcker Informatik,...
- Reutilización de código libre
- Generación de código nativo:
 - JIT
 - AOT (Avoid startup time)



Arquitecturas soportadas

- Actualmente soporta
 - Unix
 - X86: Linux, xxxBSD
 - PPC: Linux
 - StrongARM: Linux
 - Sparc: Linux y Solaris
 - S390: Linux
 - Windows NT:
 - NT y XP (x86)
 - JIT: por ahora solo en x86



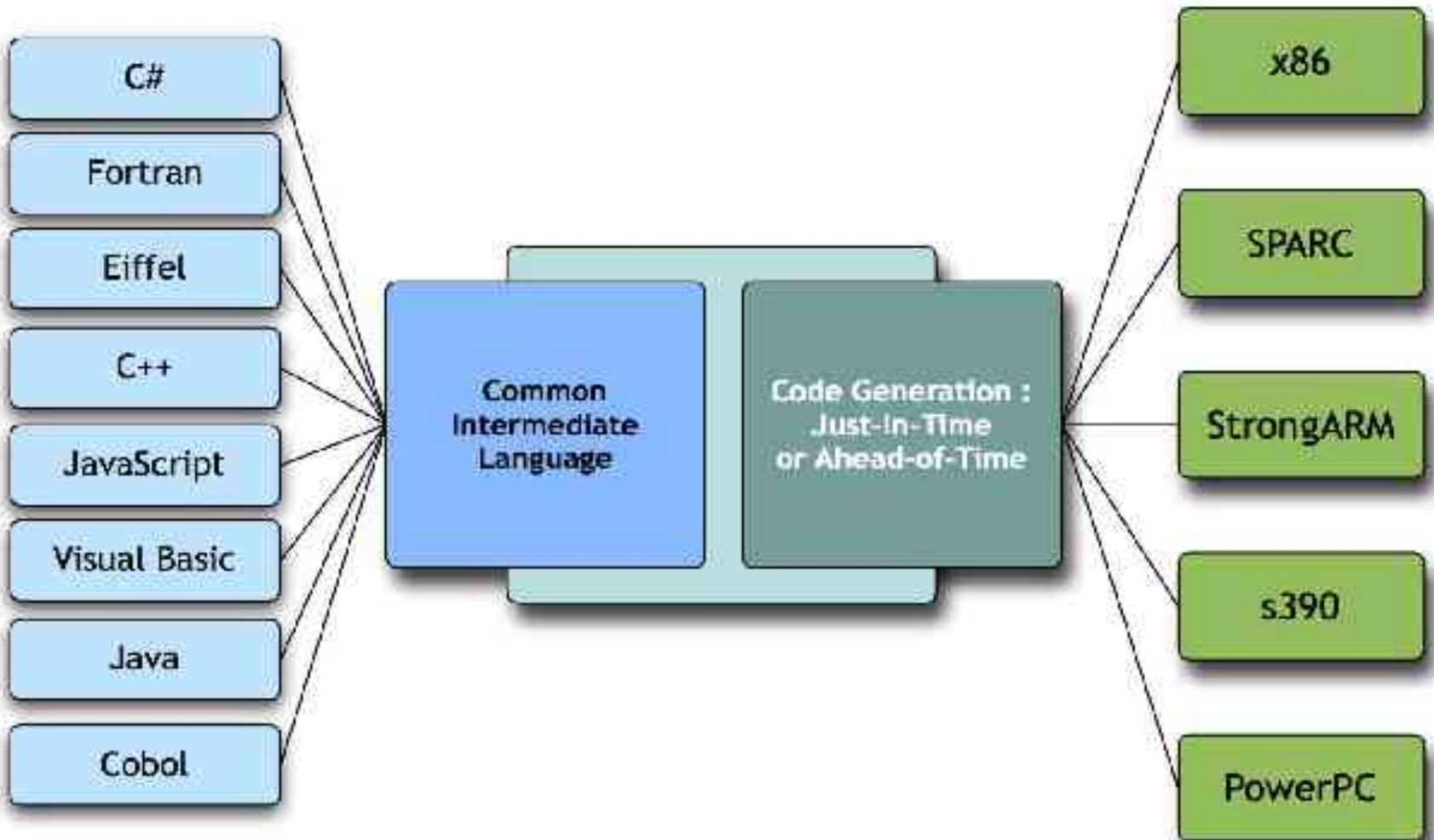


Programación de Mono

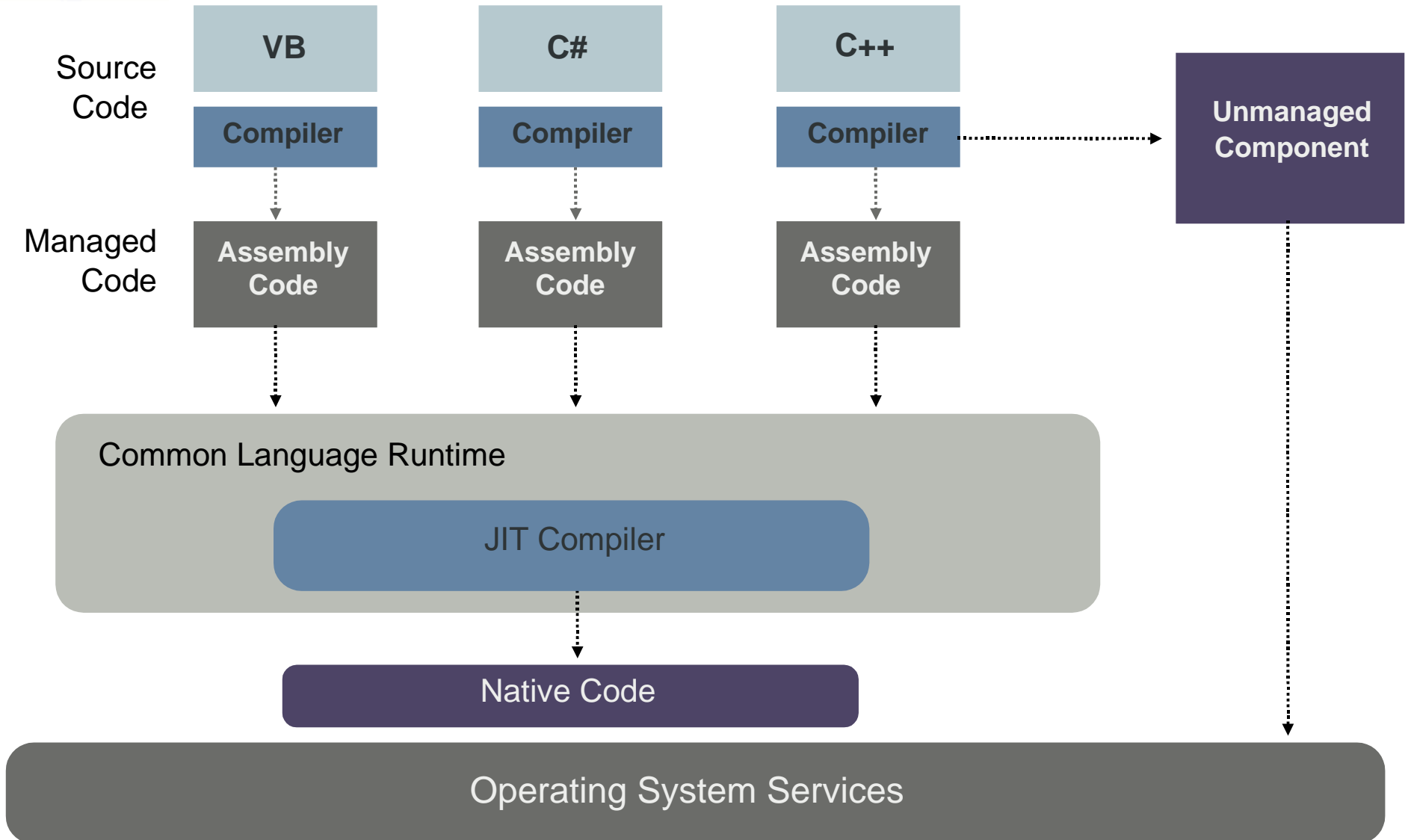
- Compiladores y herramientas libres:
 - Actualmente: C#, Java, Python, Boo, Nemerle.
 - En preparación: VB.NET, Jscript.
- También existen compiladores de terceros:
 - Eiffel, Ada, Fortran, C/C++
- Bibliotecas de terceros escritas en C#, VB, etc.



Multi-lenguaje y multi-plataforma



.NET modelo ejecución





Common Language Infrastructure (CLI)

- Parte del CLR enviado al European Computer Manufacturers Association (ECMA) para estandarización
- Terceras personas pueden implementar CLR para cualquier plataforma
- Enviado para estandarización:
 - C#
 - Semántica de los metadatos
 - Microsoft Intermediate Language (MSIL)
 - Partes de la biblioteca de clases de .NET



The Microsoft Intermediate Language (MSIL)

- Formato intermedio que se ejecuta en el CLR
- No es específico de la arquitectura
- Es una representación intermedia
- Se convierte a código nativo just-in-time por el framework .NET



Managed Code

- **Definición:**

Code that is executed by the CLR. Managed code provides information (i.e., metadata) to allow the CLR to locate methods encoded in assembly modules, store and retrieve security information, handle exceptions, and walk the program stack. Managed code can access both managed data and unmanaged data. Managed data—Memory that is allocated and released by the CLR using Garbage Collection. Managed data can only be accessed by managed code.



Common Type System (CTS)

- .NET soporta integración de lenguajes
 - Todos los objetos derivan de System.Object
- CTS es un sistema de tipos compartido entre compiladores, herramientas y el CLR
- CTS soporta dos tipos diferentes:
 - Referencias
 - Valores

Los dos conjuntos de bibliotecas

ASP.NET

ADO.NET

Windows.Forms (1.2 only)

Microsoft Compatibility Libraries

Java Runtime.

iFolder

Evolution#

GTK#

Open GL

Novell.LDAP

Rendezvous: mDNS

MySQL/Postgress/ZipLib

Apache Mono

Mozilla

Mono Libraries

Compilers and Tools.

Mono Runtime: .NET and Java.



Java IKVM

- IKVM es una JVM sobre Mono
- El soporte de Java es tan bueno como GNU Classpath
- Permite exponer .NET a Java
- Y consumir Java desde .NET
- Java Nativo:
 - Conversión binaria de JVM a CIL
 - Conversión de CIL a código nativo



IronPython

Programado por Jim Hugunin (Jpython) sobre .NET

- Resultados impresionantes:
 - Los lenguajes dinámicos puede ser rápidos (o muy rápidos)

Puede usar todas las clases disponibles en Mono/.NET



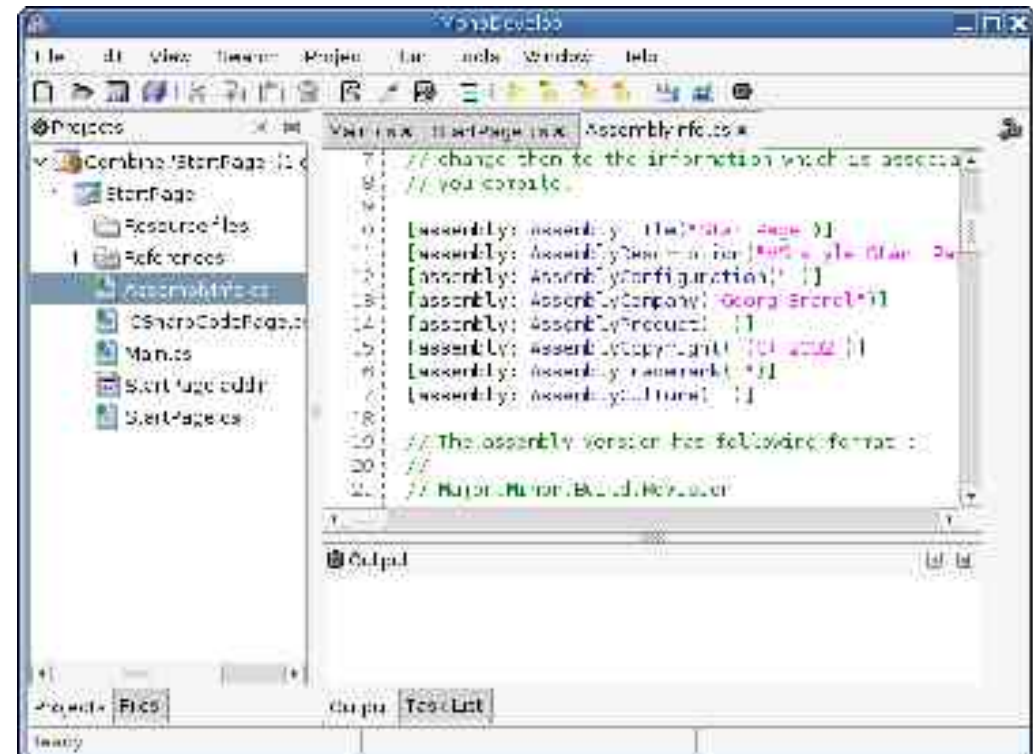
Portabilidad

- Si escribes bien, será portable
- No todas las clases de Microsoft funcionan en Mono
- Cuidado con las bibliotecas nativas
- Muchas aplicaciones requieren pequeños retoques

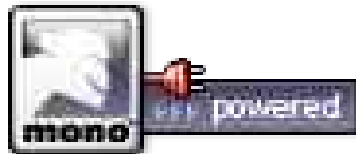
Monodevelop portado en tiempo record

Interfaces de usuario

- Gtk#
- Cocoa#
- Windows.Forms
- wxWidgets



- Mucho más potente que ASP
 - WebForms
- Funciona en Mono
- Integrado con Apache
- mod_mono
- XSP





Aplicaciones de escritorio

- Muine
- F-spot
- Blam
- Tomboy
- Beagle
- Monodoc/Monodevelop
- Muchas más por venir

El lenguaje C#

- Similar a C++ y Java
- Usa espacios de nombres
- Orientado a objetos
 - Todas las clases derivan de `System.Object`
- Tiene herencia simple
 - aunque admite múltiples interfaces
- Incorpora números complejos
- Main se escribe con mayúscula

- Interoperabilidad entre lenguajes
- Entorno de ejecución seguro
- Biblioteca de clases completa y detallada
- Extensible
- Deployment tools
- Servicios web integrados



- Sobre C/C++
 - Seguridad
 - Biblioteca de clases
 - Reflexión
 - Orientación a objetos (C)
- Sobre Java:
 - Múltiples lenguajes
 - Mejor eficiencia en GUI y apariencia

Variables

- Las variables básicas se usan por valor
 - Int, Float,..
 - String
 - Struct
 - ...
- Las variables complejas se usan por referencia
 - Class
 - List
 - Array
 - ...

- Sobrecarga de get y set

```
class A{  
    string name;  
    public string Name{  
        get{ return name; }  
        set{ name = value; }  
    }  
}
```

Delegados

- Similares a punteros a funciones
- Pueden agruparse funciones en un único delegado

```
delegate void MiDelegado (int a);  
MiDelegado foo = new MiDelegado (otrafuncion);  
foo += new MiDelegado (otrafuncionmas);  
foo(10);
```



PInvoke

- Llamadas a código nativo en la plataforma
- Mucho más fácil que JNI

```
[DllImport ("glade-2.0")]
```

```
static extern IntPtr glade_xml_new(string fname, string root, string domain);
```

Ya puedo usar la función

Hay ciertos cambios:

- `const char -> string`
- Punteros pasan a `IntPtr`



Diferencias con Java

- Mejoras frente a Java
 - Estructuras, enumeraciones
 - Propiedades:
`boton.Text = "Pulsa aquí";`
 - Eventos:
`boton.Clicked += delegate { WriteLine("botón pulsado"); }`
 - Atributos
`[Widget] Button boton;`



Otras cosas

- ADO.NET
- ASP.NET
- Gtk#, Gnome#
- Gecko#
- Cairo#
- OpenGL
- ...



Referencias

- <http://www.mono-project.com>
- <http://gnome.org>
- <http://monohispano.org>
- <http://msdn.microsoft.com>

¿Preguntas?