

# Proyecto MONO

**Juantomás García**  
GNOME Hispano

**juantomas@lared.es**

¿Qué es el proyecto MONO?. Estado actual del proyecto. ¿Por qué es interesante para el software libre disponer de la tecnología relacionado con el proyecto MONO?

## 1. Introducción.

Si analizamos el mundo del desarrollo de software descubriremos que el estado actual es bastante mejorable: la tecnología que estamos utilizando es la misma que hace dos décadas y cada vez se esta complicando más, realmente no se ha evolucionado demasiado, y, lo que es peor, la curva de aprendizaje para los nuevos desarrolladores es más dura.

Por otra parte la complejidad de los sistemas operativos y el número de versiones que coexisten concurrentemente es mayor. Ante esta situación parece sensato realizar una revisión de las herramientas actuales y la arquitectura de desarrollo.

Y aquí es donde entra el proyecto MONO. El proyecto tiene como objetivo crear una implementación libre de algunas herramientas y parte de la arquitectura de .NET . La arquitectura .NET ha sido propuesta por Microsoft. Parte de esta tecnología se basa en un estándar propuesto a la ECMA y lo interesante es que tiene ideas muy buenas. Realmente son tan buenas que el mundo del software libre debe poder disponer de ellas sin que tengamos que esperar a que Microsoft lo haga. Este es el objetivo del proyecto MONO.

Actualmente la reutilización de software no es óptima. Aunque el modelo de software libre nos permite reutilizar el código con mucha facilidad. El precio en horas o en dificultad de usar código en nuestros proyectos es poco óptimo y esto se debe mejorar.

Disponer una biblioteca de funciones en forma de librería dinámica esta bien pero no soluciona el problema.

La interoperabilidad entre lenguajes no es buena. Una de las bendiciones del software libre es la cantidad de posibilidades que tenemos a la hora de poder seleccionar un lenguaje de desarrollo. La mala noticia es que aunque prácticamente todos disponen de la posibilidad de interoperar con otros lenguajes, ésta no es fácil, ni sencilla.

Desarrollar aplicaciones no es fácil. Con el modelo actual de desarrollo, el diseño y la creación de proyectos a partir de un determinado tamaño de proyecto se complica demasiado. Lo ideal es contar con un entorno que nos permita desarrollar prototipos, reutilizar código y sobre todo hacer aplicaciones sencillas sin un esfuerzo significativo. Un entorno parecido al "entorno de programación" del Visual Basic, pero eliminando las carencias que tiene.

Código gestionado (Managed Code). Mayoritariamente los lenguajes utilizados en la actualidad son procedurales y no gestionados. Las ventajas del código gestionado: GC (Recolector de Memoria ), Manejo de Excepciones, etc, permiten que se mejore la calidad y eficacia de los proyectos.

En esta ponencia se va a exponer de una manera global cuales son los objetivos del proyecto, los componentes e ideas de .NET que son más interesantes.

## **2. Que es proyecto MONO?**

El proyecto MONO ha sido propuesto por Miguel de Icaza para crear un entorno de desarrollo y producción utilizando alguna de las tecnologías propuestas en la arquitectura .NET.

Básicamente el proyecto MONO ha comenzado con un compilador de C#. La idea inicial es explorar hasta que punto el lenguaje C# y la arquitectura propuesta para su máquina virtual (VM) puede ser una opción que mejore las condiciones de producción de los desarrolladores de software libre.

Como lenguaje de programación C# recuerda mucho al lenguaje JAVA. Hay varios factores que han hecho que C# genere un interés en la comunidad de desarrolladores. La primera es que las especificaciones del lenguaje están siendo estandarizadas por la ECMA desde el principio. Esta ya es un hecho excepcional viniendo de Microsoft.

Otra parte interesante de usar C# es el conjunto de clases que lleva asociado. La idea es que este conjunto de clases permita realizar cualquier tipo de programación. La parte innovadora es que estas clases, y en general el uso de librerías, esta también normalizado de modo que estas pueden ser utilizadas por otros lenguajes. Para que

estas librerías puedan ser utilizadas estos lenguajes tienen que tener compiladores que generen CLI.

Además C# tiene una gestión de memoria y recolección de memoria no usada que le hace muy interesante. Realmente esta opción está implementada en la máquina virtual que ejecuta el código CLI generado.

El compilador de C# del proyecto MONO está siendo desarrollado en C#. De las múltiples alternativas para realizar el compilador se ha elegido ésta por razones "educativas". Desarrollar un compilador es un ejercicio de programación complicado y hacerlo en el lenguaje que se está implementando hace que se descubra todo el potencial que tiene dicho lenguaje.

Existen otros proyectos libres interesados en la creación de un compilador de C#, pero no ha habido una convergencia por disparidad de criterios o herramientas utilizadas.

Aunque C# va a ser el primer lenguaje soportado por el compilador del proyecto MONO, ya se están empezando a implementar otros lenguajes como VB Object. No está previsto de momento el soporte de otros lenguajes en el compilador pero la arquitectura actual de MONO permite con bastante facilidad que se puedan integrar otros lenguajes gestionados como Python .NET o Perl .NET .

### **3. Estado del proyecto MONO.**

En la primera fase del proyecto MONO se está trabajando en tres áreas:

#### **Compilador C#.**

Esta ha sido la primera fase del proyecto. Solo existe un compilador completo de C# actualmente es el desarrollado por MS para Windows. El compilador de C# está a un 50% y está desarrollado en C#. La idea era hacerlo en C# para aprender a programar y conocer al 100% el lenguaje. Actualmente parsea el código y es capaz de verificar la sintaxis y generar el 95% de los árboles de código. La idea es que genere CLI y código C. Cuando esté terminado se compilara con el compilador de C# y generara su propio código fuente en CLI y C haciéndolo transportable a otros sistemas.

Actualmente está compilando programas sencillos estilo: impresión de "hello world", cálculo de series de fibonacci o creación de una ventana basada en las GTK. No solo compila estos códigos sino que además genera ejecutables que son correctamente interpretados por el JIT de MONO y el de MS.

#### Clase de Librerías C#.

Esta es la base fundamental del lenguaje C#. Realmente C# como lenguaje no es más que otro lenguaje. Lo que hace realmente potente a C# y todos los lenguajes que utilicen la arquitectura .NET son las bibliotecas de clases que pueden ser utilizadas por cualquier lenguajes. Dentro del proyecto MONO y solo en unos meses ya están implementadas el 50% de las clases.

#### Máquina Virtual.

La máquina virtual que ejecute CLI es la parte clave del proyecto. Si se consigue que funcione óptimamente y que cualquier lenguaje pueda generar CLI podemos estar ante la solución a los problemas de portabilidad e integración entre lenguajes.

En el diseño del CLR se esta teniendo en cuenta la portabilidad desde el comienzo del diseño. Esto nos permitirá que funcione bajo otros sistemas operativos sin demasiados esfuerzos.

## **4. ¿Por qué es interesante el proyecto MONO?**

El proyecto MONO pretende convertirse en la opción libre a la arquitectura .NET de Microsoft. Pero esto genera muchas cuestiones en la comunidad. ¿Debe existir una opción libre que soporte las virtudes de .NET? ¿Estamos apoyando a Microsoft y su iniciativa? ¿Se están perdiendo recursos implementando MONO? ¿Volverá Microsoft a utilizar la estrategia de primero ayudar y luego ahogar a sus acompañantes de viaje?

Mi punto de vista sobre el proyecto es que debemos promocionarlo. Hay demasiadas buenas ideas que no se deben desaprovechar aunque vengan del departamento de ingeniería de Microsoft.

Microsoft no va a solucionar nuestros problemas, ni mejorar las condiciones de programación y desarrollo del software libre. Luego haga lo que haga solo las iniciativas libres nos van a beneficiar con toda seguridad.

## **5. Referencias e Información relacionada con el**

# Proyecto

1. *Proyecto MONO* <http://www.go-mono.com>
2. *Proyecto MONO FAQ* <http://www.go-mono.com/faq.html>
3. *Documentación de la ECMA en INTEL.*  
<http://developer.intel.com/software/idap/ecma>
4. *Documentación de las clases en MSDN.*  
[http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpref/html/cpref\\_start.asp](http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpref/html/cpref_start.asp)