

La plataforma .NET: ¿el futuro de la Web?

Unai Extremo Baigorri
uextremo@eside.deusto.es

Borja Sotomayor Basilio
borja@borjanet.com

Introducción

La World Wide Web, o simplemente *la Web*, fue creada en 1990 como un modesto sistema de intercambio de documentos, y ha progresado hasta convertirse en un potente y versátil medio de comunicación. Gracias a la Web hoy es posible acceder fácilmente a todo tipo de información, comprar libros y CDs desde nuestro ordenador, reservar billetes de avión, y mucho más.

Durante los últimos cinco años, las tecnologías sobre las que se apoya la Web han ido evolucionando a un ritmo más o menos uniforme. Sin embargo, la **Plataforma .NET** (.NET) de Microsoft, de la que tanto se habla últimamente, promete revolucionar la Web, tanto para los usuarios como

para los programadores. En este artículo se expone la evolución de la Web, las características generales de .NET, y se analiza si esta nueva plataforma puede suponer el empujón definitivo para que se produzca un nuevo «salto evolutivo» en la Web.

La evolución de la Web

La World Wide Web fue creada por Tim Berners-Lee en 1990, un año en el que Internet era utilizado casi exclusivamente por investigadores y profesores de un puñado de selectas universidades. De hecho, la Web inicialmente no era más que un sistema para intercambiar documentos científicos. El principal atractivo de la web era el hecho de que los documentos de la web (o **páginas web**) se escribían con el lenguaje

HTML. Este lenguaje permite escribir *hipertexto*, un tipo de texto en el que se puede «saltar» fácilmente de un documento a otro haciendo clic sobre un *hiper enlace* (o *enlace*). Por eso a esta primera fase de la Web se le conoce como *Web Orientada al Hipertexto*.

Inicialmente, la Web ofrecía una interactividad prácticamente nula (los usuarios se limitaban a acceder a documentos estáticos). Sin embargo, a mediados de los 90 la Web (junto con el resto de Internet) abandonó los confines de la comunidad académica y empezó a ser accedida por el público general, que demandaba una mayor interactividad. Esto provocó la llegada de la siguiente fase de la Web: *la Web Orientada a las Bases de Datos*. Las

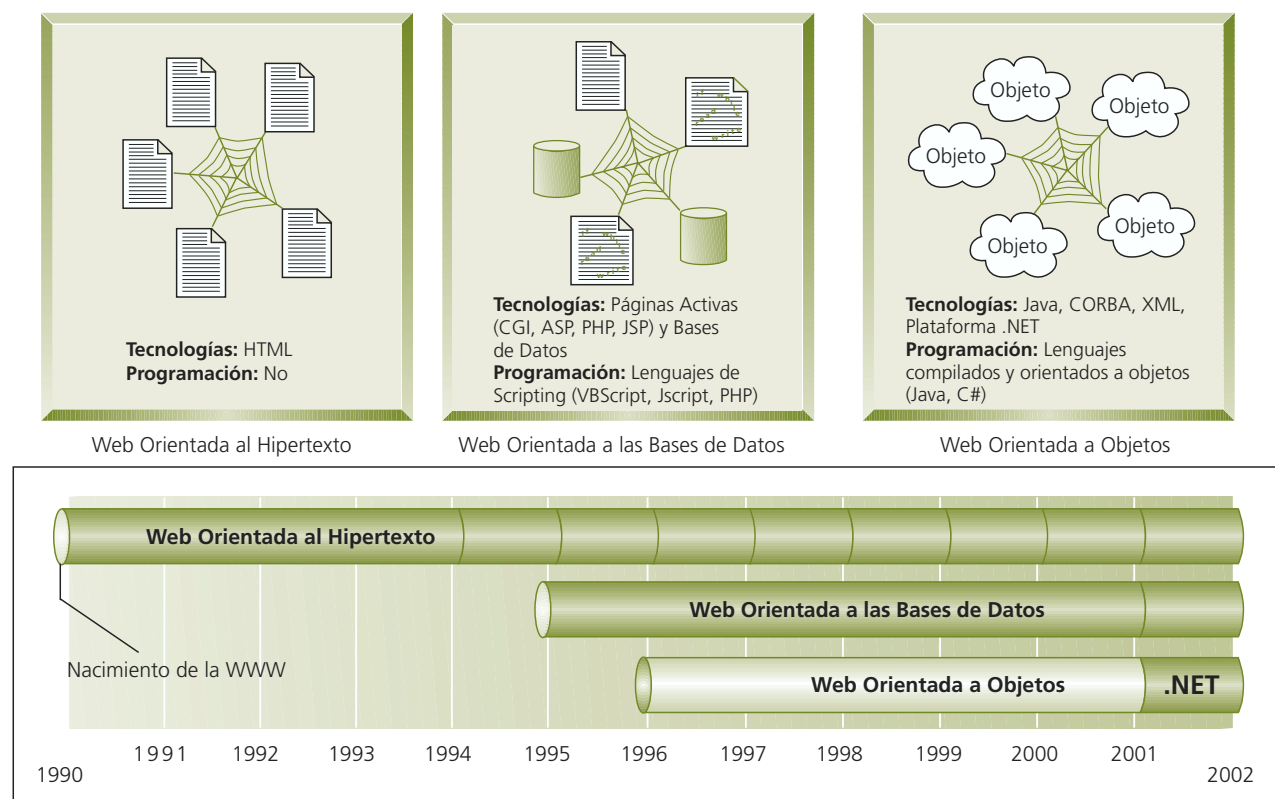


Figura 1. La evolución de la World Wide Web

páginas web pasaron de ser documentos estáticos a ser documentos *dinámicos*, donde los contenidos no eran constantes e inmutables sino que solían generarse dinámicamente a partir de una base de datos. Además, la web ganó en interactividad, no sólo por la incorporación de tecnologías multimedia, sino gracias a las **páginas activas**, un tipo de página web que puede «reaccionar» a las acciones del usuario (generalmente a través de un formulario). Por ejemplo, un usuario puede acceder a una guía telefónica e introducir en un formulario unos parámetros de búsqueda. Al pulsar el botón de envío, la página activa recoge los datos del formulario, consulta la base de datos, y genera dinámicamente una página web con los resultados de la consulta. A lo largo de esta fase surge el concepto de **aplicación web**, que no es más que una aplicación que, en lugar de utilizar una tradicional interfaz de ventanas, utiliza las tecnologías de la web: una interfaz consistente en páginas web y una lógica interna implementada con páginas activas. La guía telefónica por web, por ejemplo, es una aplicación web.

La Web Orientada a las Bases de Datos, sin embargo, tiene varios inconvenientes:

- Existe interactividad entre el usuario y la aplicación web. Sin embargo, es difícil que una aplicación web pueda comunicarse con otra aplicación web.
- Para la programación se utilizan, sobre todo, lenguajes de *scripting*. Estos lenguajes son poco potentes y producen código difícil de mantener y reutilizar.

La Web Orientada a Objetos

La Web Orientada a Objetos (WOO) es, en teoría, la siguiente fase evolutiva de la Web, en la que se solucionan casi todos los inconvenientes de la fase anterior. La WOO parte de aplicar el paradigma de la orientación a objetos a la Web en el sentido más amplio imaginable. Es decir, no se limita a utilizar lenguajes orientados a objetos para la programación, sino que propone concebir la Web como una enorme colección de objetos (en el sentido informático) que pueden

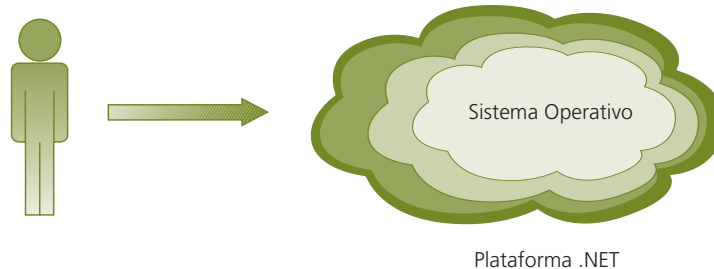


Figura 2. La plataforma .NET abstrae al programador del SO

realizar cualquier trabajo, desde acceder a una base de datos hasta realizar complicados cálculos matemáticos. En la WOO, estos objetos pueden ser accedidos desde cualquier punto utilizando una serie de protocolos y lenguajes estándares, con lo que ahora resulta muy fácil que una aplicación web pueda comunicarse con otra. De hecho, en la WOO se pasa a hablar de **aplicaciones web distribuidas**, donde las distintas partes de una aplicación web pueden estar dispersas en la web y comunicarse utilizando objetos.

Hay dos tecnologías actualmente que permiten materializar una Web Orientada a Objetos: Java (junto con CORBA) y **.NET**. Java y CORBA, a pesar de ser los más veteranos (Java data de 1996 y CORBA de 1991) y de ser ampliamente utilizadas en la Web, no han conseguido impulsar un cambio de paradigma en la Web, posiblemente debido a la falta de apoyo de grandes compañías y de muchos desarrolladores. En 2001, **.NET** entra en escena y, a pesar de su juventud, es posible que consiga impulsar por fin la llegada de la WOO.

¿Que es .NET?

En 1998 un equipo de trabajo de Microsoft comenzó a trabajar en un proyecto que denominaron Next Generation Windows Services (NGWS). Este equipo se fusionó con el grupo encargado de desarrollar la versión 7 del Visual Studio con el fin de desarrollar un entorno de ejecución común para todos los lenguajes incluidos en él de forma que permitiese a terceras empresas crear lenguajes adaptados al entorno. Finalmente, en el 2000 Microsoft dio a conocer todo este trabajo que denominaron Microsoft.NET.

De esta forma Microsoft.NET es el conjunto de nuevas tecnologías en las que han estado trabajando a lo largo de dos años. Este conjunto de nuevas tecnologías podrían resumirse en las siguientes:

- Plataforma .NET
- SDK de la plataforma .NET
- Visual Studio.NET
- Servicios Web
- Servidores para empresas

En este artículo nos centraremos en la plataforma .NET y sus características, así como en las novedades que incorpora el Visual Studio.NET respecto a sus antecesores.

La Plataforma .NET

La plataforma .NET es una capa de software que se coloca entre el Sistema Operativo (SO) y el programador y que abstrae los detalles internos del SO. Las características fundamentales de esta plataforma son las siguientes:

- Portabilidad: Debido a la abstracción del programador respecto al SO, una aplicación .NET puede ser ejecutada en cualquier SO de cualquier máquina que disponga de una versión de la plataforma. En estos momentos la plataforma .NET tan sólo está disponible para la familia Windows aunque se está desarrollando una versión para Linux de Corel.
- Multilenguaje: Cualquier lenguaje de programación puede adaptarse a la plataforma .NET y ejecutarse en ella.
- Interoperabilidad: La interoperabilidad entre diferentes trozos de código escritos en diferentes lenguajes es total.

Microsoft define la plataforma .NET como «un entorno para la construcción, desarrollo y ejecución de servicios web y otras aplicaciones que consiste en tres partes fundamentales: el Common Language Runtime (entorno de ejecución), las Framework Classes (clases de la plataforma) y ASP.NET». Vamos a ver en detalle cada una de estas tres partes y sus características.

El Common Language Runtime (CLR) es el entorno de ejecución de la plataforma .NET, y constituye su núcleo. El CLR es el entorno en el que se ejecutan nuestras aplicaciones .NET. Estas aplicaciones pueden escribirse en cualquiera de los múltiples lenguajes que ofrece .NET (Visual C#.Net, Visual Basic.NET...) que en lugar de compilarse a código máquina (que es lo más habitual) se compila a un lenguaje intermedio llamado Microsoft Intermediate Language o MSIL (Lenguaje Intermedio de Microsoft). El MSIL es el único lenguaje que el CLR comprende. Esta característica permite, por ejemplo, utilizar un fragmento de código en una aplicación sin depender del lenguaje en el que está escrito.

Las Framework Classes forman otra de las capas que constituyen la plataforma .NET. Esta capa provee al programador de servicios, estructuras y modelos de objetos para datos ADO.Net (siguiente generación de ADO), entrada/salida, seguridad, manejo de documentos XML...

ASP.NET es la parte más importante de la capa superior de la plataforma .NET. Para los programadores web ASP.NET es mucho más que una nueva versión de la tecnología ASP ya que supone una nueva idea y forma de programar aplicaciones Web. ASP.NET provee una plataforma más robusta para el desarrollo de aplicaciones, y ofrece mayores beneficios. A diferencia de ASP, los ASP.NET permiten separar limpiamente la lógica de la aplicación de la interfaz. De esta manera, el programador puede centrarse exclusivamente en la lógica de la aplicación sin preocuparse de los detalles de la interfaz.

ASP.NET además incorpora un nuevo concepto en el desarrollo de tecnologías Internet: los Servicios Web. Estos servicios representan un paso más hacia la descentralización del software en la red y de hecho, son un factor clave para el desarrollo de una

web orientada a objetos. Los servicios Web permiten a los desarrolladores construir aplicaciones combinando recursos locales y remotos para una solución distribuida e integrada. La comunicación a través de la web se hace utilizando el protocolo SOAP, lo cual no supone ningún problema para el desarrollador ya que es la plataforma .NET la que se encarga de tratarlo.

Visual Studio .NET

Visual Studio .NET es la herramienta que Microsoft distribuye junto a la plataforma que permite construir y desarrollar aplicaciones .NET. Esta nueva versión no revoluciona la anterior sino que se limita a añadir una serie de nuevas características y funciones. Es una mezcla de los diferentes entornos que Microsoft utilizaba hasta ahora (Visual Basic 6 IDE, Visual InterDev...).

La principal diferencia respecto a versiones anteriores es que Microsoft utiliza exactamente el mismo entorno para todos los lenguajes incluidos en la plataforma. De hecho, este entorno está creado para poder manejar proyectos que usen más de un lenguaje a la vez, teniendo en cuenta la característica multilenguaje de la plataforma.

El Visual Studio.NET incluye los siguientes lenguajes de programación:

- Visual Basic.NET es la adaptación de Visual Basic a la plataforma .NET
- Visual C++.NET permite tanto escribir código adaptado a la plataforma .NET (código gestionado o compilado a MSIL) como código C++ nativo (código no gestionado o no compilado a MSIL).
- Visual C#.NET (C Sharp) es un nuevo lenguaje de programación orientado a objetos con el que se ha desarrollado parte de la plataforma .NET.
- Visual J#.NET (J Sharp) es la adaptación de Visual J++ para la plataforma .NET.

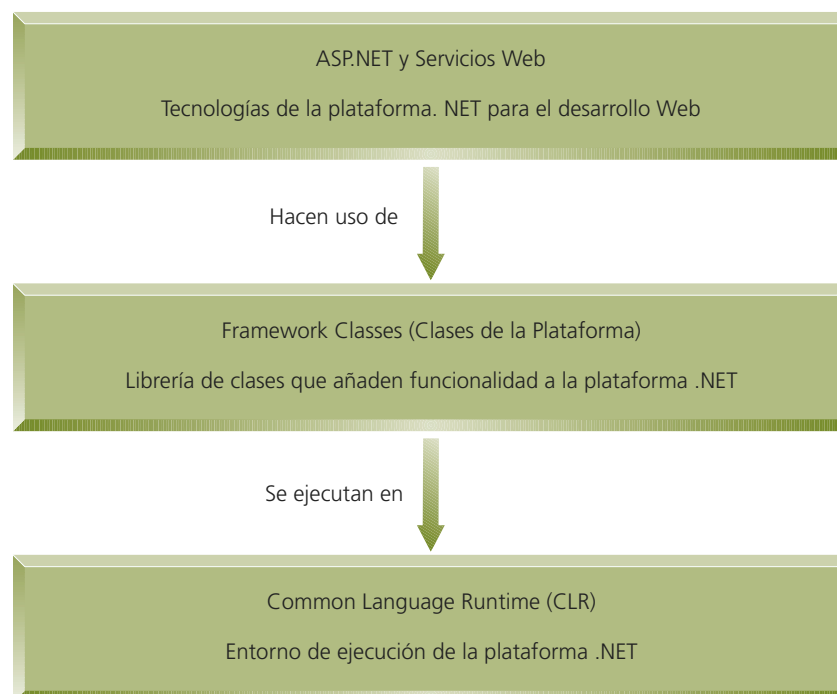


Figura 3. Las capas de la plataforma .NET

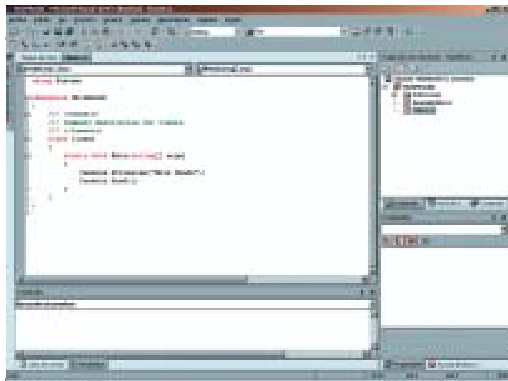


Figura 4. Visual Studio .NET

Un nuevo lenguaje de programación: Visual C#.NET

C# (se lee C Sharp) es el nuevo lenguaje de programación introducido por Microsoft en la plataforma .NET. Es el lenguaje mejor adaptado a la plataforma ya que ha sido exclusivamente creado para trabajar sobre ella. De hecho, las Framework Classes o clases de la plataforma han sido programadas con C# luego a pesar de ser un lenguaje de reciente creación ya ha servido como base para un desarrollo complejo, lo cual ha contribuido enormemente a su mejora.

C# es definido por muchos autores como un sub-conjunto más simple y más seguro de Visual C++ para crear código gestionado (código ejecutable por el CLR de la plataforma .NET). El código fuente de C#, al igual que todos los lenguajes de la plataforma, es compilado a MSIL y ejecutado por el CLR.

Según Microsoft, C# es «un lenguaje de programación con la potencia de C, la productividad de Visual Basic y la elegancia de Java». Ciertamente, si vemos un trozo de código de C# nos daremos cuenta del asombroso parecido con Java. A su vez tenemos la sintaxis utilizada en Visual C++ y determinadas características que lo hacen muy potente (como la sobrecarga de operadores) combinada con la sencillez y facilidad de Visual Basic que hace que sea un lenguaje muy productivo.

Desde otro punto de vista, Microsoft ha creado un lenguaje completo, orientado a objetos, que se acopla perfectamente con el desarrollo Web. Hasta ahora Microsoft hacía uso de

lenguajes de script para desarrollar productos para la web (Visual Basic Script, JavaScript...) pero en estos momentos la web necesita algo más que un lenguaje de script para su desarrollo. Necesita lenguajes estructurados orientados a objetos que puedan separar la lógica de la aplicación de la presentación. Microsoft ofrece su solución, la plataforma .NET, y dentro de ésta un lenguaje que se adapta perfectamente: Visual C#.NET.

Conclusiones

En este artículo hemos podido ver que la web ha crecido espectacularmente desde su nacimiento. Microsoft ha demostrado que apuesta firmemente por las actuales tendencias de la web. Tanto así que ha creado una plataforma de desarrollo *totalmente nueva* en lugar de simplemente ampliar las tecnologías de las que actualmente dispone (COM/MTS, DCOM, ASP...).

La plataforma .NET es en potencia un vehículo excelente para cambiar notablemente la web promoviendo por fin una web orientada a objetos. Sin embargo, como ya se ha comentado en este artículo, hay otras tecnologías como Java y CORBA que llevan varios años promoviendo este mismo cambio en la web. La llegada de una propuesta tan fuerte como la de Microsoft no solamente acelerará este cambio sino que, sin duda, va a desencadenar un duro enfrentamiento entre ambas tecnologías.

A nuestro parecer ambas tecnologías podrían convivir pacíficamente. No se puede decir en términos absolutos que una plataforma sea mejor que la otra. En función de nuestras necesidades, teniendo en cuenta los puntos fuertes y débiles de cada plataforma, optaremos por una u otra. Por ejemplo:

- Mientras que Microsoft hace especial énfasis en la característica multilenguaje de su plataforma, Java se preocupa más por la interoperabilidad entre distintas máquinas (característica multiplataforma). **.NET** es «técnicamente» multiplataforma aunque actualmente tan sólo existen versiones de la plataforma para SO de

la familia Windows. Java, a su vez, permite utilizar múltiples lenguajes a través de CORBA pero de una manera menos transparente que **.NET**.

- **.NET** integra en la propia plataforma soporte para XML y SOAP mientras que Java simplemente incorpora una serie de librerías para su manejo.

- **.NET** proporciona un mayor nivel de abstracción al programador que Java haciéndole más sencillo su trabajo. Sin embargo, esto se hace a costa de ocultar ciertos detalles internos que podrían serle útiles.

Este nuevo impulso a la web por parte de Microsoft, sin duda, debería provocar una reacción por parte de la comunidad Java para mantener su competitividad. Aun así el efecto inmediato de la plataforma .NET todavía es difícil de prever ya que aún se encuentra en fase de Beta (a diciembre de 2001). Lo que está claro es que no dejará indiferente a nadie en el mundo tecnológico.