

MICROSOFT NET

Introducción

.NET es la plataforma de Microsoft para servicios Web XML, la siguiente generación de software que conecta nuestro mundo de información, dispositivos y personas de una manera unificada y personalizada.

La Plataforma .NET permite la creación y uso como servicios de aplicaciones, procesos y sitios Web basados en XML, que compartan y combinen información y funcionalidad entre ellos por diseño, en cualquier plataforma o dispositivo inteligente, para proveer soluciones a la medida para empresas e individuos.

La Plataforma .NET incluye una familia de productos integral, construida en estándares de la industria y de Internet, que provee servicios Web XML para cada aspecto del desarrollo (herramientas), administración (servers), uso (bloques ensamblables y clientes inteligentes) y experiencias (ricas experiencias de usuario). .NET será parte de las aplicaciones, herramientas, y servers de Microsoft que usa actualmente - así como será parte de nuevos productos que extienden las capacidades de los servicios Web XML a todas sus necesidades de negocio.

Este trabajo examina lo que significa la visión Microsoft® .NET para profesionales de tecnología y de negocios y para el futuro de las aplicaciones de software. Describe la visión .NET y explica las ventajas competitivas que brindará a los negocios.

¿Qué es Microsoft .net?

.NET es la plataforma de Microsoft para servicios Web XML, la siguiente generación de software que conecta nuestro mundo de información, dispositivos y personas de una manera unificada y personalizada.

La Plataforma .NET permite la creación y uso como servicios de aplicaciones, procesos y sitios Web basados en XML, que compartan y combinen información y funcionalidad entre ellos por diseño, en cualquier plataforma o dispositivo inteligente, para proveer soluciones a la medida para empresas e individuos.

Microsoft® .NET™ ayuda a acelerar el cambio de la industria de la informática a un modelo de computación distribuido basado en Internet.

Con base en los estándares abiertos de Internet, Microsoft .NET es la plataforma de servicios Web XML para potenciar aplicaciones, servicios y dispositivos a fin de que trabajen juntos para permitir el acceso y acción a la información en cualquier momento, en todo lugar y desde cualquier dispositivo; proporcionando desarrollo simplificado y experiencias ricas de usuario, así como oportunidades sin precedentes para los negocios.

¿Qué es .net?

.NET es la plataforma Microsoft para la siguiente generación de software que conecta el mundo de información, dispositivos electrónicos y personas de una manera unificada y personalizada.

Se compone de la nueva plataforma .NET y de experiencias integradas a la plataforma.

Plataforma .net

Desde el punto de vista tecnológico, Microsoft .NET es la plataforma .NET y las experiencias .NET construidas sobre la plataforma. La plataforma .NET incluye:

- Herramientas - para construir aplicaciones y servicios Web XML (el Entorno .NET y Visual Studio.NET).
- Servers - con los cuales construir, proveer y desplegar esas aplicaciones y servicios (Windows 2000 Server y los Servidores Empresariales .NET) ,
- Servicios - un conjunto central de servicios ensamblables .NET (servicios "Hailstorm"),
- Software cliente - el software que potencia los dispositivos inteligentes, permitiendo a los usuarios interactuar y experimentar la plataforma .NET,
- Experiencias - la combinación de los componentes anteriores de la plataforma .NET permiten experiencias de usuario más personales e integradas - experiencias .NET.

Analizando cada uno de estos componentes:

Herramientas del desarrollador

Estas son el modelo de programación, ambiente de desarrollo y herramientas para el desarrollo de la siguiente generación de aplicaciones distribuidas. Las herramientas de desarrollo tales como Visual Studio.NET, hacen que escribir servicios Web XML y aplicaciones Web sean tan simples como sea posible. El Entorno .NET y Visual Studio.NET, ofrecen la mejor manera, más sencilla, más rápida y más económica de escribir servicios Web XML.

Servers

A través de la plataforma .NET, los negocios y otros procesos podrán convertir procesos clave en servicios Web XML. Desde una cita al médico hasta el catálogo de una librería o el mecanismo que determine la tasa de cambio apropiada para las monedas mundiales - los servicios Web XML creados y proveídos por una compañía o individuo pueden ser usados por millones en cualquier número de combinaciones para producir una variedad de experiencias computacionales altamente personales e inteligentes.

Este modelo computacional distribuido pondrá demandas crecientes en las infraestructuras de servidores, requiriendo una plataforma ágil y altamente confiable.

Los Servidores Empresariales .NET de Microsoft representan la más rápida y confiable manera de integrar, administrar, y habilitar aplicaciones para la Web dentro de la empresa. Microsoft provee ofertas de servidores listos para la empresa, con la confiabilidad, desempeño, escalabilidad y administración requeridas por las soluciones empresariales. Una red global de asistencia soporta esto, los cuales junto con un rico conjunto de socios de la industria proveen soluciones completas basadas en Internet y la intranet.

Servicios

El primer de los servicios ensamblables .NET que serán ofrecidos incluyen los servicios "Hailstorm" que están enfocados al usuario. Microsoft está desarrollando un conjunto de servicios Web XML centrados en el usuario, nombre código "HailStorm" (Tormenta de Granizo), el cual será necesario para muchas aplicaciones. Los servicios de Hailstorm están orientados alrededor de las personas, en lugar de alrededor de un dispositivo, aplicación, servicio o red específicos. Ponen a los usuarios en control de sus propios datos e información, y les permiten accederlos en cualquier momento desde cualquier dispositivo. También protegen la información personal del usuario permitiéndoles controlar quién tiene acceso a ella y proveyendo un nuevo nivel de facilidad de uso y personalización. Los servicios "Hailstorm" son el primer conjunto de servicios Web XML contruidos por Microsoft.

Software cliente para dispositivos inteligentes

El software cliente permite que las PCs, computadoras portátiles, estaciones de trabajo, teléfonos inteligentes, computadoras de mano, PCs de escritorio, consolas de juegos y otros dispositivos inteligentes operen en el universo .NET.

Un dispositivo inteligente es:

- Inteligente sobre usted. Utiliza su identidad .NET, y datos de perfil para simplificar su experiencia de usuario.
- Inteligente acerca de la red. Responde a las restricciones de ancho de banda, proporciona soporte tanto para el uso en línea y fuera de línea de aplicaciones, y entiende qué servicios están disponibles.
- Inteligente acerca de la información. Puede acceder, analizar y actuar sobre datos en cualquier lugar y momento.
- Inteligente sobre otros dispositivos. Descubre otras PCs, dispositivos inteligentes, servidores e Internet y sabe cómo proporcionar servicios a otros dispositivos. También es inteligente acerca de acceder información desde la PC.

- Inteligente sobre software y servicios. Las aplicaciones y datos se presentan de manera óptima para el factor de forma actual. Los métodos de entrada y conectividad son adecuados para el dispositivo actual proporcionando una excelente interacción. Consume servicios Web utilizando XML, SOAP, y UDDI y es programable y ampliable por los desarrolladores.

Algunos de los paquetes de software para dispositivos inteligentes en los que Microsoft está trabajando incluyen Windows® XP, Windows CE, Windows Embebido, el Entorno .NET, y el Entorno .NET Compacto.

Experiencias .NET

Las experiencias .NET son los medios para que los usuarios interactúen con las aplicaciones .NET. Las experiencias .NET son combinaciones de los servicios "Hailstorm" centrados en el usuario, otros servicios Web XML, y código de aplicación local, que son presentados a través de un dispositivo inteligente. Microsoft está tornando cuatro populares productos en experiencias .NET:

- Microsoft Office XP está tomando los primeros pasos destinados a ofrecer una experiencia .NET a los trabajadores expertos. Futuras versiones de Office integrarán completamente los servicios Web XML, a la vez que también expondrán funcionalidad clave a otras aplicaciones en la forma de servicios Web XML.

- MSN®, incluyendo el uso del cliente local MSN Explorer, está en vías de crear una experiencia .NET destinada al consumidor.

- El portal de pequeñas empresas de bCentral™ proveerá los servicios Web XML necesarios para pequeñas empresas (tales como gestión de inventarios) y eventualmente consumirá importantes servicios Web XML (tales como servicios de remate)

- El sistema de desarrollo Visual Studio® proporcionará una experiencia .NET para los desarrolladores, exponiendo información de MSDN y lineamientos de codificación específicos

de la compañía directamente en las herramientas que los desarrolladores utilizarán.

Las experiencias .NET de usuario tienen las siguientes características:

- Son ofrecidas a través de múltiples dispositivos. En lugar de escribir un servicio Web XML diferente y una aplicación .NET diferente para cada dispositivo posible, las aplicaciones .NET pueden leer las características del dispositivo que el usuario final elige para acceder y le ofrece una interface adecuada.

- Utilizan servicios Web XML. Las experiencias .NET aprovechan los servicios Web XML cuando están conectados a la red, consolidando más información relevante para los usuarios, a fin de resolver los problemas cotidianos.

El usuario es el núcleo. Las experiencias .NET se enfocan en los usuarios finales, utilizando los servicios ensamblables ("Hailstorm") para identificación de usuarios, preferencias, notificaciones y datos del usuario. Debido a que los datos del usuario son administrados por servicios ensamblables en lugar de una aplicación, los usuarios tienen el control de sus propios datos, pueden asegurar su exactitud y pueden coordinar los datos entre las diferentes aplicaciones y servicios.

Servicios base de .net

Para hacer realidad la visión .NET, Microsoft operará los siguientes servicios en la nube de Internet:

- a) **Identidad .NET Passport.** Personalización y privacidad que identifica al usuario por contraseñas, carteras digitales, tarjetas inteligentes y otras opciones de identificación avanzadas como huellas digitales, imágenes oculares, etc.
- b) **Notificaciones .NET Alerts.** Permite al usuario estar enterado de lo que le interesa y se integra con la mensajería instantánea Messenger y el correo Hotmail.

- c) **.NET My Services.** Servicios de suscripción para desarrollar aplicaciones centradas en usuario.

.NET myAddress - Dirección electrónica y geográfica de ubicación

.NET myProfile - Nombre, apodo, fotografía, fechas especiales

.NET myContacts - Libreta de direcciones

.NET myLocation - Ubicación geográfica para reuniones

.NET myNotifications - Administración personal de .NET Alerts

.NET myInbox - Correo de voz, internet y empresariales

.NET myCalendar - Administración del tiempo y tareas

.NET myDocuments - Almacenamiento de documentos

.NET myApplicationSettings - Configuración de aplicaciones

.NET myFavoriteWebSites - Sitios favoritos

.NET myWallet - Recibos, cupones, instrucciones de pago

.NET myDevices - Configuración de dispositivos electrónicos

.NET myServices - Servicios disponibles para un individuo

.NET myUsage - Reportes de uso de *.NET My Services*

Herramientas de desarrollo

- Ø **Los compiladores Microsoft para .NET** son distribuidos gratuitamente en el ambiente de ejecución .NET (*.NET Framework*). Son compiladores de línea de comandos, depurador en modo texto y gráficos y kit de desarrollo.
- Ø **Microsoft Visual Studio .NET** simplifica el desarrollo de aplicaciones .NET dramáticamente. Existen 6 ediciones distintas. Se explican más adelante.
- Ø La **suscripción MSDN** (Microsoft Developer Network) ofrece todo lo que el desarrollador necesita para crear software.

Incluyendo Visual Studio, Visual FoxPro, .NET Servers, Betas, Office, Project, Visio, Windows, etc.

Visual Studio .NET

Una de las piezas más importantes en el esquema de Microsoft .NET es el propio Visual Studio .NET. Éste es un entorno de trabajo tan distinto a lo hecho anteriormente por la empresa de las ventanas (o por cualquier otra) que Microsoft tuvo que darle otro nombre y prácticamente salirse de la secuencia de versiones de la plataforma de Visual Studio, aunque, al final, también lo mencionó como la séptima versión de la saga.

Para comprender a Visual Studio .NET hay que pensar como desarrollador y hacer una ligera historia de la tecnología de programación. Los primeros días de la programación se distinguieron por el código lineal, poco útil en aplicaciones grandes dado que arrojaba engendros difíciles de mantener. El código lineal evolucionó al código estructurado, que mejoró la disposición del código y que, por mucho tiempo, se estableció como el rey de la codificación.

Sin embargo, ya desde 1969, se avizoró otro esquema de programación que lograría una revolución real hasta mediados de la década de los años ochenta, con la presentación de C++, y más específicamente en la de los noventa con la proliferación de los entornos gráficos que le dieron mayor razón de ser.

La máxima de la programación orientada a objetos es la reutilización; es decir, aprovechar el código ya hecho para evitar la duplicación de trabajo. Sin embargo, una de las mayores limitantes de esto es, precisamente, la diversidad de lenguajes de programación: Un objeto generado en C++ no puede usarse directamente en lenguajes como Visual Basic, Object Pascal o SmallTalk. En realidad, un objeto generado en C++ sólo podrá usarse en C++, así como uno de Visual Basic, sólo podrá usarse en Visual Basic (claro está que podría compilar el objeto y hacerlo binario para, así, poder integrarlo en un entorno de programación, pero ello traería la necesidad de usar protocolos propietarios como COM, DCOM, COM+, CORBA, SOM, etcétera, lo cual rompe con ese sueño del reuso integral).

Con esto en mente, Microsoft quiso aprovechar la oportunidad de la presentación de Microsoft .NET para generar un marco de trabajo que fuera aprovechado por cualquier lenguaje de programación que se ciñera a sus estándares. Ese marco de trabajo es el .NET Framework, que es el meollo de la tecnología .NET. En pocas palabras, el .NET Framework es un cúmulo de clases expuestas para que, quien quiera, haga uso de su funcionalidad. A su vez, este cúmulo de clases conforma un estándar abierto que puede integrarse a cualquier plataforma o lenguaje.

Esta apertura permitió el diseño de un entorno de desarrollo tan amplio como lo es el Visual Studio .NET, que no sólo incluye los lenguajes de C# .NET, Visual Basic .NET, Visual C++ .NET y, próximamente, J# .NET, sino que hay más de 20 lenguajes de otros fabricantes que pueden funcionar en él, como Pascal .NET, Cobol .NET, y muchos otros.

El que tantos lenguajes distintos puedan funcionar en un mismo entorno, tiene un beneficio adicional: puede incluirse un objeto hecho en cualquiera de estos lenguajes en un proyecto generado en otro lenguaje. Por ejemplo, pueden incluirse clases generadas con C# .NET en un proyecto de Visual Basic .NET. Las clases de C# .NET no tendrán que compilarse para que esto sea posible, dado que el entorno interpretará adecuadamente las instrucciones que tenga para poder aprovechar su funcionalidad sin problemas.

De esta forma, Visual Studio .NET es lo más cercano a la máxima de la programación orientada a objetos: la reutilización. Claro está que todos los lenguajes tendrán que cumplir con las prerrogativas del .NET Framework, pero con la cantidad de lenguajes que están apareciendo para este entorno, esto podría ya no ser un problema.

Es importante mencionar que la tecnología Microsoft .NET es mucho más que lo poco que puede mencionarse en un espacio como el ocupado hasta ahora. Por ello mismo, puede hacer uso de los recursos que ponemos a su disposición en este medio (y en muchos otros) para ampliar su información al respecto. No obstante, espero haberle ayudado para comprender mejor el porqué de esta tecnología y empezar a aprovecharla aún mejor.

Visual Basic .NET

¿Pueden mis aplicaciones de Visual Basic 6.0 ser actualizadas a Visual Basic .NET?

Sí. Microsoft Visual Basic® .NET incluye un Asistente de actualización para que pueda tomar una máxima ventaja del nuevo lenguaje. El Asistente de actualización a Visual Basic .NET actualiza automáticamente proyectos de Visual Basic 6.0 cuando son abiertos en Visual Basic .NET. El Asistente de actualización los cambios de sintaxis en el lenguaje y convierte formas de Visual Basic 6.0 a Microsoft Windows® Forms. Además, el Asistente de actualización genera un reporte que le advierte de cualquier cambio manual que necesite hacer en su código. Los comentarios son visualizados como tareas en la nueva ventana Task List, y usted puede navegar al código simplemente haciendo doble clic a una tarea.

CARACTERISTICAS DE VISUAL BASIC.NET

Con Visual Basic .NET, Visual Basic sigue siendo la herramienta más productiva para la creación de aplicaciones que se ejecutan en el sistema operativo Microsoft Windows®. Con Visual Basic .NET, los programadores pueden seguir aprovechando sus conocimientos y capacidades para crear la próxima generación de aplicaciones y servicios Web XML.

Crear aplicaciones para Windows y el Web

Eficaces aplicaciones basadas en Windows

Con la herencia visual, los programadores pueden simplificar enormemente la creación de aplicaciones basadas en Windows, centralizando la interfaz de usuario y la lógica común de toda su solución en formularios primarios. Utilizando delimitadores y acoplamiento de controles, los programadores pueden generar formularios redimensionables automáticamente sin código, mientras el editor de menús in situ permite crear menús de manera visual directamente desde el Diseñador de Windows Forms.

Aplicaciones basadas en el Web

Cree soluciones Web en Visual Basic .NET utilizando el Diseñador de Web Forms y el Diseñador XML compartidos. Los programadores pueden utilizar la tecnología Microsoft IntelliSense® y la capacidad para completar etiquetas; o bien, elegir el editor WYSIWYG (lo que ve es lo que se imprime) para poder crear aplicaciones Web interactivas arrastrando y colocando elementos.

Aplicaciones móviles

Llegue hasta la gama más amplia de dispositivos compatibles con Internet utilizando las características de Mobile Internet de Visual Studio .NET. Estas nuevas características ofrecen a los programadores una única interfaz Web móvil para proporcionar compatibilidad con una amplia gama de dispositivos Web, incluidos WML 1.1 para teléfonos móviles WAP, HTML compacto (cHTML) para teléfonos i-mode y HTML para Pocket PC, dispositivos de mano y localizadores (paggers).

El camino mas rápido a programación en .Net

El centro de control más avanzado para los programadores

La página de inicio de Visual Basic .NET es un portal para programadores que permite tener acceso con un solo clic a información acerca de los proyectos usados recientemente, las preferencias personales, las actualizaciones de productos y la comunidad MSDN Online. El Cuadro de herramientas ampliado muestra una vista dinámica de los componentes del proyecto, como los controles de Windows Forms y Web Forms, los elementos HTML, los objetos y los miniprogramas.

Plantillas y asistentes que permiten ahorrar tiempo

La plantilla de servicios Web XML crea e implementa automáticamente los diversos componentes de un servicio Web. El Asistente para la instalación permite distribuir las aplicaciones .NET de forma sencilla.

Ayuda disponible al instante

La Ayuda dinámica proporciona acceso con un solo clic a la ayuda pertinente, independientemente de la tarea que se esté realizando. MSDN Online Access proporciona vínculos directos a ejemplos,

grupos de noticias, actualizaciones y descargas de Visual Basic .NET en el entorno de desarrollo integrado (IDE).

El lenguaje Visual Basic más reciente e intuitivo

El lenguaje más sencillo y popular

Desarrolle aplicaciones utilizando el lenguaje de programación más fácil de leer y de escribir que existe. La compilación en segundo plano proporciona información al instante y señala los errores con un subrayado ondulante.

Funciones de programación ampliadas

La implementación lado a lado acaba con los conflictos entre versiones y la herencia permite reutilizar el código de cualquier lenguaje basado en .NET. El Control de excepciones estructurado proporciona un código de control de errores más elegante y fácil de mantener.

La mayor comunidad de programadores

Incorpore recursos, componentes y código de más de 3 millones de programadores de Visual Basic de todo el mundo. Utilice componentes del gran mercado de proveedores de controles para crear completas aplicaciones basadas en .NET.

REQUERIMIENTOS DEL SISTEMA

Para utilizar Microsoft Visual Basic® .NET, usted necesita:

Requerimientos mínimos	
Procesador	Computadora personal (PC) con un procesador Pentium II, 450 megahertz (MHz)
Sistema Operativo	· Microsoft Windows® XP Professional
	· Microsoft Windows 2000 Professional

	<ul style="list-style-type: none"> · Microsoft Windows 2000 Server · Microsoft Windows NT® 4.0 Workstation · Microsoft Windows NT 4.0 Server
Memoria	<ul style="list-style-type: none"> · Microsoft Windows XP Professional 160 MB de RAM; se recomiendan 192 MB · Windows 2000 Professional 96 MB de RAM; se recomiendan 128 MB · Windows 2000 Server 192 MB de RAM; se recomiendan 256 MB · Windows NT 4.0 Workstation 64 MB de RAM; se recomiendan 96 MB · Windows NT 4.0 Server 160 MB de RAM: 160 MB de RAM; se recomiendan 192 MB
Disco Duro	500 MB en el disco de sistema, 2.0 gigabytes (GB) en el disco a instalarse
Unidad	CD-ROM u DVD-ROM
Video	Monitor Super VGA (800 x 600) o superior a 256 colores
Mouse	Microsoft Mouse o dispositivo compatible

PREGUNTAS MAS FRECUENTES

¿Cómo se llama la nueva versión de Visual Basic?

La versión de Visual Basic posterior a Visual Basic 6.0 es Visual Basic .NET.

¿Se puede usar Visual Basic 6.0 y Visual Basic .NET en el mismo equipo?

Sí, se puede instalar Visual Basic .NET en un equipo que esté ejecutando Visual Basic 6.0 y utilizar las dos versiones, siempre

que los requisitos del sistema lo permitan. Los entornos de tiempo de diseño y las aplicaciones creadas con Visual Basic 6.0 y Visual Basic .NET funcionarán juntas sin problemas en el mismo equipo.

¿Por qué se ha actualizado el lenguaje Visual Basic en Visual Basic .NET?

Visual Basic siempre ha estado a la vanguardia de la tecnología. Con Microsoft .NET Framework, se pretende garantizar que Visual Basic siga en primera línea. También se pretende proporcionar a los programadores de Visual Basic capacidades de programación, simplicidad y acceso a la plataforma sin precedentes. Estos son los objetivos de diseño que dieron lugar al lenguaje Visual Basic .NET.

Para obtener más información acerca de las mejoras del lenguaje Visual Basic .NET, consulte [Actualizar a Visual Basic .NET](#).

¿Qué es el Asistente para actualización de Visual Basic .NET?

El Asistente para actualización de Visual Basic .NET forma parte de Visual Basic .NET y Microsoft Visual Studio® .NET. Permite actualizar los productos de Visual Basic existentes a Visual Basic .NET para aprovechar todas las ventajas de .NET Framework.

¿Qué es .NET Framework?

.NET Framework es un nuevo modelo de programación diseñado para simplificar la programación de aplicaciones en un entorno sumamente distribuido: Internet. El software que se ejecuta en .NET Framework puede comunicarse con el software que se ejecuta en otro entorno a través de SOAP y puede utilizar objetos estándar localmente o distribuidos en Internet. En consecuencia, la experiencia del programador es siempre igual y le permite centrarse en las funciones en lugar de en la estructura.

¿Con los cambios en el lenguaje Visual Basic, ¿cómo se pueden conservar mejor las inversiones existentes en Visual Basic?

Microsoft garantiza la viabilidad de las inversiones existentes en Visual Basic. Se han implementado varias características y diseños en Visual Basic .NET que garantizan que las inversiones existentes seguirán funcionando. Por ejemplo:

- El Asistente para actualización de Visual Basic .NET, que actualizará el 95% del código existente en Visual Basic 6.0 a Visual Basic .NET.
- Una zona completa de interfuncionalidad de alto rendimiento, que permite la comunicación bidireccional sin problemas entre componentes de Visual Basic .NET y Visual Basic 6.0.
- La versión Beta 2 de Visual Studio .NET agrega coherencia semántica entre Visual Basic 6.0 y Visual Basic .NET. De esta manera, se garantiza que la mayor parte del código escrito en Visual Basic 6.0 tenga el mismo significado semántico en Visual Basic .NET.

¿Es necesario programar la actualización de todo el código de Visual Basic 6.0 a Visual Basic .NET?

En algunos casos, se puede decidir no actualizar el código de Visual Basic 6.0 a Visual Basic .NET. Cuando el código de Visual Basic 6.0 se implementa, se ejecuta o "está en mantenimiento", puede ser preferible llamar simplemente a ese código desde los nuevos componentes de Visual Basic .NET, utilizando la interfuncionalidad de COM.

¿Visual Basic .NET aún permite crear aplicaciones de escritorio completas?

Por supuesto. Aunque Microsoft .NET está diseñado para ser la plataforma de elección para crear y conectar servicios Web XML, también creemos que .NET es la mejor plataforma para crear aplicaciones de escritorio completas. Microsoft .NET se centra en Internet y, en un mundo basado en Internet, las aplicaciones de escritorio completas desempeñan un papel tan importante como las

aplicaciones y los servicios Web XML. Precisamente por eso, Microsoft creó Microsoft Windows® Forms. La tecnología Windows Forms permite a los programadores de Visual Basic crear aplicaciones de escritorio con funciones y capacidad de mantenimiento mejoradas, y con la capacidad de utilizar servicios Web XML. Visual Basic .NET es el camino más fácil para la programación de Windows Forms y permite a los programadores de Visual Basic utilizar sus capacidades para implementar la próxima generación del cliente inteligente.

¿Qué otros tipos de aplicaciones permite crear Visual Basic .NET?

Los programadores de Visual Basic tienen acceso a una gama mucho más amplia de aplicaciones en Visual Basic .NET. Con las mismas capacidades que se utilizan para programar aplicaciones basadas en Windows, ahora se pueden crear aplicaciones Web de largo alcance, aplicaciones dirigidas a dispositivos móviles y servicios Web XML. Asimismo, Visual Basic .NET permite crear visualmente componentes de lógica empresarial en el extremo del servidor utilizando el Explorador de servidores y el Diseñador de componentes visual. Por último, los programadores que utilizan Visual Basic .NET ahora pueden crear fácilmente servicios de Windows y aplicaciones de consola utilizando las nuevas plantillas de aplicaciones.

¿Cuáles son las diferencias entre Visual Basic .NET y Visual C# .NET?

Visual Basic .NET y Microsoft Visual C#™ .NET se han diseñado para las comunidades de Visual Basic y Microsoft Visual C++®, respectivamente. Visual Basic .NET se creó con la idea de otorgar a los programadores de Visual Basic unas capacidades y una productividad sin precedentes para crear servicios Web XML, aplicaciones de escritorio, aplicaciones Web y aplicaciones móviles. Visual C# surgió por el deseo de los programadores de la comunidad de C++ de una mayor productividad para crear las aplicaciones. Visual Basic .NET y Visual C# .NET contienen un conjunto de capacidades muy similar; las aplicaciones creadas en estos dos lenguajes funcionarán a la misma velocidad. Sin embargo, Visual Basic .NET y Visual C# .NET difieren considerablemente en la experiencia de los usuarios. Visual Basic .NET está dirigido a los programadores de Visual Basic, por eso incorpora muchos

de los conceptos y construcciones que son familiares para los programadores de Visual Basic, como código que no diferencia entre mayúsculas y minúsculas, un compilador en segundo plano y la opción de realizar vínculos en el último momento. Del mismo modo, Visual C# incorpora características que son familiares para los programadores de C++, como código no seguro.

¿Crearé Microsoft una actualización no administrada para el compilador de Visual Basic?

Microsoft continuará publicando Service Packs para Visual Basic 6.0 con el fin de corregir errores y aumentar la estabilidad global del producto.

¿Qué papel desempeña Visual InterDev en Visual Studio .NET?

Muchas de las ventajas de Microsoft Visual InterDev® se han incorporado a Visual Studio .NET, para permitir a los programadores de todos los ámbitos crear rápidamente aplicaciones Web. En Visual Studio .NET, se pueden crear aplicaciones Web utilizando un nuevo paquete de formularios denominado Web Forms. El paquete Web Forms permite crear aplicaciones Web con las mismas técnicas de arrastrar y colocar, y los formularios con código oculto que los programadores de Visual Basic han usado durante años para crear aplicaciones de escritorio completas.

¿Funcionan los controles ActiveX en el contexto de Visual Basic .NET?

Sí. Visual Basic .NET emplea un nuevo paquete para crear aplicaciones de escritorio denominado Windows Forms. El paquete Windows Forms se diseñó como un excelente contenedor para los controles Microsoft ActiveX® existentes, con el fin de proteger las inversiones actuales.

¿Quién utiliza Visual Basic .NET actualmente?

Cada vez hay más organizaciones de todos los tamaños que aprecian el conjunto de características mucho más amplio de Visual Basic .NET e implementan sus aplicaciones de software críticas utilizando Visual Basic .NET. Esto se hizo evidente incluso con la versión beta de Visual Basic .NET, con clientes como Verizon Wireless, Zagat Survey, DataReturn y muchos otros que crean servicios Web XML, aplicaciones Web para pocos clientes, aplicaciones de escritorio completas y aplicaciones móviles ágiles utilizando Visual Basic .NET.

¿Cómo puedo configurar el entorno para que utilice las combinaciones de teclas a las que estoy acostumbrado?

En la página de inicio de Visual Studio .NET, haga clic en Mi perfil. En Perfil, seleccione Visual Basic Developer.

¿Dónde están las versiones Professional y Enterprise de Visual Basic .NET y Microsoft Visual C++ .NET?

La funcionalidad que antes estaba disponible en las versiones Professional y Enterprise de los productos de lenguaje individuales ahora está disponible en las versiones Professional y Enterprise de Visual Studio .NET. Asimismo, está disponible un nuevo Visual Studio .NET Enterprise Architect con herramientas diseñadas específicamente para los arquitectos de software que crean software empresarial, como Microsoft Visio® Enterprise Architect.