
Guia de desarrollo AutoCAD .NET

Stephen Presto

Autodesk Developer Technical Services Team (DevTech)

Contenido

1	Introducción	3
1.1	Organización de la guía	3
1.2	Visión general de la API .NET de AutoCAD	3
1.3	Componentes de la API .NET de AutoCAD	4
1.3.1	Utilizar una DLL del API .NET de AutoCAD	4
1.3.2	Obtener las DLL de la API .NET de AutoCAD	5
1.4	Descripción general de Microsoft Visual Studio	6
1.4.1	Qué versión de Microsoft Visual Studio debo utilizar	6
1.4.2	Utilice la interoperabilidad COM con .NET	7
1.4.3	Dependencias y restricciones	8
1.5	Para más información	8
1.6	Código de muestra	8
1.7	Transición de ActiveX Automation a .NET	9
2	Introducción a Microsoft Visual Studio	10
2.1	Entender Microsoft Visual Studio Projects	10
2.2	Definir los componentes de un proyecto	11
2.2.1	Módulos de clase	11
2.2.2	formas	11
2.2.3	Referencias	11
2.3	Ver información sobre el proyecto	11
2.4	Trabajar con Microsoft Visual Studio Projects	12
2.4.1	Crear proyecto nuevo	12
2.4.2	Abrir un proyecto existente o una solución	13
2.4.3	Guardar un proyecto o una solución	14
2.4.4	Trabajar con varios proyectos en una solución	14
2.4.5	Agregar un proyecto a una solución	14
2.5	Descarga de un proyecto de una solución	15
2.6	Editar un proyecto existente o una solución	15
2.6.1	Agregar nuevos elementos	15
2.6.2	Importación de elementos existentes	16
2.7	Editar elementos	16
2.7.1	Utilice la ventana de código	17
2.7.2	Use el Diseñador de Windows Forms	17
2.7.3	Utilice la ventana Propiedades	18
2.7.4	Cambiar el nombre de un proyecto	18
2.7.5	Establezca las opciones para Microsoft Visual Studio	20
2.8	Carga un ensamblado en AutoCAD	20

2.9	Acceder y buscar en bibliotecas de referencia (Explorador de objetos)	21
2.10	Ejercicios: Crear su primer proyecto	22
2.10.1	Ejercicio: Crear un proyecto nuevo	22
2.10.2	Ejercicio:Referencia de los archivos API .NET de AutoCAD	22
2.10.3	Ejercicio: Crear un comando nuevo	23
2.10.4	Ejercicio: Crear y cargar una ensamblado .NET en AutoCAD.	26
2.11	Comandos relacionados con AutoCAD y Terminología	26
2.11.1	Comandos AutoCAD	26
2.11.2	Terminología	26
2.12	Más Información	27

1

Introducción

En esta introducción se describen los conceptos de la interfaz de programación de aplicaciones (API) AutoCAD® administradas con el Framework .NET. La API .NET AutoCAD® le permite automatizar tareas como la creación y modificación de objetos almacenados en la base de datos de un archivo de dibujo o cambiar el contenido de un archivo de personalización. En esta guía se explica el uso de Microsoft® Visual Studio® 2010, y los lenguajes de programación Microsoft® Visual Basic® .NET (denominada en esta guía como VB .NET) y Microsoft Visual C#® con API .NET AutoCAD®.

Temas de esta sección

- Organización de la guía
- Visión general de la API .NET de AutoCAD
- Componentes de la API .NET de AutoCAD
- Descripción general de Microsoft® Visual Studio®
- Para obtener más información
- Ejemplo de código
- Transición de ActiveX Automation a .NET

1.1 Organización de la guía

Esta guía proporciona información sobre cómo utilizar la API .NET de AutoCAD con Microsoft Visual Studio y los lenguajes de programación VB.NET y C#. Información específica para el desarrollo de aplicaciones con Microsoft Visual Studio se pueden encontrar en los temas "Introducción a Microsoft Visual Studio" y "Desarrollo de aplicaciones con Microsoft Visual Studio".

Los programadores familiarizados con el Framework .NET utilizando un entorno de desarrollo de Microsoft Visual Studio pueden omitir estos dos capítulos. Sin embargo, todo el código de ejemplo en esta guía se presenta en VB .NET y C#.

1.2 Visión general de la API .NET de AutoCAD

La API .NET de AutoCAD le permite manipular archivos de dibujo de AutoCAD con los ensamblados o bibliotecas que están disponibles. A estos objetos se pueden acceder mediante muchos lenguajes de programación y entornos diferentes.

Hay varias ventajas de la implementación de un API .NET para AutoCAD:

- El acceso mediante programación a los dibujos de AutoCAD se abre a más entornos de programación. Antes de la API .NET, los desarrolladores se limitaban a ActiveX Automation® y los idiomas que apoyaron COM, AutoLISP® y con ObjectARX C++.
- La integración con otras aplicaciones basadas en Windows® como Microsoft Excel y Word, se hace más fácil mediante el uso de una aplicación nativa API .NET o enlazados a una biblioteca ActiveX/COM.
- El Framework .NET está diseñado para sistemas operativos de 32-bit y 64-bit. Visual Basic para Aplicaciones ha sido diseñado sólo para sistemas operativos de 32 bits.
- Permite el acceso a las interfaces avanzadas de programación con una curva de aprendizaje más baja comparada a los lenguajes de programación más tradicionales, tales como C++.

Los objetos son los bloques de construcción principales de la API .NET de AutoCAD. Cada objeto representa una parte de AutoCAD, y se agrupan en diferentes ensamblados y espacios de nombres. Hay muchos tipos diferentes de objetos de la API .NET de AutoCAD. Por ejemplo:

- Los objetos gráficos como líneas, arcos, texto y cotas
- Los ajustes de estilo tales como estilos de texto y dimensión
- Estructuras organizacionales, tales como capas, grupos y bloques
- Las presentaciones de dibujo como la vista y la vista preliminar
- El dibujado y aplicaciones AutoCAD

Para obtener información sobre algunos de los archivos que componen el API .NET de AutoCAD, consulte Componentes de la API .NET de AutoCAD.

1.3 Componentes de la API .NET de AutoCAD

La API .NET de AutoCAD se compone de archivos DLL diferentes que contienen una amplia gama de clases, estructuras, métodos, y eventos que proporcionan acceso a los objetos en un archivo de dibujo o una aplicación AutoCAD. Cada archivo DLL define espacios de nombres diferentes que se utilizan para organizar los componentes de las bibliotecas basadas en su funcionalidad.

Las tres principales archivos DLL de la API .NET de AutoCAD que con frecuencia se utilizan son:

- *AcDbMgd.dll* Se utiliza cuando se trabaja con objetos en un archivo de dibujo.
- *AcMgd.dll* Se utiliza cuando se trabaja con la aplicación AutoCAD.
- *AcCui.dll* Se utiliza cuando se trabaja con archivos de personalización.
- *AcCoreMgd.dll* Necesario en AutoCAD 2013

1.3.1 Utilizar una DLL del API .NET de AutoCAD

Antes de clases, estructuras, métodos y eventos que se encuentran relacionados en con una DLL de la API .NET de AutoCAD para ser utilizado debe hacer referencia al archivo DLL en un proyecto. Después que una DLL se referencia a un proyecto, puede utilizar los espacios de nombres y los componentes en el archivo DLL en su proyecto.

Una vez que una DLL de la API .NET de AutoCAD esta referenciada, debe establecer la propiedad Copia Local de la DLL en False. La propiedad Copia Local determina si Microsoft Visual Studio creara una copia del archivo DLL referenciado y lo coloca en el mismo directorio que el archivo de ensamblado del proyecto cuando se construye. Dado que los archivos de referencia ya vienen con AutoCAD, la creación de copias de los archivos DLL de referencia puede producir resultados inesperados cuando se carga un archivo de ensamblado en AutoCAD.

1.3.2 Obtener las DLL de la API .NET de AutoCAD

El ObjectARX SDK para AutoCAD 2013 se puede descargar desde el sitio <http://www.objectarx.com>. Después de que el ObjectARX SDK está instalado, los archivos DLL se pueden encontrar en el C:\ObjectARX 2013\inc.

Procedimiento

Para descargar e instalar el ObjectARX SDK de AutoCAD 2013

1. Inicie su aplicación de navegador de Internet predeterminado y vaya a <http://www.objectarx.com>.
2. En la página Web, haga clic en *Licencia & Descargar*.
3. Rellene los campos necesarios y seleccione ObjectARX para AutoCAD 2013. Haga clic en *Enviar*.
4. En la página de descarga, haga clic en *Descargar ahora* para utilizar el administrador de descargas o haga clic en *Descargar Método estándar* que se utilizará el *método de descarga por defecto* de su navegador de Internet.
5. Haga clic en *Guardar* o en la opción utilizada para guardar el archivo en el disco local.
6. Especifique una ubicación para descargar el archivo ObjectARX SDK.
7. Una vez que el paquete se ha descargado el archivo, busque la ubicación en la que lo guardó y haga doble clic en él.
El asistente de instalación aparecerá.
8. En el cuadro de diálogo ObjectARX <release>, especifique una nueva ubicación de instalación o salir de la ubicación de instalación predeterminada. Haga clic en *Instalar*.
El asistente de instalación se cierra después de que se termine si no ha habido problemas.

Para instalar el Asistente ObjectARX y las plantillas de proyecto Managed

1. Una vez que el SDK ObjectARX está instalado, vaya a su carpeta de instalación que por defecto es c: ObjectARX <release>.
2. Después de navegar a la carpeta de instalación, abra la carpeta Utilidades y luego la carpeta ObjARXWiz.
3. En la carpeta ObjARXWiz, haga doble clic ArxWizards.msi.
4. En los Wizards ObjectARX para la caja de diálogo de AutoCAD 2010, haga clic en *Siguiente*.
5. En el cuadro Escriba su defecto preferidos cuadro Símbolo RDS, introduzca una abreviatura del nombre de su empresa y haga clic en *Siguiente*.
El texto introducido por el RDS se utiliza como prefijo a los nombres de la clase predeterminada o comandos creados con el Asistente ObjectARX y plantillas de proyecto.
Nota RDS representa Símbolo desarrollador registrado y se utiliza para crear comandos unquely nombre y clases para ayudar a evitar posibles conflictos con otras aplicaciones.
6. En la página Asistente para la variable miembro, deje Reemplace Asistente MS variables miembro por Wizard Autodesk variable miembro seleccionado y haga clic en *Siguiente*.

7. En la página Seleccionar carpeta de instalación, haga clic en Examinar para especificar una nueva ubicación de instalación para el asistente o salir de la ubicación predeterminada. Haga clic en Siguiente.
8. Haga clic en Siguiente para instalar el asistente.
9. Haga clic en Cerrar para cerrar el instalador.

Para hacer referencia a un archivo DLL de AutoCAD. NET API

1. En Microsoft Visual Studio, haga clic en el menú Ver> - Explorador de soluciones para mostrar el Explorador de soluciones, si no se ha mostrado aún.
2. En el Explorador de soluciones, en la barra de herramientas en la parte superior, haga clic en Mostrar todos los archivos.
3. Haga clic en el nodo Referencias y seleccione Agregar referencia.
4. En el cuadro de diálogo Agregar referencia, Examinar, seleccione el archivo DLL que contiene la biblioteca que desea utilizar y haga clic en Aceptar.
5. En el Explorador de soluciones, haga clic en el signo más a la izquierda del nodo Referencias para expandirlo.
6. Seleccione la biblioteca de referencia desde el nodo Referencias.
7. Haga clic derecho sobre la referencia seleccionada y haga clic en Propiedades.
8. En la ventana Propiedades, haga clic en el campo Copy Local y seleccione False en la lista desplegable.

1.4 Descripción general de Microsoft Visual Studio

Microsoft Visual Studio es un entorno de programación orientado a objetos que se ejecuta independientemente de AutoCAD. Mientras Microsoft Visual Studio es externo a AutoCAD y otras aplicaciones, es capaz de interactuar con aplicaciones nativas API .NET o bibliotecas ActiveX/COM.

Temas de esta sección

- Qué edición de Microsoft Visual Studio debo utilizar
- Utilizar la interoperabilidad COM con. NET
- Dependencias y restricciones

1.4.1 Qué versión de Microsoft Visual Studio debo utilizar

Microsoft Visual Studio está disponible en varias versiones y ediciones. Para utilizar la API .NET de AutoCAD 2013 es necesario utilizar:

- Microsoft Visual Studio 2010 con Service Pack 1
- Microsoft. NET Framework 4.0 o superior

Microsoft Visual Studio se ofrece en dos versiones: de forma gratuita y de pago. La edición gratuita se conoce como Microsoft Visual Studio 2010 Express Edition, mientras que el de las ediciones de pago varían según el nombre y el precio debido a las herramientas de desarrollo diferentes que se incorporan a ellos. Microsoft Visual Studio 2010 Standard Edition es la edición de nivel de entrada que ofrece la depuración mejorada en Microsoft Visual Studio

2010 Express Edition junto con un número de otras características. La edición más común de Microsoft Visual Studio utilizada por desarrolladores es Microsoft Visual Studio 2010 Professional Edition.

Sin embargo es posible usar Microsoft Visual Studio Express con la API .NET de AutoCAD, esta guía asume que usted está utilizando una de las versiones, como Microsoft Visual Studio 2010 Standard Edition o Microsoft Visual Studio 2010 Professional Edition.

Hay cuatro ventajas principales al utilizar Microsoft Visual Studio:

- Robusto y entorno de desarrollo accesible que tiene una curva de aprendizaje modesta.
- La sintaxis de VBA y VB.NET son similares, lo que lo convierte en un entorno ideal para actuales usuarios de VBA
- Visualmente intuitiva y con una amplia gama de herramientas en el cuadro de diálogo de creación.
- Los proyectos se pueden construir como un ejecutable independiente o un conjunto de DLL's que pueden ser cargados en AutoCAD para su ejecución.

Los proyectos de VBA,.NET no sufren la caída del rendimiento cuando se carga y se ejecuta en AutoCAD 64-bit.

Para obtener más información sobre las diferentes ediciones de Microsoft Visual Studio, vea <http://www.microsoft.com/vstudio> y <http://www.microsoft.com/express>.

1.4.2 Utilice la interoperabilidad COM con .NET

Microsoft Visual Studio pueden utilizar tanto nativos .NET y COM en el mismo proyecto. Al utilizar la interoperabilidad COM, puede migrar el código existente que podría haber sido escrito en Visual Basic 6 o VBA sin tener que reescribir por completo. Para acceder a los objetos de AutoCAD automatización de un proyecto creado en Microsoft Visual Studio, cree referencias a los siguientes archivos:

- La biblioteca de tipos AutoCAD 2010, `acax18enu.tlb`, ubicado en `C:\Archivos de programa\Archivos comunes\Autodesk Shared`.
- La AutoCAD\ObjectDBX Común 18,0 tipo biblioteca, `axdb18enu.tlb`, ubicado en `C:\Archivos de programa\Archivos comunes\Autodesk Shared`.

Nota: Los anteriores mencionadas bibliotecas de tipos también están disponibles como parte del SDK ObjectARX. Para obtener información sobre cómo descargar e instalar el SDK ObjectARX, consulte [Componentes de la API de AutoCAD .NET](#).

Estas referencias se hacen los siguientes ensamblados de interoperabilidad primarios disponibles:

- `Autodesk.AutoCAD.Interop.dll` (para AutoCAD específicos de tipos)
- `Autodesk.AutoCAD.Interop.Common.dll` (en los modelos compartidos por las aplicaciones anfitrionas ObjectDBX™)

Los ensamblados de interoperabilidad se encuentran en la caché de ensamblados global, se asignan a objetos de automatización de su red de contrapartes.

Después de hacer referencia a las bibliotecas de tipos, puede declarar variables basadas en AutoCAD en Microsoft Visual Studio, como en los siguientes ejemplos:

VB.NET

```
Dim objAcApp As Autodesk.AutoCAD.Interop.AcadApplication
Dim objLine As Autodesk.AutoCAD.Interop.Common.AcadLine
```

C#

```
Autodesk.AutoCAD.Interop.AcadApplication.objAcApp;  
Autodesk.AutoCAD.Interop.Common.AcadLine.objLine;
```

Utilizando los ensamblados de interoperabilidad pueden hacer la transición de los sus proyectos de VBA a mucho más fácil VB.NET. Sin embargo, con el fin de sacar el máximo provecho de todo eso. NET y el AutoCAD. API NET tienen que ofrecer, usted tendrá que reescribir su código existente de VBA.

Crear y hacer referencia a la aplicación AutoCAD.

Aplicaciones de AutoCAD 2010. NET pueden utilizar la misma biblioteca de tipos (acax18enu.tlb) como proyectos de automatización de AutoCAD. La biblioteca de tipos se encuentra en *C:\Archivos de programa\Archivos comunes\Autodesk Shared*.

Aplicaciones de AutoCAD 2010. NET también utiliza la versión dependiente del mismo ProgID para el CreateObject, GetObject y funciones GetInterfaceObject. Por ejemplo, CreateObject ("AutoCAD.Application.18") le permite crear una instancia de AutoCAD y obtener un objeto que representa la nueva instancia de la aplicación.

1.4.3 Dependencias y restricciones

A diferencia de ActiveX Automation, hay menos conflictos con bibliotecas cuando otras aplicaciones se instalan, vuelven a instalar o desinstalar. Los problemas de compatibilidad son menores en el Framework .NET debido a que es una plataforma estandarizada. Sin embargo, todavía pueden encontrarse con problemas de dependencia. Para evitar problemas de dependencia con el Framework .NET, asegúrese de usar la misma o una versión anterior del Framework .NET en un proyecto con C# o VB.NET para su uso en AutoCAD 2013. Para obtener información sobre la versión del Framework .NET, véase [Qué versión de Microsoft Visual Studio debo utilizar](#).

1.5 Para más información

Esta guía asume que posee los conocimientos necesarios en lenguajes de programación VB.NET o C#, y no trata de duplicar o sustituir la abundante documentación disponible en cualquiera de estos lenguajes de programación. Si necesita más información acerca de VB.NET o C#, y el uso de Microsoft Visual Studio, vea el sistema de Ayuda de Microsoft Visual Studio de Microsoft, disponible en el menú Ayuda de Microsoft Visual Studio.

1.6 Código de muestra

Esta guía y el ObjectARX SDK AutoCAD 2013 en conjunto contienen un gran número de proyectos de muestra, subrutinas y funciones que muestran el uso de las clases, estructuras, métodos, propiedades y eventos que conforman la API .NET AutoCAD.

También se pueden encontrar proyectos de ejemplo que muestran algunos de los aspectos de la API .NET de AutoCAD en la página web de Autodesk en <http://www.autodesk.com/developautocad>. Estos proyectos de ejemplo muestran una amplia gama de funcionalidades, desde la extracción de datos de dibujo de AutoCAD en hojas de cálculo Microsoft Excel para la elaboración y la realización de análisis de tensión en una torre de transmisión eléctrica. Muchas de estas muestras indican cómo combinar diversos aspectos de VB.NET y C# mediante AutoCAD para crear aplicaciones personalizadas.

Además, el código de ejemplo en la Guía de AutoCAD desarrollo en .NET puede ser copia de los archivos de Ayuda, pegado directamente en una ventana del editor de código abierto en Microsoft Visual Studio y, a continuación, construido y cargado en AutoCAD. En la parte superior de la mayoría de las muestras de código en esta guía son los espacios de nombres que se requieren para esa muestra particular.

Añadir lo que se necesite al principio de la ventana código.

Nota: El código de ejemplo en los archivos de Ayuda contiene un control de errores limitado para mantener los conceptos simples y fáciles de leer. Usted debe adicionar el manejo y el control de errores cuando se utiliza un código de ejemplo en su proyecto. Para obtener más información sobre el tratamiento de errores, vea [Controlar errores](#).

Procedimiento

Para ejecutar el código de ejemplo de los archivos de Ayuda

1. En Microsoft Visual Studio, abra una ventana del editor de código, agregue los espacios de nombres propios y defina una clase si no existe ya.
2. Copie el código de ejemplo desde el archivo de ayuda y pegue el código copiado en la clase definida.
3. En Microsoft Visual Studio, haga clic en Generar > Generar <Proyecto>.
4. En AutoCAD, en la ventana de comandos, escriba *NetLoad* y pulse Intro.
5. En la caja de diálogo elija un ensamblado .NET, seleccione el archivo ensamblado construido.
6. Haga clic en *Abrir*.
7. En la ventana de comandos, escriba el nombre del comando o función AutoLISP y los parámetros necesarios definidos en el ensamblado cargado.

1.7 Transición de ActiveX Automation a .NET

AutoCAD 2013 continúa apoyando a VBA, los componentes VBA pueden descargarse de <http://www.autodesk.com/vba-download>. Le recomendamos que comience la migración de los actuales proyectos de VBA para asegurarse de que este listo cuando el soporte de VBA no exista. Le recomendamos desarrollar nuevas aplicaciones con Microsoft Visual Studio y la API .NET de AutoCAD, AutoLISP, C++ y ObjectARX.

Puede seguir utilizando la biblioteca de automatización COM en .NET para ayudar a hacer la transición de VBA a VB.NET más fácil, consulte [Uso de interoperabilidad COM con .NET](#). Si usted es nuevo en .NET, vea [Introducción a Microsoft Visual Studio](#) para obtener información básica sobre cómo trabajar con Microsoft Visual Studio y la API .NET AutoCAD.

2

Introducción a Microsoft Visual Studio

Este capítulo es una introducción al AutoCAD® .NET API y algunos de los conceptos básicos de Microsoft® Visual Studio® entorno de desarrollo. Junto con una introducción al entorno de desarrollo, usted también aprenderá sobre el comando NETLOAD en AutoCAD y algunos de la terminología utilizada en esta guía.

Temas de esta sección

- Entender Microsoft Visual Studio Projects
- Definir los componentes de un proyecto
- Ver información sobre el proyecto
- Trabajar con Microsoft Visual Studio Projects
- Editar un proyecto existente o una solución
- Cargue un ensamblado en AutoCAD
- Acceso y búsqueda referenciados Bibliotecas (Explorador de objetos)
- Ejercicios: Crear su primer proyecto
- Comandos relacionados con AutoCAD y Terminología
- Más información

2.1 Entender Microsoft Visual Studio Projects

Los archivos de proyecto creados con Microsoft Visual Studio no son específicas de AutoCAD, pero no contienen la configuración específica del proyecto que tendrá que estar familiarizado con el fin de crear un archivo de ensamblado DLL que se puede cargar en AutoCAD. Esta guía trata sobre la creación de proyectos de Visual Basic® (VB). NET y Visual C#® con y sin el Asistente de ObjectARX de AutoCAD 2013.

Un proyecto es un conjunto de código y archivos de recursos, módulos de clase, ventanas WPF y Windows Forms que trabajan juntas para realizar una función determinada. Cuando se crea un proyecto, un también se crea una solución que contiene el proyecto que está creando. Una solución se utiliza para hacer referencia a uno o más proyectos. Por lo general, usted no tiene que trabajar con una solución directamente a menos que desee hacer referencia a la funcionalidad expuesta de otros proyectos.

Un ejemplo de una solución de varios proyectos es cuando usted tiene un nuevo proyecto y un proyecto existente que contiene un conjunto de métodos, funciones y clases que desea utilizar con un nuevo proyecto.

Una solución puede contener cualquier combinación de proyectos de VB .NET y C#. Utilizando diferentes tipos de proyectos en una única solución permite que cada desarrollador utilice el lenguaje de programación de su elección. Los archivos de proyecto y solución tienen las siguientes extensiones de archivo:

- *VBPROJ* - Microsoft Visual Basic (VB) .NET project file
- *CSPROJ* - Microsoft Visual C# project file
- *SLN* - Microsoft Visual Studio solution file

2.2 Definir los componentes de un proyecto

Cada proyecto puede contener muchos componentes diferentes. Los diferentes componentes de un proyecto puede contener módulos de clase, formularios, referencias y recursos.

2.2.1 Módulos de clase

Los componentes que contienen procedimientos públicos y privados y funciones. Los módulos de clase se utilizan para definir espacios de nombres personalizados. Dentro de un espacio de nombres, se definen los procedimientos utilizados para su programa y definir las estructuras de implementadas comandos personalizados y funciones de AutoLISP.

2.2.2 formas

Componentes forman contienen cuadros de diálogo personalizados que jalonan para su uso con su proyecto. Formas en el proyecto se muestran mediante un procedimiento o función, a menos que tenga una aplicación independiente. Windows Forms, WPF ventanas y los controles de usuario son algunos de los tipos de formularios que pueden ser parte de un proyecto. Para esta guía, las formas pueden significar Windows Forms y WPF ventanas.

2.2.3 Referencias

Las referencias se utilizan para indicar qué proyectos o bibliotecas de su proyecto utiliza.

2.3 Ver información sobre el proyecto

El Explorador de soluciones, que muestra la solución actual y una lista de todos los proyectos cargados, también muestra los archivos del código, clase y forma de módulos incluidos en cada proyecto cargado, y las referencias a las bibliotecas externas utilizadas por cada proyecto. El Explorador de soluciones tiene su propia barra de herramientas, que se puede utilizar para abrir diversos componentes del proyecto para su edición. Utilice el botón Ver código para abrir el código de un módulo seleccionado en la ventana del editor. Utilice el botón Diseñador de vistas para mostrar el Diseñador Visual para una forma seleccionada o el diagrama de clases para trabajar con Vista de clases.

TRIAL RESTRICTION

El Explorador de soluciones es visible por defecto. Si no está visible, haga clic en el menú Vista Explorador de soluciones o pulse Ctrl + Alt + L.

2.4 Trabajar con Microsoft Visual Studio Projects

Una solución abierta o archivo de proyecto se pueden ver y acceder desde el Explorador de soluciones. El Explorador de soluciones le permite añadir nuevos elementos y proyectos a la solución actual y los proyectos, descargar un proyecto de la solución actual, cambiar las propiedades de una solución abierta o proyecto, y agregar referencias a un proyecto de la solución actual, entre muchas opciones. Todas las tareas disponibles para trabajar con la solución actual o proyecto se puede acceder a través del menú contextual del Explorador de soluciones. Para obtener información sobre el Explorador de soluciones, consulte Ver [Información del proyecto](#).

Nota: Después crear un proyecto nuevo, asegúrese de que el Service Pack 1 para Microsoft Visual Studio 2010 está instalado.

Procedimiento

Para iniciar Microsoft Visual Studio • En el menú Inicio, haga clic en *Todos los programas \ Microsoft Visual Studio 2010 \ Microsoft Visual Studio 2010*.

Temas de esta sección

- Crear un proyecto nuevo
- Abrir un proyecto existente o una solución
- Guardar un proyecto o una solución
- Trabajar con varios proyectos en una solución

2.4.1 Crear proyecto nuevo

Los nuevos proyectos se crean sobre la base de una plantilla de proyecto. Al crear un nuevo proyecto que se construirá en un conjunto de DLL que se cargan en AutoCAD, utilice la plantilla de biblioteca de clases que viene con Microsoft Visual Studio o una de las plantillas de proyecto de AutoCAD administrados de aplicaciones que se instalan con el Asistente ObjectARX de AutoCAD 2013. Ambos tipos de plantillas disponibles para la VB .NET y C#.

Después de un proyecto se crea una nueva, tendrá que hacer referencia a los archivos que componen el AutoCAD. NET API. Para obtener información acerca referencia a los archivos asociados con el AutoCAD. API NET e instalar el Asistente ObjectARX de AutoCAD 2013, consulte Componentes de la API .NET de AutoCAD.

Procedimiento

Para Crear un nuevo proyecto utilizando una plantilla estándar

1. En Microsoft Visual Studio, haga clic en el menú Archivo \ Nuevo \ Proyecto.
 - (a) En el cuadro de diálogo Nuevo proyecto, Visual Basic o Visual C# y haga clic en Windows.
2. En Plantillas, seleccione Biblioteca de clases.
 - (a) En el cuadro Nombre, escriba un nombre para el nuevo proyecto.
 - i. En el cuadro Ubicación, escriba una ubicación o haga clic en Examinar para seleccionar una carpeta para el nuevo proyecto.
 - ii. En el cuadro Nombre de la solución, escriba un nombre para la nueva solución que el proyecto se va a sumar.
 - iii. Opcionalmente, active la casilla Crear directorio para la solución de verificación para crear una subcarpeta antes de crear la nueva solución y de proyecto. Seleccione la casilla de verificación Agregar a control de código fuente para agregar la solución y proyecto para una base de datos de control de código fuente.

- (b) Haga clic en Aceptar.

Para crear un proyecto nuevo utilizando una plantilla de AutoCAD Managed

Antes de utilizar una de las plantillas de AutoCAD administrados, primero debe descargar e instalar el SDK ObjectARX de AutoCAD 2013. Para obtener información sobre la instalación del AutoCAD Gestionado plantillas de proyecto, consulte Componentes de la API de AutoCAD. NET.

1. En Microsoft Visual Studio, haga clic \ Archivo \ Nuevo proyecto.
2. En el cuadro de diálogo Nuevo proyecto, Visual Basic o Visual C#.
3. En Plantillas, seleccione AutoCAD Gestionado VB proyecto de aplicación o aplicación AutoCAD Managed C# Project.

TRIAL RESTRICTION

1. En el cuadro Nombre, escriba un nombre para el nuevo proyecto.
 - (a) En el cuadro Ubicación, escriba una ubicación o haga clic en Examinar para seleccionar una carpeta para el nuevo proyecto.
 - (b) En el cuadro Nombre de la solución, escriba un nombre para la nueva solución que el proyecto se va a sumar.
 - (c) Opcionalmente, seleccione Crear directorio para la solución de crear una subcarpeta antes de crear la nueva solución y de proyecto. Seleccione Agregar para control de código fuente para agregar la solución y el proyecto de una base de datos de control de código fuente.
2. Haga clic en Aceptar.
 - (a) En el AutoCAD Gestionado <idioma> Application Wizard - caja de diálogo <Nombre proyecto>, vaya a la caja de registro del símbolo Su desarrollador, e introduzca una cadena de hasta cuatro caracteres.

El símbolo introducido se usa como prefijo para la clase predeterminada y mandamos que se añada al módulo de código inicial, y para evitar posibles conflictos de nombres con otras bibliotecas de referencia.

TRIAL RESTRICTION

1. Opcionalmente, si usted va a trabajar con las bibliotecas que utilizan código no administrado, seleccione Habilitar depuración no administrada.
2. Opcionalmente, seleccione cualquiera de las extensiones adicionales netos. Usted puede ser que necesite para su proyecto.
3. Haz clic en Terminar.

2.4.2 Abrir un proyecto existente o una solución

Al abrir un proyecto o solución en Microsoft Visual Studio, el editor de código y las ventanas de Windows Form Designer se abren en el mismo estado en que estaban cuando el proyecto se guardó por última vez. Una vez que un proyecto o solución que se abre, utilice el Explorador de soluciones para navegar entre los archivos del proyecto abierto o solución. Para obtener información sobre cómo trabajar con un proyecto abierto o solución ver, editar un proyecto existente o soluciones.

Procedimiento

Para abrir un proyecto existente o una solución de archivo

1. En Microsoft Visual Studio, haga clic en el menú Archivo-Open>Proyecto/ Solución.

2. En el cuadro de diálogo Abrir, busque y seleccione el archivo de proyecto para abrir.

TRIAL RESTRICTION

Aparecerá el cuadro de diálogo Abrir proyecto le permitirá abrir una amplia gama de archivos de proyectos que incluyen VB .NET y C#, proyectos de Microsoft Visual Studio. Net Solutions, Microsoft Visual C++, e incluso antiguos proyectos de Microsoft Visual Basic 6. Utilice los objetos de tipo lista desplegable para controlar qué tipo de proyectos o soluciones que se ven en el área de selección de archivo del cuadro de diálogo Abrir.

Nota No se puede abrir un proyecto de VBA guarda como un archivo DVB con Microsoft Visual Studio. Si va a migrar de VBA a VB .NET o C#, se puede copiar el código desde el entorno de AutoCAD VBA desarrollo a una ventana del editor de código abierto en Microsoft Visual Studio y haga los cambios necesarios.

3. Haga clic en Abrir.

2.4.3 Guardar un proyecto o una solución

Microsoft Visual Studio es un entorno de desarrollo basado contextual, lo que significa que el elemento actual seleccionado determina las funciones disponibles en el entorno de desarrollo. Controlar que los archivos modificados ahorrar o cómo desea guardarlos está determinado por el Explorador de soluciones.

Una vez que un proyecto o solución se selecciona desde el Explorador de soluciones, puede guardarlo con su nombre actual, o realizar un Guardar como en él para crear una copia del archivo. Puede seleccionar uno o varios elementos a la vez en el Explorador de soluciones pulsando y manteniendo pulsada la tecla Ctrl antes de seleccionar un elemento. Muy a menudo, usted se estará ahorrando todos los elementos modificados en el momento mismo.

Procedimiento

Para guardar un proyecto o solución

1. En Microsoft Visual Studio, en el Explorador de soluciones, seleccione el proyecto o la solución para salvar.
2. Haga clic en el menú Archivo \ Guardar <nombre proyecto> o solución o Guardar <nombre de proyecto> o solución como.

Si selecciona Guardar \ <nombre proyecto> Como solución, el cuadro Guardar como diálogo.

1. Si el Guardar como cuadro de diálogo, busque una ubicación para crear una nueva copia del proyecto o la solución e introduzca un nuevo nombre. Haga clic en Guardar

Para guardar todos los elementos modificados

En Microsoft Visual Studio, haga clic en el menú Archivo \ Guardar todo.

2.4.4 Trabajar con varios proyectos en una solución

Una solución puede contener varios proyectos. Trabajar con varios proyectos no es muy diferente a trabajar con un solo proyecto. Utilice el Explorador de soluciones para navegar entre los proyectos de la solución actual.

2.4.5 Agregar un proyecto a una solución

Usted puede agregar un proyecto a una solución para copiar el código entre proyectos. Es posible que desee hacer referencia a los procedimientos, funciones y clases en un proyecto y utilizarlos en otro proyecto. Mediante la adición de varios proyectos a una solución única, que le permite utilizar un proyecto como un conjunto de utilidades comunes que se pueden utilizar con más de un proyecto.

2.5 Descarga de un proyecto de una solución

Los proyectos pueden ser descargados de una solución cuando ya no son necesarios. Si ya no desea un proyecto para cargar con una solución, puede descargar el proyecto y luego hacer referencia al ensamblado DLL compilada del proyecto en su lugar. Trabajar con el archivo DLL compilado ayuda a prevenir accidental ediciones en el código fuente.

Procedimiento

Para agregar un proyecto a una solución

- En Microsoft Visual Studio, realice una de las siguientes acciones:

Haga clic o el menú Archivo \ Agregar \ Nuevo proyecto para crear un proyecto nuevo y agregarlo a la solución actual. Para obtener información sobre cómo crear un nuevo proyecto, consulte Crear un nuevo proyecto.

Haga clic en el menú Archivo \ Agregar \ Proyecto existente para que aparezca el cuadro de diálogo Agregar proyecto y agregar un proyecto existente a la solución actual. En el cuadro de diálogo Agregar proyecto, busque y seleccione el proyecto para agregar a la solución actual.

Para descargar un proyecto de una solución

- En Microsoft Visual Studio, el Explorador de soluciones, haga clic en el proyecto que desea descargar de la solución actual y haga clic en Descargar proyecto.

2.6 Editar un proyecto existente o una solución

Una vez que haya abierto un proyecto o solución en Microsoft Visual Studio, podrá editar los proyectos, módulos de clase, las formas, y las referencias utilizando el entorno de desarrollo común. También puede depurar y ejecutar los proyectos desde el entorno de desarrollo común.

Temas de esta sección

- Agregar nuevos elementos
- Importar elementos existentes
- Editar elementos
- Cambiar el nombre de un proyecto
- Añadir y referencia de otros proyectos
- Configure las opciones para Microsoft Visual Studio

2.6.1 Agregar nuevos elementos

Nuevos temas tales como módulos de clase y formularios de Windows se puede agregar a un proyecto. Usted es responsable de la actualización de las propiedades de un elemento (como el nombre) y de rellenar el código apropiado. Al nombrar nuevos elementos, recuerde que otros desarrolladores pueden querer utilizar sus artículos en futuras aplicaciones. Asegúrese de seguir su empresa establecidos convenios de denominación o las normas establecidas por Microsoft.

Procedimiento

Para agregar un nuevo elemento a su proyecto

1. En Microsoft Visual Studio, el Explorador de soluciones, haga clic en el proyecto al que desea agregar un nuevo elemento y haga clic en Agregar \ Nuevo elemento.

En lugar de hacer clic en Nuevo elemento, también puede hacer clic en Windows Form, Módulo (VB .NET solamente), o clase para agregar uno de los principales tipos de elementos a un proyecto.

1. En el Agregar nuevo elemento Caja <Proyecto> diálogo, seleccione la plantilla que representa el elemento que desea agregar.
2. Archivo o Código representa un módulo de código
3. Windows Form representa una forma (o cuadro de diálogo)
4. Clase Representa un módulo de clase

3. Escriba un nombre para el nuevo elemento y haga clic en Agregar.

2.6.2 Importación de elementos existentes

La importación le permite agregar un elemento existente a un proyecto que ya exista en otro proyecto. Usted puede importar código o módulos de clase y las formas. Los archivos con la extensión VB sólo se pueden importar en un proyecto de VB .NET y archivos CS en un proyecto de C#. Los diagramas de clases se almacenan como archivos de CD y se pueden importar en cualquier tipo de proyecto.

Al importar un artículo, una copia del archivo o un enlace al archivo que va a importar se agrega al proyecto. Cuando una copia del elemento es creado, el archivo original permanece inalterado en su lugar original. Los archivos enlazados acceso al archivo original, que es ideal para la reutilización de código común que puede ser compartido entre varios proyectos.

Si se importa un elemento con el mismo nombre que un elemento existente, se le pregunte si desea reemplazar el elemento existente en el proyecto con el que se está importando.

El componente importado se agrega al proyecto y aparece en el Explorador de soluciones. Para editar las propiedades del elemento, seleccione el elemento en el Explorador de soluciones y cambiar sus propiedades en la ventana Propiedades.

Procedimiento

Para importar un elemento existente en un proyecto

TRIAL RESTRICTION

1. En Microsoft Visual Studio, el Explorador de soluciones, haga clic en el proyecto al que desea agregar un elemento existente y haga clic en Agregar \ Elemento existente.
2. En la página Añadir elemento existente <Proyecto> caja de diálogo, busque y seleccione el archivo que contiene el elemento que desea agregar.
3. Haga clic en Agregar para crear una copia del archivo seleccionado, o haga clic en la flecha hacia abajo a la derecha de en el botón Agregar y haga clic en Agregar como vínculo para crear un enlace al archivo en lugar de crear un copia del archivo.

2.7 Editar elementos

Puede editar los módulos de código y de clase, y las formas en el entorno de desarrollo común. Módulos de código y de clase se editan en una ventana de código, mientras que las formas se editan en el Diseñador de Windows Forms. Puede abrir tantas ventanas como usted tiene artículos, lo que le permite ver el código en diferentes formas o módulos, así como copiar y pegar entre ellos.

Procedimiento

Para editar un elemento de su proyecto

- En Microsoft Visual Studio, el Explorador de soluciones, haga doble clic en el elemento que desea editar.

Si desea editar el código de un formulario o un control en un formulario, mientras que en el Diseñador de formularios de Windows, haga doble clic en el formulario o control para mostrar la ventana de código para el formulario.

Temas de esta sección

- Utilice la ventana de código
- Use el Diseñador de Windows Forms
- Utilice la ventana Propiedades

2.7.1 Utilice la ventana de código

La ventana de código contiene dos listas desplegables, una barra de división y un bar margen del indicador.

Las dos listas desplegables en la parte superior de la ventana Código mostrar el objeto actual (clase o forma) a la izquierda y los procedimientos para el objeto actual de la derecha. Puedes moverte por el proyecto, cambiando el objeto o procedimiento en las listas desplegables.

La barra de división en la parte derecha de la ventana de código le permite dividir la ventana horizontal. Simplemente arrastre la barra de división hacia abajo para crear un cristal de la ventana segundo. Esta característica le permite ver dos partes de su código de forma simultánea en el módulo mismo código. Para cerrar el panel, arrastre la barra de división a su ubicación original o doble clic en él.

La barra indicadora de margen está situado verticalmente a lo largo del lado izquierdo de la ventana de código. Se utiliza para mostrar los indicadores de margen que se utilizan durante la edición de código y depuración.

TRIAL RESTRICTION

2.7.2 Use el Diseñador de Windows Forms

El Diseñador de formularios de Windows le permite crear cuadros de diálogo personalizados para su proyecto. Para agregar un control a un formulario, haga clic en el control que desee en la ventana Cuadro de herramientas para activar la herramienta, a continuación, haga clic en el formulario en el Diseñador de Windows Forms en lugar y tamaño del control. Puede alinear controles con una rejilla u otros controles al establecer la configuración adecuada forma de la categoría General, bajo la carpeta Diseñador de Windows Forms en el cuadro de diálogo Opciones. La categoría General le permite controlar otros parámetros relacionados con la red, como el tamaño y el modo de diseño.

TRIAL RESTRICTION

Cada forma que tiene el estándar de diseño Maximizar, Minimizar y Cerrar. Estos botones se implementa como parte de la plantilla de elementos de Windows Form. Puede cambiar el aspecto y el estilo del cuadro de diálogo, seleccione el objeto de formulario en el Diseñador de Windows Forms y luego utilizando la ventana Propiedades. Para agregar código al formulario o un evento de control, haga doble clic en el formulario o control una vez que se coloca en el formulario. Esto abre la ventana de código para el formulario y agrega el evento predeterminado para el control. Use el extremo derecho en la lista desplegable en la parte superior de la ventana de código para seleccionar un evento diferente cuando se desea añadirlo a la ventana de código para el formulario o control.

Procedimiento

Para activar complemento y establecer el tamaño de la cuadrícula

1. En Microsoft Visual Studio, haga clic en el menú Herramientas \ Opciones
2. En el cuadro de diálogo Opciones, expanda el nodo Diseñador de Windows Forms carpeta en el árbol de Opciones y seleccione General.

3. En el panel Propiedades, haga clic y seleccione True SnapToGrid de la lista desplegable.
4. Haga clic GridSize y haga clic en la casilla situada a la derecha de GridSize. Introduzca un valor de espaciado de la cuadrícula utilizando el formato de ancho, altura.
5. Haga clic en Aceptar.

2.7.3 Utilice la ventana Propiedades

La ventana Propiedades se utiliza para cambiar las propiedades de los archivos del proyecto actual o solución, así como las formas de las ventanas y los controles que se colocan en un formulario Windows.

TRIAL RESTRICTION

Seleccione el archivo u objeto (formulario o control de Windows) que desea modificar. Esta muestra sus propiedades en la ventana Propiedades. Seleccione la propiedad de la cuadrícula de la ventana de propiedades que desea cambiar y, a continuación, escriba o seleccione un valor nuevo basado en la propiedad seleccionada.

Procedimiento

Para cambiar una propiedad

1. En Microsoft Visual Studio, seleccione un archivo o referencia desde el Explorador de soluciones, o un objeto (un formulario Windows Forms en el Diseñador de Windows Forms o un control en un formulario Windows).

2. Abra la ventana Propiedades, si no se ha mostrado aún.

Para mostrar la ventana Propiedades, en Microsoft Visual Studio, realice una de las siguientes acciones:

Haga clic en el archivo seleccionado, referencia, o el objeto y haga clic en Propiedades

3. En la ventana Propiedades, haga clic en la propiedad que desea editar y escriba o seleccione un nuevo valor en el campo a la derecha del nombre de la propiedad seleccionada.

2.7.4 Cambiar el nombre de un proyecto

Cuando se crea un proyecto, el nombre del archivo de proyecto se guarda con el proyecto. Si el nombre de un archivo de proyecto se cambia fuera de Visual Studio, el nombre del proyecto almacenado no se cambiará para que coincida. Como una mejor práctica, sólo cambia el nombre del proyecto o la solución de Visual Studio para mantener la coherencia. Para obtener más información sobre cómo crear un proyecto, consulte Crear un nuevo proyecto.

TRIAL RESTRICTION

Puede cambiar el nombre de un archivo de proyecto desde la ventana Propiedades o mediante el Explorador de Windows. El nombre del proyecto se establece cuando el proyecto se creó por primera vez, pero se puede cambiar a través del Explorador de soluciones. Advertencia: Un archivo de proyecto puede cambiar el nombre a través del Explorador de Windows, no se recomienda hacer esto. Se puede romper la relación con el archivo de solución o de otro archivo de proyecto. Si se rompe la relación con una solución o un proyecto a otro, al abrir el archivo de solución o proyecto, se le notificará que una referencia que falta.

Procedimiento

Para Cambiar el nombre de un proyecto

1. En Microsoft Visual Studio, el Explorador de soluciones, haga clic en el proyecto que desee cambiar el nombre y haga clic en Cambiar nombre.
2. En el editor de texto, introduzca un nuevo nombre.

TRIAL RESTRICTION

1. Pulse Intro o haga clic fuera del editor de texto en lugar de terminar el cambio de nombre del proyecto.

Añadir referencia a otros Proyectos

Agregar referencia a los proyectos le permite compartir código en varios proyectos. Puede crear bibliotecas centralizadas de los métodos comúnmente utilizados, funciones y clases, y después hacer referencia a la biblioteca cuando sea necesario. Los proyectos pueden ser referenciados en las siguientes formas:

- Agregue el proyecto a una solución y hacer referencia a un proyecto dentro de la solución
- Hacer referencia a un archivo ensamblado compilado a un proyecto

Cuando se agrega un proyecto a una solución, un nuevo nodo del proyecto se muestra en el Explorador de soluciones. Este nodo se titula igual que el proyecto que se hace referencia. Después de un proyecto se añade, se debe hacer referencia al proyecto mediante el cuadro de diálogo Agregar referencia.

Si el proyecto ha sido compilada en un archivo DLL de ensamblado, puede hacer referencia al archivo a un proyecto mediante el cuadro de diálogo Agregar referencia. Después de un proyecto o un archivo DLL se hace referencia a un proyecto, se utilizan las importaciones o mediante una declaración con el nombre de espacio de nombres o proyecto para utilizar cualquiera de los objetos expuestos (clases o formularios) en su proyecto.

Procedimiento

Para hacer referencia a otro proyecto

1. Agregar un nuevo proyecto o proyecto existente a la solución actual.
2. Una vez que el proyecto ha sido añadido a la solución actual, en el Explorador de soluciones, haga clic en el proyecto que desea agregar la referencia al proyecto y haga clic en Agregar referencia.
3. En el cuadro de diálogo Agregar referencia, Proyectos, seleccione el proyecto que desea hacer referencia. Haga clic en Aceptar.

TRIAL RESTRICTION

El proyecto seleccionado aparecerá en la carpeta Referencias en el Explorador de soluciones.

4. En el Explorador de soluciones, haga doble clic en el módulo de código que desea utilizar las funciones expuestas públicamente, métodos u objetos que están en el proyecto de referencia.
5. En la parte superior del módulo de código, agregue una instrucción Imports o using para el proyecto o espacio de nombres en el proyecto que contiene las funciones, métodos y objetos que usted desea utilizar.

Por ejemplo, si un espacio de nombres que contiene una clase expuesta en el proyecto de referencia se denomina AdskUtilities, deberá añadir lo siguiente al módulo de código para indicar que se va a utilizar ese espacio de nombres del proyecto de referencia:

VB.NET

```
Imports AdskUtilities
```

C#

```
using AdskUtilities;
```

6. Utilice el espacio de nombres de la misma manera que lo haría cualquier otra red, biblioteca COM o referencia a un proyecto.

2.7.5 Establezca las opciones para Microsoft Visual Studio

Usted puede cambiar las características del entorno de desarrollo común mediante el cuadro de diálogo Opciones. El cuadro de diálogo Opciones contiene una variedad de ajustes que afectan al entorno de desarrollo, editor de código de Windows, Diseñador de Windows Forms y Depuración. Los ajustes se organizan en carpetas que se encuentran disponibles en el panel de navegación de la izquierda. Cada carpeta contiene grupos que se utilizan para organizar las opciones relacionadas. Una vez que una carpeta o grupo es seleccionado, la configuración de la carpeta o grupo se muestran en el lado derecho.

Algunas de las carpetas de configuración son:

- Ambiente: Configuración utilizada para controlar la visualización y comportamiento de los elementos dentro del entorno de desarrollo.
- Proyectos y Soluciones: Parámetros utilizados para crear y construir proyectos y soluciones.
- Editor de texto: Ajustes utilizados para controlar el comportamiento de texto escrito en una ventana de código.
- Depuración: Configuración para controlar el entorno de depuración.
- Diseñador de Windows Forms: Configuración utilizada para controlar la visualización y el comportamiento del Diseñador de Windows Forms.

Para obtener más información sobre todas las opciones disponibles en el cuadro de diálogo Opciones, seleccione una carpeta en el panel de navegación del cuadro de diálogo Opciones y, a continuación, haga clic en el '?' botón en la esquina superior derecha. Esto muestra el tema relacionado en el archivo de Ayuda de Microsoft Visual Studio de Microsoft.

Procedimiento

- Para mostrar el cuadro de diálogo Opciones
- En Microsoft Visual Studio, haga clic en Opciones del menú Herramientas y.

2.8 Carga un ensamblado en AutoCAD

Una vez que una solución y el proyecto se crean, un espacio de nombres y clases se definen, y uno o más comandos AutoLISP® se implementan, puede utilizar el comando NETLOAD cargar un ensamblado .NET en AutoCAD.

Use el Entorno de depuración

Antes de cargar un ensamblado .NET, debe determinar si necesita utilizar el entorno de depuración de Microsoft Visual Studio para probar cualquier lógica definida en los procedimientos y funciones que pudiera haber creado. El ambiente de depuración le permite recorrer el código en el ensamblado .NET, ya que se ejecuta en tiempo real. Como el código se está ejecutando, usted puede comprobar los valores de las variables y ver qué caminos lógicos del programa se ejecutan.

Para obtener más información sobre cómo utilizar el entorno de depuración, consulte la documentación que viene con su entorno de desarrollo.

Procedimiento:

Para cargar un ensamblado .NET en modo de depuración en AutoCAD

1. En Microsoft Visual Studio, el Explorador de soluciones, haga clic en el proyecto que desea cargar en AutoCAD. Haga clic en Propiedades.
2. En la ficha <Proyecto> Propiedades, haga clic en la ficha Depurar.

3. En la ficha Depurar, bajo Acción de inicio, haga clic en Programa externo de inicio y, a continuación, haga clic en el botón de puntos suspensivos a la derecha del cuadro de texto.
4. En el cuadro de diálogo Seleccionar archivo, vaya a C:\Archivos de programa\AutoCAD 2013\acad.exe. Haga clic en Abrir.
5. Con el proyecto seleccionado en el Explorador de soluciones, haga clic en Depurar → Iniciar depuración.
6. En la ventana de comandos de AutoCAD, escriba NetLoad y pulse Intro.
7. En ventana de diálogo de ensamblados .NET, busque la versión de depuración del archivo ensamblado. Haga clic en Abrir. Tip: La ubicación del archivo de ensamblado construido en el panel de salida en Microsoft Visual Studio.
8. En la ventana de comandos de AutoCAD, escriba el nombre del comando o función AutoLISP y los parámetros requeridos.

Para cargar un NET en AutoCAD con NETLOAD

1. En Microsoft Visual Studio, con una solución o proyecto abierto, haga clic en Generar → Generar solución o Generar <nombre de proyecto>.
2. En la ventana de comandos de AutoCAD, escriba NetLoad y pulse Intro.
3. En ventana de diálogo de ensamblados .NET, busque el archivo de ensamblado construido. Haga clic en Abrir. Tip: La ubicación del archivo de ensamblado construido en el panel de salida en Microsoft Visual Studio.
4. En la ventana de comandos, escriba el nombre del comando o función AutoLISP y los parámetros requeridos.

2.9 Acceder y buscar en bibliotecas de referencia (Explorador de objetos)

El Examinador de objetos se utiliza para ver los objetos de una biblioteca de referencia en los proyectos de una solución actual. Desde el Explorador de objetos, puede navegar a través de una biblioteca de referencia como si fuera una estructura de directorios. También puede usar el menú Buscar y localizar un objeto a partir de una palabra clave introducida.

El alcance de las bibliotecas se ven en el Examinador de objetos está limitado por el menú Examinar la lista desplegable. Puede limitar el alcance de la solución actual, una versión específica del Framework .NET o una lista de componentes personalizado.

El lado izquierdo del Explorador de objetos contiene un panel de navegación que muestra las bibliotecas disponibles que puede navegar. El lado derecho se divide en dos mitades: superior (panel Miembros) e inferior (panel Descripción).

Procedimiento

Para utilizar el Examinador de objetos

1. En Microsoft Visual Studio, haga clic en Ver → Examinador de objetos (o presione Ctrl + Alt + J). El Examinador de objetos deben aparecer como una pestaña por defecto en el centro del entorno de desarrollo.
2. Opcionalmente, en el menú Examinar, seleccione un ámbito en el que desea navegar o buscar componentes.
3. Opcionalmente, en el menú Buscar, escriba o seleccione una palabra clave anterior que desea filtrar el panel Objetos.
4. En el panel Objetos, desplácese al objeto que desea ver y selecciónelo.

5. En el panel de miembros, seleccione uno de los miembros mostrados del objeto seleccionado para aprender más sobre él.

TRIAL RESTRICTION

2.10 Ejercicios: Crear su primer proyecto

Ahora que usted ha aprendido los fundamentos del uso de Microsoft Visual Studio, vamos a crear un nuevo proyecto que define un nuevo comando. Una vez que el nuevo comando está definido, usted aprenderá a cargar el ensamblado .NET en AutoCAD.

Temas de esta sección:

- Ejercicio: Crear un proyecto nuevo
- Ejercicio: Referencia de los archivos API .NET de AutoCAD
- Ejercicio: Crear un comando nuevo
- Ejercicio: Establecer el marco de destino para un proyecto
- Ejercicio: Crear y cargar un ensamblado de Framework en AutoCAD.

2.10.1 Ejercicio: Crear un proyecto nuevo

En este ejercicio, va a crear un nuevo proyecto llamado "MiPrimerProyecto" y hacer referencia a los archivos de la API de AutoCAD. NET.

Para crear un nuevo proyecto llamado "MiPrimerProyecto"

1. En el menú Inicio, haga clic en Todos los programas → Microsoft Visual Studio 2010 → Microsoft Visual Studio 2010 para iniciar Microsoft Visual Studio.
2. En Microsoft Visual Studio, haga clic en el menú Archivo → Nuevo Proyecto.
3. En el cuadro de diálogo Nuevo proyecto, haga clic en Visual Basic o C#.
4. En la lista Plantillas, seleccione Biblioteca de clases.
5. En el cuadro Nombre, escriba MiPrimerProyecto.
6. Haga clic en Aceptar.

Consejo: Para cambiar la ubicación por defecto para nuevos proyectos, seleccionar Herramientas → Opciones y Proyectos y soluciones desde el panel de navegación. Haga clic en el botón de puntos suspensivos a la derecha y busque una nueva ubicación predeterminada.

2.10.2 Ejercicio: Referencia de los archivos API .NET de AutoCAD

En este ejercicio, se referenciará a AcMgd.dll, AcDbmgd.dll y AcCoreMgd.dll, después de referenciar a los tres archivos, se ajustará las propiedades de los archivos de referencia por lo que no se copian en el directorio de construcción.

Para obtener más información sobre la referencia a los archivos la API .NET de AutoCAD, consulte Componentes de la API .NET de AutoCAD.

Para hacer referencia a los archivos de la API .NET de AutoCAD.

Para VB .NET:

1. En Microsoft Visual Studio, haga clic en Ver → Explorador de soluciones para mostrar el Explorador de soluciones, si no se ha mostrado aún.
2. Haga clic derecho en el nodo MiPrimerProyecto y seleccione Propiedades
3. En el cuadro de diálogo haga click en Referencia, haga click en Agregar.
4. Haga click en Examinar, busque la carpeta C:\ObjectARX 2013\inc. Pulse la tecla Ctrl, seleccione AcDbMgd, AcMdg y AcCoreMgd.
5. Haga clic en Aceptar.
6. Pulse la tecla Ctrl seleccione AcDbMgd, AcMdg y AcCoreMgdhaga haga click en el campo Copia local y luego seleccione False en la lista desplegable.

Para C#:

1. En Microsoft Visual Studio, haga clic en Ver → Explorador de soluciones para mostrar el Explorador de soluciones, si no se ha mostrado aún.
2. Haga clic en el nodo Referencias y seleccione Agregar referencia.
3. En el cuadro de diálogo haga click en Examinar, busque la carpeta C:\ObjectARX 2013\inc. Pulse la tecla Ctrl, seleccione AcDbMgd, AcMdg y AcCoreMgd.
4. Haga clic en Aceptar.
5. Pulse la tecla Ctrl seleccione AcDbMgd, AcMdg y AcCoreMgdhaga haga click click en el campo Copia local y luego seleccione False en la lista desplegable.

Nota: Configurar Copia local en False instruye a Microsoft Visual Studio para que no incluya el archivo DLL en la compilación del proyecto. Si el archivo DLL se copia en la carpeta de compilación, se pueden producir resultados inesperados cuando se carga un archivo de ensamblado en AutoCAD.

2.10.3 Ejercicio: Crear un comando nuevo

Ahora que ha creado un proyecto y añadido las referencias de biblioteca necesarias, es hora de crear un comando. El comando creará un nuevo texto de líneas múltiples TextoM en el espacio modelo. Estos y otros conceptos se analizan en profundidad en capítulos posteriores.

Para definir un nuevo comando que crea un nuevo objeto TextoM

- En el Explorador de soluciones, haga doble clic Class1.vb o Class1.cs en función del tipo de proyecto que ha creado.

Una ventana de código se abre para el módulo de Class1 y debe tener el siguiente aspecto:

VB.NET

```
Public Class Class1
```

```
End Class
```

C#

```
using System;
using System.Collections.Generic;
using System.Linq;
```

```
using System.Text;

namespace ClassLibrary1
{
    public class Class1
    {
    }
}
```

2. Cambiar el código en la ventana de código para que coincida con la siguiente: VB.NET

```
Imports Autodesk.AutoCAD.Runtime
Imports Autodesk.AutoCAD.ApplicationServices
Imports Autodesk.AutoCAD.DatabaseServices
Public Class Class1
    <CommandMethod("AdskGreeting")>Public Sub AdskGreeting()
        ' Get the current document and database, and start a transaction
        Dim acDoc As Document = Application.DocumentManager.MdiActiveDocument
        Dim acCurDb As Database = acDoc.Database
        Using acTrans As Transaction =
            acCurDb.TransactionManager.StartTransaction()
                ' Open the Block table record for read
                Dim acBlkTbl As BlockTable
                acBlkTbl = acTrans.GetObject(acCurDb.BlockTableId, OpenMode.ForRead)
                ' Open the Block table record Model space for write
                Dim acBlkTblRec As BlockTableRecord
                acBlkTblRec =
                    acTrans.GetObject(acBlkTbl(BlockTableRecord.ModelSpace), OpenMode.ForWrite)
                ' Creates a new MText object and assigns it a location,
                ' text value and text style
                Dim objText As MText = New MText
                ' Set the default properties for the MText object
                objText.SetDatabaseDefaults()
                ' Specify the insertion point of the MText object
                objText.Location = New Autodesk.AutoCAD.Geometry.Point3d(2, 2, 0)
                ' Set the text string for the MText object
                objText.Contents = "Greetings, Welcome to the AutoCAD .NET"
                Developer() 's Guide"
                ' Set the text style for the MText object
                objText.TextStyleId = acCurDb.Textstyle
                ' Appends the new MText object to model space
                acBlkTblRec.AppendEntity(objText)
                ' Appends to new MText object to the active transaction
                acTrans.AddNewlyCreatedDBObject(objText, True)
                ' Saves the changes to the database and closes the transaction
                acTrans.Commit()
            End Using
        End Sub
    End Class
```

C#

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

using Autodesk.AutoCAD.Runtime;
using Autodesk.AutoCAD.ApplicationServices;
using Autodesk.AutoCAD.DatabaseServices;
[assembly: CommandClass(typeof(MyFirstProject1.Class1))]
namespace MyFirstProject1
{
    public class Class1
    {
        [CommandMethod("AdskGreeting")]
        public void AdskGreeting()
        {
            // Get the current document and database, and start a transaction
            Document acDoc = Application.DocumentManager.MdiActiveDocument;
            Database acCurDb = acDoc.Database;
            // Starts a new transaction with the Transaction Manager
            using (Transaction acTrans = acCurDb.TransactionManager.StartTransaction())
            {
                // Open the Block table record for read
                BlockTable acBlkTbl;
                acBlkTbl = acTrans.GetObject(acCurDb.BlockTableId, OpenMode.ForRead) as BlockTable;
                // Open the Block table record Model space for write
                BlockTableRecord acBlkTblRec;
                acBlkTblRec = acTrans.GetObject(acBlkTbl[BlockTableRecord.ModelSpace],
                OpenMode.ForWrite) as BlockTableRecord;
                /* Creates a new MText object and assigns it a location,
                text value and text style */
                MText objText = new MText();
                // Set the default properties for the MText object
                objText.SetDatabaseDefaults();
                // Specify the insertion point of the MText object
                objText.Location = new Autodesk.AutoCAD.Geometry.Point3d(2, 2, 0);
                // Set the text string for the MText object
                objText.Contents = "Greetings, Welcome to the AutoCAD .NET Developer's Guide";
                // Set the text style for the MText object
                objText.TextStyleId = acCurDb.Textstyle;
                // Appends the new MText object to model space
                acBlkTblRec.AppendEntity(objText);
                // Appends to new MText object to the active transaction
                acTrans.AddNewlyCreatedDBObject(objText, true);
                // Saves the changes to the database and closes the
                transaction
                acTrans.Commit();
            }
        }
    }
}

```

```

    }
}
}

```

2.10.4 Ejercicio: Crear y cargar una ensamblado .NET en AutoCAD.

Ahora que tiene creado un proyecto y definió un comando, está casi listo para ejecutar el comando en AutoCAD. Antes de que pueda ejecutar el comando en AutoCAD, debe compilar primero o crear un archivo ensamblado .NET para su proyecto. Para este ejercicio, va a generar una versión de depuración del proyecto, pero no aprenderá acerca de cómo utilizar el entorno de depuración en este momento. Para obtener más información sobre la depuración de un proyecto, consulte la documentación que viene con su entorno de desarrollo. Para cargar utilizando el entorno de depuración con AutoCAD, consulte Cargar un ensamblado en AutoCAD. Si quieres distribuir tu proyecto para que otros puedan utilizar, usted tendrá que construir un instalador .NET de su proyecto. Para obtener más información sobre cómo compartir archivos .NET con otros, ver distribuir la aplicación. Una vez ensamblado .NET está construido, puede cargarlo en AutoCAD con el comando NETLOAD.

Para generar un proyecto y cargar un ensamblado .NET en AutoCAD

1. En Microsoft Visual Studio, haga clic en Generar → Generar MiPrimerProyecto.

El proyecto debe ser construir con éxito, a menos que algo este mal con el código del proyecto. Mire en la ventana de resultados para obtener información sobre el estado de construcción del proyecto. La ubicación del archivo MiPrimerProyecto.dll que se construye también se muestra en la ventana de resultados.

1. Inicie AutoCAD si no se está ejecutando.
2. En la ventana de comandos de AutoCAD, escriba NetLoad y pulse Intro.
3. En el cuadro de diálogo, vaya a la ubicación de la MiPrimerProyecto.dll que se muestra en la ventana de resultados en Microsoft Visual Studio y seleccionarlo.
4. Haga clic en Abrir.
5. En la ventana de comandos de AutoCAD, escriba adskgreeting y pulse Intro.

Un nuevo objeto TextoM se crea en las coordenadas (2,2) con la cadena de texto "Saludos, bienvenidos a la Guía de desarrollo .NET AutoCAD".

2.11 Comandos relacionados con AutoCAD y Terminología

2.11.1 Comandos AutoCAD

Comando NETLOAD Carga un ensamblado .NET en AutoCAD, se muestra una cuadro de diálogo estándar para elgir los ensamblados .NET. Cuando FILEDIA se establece en 0 (cero), NETLOAD muestra el siguiente mensaje: *Nombre del ensamblado: Escriba un nombre de archivo y pulse Intro*

2.11.2 Terminología

Ensamblado Un proyecto compilado que tiene la extensión de archivo DLL.

VB.NET Project Un archivo de proyecto creado con Microsoft Visual Studio que tiene la extensión de archivo vbproj.

C# Project Un archivo de proyecto creado con Microsoft Visual Studio que tiene la extensión csproj.

Ventana del editor de código Ventana del editor de código, se utiliza para editar el código almacenado en un módulo clase o form.

Solución Archivo que se utiliza para administrar uno o más archivos de proyecto cargado en Microsoft Visual Studio.

Referencia Un enlace a un archivo de biblioteca de API que utiliza un proyecto. Un archivo de proyecto también pueden hacer referencia a otro proyecto.

2.12 Más Información

Más información sobre el entorno de desarrollo Microsoft Visual Studio, y los lenguajes de programación VB.NET o C#, están disponibles en los archivos de ayuda proporcionados por Microsoft. Para acceder a los archivos de Ayuda, elija una de las opciones en el menú Ayuda de Microsoft Visual Studio. También podría considerar la compra de un libro publicado por Microsoft Press® o un editor de terceros.