

# **Capítulo 6**

## **Administración de múltiples servidores**

Al finalizar el capítulo, el alumno podrá:

- Identificar las diferentes opciones en la administración centralizada de servidores.
- Implementar objetos de administración desde un servidor centralizado.
- Monitorear la operación de múltiples servidores desde un servidor centralizado.

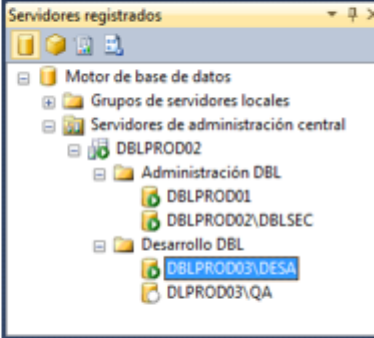
### **Temas**

1. Configuración de servidores múltiples
2. Ejecución de consultas y directivas a través de múltiples servidores
3. Administración de trabajos y alertas de múltiples servidores
4. Generando aplicaciones de capa de datos

## 1. Configuración de servidores múltiples

### Configuración de servidores múltiples

- Permite organizar los servidores en grupos.
- A través de la opción Servidores registrados de Management Studio.
- Se configura por niveles:
  - Servidor de administración central.
  - Grupos de servidores.
  - Registro de servidores pertenecientes al grupo.



6 - 4

Copyright © Todos los Derechos Reservados - Cibertec Perú S.A.C.

Los administradores de bases de datos que trabajan con SQL Server están siendo cada vez más requeridos para trabajar un número mayor de servidores dentro de una organización. La necesidad de administrar múltiples servidores representa un reto adicional para los DBA. SQL server ha venido introduciendo nuevas herramientas y mejoras a las ya existentes para asistir a los administradores.

La herramienta para administrar múltiples servidores SQL Server se llama Servidor Central de Administración o CMS (Central Management Server).

CMS fue introducido en la versión 2008 y permite definir grupos de servidores para ejecutar consultas Transact-SQL y evaluar directivas contra un grupo de servidores definidos como objetivo.

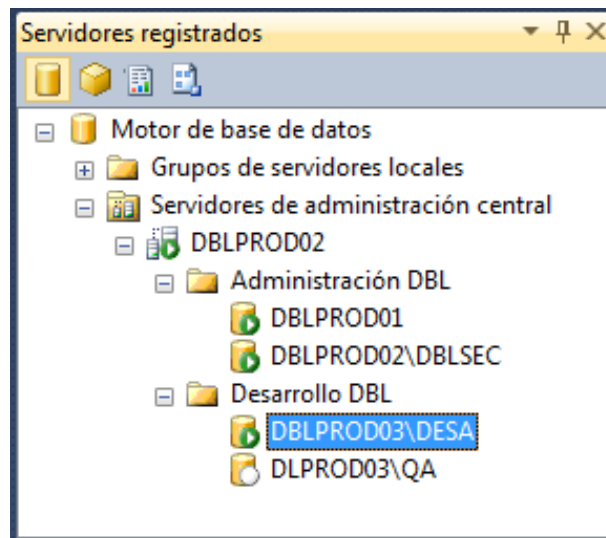
Desde varias versiones atrás, SQL Server Management Studio ha provisto de herramientas que tienen la capacidad de definir grupos de servidores a través de la herramienta servidores registrados. El uso típico de esta característica era agrupar conjuntos de servidores en diferentes entornos. Por ejemplo, se puede crear un grupo de servidores de prueba, otro grupo de servidores de desarrollo y un grupo de servidores de producción.

En versiones anteriores no se permitían acciones contra grupos enteros de servidores y un reto clave era lograr que la definición de los grupos de servidores pudiera ser almacenada localmente en cada computadora que estuviera ejecutando Management Studio. Esto debido a que, si un usuario cambiaba de computadora, los grupos de servidores no existirían en el nuevo equipo y se necesitaría recrear las agrupaciones.

Además, si el grupo de servidores cambiaba, la nueva estructura debía ser modificada en cada computadora ejecutando Management Studio.

### Servidor de administración central (CMS - Central Management Server)

Un servidor de administración central es un servidor configurado para mantener una lista centralizada de los grupos de servidores, los cuales solo se necesita que sean definidos una vez y pueden ser accedidos por todas las computadoras que necesitan conectarse a alguno de ellos.



La lista centralizada está almacenada en tablas dentro de la base de datos **msdb** del servidor designado como servidor de administración central.

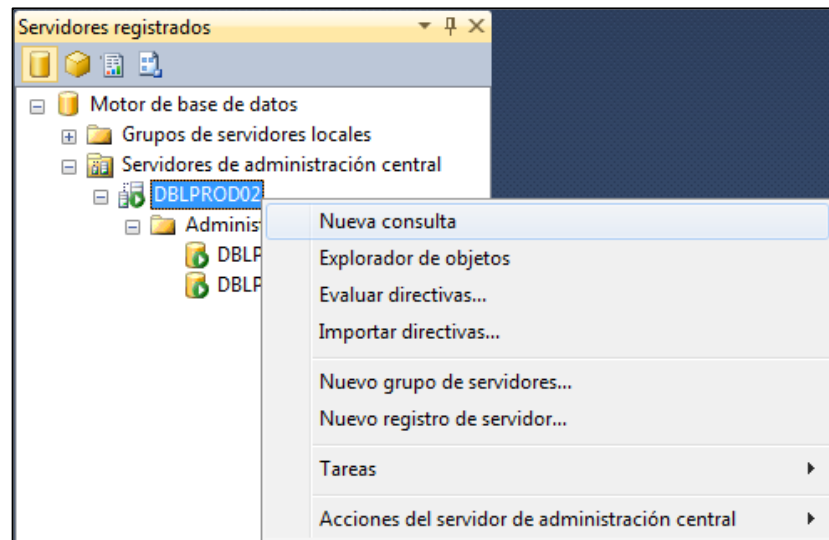
Los usuarios que requieran tener acceso a la lista de servidores necesitan ser miembros del rol **ServerGroupReaderRole** que se encuentra en la base de datos msdb, del servidor designado como servidor de administración central.

Todas las ediciones y versiones de SQL server pueden ser registradas a través de CMS, teniendo en cuenta que la autenticación tiene que ser de Windows, por lo que se recomienda el uso de cuentas de dominio. El servidor administración de central tiene que ser versión SQL Server 2008 R2 o superior.

Como se ha comentado anteriormente, desde versiones anteriores se puede mantener una lista de los servidores y sus grupos, pero desde SQL server 2008 en adelante se cuenta con la capacidad de ejecutar acciones contra todo un grupo de servidores, mediante el menú contextual al que se invoca haciendo clic derecho.

Las operaciones que se pueden ejecutar son:

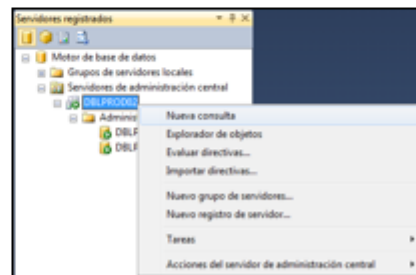
- Comandos Transact-SQL.
- Evaluar e importar directivas de administración.
- Navegar por el explorador de objetos.



## 2. Ejecución de consultas y directivas a través de múltiples servidores

### Ejecución de consultas y políticas en múltiples servidores

- Las políticas pueden ser evaluadas en múltiples servidores, a la misma vez.
- Los comandos pueden ser ejecutados en múltiples servidores, a la misma vez.
- Los resultados pueden ser unidos.
- Se incluyen columnas adicionales para identificar el servidor.



6 - 6

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.

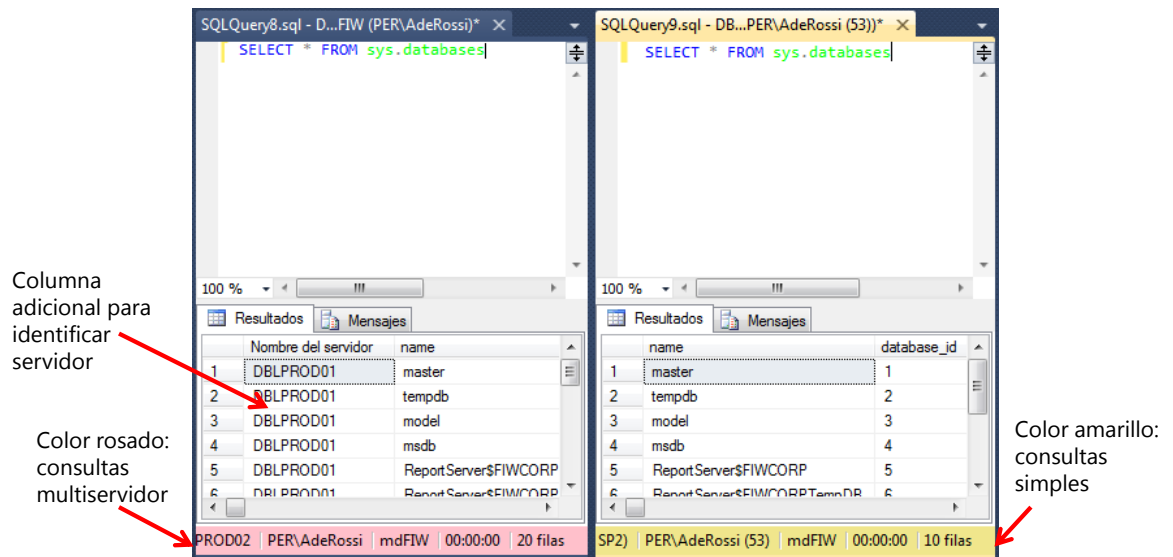


Cuando se inicia una sesión de consulta a un grupo de servidores, Management Studio asigna un color a la barra de estado, al pie de la ventana de consulta, que indica si se trata de una consulta estándar o una consulta multiservidor.

Color amarillo. - Consulta estándar.

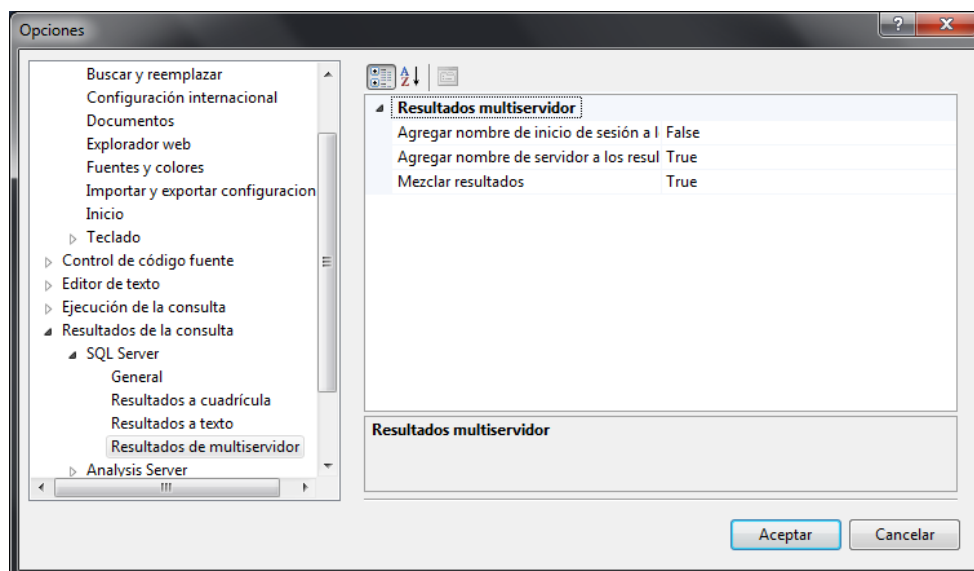
Color rosado. - Consulta multiservidor.

Adicionalmente, en la barra de estado, cuando la consulta es multiservidor se indica el nombre del grupo de servidores, en vez del nombre de un servidor:



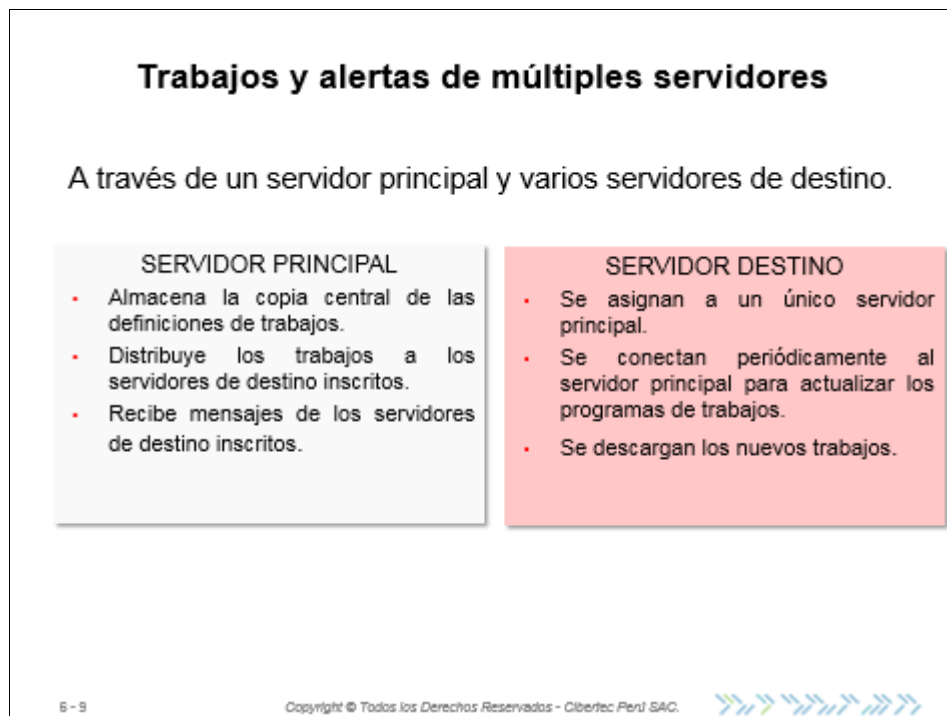
Nótese que en la consulta multiservidor se añade una columna adicional que no forma parte de la consulta pero que identifica al servidor al cual pertenece la información. Esto permite identificar al administrador de cuál de los servidores pertenecientes al grupo proviene la información.

Para la ejecución de consultas multiservidor, Management Studio provee opciones para determinar cómo se deben presentar los resultados. Desde el menú herramientas, seleccionar el submenú opciones. Desplegar la opción resultados de la consulta / SQL Server / resultados multiservidor, tal como se muestra en la siguiente figura.



Por defecto, Management Studio va a unir los resultados de cada servidor en un solo conjunto de resultados e insertar una columna a la izquierda que contenga el nombre del servidor para poder identificar la proveniencia de los datos. Opcionalmente, se puede quitar esa columna, agregar la columna del nombre de inicio de sesión o requerir que el resultado de la consulta no sea unido si no que muestre un conjunto de datos por cada servidor del grupo que ha sido consultado.

### 3. Administración de trabajos y alertas de múltiples servidores



SQL Server permite automatizar los trabajos en varios servidores de una red. La automatización de los trabajos multiservidor consiste en definir un servidor principal, y uno o varios servidores de destino. Un servidor principal distribuye los trabajos y recibe los eventos de los servidores de destino en red.

Tener una configuración de administración de varios servidores permite:

- Agrupar varios servidores en unidades empresariales de funcionamiento lógico.
- Administrar varios servidores desde una ubicación.

#### 3.1 Servidores principal y de destino

Para aprovecharse de la administración de varios servidores, debe tener por lo menos, un servidor principal y uno de destino.

Servidor principal: almacena la copia central de las definiciones de trabajo para trabajos que se ejecutan en servidores de destino. Un servidor principal distribuye los trabajos y recibe los eventos de los servidores de destino que se muestran.

Servidores de destino: se conectan periódicamente al servidor principal para actualizar su programación de trabajos. Si existe un nuevo trabajo en el servidor principal, el servidor de destino descarga el trabajo. Después de que el servidor

de destino finalice el trabajo, vuelve a conectarse al servidor principal y crea un informe con el estado del trabajo.

Para automatizar los trabajos en varios servidores, debe definir primero un servidor principal, y uno o varios servidores de destino. Un servidor principal define, programa y administra los trabajos en todos los servidores de destino.

Debido a que un servidor principal necesita una carga de conexión más alta, solo es posible definir un servidor principal en un equipo con Windows Server 2003 o superior.

### Definir un servidor principal

Puede usar el Asistente para servidor principal en SQL Server Management Studio o ejecutar el procedimiento almacenado del sistema `sp_msx_enlist` para definir el servidor principal. El asistente le guiará, a través de los pasos siguientes:

- Comprobación de la configuración de seguridad del servicio Agente SQL Server y SQL Server en todos los servidores que vayan a ser servidores de destino.

**Nota**

*Ambos servicios deben estar ejecutándose como cuentas de dominio de Windows.*

- Creación de un operador de servidor principal (MSXOperator) en el servidor principal. MSXOperator es el único operador que puede recibir notificaciones para los trabajos multiservidor.
- Inicio del servicio Agente SQL Server en el servidor principal.
- Inscripción de uno o varios servidores como servidores de destino

### Definir un servidor de destino

Se puede usar el Asistente para servidor de destino en SQL Server Management Studio o ejecutar el procedimiento almacenado del sistema `sp_msx_enlist` para definir servidores de destino adicionales. La definición del servidor de destino se almacena en la tabla del sistema `systargetservers` en la base de datos `msdb`.

Los servidores de destino:

- Se asignan a un único servidor principal.
- Deben residir en el mismo dominio de Windows que el servidor principal o en un dominio de Windows de confianza.
- No pueden ser miembros de otros servidores principales hasta que dejen sus servidores principales actuales.



### 3.2 Implementar jobs multiservidor

Después de definir los servidores, principal y de destino, se pueden crear trabajos en el servidor principal para que se ejecuten en uno o varios servidores de destino.

Los servidores de destino para el trabajo se especifican usando la página Destinos en el cuadro de diálogo, Propiedades del trabajo. Esta página no se muestra hasta que SQL Server se haya definido como servidor principal o de destino en un entorno multiservidor.

SQL Server pasa por los pasos siguientes para procesar los trabajos en un entorno multiservidor:

1. El servidor principal expone los trabajos para los servidores de destino en una lista de descargas en la tabla de sistema sysdownloadlist en la base de datos msdb.
2. Los servidores de destino se conectan periódicamente al servidor principal para determinar si se ha expuesto algún trabajo nuevo o actualizado para descargarlo.
3. El servidor de destino carga el estado del resultado del trabajo en el servidor principal cuando el trabajo finaliza.

**Nota**

*Las rutas de acceso y sintaxis en el trabajo deben ser idénticas en todos los servidores de destino; de lo contrario, el trabajo no funcionará en todos los servidores.*

### **Modificar las definiciones de trabajo multiservidor**

El servidor principal almacena la copia principal de definiciones de trabajo y las programaciones. Al realizar cualquier cambio en los trabajos en un entorno multiservidor, tenga en cuenta lo siguiente:

- Las definiciones de trabajo no se pueden modificar en el servidor de destino sino que debe hacerse en el servidor principal.
- El Administrador corporativo de SQL Server expone, automáticamente, las instrucciones necesarias en la lista de descargas.

### 3.3 Envío de Eventos (Event Forwarding)

Se pueden reenviar todos los mensajes de eventos que cumplen o exceden un nivel de gravedad de error concreto a una instancia de SQL Server. Esto se denomina reenvío de eventos. El servidor de reenvío es un servidor dedicado que también puede ser un servidor principal. Puede usar el reenvío de eventos para centralizar la administración de alertas de un grupo de servidores, con lo que se reduce la carga de trabajo de los servidores con un gran uso.

Cuando un servidor recibe eventos para un grupo de otros servidores, el servidor que los recibe se denomina servidor de administración de alertas. El uso del reenvío de eventos no está relacionado con los servidores: principal y de destino,

pero si ha configurado un entorno de varios servidores, por lo general, debería designar el servidor principal como el servidor de administración de alertas.

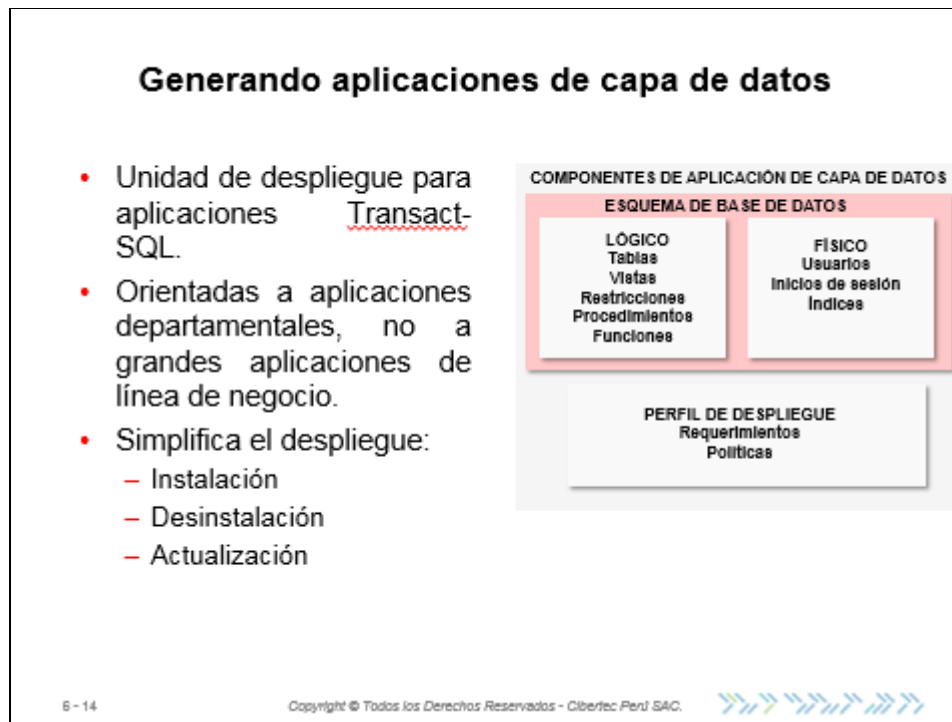
**Ventajas:**

- **Centralización.** Es posible realizar un control centralizado y una vista consolidada de los eventos de varias instancias de SQL Server desde un servidor único.
- **Escalabilidad.** Se pueden administrar muchos servidores físicos como un servidor lógico. Se pueden agregar o quitar servidores de este grupo de servidores físicos, según sea necesario.
- **Eficiencia.** Se reduce el tiempo de configuración porque solo se necesita definir las alertas y operadores una vez.

**Desventajas**

- **Aumento de tráfico.** El reenvío de eventos a un servidor de administración de alertas puede aumentar el tráfico de la red. Este aumento puede ser moderado, si se restringe el reenvío de eventos a eventos que están por encima de un nivel de gravedad designado.
- **Un único punto de error.** Si el servidor de administración de alertas se queda sin conexión, no se emitirá ninguna alerta para cualquier evento del grupo de servidores administrado.
- **Carga del servidor.** El control de alertas para los eventos reenviados provoca un aumento en la carga de procesamiento en el servidor de administración de alertas.

## 4. Generando aplicaciones de capa de datos



En la versión 2008 R2, SQL server introduce una nueva forma de proyecto de base de datos llamada **Aplicaciones de capa de datos** (datatier applications).

Mientras que los desarrolladores de base de datos no necesariamente requieren entender cómo crear este tipo de aplicaciones, los administradores de bases de datos necesitan cómo extraer, desplegar y actualizar la capa de datos de una aplicación. Por ahora, este mecanismo está pensando para ser utilizado con bases de datos pequeñas o medianas utilizadas en aplicaciones departamentales.

### 3.1 Aplicaciones de capa de datos

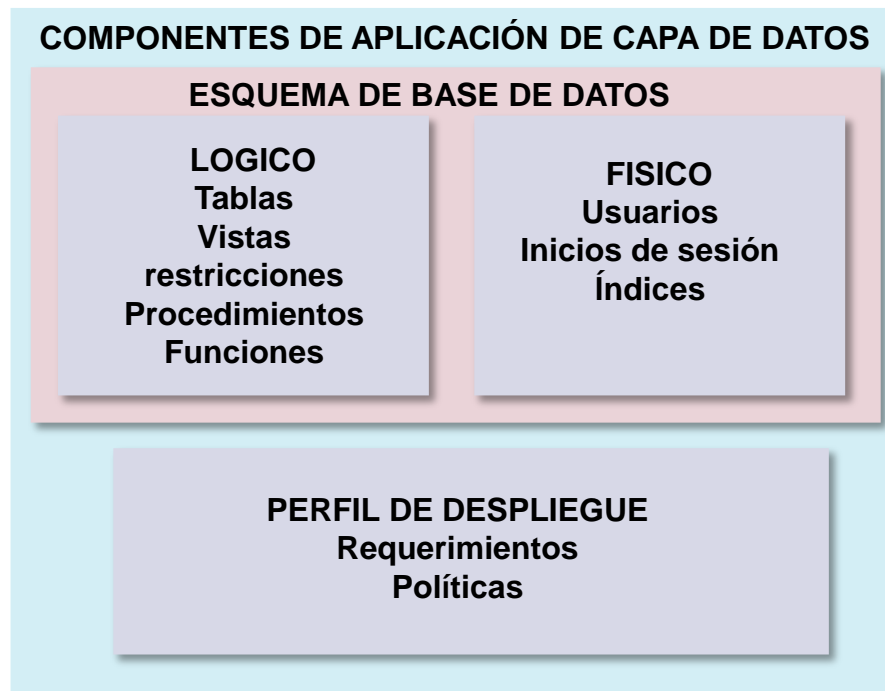
Las aplicaciones de capa de datos están diseñadas para ofrecer una experiencia de instalación y actualización de bases de datos, similar a como ocurre cuando se instala una aplicación de Windows. El desarrollador crea una aplicación de capa de datos usando Visual Studio, con todos los objetos requeridos y define las directivas que limitan la forma cómo puede ser instalada la aplicación.

Por ejemplo, una directiva de despliegue puede indicar que la aplicación solo puede ser instalada en servidores SQL Server versión 10.5 o superior (SQL Server 2008 R2 o superior)

Cuando el proyecto de aplicación de capa de datos es construido, el archivo resultante en un paquete DATPAC, el cual puede ser entregado al administrador de base de datos. Un solo archivo DATPAC puede ser utilizado tanto para

instalar como para actualizar una aplicación y es portable, a través de varios entornos como son los de desarrollo, pruebas y producción.

La instalación y actualización de la aplicación de capa de datos es automatizada y se ejecuta, a través de asistentes, paso a paso.



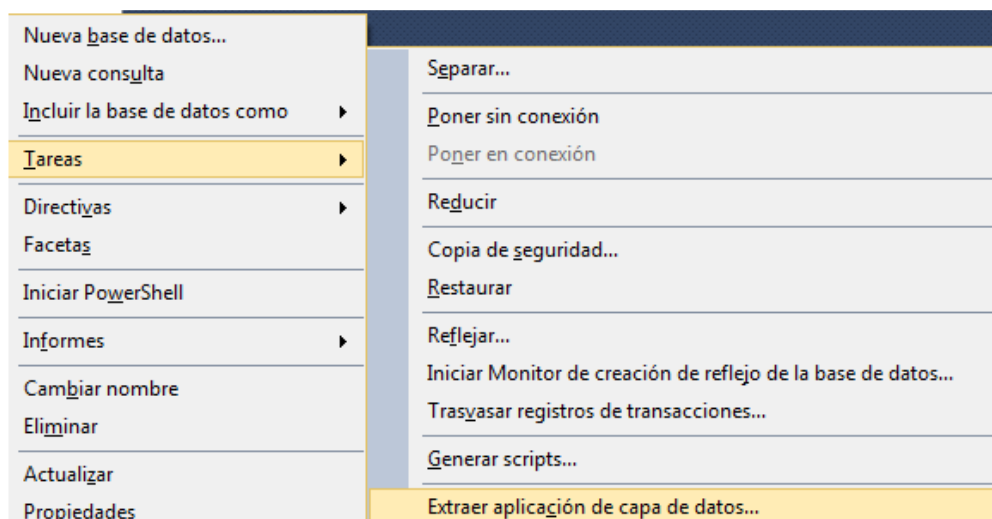
Se debe tener en cuenta que las aplicaciones de capa de datos no están pensadas para grandes aplicaciones de negocio o sistemas de misión crítica. La intención es simplificar el despliegue de un número grande de aplicaciones simples.

### 3.2 Extracción de aplicación de capa de datos

No todas las aplicaciones se desarrollan desde cero. Hay una necesidad de mejorar permanentemente aplicaciones que ya existen.

Management Studio provee dos formas de trabajar con aplicaciones existentes:

- Registrar una base de datos existente como una aplicación de capa de datos.
- Extraer la aplicación de capa de datos, a través de un DATPAC, desde una base de datos existente.



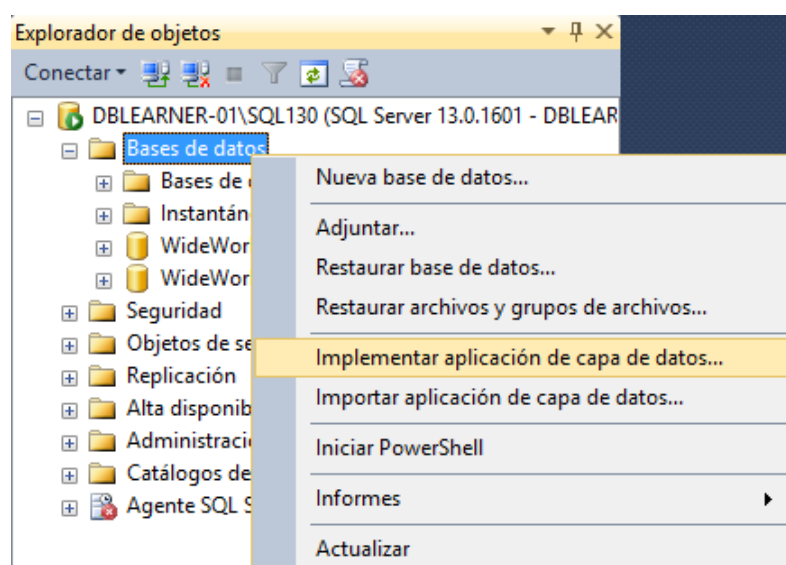
Es importante definir que, actualmente, las aplicaciones de capa de datos no soportan todos los objetos. Algunos de los objetos aun no soportados son:

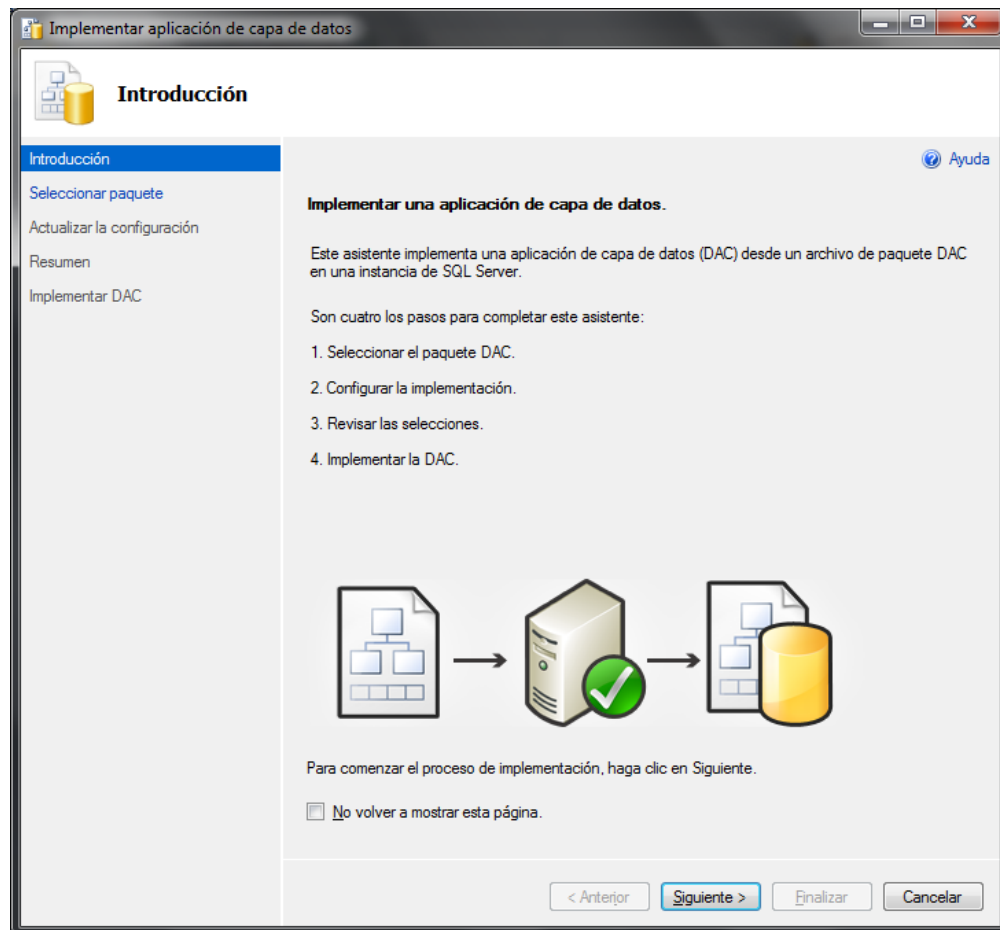
- XML
- Schema collections
- Objetos basados en SQL CLR

Es por esta razón que no todas las bases de datos pueden ser registradas como aplicación de capa de datos o extraídas a un archivo DATPAC. El asistente paso a paso, indica cuando existen objetos no soportados y detalla cuales son, no pudiendo continuar con el proceso de registro o extracción de capa de datos.

### 3.3 Despliegue de aplicación de capa de datos

El proceso de despliegue de aplicaciones de capa de datos ha sido diseñado para ser bastante directo y automatizado. Management Studio incluye un asistente, paso a paso, para la instalación de la aplicación de capa de datos. El asistente inicia solicitando el nombre de la base de datos y la ubicación de los archivos de base de datos.





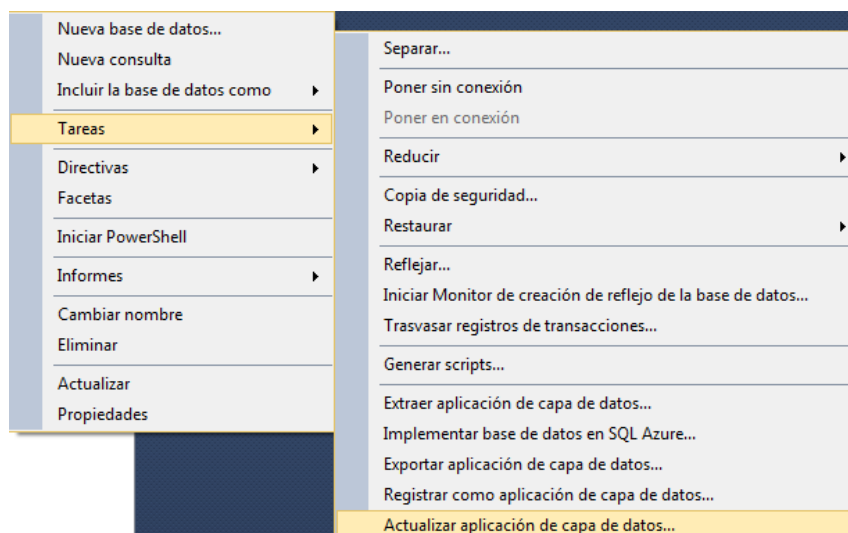
Antes de realizar cualquier cambio, el asistente paso a paso, verifica las directivas incluidas en el paquete de despliegue. En la versión SQL Server 2008 R2, la única directiva disponible era aquella que determinaba en qué versión de SQL Server podía ser instalada la aplicación. Posteriores versiones de aplicaciones de capa de datos, incluyen otros tipos de directivas como la definición de requerimientos de conmutación por falla o alta disponibilidad.

Antes de desplegar el archivo de paquete DATPAC como una aplicación de capa de datos, es importante asegurarse de la confianza de la fuente de dicho archivo. Si la proveniencia no es de confianza, se recomienda inspeccionar y verificar los archivos contenidos en el paquete DATPAC antes de desplegarlos. Además, se recomienda que, así como se hace con cualquier tipo de aplicación, antes de instalar la aplicación de capa de datos en un entorno de producción, esta sea validada en un entorno de prueba.

Es posible visualizar el contenido de un archivo de paquete DATPAC importando el archivo en un proyecto de aplicación de capa de datos en Visual Studio. Si no se cuenta con Visual Studio, se cuenta con la opción para desempaquetar el archivo, a través del menú contextual al cual se accede haciendo clic derecho sobre el archivo.

### 3.4 Actualización de aplicación de capa de datos

Las actualizaciones de las aplicaciones de capa de datos son también automatizadas. En Management Studio, haciendo clic derecho en una aplicación de capa de datos existente o en una base de datos existente, se cuenta con una opción para actualizar la aplicación. Esta actualización es también basada en un asistente paso a paso, diseñada para aplicar la actualización con facilidad.



Cuando se actualiza una aplicación de capa de datos, una nueva base de datos es creada y todos los datos de la base de datos que se está actualizando es migrada hacia la nueva base de datos actualizada. La base de datos original es renombrada y su estado es cambiado a solo lectura como una copia segura en caso de fallas, por si sucede algún problema durante el proceso de actualización de la base de datos.

Copiar los datos desde la base de datos anterior a la nueva puede ser un proceso intenso en el uso de recursos. Esta es la principal razón por la cual las aplicaciones de capa de datos no deben ser utilizadas para bases de datos que almacenan grandes volúmenes de información. Los requerimientos de espacio en disco para el proceso de copia también deben ser considerados.

Todas las funcionalidades de las aplicaciones de capa de datos que es manejada desde Management Studio pueden ser también administradas mediante el uso de scripts de PowerShell.

Una gran ventaja de las aplicaciones de capa de capa de datos es que, si ocurriera algún error en la actualización, los cambios son deshechos y la base de datos original es puesta en línea nuevamente. A diferencia de las actualizaciones basadas en scripts, los cuales pueden llevar a dejar la base de datos en un estado inconsistente en caso de error.