

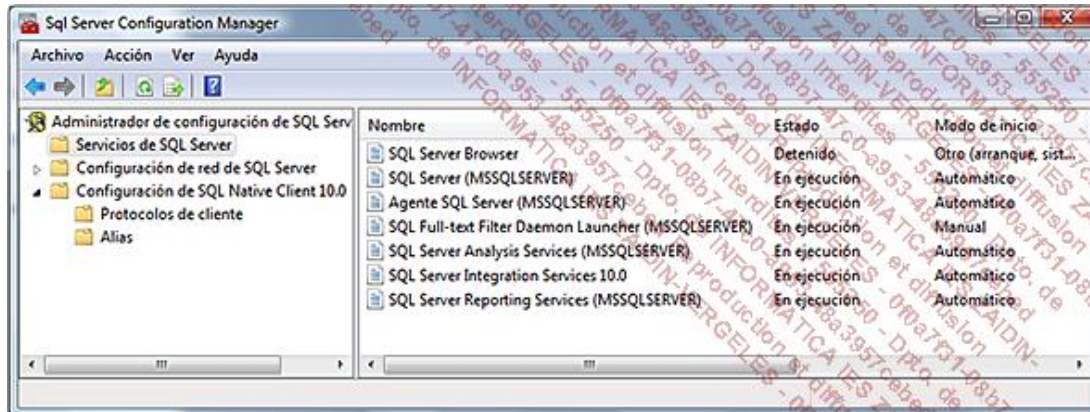
# La configuración

Antes de poner funcionamiento el servicio del servidor SQL y hacerlo accesible para todos los usuarios, es importante realizar cierto número de operaciones de configuración del servidor y de las herramientas de administración cliente con el objetivo de protegerse contra cualquier operación delicada.

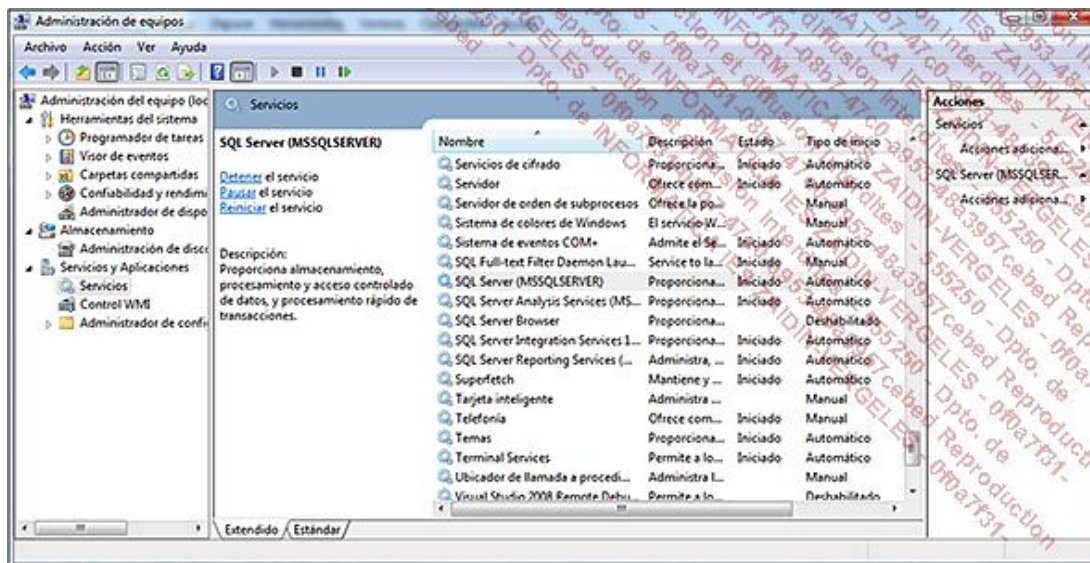
## 1. Los servicios

Los diferentes componentes del servidor se ejecutan en forma de servicios. Por lo tanto, es necesario que estos servicios sean iniciados para poder trabajar con el servidor. Estos servicios se pueden gestionar con el administrador de configuración de SQL Server aunque también se pueden gestionar como todos los servicios de Windows.

Desde el administrador de configuración, es sencillo visualizar el estado del servicio y modificar sus propiedades.



Como todos los servicios de Windows, se pueden gestionar de manera centralizada en el servidor Windows.



Por último, es posible actuar sobre estos servicios directamente en **línea de comandos** por medio de los comandos **net start** y **net stop**. En el momento de un inicio por medio de línea de comandos, es posible anular la configuración por defecto del servicio especificando la configuración que se debe utilizar en forma de parámetros. Por ejemplo, la opción **m** (**net start mssqlserver m**) permite iniciar el servidor en modo monousuario.

En caso de problemas de inicio, es posible iniciar el servidor SQL Server como una aplicación con la ayuda de **sqlservr.exe**. La utilización de este ejecutable permite iniciar la instancia **sin tener en cuenta** todas las opciones de configuración definidas.

### Los diferentes estados de los servicios

#### Iniciado

Cuando el servicio MSSQL Server está iniciado, los usuarios pueden establecer nuevas conexiones y trabajar con los

datos contenidos en base. Cuando el servicio SQL Server Agent está iniciado, el conjunto de tareas planificadas, las alertas y la replicación está activo.

### Suspendido

Si el servicio MSSQL Server está suspendido, el usuario no puede establecer ninguna conexión con el servidor. **Los usuarios conectados no se ven afectados por esta medida**. La suspensión del servicio **SQL Server Agent desactiva** la planificación de todas las tareas y las alertas.

### Detenido

La parada del servicio MSSQL Server desactiva todas las conexiones de usuario y **lanza un proceso de CHECKPOINT** (el conjunto de los datos validados presentes en memoria persisten en el disco duro y se inscribe el punto de sincronización en el diario). Este mecanismo permite asegurar que el próximo inicio del servidor será correcto. Sin embargo, el servicio espera a que terminen todas las instrucciones en curso de ejecución antes de detener el servidor. La parada del servicio SQL Server Agent desactiva la ejecución planificada de todas las tareas y la gestión de las alertas.

## 2. SQL Server Management Studio

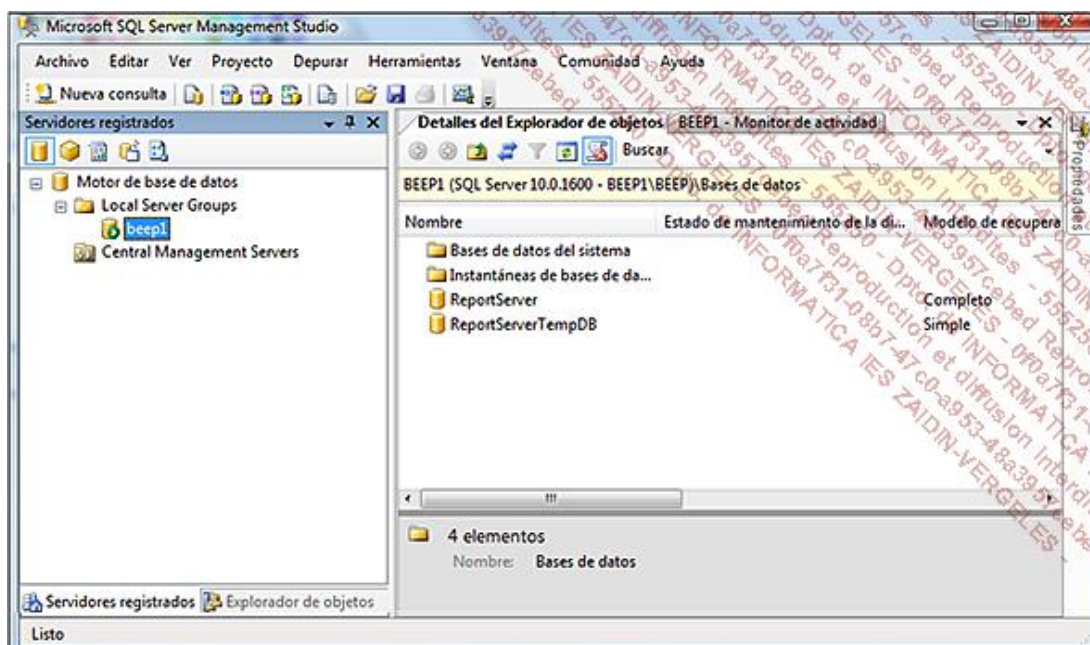
SQL Server Management Studio es la herramienta de gestión gráfica de SQL Server que permite realizar las tareas administrativas y todas las operaciones de desarrollo. La utilización de la misma herramienta permite reducir la distinción entre los grupos de usuarios, administradores y desarrolladores. Al compartir la misma herramienta, es más fácil saber lo que se puede hacer de otra manera.

Para poder navegar de una instancia a otra de SQL Server, posiblemente con servidores diferentes, es necesario registrar cada servidor en la consola de administración. Este registro no es necesario para la instancia local de SQL Server, ya que en el momento de la creación de la instancia la información relativa a esta instancia se añade a SQL Server Management Studio.

### Registrar un servidor

La ventana **Servidores registrados** permite conocer la lista de servidores registrados en SQL Server Management Studio. Si esta ventana no está visible, es posible iniciarla mediante la opción de menú **Ver - Servidores registrados** o mediante la combinación de teclas [Ctrl][Alt] **G**.

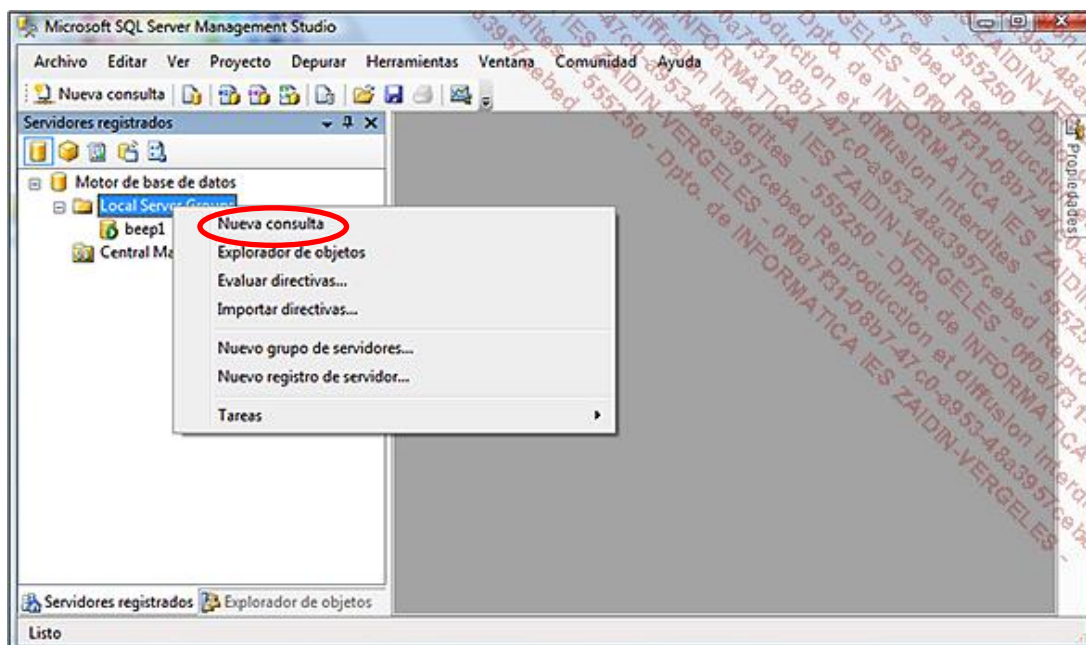
Los servidores se agrupan por tipo. Para cada tipo es posible definir los grupos de servidores con el objetivo de agruparlos por algún otro criterio, como por ejemplo la ubicación física. Los grupos de servidores no tienen ninguna influencia en el registro del servidor. Es posible cambiar de lugar un servidor y situarlo en otro grupo seleccionando la opción **Tareas - Desplazar hacia** desde el menú contextual asociado al servidor. Por analogía, es posible comparar los grupos de servidores con directorios y los servidores registrados con archivos. Los archivos no se ven afectados cuando se desplazan de un directorio a otro. Sucede lo mismo con el registro de servidores. Los directorios se definen para agrupar los archivos siguiendo una lógica, que es exactamente la función de los grupos de servidores.



Para registrar un servidor nuevo utilizando SQL Server Management Studio, es necesario seleccionar la opción **Nuevo**

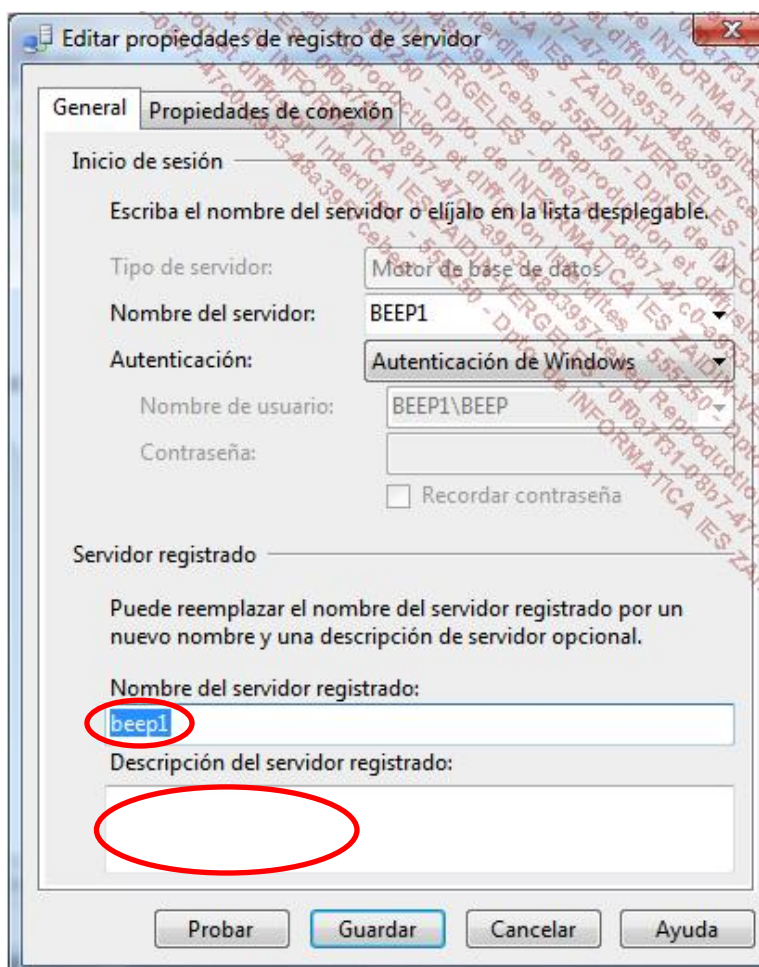


**registro de servidor** del menú contextual asociado al nodo **Local Server Groups** en la ventana **Servidores registrados**.



El cuadro de diálogo que permite realizar el registro está compuesto de dos pestañas. La primera pestaña permite completar todos los datos generales de registro, como el nombre del servidor, así como el tipo de autenticación utilizada para establecer la conexión con el servidor.

➤ Desde SSMS es posible registrar servidores SQL Server, Analysis Services, SQL Server Compact Edition, Reporting Services e Integración Services.



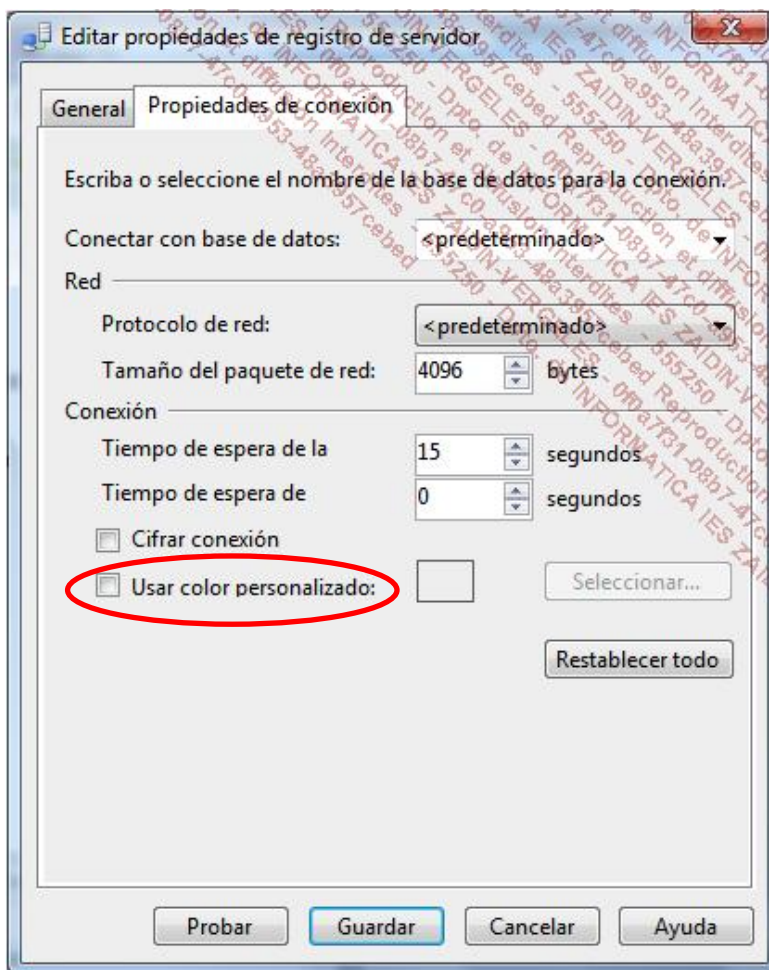
El botón **Probar** permite comprobar si la conexión elegida permite trabajar correctamente con el servidor seleccionado.

➤ Es posible registrar un servidor con un nombre diferente al que tiene el servidor.

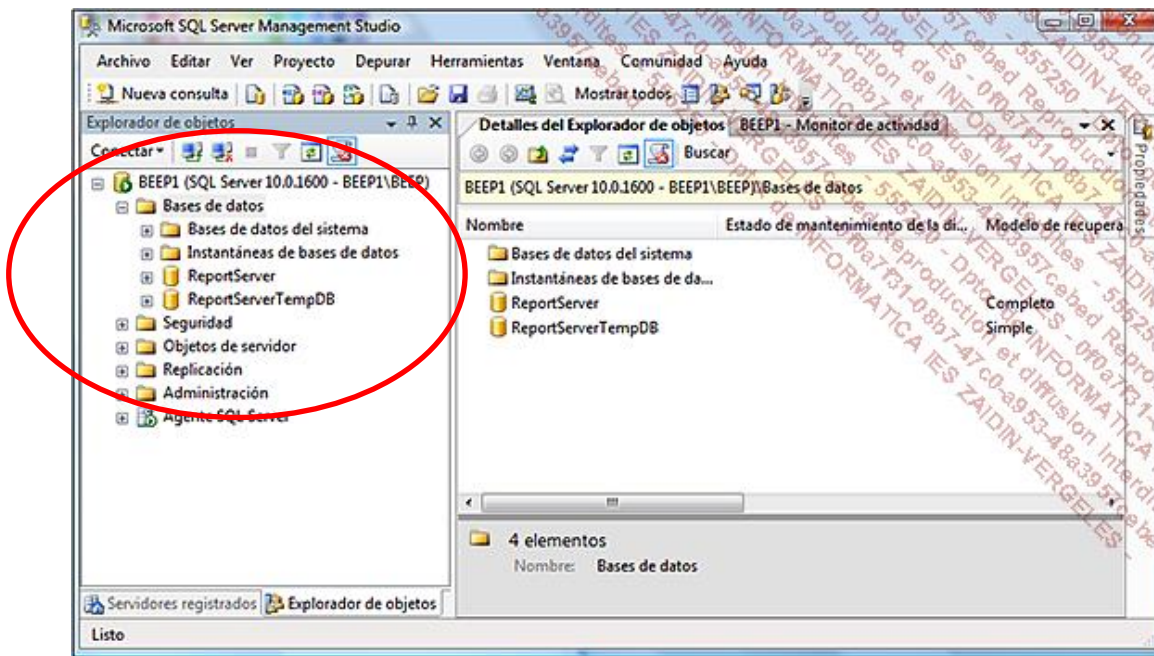
En lo referente a la segunda pestaña, esta permite fijar las opciones más avanzadas, como la base de datos predeterminada o el tipo de protocolo de red utilizado para establecer la conexión con el servidor.

➤ Por razones de seguridad, es preferible elegir siempre que sea posible el modo de autenticación Windows.

Al registrar el servidor, es posible seleccionar algunas opciones, tales como el tiempo de expiración de la conexión.



Para facilitar el trabajo del administrador, las bases de datos de sistema se agrupan en un directorio. De esta manera, son las bases de datos de los usuarios las que aparecen visibles en primer plano. Esta separación permite no sobrecargar la consola con las bases sistema que sólo son importantes para SQL Server.



El mismo tipo de separación se efectúa en las bases de datos entre las tablas de sistema y las tablas creadas por los usuarios. Son estas últimas las que contienen información y a las que el administrador debe dedicar todos sus esfuerzos.

### 3. Configuración del servidor

Antes de abrir completamente el acceso al servidor y permitir a los usuarios trabajar con él, conviene vigilar atentamente los dos puntos siguientes:

#### Contraseña del administrador

Esta preocupación afecta únicamente a los servidores que son configurados en modo de **seguridad mixta**. Si se ha seleccionado esta opción durante la instalación, es necesario asegurarse de que la contraseña del administrador SQL Server (sa) es suficientemente segura. Si no es el caso, es necesario modificarla.

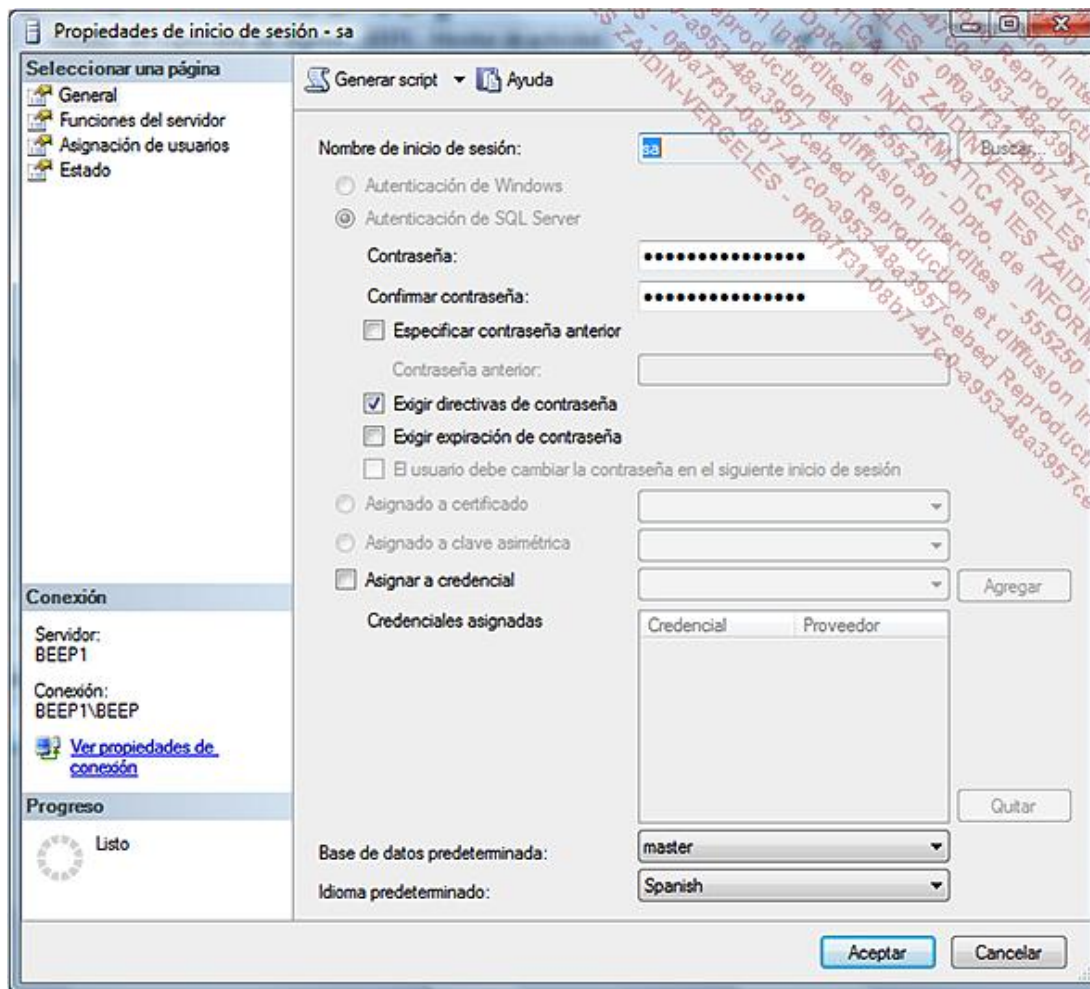
Durante la instalación de SQL Server, **se predefinen dos usuarios. El primero es el grupo local de los administradores** (utilizado con la seguridad Windows). **El segundo es el usuario sa**. Estos dos usuarios tienen los derechos de administrador del servidor SQL. El usuario **sa** se basa en la seguridad de SQL Server y su contraseña se solicita durante el procedimiento de instalación.

**Si en el momento de la instalación sólo se activa el modo de seguridad de Windows, la conexión sa no está activada.** Es necesario definir una contraseña segura antes de activar la conexión. Sin embargo, la conexión sólo podrá ser utilizada si el servidor está configurado en modo de seguridad mixta.

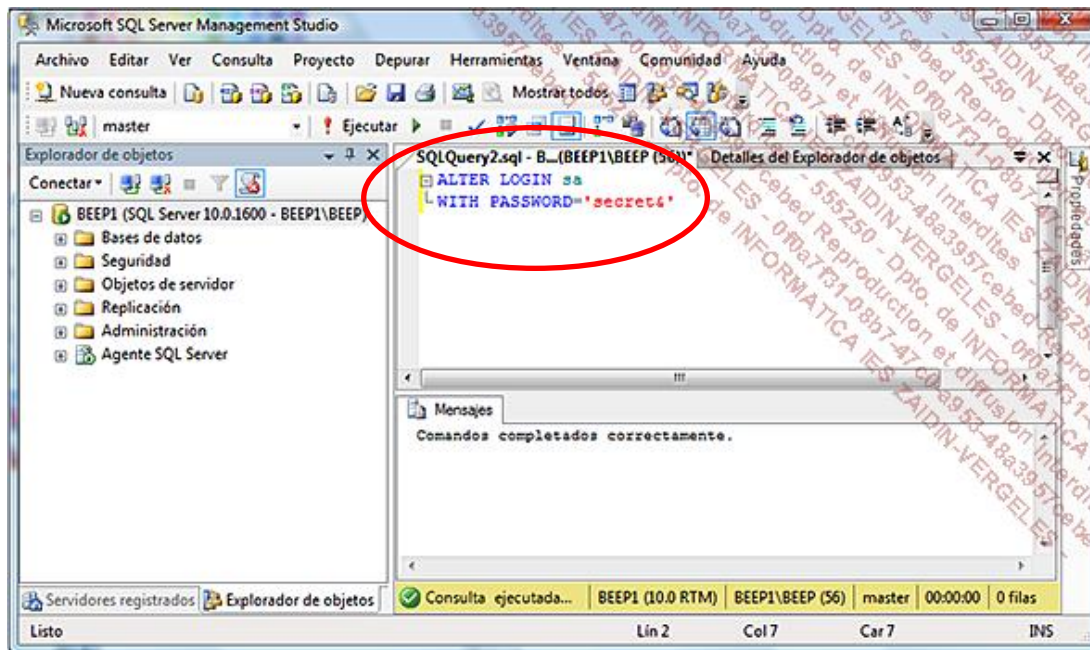
Se presentan dos posibilidades.

#### Por SQL Server Management Studio



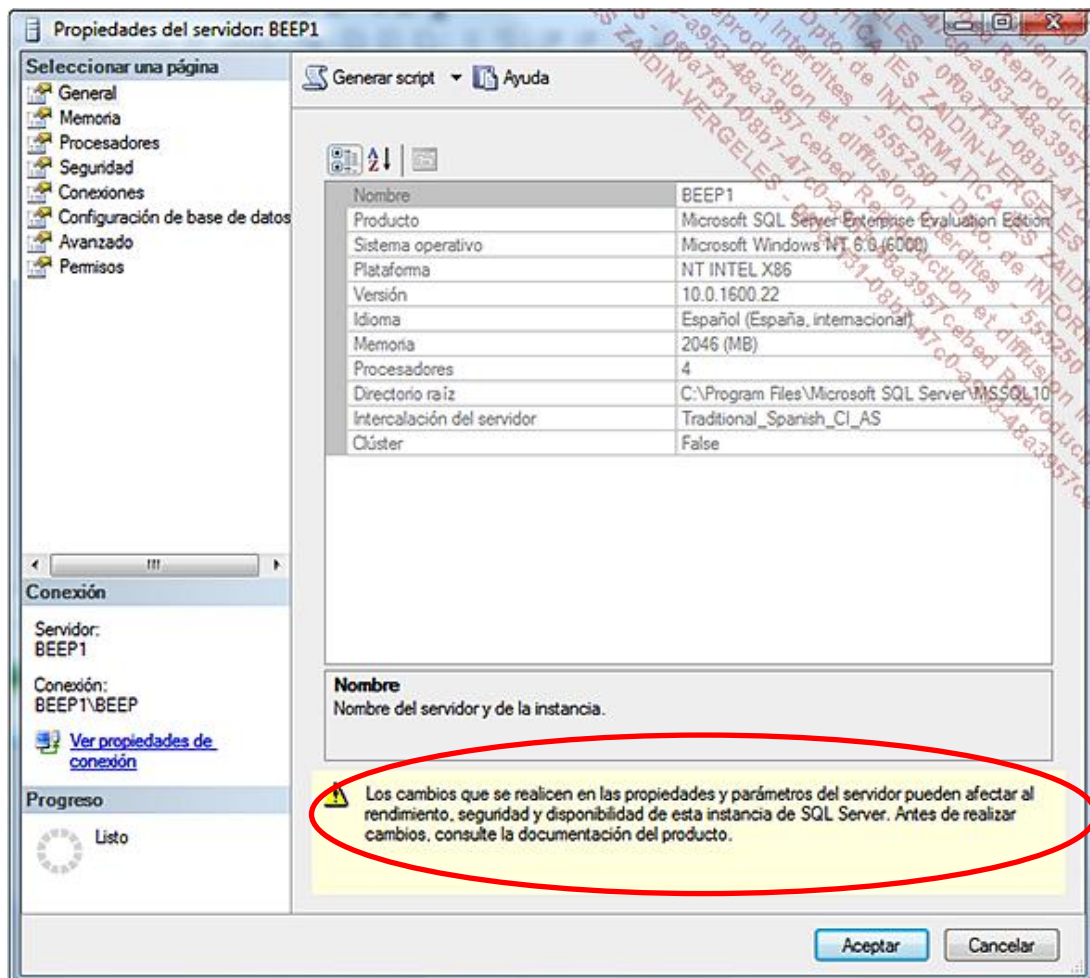


## Por Transact SQL

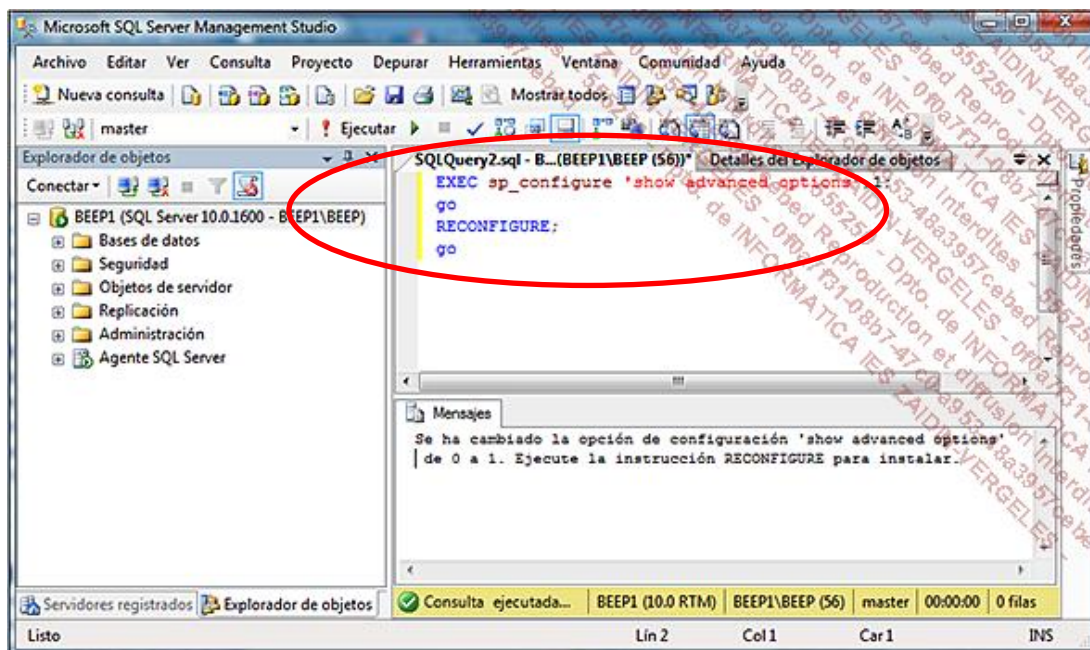


## Gestión de los recursos

Los recursos de la máquina se gestionan dinámicamente. Esta gestión automática de los recursos permite ofrecer las mejores funcionalidades posibles del servidor realizando un mínimo de tareas administrativas. Sin embargo, algunas veces puede ser interesante gestionar manualmente algunos recursos para realizar una optimización de la utilización de los recursos del servidor. Algunos de estos ajustes se pueden realizar a través de SQL Server Management Studio.



Para acceder a la totalidad de los parámetros del servidor, es necesario utilizar el procedimiento almacenado **sp\_configure**.



Si se modifica una opción con la ayuda del procedimiento **sp\_configure**, ésta no tendrá efecto hasta que se vuelva a iniciar el servidor SQL. Es posible aplicar la modificación de manera inmediata ejecutando el comando **RECONFIGURE WITH OVERRIDE**.



## 4. La gestión del proceso SQL Server

Para el sistema operativo, cada aplicación se ejecuta en forma de proceso. Cada proceso dispone de sus propios threads que se corresponden con las unidades de trabajo que el sistema operativo debe enviar al procesador. A un proceso le corresponde siempre al menos un thread.

Cada instancia SQL Server gestiona sus propios threads y también su sincronización sin pasar por el núcleo Windows. El objetivo de SQL Server es responder eficaz y rápidamente a las peticiones que aumentan la carga de manera brusca y repentina. Para estar siempre disponible, SQL Server gestiona su propio conjunto de threads, cuyo número máximo lo controla el parámetro **max worker threads**. Con el valor por defecto (0) SQL Server se encarga de gestionar él mismo este número de threads aunque también es posible fijar el número máximo de threads. Estos threads tienen como objetivo tratar las peticiones de los usuarios. Teniendo en cuenta que un usuario no trabaja el 100 % de su tiempo en el servidor, ya que debe leer o modificar los datos antes de enviar una nueva consulta, es preferible para SQL Server compartir un mismo thread entre varios usuarios.

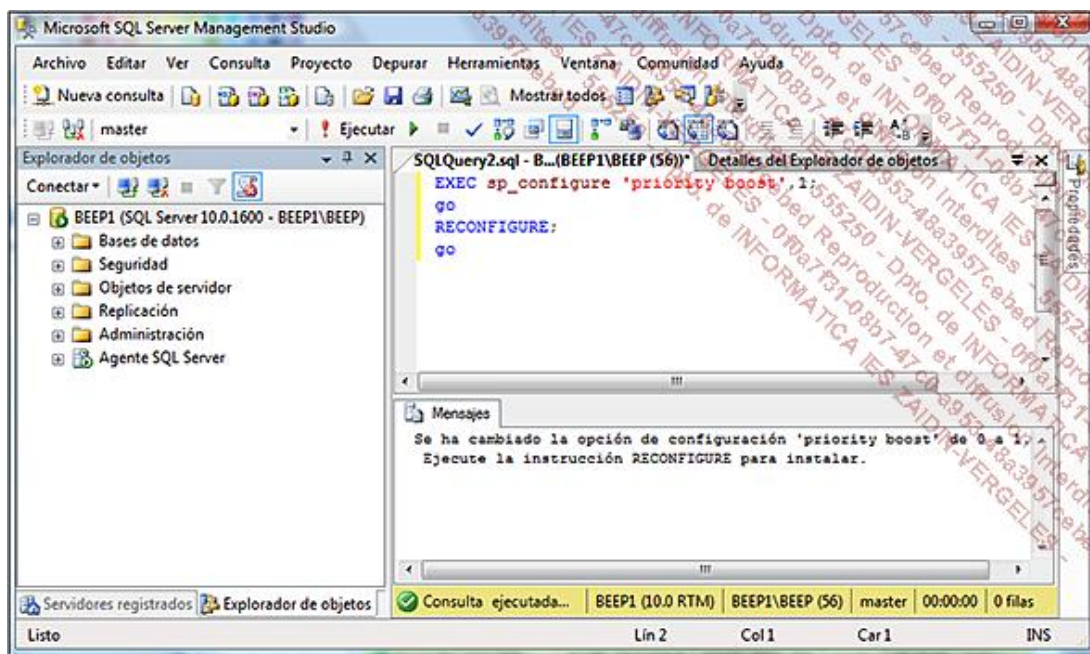


El valor máximo de este parámetro era 255 con SQL Server 2000. En el marco de una migración de servidor se recomienda posicionar este valor a 0.

~~Windows ofrece también gestionar los hilos que representan una unidad de trabajo más ligera que un thread. Es posible pedir a SQL Server trabajar con estos hilos en lugar de con threads. Esta parametrización se efectúa con la opción de configuración **lightweight pooling**. Sin embargo, la activación de esta opción no permite utilizar código CLR en SQL Server. Por otra parte, la activación de esta opción se traducirá en una ganancia significativa en términos de rendimiento únicamente para los servidores multiprocesadores con una tasa importante de utilización de los procesadores.~~

~~En el marco de una arquitectura multiprocesador y a menudo multiinstancia, es posible, con la ayuda de la opción de configuración **affinity**, especificar los procesadores que se van a utilizar para cada instancia. Esta opción contiene un valor binario donde cada bit representa la autorización (1) o la falta de la misma (0) para utilizar un procesador.~~

Por último, para programar la ejecución de los diferentes threads, Windows asigna a cada proceso una **prioridad** que va desde el 1 (el menos prioritario) a 31 (el más prioritario). Esta gestión de prioridad no afecta a los procesos de sistema. Por defecto, el proceso **SQL Server recibe un nivel de prioridad igual a 7**. Es decir, un proceso normal. Es posible dar una prioridad superior por medio de la opción de configuración **priority boost**. Esta opción puede ser especialmente interesante en el momento en que varias instancias SQL Server se ejecutan en el mismo puesto y se desea dar prioridad a una de ellas.



## 5. La gestión de la memoria

Por defecto, SQL Server gestiona dinámicamente la cantidad de memoria que necesita. Se recomienda conservar una gestión dinámica de la memoria, ya que permite un reparto óptimo de la memoria entre los diferentes procesos que se ejecutan en el servidor.

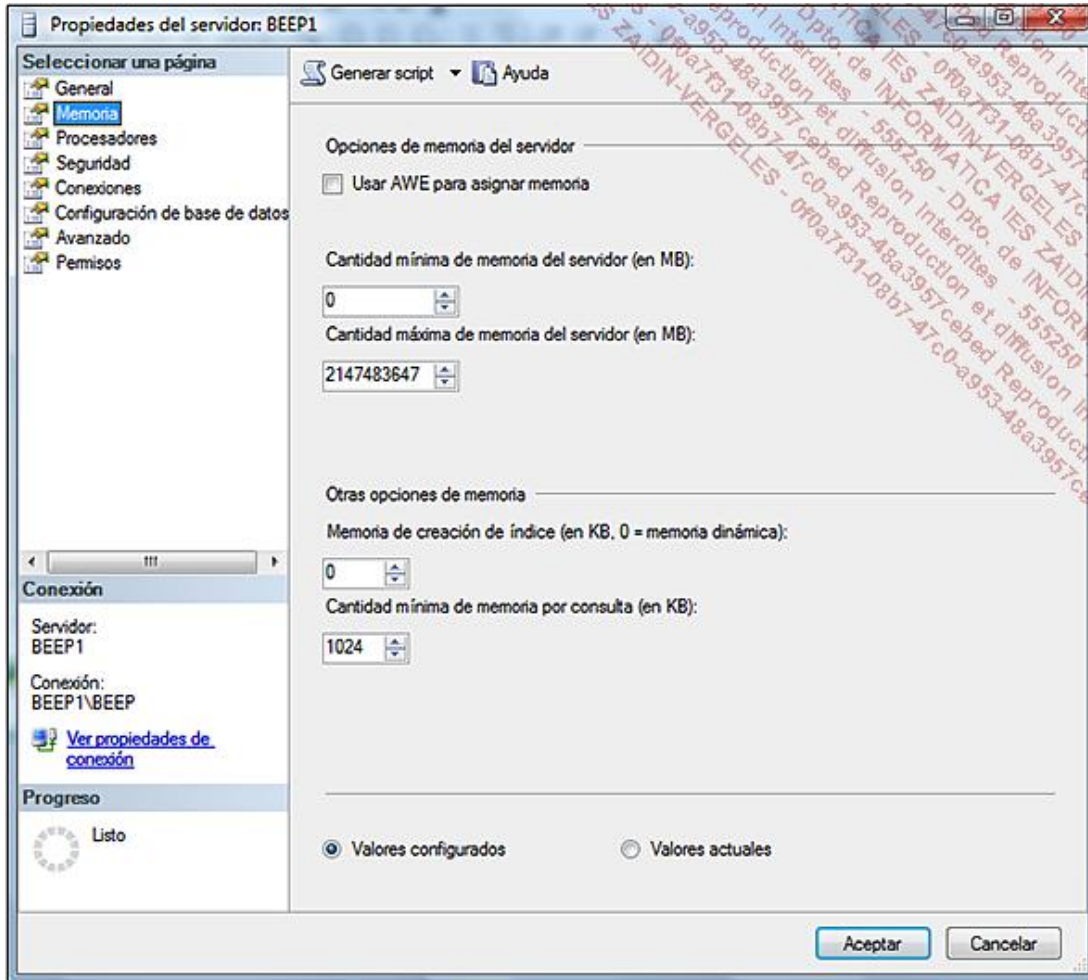
En plataforma de 32 bits, SQL Server es capaz de explotar las extensiones AWE (*Address Windowing Extensions*) con el objetivo de direccionar hasta 64 Gb de RAM.



Es importante que el servidor disponga de una cantidad de memoria suficiente ya que esto permite minimizar el número de lecturas físicas y favorecer las lecturas lógicas. Cuanto más numerosas sean está últimas, mejor es el tiempo respuesta del servidor. La ratio entre estos dos tipos de lecturas puede obtenerse mediante el analizador de rendimiento. Este punto se trata en el capítulo Optimización - Analizador de rendimiento (Monitor de sistema).

Para permitir la gestión dinámica de la cantidad de memoria utilizada, SQL Server se basa en el API (*Application Programming Interface*) de gestión de la memoria de Windows para adquirir el máximo de memoria sin privar al sistema de la cantidad de memoria que necesita.

Esta gestión dinámica puede limitarse utilizando los parámetros **min server memory** y **max server memory**. La instancia SQL Server, aunque sea poco utilizada, conservará siempre la cantidad de memoria especificada en **min server memory**. En caso de carga de trabajo, es posible obtener memoria, sin superar nunca el valor especificado en **max server memory**.



La memoria AWE se obtiene configurando el parámetro **awe enabled** de la manera siguiente:

