

Unidad III: Configuración y administración del espacio en disco

3.1 Estructuras lógicas de almacenamiento

Para la gestión del almacenamiento de una base de datos existen 4 conceptos bien definidos que deben ser conocidos para poder comprender la forma en la que se almacenan los datos. Vamos a ver la diferencia entre bloque, extensión, segmento y espacio de tablas.

Bloques: Se tratan de la unidad más pequeña. Generalmente debe múltiple del tamaño de bloque del sistema operativo, ya que es la unidad mínima que va a pedir Oracle al sistema operativo. Si no fuera múltiple del bloque del sistema se añadiría un trabajo extra ya que el sistema debería obtener más datos de los estrictamente necesarios. Se especifica mediante `DB_BLOCK_SIZE`

Extensiones: Se forma con uno o más bloques. Cuando se aumenta tamaño de un objeto se usa una extensión para incrementar el espacio.

Segmentos: Grupo de extensiones que forman un objeto de la base de datos, como por ejemplo una tabla o un índice.

Espacio de tablas: Formado por uno o más datafiles, cada datafile solo puede pertenecer a un determinado tablespace

En general, el almacenamiento de los objetos de la base de datos (tablas e índices fundamentalmente) no se realiza sobre el archivo o archivos físicos de la base de datos, sino que se hace a través de estructuras lógicas de almacenamiento que tienen por debajo a esos archivos físicos, y que independizan por tanto las sentencias de creación de objetos de las estructuras físicas de almacenamiento. Esto es útil porque permite que a esos "espacios de objetos " les sean asociados nuevos dispositivos físicos (es decir, más espacio en disco) de forma dinámica cuando la base de datos crece de tamaño más de lo previsto. Posibilita además otra serie de operaciones como las siguientes:

- Asignar cuotas específicas de espacio a usuarios de la base de datos.

- Controlar la disponibilidad de los datos de la base de datos, poniendo fuera de uso alguno de esos espacios de tablas individualmente.
- Realizar copias de seguridad o recuperaciones parciales de la base de datos.
- Reservar espacio para almacenamiento de datos de forma cooperativa entre distintos dispositivos.

El administrador de la base de datos puede crear o borrar nuevos espacios lógicos de objetos, añadir o eliminar ficheros físicos de soporte, utilizados como espacio temporal de trabajo, definir parámetros de almacenamiento para objetos destinados a ese espacio de datos, todos los gestores relacionales que venimos introduciendo como ejemplos siguen esta filosofía. En el caso de Oracle, sobre los ficheros físicos de datos (datafiles) se definen los tablespaces. Por lo tanto, una base de datos Oracle se compone lógicamente de tablespaces, y físicamente de datafiles. Su creación es sencilla, con la sentencia `CREATE TABLESPACE usuarios DATAFILE 'data1.ora' SIZE 50M`

También es sencillo ampliar el espacio destinado a un tablespace utilizando el comando `ALTER TABLESPACE`:

```
ALTER TABLESPACE usuarios ADD DATAFILE 'data2.ora' SIZE 25M
```

Para hacer más grande una base de datos, las opciones disponibles son tres:

1. Añadir más ficheros de datos a tablespaces existentes, como acabamos de ver.
2. Crear tablespaces nuevos con sus correspondientes ficheros de datos asociados.
3. Permitir que el tablespace crezca dinámicamente en función de sus necesidades de espacio. Para ello existe una sintaxis asociada a las sentencias `CREATE` o `ALTER DATABASE/TABLESPACE` que permite añadir ficheros con propiedades de extensión dinámica, como vemos en el siguiente ejemplo:

```
ALTER DATABASE
```

```
1) DATAFILE 'data2.ora' AUTOEXTEND ON NEXT 20M MAXSIZE 1000M
```

que activa la propiedad de autoextensión (`AUTOEXTEND = ON`) y hace que el fichero pueda ir adquiriendo extensiones de 20M (20 megabytes) conforme va siendo necesario, con un tamaño máximo de 1000M.

Cada base de datos contiene un tablespace llamado SYSTEM que es creado automáticamente al crear la base de datos. Contiene las tablas del diccionario de datos para la base de datos en cuestión. Es recomendable no cargar datos de usuario en SYSTEM, para dejarlos como espacio de objetos del sistema. Si además los datos de usuario están en tablespaces sitos en otros dispositivos, el rendimiento mejorará porque las tablas del diccionario de datos se acceden frecuentemente y por lo tanto son un cuello de botella potencial desde el punto de vista del acceso a disco. A la hora de estimar el espacio necesario para el tablespace sys-nsm hay que tener en cuenta que las unidades de programación PL-SQL (entorno de programación SQL proporcionado por Oracle) almacenadas en la base de datos (procedimientos, paquetes, disparos y funciones) almacenan sus datos en SYSTEM.

De acuerdo con lo comentado anteriormente, tablas e índices se ubicarán en el tablespace indicado en el momento de su creación con la correspondiente sentencia CREATE. Si no se dice nada, se situarán en el tablespace por defecto asociado al usuario creador.

3.1.1. Definición de espacio de almacenamiento

Almacenamiento físico de bases de datos

La mayoría de las bases de datos se almacenan en las llamadas memorias secundarias, especialmente discos duros, aunque, en principio, pueden emplearse también discos ópticos, memorias flash, etc.

Las razones por las cuales las bases de datos se almacenan en memorias secundarias son:

- * En general, las bases de datos son demasiado grandes para entrar en la memoria primaria.
- * La memoria secundaria suele ser más barata que la memoria primaria (aunque esta última tiene mayor velocidad).

* La memoria secundaria es más útil para el almacenamiento de datos permanente, puesto que la memoria primaria es volátil.

En cuanto al respaldo de las bases de datos (ver backup), suelen emplearse tanto discos duros, como cintas magnéticas, discos ópticos o similares.

Técnicas de almacenamiento y recuperación de bases de datos

Las técnicas empleadas para almacenar bases de datos son sumamente importantes para la velocidad de acceso y recuperación de datos. Las técnicas dependen del tipo de almacenamiento, el uso que se le da o se le dará a la base de datos, la estructura de la misma, el SGBD empleado, etc.

Esta dependencia no significa necesariamente que haya que cambiar la estructura de la base de datos si se cambian las técnicas empleadas. Las técnicas de almacenamiento son independientes de la base de datos, pero, de todas maneras, las mejores técnicas muchas veces pueden determinarse viendo la estructura de la base de datos, entre otras características.

Los encargados de elegir estas técnicas son los diseñadores y administradores de bases de datos, y dependen también de las capacidades del SGBD. En general, el SGBD ofrece diferentes opciones y técnicas para organizar los datos.

La idea es que los encargados de la base de datos encuentren las técnicas idóneas, o sea, aquellas que permitan la mayor velocidad posible de acceso a los datos. Una mala decisión en esta área puede resultar en una menor velocidad de acceso a la base de datos, o en un uso excesivo del espacio de almacenamiento, o incluso, puede aumentar la velocidad de consulta de una base de datos, pero disminuir la velocidad de actualización de la misma.

El almacenamiento en archivos de las bases de datos

Las bases de datos se almacenan en ficheros o archivos. Existen diferentes formas de organizaciones primarias de archivos que determinan la forma en que

los registros de unos archivos se colocan físicamente en el disco y, por lo tanto, cómo se accede a éstos.

Las distintas formas de organizaciones primarias de archivos son:

- * Archivos de montículos (o no ordenados): esta técnica coloca los registros en el disco sin un orden específico, añadiendo nuevos registros al final del archivo.
- * Archivos ordenados (o secuenciales): mantiene el orden de los registros con respecto a algún valor de algún campo (clave de ordenación).
- * Archivos de direccionamiento calculado: utilizan una función de direccionamiento calculado aplicada a un campo específico para determinar la colocación de los registros en disco.
- * Árboles B: se vale de la estructura de árbol para las colocaciones de registros.

Existe una segunda forma de acceder a los datos llamada organización secundaria o estructura de acceso auxiliar. Estas permiten que los accesos a los registros de un archivo basado en campos alternativos, sean más eficientes que los que han sido utilizados para la organización primaria de archivos.

3.1.2. Definición y creación del espacio asignado para cada base de datos

Las bases de datos se almacenan en ficheros o archivos. Existen diferentes formas de organizaciones primarias de archivos que determinan la forma en que los registros de un archivo se colocan físicamente en el disco y, por lo tanto, cómo se accede a éstos.

Las distintas formas de organizaciones primarias de archivos son:

- Archivos de montículos (o no ordenados): esta técnica coloca los registros en el disco sin un orden específico, añadiendo nuevos registros al final del archivo.
- Archivos ordenados (o secuenciales): mantiene el orden de los registros con respecto a algún valor de algún campo (clave de ordenación).
- Archivos de direccionamiento calculado: utilizan una función de direccionamiento calculado aplicada a un campo específico para determinar la colocación de los registros en disco.
- Árboles B: se vale de la estructura de árbol para las colocaciones de registros.

Existe una segunda forma de acceder a los datos llamada organización secundaria o estructura de acceso auxiliar. Estas permiten que los accesos a los registros de un archivo basado en campos alternativos, sean más eficientes que los que han sido utilizados para la organización primaria de archivos.

El DBMS asigna espacio de almacenamiento a las bases de datos cuando los usuarios introducen `create database` o `alter database`. El primero de los comandos puede especificar uno o más dispositivos de base de datos, junto con la cantidad de espacio en cada uno de ellos que será asignado a la nueva base de datos.

Si se utiliza la palabra clave `default` o se omite completamente la cláusula `on`, el DBMS pone la base de datos en uno o más de los dispositivos predeterminados de base de datos especificados en `master..sysdevices`

Para especificar un tamaño (en este ejemplo, 4MB) para una base de datos que se va a almacenar en una ubicación predeterminada, utilice `on default = size` de esta forma:

```
create database newpubs on default = 4
```

Para situar la base de datos en dispositivos específicos, dé el nombre del dispositivo o dispositivos en que desea almacenarla. Como la sintaxis indica, puede solicitar que se almacene en más de un dispositivo de base de datos, con una cantidad de espacio diferente en cada uno. Todos los dispositivos mencionados en `create database` deben estar enumerados en `sysdevices`. En otras palabras, deben haberse inicializado con `disk init`.

La instrucción siguiente crea la base de datos newdb y asigna 3MB en mydata y 2MB en newdata . Como en el ejemplo anterior, la base de datos y el diario de transacciones no se separan:

```
create database newdb on mydata = 3, newdata = 2
```

Warning! A menos que cree una base de datos pequeña o que no sea crucial, sitúe siempre el diario en un dispositivo de base de datos aparte.

Si la cantidad de espacio solicitada a un dispositivo específico de base de datos no está disponible, el DBMS crea la base de datos con tanto espacio como sea posible en cada dispositivo y muestra un mensaje informando el espacio asignado en cada uno. (Esto no se considera un error.) Si hay menos espacio del mínimo necesario para una base de datos en el dispositivo especificado (o en el predeterminado, si no se especifica un nombre), el comando create database falla.

3.1.3. Bitácoras

Una partición de disco, en mantenimiento, es el nombre genérico que recibe cada división presente en una sola unidad física de almacenamiento de datos. Toda partición tiene su propio sistema de archivos (formato); generalmente, casi cualquier sistema operativo interpreta, utiliza y manipula cada partición como un disco físico independiente, a pesar de que dichas particiones estén en un solo disco físico.

Una partición de un disco duro es una división lógica en una unidad de almacenamiento (por ejemplo un disco duro o unidad flash), en la cual se alojan y organizan los archivos mediante un sistema de archivos. Existen distintos esquemas de particiones para la distribución de particiones en un disco. Los más conocidos y difundidos son MBR (Master Boot Record) y GPT (GUID Partition Table). Las particiones, para poder contener datos tienen que poseer un sistema de archivos. El espacio no asignado en un disco no es una partición, por lo tanto no puede tener un sistema de archivos. Existen múltiples sistemas de archivos con

diferentes capacidades: como FAT, NTFS, FAT32, EXT2, EXT3, EXT4, Btrfs, FedFS, ReiserFS, Reiser4 u otros. Los discos ópticos (DVD, CD) utilizan otro tipo de particiones llamada UDF (Universal Disc Format) Formato de Disco Universal por sus siglas en inglés, el cual permite agregar archivos y carpetas y es por ello que es usado por la mayoría de software de escritura por paquetes, conocidos como programas de grabación de unidades ópticas. Este sistema de archivos es obligatorio en las unidades de (DVD) pero también se admiten en algunos (CD)

En Windows, las particiones reconocidas son identificadas con una letra seguida por un signo de doble punto (p.ej. C:\). prácticamente todo tipo de discos magnéticos y memorias flash (como pendrives) pueden particionarse. En sistemas UNIX y UNIX-like las particiones de datos son montadas en un mismo y único árbol jerárquico, en el cual se montan a través de una carpeta, proceso que sólo el superusuario (root) puede realizar.

- Algunos sistemas de archivos (p.e. versiones antiguas de sistemas FAT de Microsoft) tienen tamaños máximos más pequeños que los que el tamaño que proporciona un disco, siendo necesaria una partición de tamaño pequeño, para que sea posible el adecuado funcionamiento de este antiguo sistema de archivos.
- Se puede guardar una copia de seguridad de los datos del usuario en otra partición del mismo disco, para evitar la pérdida de información importante. Esto es similar a un RAID, excepto en que está en el mismo disco.
- En algunos sistemas operativos aconsejan más de una partición para funcionar, como por ejemplo, la partición de intercambio (swap) en los sistemas operativos basados en Linux.
- A menudo, dos sistemas operativos no pueden coexistir en la misma partición, o usar diferentes formatos de disco “nativo”. La unidad se particiona para diferentes sistemas operativos.

□ Uno de los principales usos que se le suele dar a las particiones (principalmente a la extendida) es la de almacenar toda la información del usuario (entiéndase música, fotos, vídeos, documentos), para que al momento de reinstalar algún sistema operativo se formatee únicamente la unidad que lo contiene sin perder el resto de la información del usuario. A lo largo de los años han aparecido numerosos sistemas de particionamiento, para casi todas las arquitecturas de ordenadores existentes. Muchos son relativamente transparentes y permiten la manipulación conveniente de las particiones de disco; algunos, sin embargo, son obsoletos. Las tablas de particiones (MBR) sólo admiten hasta 2,2 TB por partición. Dado que sólo soportan 4 particiones primarias, el tamaño máximo admisible para un disco duro sería de 8,8 TB (el resto de capacidad no se podría utilizar). Como la arquitectura IBM PC es muy común, las tablas de partición probablemente subsistirán cierto tiempo. Sin embargo, existe un proyecto de Intel llamado Extensible Firmware Initiative (EFI) con el sistema GPT, que soporta teóricamente hasta 9,4 ZB.

3.1.4 Particiones

- **Partición:** Parte del disco duro. En muchos casos puede ser el espacio completo del disco duro.
- El almacenamiento en un disco duro ha sido desde hace mucho tiempo el método normal de almacenamiento para las computadoras modernas, desde las computadoras centrales a los equipos de escritorio, y es poco probable que cambie, considerando incluso las opciones de almacenamiento más ricas que se han añadido en Windows 2000.
- *RAID (Array redundante de discos independientes) es un término usado para describir una técnica que ha aparecido como una solución final exotérica para una suposición normal en la mayoría de los servidores. Hace siete u ocho años, RAID no era un término demasiado conocido, aunque el artículo original en el que se definía RAID se escribió en 1988. Hasta hace poco, la mayoría de los sistemas de servidor dependían de discos duros*

caros y de gran calidad, con una frecuente copia de seguridad. Las copias de seguridad todavía son críticas, pero ahora se puede utilizar uno u otro formato de RAID para proporcionar una protección sustancial frente a fallos de disco duro. Aún más, esta protección cuesta mucho menos de lo que costaban aquellos enormes discos de servidor. RAID se puede implementar a nivel de software o de hardware. Cuando se implementa a nivel de hardware, el vendedor del hardware proporciona una interfaz de administración para los arrays y las unidades para admitir que pueda trabajar con los diversos sistemas operativos. Windows 2000 implementa los niveles 0, 1 y 5 de RAID en software.

3.1.5 Espacios privados

Un «espacio privado» permite que los administradores y redactores gestionen el conjunto de datos del sitio. Algunas bases de datos tienen estos espacios privados llamados comúnmente paneles de control, que son formularios que aparecen al abrir la base de datos.

Los paneles de control sirven de "puerta principal" o "recibidor" de una base de datos en el sentido de que dirigen a las personas hacia determinadas tareas, como introducir o buscar datos. Sirven también para mantener alejados a los usuarios de las tablas que contienen los datos en tiempo real.

Cuando reciba una base de datos, debe adentrarse más allá del panel de control para averiguar cómo están estructurados los datos, pero merece la pena echar un vistazo inicial al panel de control. Le puede ofrecer algún indicio sobre las tareas que el diseñador de la base de datos consideró que realizarían los usuarios habitualmente con los datos.

Puede hacer clic en los vínculos del panel de control para ver qué objetos, como formularios e informes, abren.

Server: localhost Database: base2 Table: persons

Structure Browse SQL Search Insert Export Operations Empty Drop

table personnes aaa; InnoDB free: 343040 kB; ('town_code') REFER 'base2/towns ('town_code'); ('country_code') REFER 'base2/countries ('country_code')

Field	Type	Attributes	Null	Default	Extra	Action
<input type="checkbox"/> id	tinyint(4)		No		auto_increment	
<input type="checkbox"/> person_name	varchar(200)		No			
<input type="checkbox"/> town_code	varchar(5)		Yes	0		
<input type="checkbox"/> country_code	char(1)		Yes	NULL		
<input type="checkbox"/> car_code	char(3)		No			

Check All / Uncheck All With selected:

Print view Relation view Propose table structure

Add field(s) At End of Table At Beginning of Table After id Go

Indexes:	Keyname	Type	Cardinality	Action	Field
PRIMARY	PRIMARY	2		id	
town_code	INDEX	2		town_code	
country_code	INDEX	2		country_code	
pays-ville	INDEX	2		country_code town_code	

Space usage: Type Usage Data 16,384 Bytes Index 49,152 Bytes Total 65,536 Bytes

Row Statistic: Statements Value Format dynamic Next Autoindex 5

More than one INDEX key was created for column 'country_code'

Create an index on 1 columns Go

3.1.6 Espacios para objetos

Los DBMS se basan en archivos para almacenar datos, y estos archivos, o conjuntos de datos, residen en medios de almacenamiento, o dispositivos. Una buena parte del trabajo del DBA implicará la planificación para el almacenamiento real de la base de datos.

Algunas tecnologías de almacenamiento son más adecuadas que otras. Sin embargo, la naturaleza mecánica de la unidad de disco los hace más vulnerables al fracaso de los componentes de otro equipo. Además, las formas en que las unidades de disco son utilizados por las bases de datos pueden hacer que la gestión del almacenamiento impredecibles, como la barra lateral "Modern DBMS de uso de disco" Puede usarse RAID para mejorar la seguridad de los datos.

Para aplicaciones de misión crítica la integridad de los datos puede ser más importante que la disponibilidad de datos. Si el soporte es poco fiable y un fallo de las causas de corrupción de datos, los datos perdidos puede ser más de un problema que el tiempo de inactividad. Es imperativo, por tanto, que las soluciones de almacenamiento de base de datos para protegerlos a toda costa. La

recuperación de datos desde medios de almacenamiento lleva mucho más tiempo en completarse que la recuperación de datos desde la memoria caché o la memoria.

El rendimiento de la base de datos depende de la entrada y salida a disco. La cantidad de datos almacenados es mayor que nunca antes, y los datos se almacenan por más tiempo.

Algunos DBMS permiten al tamaño de los archivos temporales de expandirse y contraerse de forma automática. Dependiendo del tipo y la naturaleza de las operaciones de base de datos en proceso, esta fluctuación puede provocar picos de uso del disco

El crecimiento de la capacidad de almacenamiento aumenta aún más la complejidad de la gestión de datos y bases de datos. Muchas organizaciones están implementando nuevas tecnologías de almacenamiento, tales como almacenamiento en red (NAS) y redes de área de almacenamiento (SAN), para ayudar a controlar la cantidad cada vez mayor de almacenamiento necesario para los usos modernos. La gestión del almacenamiento en el entorno dinámico de hoy es una tarea difícil DBA.

Hay muchos problemas de almacenamiento que deben ser resueltos antes de que un DBA pueda crear una base de datos. Uno de los temas más importantes es la cantidad de espacio para permitir la base de datos.

El cálculo espacial debe tener en cuenta no sólo tablas, índices, sino también, y dependiendo del DBMS, el registro de transacciones. Cada una de estas entidades probablemente requerirá un archivo separado o conjunto de datos, para el almacenamiento persistente.

3.2. Segmentos

Un segment es aquel espacio reservado por la base de datos, dentro de un datafile, para ser utilizado por un solo objeto. Así una tabla (o cualquier otro objeto) está dentro de su segmento, y nunca podrá salir de el, ya que si la tabla crece, el segmento tambien crece con ella. Físicamente todo objeto en base de

datos no es más que un segmento dentro de un datafile. Se puede decir que, un segmento es a un objeto de base de datos, lo que un datafile a un tablespace; el segmento es la representación física del objeto en base de datos (el objeto es solo una definición lógica).

Los segmentos son los equivalentes físicos de los objetos que almacenan datos. El uso efectivo de los segmentos requiere que el DBA conozca los objetos, que utiliza una aplicación, cómo los datos son introducidos en esos objetos y el modo en que serán recuperados.

Un segmento está constituido por secciones llamadas extensiones, que son conjuntos contiguos de bloques Oracle. Una vez que una extensión existente en un segmento no puede almacenar más datos, el segmento obtendrá del espacio de tabla otra extensión. Este proceso de extensión continuará hasta que no quede más espacio disponible en los ficheros del espacio de tablas, o hasta que se alcance un número máximo de extensiones por segmento.

Existen 5 tipos de segmento:

- De datos.
- De índices.
- De rollback.
- Temporales.
- De bootstrap.

3.3. Memoria Compartida

Memoria compartida distribuida

Los sistemas de memoria compartida distribuida (DSM) representan la creación híbrida de dos tipos de computación paralelos: la memoria distribuida en sistemas multiprocesador y los sistemas distribuidos. Ellos proveen la abstracción de

memoria compartida en sistemas con memorias distribuidas físicamente y consecuentemente combinan las mejores características de ambos enfoques. Debido a esto, el concepto de memoria compartida distribuida es reconocido como uno de los enfoques más atractivos para la creación de sistemas escalables, de alto rendimiento de sistemas multiprocesador.

Memoria compartida basada en páginas

El esquema de DSM propone un espacio de direcciones de memoria virtual que integra la memoria de todas las computadoras del sistema, y su uso se realiza mediante paginación. Las páginas quedan restringidas a estar necesariamente en un único nodo. Cuando un programa intenta acceder a una posición virtual de memoria, se comprueba si esa página se encuentra de forma local. Si no se encuentra, se provoca un fallo de página, y el sistema operativo solicita la página al resto de nodos. El sistema funciona de forma análoga al sistema de memoria virtual tradicional, pero en este caso los fallos de página se propagan al resto de ordenadores, hasta que la petición llega al nodo que tiene la página virtual solicitada en su memoria local. A primera vista este sistema parece más eficiente que el acceso a la memoria virtual en disco, pero en la realidad ha mostrado ser un sistema demasiado lento en ciertas aplicaciones, ya que provoca un tráfico de páginas excesivo.

Una mejora dirigida a mejorar el rendimiento sugiere dividir el espacio de direcciones en una zona local y privada y una zona de memoria compartida, que se usará únicamente por procesos que necesiten compartir datos. Esta abstracción se acerca a la idea de programación mediante la declaración explícita de datos públicos y privados, y minimiza el envío de información, ya que sólo se enviarán los datos que realmente vayan a compartirse.

Memoria compartida basada en objetos

Una alternativa al uso de páginas es tomar el objeto como base de la transferencia de memoria. Aunque el control de la memoria resulta más complejo, el resultado

es al mismo tiempo modular y flexible, y la sincronización y el acceso se pueden integrar limpiamente. Otra de las restricciones de este modelo es que todos los accesos a los objetos compartidos han de realizarse mediante llamadas a los métodos de los objetos, con lo que no se admiten programas no modulares y se consideran incompatibles.

3.4. Instancias múltiples

Una instancia de Motor de base de datos funciona como un servicio que controla todas las solicitudes de aplicación para trabajar con datos de cualquiera de las bases de datos administradas por dicha instancia. Es el destino de las solicitudes de conexión (inicios de sesión) de aplicaciones. La conexión se ejecuta en una conexión de red si la aplicación y la instancia están en equipos independientes. Si la aplicación y la instancia están en el mismo equipo, la conexión de SQL Server se puede ejecutar como una conexión de red o una conexión en memoria. Cuando una conexión se ha completado, una aplicación envía instrucciones Transact-SQL a través de la conexión hasta la instancia. La instancia resuelve las instrucciones de Transact-SQL en operaciones con los datos y objetos de las bases de datos y, si se han concedido los permisos necesarios a las credenciales de inicio de sesión, realiza el trabajo. Los datos recuperados se devuelven a la aplicación, junto con cualesquiera mensajes como errores.

Puede ejecutar múltiples instancias de Motor de base de datos en un equipo. Una instancia puede ser la instancia predeterminada. La instancia predeterminada no tiene nombre. Si una solicitud de conexión especifica solo el nombre del equipo, se establece la conexión a la instancia predeterminada. Una instancia con nombre es una instancia en la que se especifica un nombre de instancia al instalar la instancia. Una solicitud de conexión debe especificar el nombre del equipo y el nombre de instancia para conectar a la instancia. No hay ningún requisito para instalar una instancia predeterminada; todas las instancias que se ejecutan en un equipo pueden ser instancias con nombre.