

Estructura de almacenamiento y archivos de datos

La estructura de almacenamiento y archivos de datos se refiere a cómo se organizan, guardan y acceden los datos en un sistema informático. El objetivo es optimizar el rendimiento, seguridad y disponibilidad de la información.

Tipo de almacenamiento	Descripción	Tipo de archivo o extensión usada	Ejemplo de sistema	Qué se almacena
Primario (Memoria principal)	Es la memoria temporal donde se cargan los datos y programas mientras se ejecutan. Es volátil (se borra al apagar el equipo).	No usa archivos físicos permanentes (usa direcciones de memoria RAM).	RAM, Caché, Memoria del CPU	Variables, registros temporales, índices en ejecución, cachés de consulta.
Secundario (Almacenamiento permanente)	Es el lugar donde los datos se guardan de forma persistente. Los sistemas gestores de bases de datos utilizan archivos físicos específicos.	.mdf (archivo de datos principal), .ndf (archivos secundarios), .ldf (archivo de log o transacciones) .	SQL Server, MySQL (usa .ibd y .frm)	Tablas, índices, procedimientos, registros, logs de transacciones.
Terciario (Almacenamiento en la nube o respaldo)	Es un almacenamiento remoto, usado para copias de seguridad o acceso distribuido.	.bak (archivo de respaldo de BD), .zip, .gz, .sql exportado.	Azure Blob, AWS S3, Google Cloud Storage	Copias de seguridad, exportaciones de BD, respaldos automáticos.

-- Ver propiedades generales

```
exec sp_helpdb 'QhatuPeru';
```

--Para ver toda la base de datos creadas--

```
SELECT name FROM sys.databases;
```

name	db_size	owner	dbid	created	status	compatibility_level
QhatuPERU	16.00 MB	LAPTOP-SOMH2M65\Megan	5	Nov 4 2025	Status=ONLINE, Updateability=READ_WRITE, Userdoc...	100

name	fileid	filename	filegroup	size	maxsize	growth	usage
QhatuPERU	1	C:\Program Files\Microsoft SQL Server\MSSQL16.MSS...	PRIMARY	8192 KB	Unlimited	65536 KB	data only
QhatuPERU_log	2	C:\Program Files\Microsoft SQL Server\MSSQL16.MSS...	NULL	8192 KB	2147483648 KB	65536 KB	log only

Propiedades y configuraciones de bases de datos

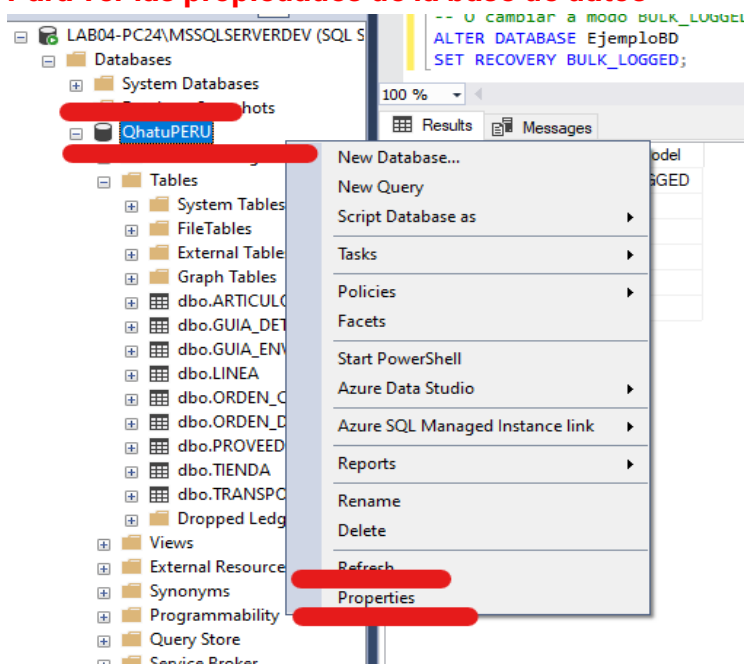
Propiedades de las bases de datos

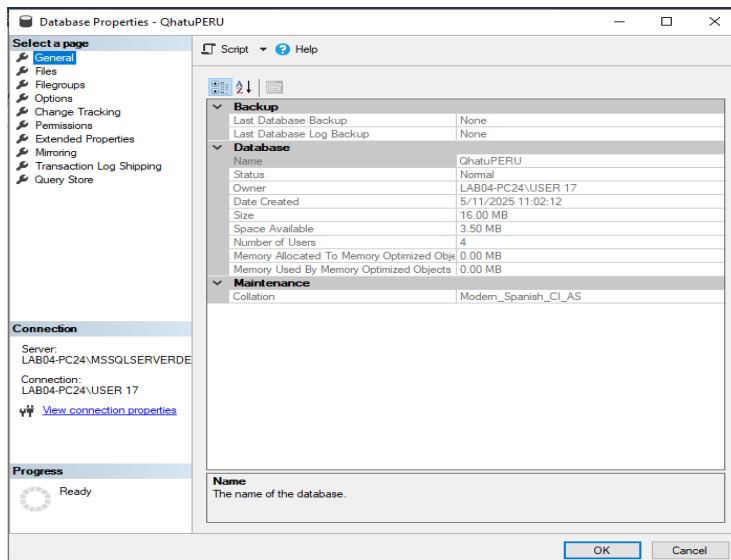
- **Independencia de datos:** Permite modificar la estructura física o lógica sin afectar los programas de aplicación.
- **Integridad de datos:** Asegura que los datos sean precisos y no haya duplicados ni datos faltantes.
- **Seguridad:** Protege la información y gestiona los permisos de acceso para los usuarios.
- **Acceso concurrente:** Permite que múltiples usuarios accedan a la base de datos simultáneamente sin interferir entre sí.
- **Control de redundancia:** Evita la duplicación de datos para ahorrar espacio y mantener la consistencia.

Propiedades ACID de las transacciones

- **Atomicidad:** Una transacción se completa totalmente o no se realiza ninguna parte.
- **Consistencia:** Una transacción lleva la base de datos de un estado válido a otro estado válido.
- **Aislamiento:** Las transacciones concurrentes no se interfieren entre sí.
- **Durabilidad:** Una vez que una transacción es confirmada, sus cambios son permanentes.

Para ver las propiedades de la base de datos



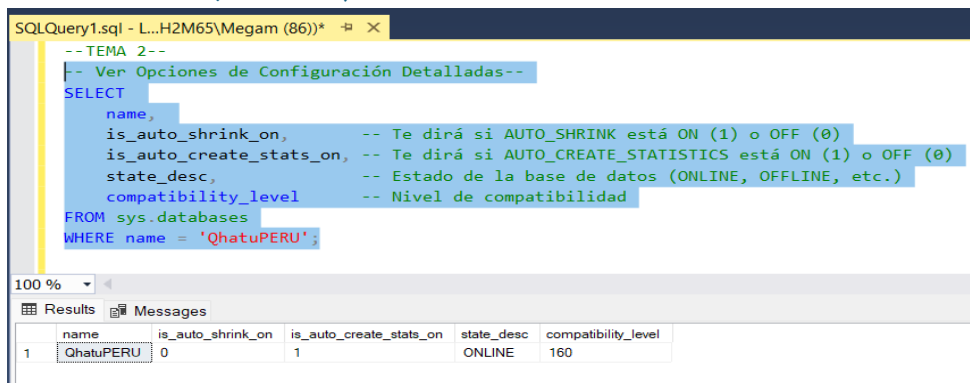


Configuraciones de bases de datos

- **Configuraciones de rendimiento:** Ajustes que controlan cómo el sistema maneja consultas y carga de trabajo, como la cantidad de memoria asignada.
- **Configuraciones de seguridad:** Permisos de acceso a nivel de usuario, configuración de firewalls y cifrado de datos.
- **Configuraciones de servicio:** Parámetros específicos del servicio de la base de datos, como los puertos de red o la configuración de replicación.
- **Configuraciones de sistema:** Variables que afectan al funcionamiento general del sistema operativo y de la base de datos en conjunto.

Ver Opciones de Configuración Detalladas

```
SELECT name,
is_auto_shrink_on, -- Te dirá si AUTO_SHRINK está ON (1) o OFF (0)
is_auto_create_stats_on, -- Te dirá si AUTO_CREATE_STATISTICS está ON (1) o OFF (0)
state_desc, -- Estado de la base de datos (ONLINE, OFFLINE, etc.)
compatibility_level -- Nivel de compatibilidad
FROM sys.databases
WHERE name = 'QhatuPERU';
```



--Ver Opciones de Configuración Específicas del Archivo--

Consultar la información de los archivos de la base de datos

```
SELECT
name,
physical_name AS FileLocation,
size * 8 / 1024 AS SizeMB, -- Tamaño actual en MB
growth * 8 / 1024 AS GrowthIncrementMB -- Incremento de crecimiento
FROM sys.master_files
WHERE database_id = DB_ID('QhatuPERU');
```

SQLQuery1.sql - L...H2M65\Megam (86)*

```
-- Consultar la información de los archivos de la base de datos
SELECT
    name,
    physical_name AS FileLocation,
    size * 8 / 1024 AS SizeMB, -- Tamaño actual en MB
    growth * 8 / 1024 AS GrowthIncrementMB -- Incremento de crecimiento
FROM sys.master_files
WHERE database_id = DB_ID('QhatuPERU');
```

100 %

Results Messages

	name	FileLocation	SizeMB	GrowthIncrementMB
1	QhatuPERU	C:\Program Files\Microsoft SQL Server\MSSQL16.MSS...	8	64
2	QhatuPERU_log	C:\Program Files\Microsoft SQL Server\MSSQL16.MSS...	8	64

--Configuraciones de Rendimiento (Memoria, Consultas)--

-- Muestra todas las configuraciones actuales del servidor EXEC sp_configure; --

```
--Configuraciones de Rendimiento (Memoria, Consultas)--
-- Muestra todas las configuraciones actuales del servidor
EXEC sp_configure;
--Habilitar Opciones Avanzadas--
EXEC sp_configure 'show advanced options', 1;
RECONFIGURE;
-- Ejemplo: Ver la memoria máxima del servidor (MB)
EXEC sp_configure 'max server memory (MB)';
```

100 %

Results Messages

	name	minimum	maximum	config_value	run_value
1	access check cache bucket count	0	65536	0	0
2	access check cache quota	0	2147483647	0	0
3	Ad Hoc Distributed Queries	0	1	0	0
4	ADR cleaner retry timeout (min)	0	32767	15	15
5	ADR Cleaner Thread Count	1	32767	1	1
6	ADR Preallocation Factor	0	32767	4	4
7	affinity I/O mask	-2147483648	2147483647	0	0
8	affinity mask	-2147483648	2147483647	0	0
9	affinity64 I/O mask	-2147483648	2147483647	0	0
10	affinity64 mask	-2147483648	2147483647	0	0
11	Agent XPs	0	1	0	0

	name	minimum	maximum	config_value	run_value
1	max server memory (MB)	128	2147483647	2147483647	2147483647

Tipos de recuperación (Simple, Full, Bulk-Logged)

Simple

- **Descripción:** Es el modelo más básico y simple.
- **Registro de transacciones:** El espacio del registro de transacciones se libera automáticamente después de cada punto de control, manteniendo el archivo de registro pequeño.
- **Copias de seguridad:** Solo admite copias de seguridad completas y diferenciales. No se pueden hacer copias de seguridad del registro de transacciones.
- **Restauración:** Solo es posible restaurar la base de datos hasta el momento en que se completó la última copia de seguridad completa o diferencial.
- **Casos de uso:** Bases de datos de desarrollo, de pruebas o aplicaciones donde la pérdida de datos hasta un punto reciente es tolerable.

Full (Completo)

- **Descripción:** Es el modelo más robusto y seguro.
- **Registro de transacciones:** Mantiene un registro completo de todas las transacciones hasta que se realiza una copia de seguridad del registro de transacciones.
- **Copias de seguridad:** Admite copias de seguridad completas, diferenciales, de archivos, de grupos de archivos y del registro de transacciones.
- **Restauración:** Permite restaurar la base de datos a cualquier punto específico en el tiempo.
- **Casos de uso:** Bases de datos críticas que requieren alta disponibilidad y la capacidad de recuperarse con precisión.

Bulk-Logged (Registro Masivo)

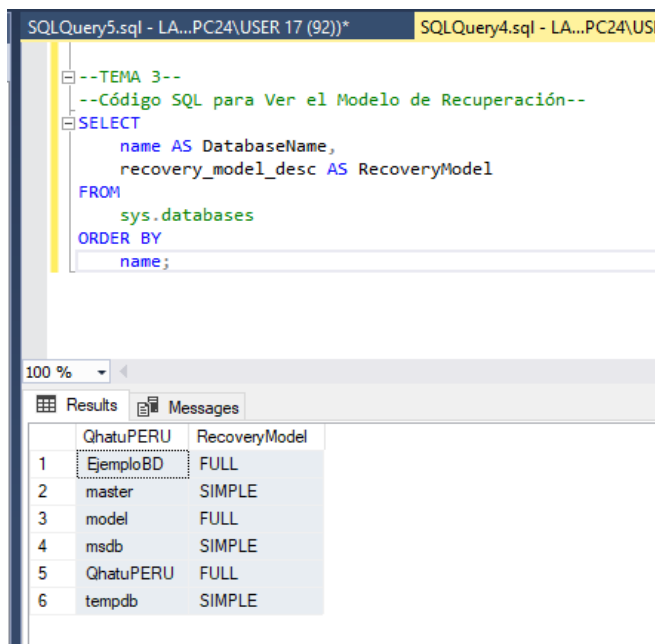
- **Descripción:** Es un modelo híbrido que equilibra rendimiento y recuperación.
- **Registro de transacciones:** Registra mínimamente las operaciones masivas (como `BULK INSERT`, `SELECT INTO`, `CREATE INDEX`) para optimizar el rendimiento. Se comporta como un modelo FULL en las transacciones normales.
- **Copias de seguridad:** Admite copias de seguridad completas, diferenciales y del registro de transacciones.
- **Restauración:** No se puede realizar una restauración puntual si el punto de restauración se encuentra dentro de una operación masiva que fue registrada mínimamente.
- **Casos de uso:** Bases de datos que realizan grandes volúmenes de inserciones de datos, como cargas masivas durante la noche, para optimizar

el tamaño del registro. Se recomienda usar el modelo FULL como modelo principal y cambiar temporalmente a BULK-LOGGED para estas operaciones, volviendo a FULL después.

Código SQL para Ver el Modelo de Recuperación

El siguiente código te permite ver el modelo de recuperación (Simple, Full o Bulk-Logged) de todas las bases de datos en tu servidor:

```
SELECT name AS DatabaseName,  
recovery_model_desc AS RecoveryModel  
FROM sys.databases  
  
ORDER BY  
  
name;
```



Para una Base de Datos Específica (Ejemplo: QhatuPERU)

```
SELECT  
    name AS DatabaseName,  
    recovery_model_desc AS RecoveryModel  
FROM  
    sys.databases  
WHERE  
    name = 'QhatuPERU';
```

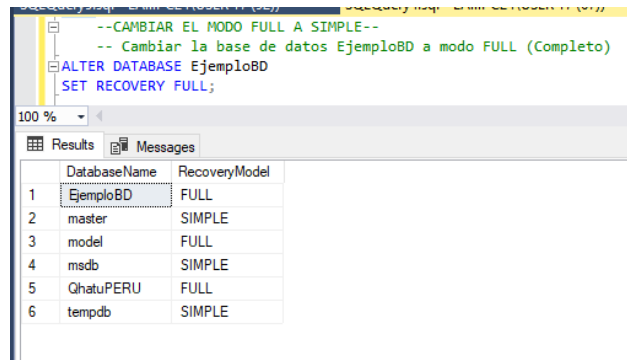
Código SQL para CAMBIAR el Modelo de Recuperación

Si deseas cambiar el modelo de recuperación de una base de datos (por ejemplo, de SIMPLE a FULL para habilitar la restauración puntual), usa el comando ALTER DATABASE:

-- Cambiar la base de datos EjemploBD a modo FULL (Completo)

ALTER DATABASE EjemploBD

SET RECOVERY FULL;



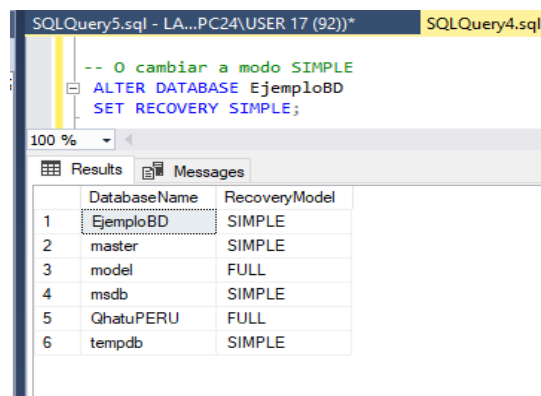
The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the server hierarchy, and the right pane shows the properties of the 'EjemploBD' database. The 'RecoveryModel' property is set to 'FULL'.

DatabaseName	RecoveryModel
1 EjemploBD	FULL
2 master	SIMPLE
3 model	FULL
4 msdb	SIMPLE
5 QhataPERU	FULL
6 tempdb	SIMPLE

-- O cambiar a modo SIMPLE --

ALTER DATABASE EjemploBD

SET RECOVERY SIMPLE;



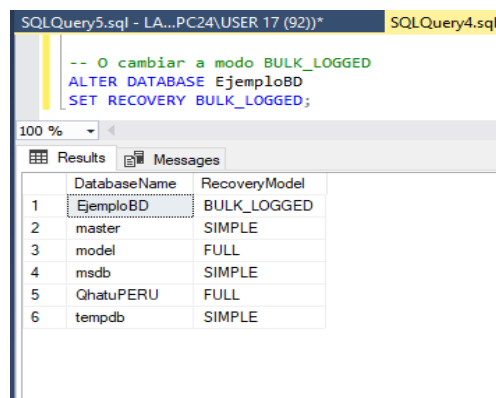
The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the server hierarchy, and the right pane shows the properties of the 'EjemploBD' database. The 'RecoveryModel' property is set to 'SIMPLE'.

DatabaseName	RecoveryModel
1 EjemploBD	SIMPLE
2 master	SIMPLE
3 model	FULL
4 msdb	SIMPLE
5 QhataPERU	FULL
6 tempdb	SIMPLE

-- O cambiar a modo BULK_LOGGED --

ALTER DATABASE EjemploBD

SET RECOVERY BULK_LOGGED;



The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the server hierarchy, and the right pane shows the properties of the 'EjemploBD' database. The 'RecoveryModel' property is set to 'BULK_LOGGED'.

DatabaseName	RecoveryModel
1 EjemploBD	BULK_LOGGED
2 master	SIMPLE
3 model	FULL
4 msdb	SIMPLE
5 QhataPERU	FULL
6 tempdb	SIMPLE

Administración de usuarios y roles de seguridad

La administración de usuarios y roles de seguridad es el conjunto de acciones que realiza el administrador de la base de datos (DBA) para controlar quién puede acceder, qué puede hacer y a qué datos puede acceder dentro del sistema de gestión de base de datos (SGBD). Su objetivo es garantizar la seguridad, confidencialidad e integridad de los datos almacenados.

Elementos principales:

Elemento	Descripción
Usuario	Persona o aplicación que se conecta a la base de datos mediante un nombre de usuario y contraseña.
Rol (Role)	Conjunto de permisos agrupados que se pueden asignar a varios usuarios.
Permiso (Privilege)	Acción específica que un usuario puede realizar sobre un objeto (tabla, vista, procedimiento, etc.).
Autenticación	Proceso de verificar la identidad de un usuario antes de permitir el acceso.
Autorización	Determina qué operaciones puede realizar un usuario una vez autenticado.

Tipos de roles de seguridad comunes:

Rol	Descripción	Ejemplo de permisos
Administrador (DBA)	Tiene control total del sistema y puede crear o eliminar bases de datos, usuarios y roles.	CREATE DATABASE, DROP USER, GRANT ALL
Desarrollador	Crea tablas, vistas y procedimientos, pero no gestiona seguridad global.	CREATE TABLE, CREATE VIEW
Usuario estándar	Solo puede consultar y modificar datos de tablas específicas.	SELECT, INSERT, UPDATE, DELETE
Invitado (Guest)	Solo puede leer datos públicos, sin modificar nada.	SELECT