

# Capítulo 8

## Alta disponibilidad con AlwaysOn

Al finalizar el capítulo, el alumno podrá:

- Reconocer las capacidades de alta disponibilidad de SQL Server.
- Identificar las necesidades y retos de planear y gestionar un entorno de base de datos altamente disponible.
- Reconocer e implementar las diferentes capas de protección ofrecidas por SQL Server AlwaysOn.
- Conocer las opciones de alta disponibilidad híbrida con nodos de Always-On en la nube

### Temas

1. Conceptos de alta disponibilidad
2. Herramientas de alta disponibilidad
3. Protección a nivel de instancias de SQL Server
4. Protección a nivel de bases de datos de SQL Server con AlwaysOn

## 1. Conceptos de alta disponibilidad


### Conceptos de alta disponibilidad

Alta Disponibilidad: Solución para minimizar o mitigar el impacto de paradas de sistemas

- Idealmente, un sistema de base de datos debería estar disponible las 24 horas del día, cada día.
- Técnicas y herramientas para incrementar la disponibilidad de los sistemas.
- Implica redundancia de algún tipo.

8 - 4

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



Implementar alta disponibilidad implica la puesta en marcha de una solución para minimizar o mitigar el impacto de paradas de sistemas, para que idealmente, un sistema de base de datos esté disponible las 24 horas del día.

### Factores que afectan a la disponibilidad

Errores de software. Una aplicación estará compuesta de muchos componentes de software, la mayoría de los cuales son necesarios para que la aplicación funcione. Además de las aplicaciones de cliente y servidor, puede haber aplicaciones de nivel medio y de servidor Web. Los sistemas operativos en todos los niveles son cruciales para que una aplicación se ejecute sin problemas. Es más, aplicaciones y servicios aparentemente no relacionados pueden provocar errores en la aplicación, si usan demasiados recursos o entran en conflicto con la aplicación.

Errores de los componentes de hardware. Se deben tener en cuenta los efectos que puede causar un error de los componentes de hardware en un sistema. Los discos duros, procesadores, memoria y tarjetas de red, producirán problemas concretos, pero otros componentes, como los ventiladores y el suministro eléctrico, también pueden ser igual de esenciales para el buen funcionamiento del sistema.

Errores de red. Un error de red puede consistir en un cable defectuoso, pero existen las mismas probabilidades de que se deba a problemas en la configuración de red. Una red basada en un protocolo de control de transporte/Protocolo de Internet (TCP/IP) moderno usará servidores de Sistema de nombres de dominio (DNS) y servidores de protocolo de configuración dinámica de host (DHCP), así como, enrutadores, concentradores y conmutadores. Todos ellos podrán producir errores o sufrir problemas de configuración.

Interrupción del suministro eléctrico y desastres naturales. Es relativamente fácil protegerse durante cortos períodos de tiempo de las interrupciones de suministro eléctrico o subidas de tensión. Sin embargo, las interrupciones de suministro eléctrico de larga duración y los desastres naturales, como inundaciones y huracanes, pueden afectar a toda una zona durante un largo período de tiempo.

La alta disponibilidad se obtiene mediante un conjunto de técnicas y herramientas para incrementar la disponibilidad de los sistemas, que por lo general, implica redundancia de algún tipo. La redundancia puede resolver la mayoría de factores que afectan a la disponibilidad, puesto que puede suponer la duplicación de bases de datos, componentes de hardware y red, servidores o incluso sitios enteros.

Redundancia de los componentes de hardware. Los servidores modernos pueden duplicar la mayoría de los componentes para mejorar la disponibilidad. Los suministros eléctricos, los ventiladores, la memoria y las tarjetas de interfaz de red (NIC) redundantes ofrecen componentes secundarios en caso de fallo.

Redundancia de la red. Si se tienen varias NIC, estas pueden ser asociadas a las diferentes subredes y proporcionar redundancia si una de las subredes falla. Las NIC pueden combinarse usando el software del proveedor de NIC mediante un proceso llamado agrupamiento de NIC. Cada NIC usará una dirección IP virtual compartida, con ello, cuando todas estén en funcionamiento aumentará el ancho de banda.

RAID. La matriz redundante de discos independientes (RAID) es una solución de servidor única que proporciona redundancia de disco duro y mejoras en el rendimiento. RAID puede ser una solución de Microsoft Windows® o de hardware, con aplicaciones de hardware que proporcionan mejor rendimiento y protección, aunque a un precio elevado. Las formas más populares de RAID disponibles son RAID 1 (creación de reflejo de disco), RAID 5 (intercalado de datos en discos con paridad distribuida) y RAID 10 (creación de reflejos con bandas, también conocido como RAID 1+0).

Redundancia del servidor y de la base de datos. Se explica en el punto 8.2.

## Acuerdos de nivel de servicios

Un acuerdo de nivel de servicio o Service Level Agreement (SAL) es un contrato escrito entre un proveedor de servicio y su cliente (externo o interno) con objeto de fijar el nivel acordado para la calidad de dicho servicio:

- Ayuda a ambas partes a llegar a un consenso en términos del nivel de calidad del servicio, en aspectos tales como: tiempo de respuesta, disponibilidad horaria, documentación disponible y personal asignado al servicio,
- Identifica y define las necesidades del cliente, a la vez que controla sus expectativas de servicio en relación a la capacidad del proveedor.
- Proporciona un marco de entendimiento, simplifica asuntos complicados, reduce las áreas de conflicto y favorece el diálogo ante la disputa.

Se ha comentado que entre los aspectos que se definen está la disponibilidad de los equipos, la cual es expresada en función a la cantidad de números 9:

Número de 9s	% disponibilidad	Tiempo de parada anual
2	99%	3 días y 15 horas
3	99.9%	8 horas y 45 minutos
4	99.99%	52 minutos y 34 segundos
5	99.999%	5 minutos y 15 segundos

Esta disponibilidad incluye las paradas planeadas y no planeadas.

## Modos de espera

Determina la forma cómo va a responder el sistema en caso de falla:

### HOT STANDBY

- Se mantiene una copia de los datos del servidor primario en el secundario.
- El nodo secundario provee una copia transaccionalmente consistente de los datos del servidor primario.
- Detección automática de errores y recuperación.

### WARM STANDBY

- También, se mantiene la copia de los datos en el secundario, pero los datos pueden o no, ser confirmados concurrentemente.
- El error y recuperación puede no ser automático

### COLD STANDBY

- Servidor donde se pueden restaurar los datos.
- Se necesita OS apropiado, software y copia de seguridad.
- Puede llevar un tiempo considerable.


## 2. Herramientas de alta disponibilidad

### Herramientas de alta disponibilidad

- SQL Server provee tecnologías que pueden ayudarlo a crear sistemas de base de datos altamente disponibles.
- Protección de base de datos contra fallos:
  - Trasvase de registro de transacciones.
  - Base de datos reflejada.
  - AlwaysOn.
- Protección de instancia contra fallos:
  - Failover Clustering.
- Algunas veces se combina más de una tecnologías de Alta Disponibilidad.

8 - 7

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



SQL Server proporciona varias opciones para crear una alta disponibilidad para un servidor o base de datos. Las opciones de alta disponibilidad incluyen las siguientes:

**Trasvase de registros.** Es un método económico para crear un servidor de reserva mediante hardware estándar. En principio, restaura una copia de seguridad completa de la base de datos del servidor primario en un sistema de reserva, que se actualizará periódicamente mediante la aplicación de registros de transacción del servidor primario en el sistema de reserva. El trasvase de registros está disponible para las bases de datos de usuario, mientras que las copias de seguridad manuales y las restauraciones son necesarias para las bases de datos del sistema. Si se produce un error en el servidor primario, se debe conectar manualmente el servidor de reserva.

**Reflejo de la base de datos.** La creación de reflejo de la base de datos es una forma mejorada de trasvase de registros disponible en SQL Server Enterprise Edition. Tal y como ocurre en el trasvase de registros, sólo se protegen las bases de datos del usuario. Las transacciones se aplican desde un servidor primario a uno en reserva, pero a diferencia del trasvase de registros, las transacciones se aplican prontamente, en lugar de en intervalos predefinidos. Los conjuntos de reflejos pueden conmutarse por error automáticamente, si se produce un error en el servidor primario. Los clientes podrán usar automáticamente el servidor de reserva.

**Always On.** Es una solución de alta disponibilidad en recuperación de desastres (HADR - High Availability Disaster Recovery) resultado de la combinación de las soluciones de Clúster y Reflejo de la base de datos. De Clúster toma la parte de poder administrar recursos en grupos denominados grupos de disponibilidad y

principalmente, fallar entre nodos sin la necesidad de tener un disco compartido entre los nodos. De Reflejo de la base de datos toma el sincronizar bases añadiendo la facultad de poder hacer uso de las réplicas secundarias.

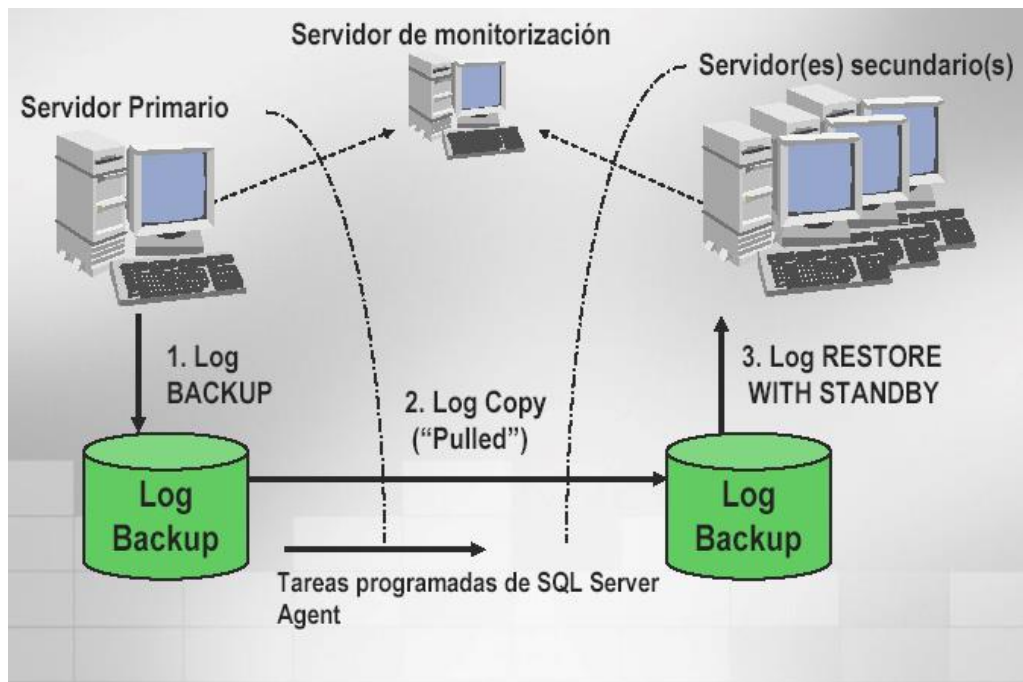
**Clúster de servidores.** Los clústeres ofrecen una solución de alta disponibilidad para los servidores. En caso de error, el sistema operativo y los servicios actúan juntos para ofrecer una conmutación por error automática, en menos de un minuto. El clúster de conmutación por error no requiere ninguna intervención manual durante la conmutación por error en el servidor o en los clientes.

	Base de datos reflejada	Failover Clustering	Always On	Trasvase de registro de transacciones
<b>Unidad de falla</b>	Base de datos	Servidor	Grupo de Bases de Datos	Base de datos
<b>Falla automática</b>	Sí, con testigo	Sí	Sí	No
<b>Réplicas de datos</b>	1	0	6	Ilimitado
<b>Complejidad</b>	Media	Media	Alta	Baja
<b>Disponibilidad de la réplica</b>	Solo lectura con snapshot	N/A	Solo lectura	Solo lectura entre restauración
<b>Pérdida de datos (RPO)</b>	Cero (AD) Segundos (AP)	N/A	Cero (sync) Segundos (async)	Minutos a Horas
<b>Tiempo de recuperación (RTO)</b>	Segundos (AD) Minutos (AP)	Segundos a minutos	Segundos (sync) Minutos (async)	Horas a días

## 2.1 Trasvase de registro

El trasvase de registros es un método económico para crear un servidor secundario mediante el hardware estándar. Funciona mediante la restauración inicial de una copia de seguridad completa de la base de datos en el servidor primario en un servidor secundario.

A continuación, se aplican periódicamente los registros de transacciones desde el servidor primario al sistema secundario. El trasvase de registros está disponible para las bases de datos de usuario, pero no para las de sistema.



El trasvase de registros es una técnica de alta disponibilidad en la que el registro de transacciones del servidor primario se restaura periódicamente, en un servidor secundario. Puede programar las copias de seguridad del registro para que se produzcan con una frecuencia que se adapte a los requisitos de disponibilidad y rendimiento. Además de proporcionar redundancia, el servidor secundario se puede usar para consultas de solo lectura y de esta forma, aliviar parte de la carga del servidor primario.

En el caso de que se produzca un error en el servidor primario, la conmutación por error automática no tendrá lugar. Por ello, se deben promover manualmente el servidor secundario y reconfigurar todos los clientes para que se conecten a él. Si es necesaria una solución más automatizada, se debería considerar la creación de reflejo de la base de datos o el clúster de servidores.

De manera opcional, se puede crear un servidor de supervisión. El servidor de supervisión registra cualquier problema con el trasvase de registros además de enumerar las últimas operaciones de copia de seguridad y restauración. Los servidores de supervisión deberían separarse de los servidores primario y secundario en caso de error en los servidores.

El trasvase de registros se puede configurar mediante SQL Server Management Studio o usando Transact-SQL. Antes de configurar el trasvase de registros, sin embargo, se deberán realizar las siguientes tareas:

- Crear una carpeta compartida de archivos para las copias de seguridad del registro de transacciones, preferiblemente, en un servidor tolerante a errores que no forma parte de la configuración del trasvase de registros. Para maximizar la disponibilidad del servidor primario, Microsoft recomienda que se coloque el recurso compartido de copia de seguridad en un equipo host independiente.

- Crear una carpeta para cada servidor secundario en el que el trasvase de registros copiará los archivos de copia de seguridad del registro de transacciones. Estas carpetas se colocan en los servidores secundarios.

Para configurar el trasvase de registros utilizando Transact-SQL se deben utilizar los siguientes procedimientos almacenados:

- sp\_add\_log\_shipping\_primary\_database
- sp\_add\_jobschedule
- sp\_add\_log\_shipping\_alert\_job
- sp\_add\_log\_shipping\_secondary\_primary
- sp\_add\_log\_shipping\_secondary\_database
- sp\_add\_log\_shipping\_primary\_secondary

### Conmutación manual de trasvase de registro

El cambio de funciones convierte el servidor de reserva en servidor primario. Se necesita realizar un cambio de funciones si el servidor primario se queda inesperadamente sin conexión o si necesita que el servidor primario esté sin conexión para realizar tareas de mantenimiento.

La primera vez que se cambian las funciones, se necesitará configurar el trasvase de registros en la base de datos secundaria. Esto no es necesario para los cambios de funciones siguientes, por lo que será más fácil cambiar y volver al estado previo.

Se deben ejecutar los siguientes pasos para cambiar las funciones o para que el servidor de reserva pase a ser el servidor primario:

1. Copiar cualquier copia de seguridad del registro de transacciones del recurso compartido de copia de seguridad en la carpeta de destino y restaurar estas y otras copias de seguridad de la carpeta en el servidor de reserva.
2. Realizar una copia de seguridad del registro con NORECOVERY si el servidor primario está disponible, tal y como se muestra en el siguiente código de Transact-SQL:

```
BACKUP LOG BaseDeDatos TO DBLogBackup WITH NORECOVERY
```

3. Restaurar la copia de seguridad del paso anterior en el servidor de reserva con la instrucción RECOVERY, tal y como se muestra en el siguiente código de Transact-SQL:

```
RESTORE LOG BaseDeDatos FROM DBLogBackup WITH RECOVERY
```

De manera alternativa, si no hay disponible ninguna copia de seguridad, se llevará a cabo una restauración con RECOVERY, sin especificar un archivo de copia de seguridad, tal y como se muestra en el siguiente código de Transact-SQL:

```
RESTORE LOG DBLogBackup WITH RECOVERY
```



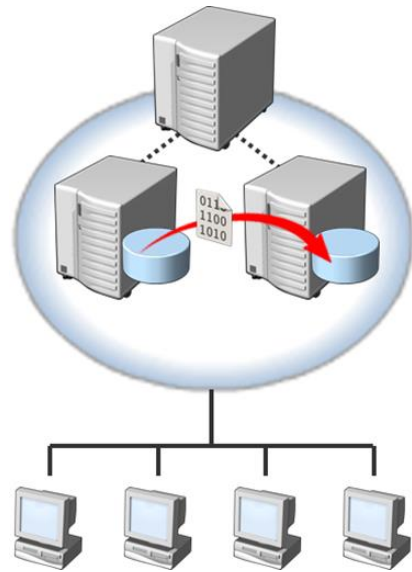
4. Deshabilite los trabajos de trasvase de registros en el servidor primario original y deshabilite la copia y restaure los trabajos en el servidor secundario original.
5. Si es la primera vez que se han cambiado las funciones del servidor, se tendrá que configurar el trasvase de registros en la base de datos secundaria. Ahora, debería ser tratada como una base de datos primaria. Usar el mismo recurso compartido para crear copias de seguridad que ya fueron creadas para el servidor primario original. Al agregar la base de datos secundaria, en el cuadro de diálogo, Configuración de base de datos secundaria, escribir el nombre de la base de datos primaria original en el cuadro Base de datos secundaria y a continuación, seleccionar **No, la base de datos secundaria está inicializada**.

## 2.2 Base de datos reflejada

La creación de reflejo de la base de datos es una solución de alta disponibilidad, alternativa al clúster de conmutación por error en SQL Server Enterprise Edition. La creación de reflejo de la base de datos admite la conmutación por error automática, pero no requiere un hardware compatible con los clústeres. Por tanto, puede proporcionar una alternativa rentable al clúster de conmutación por error.

La creación de reflejo de la base de datos se puede implementar con hardware estándar. Toda la administración tendrá lugar en su totalidad en SQL Server.

En una solución de creación de reflejo de la base de datos, se almacena una base de datos en un servidor y se copia en otro. De esta forma, se proporciona una copia secundaria de la base de datos que puede prestar servicio a los clientes en caso de un error de servidor.



### Funciones de servidor en la creación de reflejo

La creación de reflejo de la base de datos requiere varias instancias de SQL Server, que se deberían instalar en equipos independientes para proporcionar protección frente a los errores del servidor. Las funciones del servidor en una solución de creación de reflejo de la base de datos son:

**Servidor principal.** El servidor principal aloja la copia activa de la base de datos (denominada base de datos principal) y presta servicio a las solicitudes de los clientes. El servidor principal reenvía todas las transacciones al servidor reflejado antes de aplicarlas en la base de datos principal.

**Servidor reflejo.** El servidor reflejo aloja una copia de la base de datos principal (denominada base de datos reflejada) y aplica las transacciones reenviadas por la base de datos principal para mantener la base de datos reflejada sincronizada con la principal.

**Servidor testigo.** El servidor testigo es un componente opcional de una solución de creación de reflejo de la base de datos. Cuando está presente, un servidor testigo supervisa los servidores principales y reflejados, para asegurar una conectividad continuada y la participación en la sesión de reflejo (denominada quórum). Si uno de los servidores pierde el quórum, el servidor testigo asigna la función de servidor principal, con lo que se produce la conmutación por error automática del servidor principal al reflejado si fuera necesario. Los servidores testigo son necesarios para la conmutación por error automática; sin embargo, un servidor testigo puede admitir varias sesiones de reflejo porque no se trata de una tarea intensiva.

## Redirección transparente de cliente

Cuando una sesión de creación de reflejo de la base de datos se conmuta por error, todas las aplicaciones cliente deben conectarse al nuevo servidor principal (el servidor reflejado anterior). Las aplicaciones cliente que usan SQL Native Client o el proveedor de datos Microsoft .NET Framework 2.0 para Microsoft SQL Server, admiten la redirección automática de clientes y pueden controlar de forma transparente la conmutación por error al servidor reflejado. Las aplicaciones cliente que usen otras tecnologías de acceso de datos deben adaptarse para redirigir las solicitudes al servidor reflejado en el caso de conmutación por error.

## Configuración de bases de datos reflejadas

Antes de poder establecer una sesión de creación de reflejo de la base de datos, se deben realizar las tareas de preparación descritas en la lista siguiente:

Crear extremos de reflejo e inicios de sesión. Se deben crear extremos y un inicio de sesión en la base de datos máster para cualquier instancia del servidor que se ejecute como una cuenta de usuario de dominio diferente del servidor principal.







Establecer el modelo de recuperación. Se debe establecer el modelo de recuperación para la base de datos que se va a reflejar en COMPLETO (full).

Realizar una copia de seguridad de la base de datos principal y restaurarla en el servidor reflejado. Se debe realizar una copia de seguridad completa de la base de datos principal y restaurarla en la instancia reflejada mediante la especificación de NORECOVERY y el uso del mismo nombre como la base de datos principal.

De manera alternativa, se podría usar una copia de seguridad completa reciente. Si es el caso, se debería restaurar cualquier copia de seguridad del registro desde la copia de seguridad completa y realizar una copia de seguridad, así como, una restauración del registro de transacciones para asegurarse de que los datos están al día.

Copiar recursos de nivel de servidor. Se debería copiar manualmente cualquier recurso de nivel de servidor, como inicios de sesión o trabajos del Agente SQL, que serían necesarios en el caso de conmutación por error a la instancia reflejada.

### Opciones de funcionamiento de bases de datos reflejadas

Modo	Conmutación automática	Protección completa de pérdida de datos
Alta disponibilidad		
Alta protección		
Alto rendimiento		

Modo de alta disponibilidad. En el modo de alta disponibilidad, se establece la seguridad de la transacción en COMPLETA, lo que provoca que las transacciones se apliquen a las bases de datos principal y reflejada de manera sincronizada. Cuando el servidor principal confirma una transacción, el servidor reflejado también lo hace. El servidor principal solo emite una confirmación una vez que el servidor reflejado ha enviado la confirmación de que se ha almacenado la transacción en el disco.

El modo de alta disponibilidad usa un servidor testigo. Debería colocarse en un tercer servidor (no en el principal ni en el reflejado) para ofrecer redundancia. El modo de alta disponibilidad permite la conmutación por error automática o manual del servidor principal en el reflejado.

Si se produce un error en el servidor principal en el modo de alta disponibilidad, el servidor testigo inicia la conmutación por error automática en el reflejado. Si se produce un error en el servidor reflejado, la base de datos se sigue con conexión mientras se mantenga el quórum entre los servidores: principal y testigo.

Modo de alta protección. En el modo de alta protección, la seguridad de la transacción se establece en COMPLETA para aplicar transacciones de manera sincronizada, tal y como ocurre en el modo de alta disponibilidad; sin embargo, el modo de alta protección no usa un servidor testigo.

Si se produce un error en el servidor principal en el modo de alta protección, habrá una copia completa de los datos en el servidor reflejado, pero deberá realizar manualmente la conmutación por error. Si se produce un error en el servidor reflejado, el principal se desconecta por sí mismo para evitar el riesgo de pérdida de datos.

Modo de alto rendimiento. En el modo de alto rendimiento, la seguridad de la transacción se desactiva y las transacciones se aplican de forma asíncrona. De esta forma, el servidor principal responde a los clientes sin comprobar primero que las transacciones se hayan aplicado en el servidor reflejado. Aunque así se obtiene mejor rendimiento, pero se sacrifica la alta disponibilidad.

Si se produce un error en el servidor principal en el modo de alto rendimiento, se deberá conmutar por error, manualmente, al servidor reflejado. Sin embargo, debido a que es posible que algunas transacciones se completen en el servidor principal pero no en el reflejado, algunos datos pueden perderse. Si se produce

un error en el servidor reflejado en el modo de alto rendimiento, el servidor principal no se verá afectado.

### Conmutación de bases de datos reflejadas

Si se produce un error en el servidor principal en el modo de alta disponibilidad, la conmutación por error será automática. Cuando el servidor principal está sin conexión, el reflejado y el testigo, formarán un quórum y harán que el reflejado sea ahora el principal. Si el servidor principal original vuelve a conectarse, pasará a ser el reflejado.

### Conmutación manual

Si se produce un error en el servidor principal en el modo de alta protección, debe iniciar la conmutación por error manual realizando los pasos siguientes en el servidor principal:

1. Conectar con la instancia del servidor principal y, en el panel **Explorador de objetos**, hacer clic en el nombre del servidor para expandir el árbol de servidores.
2. Expandir **Bases de datos** y a continuación, seleccionar la base de datos para la que se va a crear el reflejo.
3. Hacer clic con el botón secundario en la base de datos y luego, clic en **Propiedades**; se abrirá el cuadro de diálogo **Propiedades de la base de datos**.
4. En el panel **Seleccionar una página**, hacer clic en **Creación de reflejo**.
5. Hacer clic en **Conmutación por error**.

De manera alternativa puede iniciar la conmutación por error manual usando Transact-SQL en el servidor principal, tal y como se observa en la siguiente muestra de código:

```
ALTER DATABASE BaseDeDatos SET PARTNER FAILOVER
```

### Servicio forzado (con posible pérdida de datos)

Si se produce un error en el servidor principal en el modo de alto rendimiento, se puede forzar el servicio en el servidor reflejado, con posible pérdida de datos, usando la instrucción ALTER DATABASE con el parámetro FORCE\_SERVICE\_ALLOW\_DATA\_LOSS, tal y como se observa en la siguiente muestra de código:

```
ALTER DATABASE BaseDeDatos  
SET PARTNER FORCE_SERVICE_ALLOW_DATA_LOSS
```

### 3. Protección a nivel de instancias de SQL Server

#### Protección a nivel de instancias de SQL Server

##### Clúster de tolerancia a fallos

- Hot Standby.
- Soporte de servidor completo y alta disponibilidad en caso de falla de hardware o por mantenimiento.
- En caso de falla, el sistema operativo y SQL Server trabajan juntos para proveer un sistema automatizado para fallas.



8 - 32

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.

#### 3.1 Clúster de Conmutación por Error de SQL Server

Los clústeres son una estrategia de alta disponibilidad en la que se configuran varios servidores físicos para que se comporten como un único servidor virtual. Cada servidor físico se conoce como un nodo y cada aplicación de base de datos alojada en el clúster tiene un nodo activo designado que presta servicio a las solicitudes de los clientes. Los nodos de un clúster comparten una matriz de almacenamiento única y, en el caso de que se produzca un error en un nodo activo de la aplicación, otro nodo del clúster asumirá automáticamente la función del nodo activo. Esta reasignación automática del nodo activo se conoce como conmutación por error automática, y una configuración de clústeres que admita la conmutación por error automática se conoce como clúster de conmutación por error. El clúster de conmutación por error está admitido en SQL Server Enterprise Edition, Developer Edition y, con algunas restricciones, en Standard Edition.

SQL Server Enterprise Edition y SQL Server Developer Edition son completamente compatibles con los clústeres hasta un máximo de ocho nodos. Se pueden usar clústeres en Standard Edition pero sólo puede usar dos nodos. No se pueden usar clústeres en otras ediciones.

La tabla siguiente muestra cómo varias ediciones de Windows admiten el clúster de conmutación por error.

- Microsoft Windows Server 2008 R2
- Windows 2012 Datacenter Server 4
- Windows Server 2012 Enterprise Edition

Se puede usar clúster de conmutación por error en los siguientes casos:

- Cuando se necesita la conmutación por error automática en el caso de un error del servidor.
- Cuando se necesita que se conmuten por error automáticamente los recursos a nivel de instancia como los inicios de sesión, los extremos y los trabajos y configuración del agente SQL Server.
- Se debe disponer de hardware compatible con los clústeres. No todo el hardware es compatible.

Los clústeres exigen requisitos específicos de hardware y de software. Es importante consultar la última lista de compatibilidad de Hardware (HCL).

El hardware debe aparecer en la Lista de compatibilidad de hardware y el catálogo de Microsoft Windows. El sistema de hardware debe aparecer bajo la categoría de una solución de clústeres. Es importante que el hardware sea compatible como sistema y que no sea solo una selección de componentes compatibles unidos.

Al usar una red de área de almacenamiento (SAN), la solución de hardware completa debe estar en la categoría de clúster/dispositivo de clústeres múltiples de la Lista de compatibilidad de hardware y el catálogo de Microsoft Windows.

### **Configuración de clúster de Windows**

El sistema operativo debe admitir el clúster de conmutación por error. Para obtener más información sobre los sistemas operativos compatibles, se deberá consultar “Requisitos de hardware y software para instalar SQL Server” en los Libros en pantalla de SQL Server.

Se debe activar el Proveedor de servicios de cifrado de Windows (CSP) en Microsoft Windows Server. Si el servicio CSP no se está ejecutando en ningún nodo del clúster, se producirá un error de instalación de SQL Server y aparecerá un cuadro de diálogo de requisitos con el logotipo de Windows.

Activar el servicio coordinador de transacciones distribuidas en todos los sistemas operativos para la instalación de clústeres y remota. Si el coordinador de transacciones distribuidas está deshabilitado, se producirá el error 1058 en la instalación de SQL Server.

## **Disco compartido**

Un clúster usa los discos compartidos de manera que, si se produce un error, otro nodo podrá tomar la propiedad de los discos.

SQL Server admite puntos de montaje. Las instalaciones de clúster de SQL Server se limitan al número de letras de unidad disponibles. Suponiendo que usa sólo una letra de unidad para el sistema operativo y que todas las demás letras de la unidad están disponibles como unidades de clúster normales o como unidades de clúster que alojan puntos de montaje, el número máximo de instancias de SQL Server por servidor está limitado a 25. Los volúmenes montados están disponibles con Windows Server.

## **Configurar el servidor virtual**

Si se supone que ya hay un clúster de conmutación por error instalado, se selecciona el disco del clúster donde se ubicarán los archivos de datos de SQL Server y ejecutar a continuación, el programa de instalación de SQL Server en el nodo que controla este disco. También, se deben especificar los nodos del clúster adicionales que deberían incluirse en el servidor virtual. El programa de instalación, automáticamente, instalará los componentes de SQL Server necesarios en cada nodo del servidor virtual.

Crear una instancia predeterminada de SQL Server y especificar un nombre de servidor virtual. Siempre que se conecte con SQL Server, se deberá usar este nombre.

## **Instalar instancias de SQL Server en un clúster**

Cada grupo de recursos puede contener, a lo sumo, una instancia de SQL Server. Para instalar otra, ejecutar de nuevo la instalación en el nodo del clúster que controla el disco del clúster donde se encontrarán los archivos de datos de SQL Server. Crear una instancia con un nombre de servidor virtual nuevo en un grupo diferente de recursos de clúster de la Organización por clústeres de Windows.

Cada servidor virtual reside en un grupo distinto de recursos de organización por clústeres de Windows y cada uno tiene un conjunto único de direcciones IP, un nombre de red distinto y archivos de datos que residen en un conjunto independiente de discos de clúster compartidos.

Cuando una conmutación por error ocurre en cualquier recurso en un grupo de recursos de la organización por clústeres de Windows, también se producirá en todos los recursos que sean miembros de ese grupo.



## Consideraciones de seguridad para clúster de tolerancia a fallos

### Mejorar la seguridad física

El aislamiento físico y lógico constituye la base de la seguridad de SQL Server. Para mejorar la seguridad física de la instalación de SQL Server, realizar las siguientes tareas:

- Colocar el servidor en una sala que solo sea accesible a personas autorizadas.
- Colocar los equipos que hospedan bases de datos en una ubicación protegida físicamente, como una sala de equipos cerrada con sistemas supervisados de detección de inundaciones y de extinción.
- Instalar las bases de datos en una zona segura de la intranet corporativa y no conectar los servidores SQL Server a Internet.
- Realizar periódicamente copias de seguridad de todos los datos y mantenerlos protegidos en una ubicación fuera de las instalaciones.

### Usar firewalls

Los firewalls son importantes para ayudar a proteger la instalación de SQL Server. Los firewalls serán más efectivos si sigue estas instrucciones:

- Instalar un firewall entre el servidor e Internet. Habilitar el firewall. Si el firewall está desactivado, activarlo; caso contrario, no desactivarlo.
- Dividir la red en zonas de seguridad separadas por firewalls. Bloquear todo el tráfico y admitir el necesario.
- En un entorno de varios niveles, utilizar varios firewalls para crear subredes filtradas.
- Si se instala el servidor en un dominio de Windows, configurar firewalls internos para permitir la autenticación de Windows.
- Si la aplicación utiliza transacciones distribuidas, puede ser que se deba configurar el firewall para permitir que el tráfico del Coordinador de transacciones distribuidas de Microsoft (MS DTC, Microsoft Distributed Transaction Coordinator) fluya entre instancias independientes de MS DTC. También se tendrá que configurar el firewall para permitir que el tráfico fluya entre los administradores de recursos y MS DTC como SQL Server.

### Aislar servicios

El aislamiento de servicios reduce el riesgo de que se utilice un servicio cuya seguridad se haya vulnerado para vulnerar la seguridad de otros servicios. Para aislar los servicios, tener en cuenta estas instrucciones:

- Ejecutar los servicios de SQL Server por separado en distintas cuentas de Windows. Siempre que sea posible, utilizar derechos de Windows independientes y bajos, o cuentas de usuario local para cada servicio de SQL Server.

### Configurar un sistema de archivos seguro

Si se usa el sistema de archivos correcto, se aumenta la seguridad. En las instalaciones de SQL Server. Para ello, se debería hacer lo siguiente:

- Usar el sistema de archivos NTFS (es el sistema de archivos preferido para las instalaciones de SQL Server porque es más estable y recuperable que los sistemas de archivos FAT). NTFS también habilita opciones de seguridad como las listas de control de acceso de directorios (ACL, Access Control List) y archivos, y el cifrado de archivos del Sistema de archivos de cifrado (EFS, Encrypting File System). Durante la instalación, SQL Server establecerá las ACL adecuadas en las claves del registro y archivos si detecta NTFS. No se deberían cambiar estos permisos. Puede que las versiones futuras de SQL Server no admitan la instalación en equipos con sistemas de archivos FAT.

### Deshabilitar NetBIOS y bloque de mensajes de servidor

Los servidores de la red perimetral deben tener deshabilitados todos los protocolos innecesarios, incluidos NetBIOS y Bloque de mensajes de servidor (SMB).

NetBIOS utiliza los siguientes puertos:

- UDP/137 (servicio de nombre NetBIOS)
- UDP/138 (servicio de datagrama NetBIOS)
- TCP/139 (servicio de sesión NetBIOS)

SMB utiliza los siguientes puertos:

- TCP/139
- TCP/445

Los servidores web y los servidores del sistema de nombres de dominio (DNS) no requieren NetBIOS o SMB. En estos servidores, se deshabilitan los dos protocolos para reducir la amenaza de enumeración de usuarios.

## 4. SQL Server con AlwaysOn


### Protección a nivel de bases de datos con AlwaysOn

#### ¿Qué es AlwaysOn?

- Nueva característica en SQL Server 2016
- Ambiente tolerante a fallos para bases de datos
- Combina de diferentes tecnologías: clúster, reflejo y trasvase
- Conmutación planeada o automática
- Opciones para pérdida o no de datos en caso de falla

8 - 42

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



AlwaysOn es una funcionalidad nueva liberada con la versión 2012 de SQL Server, para ofrecer alta disponibilidad para las bases de datos.

Utilizar tecnologías existentes como es el reflejo de base de datos con la que se puede implementar un modelo asíncrono y síncrono, pero solo con un servidor secundario. AlwaysOn también implementa tecnologías de clúster en la que se puede utilizar el servidor virtual de forma compartida, con la limitación de una implementación de solo instancias para el servicio y la otra instancia esperando a que la primera falle.

Al igual que en el trasvase de registro, se podrá contar con múltiples servidores secundarios.

Always On utiliza las mejores prácticas de clúster y trasvase de registro, en donde se aprovecha la funcionalidad con el concepto de políticas de conmutación por falla para asignar prioridades. Respecto a la protección de datos que ofrece reflejo de base de datos y trasvase de registro, ofrece múltiples bases de datos, agrupadas y la posibilidad de tener múltiples elementos secundarios, hasta 4 elementos de ellos, así como, un elemento primario para escalabilidad en el servicio.

Para implementar AlwaysOn se configuran grupos de disponibilidad en base a contenedores de bases de datos que ofrecen la posibilidad de tener hasta 4 réplicas secundarias para realizar operaciones de lectura a usuarios que requieren hacer esta tarea, y el acceso a lectura y escritura a quienes así lo requieran.

En resumen, un grupo de disponibilidad es una unidad de replicación que permite encapsular varias bases de datos y distribuirlas entre varios servidores, en donde se

tendrá un servidor primario que recibirá y procesará las transacciones de escritura, además de los servidores secundarios que recibirán la información del primario y que, en ciertos casos, permitirán hacer consultas de solo lectura.

Otro concepto de AlwaysOn, es la conmutación automática por falla que detecta si el servidor primario ha fallado, habilitando un servidor secundario, según como se configure, cambiando automáticamente al rol de servidor primario.

### **Grupos de disponibilidad**

Un grupo de disponibilidad es un entorno de conmutación por error para un conjunto discreto de bases de datos de usuario, conocido como bases de datos de disponibilidad, que realizan la conmutación por error conjuntamente.

Un grupo de disponibilidad admite un conjunto de bases de datos principales y de uno a cuatro conjuntos de bases de datos secundarias correspondientes. Las bases de datos secundarias no son copias de seguridad. Se recomienda continuar ejecutando copias de seguridad de las bases de datos y de sus registros de transacciones periódicamente.

### **Modos de disponibilidad**

El modo de disponibilidad determina si la réplica principal espera la confirmación de transacciones en una base de datos hasta que una réplica secundaria haya escrito las entradas del registro de transacciones en el disco (protegido el registro):

Modo de confirmación asincrónica. La réplica principal confirma las transacciones sin esperar la notificación de que una réplica secundaria de confirmación asincrónica ha protegido el registro. El modo de confirmación asincrónica minimiza la latencia de las transacciones en las bases de datos secundarias, pero permite que se retrasen detrás de las bases de datos principales, haciendo posible alguna pérdida de datos.

Modo de confirmación sincrónica. En este modo, antes de la confirmación de transacciones, una réplica principal de confirmación sincrónica espera a que una réplica secundaria de confirmación sincrónica notifique que ha terminado de proteger el registro. El modo de confirmación sincrónica asegura que, una vez que una base de datos secundaria se sincroniza con la base de datos principal, las transacciones confirmadas quedan totalmente protegidas. Esta protección se produce a costa de que aumente la latencia de las transacciones.

### **Conmutación de AlwaysOn**

Durante una conmutación por error, la réplica secundaria de destino realiza la transición al rol principal, pasando a ser la nueva réplica principal. La nueva réplica principal pone sus bases de datos en línea como bases de datos principales y las aplicaciones cliente pueden conectarse a ellas.

Existen tres formas de conmutación por error: automática, planeada manual y forzada (con posible pérdida de datos), dependen de su modo de disponibilidad.

El modo de confirmación sincrónica admite dos formas de conmutación por error:

Conmutación por error manual planeada (sin pérdida de datos). Se produce después de que un administrador de base de datos emite un comando de conmutación por error y produce la transición de una réplica secundaria sincronizada al rol principal (con protección de datos garantizada) y la transición de la réplica principal al rol secundario. Una conmutación por error manual requiere que la réplica principal y la réplica secundaria de destino se ejecuten en modo de confirmación sincrónica, y la réplica secundaria ya debe estar sincronizada.

Conmutación por error automática (sin pérdida de datos).- Una conmutación por error automática se produce en respuesta a un error que produce la transición de una réplica secundaria al rol principal (con protección de datos garantizada). Cuando la réplica principal anterior está disponible, efectúa la transición al rol secundario. La conmutación por error automática requiere que la réplica principal y la réplica secundaria de destino se ejecuten en modo de confirmación sincrónica con el modo de conmutación por error establecido en "Automático". Además, la réplica secundaria debe estar sincronizada, tener quórum de WSFC y cumplir las condiciones especificadas por la directiva de conmutación por error flexible del grupo de disponibilidad.

En el modo de confirmación asincrónica se admite una forma de conmutación por error.

Conmutación por error manual forzada (con posible pérdida de datos). Denominada, normalmente, conmutación por error forzada. La conmutación por error forzada se considera una forma de conmutación por error manual porque solo se puede iniciar manualmente. Es una opción de recuperación ante desastres. Es la única forma de conmutación por error posible cuando la réplica secundaria de destino no está sincronizada con la réplica principal.