SQL Server Mirroring

To Prepare a New Mirror Database

To prepare a mirror database

Note

For a Transact-SQL example of this procedure, see Example (Transact-SQL), later in this section.

1. Connect to principal server instance.
2. Create either a full database backup or a differential database backup of the principal database.
   - Create a Full Database Backup (SQL Server)
   - Create a Differential Database Backup (SQL Server).
3. Typically, you need to take at least one log backup on the principal database. However, a log backup might be unnecessary, if the database has just been created and no log backup has been taken yet, or if the recovery model has just been changed from SIMPLE to FULL.
   - Back Up a Transaction Log (SQL Server)
4. Unless the backups are on a network drive that is accessible from both systems, copy the database and log backups to the system that will host the mirror server instance.
5. Connect to mirror server instance.
6. Using RESTORE WITH NORECOVERY, create the mirror database by restoring the full database backup and, optionally, the most recent differential database backup, onto the mirror server instance.

   Note

   If you restore the database filegroup by filegroup, be sure to restore the whole database.

   - Restore a Database Backup (SQL Server Management Studio)
   - RESTORE (Transact-SQL) and RESTORE Arguments (Transact-SQL).
7. Using RESTORE WITH NORECOVERY, apply any outstanding log backup or backups to the mirror database.
   - Restore a Transaction Log Backup (SQL Server)

Example (Transact-SQL)

Before you can start a database mirroring session, you must create the mirror database. You should do this just before starting the mirroring session.

This example uses the AdventureWorks sample database, which uses the simple recovery model by default.

1. To use database mirroring with the AdventureWorks database, modify it to use the full recovery model:

   Copy

   ```
   USE master;
   GO
   ALTER DATABASE AdventureWorks
   SET RECOVERY FULL;
   GO
   ```

2. After modifying the recovery model of the database from SIMPLE to FULL, create a full backup, which can be used to create the mirror database. Because the recovery model has just been changed, the WITH FORMAT option is specified to create a new media set. This is useful to separate the backups under the full recovery model from any previous backups made under the simple recovery model. For the purpose of this example, the backup file (C:\AdventureWorks.bak) is created on the same drive as the database.

   Note

   For a production database, you should always back up to a separate device.

3. On the principal server instance (on PARTNERHOST1), create a full backup of the principal database as follows:
4. Copy
5. `BACKUP DATABASE AdventureWorks`
6. `    TO DISK = 'C:\AdventureWorks.bak'`
7. `    WITH FORMAT`
8. `GO`
9. Copy the full backup to the mirror server.
10. Using RESTORE WITH NORECOVERY, restore the full backup onto the mirror server instance. The restore command depends on whether the paths of principal and mirror databases are identical.
    - If the paths are identical:

      On the mirror server instance (on PARTNERHOST5), restore the full backup as follows:

      Copy

      ```
      RESTORE DATABASE AdventureWorks
          FROM DISK = 'C:\AdventureWorks.bak'
          WITH NORECOVERY
      GO
      ```

    - If the paths differ:

If the path of the mirror database differs from the path of the principal database (for instance, their drive letters differ), creating the mirror database requires that the restore operation include a MOVE clause.

Important

If the path names of the principal and mirror databases differ, you cannot add a file. This is because on receiving the log for the add file operation, the mirror server instance attempts to place the new file in the location used by the principal database.

For example, the following command restores a backup of a principal database residing in C:\Program Files\Microsoft SQL Server\MSSQL.n\MSSQL\Data\ to a different location, D:\Program Files\Microsoft SQL Server\MSSQL.n\MSSQL\Data\, where the mirror database is to reside.

Copy

```
                    RESTORE DATABASE AdventureWorks
   FROM DISK='C:\AdventureWorks.bak'
   WITH NORECOVERY,
      MOVE 'AdventureWorks_Data' TO
         'D:\Program Files\Microsoft SQL
Server\MSSQL.n\MSSQL\Data\AdventureWorks_Data.mdf',
      MOVE 'AdventureWorks_Log' TO
         'D:\Program Files\Microsoft SQL
Server\MSSQL.n\MSSQL\Data\AdventureWorks_Log.ldf';
GO
```

11. After you create the full backup, you must create a log backup on the principal database. For example, the following Transact-SQL statement backs up the log to the same file used by the preceding full backup:

    Copy

    ```
    BACKUP LOG AdventureWorks
        TO DISK = 'C:\AdventureWorks.bak'
    GO
    ```

12. Before you can start mirroring, you must apply the required log backup (and any subsequent log backups).

    For example, the following Transact-SQL statement restores the first log from C:\AdventureWorks.bak:

    Copy

    ```
    RESTORE LOG AdventureWorks
    ```

```
    FROM DISK = 'C:\AdventureWorks.bak'
    WITH FILE=1, NORECOVERY
GO
```

13. If any additional log backups occur before you start mirroring, you must also restore all of those log backups, in sequence, to the mirror server using WITH NORECOVERY.

    For example, the following Transact-SQL statement restores two additional logs from C:\AdventureWorks.bak:

    Copy

    ```
    RESTORE LOG AdventureWorks
        FROM DISK = 'C:\AdventureWorks.bak'
        WITH FILE=2, NORECOVERY
    GO
    RESTORE LOG AdventureWorks
        FROM DISK = 'C:\AdventureWorks.bak'
        WITH FILE=3, NORECOVERY
    GO
    ```

For a complete example of setting up database mirroring, showing security setup, preparing the mirror database, setting up the partners, and adding a witness, see Setting Up Database Mirroring (SQL Server).

Setting Up Database Mirroring (SQL Server)

This section describes the prerequisites, recommendations, and steps for setting up database mirroring. For an introduction to database mirroring, see Database Mirroring (SQL Server).

Important
We recommend that you configure database mirroring during off-peak hours because configuration can impact performance.

In this Topic:

- Preparing Server Instances to Participate in Database Mirroring
- Overview: Establishing a Database Mirroring
- In This Section
- Related Tasks

Preparing a Server Instance to Host a Mirror Server

For each database mirroring session:

1. The principal server, mirror server, and witness, if any, must be hosted by separate server instances, which should be on separate host systems. Each of the server instances requires a database mirroring endpoint. If you need to create a database mirroring endpoint, ensure that it is accessible to the other server instances.

   The form of authentication used for database mirroring by a server instance is a property of its database mirroring endpoint. Two types of transport security are available for database mirroring: Windows Authentication or certificate-based authentication. For more information, see Transport Security for Database Mirroring and AlwaysOn Availability Groups (SQL Server).

   The requirements for network access are specific to the form of authentication, as follows:

   - If using Windows Authentication

     If server instances are running under different domain user accounts, each requires a login in the **master** database of the others. If the login does not exist, you must create it. For more information, see Allow Network Access to a Database Mirroring Endpoint Using Windows Authentication (SQL Server).

   - If using certificates

    To enable certificate authentication for database mirroring on a given server instance, the system administrator must configure each server instance to use certificates on both outbound and inbound connections. Outbound connections must be configured first. For more information, see Use Certificates for a Database Mirroring Endpoint (SQL Server).

2. Make sure that logins exist on the mirror server for all the database users. For more information, see Set Up Login Accounts for Database Mirroring or AlwaysOn Availability Groups (SQL Server).
3. On the server instance that will host the mirror database, set up the rest of the environment that is required for the mirrored database. For more information, see Manage Metadata When Making a Database Available on Another Server Instance (SQL Server).

Overview: Establishing a Database Mirroring Session

The basic steps for establishing a mirroring session are as follows:

1. Create the mirror database by restoring the following backups, using RESTORE WITH NORECOVERY on every restore operation:
    1. Restore a recent full database backup of the principal database, after making sure that the principal database was already using the full recovery model when the backup was taken. The mirror database must have the same name as the principal database.
    2. If you have taken any differential backups of the database since the restored full backup, restore your most recent differential backup.
    3. Restore all the log backups done since the full or differential database backup.

    For more information, see Prepare a Mirror Database for Mirroring (SQL Server).

    Important

    Complete the remaining setup steps as soon as you can after taking the backup of the principal database. Before you can start mirroring on the partners, you should create a current log backup on the original database and restore it to the future mirror database.

2. You can set up mirroring by using either Transact-SQL or the Database Mirroring Wizard. For more information, see one of the following:
    - Establish a Database Mirroring Session Using Windows Authentication (Transact-SQL)
    - Establish a Database Mirroring Session Using Windows Authentication (SQL Server Management Studio)
3. By default a session is set to full transaction safety (SAFETY is set to FULL), which starts the session in synchronous, high-safety mode without automatic failover. You can reconfigure the session to run in high-safety mode with automatic failover or in asynchronous, high-performance mode, as follows:

- High-safety mode with automatic failover

  If you want a high-safety mode session to support automatic failover, add a witness server instance.

  To add a witness

  - Add a Database Mirroring Witness Using Windows Authentication (Transact-SQL)
  - Establish a Database Mirroring Session Using Windows Authentication (SQL Server Management Studio)

  Note

  The database owner can turn off the witness for a database at any time. Turning off the witness is equivalent to having no witness, and automatic failover cannot occur.

- High-performance mode

  Alternatively, if you do not want automatic failover and you prefer to emphasize performance over availability, turn off transaction safety. For more information, see Change Transaction Safety in a Database Mirroring Session (Transact-SQL).

  Note

  In high-performance mode, WITNESS needs to be set to OFF. For more information, see Quorum: How a Witness Affects Database Availability (Database Mirroring).

  Note

  For an example of using Transact-SQL to set up database mirroring using Microsoft Windows Authentication, see Example: Setting Up Database Mirroring Using Windows Authentication (Transact-SQL).

  For an example of using to set up database mirroring using certificate-based security, see Example: Setting Up Database Mirroring Using Certificates (Transact-SQL).

In This Section

Prepare a Mirror Database for Mirroring (SQL Server)

Summarizes the steps for creating a mirror database or preparing a mirror database before resuming a suspended session. Also provides links to how-to topics.

Specify a Server Network Address (Database Mirroring)

Describes the syntax of a server network address, how the address identifies the database mirroring endpoint of the server instance, and how to find the fully-qualified domain name of a system.

Establish a Database Mirroring Session Using Windows Authentication (SQL Server Management Studio)

Describes how to use the Configure Database Mirroring Security Wizard to start database mirroring on a database.

Establish a Database Mirroring Session Using Windows Authentication (Transact-SQL)

Describes the Transact-SQL steps for setting up database mirroring.

Example: Setting Up Database Mirroring Using Windows Authentication (Transact-SQL)

Contains an example of all the stages required to create a database mirroring session with a witness, using Windows Authentication.

Example: Setting Up Database Mirroring Using Certificates (Transact-SQL)

Contains an example of all the stages required to create a database mirroring session with a witness, using certificate-based authentication.

Set Up Login Accounts for Database Mirroring or AlwaysOn Availability Groups (SQL Server)

Describes creating a login for a remote server instance that is using a different account than the local server instance.

Related Tasks

SQL Server Management Studio

- Start the Configuring Database Mirroring Security Wizard (SQL Server Management Studio)
- Establish a Database Mirroring Session Using Windows Authentication (SQL Server Management Studio)

Transact-SQL

- Allow Network Access to a Database Mirroring Endpoint Using Windows Authentication (SQL Server)
- Allow a Database Mirroring Endpoint to Use Certificates for Outbound Connections (Transact-SQL)

- Allow a Database Mirroring Endpoint to Use Certificates for Inbound Connections (Transact-SQL)
- Create a Database Mirroring Endpoint for Windows Authentication (Transact-SQL)
- Establish a Database Mirroring Session Using Windows Authentication (Transact-SQL)
- Add a Database Mirroring Witness Using Windows Authentication (Transact-SQL)
- Set Up a Mirror Database to Use the Trustworthy Property (Transact-SQL)

Transact-SQL/SQL Server Management Studio

- Minimize Downtime for Mirrored Databases When Upgrading Server Instances
- Prepare a Mirror Database for Mirroring (SQL Server)
- Troubleshoot Database Mirroring Configuration (SQL Server)

Establish a Database Mirroring Session Using Windows Authentication (SQL Server Management Studio)

Note
This feature will be removed in a future version of Microsoft SQL Server. Avoid using this feature in new development work, and plan to modify applications that currently use this feature. Use AlwaysOn Availability Groups instead.

To establish a database mirroring session and to modify the properties of database mirroring for a database, use the Mirroring page of the Database Properties dialog box.Before you use the Mirroring page to configure database mirroring, ensure that the following requirements have been met:

- The principal and mirror server instances must be running the same edition of SQL Server—either Standard or Enterprise. Also, we strongly recommend that they run on comparable systems that can handle identical workloads.

  Note
  A witness server instance is not available in every edition of Microsoft SQL Server. For a list of features that are supported by the editions of SQL Server, see Features Supported by the Editions of SQL Server 2012.

- The mirror database must exist and be current.

  Creating a mirror database requires restoring a recent backup of the principal database (using WITH NORECOVERY) on the mirror server instance. It also requires taking one or more log backups after the full backup and restoring them in sequence to the mirror database (using WITH NORECOVERY). For more information, see Prepare a Mirror Database for Mirroring (SQL Server).

- If the server instances are running under different domain user accounts, each requires a login in the **master** database of the others. If the login does not exist, you must create it before configuring mirroring. For more information, see Allow Network Access to a Database Mirroring Endpoint Using Windows Authentication (SQL Server).

To configure database mirroring

1. After connecting to the principal server instance, in Object Explorer, click the server name to expand the server tree.
2. Expand Databases, and select the database to be mirrored.

3. Right-click the database, select Tasks, and then click Mirror. This opens the Mirroring page of the Database Properties dialog box.
4. To begin configuring mirroring, click the Configure Security button to launch the Configure Database Mirroring Security Wizard.

   Note

   During a database mirroring session, you can use this wizard only to add or change the witness server instance.

5. The Configure Database Mirroring Security Wizard automatically creates the database mirroring endpoint (if none exists) on each server instance, and enters the server network addresses in the field corresponding to the role of the server instance (Principal, Mirror, or Witness).

   Important

   When creating an endpoint, the Configure Database Mirroring Security Wizard always uses Windows Authentication. Before you can use the wizard with certificate-based authentication, the mirroring endpoint must already have been configured to use certificates on each of the server instances. Also, all the fields of the wizard's Service Accounts dialog box must remain empty. For information about creating a database mirroring endpoint to use certificates, see CREATE ENDPOINT (Transact-SQL).

6. Optionally, change the operating mode. The availability of certain operating mode(s) depends on whether you have specified a TCP address for a witness.

7. When all of the following conditions exist, click Start Mirroring to begin mirroring:
   - You are currently connected to the principal server instance.
   - Security has been configured correctly.
   - The fully-qualified TCP addresses of the principal and mirror server instances are specified (in the Server network addresses section).
   - If the operating mode is set to High safety with automatic failover (synchronous), the fully-qualified TCP address of the witness server instance is also specified.
8. After mirroring begins, you can change the operating mode and save the change by clicking OK. Note that you can switch to high-safety mode with automatic failover only if you have first specified a witness server address.

   Note

   To remove the witness, delete its server network address from the Witness field. If you switch from high-safety mode with automatic failover to high-performance mode, the Witness field is automatically cleared.