Sql Server Mirroring Library

Automatic handling of mirroring in code and both check of need configuration and the setup of that. It is expected to be used from a host program or service that can load configuration, knows server state and run the library functions. The library handles multi-mirrored servers (servers that handles multiple mirrored databases for one application and all are needed for operation).

Contents

[1 Vocabulary/Definitions 1](#_Toc445637386)

[2 Limitations 1](#_Toc445637387)

[3 States 2](#_Toc445637388)

[3.1 Startup 2](#_Toc445637389)

[3.1.1 Primary Role 2](#_Toc445637390)

[3.1.2 Secondary Role 2](#_Toc445637391)

# Figures

[Figure 1: Action Install Principal - Dir/Share Create 2](#_Toc445637918)

# Vocabulary/Definitions

Documentation is tried kept in Microsoft Sql Server Mirroring vocabulary.

MirrorLib: Reference to the SQL Server Mirroring Library.

Database Server Instance: Named installation version of the SQL Server software. First installation typically have the default name SQLSERVER.

Principal Role: Database Role in mirroring which is the active server. This is the server that the external applications connect to and run queries. This is the database that is backed up to truncate the log and make recovery faster.

Mirror Role: Database Role in mirroring which is the mirroring server. The database is the one ready to fail over and is in restoring state to make additions faster.

Multi-mirror server: Server that has several mirrored databases that need to switch over at the same time.

Primary Role: Server Role where the server runs its mirrored databases in Principal Role. When the server shifts role from Primary to Secondary all the mirrored databases shift role from Principal to Mirrored.

Secondary Role: Server Role where the server runs its mirrored database in Mirror Role. When the server shifts role from Secondary to Primary all the mirrored databases shift role from Mirrored to Principal.

Library Host: The program or service that is running the library. It is required to be able to evaluate if the server runs in Primary Role or Secondary Role. The action in the library can be handled via scripting via the console test application (which is the running as an intermediary Library Host) or via direct use of the MirrorLib in code. The Library Host needs to be able to identify a server role and possible shifts and send that information to the MirrorLib.

Action: A task that handled by MirrorLib and communicated to the log so that in a debugging situation so it is possible to see what the server is doing. An action can consist of sub-actions if several things need to be done in an action. Actions on individual databases are done as sub-threads as they are able to be done in parallel.

Switch-over: The change of server role between primary/secondary.

Degraded state: State where the mirrored databases are not ready for another failover. The mirrored databases might be running. This state can be due to a switch-over or change in configuration.

# Limitations

1. Does not have its own monitoring service that can change the role of the server.
2. A database name is only allowed to consist of letters, numbers and \_.
3. Tested on
   1. Sql Server 2014 Standard Edition
   2. Windows Server 2012 R2 Standard Edition
   3. AD User for running the mirroing that has rights to
      1. backup and restore databases.
      2. setting up mirroring which is access to the master database on each server.
      3. has the right to create endpoints on the database server instance \*.
      4. has the right to start services \*.
      5. creating needed directories \*.
      6. setting security for directories \*.
      7. creation of shares \*.
      8. setting security for shares \*.

\* This can be handled manually if the installer does not want to give the rights to the user. The state of each is tested on startup and startup fails if not configured.

# Installation

## Prerequisites

### SMO

Servers need to have “Shared Management Objects” (SMO) installed which is included for Sql Server 2014 as redistributable packages in the sub-directory “External” for both 32bit and 64bit.

### Sql Server 2014

The database instance needs to have been installed before trying to setup mirroring.

### Existing database on the Primary Role server

The databases need to be existing on the Primary Role server and the Secondary Role server is being set up via restore of a backup from Primary Role server.

# States of Sql Server Mirroring

## Startup State [Degraded]

Startup is different depending on Server Role but have a common check for if the Sql Server Intance is ready for mirroring via **Action: Instance Ready form Mirroring**.

### Next states

Can switch to **Shutting Down State**, **Running State**, **Manual Failover State** or **Force Manual Failover State**.

### Action: Instance Ready for Mirroring

This action consists of several sub-actions which are run in sequence.

#### Action: Ready shares and directories

### Primary Role

#### Action: Setup of Primary Role

The action consist of several sub-actions. They are listed in the sequence they are called.

##### Action: Primary Role – Create Dir/Share

The process creates local directories, creates local share and checks access to directories locally via aread/write test. The read/write test is done by test creating a file to the location, reading it back and deleting it afterwards. In the configuration the different folders are names. Local share is “Share”. Local directories are “Local Share Directory”, “Local Restore”, “Local Backup” which both are absolute paths and “Local Transfer”, “Remote Transfer” and “Remote Delivery” which are placed under “Local Share Directory”. Read/write access to remote share “Remote Share” sub-directories “Local Transfer”, “Remote Transfer” and “Remote Delivery” are tested. Failure might be because of missing setup on Secondary Role server so if the read/write test fails it is noted as a warning in the log to avoid a “the hen or the egg” problem.

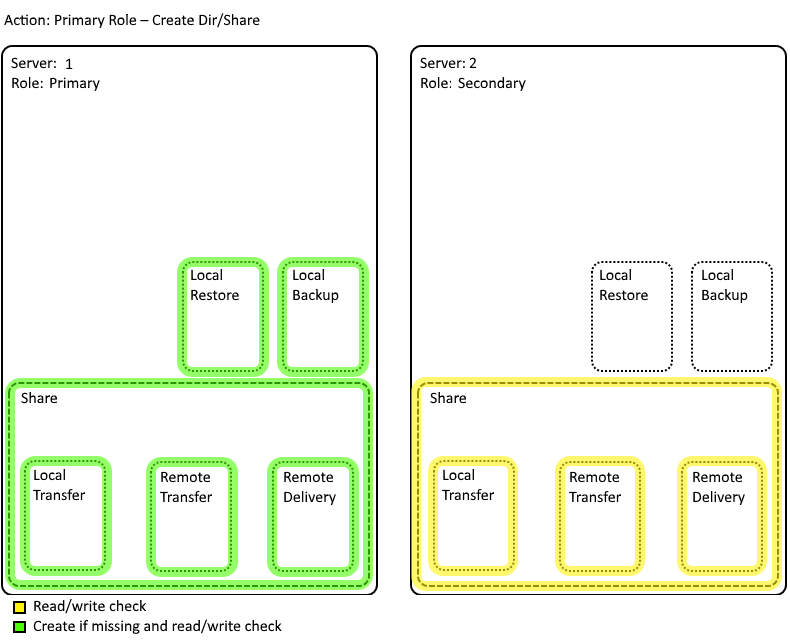


Figure 1: Action: Primary Role – Create Dir/Share

### Secondary Role

#### Action: Setup of Secondary Role

The action consist of several sub-actions. They are listed in the sequence they are called.

##### Action: Secondary Role – Create Dir/Share

The process creates local directories, creates local share and checks access to directories locally via aread/write test. The read/write test is done by test creating a file to the location, reading it back and deleting it afterwards. In the configuration the different folders are names. Local share is “Share”. Local directories are “Local Share Directory”, “Local Restore”, “Local Backup” which both are absolute paths and “Local Transfer”, “Remote Transfer” and “Remote Delivery” which are placed under “Local Share Directory”. Read/wirte access to remote share “Remote Share” sub-directories “Local Transfer”, “Remote Transfer” and “Remote Delivery” are tested. If the read/write test fails the startup fails as the connection is needed for mirroring.

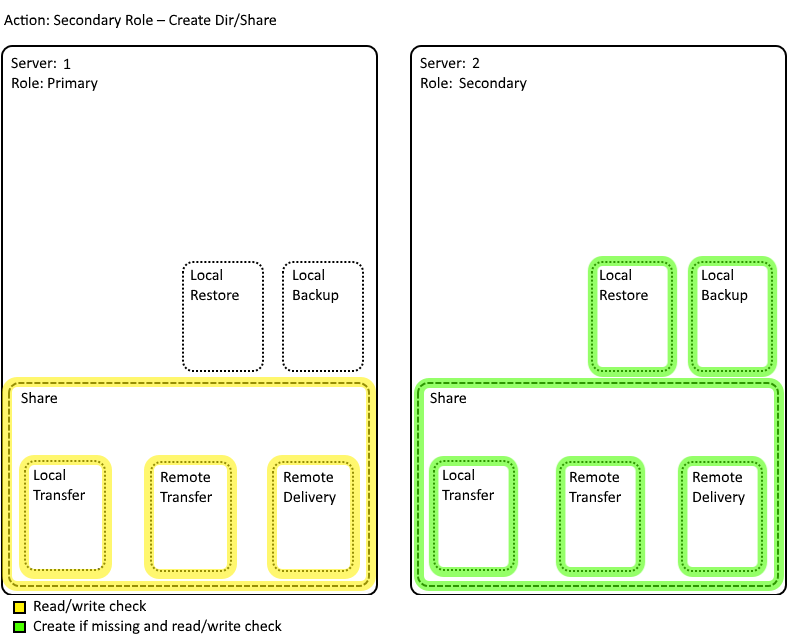


Figure 2: Action: Secondary Role – Create Dir/Share

### Individual databases

This part is mostly the same for both server Roles. Where they differs are in the Backup/Restore action.

If a database from the configuration is identified as being not set up for mirroring then the **Action: Add Database to Mirroring** is called. If a database on the instance database is identified as being set up for mirroring but is not in the configuration then the **Action: Remove Database from Mirroring**. Each of these actions are handled in a sub-thread as they can be run in parallel.

#### Action: Add Database to Mirroring

This action consist of a few sub-actions which is listed in the sequence they are done.

##### Create local database sub-directories

Create local database sub-directories for each local directory with database name to host the specific databases.

##### Primary Role server

###### Action: Backup database and move to remote share

Creates a backup of the Principal database for setting up the Mirror database on the Secondary Role server if needed.

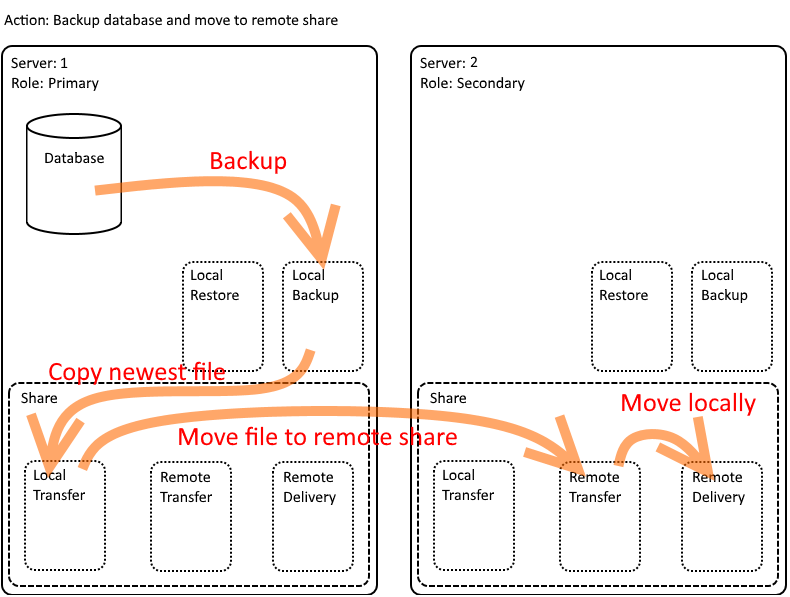


Figure 3: Action: Backup database and move to remote share

Action: Backup

A backup is made into the “Local Backup” sub-directory with the database name.

Action: Copy Backup to Local Transfer

Clears the “Local Transfer” database sub-directory and copies the newest Backup from “Local Backup” sub-directory to “Local Transfer” sub-directory.

##### Secondary Role server

###### Action: Move backup from remote directory to local and restore database

##### Action: Check Readiness for Mirroring

#### Action: Remove Database from Mirroring

This action consist of a few sub-actions which is listed in the sequence they are done.

##### Action: Disable Mirroring for Database.

##### Action: Remove Endpoint

##### Action: Remove database specific sub-directories

### Change state

Change into **Running State** if everything has run correct and into **Shutting Down State** if some steps fails.

## Running State [Full Mirroring]

### Next states

Can switch to **Shutting Down State**, **Maintenance State**, **Manual Failover State** or **Force Manual Failover State**.

## Shutting Down State [Degraded]

### Next states

Can switch to **Shutdown State**.

## Shutdown State [Degraded]

### Next states

Can switch to **Startup State** or **Maintainance State**.

## Maintenance State [Full Mirroring]

As such this state is as Running State but it is expected that external services are disabled. When shifted to this state from Running State on Primary Role server a backup is triggered on each of the mirrored databases. The state is used to update software which is always done on the Secondary Role server. If a database update fails all databases need to be restored. When the Secondary Role server has been upgraded switch over to the other server so

### Next states

Can switch to **Shutdown State**, **Startup State**, **Manual Failover State** or **Force Manual Failover State**.

## Manual Failover State [Full Mirroring]

### Next states

Can switch to **Shutting Down State**, **Maintenance State** or **Running State**.

## Force Manual Failover State [Degraded]

### Next states

Can switch to **Shutting Down State**.