MIRRORING. 06-09-2016

FOR MIRRIONG THE DB SHOULD BE IN FULL RECOVERY MODEL ONLY.IT IS DB LEVEL HIGH AVIALABILITY
DATABASE MIRRIORING
TERMS INVOLVED
PRINCIPAL – PRIAMRY SERVER IS CALLED PRINCIPAL
MIRROR—SECONDARY SERVER IS CALLED MIRROR
WITNESS—MONITOR SERVER IS CALLE D WITNESS(WITNESS IS NOT COMPULSORY IN MIRRORING)
END POINT

MORRORING IS THE AUTOMATIC FAILOVER OPTION.

NEW FEATURE OF SQL SERVER 2005, SP1(IT WILL WORK ON 2005 RTM, IF WE ENABLE ON IT, we have to enable trace flag 1400)
FEATURES
INSTANT SATND-by
VERY FAST FAIL OVER (3SEC) NO DATA LOSS
AUTOMATIC, TRASNPARENT CLIENT REDIRECT
AUTOMATIC RE-SYNC AFTER FAILOVER.

WITNESS WILL MONITOR PRINCIPAL AND SECONDARY SERVERS.
END POINT IS THE ENTRY OF THE SQ SERVER.
IN MIRRONG PERFORMANCE WILL BE KILLED.
THERE ARE 2 MODES
SYNCHRONOUS (NO DATA LOSS)(WAIT FOR ACKNOWLEDGEMENT)

AND ASYNCHRONOUS(MAY BE DATA LOSS IS THERE)WILL NOT WAIT FOR ACKNOWLEDGEMENT.
3 INSTANCES ARE COMPULSORY FOR MIRRORING.
IN MIRROR ONLY ONE ERROR 1418. ANY PROBLEM IT WILL GENERATE THIS ERROR ONLY. VERY TUFF TO FIND THE REAL PROBLEM WHY MIRROR FAILS.
WHO ARE HAVING DATABASES IS CALLED PARTNERS.
THAT MEANS PRINCIPAL AND MIRROR WILL BE PARTNERS IN THE MIRRIORING

MIRRORING DEMO

FOR MIRRORING NO AGENT INVOLVED

CREATE A DATABASE.

GO TO NODE2 CREATE ONE SHARED FOLDER, GIVE PRINCIPAL SRV ACCOUNT AND SECONDARY SRV ACCOUNT FULL RIGHTS OR GIVE EVERYONE FULL RIGHTS.
NOW GO TO NODE1 CREATE TABLE IN THE NEWLY CREATED DATABASE. INSERT RECORDS. BEFORE CONFIGURING MIRRORING TAKE FULL AND LOG BACKUP ON PRICIPAL DB AND RESTORE IT IN MIRROR

DB FOR BEST PRACTICE.

BACKUP DATABASE DBNAME TO DISK=N'\\10.10.10.2\SHAREDFOLDER\DBNAME_FULL.BAK'

LOG BACKUP

BACKUP LOG DBNAME TO DISK = N'\\10.10.10.2\SHAREDFOLDER\DBNAME_LOG.TRN'

IF WE DON'T HAVE OS PERMISSIONS TO CREATE SHARED FOLDER THEN WE HAVE TO GO WITH HIDDEN SHARE.

NODE1

GO TO RUN

\\GIVE THE IP ADDRESS OF WHICH NODE U WANT TO OPEN\NAME OF THE DRIVE U WANT TO OPEN$

\\10.10.10.2\c$

we can open the folder we can copy the backup files to the other nodes by sitting in the same node.

but we can not take backup to the hidden share.

or

Take backup to the local disk in the node1. Open the folders in which u take the backups. Then copy that files and paste it in the node2 (paste it in u r own folder) by using hidden share option.

then goto mirror server and restore the full and first log backups with norecovery.

--If we did not mention the path while taking backup by default the backup files will go to C:\programfiles\mssqlserver\mssql.namedinstanve\...\backup.

NOW CONFIGURE MIRRORING

GO TO THE DATABASE PROPERTIES, SELECT MIRROR, SELECT CONFIGURE SECURITY(MEANS CREATING ENDPOINTS.)

WITNESS U WANT –YES.

NEXT--

PRINCIPAL SERVER INSTANCE

HERE IT WILL CREATE ENDPOINT FOR PRINCIPAL SERVER

CHANGE THE NAME OF THE END POINT IF U WANT.

WE CAN CHANGE THE PORT NUMBER ALSO. MIRRORING WILL CREATE ITS OWN PORT NUMBER. THAT MEANS IT WILL CEATE ITS OWN CHANNEL, ONLY MIRRORING GUYS WILL ENTER THROUGH THAT CHANNEL.

BY DEFAULT PRINCIPAL SERVER INSTANCE MIRROR PORT NO IS :5022

CHECK THE ENCRYPT DATA SENT THROUGH THIS ENDPOINT.

--IF PRINCIPAL AND MIRROR , WITNESS ARE IN SAME MACHINE WE CANT GIVE SAME PORT NO FOR ALL THE THREE.

NEXT.

CONNECT TO MIRROR SERVER INSTANCE.

HERE ALSO SAME PORT NO WILL BE THERE Y BECAUSE NODE2 IS IN SEPARATE MACHINE.

CONNECT TO WITNESS SERVER, IF U WANT TO CHANGE THE NAME U CAN CHANGE IT.

SERVICE ACCOUNTS

--SQL SERVER ACCOUNTS IN THE SAME DOMAIN OR TRUSTED DOMAINS, OR ALL SERVERS SHOULD BE IN ONE DOMAIN.

GIVE THE SERVICE ACCOUNTS.

KDSSG\PRINCIPAL MAIN SERVICE ACCOUNT(CREATE ALL THESE SERVICE ACCOUTS IN AD)

KDSSG\MIRROR MAIN SERVICE ACCOUNT

KDSSG\WITNESS MAIN SERVICE ACCOUTN.—NEXT

CLICK FINISH

U CAN SEE THE NETWORK ADDRESS OF ALL THE THREE SERVERS

THEN U CAN START MIRRORING BY CLICKING ON 'START MIRRORING'

COMMON ERROR IN MIRRORING IS 1412 ERROR.

--in mirroring mirror database name should be same as principal database name.

--there will be only one endpoint on one server even u have configure mirroring on 10 databases. All mirror databases will use this end point only.

If we pause the mirroring it will stop transferring the logs from principal to mirror.

THERE ARE 3 OPERATING MODES IN MIRRORING

ASYNCHRONOUS – IT WILLNOT SEND ACKNOWLEDGEMENT TO THE PRINCIPAL SO IT WILL GIVE HIGH PERFORMANCE

SYNCHRONOUS—IT WILL SEND ACKNOWLEDGEMENT TO THE PRINCIPAL THAT MEAN HIGH SAFETY WILL BE THERE.

ANOTHER 'SYNCHRONOUS'(SECOND OPTION IN MIRRORING WINDOW) MEANS IF THERE IS NO WITNESS SERVER THEN THIS WILL BE USED. THIS IS AN HIGH PROTECTION OPTION.

FAIL OVER IN MIRRORING.

GO TO PRINCIPAL DB PROPERTIES, CLICK MIRRORING , THERE U CAN SEE FAIL OVER CLICK ON THAT.

BEFORE THAT MANUAL FAIL OVER, CHANGE THE MODE TO HIGH SAFETY.

IN CUI MODE

```
ALTER DATABASE DBNAME(PRINCIPAL DB NAME) SET PARTNER FAILOVER
```

WE CAN CONFIGURE ONLY ONE MIRROR DATABASE FOR ONE PRINCIPAL DATABASE.

END POINT

END POINT IS THE ENTRY POINT INTO SQL SERVER, WHICH MEANS SQL SERVER MAY COMMUNICATE OVER THE NETWORK.

BY DEFAULT 5 END POINTS WILL BE THERE IN AN INSTANCE.

PROTOCALS OF SQL SERVER TCP/IP, SHARED MEMORY, NAMED PIPE, VIA

WE CANT SEE VIA IN CONFIG MANAGER FROM SQL SERVER 2012.

PROTOCAL IS THE SET OF RULES.

EVERY PROTOCAL HAS ONE ENDPOINT.

```
SELECT *FROM SYS.ENDPOINTS
```

STOP END POINT IN CUI

```
ALTER ENDPOINT ENDPOINTNAME STATE=STOPPED
```

IF THE SQL SERVER AND SSMS BOTH ARE IN THE SAME MACHINE THEN THEY WILL USE SHARED MEMORY.

TO STOP THE SHARED MEMORY PROTOCAL

ALTER ENDPOINT ENDPOINT NAME (TSQL LOCAL MACHINE) =STOPPED
IF WE STOPPED TCP /IP AND SHARED MEMORY THEN WE CAN NOT CONNECT TO SQL SERVER. IN THIS CASE WE HAVE TO USE DAC TO CONNECT TO SQL SERVER.

STATES OF MIRRORING
1. SYNCHRONIZING
2. SYNCHRONIZED
3.SUSPEND (WHEN WE PAUSE THE MIRRORING WE CAN SEE THE SUSPEND MODE ON PRINCIPAL AND MIRROR DB)
4. DISCONNECTED(IF MY PRINCIPAL OR MIRROR SERVER DISCONNECTED OR STOPPED WE CAN SEE THIS)(IF MY PRINCIPAL OR MIRROR DB GOES TO IN RECOVERY THEN MIRROR CONFIGURATION BROKED, MEANS WE NEED TO CONFIGURE MIRRORING AGAIN
5.PENDING_FAILOVER(IF WE CLICK ON FAILOVER THEN MY PRINCIPAL BECOME MIRROR , MY MIRROR BECOMES PRINCIPAL, WHILE CHANGE THIS STATES WE CAN SEE PENDING_FAILOVER. WE CANT SEE THIS BECAUSE VERY QUICKLY FAIL OVER WILL APPLY.
6. CONNECTED
ONCE WE CONFIGURE MIRRORING IF WE GOTO PRINCIPAL DB, R.CLICK, THERE U CAN SEE "LAUNCH DATABASE MIRRORING MONITOR"(IT'S A TOOL TO TRACE ALL THE TRANSFERRED DATA)(MEANS MIRROR MONITORING)
UNSENT LOG: HOW MANY LOGS CAN NOT SENT FROM PRINCIPAL TO MIRROR.
OLDEST UNSENT TRANSACTON:HOW MUCH DATA IN TIME IS UNSENT(MEANS 10 MINS OF DATA UNSENT)
TIME TO SEND LOG: HOW MUCH TIME IT WILL TAKE TO SEND DATA TO THE MIRROR.
CURRENT SEND RATE: AT WHAT SPEED DATA TRANSFERD FROM PRINCIPAL TO MIRROR.
ALL THIS INFO WILL BE THERE IN LOG FILE ONLY. WE CAN WARN(CREATE ALERTS) IF ANYTHING FAILS IN DATA TRANSFER)

--IN SYNCHRONOUS MODE AUTOMATIC FAIL OVER IS THERE, IN ASYNCHRONOUS MODE ONLY MANUAL FAIL OVER IS THERE.

--WE CAN CHANGE WITNESS INSTANCE AT ANYTIME EVEN THE MIRRORING IS GOING ON. JUST REMOVE THE INSTANCE AND ADD THE NEW INSTANCE NAME, BUT FOR PRINCIPAL AND MIRROR WE CAN NOT CHANGE THE INSTANCE WITHOUT BREAKING THE MIRRORING.

MIRROR MONITOR
R.CLICK ON MIRROR DATABASE TASKS-- LAUCH MIRROR MONITOR.
***AT THE BOTTOM WE CAN SEE "MIRROR COMMIT OVERHEAD"

MIRROR COMMIT OVERHEAD
SPECIFIES THE NUMBER OF MILLISECONDS OF AVERAGE DELAY PER TRANSACTION TOLERATED BEFORE A WARNING I S GENERATED ON THE PRINCIPAL SERVER.
THAT MEANS HOW MUCH TIME IT WILL WAIT FOR THE ACKNOWLEDGENT. PRINCIPAL WILL WAIT

ACKNOWLEDGEMENT FROM MIRROR FOR HOW MUCH TIME. THIS OPTION WILL BE ONLY IN SYNCHRONOUS MODE. GOTO WARNINIG IN THE MIRROR MONITOR WINDOW. GO TO WARNINGS TAB. THERE AT THE END WE CAN SEE THE LAST OPTION MIRROR COMMIT OVERHEAD THRESHOLD. BY CLICKING ON THE SET THRESHOLD TAB WE CAN CHECK THE THRESHOLD AND WE CAN SET THE VALUE IN MILLISECONDS. WE HAVE TO DO THIS ACTION IN PRINCIPAL DATABASE AND WE HAVE TO CHECK FOR MIRROR AND WITNESS. IF WE CHECK THOSE OPTIONS THEN WHILE FAIL OVER NO NEED TO DO THE SAME ACTION. SO BETTER CHECK THE MIRROR AND WITNESS. THAT MEANS THESE OPTIONS WILL WARN WITH THE HELP OF ALERTS. SO IF WE CHECK THIS THEN AGENT SHOUD BE RUNNIG TO SEND THE ALERTS .

SCENARIOS

1474 ERROR OR 1486 ERROR
IF WE GET 1474 ERROR FIRST WE HAVE TO CHECK WHETHER ENDPOINTS ARE STARTED OR STOPPED. IF END POINT STOPPED OR IF THERE IS NO CONNECT PERMISSION TO THE PRINCIPAL SERVICE ACCOUNT ON MIRROR ENDPOINT. THEN ONLY WE WILL GET THIS ERROR.
IF ANY ERROR CAME FIRST CHECK THE MIRROR MONITOR. REFRESH IT ONCE(F5). THEN U CAME TO UNDERSTAND THE PROBLEM. THEN OPEN SQL SERVER LOGS THERE WE CAN SEE THE ERROR NO.S AND SOME INFO REGARDING ERROR.

SELECT * FROM SYS.ENDPOINTS (TO SEE THE ENDPOINTS WHETEHER STARTED OR STOPPED)
ALTER ENDPOINT ENDPOINT NAME STATE=STOPPED(WE CAN STOP THE ENDPOINT IN THIS WAY. TO START THE ENDPOINT SAME COMMAND AT THE END STATE=STARTED)

COMMAD FOR FAILOVER
ALTER DATABASE DATABASE NAME(WHICH IS PARTICIPATING MIRROR) SET PARTNER FAILOVER

IF SERVICE ACCOUNTS LOST PERMISSION ON MIRROR ENDPOINTS THAT MEANS SERVICE ACCOUNT DOES NOT HAVE CONNECT PERMISSION ON MIRROR ENDPOINT THEN 1418 ERROR WILL COME.
REMOVING CONNECT PERMISSION OF A SERVICE ACCOUNT.
GOTO LOGINS THERE WE CAN SEE THE SERVICE ACCOUNTS LOGINS. CLICK ON THE SRV ACCOUNT OF PARTICULAR NODE.GO TO PROPERTIES, CLICK ON SECURABLES. SELECT END POINT, AT THE BOTTOM WE CAN SEE THE PERMISSIONS THERE WE CAN SEE CONNECT PERMISSION. IF WE UNCHECK THAT PERMISSION THEN WE REMOVE THE CONNECT PERMISSION ON THE ENDPOINT. ONCE U RESTART THE INSTANCE THEN ONLY WE CAN SEE THE ISSUE IF WE DID NOT RESTORE NOTHING WILL HAPPENED DONE TRANSFER WILL DONE WITHOUT ANY ERRORS. IF WE RESTART THEN ONLY ERROR WILL COME.
COMMAND TO UN CHECK CONNECT PERMISSION
REVOKE CONNECT ON ENDPOINT::ENDPOINT NAME FROM SERVICE ACCOUNT NAME
IF WE WANT TO GIVE CONNECT PERMISSION ON ENDPOINT
GRANT CONNECT ON ENDPOINT::ENDPOINT NAME TO FULL SERVICE ACCOUNT NAME (LIKE

KDSSG\SQLPRI)

--MIRRORING CAN CONFIGURE IN TWO STAND BY ALSO LENGTHY AND TOUGH PROCEDURE . USING CERTIFICATES WE CAN CONFIGURE THIS. DOMAIN IS NOT COMPULSARY. BUT MAX CASES DOMAIN ACCOUNTS WILL BE USED.

VERIFY PORTS
USE TELNET TO TEST THAT THE PORT IS OPEN AND SOMETHING IS LISTENING.
OPEN CMD RUN AS ADMIN
TELNET IPADDRESS OF THE MIRROR END POINT PORT NO.
TELNET 10.10.10.2 5022
IF TELNET OPENS AND CURSOR BLINKING THEN PORT IS CONNECTED. NO ISSUES WITH PORTS. IF IT THROWS ERROR LIKE PORT IS NOT CONNECTED THEN PROBLEM WITH PORTS ONLY.
IF TEL NET COMMAND NOT WORKING INSTALL TELNET CLIENT FEATURE FROM SERVER MANAGER.
IF WE GET 1418 ERROR WE HAVE TO CHECK ALL THE ABOVE SCENARIOS ALONG WITH THAT FOLLOW THE BELOW STEPS
CHECKING PERMISSIONS THROUGH COMMANDS

SELECT *FROM SYS.ENDPOINTS

SELECT *FROM SYS.SERVER_PERMISSIONS WHERE CLASS_DESC='ENDPOINT' (U CAN GIVE THE LOGIN NAME ALSO IN THE PLACE OF ENDPOINT)

NETSTAT IS ONE COMMAND TO KNOW WHETHER CONNECTIONS ESTABLISHED BETWEEN THE NODES
NETSTAT –abn   --you can see all the protocals which is talking to which ip and port numbers.
netstat –abo -- you can also check the same by using this command.

--logship will work sqllogship.exeit is located in versions number 110—tools—binn we can find that exe file.
--sql engine will do mirroring. No exe file is thre for mirroring.
scenarios
stop the witness server (means stop sql server in witness) and stop the prinicipal also
Then try to failover to the mirror instance
alter database dbname set partner failover
if it is not working do the force fail over
alter database dbname set partner force_service_allow_data_loss
if force fail over also not happening then we have to break the mirror(This is the last option)
alter database dbname set partner off
then mirror will break and bring the mirror db online by restoring with recovery
once we make mirror db as principal no need to inform to application team why because application

itself smart enough which is mirroring and which is principal. With the help of client side redirection it will automatically switched to the mirror server.

To pause the mirroring by using the command
alter database dbname set partner suspend
To resume
alter database dbname set partner resume
changing mirroring from synchronous to asynshronous

safety {full or off}(not the full command need to verify)

Full means synchoronous, off menas asynchronous

QUORUM IN MIRRORING
QUORUM IS SIMPLY CONNECTION BETWEEN THE NODES. QUORUM IS A RELATIONSHIP THAT EXISTS WHEN TWO OR MORE SERVER INSTANCES IN A DATABASE MIRRORING SESSION ARE CONNECTED TO EACH OTHER.
THREE TYPES OF QUORUMS
1. A FULL QUORUM INCLUDES BOTH PARTNERS AND THE WITNESS
2. A WITNESS TO PARTNER QUORUM
3. PARTNER TO PARTNER QUORUM

ADDING A FILE ON MIRRORING
CREATE A FOLDER IN THE C DRIVE OF PRINICIPAL SERVER
GOTO PARTICULAR DB WHICH IS PARRICIPATING IN MIRRORING, R.CLICK, FILE ADD ONE FILE GIVE THE PATH INTO THE NEW FOLDER THAT IS CREATED RECENTLY. ONCE WE ADDED ANY FILE TO DB THEN THE MIRRORING WILL BREAK.THAT DATABASE WILL GO INTO SUSPENDED MODE.
SOLUTION IS FIRST WE HAVE TO CHECK WHY MIRROR BREAKS , OPEN SQL SERVER ERROR LOG AND CHECK THE ERROR
IN THE SAME TIME GOTO MIRROR INSTANCE ERROR LOG AND CHECK WHAT IS THE EXACT REASON.
THERE U CAN FIND THE REASON
THEN FIRST BREAK THE MIRRORING
ALTER DATABASE DBNAME SET PARTNER OFF , U NEED TO EXECUTE THIS STEP IN MIRROR INSTANCE.
SO TAKE LOG BACKUP OF PARTICULAR DB ON PRINCIPAL INSTANCE.C OPY THE FILE AND MOVE IT TO MIRROR NODE AND RESTORE IT IN MIRROR INSTANCE WITH MOVE OPTION.
RESTORE LOG DBNAME FROM DISK= ‘LOGBACKUP.TRN’ WITH MOVE ‘FILE NAME(WHICH U ADDED)’ TO ‘C:PRORGRAM FILE\PATH WHERE MY MIRROR DB IS THERE \NEWFILENAME’ WITH NORECVERY
--IN MIRRORING WE CAN NOT PUT THE DB IN STANDBY MODE.
--IF ANY PAGE CORRUPTED IN PRINCIPAL DATABASE THE RESPECTIVE OTHER PARTNER MEANS MIRROR

SERVER WILL REPAIR THAT PAGE. IF THE PAGE IS CORRUPTED IN MIRROR PRINCIPAL WILL REPAIR IT. THIS FEATURE STARTING FROM 2008.

IN THE ABOVE SCENARIOS THE RESPECTIVE PARTNER WILL REPLACE THE CORRUPTED PAGE. AT THAT WE CAN SEE BELOW ERRORS.

ERRORS
823,824,829

SNAPSHOTS
WE CANT READ DB IF THE DB IS IN RESTORING MODE. SO TO READ THAT DB WE WILL USE SNAP SHOT FEATURE . IN

DATABASE SNAPSHOTS(STARTED FROM 2005)
NO GUI OPTION TO CREATE SNAPSHOT , ONLY CUI
SNAPSHOT IS NOTHING BUT A READ ONLY COPY OF DATABASE.

CREATING SNAP SHOT
CREATE DATABASE DBNAME_TIME
(NAME=LOGICAL NAME, FILENAME='PHYSICAL PATH\DATABASENAME_DATE.SS')AS SNAPSHOT OF DATABASENAME(ON WHICH U R CREATING SNAPSHOT)

.SS IS SPARSE FILE CONTAINS THE ORIGINAL PAGE MEANS IF ANY PAGE CHANGED IN THE ORIGINAL DB THE SPARSE FILE WILL HOLD THE OLD COPY OF THE DB WHICH IS BEFORE MODIFIED.

NET MATERIAL

By default in the RTM (released to manufacturing) version of SQL Server 2005, database mirroring is not enabled, nor is it supported. To enable it to run on the RTM version of SQL Server 2005, you need to enable trace flag 1400. Database mirroring is supported and enabled by default starting in SP1.

Database requirements

While you can mirror from one database to another on different instances, you cannot mirror to:

- The same instance
- System databases
- A database with a different name (for example, you cannot mirror from the pubs database to the NorthWind database)
- Databases that are not in the full-recovery model

- A database that is not a restored copy of the source database (called the principal database), and this database must be restored using the no recovery parameter

Database mirroring is not a good choice for mirroring large numbers of databases on a single instance or server. Even though Microsoft has done functional testing on hundreds of mirrored databases per instance, it recommends around 10 databases be mirrored on a single mirrored database instance. Each database instance consumes a couple of worker threads on both the principal and the mirror, and with large numbers of mirrored database instances, the SQL Server system can quickly run out of worker threads to perform work on the server.

Server requirements

Here are considerations for server software, SQL Server versions, and hardware configuration and layout.

- Ideally, the principal and mirror server will consist of the same hardware configuration and OS, SQL Server version and service pack level.
- The disk layout should be the same. If you do an alter database statement to add a file to a filegroup and this file path refers to a non-existent path of the mirror, the mirror will fail.
- While you can mirror from a 32-bit version of SQL Server to a 64-bit version of SQL Server and vice versa, you cannot mirror between different editions of SQL Server (for example, from Enterprise Edition to Standard Edition). The exception is that you can mirror from the Enterprise to the Developer Edition because both have the same engine (the Enterprise Engine). However, the Developer Edition is not licensed for production use.
- Database mirroring will apply committed transactions that occur on the principal on the mirror. This is termed the "redo process." In the SQL Server Enterprise Edition, the redo process is multi-threaded, meaning the mirror will failover faster on the Enterprise Edition than if the mirror was running on the Standard Edition.
- With SQL Server 2005 Enterprise and Developer editions, you can create a database snapshot from a mirror and access this mirror for reporting purposes, for example. In all other cases, the mirror will be inaccessible.
- Database mirroring supports two modes – synchronous and asynchronous. Using asynchronous mode, a transaction will be committed on the principal before it will be committed on the mirror. Using synchronous mode, the transaction will be committed on the mirror before it is committed on the principal -- with some very important caveats. Before we look at what those are, we need to discuss which versions support which modes and when to use each mode.

Asynchronous mode/high performance

Asynchronous mode is only supported on the Enterprise and Developer editions of SQL Server 2005. Asynchronous mode is where transactions originate on the principal and are applied asynchronously on the mirror as soon as possible. This is also known as high-performance mode. Should the mirror go offline, users will not detect any changes in the state of the principal and, hence, continue to work as before the mirror went offline. Commands to be mirrored will be stored in the transaction log. Note: The witness does not play a role in asynchronous mode.

Synchronous mode

There are two forms of synchronous mode -- high availability and high protection.

High availability

High availability requires a witness server in addition to a principal and a mirror server. It will provide for automatic failover. Should the principal be unable to achieve a quorum with the witness and the mirror (in other words, it cannot connect to the mirror and the witness), then the database will shut down and:

- All users will be disconnected from the database.
- No transaction will be processed in the database.
- No new users will be allowed in the database.

As soon as it achieves a quorum again – when it sees the mirror or principal – it will start serving the database again. However, if the principal goes offline, the mirror may switch roles and now be the principal. You will have to fail back to get the original principal serving the database again, if this is what you require.

High protection

High protection does not require a witness server. If you are using high-protection mode and the principal is unable to achieve a quorum with the mirror – i.e., it cannot connect to the mirror – then it continues to serve the database and the loss of the mirror is transparent to the users connected to the principal. Users will not be disconnected, and they will be able to do work. New users can connect and transactions to be mirrored are stored in the transaction log.

Network requirements

While it is possible to mirror over the Internet, it is not a good practice for these reasons:

- Under any appreciable load, database mirroring latencies will increase and your exposure to data loss will increase.
- Should you be running high safety mode, your database mirroring session will never reach the synchronized state and you will be unable to failover and fail back.
- Should you be running high safety mode and your principal is unable to connect to the witness or mirror, database mirroring will be disconnected and the principal database will enter a read-only state.
- In high-performance mode, the transactions that are not mirrored to the mirror will remain in the log and the transaction log can get very large if the mirror is offline for an appreciable length of time. You'll be unable to dump the log.
- You will need to use certificates to encrypt authentication.

As database mirroring latency is very sensitive to network speed, make sure your network bandwidth – reliability and resolution – are as optimal as possible. It is a good practice to copy files back and forth between your principal and mirror to test network reliability, speed and throughput before you implement a database mirroring topology. Microsoft recommends that the network bandwidth is three times the maximum log generation rate.

For low-bandwidth connections, you can set the partner timeout and partner redo_queue to larger values. Consult this Alter Database Statement for more information on these settings.

Workload considerations

While running in high-availability and high-protection modes, there will be added latency to each transaction originating on the principal, because

it has to be mirrored over to the mirror before it is committed. The latency of short-running transactions will increase dramatically due to this split write. For example, transactions taking 10 ms, will now take 20 ms, i.e., twice as long (these numbers are purely for illustrative purposes, your results may vary).

On the other hand, the proportionate increase of long-running transactions will be considerably less. A one-second transaction, will now take 1.01 seconds. Long-running transactions will perform better with the high-protection/high-availability settings than short running-transactions do. Microsoft recommends that you open up more connections running in high-protection/high-availability modes to minimize the impact of the increased latency as well.

Witness requirements

There are no special requirements for the witness server. It should be connected via a well-connected network link. It does not have to be clustered because mirroring topologies requiring a witness can form a quorum with the witness being offline for brief periods.

QUESTIONS

- Do the principal and the mirror need to be on the same version?
  - You can have the principal be one version – say 2008R2 – and the mirror another – say 2012. This is how an upgrade with minimal downtime can be accomplished using mirroring! However, once you fail over to the mirror, you can no longer fail back.
- Do the principal and the mirror need to be on the same edition?
  - To be fully supported by Microsoft, yes.
- Can multiple databases on the same instance be in mirroring sessions?
  - Yes, but there are limitations, based on the hardware of your servers.
- How do I mirror the system database (master, model, msdb, tempdb)?
  - You can't! This isn't supported.
- If I set up mirroring, do I still have to take full, differential, or transaction log backups?
  - Yes! Mirroring increases the availability of your databases. It is not a substitute for regular backups, however.
- How does index maintenance (rebuilds) affect the mirror?
  - Transactions that fill up the log can affect the performance of mirroring. When the amount of information in the log increases, the amount of information that needs to be sent to and committed to the mirror increases also. If performance is crucial, you may want to do index maintenance more frequently, so it takes less time. If large transactions such as batch inserts are affecting performance, break those into smaller transactions.

- How can we configure mirroring between servers which are in different domains you can use certificates for security between domains.