

DBA TOPICS

FUNDAMENTALS

1. INTRODUCTION	--- 4
2. EDITIONS	--- 7
3. VERSIONS	--- 10
4. INSTALLATION PRE-REQUIREMENTS	--- 11
DIFFERENCE B/W DELETE AND TRUNCATE	--- 12
5. INSTALLATIONS	--- 14
SILENT INSTALLATIONS,	
UNINSTALATIONS	--- 25
6. SERVICE PACKS	--- 30
7. FILE & FILE GROUPS	--- 34,37
8. PAGES & EXTENTS	--- 40,43
9. DATA FILE & LOG FILE ARCHITECUTRE	--- 44,46
10. SECURITY	--- 55
ROLES & PERMISSIONS	
1. INSTANCE LEVEL	--- 58
2. DATABASE LEVEL	--- 61
3. OBJECT LEVEL	--- 65
SECURITY HARDENING RULES	--- 66
11. RECOVERY MODELS	--- 73
12. BACKUPS	--- 77
13. RESTORE & RECOVERY PROCESS	--- 88, 90
14. PIECE MEAL RESTORATION	--- 100
15. DATABASE REFRESH	--- 105
16. POINT IN TIME RECOVERY	--- 107
17. JOBS & MAINTENANCE PLANS	--- 108,110
18. ATTACH & DETACH DATABASE	--- 113
19. COPY DATABASE WIZARD	--- 115
20. SYSTEM DATABASES OVERVIEW	--- 116
21. SUSPECT DATABASE	--- 120
22. SYSTEM DATABASE CORRUPTIONS	--- 122
23. FILE MOVEMENTS SYSTEM & USER DATABASES	--- 126
24. IMPORT/ EXPORT	--- 131
25. DB MAIL CONFIGURATION	--- 133
26. LITE SPEED	--- 138

27. SHRINKING OPERATION	--- 152
28. UPGRADATION & MIGRATION	--- 154
29. TEMPDB FULL	--- 156
30. LOG FILE FULL	--- 159
31. RESOURCE GOVERNER	--- 160
32. POLICY BASED MANAGEMENT	

HIGH AVILABILITY

1. LOG SHIPPING ----- DATABASE LEVEL	--- 163
LOG SHIPPING SCENARIOS:	--- 170
2. DB MIRRORING ----- DATABASE LEVEL	--- 188
DATA BASE MIRRORING SCENARIOS:	--- 196
3. REPLICATION ----- OBJECT LEVEL	--- 210
REPLICATION SCENARIOS:	--- 234
4. CLUSTERING ----- INSTANCE LEVEL	--- 246

PERFORMANCE TUNNING

1. LOCKS	--- 252
2. BLOCKINGS	--- 255
3. DEAD LOCKS	--- 258
4. INDEXES	--- 261
5. FRAGMENTATION	--- 270
6. ISOLATION LEVELS	--- 274
7. SWITCHES	--- 282
8. LOCK PAGES IN MEMORY	--- 284
9. DAC[DEDICATED ADMIN CONNECTION]	--- 285
10. CDC [CHANGE DATA CAPTURE]	--- 288
11. WAITING TYPES	
12. SQL ARCHITECTURE	
13. PROFILER & PERFMON TOOLS	

14. DTA[DATABASE TUNING ADVISOR]	
15. ACID PROPERTIES	--- 325
16. WINDOWS TASK SCHEDULAR	--- 326
17. QUERY TUNING	--- 327
18. HIGH CPU ISSUE & MEMORY ISSUE	--- 330,334
19. UPDATE STATISTICS	--- 335
20. ACTIVITY MONITOR	--- 336
21. EXECUTION PLAN	--- 341
22. RAID LEVELS	--- 342
23. TEMPDB ARCHITECTURE	--- 345
24. SQL SERVER AUDITING	--- 349
25. NEW FEATURS LIST	--- 363
26. DMV'S & SP & DBCC	--- 365, 369, 371
27. ALWAYS ON HIGH AVILABILITY	
28. REAL TIME CLASSES	--- 378

INTRODUCTION

SQL SERVER DBA

> MS SQL Server is a Relational database server used on Web Servers to access information. Microsoft SQL Server is a database platform for large-scale online transaction processing (OLTP), data warehousing, and e-commerce applications; it is also a business platform for data integration, analysis, and reporting solutions.

- RDBMS stands for Relational Database Management System. RDBMS data is structured in database tables, fields and records. Each RDBMS table consists of database table rows. Each database table row consists of one or more database table fields.

Most popular RDBMS products are:

SQL Server [MSSQL]

Oracle

DB2

MYSQL

SYBASE

Why SQL Server is better than any other RDBMS Applications?

- Easy integration with Microsoft Operating systems.
- Easy integration with world's most common database: **[Import\Export]** Spreadsheet, Microsoft excel in particular and power pivot has significantly enhanced its value.
- User friendly interface [Not required to write queries]
- Easy to create maintenance plans.
- Integrated Security (windows authentication)
- Disaster recovery
- Licensing
- SQL Server Business Intelligence –One of the best for reporting to business people
- Administering and Monitoring
- Data Encryption
- Easy Availability
- Perfect suite of application – good clubbing and packaging of Database engine, Agent Service, Notification Service, Reporting Service, Analysis Service, Integration Service.
- Uncomplicated installation process

- BOL – help documentation is easily available and more friendly to browse to get the correct help
- Perfect match for all level of organizations... small firms to big enterprise to data store.

Facts in a market: Business mainly targets on below facts?

- Size of database
- Security of data
- Speed and concurrency
- Reliability

Having said that, if we check the real world scenarios and implementations we will find these facts:

- **Size of database:** According to the 2005 Survey of Winterport [European bank], the largest SQL Server DW database is the **19.5 terabytes**.
- **Security of data:** “Microsoft beats Oracle in security showdown. The website clearly says “Microsoft patched 59 vulnerabilities in its SQL Server 7, 2000 and 2005 databases during the period, while Oracle issued 233 patches for software flaws in its Oracle 8, 9 and 10g databases”
- **Speed and concurrency:** SQL Server 2005 system that **handles 5,000 transactions per second** and **100,000 queries a day** and can **scale up to 8 million new rows of data per day**, Is still required more performance? no
- Last but not least, search on any job site for “Oracle” and “SQL Server”, you will find more jobs for SQL Server than Oracle, why? Because more companies implement SQL Server than Oracle. For example, I searched on sites for last 7 days jobs. I found **2143 Jobs for SQL Server** and **1867 jobs for Oracle**.

Multiples of bytes:

- Kilobyte (KB)
- Megabyte (MB)
- Gigabyte (GB)
- Terabyte (TB)
- Petabyte (PB)
- Exabyte (EB)
- Zettabyte (ZB)
- Yottabyte (YB)

EDITIONS

SQL Server Editions:

- > The main purpose is to classify the features.
- > To reduce the cost as per the customer requirement

Types of editions:

- Developer Edition
- Enterprise Edition
- Standard Edition
- Workgroup Edition
- Express Edition
- Evaluation Edition

-
- We have some more editions available in market, those are
 - Express Edition with advanced services
 - Compact Edition(used for mobile applications)
 - Data Center Edition
 - Embedded Edition
 - Fast track Edition
 - Web Edition(used for web applications and web servers)
 - Azure Edition

Enterprise edition is ideally suited for the following usage scenarios:

- Mission critical deployments requiring high availability and uptime
- Existing large scale OLTP deployments
- OLTP deployments that expect to grow rapidly in the future
- Large scale reporting and analysis deployments

Standard edition is ideally suited for the following usage scenarios:

- Small to medium scale OLTP deployments
- OLTP deployments that are not expected to rapidly grow in the future
- Reporting and analysis deployments

Web edition is ideally suited for the following usage scenarios:

- Small scale OLTP deployments
- OLTP deployments that are not expected to slightly grow in the future
- Reporting and analysis deployments with limited features.

Developer Edition:

- It is especially used for R & D. It is a fully featured version. No limit for the processors and RAM. We can't use it for business purposes.

Express edition: It is purely developer's edition and free.

Evaluation Edition: It is a trail version with a validity of 180 days.

SQL Server 2012:

1. Removed datacenter and work editions in SQL Server 2012
2. From 2012 SQL supporting core based license but not processor based...

Note: Upto Sql 2008 R2 only processor based model...

3. Sql 2012, standard edition support server and cal + per core base both models. In this models UN limited users can connect to database.

4. For enterprise edition, Microsoft offering only "per core basis" but not "per server +CAL" models.

VERSIONS

SQL Server DBA Versions:

Micro Soft SQL Server Versions List:-

<http://sqlserverbuilds.blogspot.in>

Year	Version	Code Name
1989	SQL Server 1.0	NA
1990	SQL Server 3.0	NA
1992	SQL Server 4.2	NA
1993	SQL Server 4.21	NA
1995	SQL Server 6.0	SQL95
1996	SQL Server 6.5	Hydra
1999	SQL Server 7.0	Sphinx
2000	SQL Server 2000 32-bit	Shiloh
2003	SQL Server 2000 64-bit	Liberty
2005	SQL Server 2005	Yukon
2008	SQL Server 2008	Katmai
2008	SQL Server 2008 R2	Kilimanjaro
2012	SQL Server 2012	Denali

INSTALLATIONS PRE-REQUIREMENTS

Installation Pre-Requirements:

Versions	O\S	Windows Server 2000
	Memory (RAM) ¹	128 MB minimum on Windows XP 64 MB minimum on Windows 2000 32 MB minimum on all other operating systems
SQL Server 2000	Hard disk space	250 MB
	.NET Framework	1.1
	Processor type	Intel or Pentium1 [Speed: 166 MHz or higher.]
	Windows Installer	2
SQL Server 2005	O\S	Windows Server 2003
	Memory (RAM) ¹	512 MB
	Hard disk space	1.6 GB
	.NET Framework	2
	Processor type	Pentium III compatible processor or higher required [SPEED 600 MHz or higher]
	Windows Installer	3.1
SQL Server 2008	Memory (RAM) ¹	Min 512 MB: Recommended: 2.048 GB
	Hard disk space	2.0 GB
	.NET Framework	3.5
	Processor type	Itanium processor or faster [1.0 GHz or faster]
	Windows Installer	4.5

SQL Server 2008 R2	O\S	Min>> Windows Server 2003 SP2
	Memory (RAM) ¹	Min 512 MB: Recommended: 2.048 GB
	Hard disk space	3.6GB
	.NET Framework	3.5
	Processor type	Itanium processor or faster [1.0 GHz or faster]
	Windows Installer	4.5
SQL Server 2012	O\S	Min>> Windows Server 2003 SP2
	Memory (RAM) ¹	1.0 GB
	Hard disk space	6.0 GB
	.NET Framework	NET 3.5 SP1 or 4.0
	Processor type	AMD Opteron, AMD Athlon 64, Intel Xeon with Intel EM64T support
	Windows Installer	4.5
	Windows O\S	Windows 2008 SP2 Minimum

Feature	Disk space requirement
Database Engine and data files, Replication, and Full-	711 MB

Text Search

Analysis Services and
data files

345 MB

Reporting Services
and Report Manager

304 MB

Integration Services

591 MB

Client Components

(Other than Books

Online and

1823 MB

Integration Services

tools)

SQL Server Books

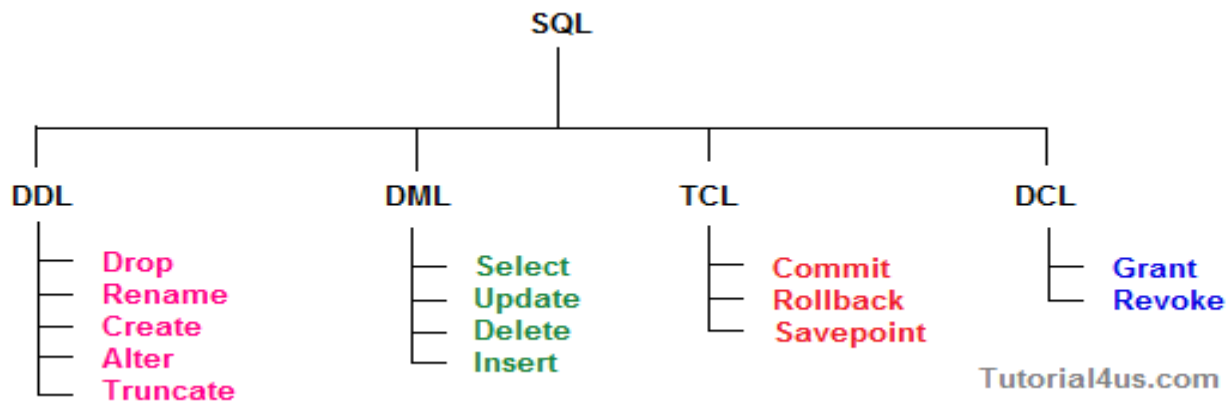
157 MB

Online

DIFFERENCE B/W DELETE AND TRUNCATE

Types of queries:

1. DML [Data Manipulation Language] - Insert, update and Delete
2. DDL [Data definition Language] - Create, alter and Drop
3. DCL [Data Control Language]: Grant, Revoke
4. TCL [Transaction Control Language]: select



Only for remember

Use can remember these all command like below:

- DDL Commands "dr. cat" d-drop, r-rename, c-create, a-alter, t-truncate.
- DML commands "sudi". S-select, u-update, d-delete, i-insert.

What are the difference between DDL, DML and DCL commands?

DDL

Data Definition Language (DDL) statements are used to define the database structure or schema. Some examples:

- **CREATE** - to create objects in the database
- **ALTER** - alters the structure of the database
- **DROP** - delete objects from the database
- **TRUNCATE** - remove all records from a table, including all spaces allocated for the records are removed
- **COMMENT** - add comments to the data dictionary
- **RENAME** - rename an object

DML

Data Manipulation Language (DML) statements are used for managing data within schema objects.

Some examples:

- **SELECT** - retrieve data from the a database
- **INSERT** - insert data into a table
- **UPDATE** - updates existing data within a table
- **DELETE** - deletes all records from a table, the space for the records remain
- **MERGE** - UPSERT operation (insert or update)
- **CALL** - call a PL/SQL or Java subprogram
- **EXPLAIN PLAN** - explain access path to data
- **LOCK TABLE** - control concurrency

DCL

Data Control Language (DCL) statements. Some examples:

- **GRANT** - gives user's access privileges to database
- **REVOKE** - withdraw access privileges given with the GRANT command

TCL

Transaction Control (TCL) statements are used to manage the changes made by DML statements. It allows statements to be grouped together into logical transactions.

- **COMMIT** - save work done
- **SAVEPOINT** - identify a point in a transaction to which you can later roll back
- **ROLLBACK** - restore database to original since the last COMMIT
- **SET TRANSACTION** - Change transaction options like isolation level and what rollback segment to use

DELETE	TRUNCATE
Delete row by row values	Truncate entire data from the table at one shot
Delete can have where condition	Truncate we cannot apply where condition
Roll back is possible	Roll back is not possible
Slow operation	Very fast operation
DML Operation	DDL operation
if we delete only data gets deleted	If we truncate only data gets deleted but structure remains same
Delete Operation will not reset seed value of identify column	Truncate resets the seed value

INSTALLATION

SQL Server Installation:

Servers and Tools:

- > **Tools:** Help only to connect to Sql servers
- > **Servers:** Can use servers for creating a database or operations.

Ex:

- > In data center servers install servers and tools.
- > In developer or local systems only we install tools.

SQL Server minimum requirements:

.Net Framework:

It helps to display the pop-up's

Note: SQL Server version increase then my .net, windows installer, o\s, memory and hard Disk are different.

Diff between X86 and X 64 bit:

1. In x86 only 32buses transferred each time. Where as in 64 bit 64 buses transferred each time.
2. In X86 have limitations when trying to consume the hard ware resources like memory.

Ex: In windows server O\S[X-86] Max memory by default can utilize only 2 GB for the SQL Server. If we want to use more than 2 GB then need to enable switches at O\S and Sql level.

In X64 bit no need to enable any switches and no limitation while consuming the resources.

3. Performance is very good in X64 bit compare to X32 bit O\S.

Installation of SQL Server 2005 Screen:

1. End user License agreement [EULA]

Note: How to what edition, version of Sql server?

Method1: Just run the setup and can get without installing

Method2: Go to image folder in softwares and can get the same info.

Path: SQL Server > Setup > Image

2. Installation pre requisites

Native Client: Internal Communication purpose

> Setup support rules [.NET Framework, windows installer]

Windows PowerShell: It is connectivity purpose to SQL Server

Windows Installer: This is one of the operating system where it helps to deployed the SQL Server services into O\S

Firewall: It will give end points to communication purpose and validate your connection

Note: Only 1 time installation is enough for many Sql server installations in the same server.

3. System configuration checks [Total 14]

Note: Ensure all checks are successful in real time.

Installing IIS performed by O\S team [Windows team] ---web application

4. Registration information

5. Component selection

3 Types:

1. OLTP: Database engine

2. OLAP: SSIS [SQL INTEGRATION SERVICE], SSRS [SQL REPORTING SERVICES], SSAS [SQL ANALYSIS SERVICE]

3. Common components: Book online [BOL], Tools, notification services...

Note:

When you run setup file for the folder SERVERS...You have option to install tools as well as...No need to install again tools for multiple times in single server.

> Please enable all advanced features in real time.

6. Instance Name: Just a name which is used to connect to SQL server

2 types of SQL Server instance names:

1. Default: Takes computer name or host name as an instance name.
2. Named Instance: Can specify the customized name as per the requirement.

Note:

1. If SQL Server edition is enterprise: total no. of instances can install 50 per server

1: default + 49 named instances

2. If SQL Server edition is other than EE: total no. of instances can install 16 per server

1: default + 15 named instances

Note: Even can install all 16 or 50 named instances as well

Limitations:

1. Max instance name char allow: 16 only
2. No special char allowed for the instance name
3. No instance can start with numbers

Note: Please collect the instance name from requestor.

7. Authentication mode: Authenticate while connecting to SQL Server user accounts.

> Windows authentication:

> Mixed mode:

Note: In real time always select "Mixed mode" and required to provide the password called "SA account".

8. Service accounts: On which service account the SQL Server services need to run.

Note: In real time always use domain level services and install SQL Server.

9. Collation settings: Nothing but language preference

Default Collation for SQL Server: **SQL_Latin1_General_CP1_CI_AS, SQL**

CP: Code page

CI: Case in sensitivity

AS: Accent sensitivity

Note: Ensure you confirmed the collation name with requestor or application team... Collation need to specify on basis of your application

10. Errors and Warning

11. Install

SQL Server configuration manager: This tool shows only SQL Server related services.

Note: Every version of SQL Server have configuration manager tool. In higher version configuration manager I can see the services of lower version of Sql as well.

Operations can perform?

Start

Stop

Pause

Restart of the Sql services.

> Report server configuration: This is one of the tool help to configure reporting services.

Note: Reporting [SSRS] component should install before configuration.

> SQL Server root directory: c:\program files\Microsoft Sql server\

Folder inside the instance:

Folder 80: Created when you install Sql server 2005 version and used for backward Compatibility

Folder 90: Contains

COM [Component management] - SQL Server exe files and DLL files related to SQL Server components

DTS: Used this folder only for SSIS

EULA: Contains a text files of license information

NOTIFICATION SERVICES: Contains command prompt for notification services.

SDK: [Software development kit] Contains developer related tools

SETUP BOOTSTRAP: Key folder contains [1033, bin, log, resource, arpwrafer, setup, dll files]

Note: Installation success or failure can get from SETUP BOOTSTRAP\LOG\summery.txt
Always contains recent installation information.

SHARED: Contains common component related information

TOOLS: Tools related exe files for tells and sample databases.

Folder inside Database engine:

Binn: Critical folder for the instance and contains instance related .DLL and .EXE files.

Repldata: Stores replicated information

Data: Contains database files

By default 2 types of database

1. System databases: Create at the time of Sql server installation

2. User defined databases: Should create by USER.

FTDATA: Full text related information gets stored in FT Data folder

JOBS: Stores only Sql server jobs related information

INSTALL: Contains some scripts

Logs: Very important folder to store event tracking inside the Sql server and used for DBA trouble shooting area.

Backup: Store backup related files.

Build Numbers:

SQL 2000: 80: 8.0

SQL 2005: 90: 9.0

SQL 2008: 100: 10.0

SQL 2008 R2: 100: 10.50

SQL 2012: 110: 11.0

SQL 2014: 120: 12.0

Connectivity Procedure:

To connect to default instance:

[Hostname or local]

To connect to named instance:

[HOSTNAME\instance name]

Shortcuts:

SQL 2000: ISQLW

2005: SQLWB [SQL WORK BENCH]

2008\8R2\2012\14: SSMS [SQL Server management studio]

Resource database is introduced in SQL Server 2005 newly for security purpose.

2 types of service classification:

1. Instance aware services:

- > SQL Server main service
- > SQL Server agent service
- > SQL full-text service
- > SSAS Service
- > SSRA service

2. Instance unaware services:

Common services:

Integration service

Browser

Notification services

RTM Build Numbers:

Note: whenever Microsoft releases new product or version into market called RTM

2000: 8.0.194

2005: 9.0.1399

2008: 10.0.1600

2008 R2: 10.50.1600

2012: 11.0.2100

Per instance how many no of database can create?

32767

5 [master+model+msdb+tempdb+resource] - system database

32762- User defined databases

Database ID's:

Master -1

Model-3

MsdB-4

Tempdb- 2

Resource-32767

Note: When you create any user database id starts from [5]. Database engine always uses or communicate with DBID

If I delete any between database then the same ID assign to a new database in sequence manner.

Note: Diff between SQL Server logs and error logs:

SQL Server logs track or contains "instance level event" but whereas error log have only "SQL Server agent related event tracking"

How to find how many instances are installed in server?

NET START | FINDSTR -I SQL

Note: SQL Server 2008 \R2 configuration manager show all SQL Server services for same and lower version of SQL as well

But where as in Sql server 2005 configuration manager does not show 2008 \R2 or higher version services.

Uninstallation process of SQL Server: [Decommission of the SQL Server]

Appwiz.cpl or Add\remove programs-----Shortcut of Control panel

Methods for Uninstallations:

1. Add\remove programs- control panel
2. Arpwraper option to uninstall Sql server

C:\Program Files\Microsoft SQL Server\90\Setup Bootstrap\ "Arpwraper"

Complete Uninstallation of SQL Server:

1. Pre installation steps:

- > Collect instance name to uninstall
- > Collect server name
- > Check whether what components to uninstall
- > Check how many instances, based on uninstall tools as well?

2. Installation steps:

Uninstall SQL Server instance

3. Post installation steps:

- > Verify whether SQL Server services are removed from configuration manager for the specific instance
- > Remove the folders which are all related to your instance which need to uninstall.
- > Remove from register editor [regedit] for the particular instance if not removed at the time of uninstallation.
- > Go to run>regedit> HKEY_LOCAL_MACHINE> SOFTWARE>MICROSOFT>MICROSOFT SQL SERVER>

To open registry in o\s level: regedit

Note: ARPWRAPER is only available in SQL Server 2005 but not in versions of 2008, 2008 R2, 2012.

Note: Before and after uninstallation of SQL please send email notification to requestor.

SQL Server installation real time process:**> Pre-Installation Requirements phase:**

1. Verify Hardware and Software configurations
2. Default\named instance
3. Collation settings
4. Service account
5. Authentication mode.
6. Check x64 or x86 bit of SQL installation required?
8. Collect SQL software\package into server or download
9. Components selection.
10. SQL version and edition

> Installation Phase:

Perform installation\configuration of SQL Server

> Post-installation phase:

1. Verify all the components are installed success... Or not?

From "Summery.txt" file

C:\Program Files\Microsoft SQL Server\90\Setup Bootstrap\LOG\ Summery.txt

2. Check connectivity to SQL instance.
3. Check all services are running or not.
- 4 Check the registry whether instance is registered or not.

ISSUES AT THE TIME OF SQL SERVER INSTALLATION?

- > Lack of Hardware and software
- > O\S in Compatibility
- > Not install right .net frame work and windows installer
- > Native client corrupted
- > Wrong service account and password
- > Permissions at O\S level to install SQL server
- > Disk space issue
- > Software\media corrupted
- > Instance name started with special char
- > Version and Edition Compatibility
- >.MSI [Microsoft installer] is missing or corrupted

Note: .MSI files locate in C:\WINDOWS\INSTALLER

SILENT INSTALLATION OR UNATTENDED INSTALLATION:

Steps:

1. From command prompt required to enter to path where "setup.exe" file is placed.

Drive:

Cd (copy path)

\QB: Will display popup's

\QN: Without displaying popup's installation will complete.

```
Start/wait setup.exe/qb INSTANCENAME= dummyinst ADDLOCAL=SQL_Engine
SAPWD=admin143$$ SQLACCOUNT=WIN-QT4CQF1BPAM\localsystem
SQLPASSWORD=admin143$$ AGTACCOUNT=WIN-QT4CQF1BPAM\administrator
AGTPASSWORD=admin143$$ SQLBROWSERACCOUNT=WIN-QT4CQF1BPAM\administrator
SQLBROWSERPASSWORD=admin143$$ SECURITYMODE=SQL collation="
```

SQL Server 2008\2008 R2 Installation steps:

1. Install .net framework 3.5
2. EULA
3. Component selection [Servers & tools]
4. Service account
5. Collation
6. Authentication mode
7. Add current user---New in SQL 2008
8. Directory selection- New in SQL 2008
9. File stream [New in SQL Server 2008 onwards]

File stream:

> FILESTREAM was introduced in SQL Server 2008 for the storage and management of unstructured data. The FILESTREAM feature allows storing BLOB data (example: word documents, image files, music and videos etc) in the NT file system and ensures transactional consistency between the unstructured data stored in the NT file system and the structured data stored in the table.

SQL Server 2008\2008 R2 install diff compare to 2005:

- > Servers and tools are integrated
- > Built in administrator group removed
- > Add current user
- > File stream
- > Share point integrated
- > Splitting the database files into multiple disks.
- > Net 3.5 and windows installer 4.5

Note: Folder creations in SQL 2005 create as MSSQL.number. It is very difficult to understand folder belongs to which instance.

But whereas from SQL Server 2008 onwards the folder creation (MSSQL 10.0 OR 50 _INSTANCE aqwaqNAME) which help to understand easy about the instance details.

Production server standards for SQL Server:

> Never keep all system databases, user databases files, backup files, binary files, instance related files into single disk.

Reason: Data loss chance are 100 % if the disk is corrupted.

Split SQL Server Files as per below:

Installation related file: C:\Program Files\Microsoft SQL Server

System database [Master, model, msdb] - Keep in SAN disk

User database DATA files: - Keep in a separate SAN disk

User database LOG files: - Keep in a separate SAN disk

Tempdb database DATA files: - Keep in a separate SAN disk

Tempdb database Log files: - Keep in a separate SAN disk

Backup files: Keep in separate SAN disk

Note: EMC2 SAN [Storage area networks] disks are act like a network disks where we can Connect to another system like external hard disk. Data recovery is very easy.

Note: After Sql server installation perform restart of the server.

Process: Inform to windows team to restart the server. Once server up, DBA team check SQL Server.

Points on versions:

Point 1:

Note: In same server if we install SQL Server 2005 and Sql 2008 or higher version?

> SQL 2005 Configuration manager shows only the services of 2005. Whereas SQL Server 2008 or higher version shows the services in configuration of "SQL Server 2008 or higher Version+ lower version ". Vice versa is not possible.

> Each version of Sql server have different configuration manager's tools.

Point 2:

> SQL 2005 instance can connect in SQL server 2008\higher version SSMS but whereas higher version instance cannot be able to connect in lower version SSMS.

Point 3:

> For Sql server 2008 and 2008 R2 summery .txt file is common and can access from 100 folder.

C:\Program Files\Microsoft SQL Server\100\Setup Bootstrap\Log

SERVICE PACKS

SQL Server Service Packs:

> Microsoft designed service packs or hot fix or cumulative updates to fix the error or bugs or code related.

Service Packs normally have three properties:

- > Provide security fixes
- > Correct software errors
- > Enhance performance

	RTM (no SP)	SP1	SP2	SP3	SP4
↓ SQL Server 2014 codename Hekaton SQL14	12.0.2000.8	12.0.4050.0 or 12.1.4050.0 downloads temporarily disabled			
↓ SQL Server 2012 codename Denali	11.0.2100.60	11.0.3000.0 or 11.1.3000.0	11.0.5058.0 or 11.2.5058.0		
↓ SQL Server 2008 R2 codename Kilimanjaro	10.50.1600.1	10.50.2500.0 or 10.51.2500.0	10.50.4000.0 or 10.52.4000.0	10.50.6000.34 or 10.53.6000.34	
↓ SQL Server 2008 codename Katmai	10.0.1600.22	10.0.2531.0 or 10.1.2531.0	10.0.4000.0 or 10.2.4000.0	10.0.5500.0 or 10.3.5500.0	10.0.6000.29 or 10.4.6000.29

↓ SQL Server 2005 codename Yukon	9.0.1399.06	9.0.2047	9.0.3042	9.0.4035	9.0.5000
↓ SQL Server 2000 codename Shiloh	8.0.194	8.0.384	8.0.532	8.0.760	8.0.2039
↓ SQL Server 7.0 codename Sphinx	7.0.623	7.0.699	7.0.842	7.0.961	7.0.1063

> **Hotfix:** A specific issue or bug fixed and released as a hotfix.

Duration: Every month 2nd week MS release hot fixes for multiple issues

Note: Hot fixes are incremental.

> **Cumulative update:** Multiple hot fix bugs are included and released as CU

Duration: Every 2 months generally release CU

> **Service packs:** Combination of hot fix+ CU=SP

Duration: Every 6 months is the duration.

Note: SP or CU or hot fixes designed for version level but not edition level.

Service pack or hot fix or cu installation process:

1. Pre-installation steps:

> Check hardware and software

> Check the current version

Select @@version...

Select serverproperty ('product version'),

Serverproperty ('product level')

> Check the current bit of Sql server (X 86 OR X64)

> Admin level permissions are required

> Download the right service pack and copy to server

Note: Take complete system and user database full backups.

Note: Take a copy of the BINN folder and resource database [MDF and LDF] file.

Note: Take approval or confirmation from application to apply sp or cu or hot fix.

Reason: when you apply downtime is required.

SQL Server services automatically goes offline state and once done services comes online.

Background process when you apply SP:

When you apply service pack the BINN and resource database files overwrite or update with Service Pack files.

> Run moksha tool to find whether any .msi or .msp files?

2. Installation steps:

Run patch setup.exe file > Select the instance name.

Note: SP always should apply for both servers and tools

3. Post service pack steps:

> Verify the summery.txt file to confirm whether sp or cu or hotfix is applied for all components or not

> Verify build number is changed or not

Select serverproperty ('product level'),

Serverproperty ('product version')

Or

xp_msver

> Verify registry whether build number is changed or path

REGEDIT> HKEY_LOCAL_MACHINE>SOFTWARE>MICROSOFT> MSSQL> INSTANCE> CURRENT
VERSION

> CHECK all databases are running?

> Inform application team to test the connectivity and additional checks

If Service pack fails or application team request rollback to RTM? Then

SQL 2005 process:

1. Uninstall Sql server
2. Reinstall Sql server with all components, same instance
3. Restore all databases including system and user databases

SQL Server 2008 \8R2\12 Process:

Directly can be able to uninstall the service\cu\hotfixes from CONTROL panel.

How to bypass restart computer policy in Sql server and applying any sp?

Go to REGEDIT> HKEY_LOCAL_MACHINE>SYSTEM>CONTROL>CURRENT CONTROL> SESSION
MANAGE> "PENDING FILE RENAME OPERATIONS">>>

Clear this file data and re run the checks.

Roll back process for 2005 or 2008 onwards?

SQL 2005:

1. Uninstall Sql server
2. Reinstall Sql server with same name
3. Restore all system and user database
4. Inform apps team to check the data.

SQL Server 2008 \8R2\12:

Method: 1 Try to uninstall Sql server service pack from control panel

If not works then go for 2005 process.

Points:

> Service pack are cumulative

> Hot fixes are incremental.

> If service pack applied on RTM then if you uninstall then build no goes to RTM only. Whereas if we apply sp2 on RTM+SP1 then build no goes to SP1.

Service pack fails reasons:

1. Missing or corruption of .MSI or .MSP files.
2. Incompatibility issue
3. Permission issue.
4. Service pack media corrupted.

1. Missing or corruption of .MSI OR .MSP files:**What is .MSI and .MSP files:**

A MSP (Microsoft Software Patch) is basically a set of transforms (modifications) against a baseline MSI (Microsoft Software Installer) file.

Findings: From summery.txt file under hotfix folder

Solution:

> Extract from Sql server software.

Process:

Extraction process: Sql service pack 3 file for example is of the format 'Sql server sp3-90000-setup.exe' in E:\sqlbits directory, then navigate to this directory in command prompt and execute the command as below:

Service pack directory>service pack file name /x

X: means extract

A dialog box will open requesting the location to extract the bits. Provide the location and press OK. SP3 bits will be extracted in the specified location.

Or

> Download the missing .msi or .msp file and place into C:\WINDOWS\INSTALLER

Or

> Copy from another same version of Sql server

How take database offline\online?

ALTER DATABASE [DBNAME] SET OFFLINE

ALTER DATABASE [DBNAME] SET ONLINE

How to find database status?

Select * from sys.databases: show all databases

Select* from sys.sysdatabases: Only online databases

FILES

Files:

In SQL Server every database should contains minimum 2 files

Without having files database cannot run.

The main purpose of this file is to store the data

Types of files:

.MDF [Master data file, main data file, primary data file]

.NDF [Next data file, new data file, secondary data file]

.LDF [Log data file]

Purpose:

1. .MDF:

> This file is the startup of the database in Sql server/database...i.e. whenever you start Sql server database the first file comes online "MDF"

q

- > MDF contains other file information [.LDF and .NDF]
- > Per database only 1 MDF file and cannot possible to delete.

2. .NDF:

- > When primary data file is full\depending on the requirement we can add multiple secondary files per each database.
- > Multiple secondary data files can add

3. .LDF:

- > The main purpose this file to recover the data whenever any databases crashes.
- > At least 1 log file should contain per database
- > Every transaction should write first into LDF file and this concept is called as "WAL [WRITE A HEAD LOGGING]".
- > Log files extension always given as .LDF

Note: per database we can be able to create 32767 file (.MDF, .NDF, .LDF)

Data file size: [.MDF or .NDF] –16 TB

Log file: .LDF –2TB

Note:

1. per database can create total max number of files can create 32767

[Default: 1 MDF +1 LDF].

32765 [Multiple NDF+ MULTIPLE LDF'S]

2. The default MDF\NDF file size: 2 MB and LDF: 1 MB

[Inherit all the properties from MODEL database]

1. How to add a file by using query analyzer?

ALTER DATABASE [DBNAME]

ADD FILE

(NAME='Logical name', Filename='path\physicalfilename.NDF OR LDF')

To filegroup [FGNAME]

Note:

1. Per database we can have multiple .NDF and .LDF files
2. Every database should contain Logical name and as well physical file name with specific extension.
3. Every file should contain one file id to communicate database by using database ID.
4. MDF and NDF: Data stores permanently
LDF: Data store for initial transaction processing
5. By default initial size of MDF or NDF: 2 MB and LDF: 1 MB
6. Auto growth option is always set depending on the file type and gives more value of file growth which increases the performance database to faster the transaction.

Note:

In SQL Server we can add files [NDF (or) LDF] to the database is online... i.e. without downtime can be able to add the files.

How about System Databases:

1. **MASTER:** We cannot be able to add any data (.NDF) or log (.LDF) files to master database.
2. **MODEL:** We cannot able to add any data (.NDF) or log (.LDF) files to model database.
3. **MSDB:** We can able to add files (.NDF or .LDF) to MSDB database.
4. **TEMPDB:** We can able to add files (.NDF or .LDF) to Tempdb database.
5. **RESOURCE DB:** We cannot able to add any files due to resource database is read_only mode and hidden.

FILE GROUPS

File Group:

Set of saved files.

Collection of files [NDF or MDF]

Advantages:

- > Ease of Administration
- > Filegroup Backups
- > Performance Benefit

Note:

Per instance > 32767 DB

Per DB> 32767 file groups

q

Per File group> 32767 Files.

Note:

Tables always create file group level but not specific files. If a file group contains multiple files then the table structure creates inside of all files.

Syntax for file group creation:

```
ALTER DATABASE [DBNAME] ADD FILEGROUP [filegroupname]
```

Create table to specific file group:

Ex:

```
CREATE TABLE [dbo]. [Stab] (  
[Sno] [nchar](10) NULL,  
[Same] [nchar](10) NULL  
) ON [FILEGROUP NAME]
```

Note: We cannot be able to read data from physical NDF, MDF or LDF.

Reason: Inside file data in encrypted formate and only Sql server engine can be able to understand.

Note:

In real time **MDF or NDF** file ALWAYS the file restriction should be "UNRESTRICTED GROWTH" and File growth value should" MB"

Ldf file ALWAYS the file restriction should be "RESTRICTED GROWTH" and File growth value should" MB"

Note:

1. Primary file group never able to set read-only and which is default set

q

2. If you set any file group as read-only then cannot be able to any DML\Alter operations in the table.

3. For a specific file group can possible to set both READ_ONLY\DEFAULT. But which always takes "READ_ONLY"

PAGES

Pages:

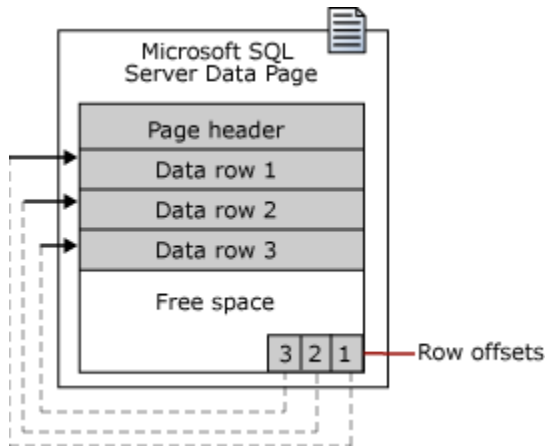
1. The fundamental unit of the data storage is called as page.

2. Default a page size is 8 KB.

- **Page is the fundamental unit of data storage** in SQL Server (data blocks in Oracle).
- **Eight (8) physically contiguous pages => One (1) EXTENT.**
- Disk space allocated to a data file (.mdf or .ndf) in a database is **logically** divided into pages numbered from 0 to n.
- **Disk I/O** operations are performed at the **page level**. (SQL Server reads/writes **WHOLE** pages).
- **Page size: 8kb.** ==> 128 pages = 1Mb.

Page architecture:

Every page contains 3 blogs



> **Page header:** type of the page, page number, object details, how much free space inside the page

Page header size: 96-Bytes

> **Data rows:** Where actual data stores inside the page and size is 8060-Bytes

> **Row offset:** This is the index of the page when data rows increases the row offset also increase. Row offset index of data rows.

Row offset size: 36-Bytes

Note: Per MB of space how many pages can allocate?

128 pages can allocate

1 GB: 131072 Pages.

Note: Always pages store inside the extents> in a file.

Types of pages:

Total 8 types of pages in SQL server

> **Data:** All data type data stores in this page except text, ntext, and image.

> **Index:** Only stores index related entries

> **Text\image:** text, ntext, image.

Note: All below 5 types of pages are designed for Sql server engine usage. Called maintenance pages.

> **GAM\SGAM:** Store information about allocated extents.

> **IAM [Index allocation map]:** Stores information about index pages in extents.

- > **PFS [Page free space]**: How much free space in a page.
- > **BCM [Bulk change mapping]**: Store entries about log backup.
- > **DCM [Differential change mapping]**: Stores information about full database backup.

GAM	SGAM
It records what extents are allocated	SGAM records which extents are current used as mixed extents and have at least one unused space
GAM has 1 bit for extent for each extent in the interval, it covers. If the bit is 1, the extent is used	SGAM if bit is 1, then it is extent is used as mixed extent and has free page
It talks about general extents	It talks about exclusively mixed extents

For a database:

Page 0 is header

Page 1 is PFS

Page 2 is GAM

Page 3 is SGAM

Page 6 is DCM

Page 7 is BCM

Page 9 is Boot Page

Objects in a database

$2^{31}-1$ Objects (Maximum)

Note: If we want to open log file, we need to go third party tool or Log Explorer, Apex SQL tool, SQL log resource

Boot page: DBCC – Database Consistency Check

> The last run DBCC Checkdb is written in the boot page. It contains most critical information about the database. Every database boot page is the 9th page

q

> When instance is restarted by looking at this boot page is a method to write, information into the event log.

Error: 5047→failed for file group (with in the filegroup, if an table is there)

> Moving a file from one filegroup to other file group, is not page the alternate way is to first move the data in the FG\file to the other filegroup through scripting/export import and once data moved successfully. Then delete the file from file groups and create the file in File group2.

EXTENTS

Extents:

> Minimum unit of data storage called an "extent"

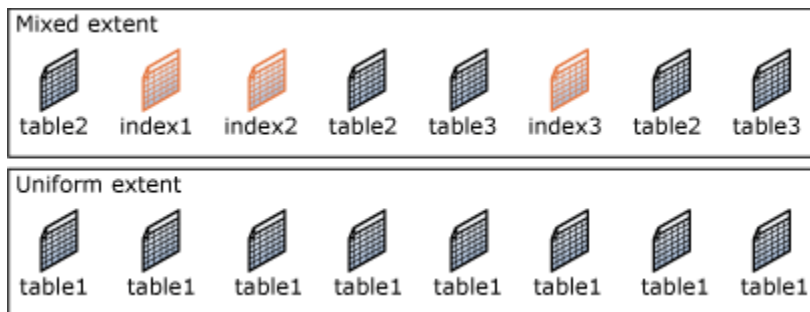
> Extent is logical storage.

> 1extent is collection of 8 pages

> Extent size is: 64 KB [8 * 8 KB]

> 16 Extents = 1 Mb

Types of extents:



1. Uniform extent: Same type of 8 pages store then call as uniform extent.

2. Mixed extent: Combination of 8 different types of pages called mixed extent.

Note: By default SQLServer engine allocates the pages (8) into mixed extent. After filled, if engine identifies all 8 pages are belongs to same type then all pages carry to "uniform extent".

File architecture and Page allocation process:

Data file architecture:

1. Pages permanently stores into data files [NDF or MDF]
2. Page allocation: Page allocation always stores in a sequence manner starting with 0.
3. Every file contains unique file ID.

MDF: Primary data file: 01

01: 0000

01:0001...ETC

NDF: Secondary data file: 03.....continue...

03: 0000

03:0001...ETC

4. Every Page stores in a specific formate: **FILE ID: PAGE ID**
5. Always page id starts with 0000 sequence number.
6. Page sequence no also stores in page header of each file.

1. How to find drive space from SQL Server:

- XP_fixeddrives

2. To know compatibility level of all databases

- Select * from sys.databases [DB ID, collation_name, compitabilitylevel, db created date, state]

3 .To know (dbname, size, owner, date created, status, compitabilitylevel) Particular databases

- Select * from sys.sysdatabases

4. To know space availability in a log file total and used space...

- Dbcc sqlperf (logspace)

5. To know space availability in a data file

- sp_spaceused 'database name'

6. To know the details of a particular database (dbname, filename, filegroup, size)

- sp_helpfile

7. To know the details of database (name, filename, path)

- select * from sys.sysaltfiles

8. How to take database online and offline?

- Alter database dbname set offline
- Alter database dbname set Online

Note: System databases [master, model, msdb, Tempdb] not possible to take offline

9. Rename of database:

- Sp_renamedb 'olddbname', 'newdbname'

LOG FILE ARCHITECTURE

- > The transaction log is used to generate the data integrity of the database and for data recovery.
- > The transaction log file contains a storing of log records. Physically the sequence of log records is stored efficiently in the set of physical files that implement the transaction log.

Transaction: Set of statements

- > Transaction is an all/nothing way of execution of set of statements.
- > SQL Server supports two types of transaction.

Explicit Transaction:

- > A transaction is started with begin Tran and finished with end/commit transaction.

Syntax: Begin Tran

Statement1

Statement2

Statement3

Commit/ End Tran

Implicit Transaction:

- > SQL Server internally adds begin and end transaction for the individual statements.

Syntax: Statement1

Statement2

- > MDF Contains pages.
- > Log file Contains Log records.
- > Log file applicable for SQL Server

Log Record:

Log Records are entries that are made into the Transaction Log files.

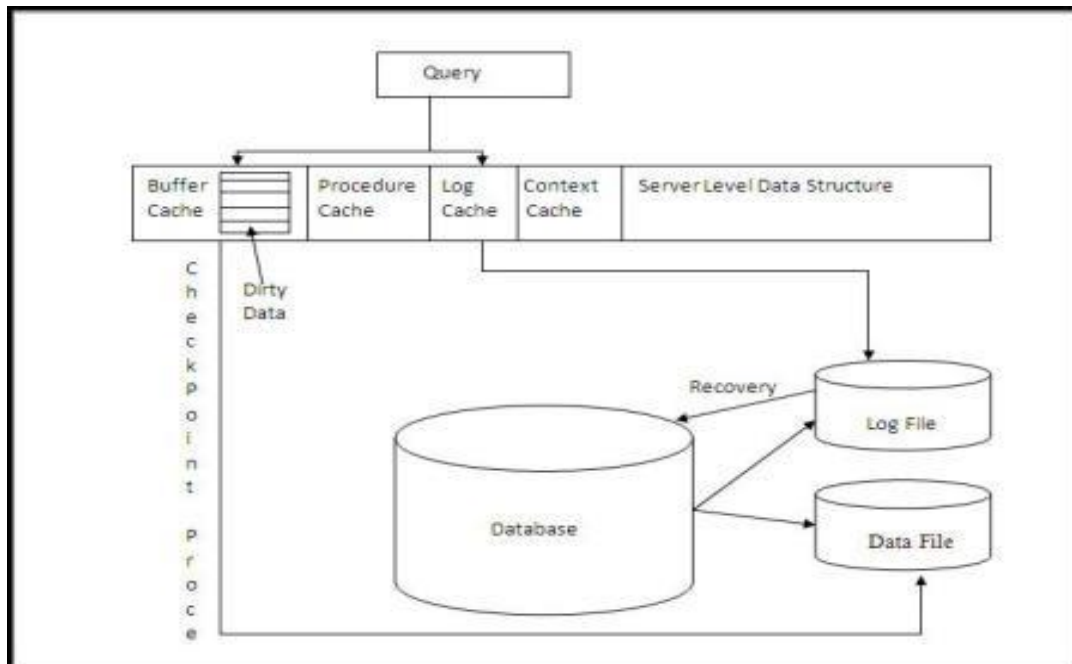
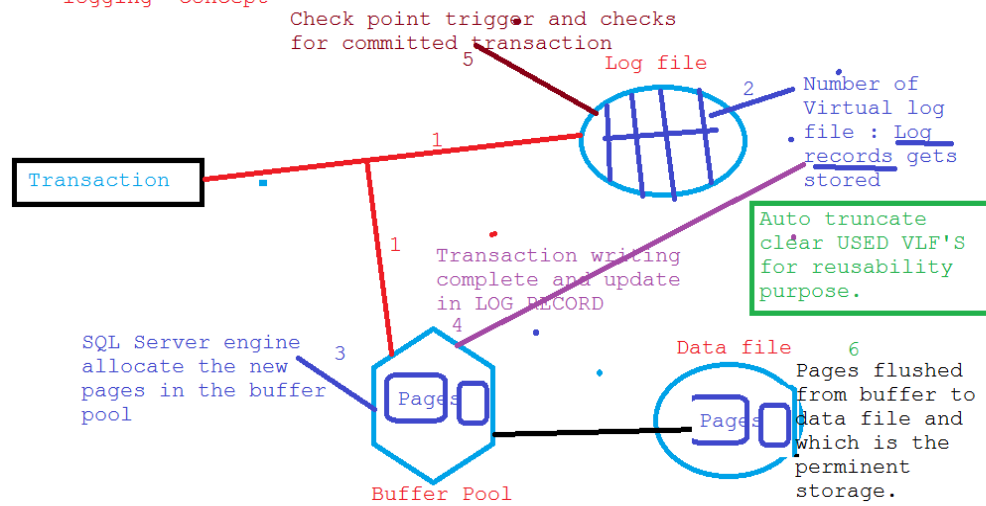
Log Records contain:

- 1) LSN Number
- 2) Transaction ID
- 3) Timestamp
- 4) Transaction statement (insert into student 1 to 10 values)
- 5) Statement (Query)
- 6) Data page details/Object details
- 7) Previous LSN, Log record type, Abort (Transaction committed (or) not redo LSN and Undo LSN)
- 8) Old image/ new image (Which ever applicable)
- 9) Committed/Uncommitted, Completion record

Log File Architecture in SQL Server

> Whenever any query is processed, the data will be passed to Data file. Below is the process how a query is processed in SQL Server/ Importance of Log File Architecture:

1:- Any transaction write into transaction log called "WAL [write ahead logging]" concept



- Database has Data file and Log file.
- The query statement is passed to the buffer cache and log cache.
- The data in the buffer cache is called as Dirty Data or Dirty Blocks.
- Buffer cache contains the dirty data (the updated data corresponding to the query given in the application).
- The process of writing the data from buffer cache to the data files of the database in the form of chunks is called as Checkpoint Process.
- Each chunks contains 512 KB.

- The query is written into the log file from the log cache.
- If any type of failure occurs while writing data to the data file, then the query in the log file is executed at the last commit transaction processed (refer commit process down) and the remaining data is written to the database whenever we start the server.
- This process of writing data to the database after a failure from the log file is called as Recovery.
- Procedure Cache contains the execution plan.
- Context Cache contains the data of the stored procedure.
- Server Level Data Structure contains the Server level information.

Commit Process:

As soon as commit statement is written to the log file it throws a token to the user that commit is completed successfully (Ex. 1 row affected), this process is called as Commit Process.

Logs file architecture points:

Log file purpose is to recover the data when there is a disaster.

WAL [Write ahead logging]: Writing every transaction into transaction log called "WAL"

VLF'S [Virtual log file]:

1. New in Sql server 2005 version
2. VLF's have no fixed size; no fixed no of VLF'S and grows dynamically.
3. VLF's creation depends on File AUTO_GROWTH option and set more which cause less VLF's creation and can get more performance.
4. VLF's are reusable
5. Max 50 VLF's then can get very good performance...If increase then performance gets down.
6. The SQL Server Database Engine divides each physical log file internally into a number of virtual log files. Virtual log files have no fixed size, and there is no fixed number of virtual log files for a physical log file. The Database Engine chooses the size of the virtual log files dynamically while it is creating or extending log files. The Database Engine tries to maintain a small number of virtual files. The size of the virtual files after a log file has been extended is the sum of the size of the existing log and the size of the new file increment. The size or number of virtual log files cannot be configured or set by administrators.

q

How to find VLF's:

DBCC LOGINFO

Status: 0[Inactive] or 2 [Used \active]

Size of VLF

File id

File sequence no

How many VLF's create on transaction size?

Virtual Log file creation by default:

<64 MB::4 VLF's create.

>=64 MB and <1 GB:: 8 VLF's create

>1 GB:: 16 VLF's create

Note:

For log files always Microsoft recommends to keep FILE GROWTH INTO "Percentage"...

For Data files always Microsoft recommends to keep FILE GROWTH INTO "MB's"...

How to find whether log file is in use?

select * from sys.databases ::Verify the column "log_reuse_wait_desc" shows "active" then log file is in use ...if not using shows "nothing"

Definitions:

Committed: Transaction written completely

Uncommitted: Transaction is in progress

Dirty Page: Current modifying data inside the pages in buffer pool called "Dirty page"

How to find whether Log file is using or not?**Select * from sys.databases****Column:** LOG_REUSE_WAIT_DESCRIPTION:

ACTIVE MEANS: currently ldf file is using

NOTHING: LDF file is not using or no transactions are running and using WAL concept

Log file full: 9002 is the error number**Data file full:** 1105 is the error number

> How to find check point information?

Select * from:: fn_dblog (null, null) WHERE [Operation] like '%CKPT'

Columns:

Check point operation [LOP_XACT_CKPT, LOP_BEGIN_CKPT, LOP_END_CKPT]

Check point start time

Check point end time

Dirty page count

Transaction LSN number

Transaction id

CHECK POINT**Check Point**

1. A checkpoint writes the current in-memory modified pages (known as dirty pages) and transaction log information from memory to disk
2. Records information about the transaction log.

Check point do:

Flush dirty to Disk

Flush only Data pages to disk

Default, Occurs approximately every 3 seconds

Does not check the memory pressure

Occurs for any DDL statement

Occurs before Backup/Detach command

Can manually /forcefully run command “Checkpoint”

Very Less performance impact

Types of check points:

1. Direct check point: Which is the default checkpoint triggered by SQL Server engine.

Default checkpoint time: 3 seconds

2. Manual check point: Which is always trigger by user

Syntax:

Checkpoint [time interval in seconds]

3. Indirect checkpoint: This check point rise when any Sql server technical activities like

Backup operation

attach\detach operation

SQL Server restart

Cluster failover

Note: How to capture check point and lazy writer information into Sql server logs?

Check point:

DBCC TRACEON (3502,-1)

3502: TRACE TO CAPTURE CHECKPOINT

-1: Runs in entire instance level

How to stop the trace?

DBCC TRACEOFF (3502,-1)

Output store in Sql server logs:

Checkpoint start time

Check point end time

Lazy writer:

- > This is one of the back ground process where works only from BUFFER Cache.
- > Lazy writer works only when system is running with memory pressure or running out of memory.
- > User cannot see or create, manual lazy writer operation in Sql server
- > We cannot able to capture lazy writer information into SQL Server logs.

Lazy Writer Do:

Allocating space for new pages in buffer pool

Checking for memory pressure.

Only works with buffer pool but not in log file.

Note: prime job of lazy writer is it flush the pages from buffer to disk.

Question? How lazy writer knows what pages to flush from buffer to disk?

Ans: lazy writer check each page header and verify whether all transactions in the pages are committed then pages will be flushed to disk... [Page header contains infusion of whether committed\uncommitted]

Explicit and Implicit commit:

1. By default system trigger auto commit called "Explicit commit"
2. If any user started transaction with begin Tran then user should fire manually commit otherwise transaction never complete.

Dirty Page:

In buffer pool what are the current modifying pages called as "Dirty page"

SP_WHO:

Spid

Status

Login name

Db name

Command

Blocking by

Host name

sp_who2 [SQL Server 2005 onwards]

CPU Time

Disk I/O

Spid

Status

Login name

Db name

Command

Blocking by

Host name

CHECKPOINT	LAZY WRITER
Check point runs in only transaction log file	Lazy writer operates from buffer pool
Check point is the logged operation and writes to Tlog file	Lazy writer is the non-logged operation and did not write to Tlog file
Check point can be controlled by user and Sql server engine as well	lazy writer only operate by Sql server engine
Check point is the background process which triggers every 3 sec	Lazy writer does not have any fixed time line and only occurs when there is memory pressure in the buffer pool
We can able to track checkpoint into Sql server logs by enabling the trace 3502	Lazy writer information cannot be able to track into SQL Server logs.
In memory free pages list not taken care and kept free list	In memory free pages list taken care and kept free list
We can fire a query to see check point information by using Select * from :: fn_dblog (null, null) WHERE [Operation] like '%CKPT'	We don't have any query to see lazy writer information
Command : Checkpoint [time in sec]	No command available

SECURITY

Security:

Protecting the SQL Server database or instance.

Real time O\S points:

1. In real time all accounts create in domain level.
2. Once they add account into domain level, automatically accounts replicate to the number of servers added into domain.
3. In O\S level, user name and password stores in NTLM OR KERBOROS KEY.

Now a days all servers uses KERBOROS keys authentication.

4. When any connection from local office laptop to server????

Check 1: Firewall checks the connection

Check 2: Verify the user name and password entered by you by the O\S from Kerberos keys.

Authentication: To access any application\server\database...etc. provide valid authentication details like user name and password

Login for Authentication

Authorization: Authorize the connection by verifying the user name and password.

User for Authentication

To login to O\S: O\S do authorize

To login to SQL Server: SQL Server do authorize

2 types of authentication in Sql server

1. Windows authentication

2. SQL Server authentication

3. MIXED MODE [WINDOWS +SQL SERVER]

1. Windows authentication:

> This is the trusted authentication

> If you want to connect Sql server first the account should be member of O\S level then only can be able to use same account to login into Sql server.

> No password transfer over the network

> When you connect to Sql server with windows authentication validation (user name and password) checks in Kerberos key. Once valid pass the connection to Sql server as well.

> Always windows authentication login formate is: DOMAIN\LOGIN NAME

IN SQL Server 2005:

1. If the account is a member of o\s admin group, by default you can get access\connect to Sql server because of "**BUILTIN\Administrators**" Group in SQL Server.
2. If this account is deleted then no admin group accounts can login until adding them manually.

Note: - Windows authentication is always gives more security.

2. SQL Server Authentication:

> No need to create in o\s level

> User name and password provided while login into SQL Server instance with Sql authentication

> We can only able to connect by installing tools in local machine from their connect to server by providing user name and password.

> Passwords travel over a network for authentication this makes Sql authentication is less secure than windows authentication.

Note: Cases of going to use SQL authentication: If any third party source databases like Ms-access, Oracle, db2...etc

Note: Windows authentication is always more secure compare to SQL Server authentication due to Sql authentication .Password are travel over a network which cause less secure than windows authentication.

If your application not support windows authentication login

Error: 18452**Message: UN trusted authentication connection**

Error reason: When my instance is configured with windows authentication, users are trying to connect with SQL Server authentication.

Note: Real time when you perform any Sql server installation, please configure with mixed mode always to avoid login failure.

ROLE OR PERMISSIONS

q

In SQL Server 3 levels of roles or can restrict users at 3 levels

Role: A role is nothing but permissions

How many levels we can secure \restrict users in Sql server?

> Instance level Role

> Database level Roles

> Object level Roles

INSTANCE LEVEL:

B D² P S⁴

Bulkadmin

Dbcreator

Diskadmin

Processadmin

Public

Securityadmin

Serveradmin

Setupadmin

Sysadmin

How to find Sever Level Roles?

Exec sp_helpsrvrole

1. Bulk admin:

User can able to perform bulk insert operation when any data loading activity from application side.

Can load data from any .CSV files

Ex: <http://blog.sqlauthority.com/2008/02/06/sql-server-import-csv-file-into-sql-server-using-bulk-insert-load-comma-delimited-file-into-sql-server/>

2. DB Creator:

Login can perform any DDL operations

Create, alter and drop the database and which is an instance level permission.

Note: If any login can perform only the specific activities as per role defined to the account.

3. Disk admin:

User can able to manage disk level files like MDF, NDF and LDF files.

Ex: User can able to move the files from one drive to another drive if member of disk admin permissions.

4. Process admin:

1. User can see list of processes which are running in Sql server including system and user defined process.

SPID: 1-50 [System defined process]

SPID: >51 [User defined process]

2. User can kill the other user defined spid's.

Note: User can not kill system defined SPID

User can not kill his own process id.

5. Security admin:

q

1. New login can create
2. Password reset can be done
3. Login removal from SSMS
4. Login can read SQL Server logs from SSMS.

Note: How to read Sql server logs?

SP_READERRORLOG-----It read from current log

SP_READERRORLOG 1-----it read from archive log1.

6. Server admin:

User can be able to change the instance level properties

Ex: Memory changes to the instance

Processor changes\allocations

Authentication mode change

From windows > Sql and SQL > to windows

7. Setup admin:

User can be able to configure linked servers.

Can be able to configure database email

8. Sysadmin:

User can have full control on the SQL Server instance and which is an administrator on SQL server.

Note: Never grant Sysadmin permissions to others\teams until there is requirements.

Creation of the login by using T-SQL:

q

Create login [login name] with password='admin123\$\$'--- SQL authentication

Create login [hostname\account name] from windows --- Windows authentication

DATABASE LEVEL ROLES:

A B D⁴ O S

DB_Accessadmin

DB_Backup operator

DB_Datareader

DB_Datawriter

DB_Ddladmin

DB_Denydatareader

DB_Denydatawriter

DB_Owner

DB_Securityadmin

User can get specific permissions at database level.

User creation:

CREATE USER [username] FOR LOGIN [loginname]

Note: Without login we cannot be able create user.

Note: When we map any login with database by default that login name create as user in specific database.

1. DB_Access admin:

1. can able to create users in the database
2. can able to drop the users from the database

2. DB_Backup operator:

1. User can able to perform backup operations in Sql server

Syntax: Backup database dbname to disk='path'

2. Can be able to trigger manual check point in the database.

Checkpoint [time interval]

3. DB_Data Reader:

User can only able to perform data reading from all tables with in the specific database.

Query Type: Select

4. DB_Data writer:

User can only able to perform data write into all tables with in the specific database.

Query Type: Insert

5. DB_DDL admin:

User can able to perform

Create

q

Alter

Drop

Ex: Create table\Alter table\ Drop table

6. DB_Deny Data reader:

User cannot be able to read the data

7. DB_Deny data writer:

User cannot be able to write the data

Note:

1. If we have db_owner & DB_Deny data reader or writer permission then we can be able to perform all operations on database except reading\writing...Preference goes to DENY
2. If we have sys admin & DB_Deny data reader\write permission then we can be able to perform all operations on database...Preference goes to SYSADMIN.
3. Provide reader and deny data reader permission to same account then user cannot be able to read the data.

8. DB_OWNER:

Users have full permissions at database level.

9. DB_Security admin:

User can get permissions of giving roles to other users at specific table\store procedure\views\functions...etc

Note:

Grant: Can grant permissions to the user account.

With grant: User gets permissions and at the same time user can pass the permissions to other users.

Deny: Deny the permissions...

10. Public:

Basic connectivity to the instance and visibility to the databases.

Limitations:

- > Public roles cannot be deleted
- > Public roles cannot be altered
- > Public roles cannot be unchecked
- > Public role is by default common for every databases

Note:

1. Always keep database owner as SA account which is inbuilt SQL Server account and we never delete that account.
2. Any account is using as owner for specific database you cannot be able to delete until you change the owner of the database.
3. You cannot be able to delete the login if any active sessions are running.
4. Unlimited logins and users possible in SQL Server.

Schema:

- > It is a collection of objects under a database principal (user)
- > It is always present in database
- > If we not mentioned any schema for user, he occupies the default schema-(db)
- > Every principal has schema

q

Create schema <schema name>

Enable & Disable App role:

Sp_setapprole 'approlename','password'

Unset_setapprole 'approlename','password'

OBJECT LEVEL ROLES

- > Insert
- > Alter
- > Control
- > Delete
- > Select
- > References
- > Take ownership
- > Update
- > View definition

Note:

- > If I grant any database level role then can get permissions on all tables inside the database
- > If I grant any table level role then can get only on specific table set permissions and cannot access other tables due to object level restriction.

PRINCIPALS:

- > Principals are entities that can request SQL Server resources like other components of the SQL Server authorization model, principals can be arranged in a hierarchy.
- > Every principal has a security identifier (SID) which is presents in hexadecimal part.

Windows- level principals:

- > Windows Domain Login/Group
- > Windows Local Login/Group

SQL Server-level principals:

- > SQL Server Login
- > Server roles

Database- level principals:

- > Database user
- > Database role
- > Application role
- > If we want to give permissions to 100 users, we need not give manually 100 times to users, simply we can put them as a group, with that group we can give permissions at one time.
- > If we want to see the principals in our server through below quires.

Select * from sys.server-principals

Select * from sys.database-principals

Principal types in SQL Server:

U—Windows user

G—Windows Group

S—SQL User

A—Application role

R—Database role

C—User mapped to a certificate

q

K—User mapped to an asymmetric key

Ex: Logins and users

SECURABLES:

> Securables are the resources to which the SQL Server Database Engine authorization system regulates access.

> The Securables scopes are server database and schema

Securable scope	Securables
Server	Endpoint , Login , Database
Database	User , Role , Application role Assemblies, Message type, Route, Service, Remote Service Binding, Full-Text Catalog, Certificate, Asymmetric key, Symmetric Key, Contract, Schema
Schema	Type,ML Schema Collation, Object
Objects	Aggregate, Constraint, Function, Procedures, Queue, Statistic, Synonym, Table, View

What you are going to secure

Ex: Instance, databases, tables, sp, functions, triggers. etc

Default logins:

1.BUILTIN\Administrators: This one of the o\s administrator group where if any account is a member of O\S level admin group can get directly access to SQLServer without adding the account in SQL Server with SYSADMIN permissions.

Note: From SQL Server 2008 onwards BUILTIN\Administrators group is removed. Even any member included in o\s administrator group should add under SQL Server instance as a login.

Windows login type and have Sysadmin by default

2. NT AUTHORITY\SYSTEM: This is one of the local system account which is windows authentication and contains Sysadmin permissions.

3. WIN-5ROTNADG8A8\SQLServer2005MSFTEUser\$WIN-5ROTNADG8A8\$B14SQL2K5: ---Login for full text search service

4. WIN-5ROTNADG8A8\SQLServer2005MSSQLUser\$WIN-5ROTNADG8A8\$B14SQL2K5: ---Login for Main service and which have Sysadmin

5. WIN-5ROTNADG8A8\SQLServer2005MSagtUser\$WIN-5ROTNADG8A8\$B14SQL2K5: ---Login for Agent service and which have Sysadmin

6. SA: Which is the default Sql authentication account and contains Sysadmin permissions.

Major improvements in 2008\8r2\12:

1. Built-in administrator group is removed.
2. Certificate is on the new feature introduced. [To improve more security]
3. Add current user is one of the option added while installing SQL Server.

SECURITY HARDENING RULES:

Hardening is nothing but protecting the Sql server from the threats or end users

- > Keep always SA password as strong. [Default pwd: SA]
- > Remove unnecessary logins from SQL Server.
- > Remove unnecessary users from SQL Server
- > Remove built-in\administrator from SQL Server
- > Always try to create windows authentication instead of SQL Server authentication
- > Remove unnecessary roles
- > Always create a group instead of individual logins.
- > Never share SA password to other teams
- > Never give Sysadmin permissions to any other teams except DBA team until strong reason.

How to enable or disable logins:**ALTER LOGIN [LOGIN NAME] ENABLE\DISABLE**

Login Failure Error number and states:

User name wrong: 18456 state 5

Password wrong: 18456 state 8

Password expired: 18456 state 18

Account locked out: 18456 state 1

How to create a service account in real time?**Real time process:**

- > Windows team create always service accounts in domain servers
- > Server account "passwords never expire" option should enable and "USER must change the password" option should be disable or uncheck.

q

> Short cut to open user management "DSA.MSC"

How to change any service account from local to domain?

Go to service> properties> Logon >Set domain account.

Note: changing account for the service required restart of SQL Server only specific service.

New feature in SQL Server 2000\2005 \8\8r2:

Database level roles can be added which is not fixed.

Note: From SQL Server 2012 onwards user defined servers can be created

Syntax:

Role Creation:

USE [dbname]

GO

CREATE ROLE [RExecRole] AUTHORIZATION [dbo]

GO

Drop Role:

USE [dbname]

DROP ROLE [RExecRole2]

GO

Note: In real time apps team define this roles and request DBA team to execute the script.

How to create a group and provide the permissions to group?

Steps:

1. Create users at o\s level
2. Create a group at o\s level
3. Add users into the group

Above 3 steps performed by windows team

4. Add the same group as a login into Sql server
5. Provide permissions to group.

Note: when any requirement to provide highest level of permissions and member in a group, then add that particular account as a login and provide permissions only.

How to provide sp level permission?

GRANT EXECUTE ON [store procedure name] TO [user name]

SQL Server security queries:

1. How to get Login information in Sql?

Select * from sys.syslogins

- > Login name
- > Login authentication type
- > Password
- > Server level roles
- > Login created date
- > Login modified date

2. How to get user information in Sql?

Select * from sys.sysusers

> User name

> User created date and time

> User modified date and time.

Default Users:

SQL Server ships with ten pre-defined schemas that have the same names as the built-in database users and roles. These exist mainly for backward compatibility

Every database has 4 default users:

1. SYS

2. INFORMATION_SCHEMA

3. GUEST

4. DBO

SYS & INFORMATION_SCHEMA:

The SYS & INFORMATION_SCHEMA schemas are reserved for system objects. You cannot create objects as a guest. Permissions granted to the guest user are inherited by users who have access to the database, but who do not have a user account in the database.

GUEST:

Each database includes a guest. Permissions granted to the guest user are inherited by users who have access to the database.

To Enable: Grant connect to GUEST

To disable: Revoke connect from GUEST

DBO:

The dbo schema is the default schema for a newly created database. The dbo schema is owned by the dbo user account. By default, users created with the CREATE USER Transact-SQL command have dbo as their default schema.

If any member provided as Sysadmin role then can able to access all the database by using DB_owner role.

SQL Server certificates:

Server principals with names enclosed by double hash marks (##) are for internal system use only

RECOVERY MODELS

Recovery Models:

- Recovery models are designed to control transaction log maintenance
- Recovery models control the behaviour of the log file.

The recovery models in SQL Server are

Full

Bulk Logged

Simple.

FULL RM:

- In FULL recovery model every transaction is logged into the transaction log file (as per WAL).
- This recovery model is generally used in Production databases (i.e. OLTP based systems)

Advantages:

- 1) Minimal/No Data Loss
- 2) Point-in-time Recovery

Disadvantages:

- 1) Performance Overhead and large transactions at times can take more time
- 2) Disk space consumption is too high
- 3) Requires manual intervention of DBA for controlling the Tlog file

BULK-LOGGED Recovery Model:

- In Bulk-logged recovery model every transaction is logged into the transaction log file, but bulk insert operations are minimally logged.
- Bulk Insert operations are SELECT INTO, BULK INSERT, TEXT/IMAGE, Online INDEXING

Advantages:

- 1) Performance Benefit for bulk operations
- 2) Disk space utilization can be reduced when compared to Full Recovery model during bulk insert operations

Disadvantages:

- 1) Chances of data loss if bulk insert operations fail
- 2) May or may not be possible to perform point-in-time recovery
- 3) Bulk logged recovery model is used in special requirement cases where bulk insert operations have to be performed with a time constraint and generally data loss is compromised to BULK LOGGED.

Simple Recovery Model:

- In simple recovery model every transaction is logged into the transaction log file, but at regular intervals the transaction log file is TRUNCATED whenever a CHECKPOINT operation occurs. Simple recovery model is generally used in Development environment where Database Priority/Point-in-time priority is less

Advantages:

- 1) Transaction log file growth can be controlled with regular truncation that occurs
- 2) Less DBA intervention in controlling log file growth

Disadvantage:

- 1) No Point-in-time recovery possible
- 2) Data loss chances are more
- 3) Point-of-failure chances are more

To check current recovery model:-

SELECT DATABASEPROPERTYEX ('dbname', 'RECOVERY') As [Recovery Model]

To set Recovery model:-

Use Master

ALTER DATABASE dbname SET RECOVERY SIMPLE/BULK_LOGGED/FULL

System database recovery models:

Master - Simple

MSDB - Simple

TEMPDB - Simple

MODEL – Full

RECOVERY MODEL DIFFERENCES:

FULL RECOVERY	BULK LOGGED	SIMPLE RECOVERY
WAL concept 100% applicable. Every transaction write into transaction log	WAL Concept is applicable only except bulk transactions.	WAL concept 100% applicable. Every transaction write into transaction Log
Transactions are fully logged	Transactions are minimally logged	Transactions are fully logged
Point in time recovery is possible	Point in time recovery is not possible	Point in time recovery is not possible
No data loss or very minimal data loss	Data loss only when we perform bulk operation	Data loss chances are very high
Performance impact is slight	Performance impact is slight in normal transaction but where as in bulk transaction no major performance impact	No performance impact
Disk consumption is high	Disk consumption is high when normal transactions but in bulk Disk consumption is low	Disk consumption is Less
Use for production OLTP environments	Use for production when there is any bulk transactions	Always use for development servers

Database Access modes:

- 1. Multi_user:** all users can be able to access the database.
- 2. Single_user:** Only one user can access database at one time.

3. Restricted mode: The users who are having only Sysadmin, db owner permissions can only access the database.

BACKUPS

Backups

> A copy of data that is used to restore and recovery the data after a system failures.

Use: Backup is safe guard to databases because data may can loss due to many failures. Such as media failures, user errors, hardware failures and natural disasters etc... with good backups, we can recover database from failures.

Backups: Copy of the database

Backup are light weight threads in SQL Server which mean never consume more resource like CPU, MEMORY.....etc

Types:

Types of backup:-

1. FULL BACKUP, 2. DIFFERENTIAL BACKUP, 3. TRANSACTION LOG BACKUP, 4. FILE AND FILE GROUP BACKUP, 5. MIRROR BACKUP, 6. SPLIT OR STRIPE BACKUP, 7. COPY ONLY BACKUP, 8. TAIL LOG BACKUP, 9. PARTIAL BACKUP, 10. PATRIAL DIFF BACKUP

1. Full database Backup: [.BAK]

> This backs up the whole database. In order to have further differential or transaction log backups you have to create the full database backup first.

Syntax: BACKUP DATABASE dbname TO DISK = 'c: \filename.bak'

>To find percentage completion of the backup file?

Syntax:

BACKUP DATABASE dbname TO DISK = 'c: \filename.bak 'with stats

Note:

> When we are performing any full database backups the backup file includes both committed and uncommitted transactions as well.

> When we perform full backup and in same time if any active transactions then upto point the active transaction included under full backup file.

2. Differential backup:

> Differential database backups are cumulative. This means that each differential database backup backs up the all the changes from the last Full database backup and NOT last Differential backup.

Syntax: BACKUP DATABASE dbname TO DISK = N'c:\filename.bak' WITH DIFFERENTIAL

Note: What are the pages are modified?? That information stores in BCM pages after full backup. SQL Server before trigger the diff backup verifies in DCM page to perform diff backup.

Note: Differential backup always depends on recent full backups.

3. T-log [Transaction log] backups:

> It takes complete log file (.LDF) and never takes backup of any .MDF OR .NDF files. Full backup is the base.

Syntax:

Backup log dbname to disk='path\filename.trn'

Points:

> Log backups are incremental

> Log backups are not possible under simple recovery model due to truncate operation after check point.

> First time log backup depends on recent full backup and after if any number of time log backup always depends on recent log backup.

> Every log backup contains one sequence number.

Note: When any log backup completion information stores under BULK CHANGE MAPPING [BCM] page.

If SQL engine need to take any more log backups checks the sequence from this BCM pages.

DCM -Differential change mapping from this page engine reads when the full backup was.

4. Copy only backup:

1. Copy-only backups are new in SQL Server 2005
2. Used to create a full database or transaction log backup without breaking the log chain
3. Copy only backup is used in production servers when any high availability concepts are configured.

Syntax: BACKUP DATABASE dbname TO DISK = N'filename.bak' WITH COPY_ONLY

Point: A copy-only full backup can't be used as a basis for a differential backup, nor can you create a differential copy only backup.

Note: Can take copy only differential and copy only t-log backups.

Copy only diff and t-log always depends on copy only full backup

Copy only Diff:

BACKUP database sbidb

to disk='path\sbidb_diff_copyonly.bak'

with copy_only, differential

Copy only log:

BACKUP log sbidb

to disk='path\sbidb_tlog_copyonly.trn'

with copy_only

5. Mirror backup:

1. Purpose is to maintain multiple copies of same data into different files.
2. Mirrored backups simply write the backup to more than one destination.
3. You can write up to four mirrors per media set. This increases the possibility of a successful restore if a backup media gets corrupted

Syntax:

Backup database dbname to disk='e: \dbname_Mirr1.bak'

Mirror

to disk='e: \test_Mirr2.bak' with format

Note: Cost and disk space utilization are high in this type of backup.

6. Partial database backup:

> Partial backups were introduced in SQL Server 2005. They are designed for use under simple recovery model to improve flexibility for backing up very large databases that contain one or more READ-ONLY file groups

> This backup only the file\file groups in read_write only mode. Skips the file \file groups in read_only mode.

Syntax:

BACKUP DATABASE dbname READ_WRITE_FILEGROUPS TO DISK = 'C: \TestBackup_Partial.BAK'

Partial Differential Backup:

The modified pages of READ_WRITE_FILEGROUPS will be backed up during partial differential backup.

Syntax:

BACKUP DATABASE dbname READ_WRITE_FILEGROUPS TO DISK = 'C: \TestBackup_Partial.bak'
with differential

7. Striped [Split] Backup:

This type of backup mainly used the case where there is a disk space issue in the server.

[Split] Backup: Striped backup will split the backup into parts and can be very useful during tight space constraints. Striped backup is taking backup onto different locations (i.e. parts of backups but not a mirror)

Syntax:

Backup database dbname to disk='C: \dbname_Part1.bak', disk='c: \dbanem_Part2.bak'

Note: use only when disk space constraint

Note: If db size 100 GB, disk1: 60 50 GB free. disk2= 50 GB of free space:

Even this case we can take a backup by splitting into 2 drives 50 GB + 50 GB.

Note: To restore split backup files both backup are mandatory. If you lost 1 split backup files then you cannot be able to recover the data by using another backup file.

> In Sql server 2005, if backup starts Sql engine directly start writing into backup file without checking any disk space.

After 99% backup fail if there is no disk space.

> Whereas from SQL 2008 version onwards, SQL Server engine always checks the disk space and then it start writing the data if sufficient space available .If not fails at the beginning.

Error: 112 (Lack of disk space)

8. Tail log backup:

This backup only apply when there any database crash.

Crash cases: - MDF file corrupt\Missing. LDF file corrupt\Missing, Page corruption, Header corruption, internal db errors

Syntax: backup log [dbname] to disk='path\taildb_TAILLOG.TRN' with no_truncate

No_truncate: Without truncate VLF'S gives active transaction log backup

Note: Always tail log backup is possible?

MDF\NDF corrupted: Always 100% possible

LDF corrupted: May or may not be possible

Note:

1. When database is in online tail log backup never works.
2. Tail log backup cannot work when database is in simple recovery model.

9. File and file group backup:

Desired file\file group backup is possible by using this method.

File Backup:-

Syntax:

BACKUP DATABASE [DBNAME] FILE = N'FILENAME' TO DISK = N'Path'

File Group Backup:-

Syntax:

BACKUP DATABASE [DBNAME] Filegroup = N'Filegroup Name' TO DISK = N'Path'

Note: By default LDF file gets backed up when you trigger any file\file group backup

Note: File or file group backup does not required base as a full backup.

If you want restore file\file group backup, MDF file should restore first.

q

>>I have a backup file? How to find the whether backup file is valid or not?

RESTORE VERIFYONLY FROM DISK='PATH\BACKUPFILENAME'

Output:

Backup file validity status

>>I have a backup file? How to find the how many files are there inside of the .bak\trn file?

RESTORE FILELISTONLY FROM DISK='PATH\BACKUPFILENAME'

Output:

Number OF FILES

Path

SIZE OF THE FILE

>>I have a backup file? How to find the version of Sql server by using the backup file?

RESTORE HEADERONLY FROM DISK='PATH\BACKUPFILENAME'

Output:

DBNAME

Compatibility

Version

Server name

Login name

Db size

Collation

Backup start time

Backup finish time

Database	Full	Differential	Tlog
MASTER	YES	NO	NO
MODEL	YES	YES	YES
MSDB	YES	YES	NO
TEMPDB	NO	NO	NO

Backup Media Terms and Definitions:

Append:

Add additional modifications into the existing backup file

Overwrite:

Entire data gets taken into backup file and due to this takes more time.

Backup compression: New inbuilt feature in Sql server 2008 onwards. The main purpose is you can save the disk space and keep more number of day's backup files.

Compression option values

1---10%

2--20%

3--30%

4--40%

...10

Note: Backup compression supports only in Enterprise edition.

Backup Set:

A backup set contains the backup from a single, successful backup operation. It can be a FULL/Diff/Tlog backup.

Media Set or backup device:

A media set is an ordered collection of backup media, tapes or disk files, to which one or more backup operations have written using a fixed type and number of backup devices.

Media Set Family:

Backup split into two files and called backup files belong to same family

> Verify Backup Option:

Just perform backup file validity by using RESTORE VERIFYONLY

> Checksum:

Checksum is verified to ensure if data is corrupted or not and performs mathematical calculations.

New in Sql server 2005

> Continue backup on error:

If any errors are reported in database then backups are generally fails. If still want to take backup you can take by selecting "Continue backup on error". But again backup file become corrupted.

Note: NEVER ENABLE THIS OPTION WHEN YOU TRIGGER BACKUP

FOR LARGE DATABASE NEVER USE RESTORE VERIFY ONLY OPTION AND IT TAKE HUGE TIME TO SCAN EACH PAGE.

Note: Backup compression is new feature in SQL server 2008 onwards and only possible in ENTERPRISE EDITION.

Upto SQL 2005 or lower version to compress the backup file projects use third party tools.

RESTORE METHODS

Restore methods:

1. Full backup file restore:

To restore entire database from backup file

Syntax:

Restore database [dbname] from disk='path\filename'

RESTORE AND ITS PHASES:

- Restoring is the process of copying data from a backup and applying logged transactions to the data to roll it forward to the target recovery point.
- A restore is a multiphase process. The possible phases of a restore include

> **Data copy,**

> **redo (roll forward),**

> **undo (roll back) phases**

Data Copy Phase:

The data copy phase involves copying all the data, log, and index pages from the backup media of a database to the database files.

The Redo Phase (Roll Forward):

From LDF both committed and UN committed transactions sent to MDF file.

Undo Phase (Roll backward Phase):

The committed transaction should stay in MDF and only UN committed transactions sent to LDF file for further processing.

Restore Full backup and diff backup files:

When you have multiple backup files you should follow some standard

1. Restore full backup with NORECOVERY
2. Restore diff backup with recovery then db comes into online

RECOVERY METHODS

Recovery Methods:

WITH RECOVERY:

Additional backups file not allowed and bring the database only immediately.

WITH NO RECOVERY:

Additional backups file are allowed and bring the database into restoring state immediately. In this feature diff or log backups can be restored.

WITH STAND BY:

Additional backups file are allowed and bring the database into standby\read_only state immediately. In this further diff or log backups can be restored.

Note: In this mode users can read the data in middle of restoration process. But no write possibility.

Example: I have 2 backup file full and diff. I want to restore and follow below script.

Restore database icicidb from disk='U: \Program files\MSSQLSERVER\Backups\icicidb_full.bak'

With norecovery

Restore database icicidb from disk='U: \Program files\MSSQLSERVER\Backups\icicidb_diff.bak'

SQL Server restore process:

Case: If database crashes how you recover the data?

1. Whenever database crashes attempt tail log backup to recover active transactions
2. Check for the recent backups [full, diff or t-log]
3. Restore full backup with no recovery
4. Restore diff backup with no recovery
5. Restore t-log backup with no recovery
6. Restore TAIL backup with recovery

Note: Tail log always need to restore last and tail log need to apply first before starting recovery process.

Project backup strategy:

Types of backups and Backup strategy in real time:

Full database backup: Saturday 10:00 PM

Differential backup: Every day 09:00 PM

Tlog backup: Every 15 min in a day.

Backup strategy 1:

Every Sunday: 10:00 AM full backup

Every day: Diff backup one time @ 11:00 PM

T-log backup: every 15 min once daily

@ DB crashes @ Thursday @ 10:03 AM...How to recover?

Recovery process:

1. Whenever database crashes always attempt tail log backup to recover active transactions.
Once tail log backup taken then start recovery process by using existing backup files

> Restore recent Sunday full backup with "NORECOVERY"

> Restore recent "Wednesday" differential backup with norecovery

> Restore each transaction log backup after wed 11:00PM with no recovery upto Thursday 10:00 AM

> Last restore tail log backup to recover 3 min of data with recovery.

Finally database comes online upto 10:03 AM.

q

Note:

> If we restore last backup with no recovery instead of recovery then Please use below query to bring db online.

Restore database [DB NAME] with recovery

> If LDF file is corrupted then DBA cannot be able to recover the transaction from LDF file.

In this case data loss chances are high.

> Standby mode restore:

RESTORE DATABASE [DBNAME]

FROM DISK = 'Path\backup file'

WITH STANDBY = 'U: \\ROLLBACK_UNDO_hdfcdb.BAK'

Backup strategy 2:

Full backup: Sunday 10:00 AM

Diff backup: Every day: 10 PM [After business hours]

T-log backup: Every 15 min [Every day]

Steps to recover the database from backup when database crashes?

DB crashes @ Sunday 9:49 AM

Recovery Process:

1. Whenever database crashes then attempt tail log backup to recover active transactions in the log file.

Backup log database name to disk='path' with no_truncate

q

2. Take recent full backup which happen recent Sunday 10:00 AM and restore with no_recovery...
3. Take recent diff backup Saturday @ 10 PM and restore with no_recovery.
4. Restore each t-log backup file after differential backup Saturday 10:15 PM...Since t-logs are incremental. Restore with no recovery [If tail log backup success]
[If tail log backup not work then last t-log backup restore with recovery].
5. Last restore tail log backup with recovery to bring database online upto the point.

Note: If LDF file corrupt 100% not possible to attempt TAIL LOG backup with error "fail activation failure"

Backup strategy 3:

Full backup-10:00 AM

DIFF: 11:00 AM

TLOG: 12:00 PM

Full: 12:30 PM [BUT BACKUP FILE IS CORRUPTED]

DIFF: 1:00 PM

Recovery process:

1. Tail log attempt
2. Take last to last full backup due to recent full backup @ 12:30 PM got corrupted and restore with no recovery
3. Take diff @ 11:00 AM and do the restoration with no recovery
4. Restore t-log @12:00 PM and restore with recovery...

Note: If full backup corrupted after recent diff or log cannot be able to use.

Note: Full, Diff1, Tlog1, Copy_only, Diff2

Recovery process:

Cannot be able to recover by using COPY_ONLY + DIFF2

You should use FULL+ DIFF2 to recovery the database.

Copy only backup never create distribute the backup sequence .son you cannot be able to use diff 2 with copy only.

Reasons for backup failure:

Lack of disk space [Error: 112]

Permission issue

Network issue

Wrong syntax

Db is unavailable

LSN mismatch

Without full backup user trying to take diff backup

SQL Server services are offline]

DB is in simple recovery model but user trying to perform tlog backup.

Reasons for restore failure:

Lack of disk space [Error: 112]

Permission issue

Network issue

Wrong syntax

LSN mismatch

q

Without full backup restore user trying to restore diff or log backup

SQL Server services are offline]

Restore with move option:

If required to restore one database backup file into another database in the same instance then use "WITH MOVE" option

Syntax:

RESTORE DATABASE [TESTDB]

FROM DISK = 'U: \Program files\MSSQLSERVER\Backups\HDFCDB_FULL.BAK'

WITH MOVE N'HDFCDB' TO 'E: \TESTDB.mdf', MOVE N'HDFCDB_log' TO N'L:\TESTDB_1.ldf'

>MDF data move from old file to new data file

>Ldf data move from old ldf to new ldf file

MSDB Tables for backup information:

1. Dbo.backupfile:

Stores:

Database file,

Number of pages backed up for each file (Backup_up_page_count),

Size of the file, file group.

q

2. Dbo. Backupfilegroup:

File group name

File group id

Backup set

Is default

Is read_only

3. Dbo.backupmediafamily

Physical_device_name [Physical path for your backup file]

Media set

Media count

4. Dbo.backupmediaset

media_family_count

Is password _protected...

5. Dbo.backupset

Backup start date

Backup finished date

Backup Isn

Check point Isn

TYPE OF BACKUP

[D: FULL, I: DIFFERENTIAL, L: Log Backup, f: file group backup]

DB Name

q

Backup triggered login name

Size

Password protected

Collation setting...

Compatibility

MSDB tables for Restore tables:

1. Dbo.restorefile

File number

Destination physical drive

Destination physical file name

2. Dbo.restorefilegroup

File group name

Restore History

3. Dbo.restorehistory

Restore date

Destination database name

What type of backup file restored.

q

Note:

2nd method to see history related to backup or restore events from
\\DATABASE>REPORTS>STANDARD REPORTS>BACKUP AND RESTORE EVENTS

Case: 1

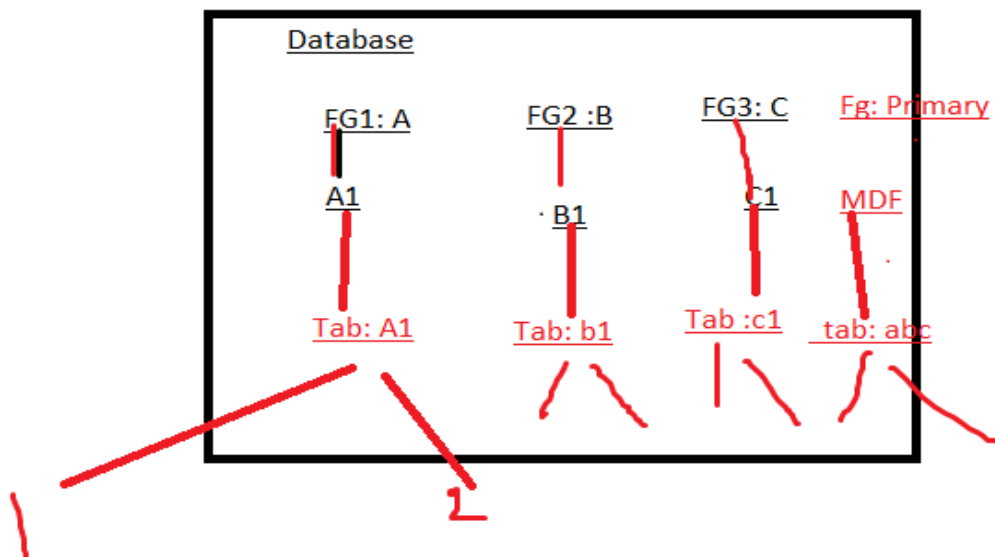
If require to keep same physical files in same instance with different database then require to use "With move" option....

RESTORE DATABASE NEWDBNAME FROM DISK='Backup File Path'

WITH MOVE 'logical name' TO 'new physical path'

PIECE MEAL RESTORE

- Piecemeal restore, introduced in SQL Server 2005, allows databases that contain multiple filegroups to be restored and recovered in stages.
- Piecemeal restore involves a series of restore sequences, starting with the primary file group and, in some cases, and one or more secondary filegroups.
- Piecemeal restore maintains checks to ensure that the database will be consistent in the end.
- Piecemeal restore works with all recovery models, but is more flexible for the full and bulk-logged models.
- Every piecemeal restore starts with an initial restore sequence called the partial-restore sequence. Minimally, the partial-restore sequence restores and recovers the primary file group.
- During the piecemeal-restore sequence, the whole database must go offline. Thereafter, the database is online and restored filegroups are available...



Piece Meal Restore:

Purpose: we restore the database pieces by piece in Sql server by restoring the database partially.

Note: -Real time : if any users want to access a specific table \tables which resides in file group ; you have an option to restore the database a specific file instead of entire database by using piece meal restore

Advantages: - Saves the time, saves the Storage cost, can test quickly, Can control the user at specific level.

Note: In standard edition piece meal restore is not possible in SQL server.

Piece Meal Restore:

ABC - Database

A - FG - A1

B - FG - B1

C - FG - C1

Create table A1 (sno int, sname varchar (50)) on A

Create table B1 (sno int, sname varchar (50)) on B

Create table C1 (sno int, sname varchar (50)) on C

Create table ABC (sno int, sname varchar (50))

Insert into A1 values (1,'A')

Insert into B1 values (1,'B')

Insert into C1 values (1,'C')

q

Insert into ABC values (1,'ABC')

Backup database ABC to disk=N'c:\dummy\ABC.bak'

Backup database ABC FILEGROUP='Primary' to disk=N'c:\dummy\ABC_Primary.bak'

Backup database ABC FILEGROUP='A' to disk=N'c:\dummy\ABC_A.bak'

Backup database ABC FILEGROUP='B' to disk=N'c:\dummy\ABC_B.bak'

Backup database ABC FILEGROUP='C' to disk=N'c:\dummy\ABC_C.bak'

Insert into A1 values (2,'A1')

Insert into B1 values (2,'B1')

Insert into C1 values (2,'C1')

Insert into ABC values (2,'ABC1')

Backup log ABC to disk=N'c:\dummy\ABC.trn'

Piece Meal Restore commands:

Restore database ABC FILEGROUP='Primary'

from disk='c: \dummy\ABC_Primary.bak'

WITH NORECOVERY, PARTIAL

q

Restore log ABC from disk='c: \dummy\ABC.trn'

Restore database ABC FILEGROUP='A'

from disk='c: \dummy\ABC_A.bak'

WITH NORECOVERY

Restore log ABC from disk='c: \dummy\ABC.trn'

> How to take backup into network share?

> Third party backup tools in market

> Backup myths

1. How to take backup into network share?

Backup database [dbname] to disk='\\ipaddress\drive\$\folder name\filename.bak'

> Third party backup tools in market

1) DELL [Quest] Lite speed

2) Idera SQL Backup

3) IBM Tivoli Storage Manager

4) Veritas NetBackup

5) Symantec Backup Exec

6) Acronis

7) EMC NetWorker

8) Backup and FTP

9) ZAMANDA (AMANDA)

10) Netapp storage tool

DATABASE REFRESH

What is database refresh?

Taking a backup into production Sql server database and restoring into development Sql server then this is called "DATABASE REFRESH"

Steps:

1. Take backup into production server
2. Always use copy method to copy the backup file from prod to test\dev server.

If Case: I don't have space in the dev\test server then what I have to do?

Step1: Check the drive whether any unnecessary \old backup file there then delete those to claim some additional space. Try to copy the backup after

OR

Step 2: If I don't have any old backup files then inform to storage team to expand the disk or add new disk...Meanwhile send email to requestor.

3. Once backup file is copied to dev\test then please do restore.
4. Move backup file from source to destination server by mapping the drive from either of the servers.

Path: Go to run > type \\10.10.10.1\d\$

5. Restore in destination server by changing the root path with destination instance.
6. Inform application team\requestor to cross check the data in dev\test server.
7. Issue after database refresh: Whenever we perform any backup and restore between different instances we used to get ORPHAN user issue.

What is orphan user: A user without having login called as "ORPHAN USER?"

How to find: `sp_change_users_login @action='report'`

How to fix: `sp_change_users_login 'update_one','user name', 'login name'`

If orphan user not fixed, then user cannot be able to connect.

POINT IN TIME RECOVERY

Point in time Recovery:

This method help you to recover the data upto point from the backup file always by using below

Syntax:

RESTORE LOG DBNAME FROM

DISK = N'PATH\BACKUP FILE'

With STOPAT = N'YY-MM-DD HR: MM:SS'

Note:

1. When we change any recovery model from full or bulk logged to simple or simple to full\bulk logged then you should trigger recent full backup to form LSN Number. Otherwise LSN mismatch and backup may fail.
2. Higher version backup files are not possible to restore in lower version. But vice versa is possible.
3. A full or differential backup clears the log: NO...But in directly when trigger these kind of backups check point operation raise.
4. Backups read data through the buffer pool: NO, Never take backup from buffer pool

Reason: every time to get pages to buffer pool is not possible and it impact the performance.

5. Backups perform consistency checks (all DBCC CHECKDB)

NEVER run check db

6. If the backup works, the restore will too?

No...Not sure

> If one spid trigger add file and another spid trigger backup then until backup completion add file in waiting status.

PAGE LEVEL RECOVERY

New feature in SQL Server 2012

- How to find which page got corrupted?

Run DBCC Checkdb (dbname)

- How to fix this only page corruption?

Do page level restore?

GUI PATH:

Database> Restore> Page> Run Checkdb> Select backup file> ok

Command:

Restore database dbname page (fileid: pageid) from disk='path'

JOBS & MAINTENANCE PLANS

JOBS

By using this job DBA can automate Maintenance tasks by defining specific schedule.

Minimum requirements:

- > SQL Server agent service should up and running
- > Job owner should have permissions
- > Job steps and schedules are configured properly.

> How to find job is ran successfully or failed?

Go to job > view history

> What is use of job activity monitor?

By using this tool

- > List of jobs
- > Job status
- > Job schedule
- > Job last ran time
- > Job last ran output
- > Job next run time.

MSDB Tables for Jobs:

dbo.sysjobactivity

dbo.sysjobhistory

dbo.sysjobs

dbo.sysjobservers

dbo.sysjobschedules

dbo.sysjobsteps

dbo.sysjobstepslogs

Disadvantage:

When we have multiple steps in a job we can define separate schedule for each step. The schedule applies to all the steps each time.

Job failure Reasons:

Permission issues --- MSDB read\write

Network failure for the backup

Job disabled

Agent stopped

Database not in online

SQL Services are stopped

T-SQL issue

Disk space issues

Job owner disabled

Note: Purpose of Job user is always should keep "SQL Server agent service account [Local\Domain]"

Note: Multiple jobs can be deleted from "OBJECT EXPLORER DETAILS [F7]"

MAINTENANCE PLANS

Maintaince Plans:

> Newly introduced in SQL Server 2005 with bugs. Fully in SQL 2005 +SP1

> Maintaince plans can be configured in 2 ways

1. Wizard
2. Flow chat

Note:

> From SQL Server 2008 onwards "Ignore offline database" option introduced which can skip if any db is in offline and complete the Maintaince plans.

Where as in SQL 2005 if any db offline Maintaince plan fails.

> Multiple tasks selected in Maintaince plans SQL 2005 then dependent multiple jobs creates. Where as in SQL 2008 onwards only 1 job for multiple tasks or steps.

> Always Maintaince plans execute or run via jobs only.

Maintaince plan types:

Total 11 MP types:

Back Up Database Task

Check Database Integrity Task

Execute SQL Server Agent Job Task

Execute T-SQL Statement Task

History Cleanup Task [NOT automated task]]

Maintenance Cleanup Task

Notify Operator Task [NOT automated task]

Rebuild Index Task

Reorganize Index Task

Update Statistics Task

q

Shrink Database Task [NOT automated task]

Monitor of Maintainece plans:

Method 1: Right click >Job > view history

Method: 2 Right click >Maintainece plan> View history

REAL TIME MAINTAINCE DBA TASKS:

Daily:

Backup -transactional log

Backup- differential

Check Database Integrity Task

Maintenance Cleanup Task

Replication Maintainece jobs

Weekly Maintainece:

Backup- Full

Rebuild Index Task

Reorganize Index Task

Update Statistics Task

Monthly Maintainece tasks:

From DBA side no monthly Maintainece plans

Application Maintainece jobs:

> Purging data

Storage SAN Maintainece jobs:

Moving the backup file from disk to tap according to the retention period.

Note: From windows side; we can also configure automated tasks by using "WINDOWS TASK SCHEDULER"

Note: When DBA can use windows task scheduler to automate Sql related in Sql server express edition we don't have "Sql server agent services". Without agent service cannot run jobs or Maintainece plans. In this case we can use Windows Automated task scheduler.

Note: cannot delete a job without deleting Maintainece plan. First need to delete Maintainece plan which automatically delete dependent job

Maintainece plan MSDB tables:

dbo.sysmaintplan_log

dbo.sysmaintplan_subplans

dbo.sysmaintplan_logdetail

ATTACH & DETACH DATABASE

[DATABASE LEVEL DOWNTIME IS REQUIRED]

> This method mainly help to move the database faster from one instance to another instance or within same instance to move the files [MDF OR LDF OR NDF] between the drives.

Note: Very fast method and save lot of time compare to backup and restore.

Real time steps:

Pre-install Steps:

1. Perform full backup of the database depending on the db size [if you have more down time]
2. Inform application team before start the activity.
3. Disconnect all the user connections.

Install steps:

Method-1 GUI:

> Go to database > right click> task> detach

Note: Detach just drop the connection db from only SSMS

> Copy MDF and LDF from source server to destination server

Note: Always use copy and paste method.

> Attach in the destination server

METHOD-2 QUERY ANALYSER

Attach \Detach:

Process to move database:

Step 1: Detach Database using following script

Exec master

SP_detach_db @dbname = N'dbname'

q

GO

Step 2: Move Data files and Log files to new location

Step 3: Attach Database using following script

```
USE [master]
```

GO

```
CREATE DATABASE [dbname] ON
```

```
(FILENAME = N'Path of MDF file'),
```

```
( FILENAME = N' path of dbname_Log.ldf')
```

```
FOR ATTACH
```

GO

Post-install steps or verification steps:

- > Validate whether database is up and running fine.
- > Inform application team to check the connectivity and data.
- > Change the compatibility number if you perform into higher version of SQL Server.

Note:

Limitations:

- > Attach\detach method not work from higher version of Sql to lower version of SQL Server.
- > System database attach \detach method not work

COPY DATABASE WIZARD

> New feature in Sql server 2005

> This method help to create a set of copy of the database in same instance or different destination instance.

Note:

> If db is in read_only then we can perform copy database method.

> This method uses SSIS services to execute in destination.

SSIS package collects:

Users

Logins

Database

Tables

Views

Store procedures...etc

Limitations:

>System databases

>Databases marked for High availability.

>Databases marked Inaccessible, Loading, Offline, Recovering, Suspect, or in Emergency Mode.

SYSTEM DATABASES OVERVIEW

1. MASTER DATABASE: [DB ID: 1]

> In SQL Server system object data stores in master database logically and physically in resource database.

EX: any TABLE OR VIEW STARTS WITH sys.

> Whenever restart SQL Server ; SQL Server engine checks for MASTER MDF and LDF path from configuration manager>advanced> start up parameter> MDF and LDF file location

Then my SQL Server services start.

Note: Master database starts first whenever restart.

> At the same time write recovery of database information into SQL Server error log.

Note: In SQL 2005 version: Master and resource database files MDF and LDF stores in DATA folder.

Whereas from SQL 2008 onwards; MASTER into data folder and resource files into BINN folder.

Information stores in master:

Linked server

Endpoint

Instance configuration

Other database information

Other database files

Login [SYSXLOGIN]

File group of other databases

2. MODEL Database [DB ID: 3]

> Just act as a template whenever we create user database.

> When you create any user database get all the below properties from model database.

Ex:

File

File size

Recovery model

Collation

Root path....etc

3. MSDB [DBID: 4]

> SQL Server agent related information stores into MSDB

Information stores:

Jobs

Maintaince plans

Alerts

Operator

Database mail

Log shipping

Backup

Restore

Copy database wizard

SSIS\DTS packages

4. Tempdb: [DB ID: 2]

- > Whenever restart SQL Server services then Tempdb MDF and LDF reset to original size by flushing all the temp information.
- > Tempdb always inherit properties from MODEL database except Tempdb MDF size: 8 MB and recovery model: Simple
- > By finding Tempdb created date by using SP_HELPDB or db> right click properties or Tempdb mdf, ldf file date creation: can conclude that is the time when SQL Server services restarted.

Tempdb Stores:

Cursors

Triggers

Functions

Joins

Local variables (#)

Global variables (##)

Indexes

Row version [SQL 2008 new feature]

Table level information

5. Resource database: [DB ID: 32767]

- > New feature in SQL Server 2005 version onwards
- > Physically stores sys object information
- > Any service pack\hot fix\cu entries or updates at resource database
- > Upgrade SQL Server entries happen in Resource database
- > Read_only database
- > Hidden database
- > No entry in master database related to resource database

q

> Only Select queries can work and to find when was the last resource database updated. Also current SQL Server version information stores in resource

```
SELECT SERVERPROPERTY ('ResourceVersion') ResourceVersion,  
SERVERPROPERTY ('ResourceLastUpdateDateTime') ResourceLastUpdateDateTime  
GO
```

OPERATIONS ALLOWED\NOT ALLOWED IN SYSTEM DATABASES:

Operations can perform on System db:

- > Backups should take for master, model and msdb ...Tempdb, resource database backup statement does not work
- > Adding a file possible for only msdb and Tempdb. File groups are only possible for msdb database
- > Can move files [mdf and ndf] from one drive to another drive called "FILE MOVEMENT"
- > Shrinking is possible in Tempdb only

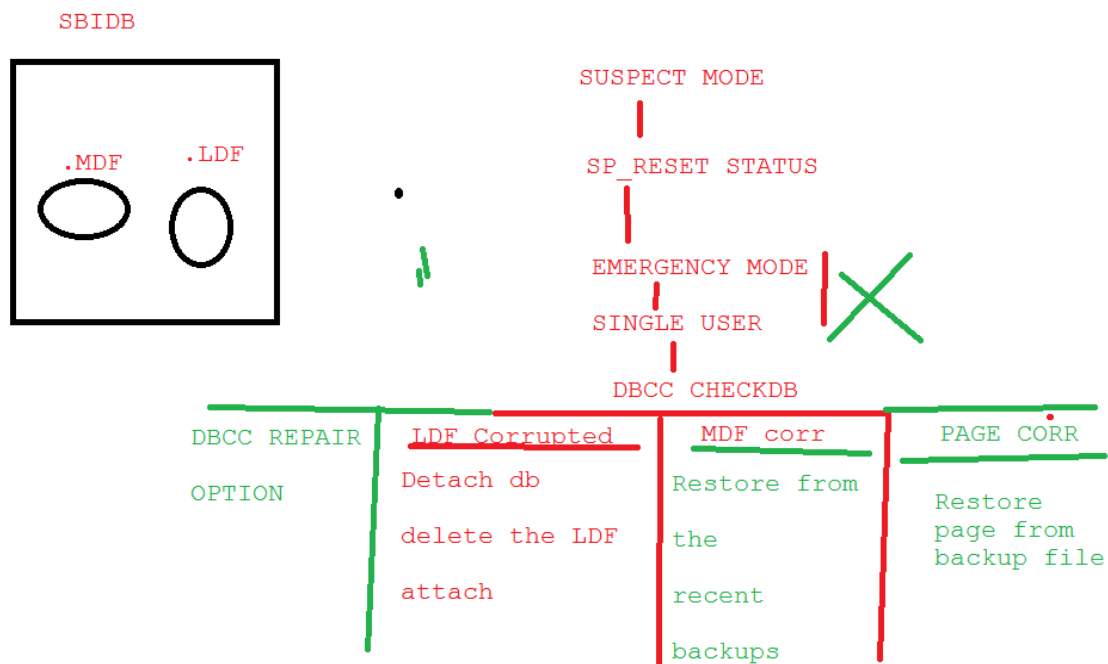
Don't do on system database:

- > Never create user defined tables
- > Never add file or file groups to master and model database

SUSPECT DATABASE

[User db corruption]

Suspect is a state where database becomes inaccessible due to different reasons



Reasons:

- 1) Data and Log files missing or corrupt
- 2) Corruption of pages in the Data and Log files.
- 3) Synchronization issues between data and log files
- 4) Issues that are caused during Recovery/Restoring process
- 5) Sudden shutdown happen to your database\instance...
- 6) Kill spid while transaction is roll back...
- 7) Database Flags in inactive status.

Case 1: LDF File corruption

Steps to Resolve:

1) Identify if database is really in suspect state or not.

Select databasepropertyex ('dbname','status')

2) Attempt to reset the suspect flag using sp_resetstatus

EXEC sp_resetstatus 'test'

3) Set the EMERGENCY mode on for the database for further troubleshooting. Emergency mode is a READ_ONLY state and gives some base for identifying the cause of the issue.

Alter database dbname set emergency

4) Put database in Single User mode, to avoid connection conflicts.

Alter database <Dbname> set Single_user with rollback immediate

5) Run DBCC CHECKDB on the database to identify if the issue is with Data files or Log files.

Running checkdb finds any consistency and allocation errors and if there are no errors found then Data file is considered to be clean. The issue might exist with Log file.

Output should say:

CHECKDB found 0 allocation errors and 0 consistency errors in database 'test'.

6) Detach the database and delete log file from the path.

sp_detach_db @dbname='dbname'

7) After attach database by using below cmd:

CREATE DATABASE databasename on

(FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL.2\MSSQL\Data\databasename.mdf') for attach

Finally database come online with new ldf file.

Inform apps team to test the data from their end.

Note: How to find estimated time for DBCC query?

Select * from sys.dm_exec_requests

Column: "Estimated_completion_time"

Case 2: If data file corrupt?

Restore from recent backup file

Note: If .mdf corrupt db will not go to emergency mode...

Case 3: If page corrupt?

Restore from recent backup file by using below cmd

Restore database dbname page= (file id: pageid) from disk='backup file path'

Note: Page level restore is possible from SQL Server 2005 version by using T-SQL and whereas from SQL 2012 Page restore included in GUI.

Note:

[Late recovery Option: Client approval is required]

If backup is not available and issues found with data file/log file use

DBCC Checkdb ('dbname', REPAIR_ALLOW_DATA_LOSS)

This command repairs the database but has the risk of data loss. Take proper approvals before performing this step.

Note: Your system databases never go to suspect mode.

Background about CHECK DB

If you run checkdb what are the internal queries will execute?

Checks the logical and physical integrity of all the objects in the specified database by performing the following operations:

>> Runs **DBCC CHECKALLOC** on the database:: Checks the consistency of disk space allocation structures for a specified database.

>> Runs **DBCC CHECKTABLE** on every table and view in the database::Checks the integrity of all the pages and structures that make up the table or indexed view.

>> Runs **DBCC CHECKCATALOG** on the database::Checks for catalog consistency within the specified database. The database must be online.

Note: IN SUSPECT MODE NOT POSSIBLE TO APPLY TAIL LOG BACKUP.

SYSTEM DATABASE CORRUPTIONS

MASTER CORRUPT:

> Master is the most crucial database in an instance, if it is corrupt entire instance gets affected.

Master Corrupt

Error number: 3411

Error Message: Timely fashion error

112 error number for disk space

Partially corrupt:

1. If master database is corrupt, it is either completely corrupt or partially corrupt. If partially corrupt, instance will start with -m;-t3608 and if it is completely corrupt instance wouldn't start.
2. Put your instance in single user mode.
3. Restore master database WITH REPLACE option

Restore database master from disk=N'F:\Master.bak' WITH REPLACE

Completely Corrupt:

- 1) Master database doesn't start with /m /t3608 and hence we need to rebuild the master database.

Use command prompt and start rebuilding master database by enter into path of setup.exe
[Path of SQL Server software]

- 2) Rebuild master

Start /wait setup.exe /qb INSTANCENAME=sql2005 REINSTALL=SQL_Engine
REBUILDDATABASE=1 SAPWD=Admin123

Net stop "SQL Server (instance name)"

Net start "SQL Server (instance name)" /m

- 4) Restore master database WITH REPLACE option

Restore database master from disk=N'F:\Master.bak' WITH REPLACE

MODEL CORRUPTION:**Error: 5172, Severity: 16, State: 15.****Error message:**

The header for file '\model.mdf' is not a valid database file header.

Solution:

Model database being one of the crucial database for new database creations and also for Tempdb recreation on every restart.

If model database is corrupt it is going to affect instance functionality.

Steps:

- 1) Verify if Model is corrupt or not in Event viewer and SQL Server Error Logs.
- 2) Confirm if a valid database backup exists or not using restore verify only/header only.
- 3) Start instance with Master database only by enabling the trace 3608.

Net start "SQL Server (MSSQLSERVER)" /t3608

- 4) Restore the Model database from backup.

Restore database model from disk=N'F:\Model.bak' WITH REPLACE

- 5) Start instance normally by removing trace 3608

Net stop "SQL Server (MSSQLSERVER)"

Net start "SQL Server (MSSQLSERVER)"

OTHER METHOD: Copy and paste model .mdf, .ndf files from other instance [Required to take instance offline]

Note:

From SQL SERVER 2008 onwards MS introduced TEMPLATE FOLDER WHICH contains fresh "MASTER, MODEL and MSDB" MDF and LDF files.

If any corruption for system database, then please use this template to bring the instance quickly and do the restore to get updated data.

MSDB CORRUPT:

1) Verify the reason of failure in the error logs and troubleshoot accordingly. If database is really corrupt then look out for an available valid backup. If backup is available restore MSDB as a normal user database and it would be restored.

2) If backup is not available, then stop the instance and start the instance in /m and /t3608 startup parameters.

Net stop "SQL Server (MSSQLSERVER)"

Net start "SQL Server (instance name)" /t3608 /m

3) Connect to the Query window and detach MSDB database and delete the OS level files.

SP_detach_db 'MSDB'

NOTE: Remove MSDB data/log files from the path.

4) Execute the script in %Root Directory%\Install\instMSDB.sql file.

ISSUE:::: after msdb rebuild then Sql server agent not able to start? How to resolve

Solution:

sp_configure 'show advanced options', 1;

RECONFIGURE;

sp_configure 'Agent XPs', 0;

RECONFIGURE;

<http://techydave.blogspot.in/2013/02/sql-server-agent-agent-xps-disabled.html>

TEMPDB CORRUPTION:

1. If Tempdb corrupt instance wouldn't respond and it would be in hung state equal to crash.
2. To resolve, restart Sql server instance so that Tempdb files will be recreated.

Error number: 824 state: 19

Note:

1. If Tempdb log file is full then SQL Server instance did not allow any connections.
2. If model database is corrupted at the same time Tempdb corrupt, then even Tempdb did not create after restart.

Trace Flags:

-t: refereing startup

-d: fully qualified path of data file

-e: error file

-l: log file

-c: quick start of instance than regular process

-m: single user mode

-s: sqlserver.exe

-n: start instance name then the information will not trace in event viewer

Trace Flag 3607:

Starts SQL Server without recovering any databases

Trace Flag 3608:

Starts SQL Server, recovering master only

User Database Status -32768:

Starts SQL Server without recovering the user database.

FILE MOVEMENTS

File movement activities: [.MDF, .LDF & .NDF]

1. File move will applicable for both user and system databases.

USER DATABASE STEPS:

Pre-Activity: Take full database backup before performing any changes on your SQL Server instance.

1. Collect logical file names for your databases [User and system]

sp_helpfile

2. Update the logical name for user database by using alter command with new path.

Alter database dbname modify file (name='logical file name', filename='newpath') -----Data file

Alter database dbname modify file (name='logical file name', filename='newpath') -----log file

Note: Verify whether logical name is updated with new path by using sp_helpfile again...

3. Take database offline. [Kill the session if any active sessions are running on the database before taking your user database offline]

4. Move physical files to new path and bring the user database online.

Note: While moving physical .mdf, .ldf, .ndf files use always copy & paste method but not cut and paste.

5. Sp_helpfile

Note: When performing user database file movement only required to take database offline but not SQL Server services offline.

Note: Ask application team to check the connectivity of the database and once received confirmation from application team then only delete the older files from physical location.

q

Note: After completing the file movement go and clear the older files from directory by using SHIFT+DELETE but not delete ...since if you use delete next go to recycle bin folder if space is not there in C:\ drive then server becomes hung state. More risk is involve

To see list of file for all databases:

Select * from sys.sysaltfiles

SYSTEM DATABASES FILE MOVEMENTS

System Database File Movements:

Note: for system database you have to take complete Sql server instance restart [instance level downtime is required]

TEMPDB FILE MOVEMENTS:

1. Collect logical file names for Tempdb
2. Update logical name for Tempdb database by using alter command with new path

```
ALTER DATABASE Tempdb modify file (name='tempdev', filename='new physical path\tempdb.mdf')
```

```
ALTER DATABASE Tempdb modify file (name='templog', filename='new physical path\templog.ldf')
```

3. Take your SQL Server instance offline.
4. Start your SQL server instance.\restart Sql instance...

Stop\Start instance from CMD:

Default Instance:

```
Net stop "SQL Server (mssqlserver)"
```

```
Net Start "SQL Server (mssqlserver)"
```

Named Instance:

```
Net stop "SQL Server (Named instance Name)"
```

```
Net Start "SQL Server (Named instance name)"
```

5. Tempdb new files [.mdf, .ldf] will automatically create in new path.
6. NOTE: After verification if your instance ok then remove your old Tempdb files.

Note: For system databases you cannot perform offline, attach, detach, drop, delete

MASTER DATABASE FILE MOVEMENTS:

Steps:

2. Go to configuration manager-->advanced tab-->startup parameters--> specify new path for your .mdf and .ldf files
3. Take SQL Server instance stop state.
4. Copy physical .mdf and .ldf files to new location or new path.
5. Start your SQL Services.

Note: Instance level down time is required.

MODEL DATABASE FILE MOVEMENTS:

1. Collect logical files name for your MODEL database.
2. Change logical name for Model database by using alter command.

Alter database model modify file (name='modeldev', filename='D:\DATA\model.mdf')

Alter database model modify file (name='modellog', filename='D:\DATA\modellog.ldf')

3. Stop Sql server instance services
 4. Move physical mdf and ldf files for model database.
 5. Execute sp_helpfile for verification...
-

MSDB DATABASE FILE MOVEMENTS:

1. Collect logical files name for your MODEL database.
2. Change logical name for Model database by using alter command.

Alter database msdb modify file (name='MSDBData', filename='D:\DATA\MSDBData.mdf')

Alter database msdb modify file (name='MSDBLog', filename='D:\DATA\MSDBLog.ldf')

3. Stop Sql server instance services
4. Move physical mdf and ldf files for model database.
5. Execute sp_helpfile for verification...

IMPORT & EXPORT

By using this we can take table level backup.

> The main purpose is to load your data from SQL Server to any third party files like notepad, excel, any other RDBMS (Oracle. etc.) by using export technique. (Or)

> If your data is in third party files like notepad, excel, any other RDBMS (Oracle.etc) to SQL Server database by using import technique.

Import & Export Links:

<http://searchsqlserver.techtarget.com/feature/The-SQL-Server-Import-and-Export-Wizard-how-to-guide>

<http://sqlage.blogspot.in/2014/03/how-to-use-importexport-wizard-in-sql.html>

> The SQL Server Import and Export Wizard is based in SQL Server Integration Services (SSIS). You can use SSIS to build extraction, transformation and load (ETL) packages and to quickly create packages for moving data between Microsoft Excel worksheets and SQL Server databases.

> Launch SQL Server Import and Export Wizard by one of the following methods:

Method 1: On the Start menu, roll the cursor over All Programs, scroll down to Microsoft SQL Server and then click Import and Export Data.

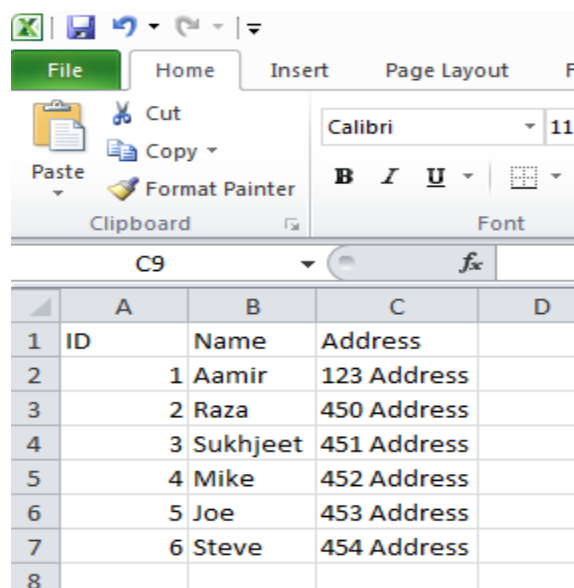
Method 2: In SQL Server Data Tools (SSDT), right-click the SSIS Packages folder and then click SSIS Import and Export Wizard.

Method 3: In SQL Server Data Tools, go to the Project menu and click SSIS Import and Export Wizard.

Method 4: In SQL Server 2014 Management Studio, connect to the Database Engine server type, expand Databases, right-click a database, point to Tasks and then click Import Data or Export data.

How to Use Import/Export Wizard in SQL Server

We have received SourceFile.xlsx file and we have to load that to SQL server Table. We can either create SSIS Package in BIDS or we can use Import/Export Wizard to load this file in SQL Server Table. In this post, we will use Import/Export Wizard.



	A	B	C	D
1	ID	Name	Address	
2	1	Aamir	123 Address	
3	2	Raza	450 Address	
4	3	Sukhjeet	451 Address	
5	4	Mike	452 Address	
6	5	Joe	453 Address	
7	6	Steve	454 Address	
8				

Fig 1: Excel Source File

Step 1:

Right Click on Database in which your table exists or you want to create it and load Excel data as shown below

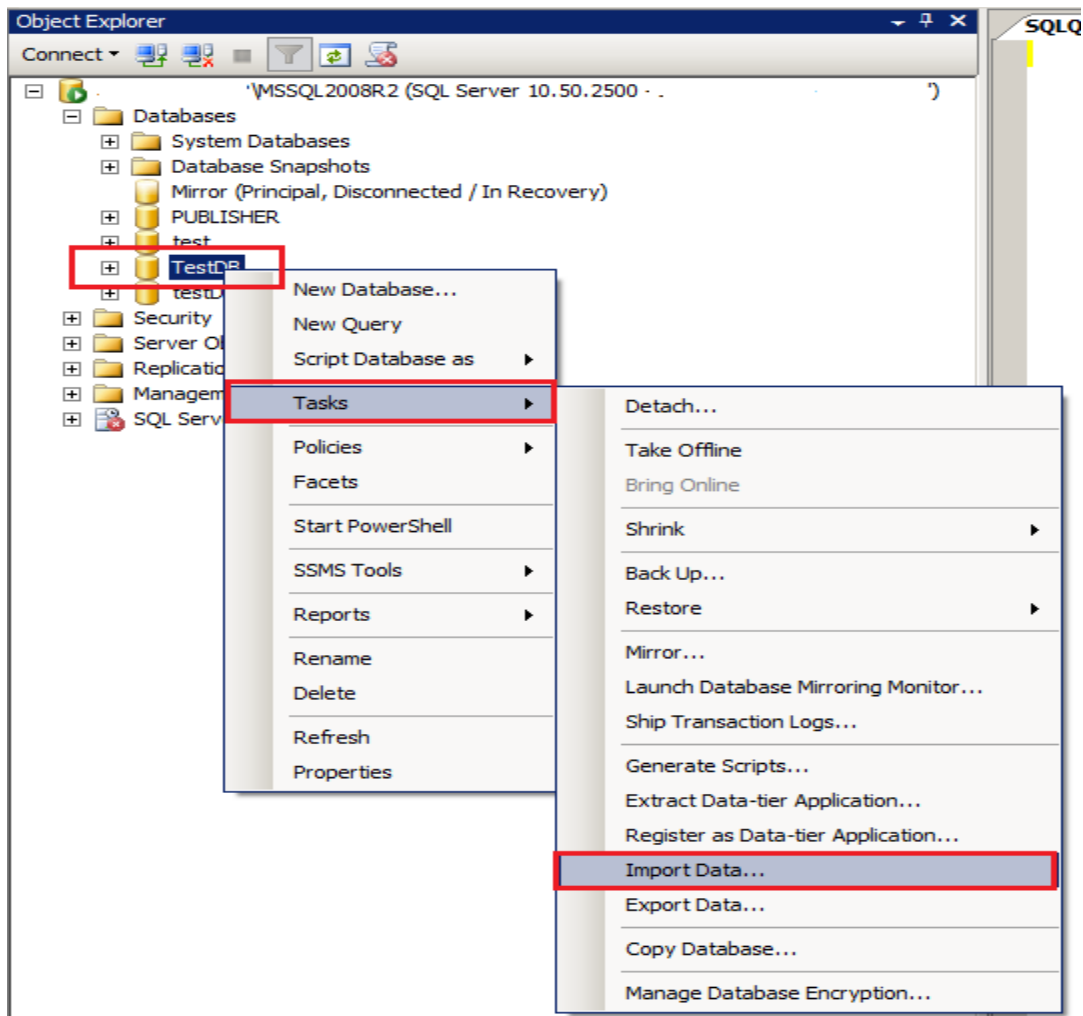


Fig 2: Import Data by using Import/Export Wizard in SQL Server Table.

Choose the Data Source:

Choose the data source which you want to use as source, as we are loading data from Excel, Choose Excel file as shown below

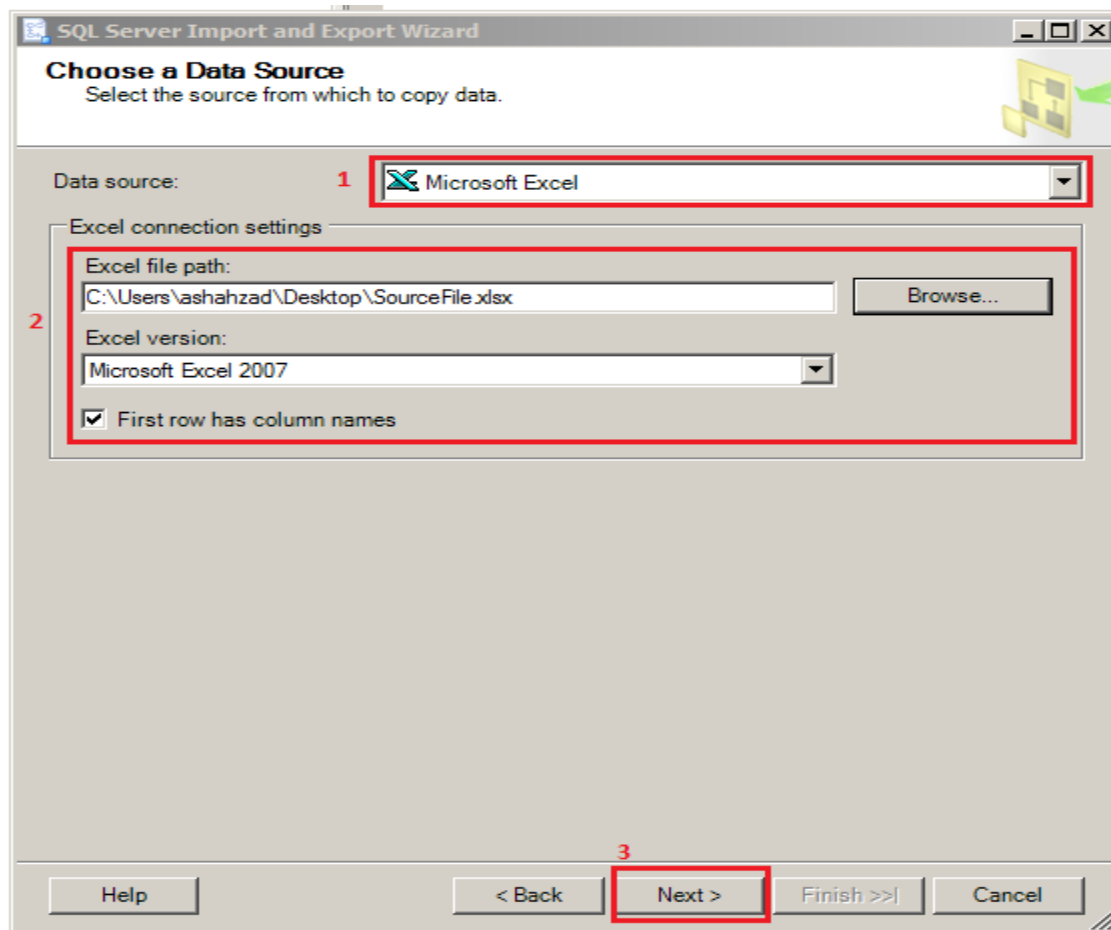


Fig 3: Choose Excel Data Source in Import Export Wizard

Choose a Destination:

Choose the destination where you want to load the data from source. In our case we are loading our data to SQL Server Table. Configure as shown below

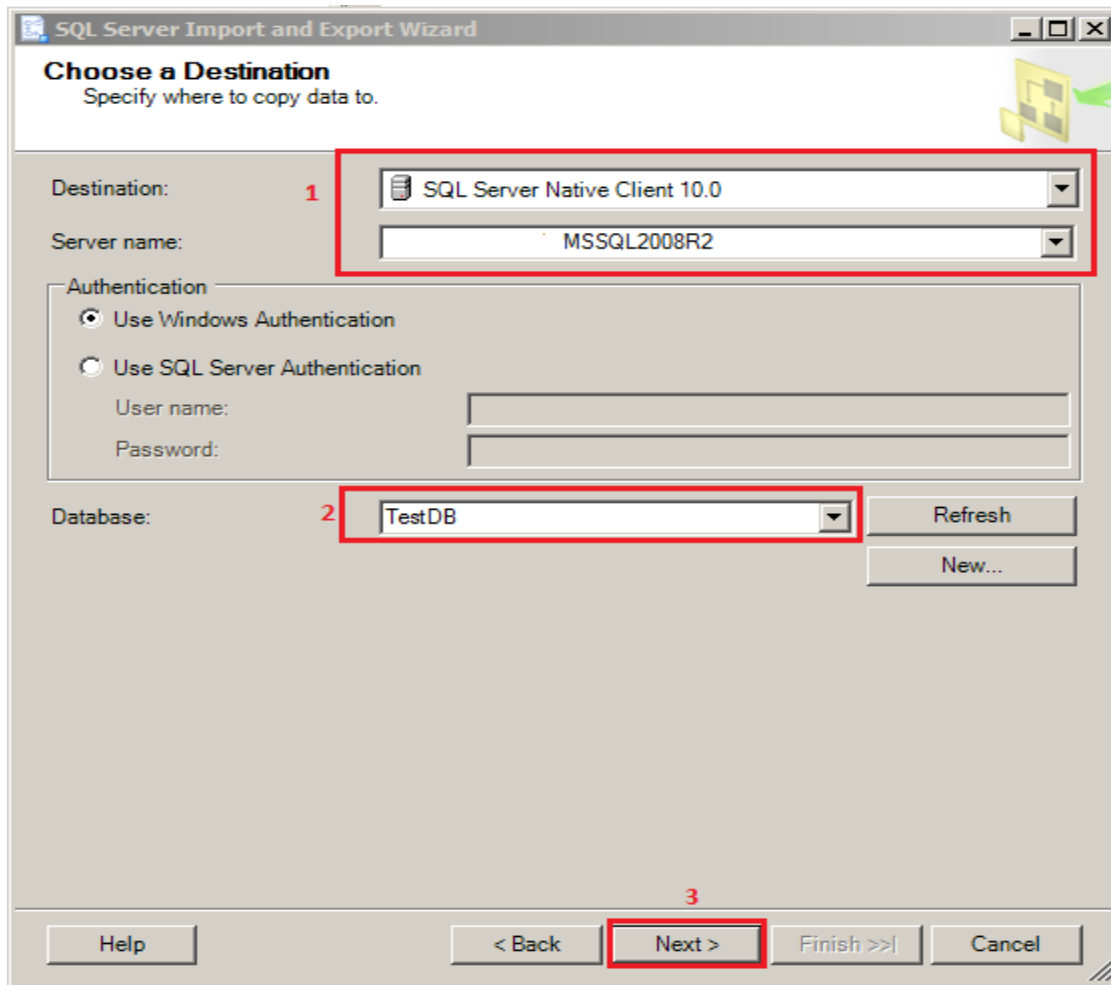


Fig 4: Choose SQL Server as Destination

Specify a Table Copy or Query:

You can directly choose the table from where do you want to load the data or you can write query if you are using Database as your source. As we are using Excel as source, we will choose Table (Sheet).

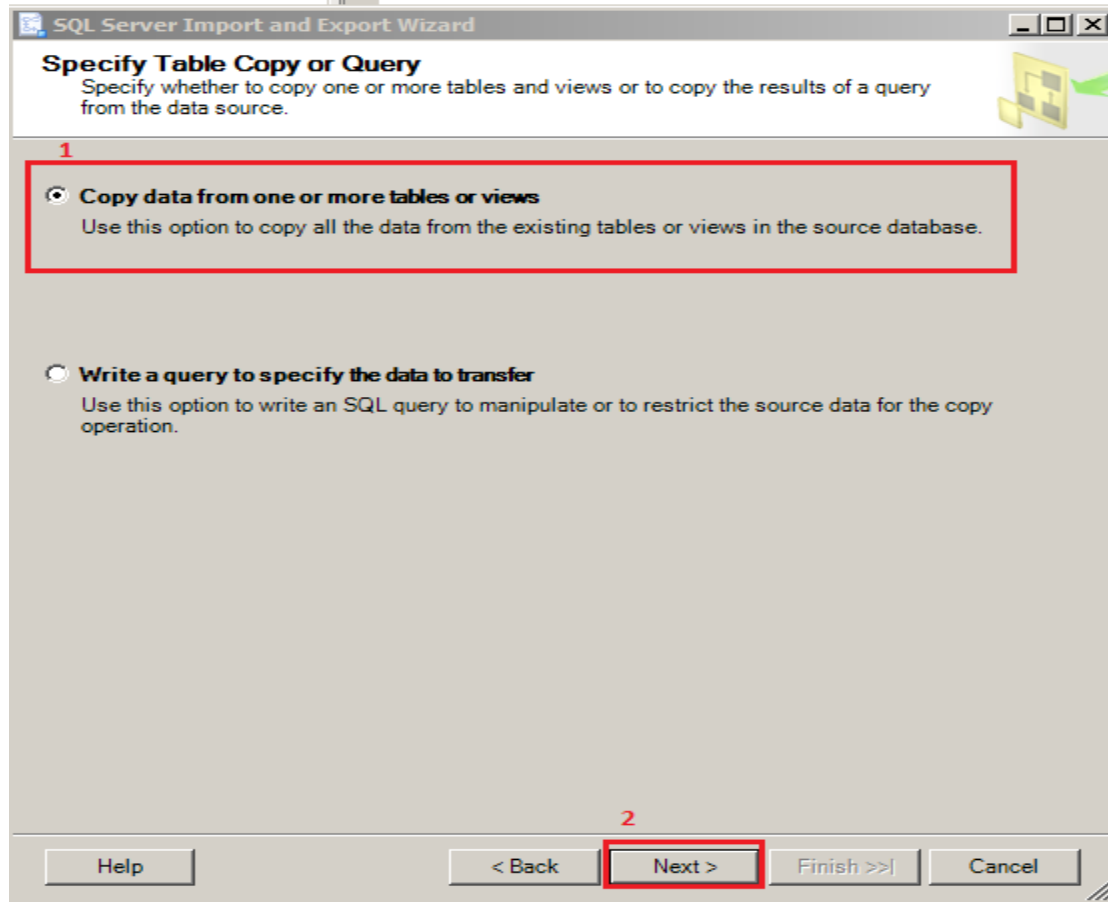


Fig 5: Choose Copy data from one or more tables or views

Select Source Tables and Views:

In this part of Wizard, we have to select the Tables or Views we want to use from source and load data to destination. As we are loading data from Excel, the Excel Tabs are shown. Choose the Sheet (Tab) which do you want to load. Under Destination, it will show you same name like Source. I have changed that to Customer Data. You can choose any name of your Table you want. You can choose multiple sheets or Tables from Source.

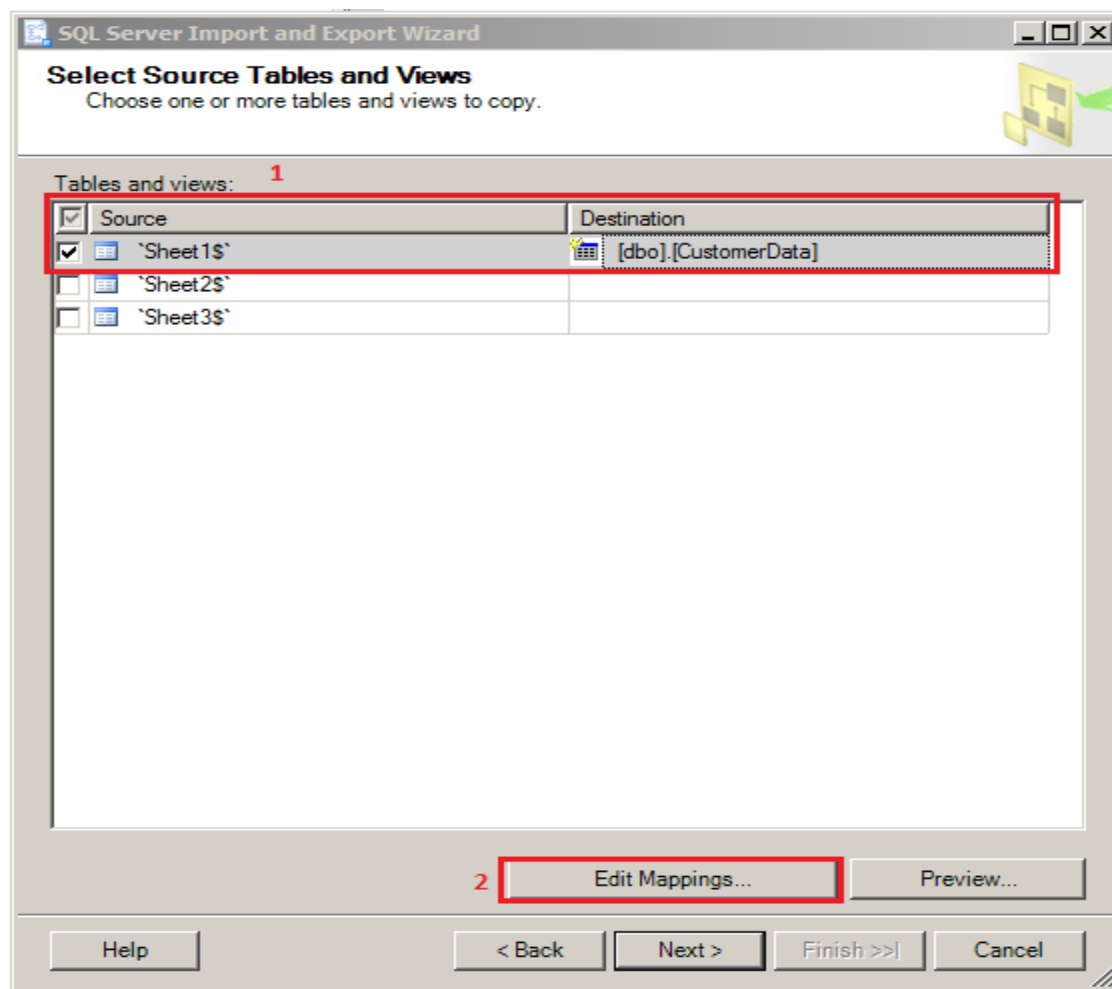


Fig 6: Select Source Tables/Views in Import Export Wizard

Column Mappings:

Click on Edit Mappings and then you can map the source columns to destination columns, also if you need to choose correct Data type, you can change here.

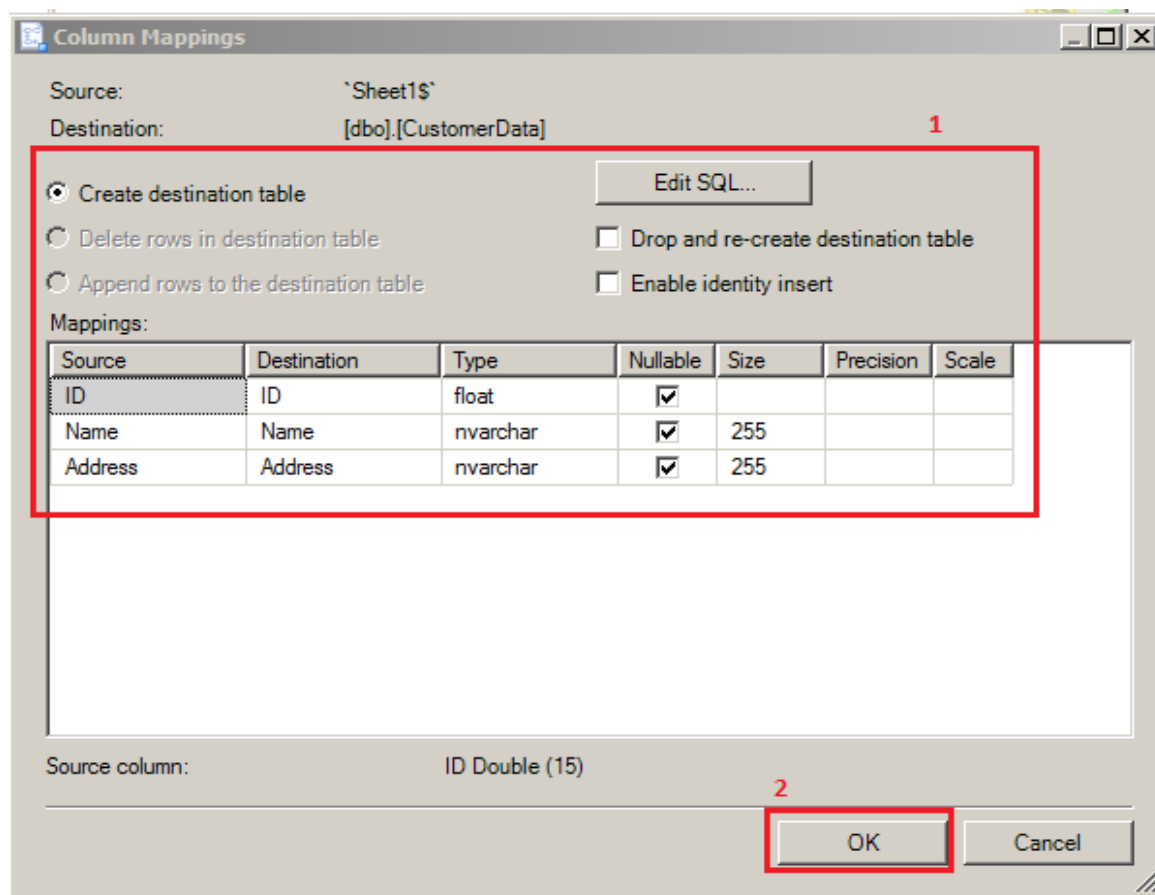


Fig 7: Column Mapping Import Export Wizard

Save and Run Package:

By Default, Run immediately is checked. I have changed the option to Save SSIS Package and provided the location where I want to save the SSIS Package. Also there is no sensitive information that I want to save in Package such as Password so I have selected Do not save sensitive data.

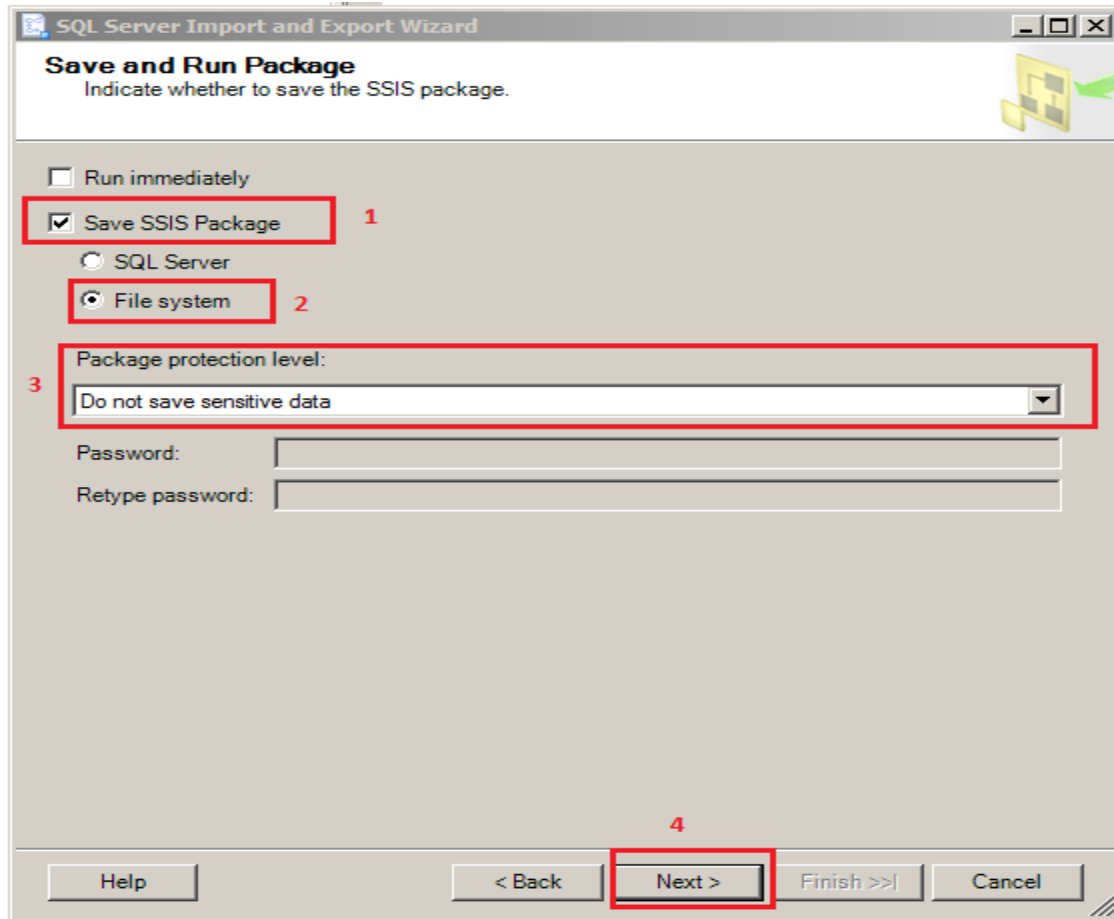


Fig 8: Save SSIS Package to File System

Save SSIS Package:

Provide the name of SSIS Package and File Location as shown below

SQL Server Import and Export Wizard

Save SSIS Package
You can save the SSIS package for reuse. You must save the package to schedule it for later execution.

Name: LoadExcelToSQLTable

Description: SSIS Package Description

Target: File System

File name: C:\Users\ashahzad\Desktop\LoadExcelToSQLTable.dtsx Browse...

Help < Back **Next >** Finish >> Cancel

Fig 10: Provide Name for SSIS Package

Complete the Wizard:

Summary of all the steps will be shown to you in this step. You can see the source and destination etc.

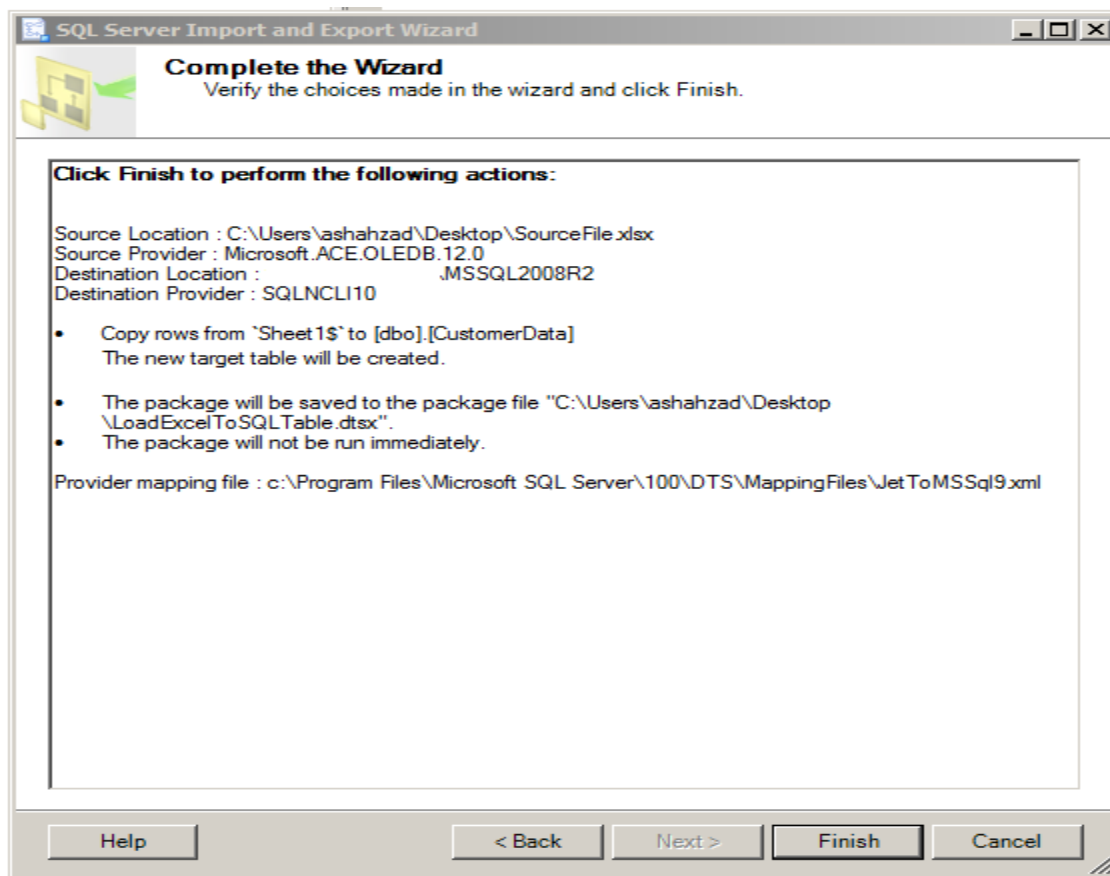


Fig 11: Summary of Steps

Once you hit Finish button, The Wizard will execute all below steps and finally save the SSIS Package.

q

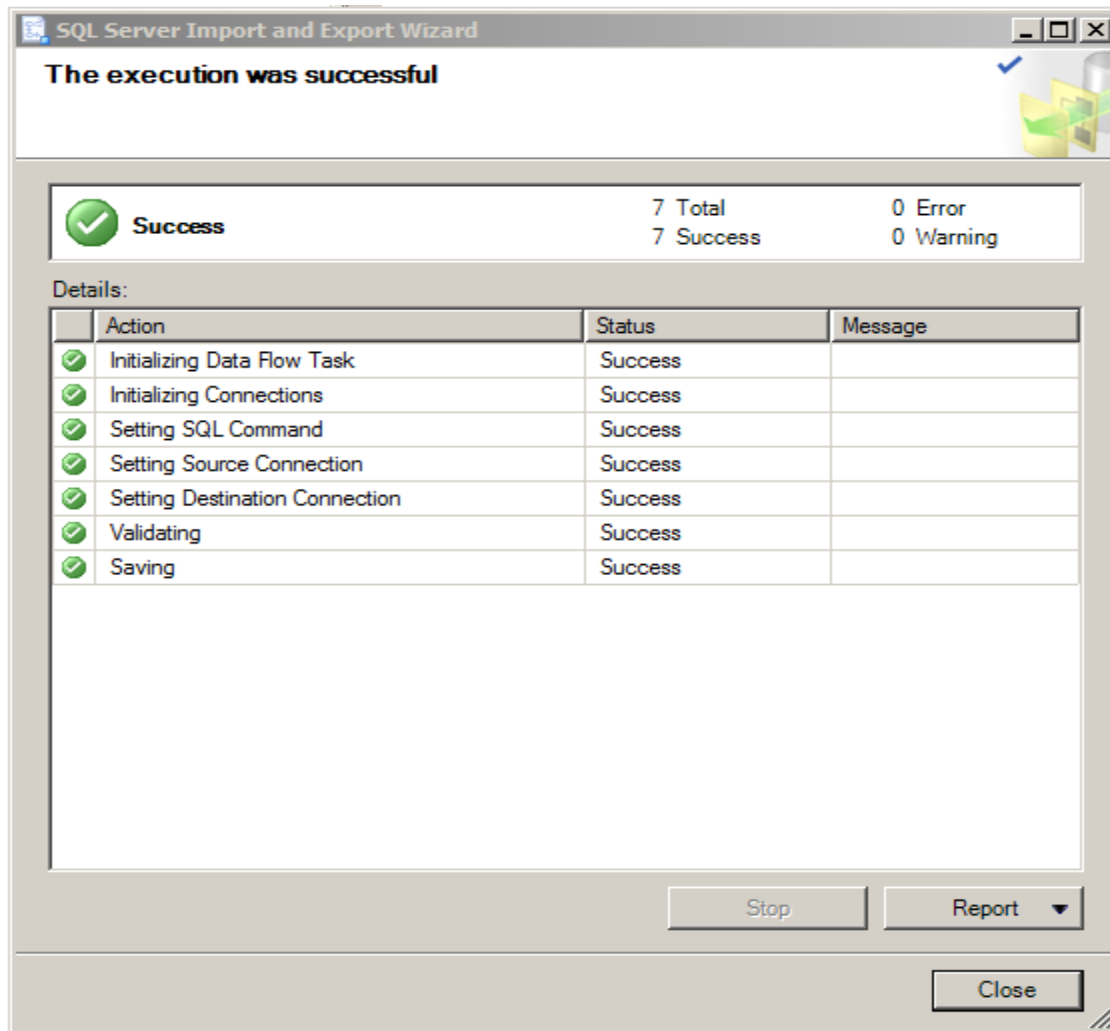


Fig 12: Save the SSIS Package to given location

The Package is created on desktop as per my given path.



Fig 13: SSIS Package created by Import/Export Wizard

To execute this package, double click on it and below window will open. If you need to change the name of File or SQL Server, you can go to Connection Managers and change it. In my case, I do not want to make any changes. Press Execute Button

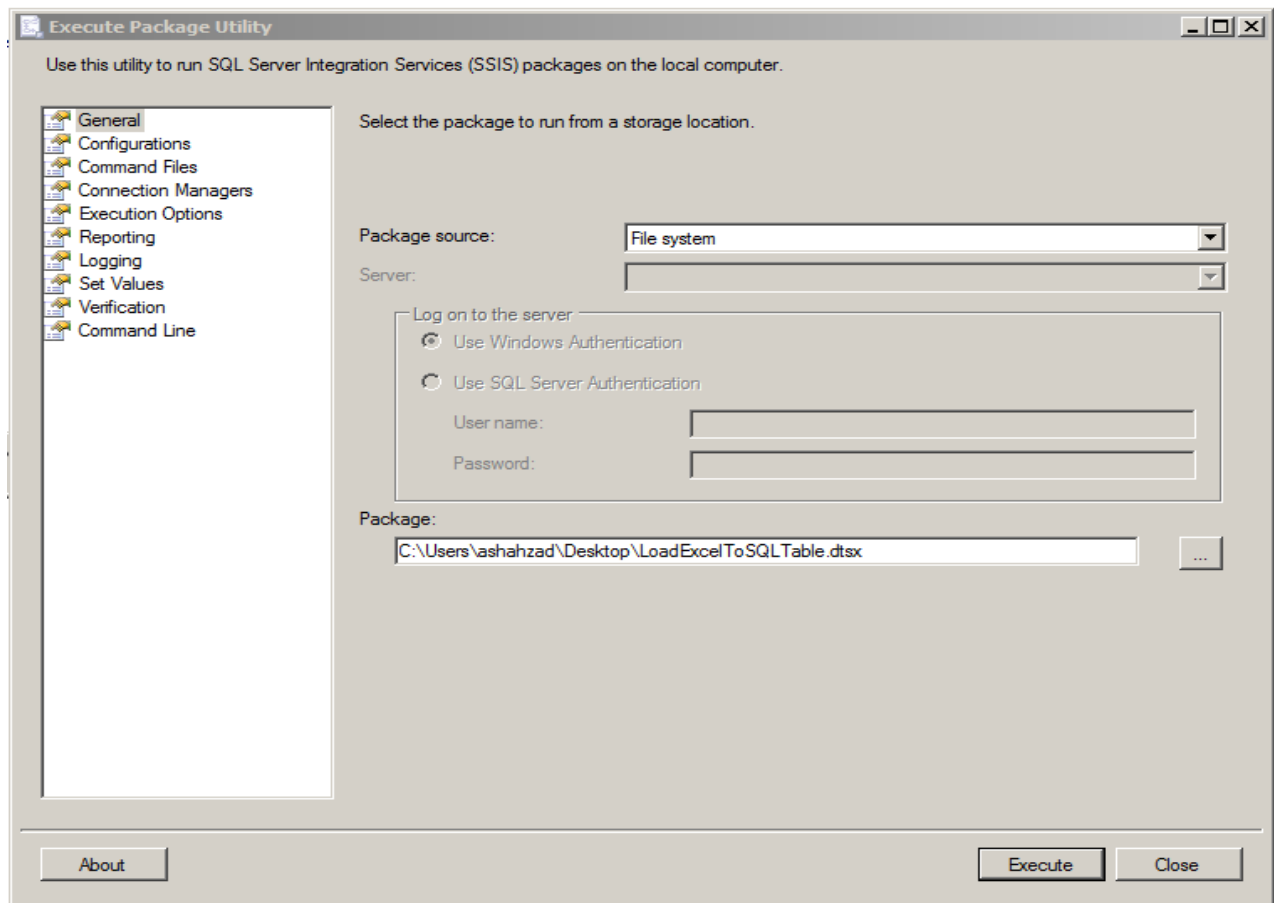


Fig: 14 Execute Package Utility

Once you hit Execute, Package Execute Progress window will appear and you will be able to see the progress of execution of your SSIS Package.

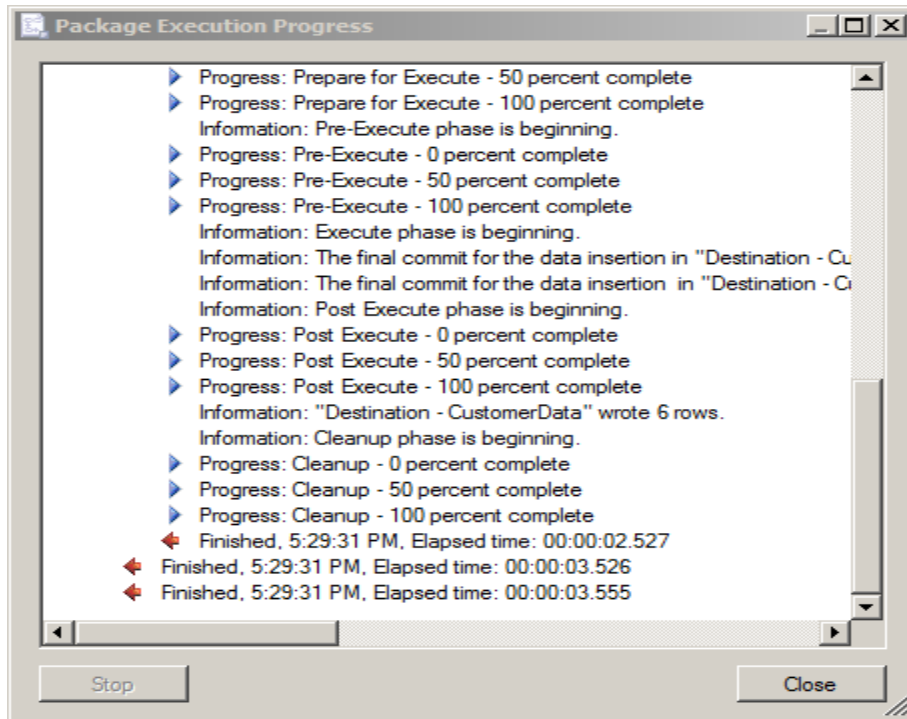


Fig 15: Package Execution Progress.

Import/Export Wizard is a way to quickly load data between different sources and destinations. You can create your SSIS Package quickly by using Import/Export Wizard and then add to SSIS Project and make changes if required.

If we need to export data from SQL Server then we need to Right Click on Database-->Tasks-->Export Data and Import/Export Wizard will start

DATABASE MAIL CONFIGURATION

INTRODUCTION

> This is an enterprise solution for sending mails from the SQL Server database engine to SMTP servers. SQL Server database applications can communicate with users through an email system. It provides features like scalability, security, and reliability.

> It uses an SMTP server to send mail. SQL Server 2000 supports SQL Mail, which supports MAPI profiles to send email instead of an SMTP server. SQL Mail requires a MAPI-compliant mail server (Microsoft Exchange Server) and a MAPI client (Microsoft Outlook).

> We can send a text message, query result, file as attachment. The database mail can be used to notify users or administrators regarding events raised in SQL Server. For example, if an automation process like replication, database mirroring fails or there are latency related problems then SQL Server can use this feature to notify the administrators or operators.

Points to Remember

- Like SQL Mail, database mail doesn't require a MAPI – a compliant mail server like Outlook Express or extended programming interface.
- Better performance. Impact of sending mails to SMTP servers by SQL Server is reduced as this task is implemented by an external process initiated by the *DatabaseMail.exe* file.
- Works fine in a cluster based environment.
- 64-bit support.
- Database mail configuration information is maintained in an **MSDB** database.
- Only members of **Sysadmin** and **DatabaseMailUserRole** database role of MSDB can send mails by default.
- Allows sending messages in different formats like text and HTML.
- Supports logging and auditing features through different system tables of MSDB.

The main components of database mail are:

- **Sp_send_dbmail**

This is a system defined stored procedure which is used by SQL Server to send email using the database mail feature. This stored procedure is present in the MSDB database.

- **MSDB Database**

Consists of all stored procedures, system tables, and database roles related to database mail.

- **Service Broker**

To establish communication between the SQL Server engine and the database mail engine we need a service broker. It submits the messages to the mail engine.

- DatabaseMail.exe

This file is present in the **Binn** folder of the respective instance. It is the database mail engine.

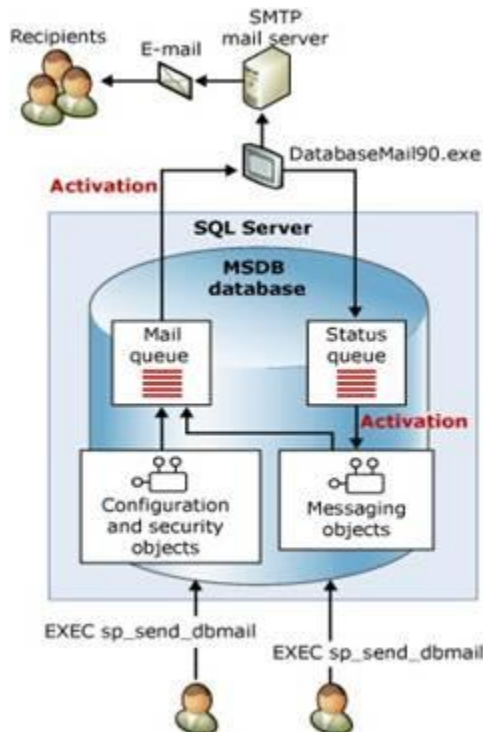


Figure – 1 (Source: BOL) Database Mail Architecture

How it works?

When a run time error occurs due to any automated task like backups, replication etc database engine raise the error and same information is submitted to Database Mail engine, then database mail engine will submit the mail to SMTP Server using EmailID and Password mentioned in profile. At the last SMTP Server sends mail to recipients.

Error --> DB Engine --> DB Mail Engine --> SMTP Server --> Recipients

FAQ: How to enable a Service Broker in MSDB?

```
USE [master]
GO
ALTER DATABASE [MSDB] SET ENABLE_BROKER WITH NO_WAIT
GO
```

MSDB tables related to Database Mail

1. **sysmail_profile**: Consists of all the profiles information.
2. **sysmail_account**: Consists of SMTP server accounts information.
3. **Sysmail_server**: Consists of SMTP server details.
4. **Sysmail_allitems**: Mail sent status. If the sent_status is 1 then success, otherwise failed.
5. **Sysmail_log**: To check the errors raised by Database Mail feature.
6. **Sysmail_configuration**: Consists of system parameter details.

Steps to configure

1. Enable the db mail feature at server level

```
sp_configure 'Database Mail XPs', 1  
Reconfigure
```

2. Enable service broker in the MSDB database.

```
USE [master]  
GO  
ALTER DATABASE [MSDB] SET ENABLE_BROKER WITH NO_WAIT  
GO
```

3. Configure mail profile (profile is a collection of accounts).
4. Add SMTP account(s).
5. Make the profile private or public.

Private profile can be used by:

- **Sysadmin** members and
 - **DatabaseMailUserRole** members of MSDB
6. Set parameters.
 7. Send the mail.

Difference between Database Mail and SQL Mail:

- 1) Database mail is newly introduced concept in SQL Server 2005 and it is replacement of SQLMail.
- 2) Database Mail is based on SMTP (Simple Mail Transfer Protocol) and also very fast and reliable whereas SQLMail is based on MAPI (Messaging Application Programming Interface).
- 3) SQL Mail needs an email client (like Outlook/Express) to send and receive emails, whereas Database mail works without any Email client.
- 4) SQL Mail works with only 32-bit version of SQL Server, whereas Database Mail works with both 32-bit and 64-bit.

1. blog.sqlauthority.com/2008/08/23/sql-server-2008-configure-database-mail-send-email-from-sql-database/

2. www.codeproject.com/Articles/485124/Configuring-Database-Mail-in-SQL-Server

LITE SPEED

Lite Speed for SQL Server

- Lite Speed for SQL Server is a revolutionary, patented development in database backup technology, encompassing the latest encryption and compression algorithms to deliver a complete solution for your archiving needs.
- Lite Speed maximizes disk space and process efficiency while greatly reducing the overhead costs associated with maintaining a state-of-the-art database facility.

Key Benefits:

- Reduces storage requirements (up to 95% compression)
- Reduces backup times (up to 75% faster than native SQL Server)
- Reduces restore times
- Reduces network load
- Integrates fully into SQL Server
- The ability to create your backups with varying types of industry-standard encryption
- The ability to do object-level restores (i.e., tables, views, and stored procedures; this feature is available only in the enterprise version)
- Mirroring of your backup files to multiple locations
- An enterprise console allowing you to control the backup and restores of all your MS SQL Servers in one location
- Integrated log shipping

Lite Speed Advantages:

- Lite Speed provides consistent compression and encryption for SQL Server and Oracle, along with efficiency

Compression:

- Lite Speed uses the same compression engine on both the Oracle and SQL Server platform, so the compression ratio for like data is reliably similar on both databases.

Encryption:

- Lite Speed provides encryption for both SQL Server and Oracle.

Efficiency:

- Lite Speed performs backup compression on the database server, in memory, before the backup is written to disk or shipped across the network to a tape system. All other methods require more storage and/or greater network utilization because the compression occurs after the backup is created.
- <http://www.techrepublic.com/blog/howdoi/how-do-i-install-configure-and-use-litespeed-for-database-backups/167>

Convert SQL Lite Speed Backup to Native Backup:

- <http://kotako.wordpress.com/2010/03/02/convert-litespeed-backups-to-sql-server-backups-and-restore-them/>
- <http://easymssql.blogspot.co.uk/2010/01/need-to-convert-sql-lite-speed-backup.html>
- <https://support.quest.com/SolutionDetail.aspx?id=SOL22045>

Lite speed backup file format: .BKP

Lite speed backup query:

```
Exec master.dbo.xp_backup_database @database = N'dbname', @filename = 'D:\test.litespeed.f0.bkp'
```

Lite speed restore query:

```
Exec master.dbo.xp_restore_database @database = N'dbname', @filename = 'D:\test.litespeed.f0.bkp'
```

SHRINKING

- > To release the space to disk when any unused space hold by the database
- > The only way to release the space from database in Sql server is shrink.

Note: Shrink only when the database \file contains available free space. If no even you try to shrink no space released to disk.

Cases:

1. Log file shrink:

- > Can shrink log file any time based on the available free space in Sql server.

To shrink log file:

DBCC SHRINKFILE ('FILE NAME', SPACE TO KEEP in MB'S)

2. Data file shrink [NDF\MDF]:

- > Check whether any open transactions are running on database before shrink data file. If any then do not perform any shrink operation.

To shrink log file:

DBCC SHRINKFILE ('MDF\NDF file name', SPACE TO KEEP in MB'S)

Note before you shrink DATA FILE:

- > Request for down time for any application that is trying to access SQL Server on the server so that resource utilization on the server is very minimal.
- > Shrink max 5 or 10 GB from database at a time to release the space but not more than that. Causes performance impact.
- > Do not shrink the data file to its maximum capacity. Always leave minimum 10-20% free space on the data file.

Note: IN BUSINESS HOURS NEVER PERFORM SHRINK OPERATION ON MDF\NDF FILE.

To shrink database:-DBCC SHRINKDB ('DBNAME', SPACE TO KEEP in MB'S)

How applications generally connect to database?

1. When application try to connect, uses Config file (Configuration file--Resides in application server). Config file (.txt) should contains SQL Server instance name + user name [service account name--DBA team should add the account into SQL Server under security] + Password [Strong]

Application use all these details from Config file and point connection to SQL server database

Real time points:

- > Any business 2 data centers maintains called PRODUCTION and DR data center.
- > Always distance between data centers should not be more than 50 KM.
- > Few people always work inside the data centers.

UPGRADATION & MIGRATION

UPGRADATION

Upgradation Steps:

> Upgradation involves overwriting existing SQL Server instance and upgrading from one version to another version.

> Applying a Patch, Upgrading to New Version and Upgrading to New Edition.

We can call it as In-place Upgradation.

Steps:

Pre-upgrade steps

Upgrade steps,

Post upgrade step

Pre-upgrade steps:

Study SQL Server minimum hardware & software requirements

Run Upgrade Advisor

Examine Upgrade Advisor report

Fix or work around the backward compatibility issues

Take database backup including system and user database completely

Use sp_configure to verify same configuration after upgrade Screen

Upgrade steps:

Run higher version Sql server setup.exe

Post upgrade steps:

1. Check all components are upgraded to higher version
2. Check services are running or not.
3. Check db, login, jobs and maintaince plans are still remain same in SQL Server after upgrade
4. Verify sp_configure to check configuration in SQL Server

Note: If any SSIS or DTS or cubes packages are there then BI team will take care...

Note: If instance is configured with high availability then break the HA and install higher version then try to reconfigure HA one more time.....

Reasons of Upgradation:

- 1) Upgrading between versions.
- 2) Upgrading with a Service Pack
- 3) Upgrading from one edition to another.

Upgrade Advisor Analysis:

Upgrade Advisor analyzes the following SQL Server components:

Database Engine

Analysis Services

Reporting Services

Integration Services

1. Issue can prevent from upgrading from lower version to higher version
2. Wizard finds any blockings to be fix before upgrading
3. Advisor do not modify any data on the server

Advisor report:

Advisor gives the report which contains issues found during analysis, to manage tasks associate list.

The analysis examines objects that can be accessed, such as scripts, stored procedures, triggers, and trace files.

Note: Upgrade Advisor cannot analyze desktop applications or encrypted stored procedures.

MIGRATION

Migration Steps:

- > It involves moving data or databases from one Instance to another Instance same in SQL Or other DBMS like oracle, Sybase etc.
- > Like OS Upgradation, Moving data from one drive to another drive, other DBMS etc.,
- > Down time is minimal and once tested new server is released.
- > We can call it as Side by side Upgradation.

<https://sqlschoolhouse.wordpress.com/category/sql-server-database-migration/>

Pre-Migration steps:

- 1) Run Upgrade Advisor to find faults before starting the migration/Upgradation.
- 2) Disable all HA options and ensure all jobs are also disabled.

Migration steps:

- 3) Take backup of Source (2000/2005) and restore at destination (2005/2008) (or) follow Detach/Attach. ONLY user databases should be backed up.

Post migration steps:

- 4) Change compatibility level from 80 to 90 for all the database(s) that have been migrated

sp_dbcmtlevel

- 5) Transfer all the logins from Source to the Destination using

sp_help_revlogin

- 6) Fix orphan users issue if any.

sp_change_users_login 'update_one', 'user name', 'login name'

- 7) DBCC UPDATEUSAGE

> The table or index row counts and the page counts for data, leaf, and reserved pages could become out of synch with time in the database system tables. DBCC UPDATEUSAGE command corrects these inaccuracies and ensures the counts are updated.

DBCC UPDATEUSAGE ('database_name') WITH COUNT_ROWS

8) DBCC CHECKDB

> DBCC CHECKDB is a commonly used command that can check the allocation, structural and logical integrity of a database and its objects.

DBCC CHECKDB ('database_name') WITH ALL_ERRORMSG

CHECKDB found 0 allocation errors and 0 consistency errors in database 'Dbname'.

9) Update Statistics

> Statistics of SQL Server 2005 can be outdated and hence it is very important to update the statistics after moving the database to 2008.

> Update statistics updates the header information (Page), count information, histograms, and metadata allocations.

sp_updatestats

10) Changing Page Verify Option

Torn Page Detection: Torn Page detection is a method that calculates Bit values for every 512 byte sector and stores the final values in the page header. While retrieving the page it checks if the page is Corrupted or not.

Checksum: Checksum generates a checksum number and stores in the page header and when retrieving the page checks if the checksum is correct or not and hence this is how it validates the page is 'not corrupted'/'corrupted'.

Note: This doesn't perform any page verification

11) Transfer of Jobs (/Logins) can also be done using DTS Packages by DTS/SSIS team. If there are any DTS Packages in SQL Server 2000 they can be migrated to SSIS Packages in 2005.

If Jobs have to be transferred by DBA, then manually he/she has to script all the jobs and execute that commands at the destination.

12) Finally after migration is completed ask Application team to perform App checks and validate if SQL Server 2005 is compatible with the respective application or not.

Once final Go is given by App team that confirms that migration is a success.

Check list for before Migration?**Migration from SQL Server 2008 to SQL Server 2014 Check list**

1. Identify databases you would like to migrate
2. Backup all user databases
3. Script out all the existing login
4. Script out all the Server roles if applicable
5. Script out all the Audit and Audit Specifications if Applicable
6. Script out backup devices if Applicable
7. Script out Server level triggers if Applicable
8. Script out Replication along with Configuration if Applicable
9. Script out Mirroring if Applicable
10. Script out Data Collection if Applicable
11. Script out Resource Governor's objects if Applicable
12. Script out Linked Server if Applicable
13. Script out Logshipping if Applicable
14. Script out SQL Server Agent jobs
15. Script out all DB Mail objects such as Profile and its settings
16. Script out all Proxy accounts and credentials if Applicable
17. Script out all Operators if Applicable
18. Script out all alerts if Applicable
19. Save SQL Server, Server configuration in a file
20. Data Encryption keys

Destination SQL Server Checklist

1. Required SQL Server is installed
2. DBA SQL Server Check List is Completed
3. Enough Space for storing Backup and source scripts
4. Applications compatibility is signed off

TEMPDB FULL

Tempdb Full:

- 1) Increase the File Size (if storage is available).
- 2). Add NDF file from another/same drive.
- 3) Find out the transaction which is occupying more space in Tempdb and troubleshoot or kill that transaction based on approval
- 4) Shrink the Data File of Tempdb (if no OPEN transactions are in progress)

DBCC OPENTRAN \sys.dm_db_session_space_usage

- 5) DBCC FREEPROCCACHE

> Command will clear the procedure cache in memory.

> There is a risk involved with this, it can affect next procedures to be reparsed and compiled. Also it might not give accurate results for DMV's.

- 6) Restart the instance and it resets the tempdb size to last stated value in sysdatabases.

LOGFILE FULL

Log file full:

Error: 9002

Steps:

1. By using select * from sys.databases and verify the column "log_reuse_wait_desc" to find any active transactions are there?

DBCC SQLPERF (Logspace)

2. Check whether log file growth is set to restricted or unrestricted?

If set restricted then increase the size for log file growth.

3. Verify whether log backups are running or not if not manually run one T-log backup.
4. Try to perform SHRINK operation which can release the space.
5. Check the disk space availability where you kept log file
6. If disk space is not available then add the over_flow file into the disk where space is available.
7. Perform file movement after approval to the disk where it have enough free space.
8. Verify whether log backups are running or not if not manually run one T-log backup.

Last option:

TRUNCATE ONLY ;;;(Not recommended in production server)

Backup log dbname with truncate_only

Note: VLF'S deleted from log file and due to truncate operation. You may get the free space in log file but chances of data loss.

RESOURCE GOVERNOR (NEW FEATURE OF SQL SERVER 2008)

> Resource Governor is a new technology in SQL Server 2008 that enables you to manage SQL Server workload and resources by specifying limits on resource consumption by incoming requests.

> The following three concepts are fundamental to understanding and using Resource Governor.

Resource pools: Two resource pools (internal and default) are created when SQL Server 2008 is installed. Resource Governor also supports user-defined resource pools.

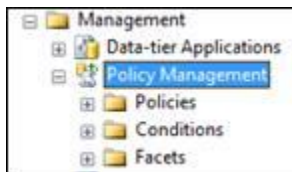
Workload Groups: Two workload groups (internal and default) are created and mapped to their corresponding resource pools when SQL Server 2008 is installed. Resource user-defined workload groups.

Classification: There are internal rules that classify incoming requests and route them to a workload group. Resource Governor also supports a classifier user-defined function for implementing classification rules.

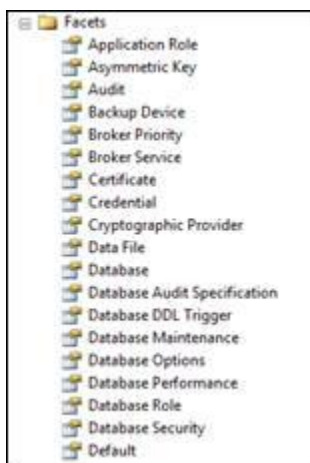
SQL SERVER SECURITY AND POLICY BASED MANAGEMENT

- The Policy Based Management feature was introduced in SQL Server 2008. The purpose of the feature is to assist SQL Server administrators in creating and enforcing policies tied to SQL Server instances and their objects. The policies can be configured on one SQL Server and re-used on other SQL Server instances to provide a SQL Server security model for instance.

- Policy Based Management allows DBAs to define the preferred state of the SQL Server system components (e.g. instances and objects) and compare the compliance status with the preferred state. Properly declared policies ensure enforcing company rules in the SQL Server environment, and are commonly a part of the SQL Server security model.
- The Policy Based Management feature is built on top of the SQL Server Management Objects collection (objects that are designed for programming all aspects of managing Microsoft SQL Server) which supports SQL Server 2000 and later versions. Therefore Policy Based Management can be utilized on versions prior to SQL Server 2008, for instance via the PowerShell subsystem and SQL Server Agent.
- Policy Management allows creating policies for various facets with a specified condition.



- Facets: Facets is the property of SQL Server which the policy will consider managing. There are several facets on which policies could be implemented. For example, we will use the “Database Option” facet to implement a policy which will ensure that the AutoShrink option should be TRUE for all hosted databases on the server. Similarly, we will be creating policies on the Stored Procedure facet.

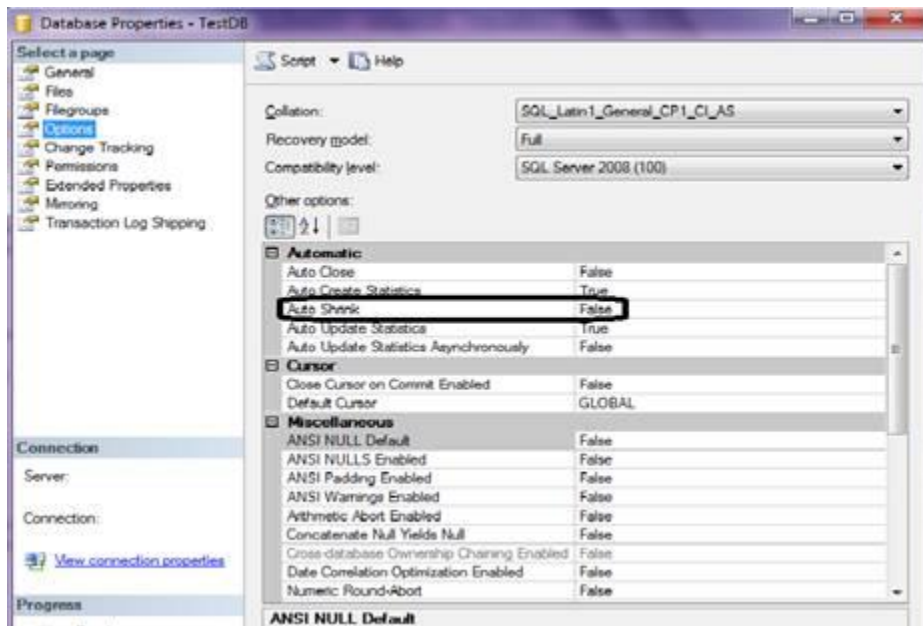


- Conditions: It is the criteria upon which the facet is evaluated. While designing a policy for the server, the first step is to create a condition which is to be evaluated by the policy for the facet.

- Policies: As the dictionary says, I reform, a SQL Server policy is a set of basic principles and associated guidelines, formulated and enforced by the Policy Manager of a server, for the desired server facets to conform with, which in the long run shall maintain the server consistent and help the DBA achieve organizational level IT norms.

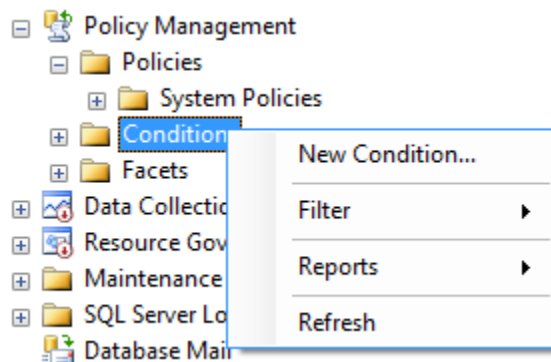
Example 1

Scenario: We will create an on demand policy to ensure that all the databases have the Auto Shrink option set to True. By default, a database that is created has Auto Shrink set to False, as shown in the figure below.



Step 1: Creating a Condition

Right click on Conditions and select New Condition...

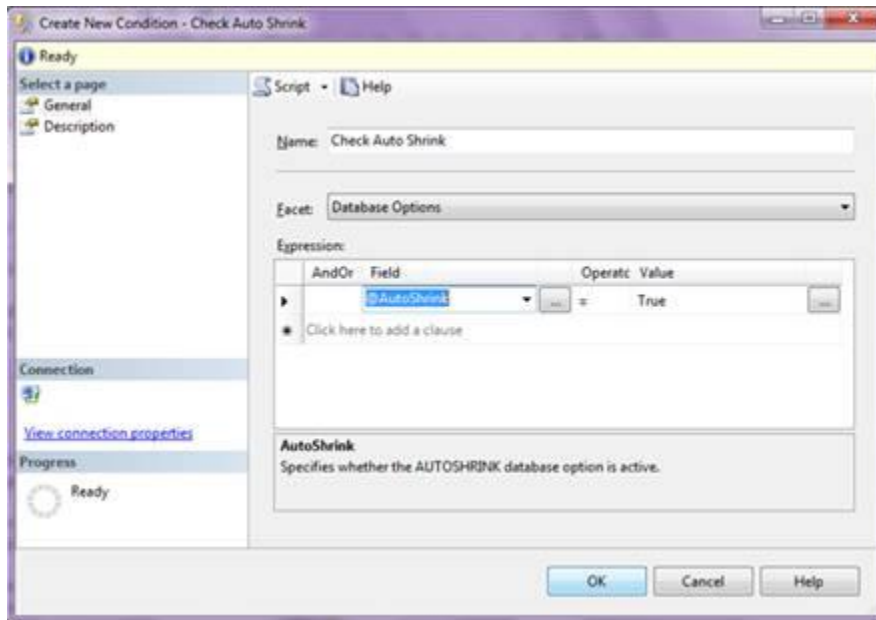


q

Next, provide a name to the Condition: “Check Auto Shrink”, and select the facet from the Facets drop down as “database option”. In the Expression Editor, choose the field from the drop down “@AutoShrink”, select operator as “=”, and value as “True”.

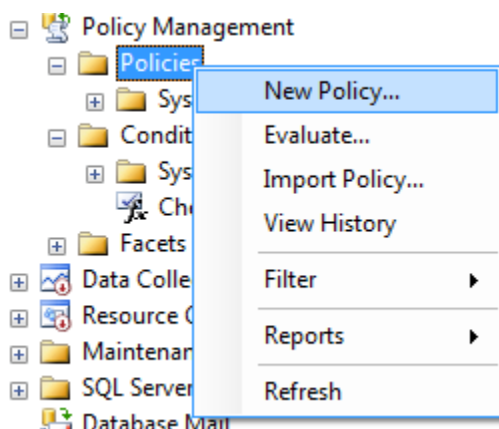
The condition will check all databases for their auto shrink properties to be true.

Click OK.



Step 2: Create a Policy

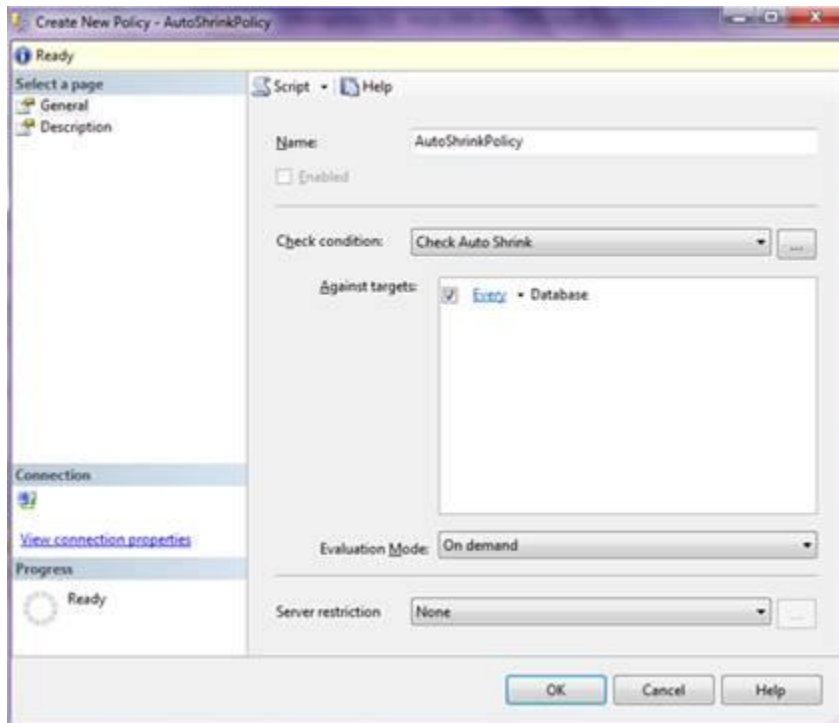
Right click on Policies and select New Policy...



Provide a name as “AutoShrinkPolicy”; from the Check condition drop down, select the Condition we just created. And from Targets, check every database as we want every database to conform to this policy.

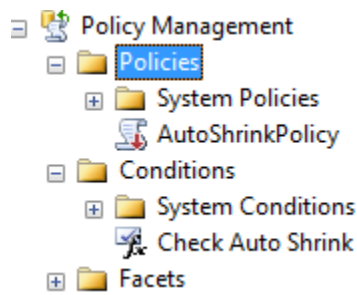
q

Next is the evaluation mode. Let's keep it "On demand" for this example. On demand means we will evaluate the policy at our will instead of at a predefined schedule.



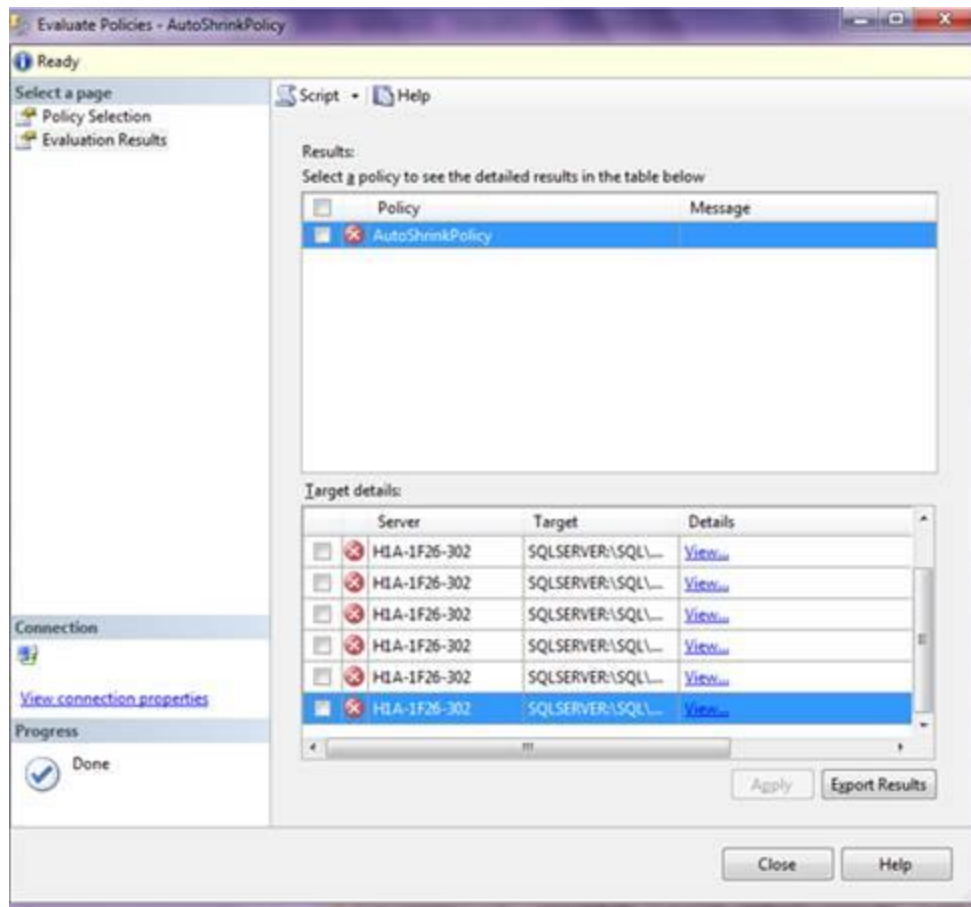
Click OK.

We are all set, the policy is in place.



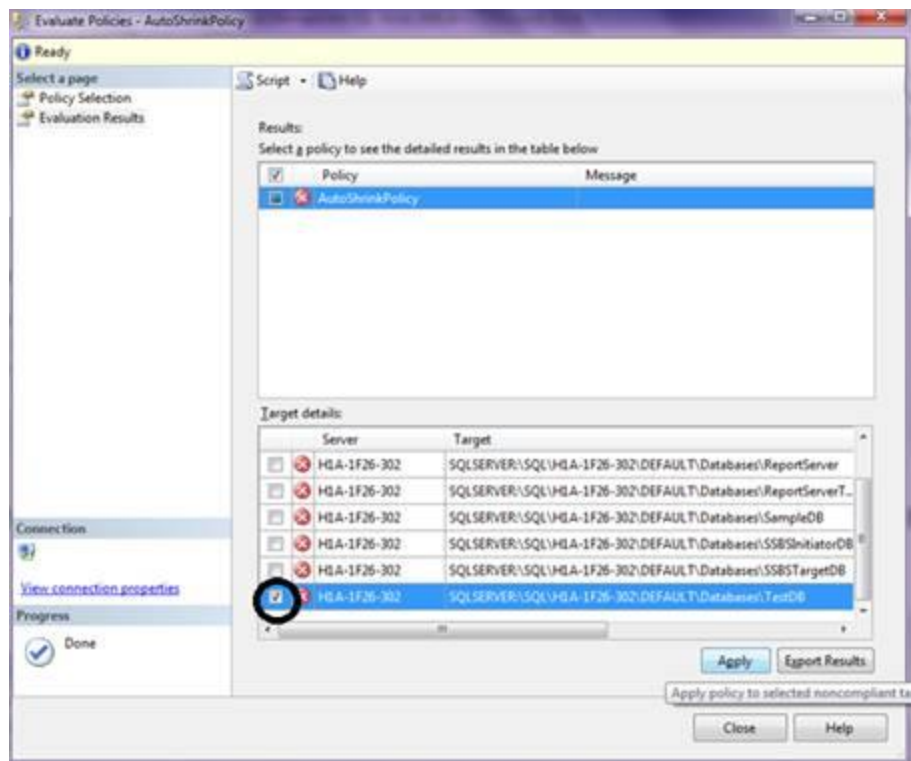
Step 3: Evaluation

We have been able to create the policy; now we will let the Policy Manager evaluate the policy. To evaluate, right click the Policy "AutoShrinkPolicy" and click Evaluate. SQL Server evaluates and lists the result as shown in the screenshot below. Since for none of my databases Auto Shrink is True, there are non-conformations for each one of the hosted databases on my server.



For conforming the results as per the Policy, check against the database and click on the Apply button.

q



This will set the Auto Shrink property for TestDB to True and a green sign will denote its conformance.

HIGH AVAILABILITY

HIGH AVIALABILITY

High Availability

- **High Availability:** SQL Server provides several options for creating high availability for a server or database.
- HA is to continue operations when a component fails. This is usually a hardware component like a CPU, Power supply, Disk failure, Memory failure or the complete server.
- With HA there is usually no loss of service when a component fails.

The high-availability options include the following:

- **LOG SHIPPING** ---- Database Level
- **DB MIRRORING** ---- Database Level
- **REPLICATION** ---- Table Level
- **CLUSTERING** ---- Instance level

LOG SHIPPING

What is Log Shipping?

Log Shipping Definition:

- Log Shipping is used to synchronize the Distributed Databases. Synchronize the database by copying Transaction logs, Backing up, Restoring data. SQL Server used SQL Server Job Agents for making those processes automatic.

Or

- It automatically sends transaction log backups from one database (Known as the primary database) to a database (Known as the Secondary database) on another server. An optional third server, known as the monitor server, records the history and status of backup and restore operations. The monitor server can raise alerts if these operations fail to occur as scheduled.

Or

- Shipping the transaction log backups from one server to another server called "Logshipping"

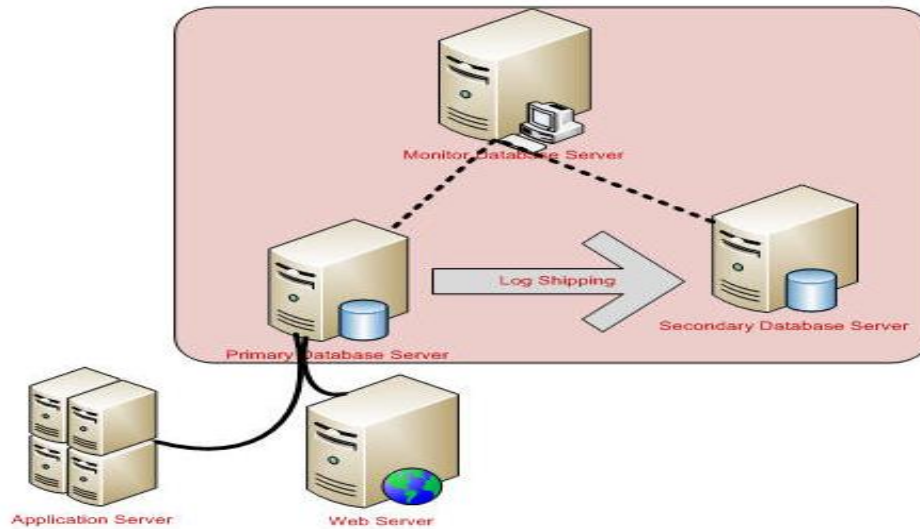
The main functions of Log Shipping are as follows:

- Backing up the transaction log of the primary database
- Copying the transaction log backup to each secondary server
- Restoring the transaction log backup on the secondary database

Log shipping pre-requisites:

1. Minimum 2 Sql server instances and 3 if we include monitor as well
2. Create database in primary server
3. Recovery model should be "FULL \BULK LOGGED"
4. SQL Server version should same and edition also same.
5. Create backup share in primary and provide read\write permissions
6. Create a copy share in secondary server and provide minimum read permissions.
7. SQL Service account should have the permissions on backup and copy share.
8. SQL Server services should run on domain level accounts.
9. Should be any edition for log shipping except express edition [No agent service]
10. SQL Server agent should be up and running fine.
11. Collation settings should be same.

Log shipping terminologies (or) Components:



- For implementing Log Shipping, we need the following components - Primary Database Server, Secondary Database Server, and Monitor Server.
- **Primary Database Server:** Primary Server is the Main Database Server or SQL Server Database Engine, which is being accessed by the application. Primary Server contains the **Primary Database**
- **Secondary Database Server:** Secondary Database Server is a SQL Server Database Engine or a different Server that contains the **backup of primary database**. We can have multiple secondary servers based on business requirements.

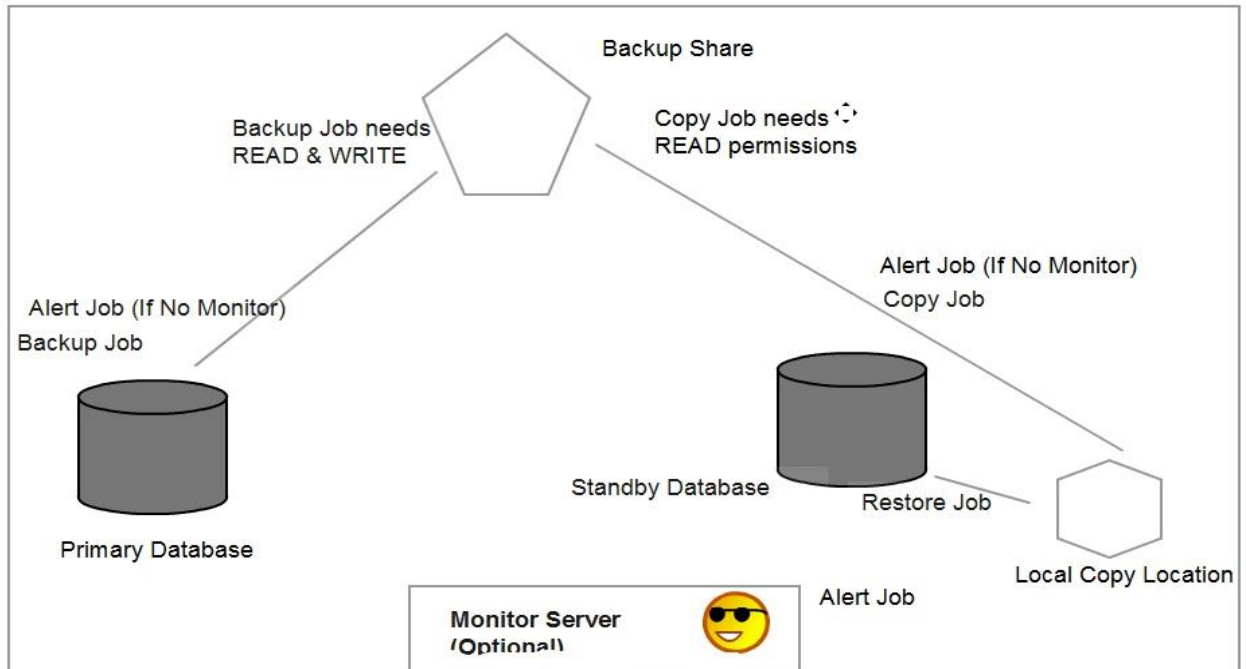
It is a copy of primary database and it is restoring\stand_by (read_only) mode.

Maximum we can add 25 Secondary Servers.

- **Monitor Server:** Monitor Server is a SQL Server Database Engine which **Track the Log Shipping process**.

Note: If monitor server is included in LS configuration then alert job gets created in monitor server itself if monitor server is not included then alert job created in both primary and secondary server.

Log shipping Architecture:



Backup job:

- A SQL Server Agent job that performs the backup operation.

Copy job:

- A SQL Server Agent job that copies the backup files from the primary server to the secondary server

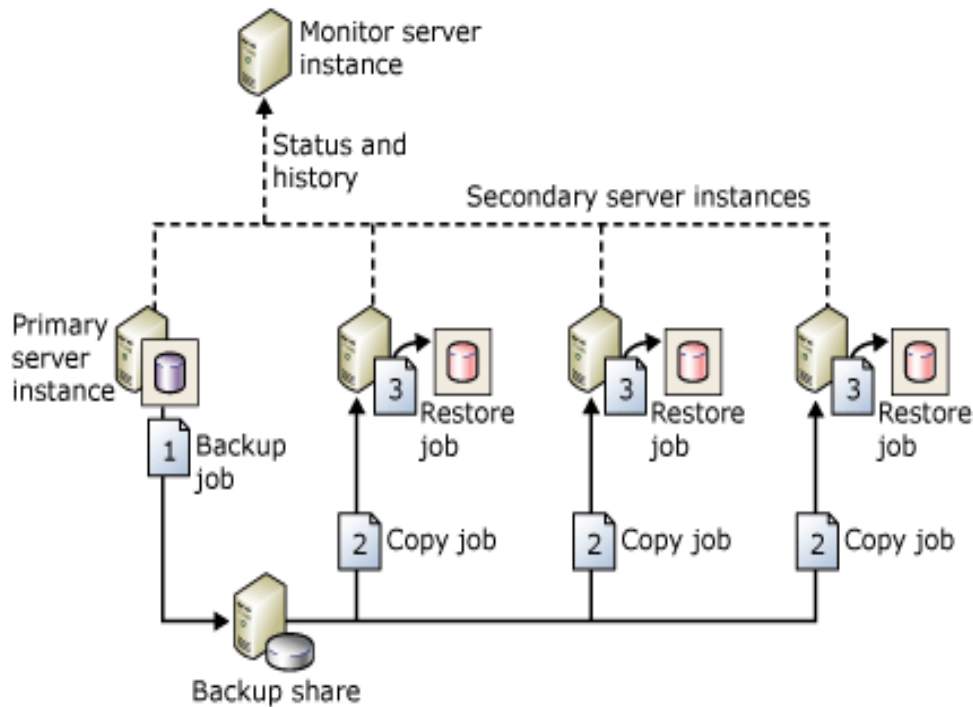
Restore job:

- A SQL Server Agent job that restores the copied backup files to the secondary database

Alert job:

- A SQL Server Agent job that raises alerts for primary and secondary databases when a backup or restore operation does not complete successfully within a specified threshold.

Architecture Points:



1. This is one of the database level HA option in SQL Server.
 2. Initial full backup of primary database directly restore to secondary server.
 3. Backup job take the transactional log backup of primary database into backup share.
 4. From backup share copy job picks the t-log backups and copy to secondary server > copy folder.
 5. From copy folder restore job picks the backup file and restore in secondary database
- This is the continues process where Log shipping works.

Log shipping configuration steps:

1. Create backup, copy share and provide read_write permissions
2. Go to primary configure backup Job by providing local and network path.

In real time always use network path to place the backup files.

Note: Taking backup into local server is very fast compare to network path.

3. Connect to secondary instance > then provide copy share path and restore database mode.

1. No recovery: No users can able to access database

2. Standby: Database in read only mode and users can able to read the data.

Note: In log shipping secondary server database can be used for "reporting purpose".

4. Add monitor server if need and click OK

5. Verify log shipping status.

SQL Server 2005 Version that Supports Log Shipping

SQL Server 2005 Version	Available
SQL Server 2005 Enterprise Edition	Yes
SQL Server 2005 Workgroup Edition	Yes
SQL Server 2000 Standard Edition	Yes
SQL Server 2005 Developer Edition	Yes
SQL Server 2005 Express Edition	No

Advantages:

- Data Transfer: T-Logs are backed up and transferred to secondary server
- Transactional Consistency: All committed and un-committed are transferred
- Server Limitation: Can be applied to multiple stand-by servers
- Secondary database mode: Stand-by mode [Read_only]...Useful for reporting purpose
- Recovery model supports Full and bulk-logged...**database to simple recovery will cause log shipping to stop functioning**
- Edition is not necessary to be same for both primary and secondary servers.

Disadvantages:

- Failover: Manual
- Failover Duration: Can take more than 30 mins
- Role Change: Role change is manual
- Client Re-direction: Manual changes required

Troubleshooting Log Shipping:

- 1) Jobs disabled can be a cause for LS failure.
- 2) Backup Share permission issues.
- 3) Space issues in the backup share/local copy location.
- 4) SQL Server Agents stopped at Primary/Standby/Monitor.
- 5) Manual log backup can cause LS break.
- 6) Recovery Model changes from Full/Simple.
- 7) Backup/Copy/Restore Job owner change can cause permission issues and break LS.
- 8) Network Issues with Backup Share.
- 9) WITH RECOVERY statement fired at Standby server can bring secondary database ONLINE breaking LS.
- 10) Service Account changes can lead to permission issues.
- 11) Log backups getting corrupted
- 12) Backup schedule is changed can cause lot of delay which might raise an alert.

LOG SHIPPING SCENARIOS:

1. Monitoring in Log-shipping
2. What are the log shipping jobs?
3. Failover
4. Switchover
5. What are the reasons log shipping fails? If fail what happened?
6. What are the common errors numbers will get in log shipping?
7. What is .tuf [Transaction undo file] purpose? Overview
8. If we delete .tuf file what happen to LS?

9. If we do insert, update, delete in primary server database, changes replicate to secondary?
10. If I shrink primary database what happen to log shipping in secondary db?
11. If I delete some records in primary database what happen to log shipping in secondary db?
12. If I do truncate in primary database what happen to log shipping in secondary db?
13. If I take a manual log backup for LS configured database will it impact secondary?
14. Adding multiple secondary server to the existing configuration?
15. Adding file to log shipping database?
16. Patch management process in LS instance?
17. Reasons a backup job, copy and restore job fails?
18. How to change recovery models in log shipping expect simple? Yes we can able to changeSimple recovery not support for LS
19. If primary database log file full then what happen to secondary? How will we resolve?

SCENARIO: 1 MONITORING IN LOG-SHIPPING

1. Go to job view history and check status to monitor log shipping status. If all backup, copy and restore jobs are running then we can say your log shipping is in sync and working fine.

MSDB Tables:

PRIMARY SERVER TABLE:

1.dbo.log_shipping_primary_databases:

Number of databases are configured in logshipping in primary

Backup share local path

q

Backup share network path

Last transaction log backup and time stamp.

Monitor server ID

2.dbo.log_shipping primaries:

primary_id

primary_server_name

primary_database_name

maintenance_plan_id

backup_threshold

threshold_alert

threshold_alert_enabled

last_backup_filename

last_updated

planned_outage_start_time

planned_outage_end_time

planned_outage_weekday_mask

source_directory

3.dbo.log_shipping_primary_secondaries:

Secondary server instance name

Secondary server log shipping database name

SECONDARY SERVER TABLE:**1.dbo.log_shipping_secondary_databases**

Secondary ls databases

Last restore file

Last restore date

Restore mode

Disconnect user option

2.dbo.log_shipping_secondary:

Primary server instance name

Primary database name

Network backup share

Copy share location

Monitor server

Last copied file

Last copied date

2.dbo.log_shipping_secondaries

primary_id

secondary_server_name

secondary_database_name

last_copied_filename

last_loaded_filename

last_copied_last_updated

last_loaded_last_updated

q

secondary_plan_id

copy_enabled

load_enabled

out_of_sync_threshold

threshold_alert

threshold_alert_enabled

planned_outage_start_time

planned_outage_end_time

planned_outage_weekday_mask

allow_role_change

MONITOR SERVER TABLE:

1.dbo.log_shipping_monitor_primary:

Primary server

Primary database

Backup threshold

Last backup file

Last backup date

2.dbo.log_shipping_monitor_Secondary:

Primary server

Last copy file

Secondary server

q

Last restore file

Last restore date

Last copy date

Threshold

3.dbo.log_shipping_monitor_history_detail:

Backup ,copy and restore information maintain.

4.dbo.log_shipping_monitor_error_detail:

Stores error information.

5.dbo.log_shipping_monitor_alert:

Alert job id.

2. FAILOVER:::::: LOG SHIPPING IS MANUAL FAILOVER AND NO AUTOMATIC FAILOVER SUPPORTS

> This is done when the primary database is No longer available. It is not pre-planned.

> When primary database goes down my application cannot be able to connect and log shipping is no more exists.Need to bring secondary database online manually is called as “failover”

Steps:

1. Disable all backup, copy and restore jobs, alert job. Inform application to stop the app to avoid any user co0nnections
2. Apply Tail log backup to recover active transactions in primary database...

3. Compare [BACKUP SHARE AND COPY SHARE] and move .trn files from primary to secondary copy folder.

Note 1: methods can be move the backup file

== Manually copy and paste from backup share to copy share by checking the time stamp and LSN number

== Or just run the copy job and which automatically copy to copy share. Please copy manually only TAIL LOG BACKUP

Note 2: methods TO RESTORE the backup file IN SECONDARY:

Note: Find what was the last restore backup file in Secondary

From msdb database, log shipping secondary tables we can get what was the last backup file restored...

Go to secondary server> MSDB> select * from sys. log_shipping_secondary_databases >> this table give data of last restored backup file.

4. Restore pending .trn backup files in secondary database with no recovery to recover the transactions (If tail log backup).

Note: Restore or copy by running manually or run the copy \restore jobs.

5. Restore last .trn backup file with recovery to get your secondary database up & running. If I have tail log then restore last TAIL log backup file with recovery

6. All login transfer to secondary database.

7. Inform to application team with new server instance name and database to start transactions.

Note: Log shipping supports manual failover method and does not support automatic failover.

3. HOW TO MOVE LOGINS IN LOG SHIPPING FROM PRIMARY TO SECONDARY SERVER DATABASE?

1. Whenever you create any login in Primary server then take the create login script in Primary server and execute in secondary server.
2. After login creation happen in secondary server.
3. But a user from primary server to secondary server is not possible to proceed manually. Due to secondary server database is in read-only\restoring state.
4. Wait or run manually for the backup\copy\restore job to run after login creation.
5. Once done, your secondary database gets added into user under database automatically.
6. Enable the login in secondary server.

4. SWITCH-OVER:

- > This is done when both primary and standby databases are available. It is pre-planned.
- > **Switch-Over:** Swapping the roles that means-- your primary become -secondary [ONLINE - Restoring or standby mode]
- > Secondary becomes- Primary [From Restoring\stand by Online state]

DISASTER RECOVERY: [DR process] this concept used for high availability solutions to test both primary and secondary are working fine instead of waiting for any disaster. Can called as proactive check.

Every 6 Months or 1 Year DR Test happens. In real time down time is required.

Pre-step:

1. Disable all backup, copy and restore jobs, alert job
2. Take a t-log backup for Primary database with no recovery mode...

Backup log dbname to disk='path' with no recovery

After primary becomes secondary server. Role changed from ONLINE TO RESTORING MODE

3. Same like your failover operations; compare your backup and copy folders, move files

4. Restore secondary database by using t-log backup with recovery to change the role. Now secondary database will be in online state (Role swapped)

5. Start reverse configuring log shipping from secondary to primary server [switchover].

Post Activities:

> Inform application to point connection to the current primary which was previous secondary to test.

> If apps and users are able to work then DR test is successful.

SCENARIO: 5 WHAT ARE THE LOG SHIPPING JOBS?

Primary server: Backup job

Secondary server: Copy and restore jobs

Monitor Server: Alert job

Note: If monitor server is not included then alert job create in both primary and secondary

SCENARIO: 6 WHAT ARE THE REASONS LOG SHIPPING FAILS? IF FAIL WHAT HAPPENED ?

Reasons:

1. backup jobs fails
2. copy job fails
3. restore job fails
4. log backup fails
5. agent not working
6. disk space
7. network issue
8. permission issue
9. incorrect network path
10. domain issue
11. .tuf file deleted ---- restore job will fail

- 12. Database not available
- 13. Instance not available
- 14. Recovery model change

SCENARIO: 7 WHAT ARE THE COMMON ERRORS NUMBERS WILL GET IN LOG SHIPPING?

Error:

14420 [Primary server –backup job related]

And

14421 [Secondary server copy and restore jobs]

SENEARIO: 8 WHAT IS .TUF [TRANSACTION UNDO FILE] PURPOSE? OVERVIEW

TUF File (Transaction UNDO file):

Contains only uncommitted transactions and create in secondary server but not in primary...ONLY SECONDARY DB IS IN STANDBY MODE

A .TUF file is a Microsoft SQL Server Transaction Undo file.

The TUF file basically contains the information with respect to any modifications that were made as part of incomplete transactions at the time the backup were performed.

- > This file contains uncommitted transactions
- > This file creates only in STANDBY mode
- > This file resides always under secondary server> copy folder

Note: If .tuf deleted, log shipping is not going to work.

Is there any way to re-create .tuf file... i have configured log shipping but unfortunately tuf file has been removed and now log shipping has been stop and we are unable to up the same.?

Ans: Impact is only restore job failed in secondary. We have reconfigure the log shipping one more time to re-create .TUF file

Note: If I delete .TUF file then impact to restore job but not copy and backup jobs.

- .Tuf file is updates dynamically whenever any log backup file restore
- .TUF file only creates in secondary server > copy folder>
- No .tuf file creates in “NORECOVERY “mode.

>.WRK [work file]: Work file creates in secondary server and the main purpose is it contains copy job information. This file used by only log shipping copy job

WORK FILE [.WRK]:- To manage the file copy process from Primary server to Secondary server, .WRK files temporarily generated.

Means, The .wrk file got generate when the transaction log backups files are being copied from the backup location (Commonly at Primary Server end) to the secondary server by the agent job named as LS-Copy on the secondary, and when file copied completely at secondary server, they renamed to the .trn extension.

The temporary naming using the .wrk extension indicates/ensure that the files will not picked up by the restore job until successfully copied.

Wrk file creates in both no recovery and standby mode.

SENEARIO: 9 IF I DELETE SOME RECORDS IN DATABASE WHAT HAPPEN TO LOG SHIPPING IN SECONDARY DB

Yes Secondary server database records will be deleted

SENEARIO: 10 IF I TRUNCATE SOME RECORDS IN DATABASE WHAT HAPPEN TO LOG SHIPPING IN SECONDARY DB

Yes Secondary server database records will be truncated

SCENARIO: 11 IF I SHRINK PRIMARY DATABASE WHAT HAPPEN TO LOG SHIPPING IN SECONDARY DB?

Shrinking: if u use shrinking database release spaces in os to your disk

If we perform shrink in primary database automatically shrinking replicate to secondary database as well...No impact to log shipping configuration

SENEARIO: 12 IF I TAKE A MANUAL FULL BACKUP FOR LS CONFIGURED DATABASE WILL IT IMPACT SECONDARY DB?

Nothing to impact Log shipping.

Due to LSN mismatch log shipping will not work (restore job fail)

Note: Recommended in LS take copy only backup if require to take full backup

SENEARIO: 13 IF I TAKE A MANUAL LOG BACKUP FOR LS CONFIGURED DATABASE WILL IT IMPACT SECONDARY DB?

Due to LSN mismatch log shipping will not work (restore job)

Note: Due to this reason in log shipping if any user used to take adhoc full backup then we will always USE” **COPY ONLY FULL BACKUP** “not to distribute any LSN number and to work log shipping as usual.

SENEARIO: 14 ADDING MULTIPLE SECONDARY SERVER TO THE EXISTING CONFIGURATION?

1. Get confirmation from client\customer.
2. Go to primary server> database>properties>Log shipping> add> secondary instance details+copy share location+db recovery state
3. In 2nd secondary server again create additional copy and restore jobs.

Note: multiple secondary server > each secondary server should contains at least 1 copy and 1 restore jobs.

No downtime is required for adding secondary server.

SENEARIO: 15 ADDING FILE TO LOG SHIPPING DATABASE?

1. Go to primary database add secondary .ndf or .ldf file.

Impact: After adding a file to log shipping db then there will be no impact to backup job, copy job but restore job gets fail.

2. Require to perform manual restoration in secondary server.
3. Identify after adding a file what is the recent backup is happened then confirm by perform

RESTORE FILELISTONLY FROM DISK='PATH'

Note: All backup files are moved to copy share and restored expect backup file after file added

4. Go to secondary and try to restore the log backup file with move option,

restore log dbname from disk='[path of log backup file n copy folder]'

With move 'logical name of file' to 'physical path of secondary', no recovery

Ex: - Restore log AMAZONDB FROM Disk='E:

\CS_AMAZONDB\AMAZONDB_20141129024726.trn'

with move 'AMAZONDB_FILE1' TO 'c: \Program Files (x86)\Microsoft SQL Server\MSSQL.4\MSSQL\DATA\AMAZONDB_FILE1.ndf', no recovery

5. Verify by running restore job whether log shipping is working or not. If works, log shipping is in sync.

Note: Always keep same version of Sql between primary and secondary

If primary -2005 and secondary-2008\higher

Then you can perform all operations but except SWICHOVER Scenario in log shipping.

Note:-

1. In log shipping for the version of Sql 2005 if any changes performed at primary server but the changes are failed

2. To roll back the data we need to restore with fresh full backup, after restoring full backup if I enable log shipping jobs my restore job is going to fail due to LSN number mismatch

Same in Sql server 2008 version just take a fresh log backup from primary and restore in secondary automatically Log shipping will going to start

SCENARIO: 16 REMOVAL A FILE INTO LOG-SHIP DATABASES:

If we remove file in primary the automatically remove from secondary after backup restore.

Note: No impact to backup, copy and restore jobs.

SCENARIO: 17 PATCH MANAGEMENT PROCESS IN LS INSTANCE?

When an instance of SQL Server is configured as part of Log Shipping, it is important to install service packs in a correct sequence; otherwise we may get unexpected issues.

For Log Shipping:

There is no required sequence to apply a service pack for Primary, Secondary and Monitor servers in a Log Shipping environment. The following is the steps about apply service pack:

1. Apply the service pack on the Monitor server.
2. Apply the service pack on the all Secondary servers.
3. Apply the service pack on the Primary server.

Dbcc sqlperf (logspace):- Find out the log file size

SCENARIO: 18 REASONS A BACKUP JOB, COPY AND RESTORE JOB FAILS?**Backup job failure:**

- Agent failure
- Disk space issue
- MSDB database corruption
- Sharing permission issue.
- Recovery model changes
- Incorrect path

Backup job disabled.

Copy Job Failure:

Lack of permission on backup folder

Network failure

Domain issue

Copy job disable

Job owner changes

Owner has no permissions.

Restore Job failure:

LSN Mismatch due to log backup file missing

Permission issue on local copy folder

Agent down

Job owner changes

Job owner has no restore permissions.

18. HOW TO CHANGE RECOVERY MODELS IN LOG SHIPPING EXCEPT SIMPLE?

Yes we can be able to change ...Simple recovery not support for LS

> In log shipping we can change recovery model from full to bulk-logged or bulk-logged to full...There will not be any impact to LS.

> But is we change from full to simple or simple to full then we need to trigger one full backup as a mandatory.

Note: System databases [master, model, msdb, Tempdb] log shipping configuration is not possible.

Can i script log shipping?

No. currently, it is not possible to script log shipping. the only supported means of setting up log shipping is through the wizard

Can I set up log shipping between servers in multiple domains?

Yes. It is possible to set up log shipping between servers that are in separate domains.

In SQL 2005 version: Not possible to configure log shipping between 2 different domains.

In SQL 2008 version onwards: Possible to configure log shipping between 2 different domains.

There are two ways to do this:

- **Use pass-through security.** Configure Windows NT accounts with the same name and passwords on the primary, secondary and monitor servers. Configure SQL Server related services to start under these accounts on all servers and use SQL authentication while setting up log shipping to connect to the monitor server. Or
- Use conventional Windows NT security. You must configure the **domains with two-way trusts**. SQL Server related services can be started under domain accounts. Either SQL authentication or Windows authentication can be used by jobs on the primary and secondary servers to connect to the monitor server

WHAT EDITION OF SQL SERVER DO I HAVE TO HAVE TO SET UP LOG SHIPPING?

Primary Server:: Enterprise or Developer Edition

Secondary Server:: Enterprise or Developer Edition

Monitor Server::Any Edition

Note: Only Enterprise and Developer editions of SQL Server support log shipping.

WHAT TO DO IF MY STANDBY DATABASE CRASHES?

- Re-establish log shipping in case of one standby
- Re-establish log shipping only on standby server (i.e. remove the Copy/Restore/Alert jobs and delete the standby database) and add the Standby instance as a new standby database.

LOG SHIPPING BETWEEN SQL SERVER VERSIONS (PERHAPS 2005 TO 2008):

- You can set up log shipping between versions, however things aren't that simple. If you use it in this way and you need to fail over to the standby server, you now have no way to swap the log shipping roles, as you can't then log ship back from the 2008 server to the 2005 server.

Configure log shipping by using third party tools:

<http://www.databk.com/walkthrough5.htm>

<http://support.microsoft.com/kb/314515>

LOGSHIPPING INTERVIEW QUESTIONS:

1. Architecture\concept of log shipping
2. How many jobs will create if logs hipping configured?
3. Where my backup, copy restores and alert jobs will create?
4. Advantages & disadvantages of LS?
5. Failover Process
6. Switch over process
7. Primary database file is full?
8. If i take a manual full backup what will happen to my LS?

9. If i take a manual T-log backup what will happen to my LS?

10. If I perform shirking on primary LS database what will happen? Is it effect to secondary server database?

11. Can I configure log shipping for system databases?

12. How to delete log shipping?

13. If change recovery model? LS there any impact to log shipping database?

> If you change the recovery mode of an existing secondary database, for example, from No recovery mode to Standby mode, the change takes effect only after the next log backup is restored to the database.

14. Scenarios:

DB MIRRORING

What is Database Mirroring?

- > Database mirroring is a primarily software solution for increasing database availability.
- > It maintains two copies of a single database that must reside on different server instances of SQL Server Database Engine.

(Or)

- > Database mirroring transfers transaction log records directly from one server to another and can quickly fail over to the standby server.

Mirroring pre-requisites:

q

> Make sure that the two partners that is the principal server and mirror server, are running the same or same version

> Same database in both principle and mirror

> Same collation should be in both principle and mirror

> Recovery model should be always full

> Minimum SQL Server 2005 SP1 is required to configure mirroring or if you have RTM then you have enable trace

DBCC TRACEON (1400)--Only for Sql 2005 RTM...From Sql 2008 onwards no need to have any sp even can configure with RTM

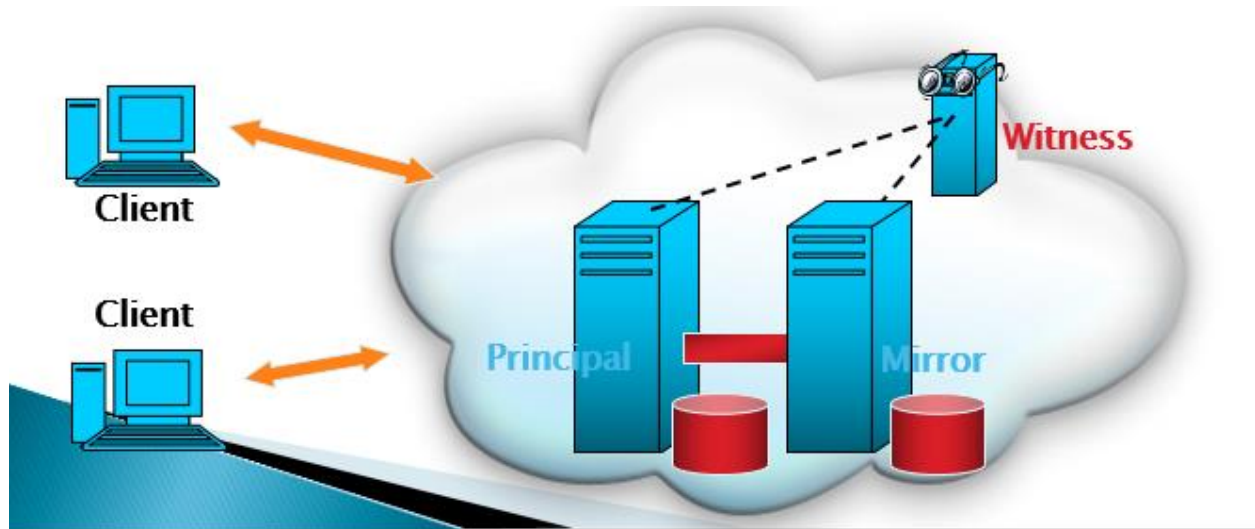
> Same build number

> Same bit of SQL [X 86 OR X-64 BIT]

> Minimum 2 servers are required.

> Collect service accounts and provide at the time of configuration.

Mirroring terminologies (or) Components:



- **Principal** – the server that holds a copy of database that is accessible to client applications at any given time.

(Or)

Where db is online and access to application\users

- **Mirror** – the server that holds copy of database that is always in restoring state that is not accessible to the applications.

(Or)

Where db is restoring and not having access to application\users

- **Witness** – the optional server that is useful to provide an automatic failover mechanism in case of any failure on principal server.

(Or)

Used mainly for failover\monitoring.

- **Endpoint:** A SQL Server object that enables Principal, Mirror & Witness servers to communicate over the network.

Encrypt data sent through this endpoints from server to server.

How to create port:

firewall.cpl-->go to advance settings--> Inbound settings

Mirroring Data transfer type:

System databases cannot be a part of mirroring.

Snapshot on mirror is possible for read-only purpose that is ideal for reporting requirements.

Transaction safety level – that determines whether the changes on the principal database are applied to the mirror database synchronously or asynchronously. Two safety levels—OFF and FULL.

1. Synchronous:

> In this mode when any log record sent from principle to mirror, acknowledgement has to send back from mirror to principle. Moreover first the transaction should commit @ mirror server.

> Performance impact is there but it is minimal

> No data loss.

2. Asynchronous:

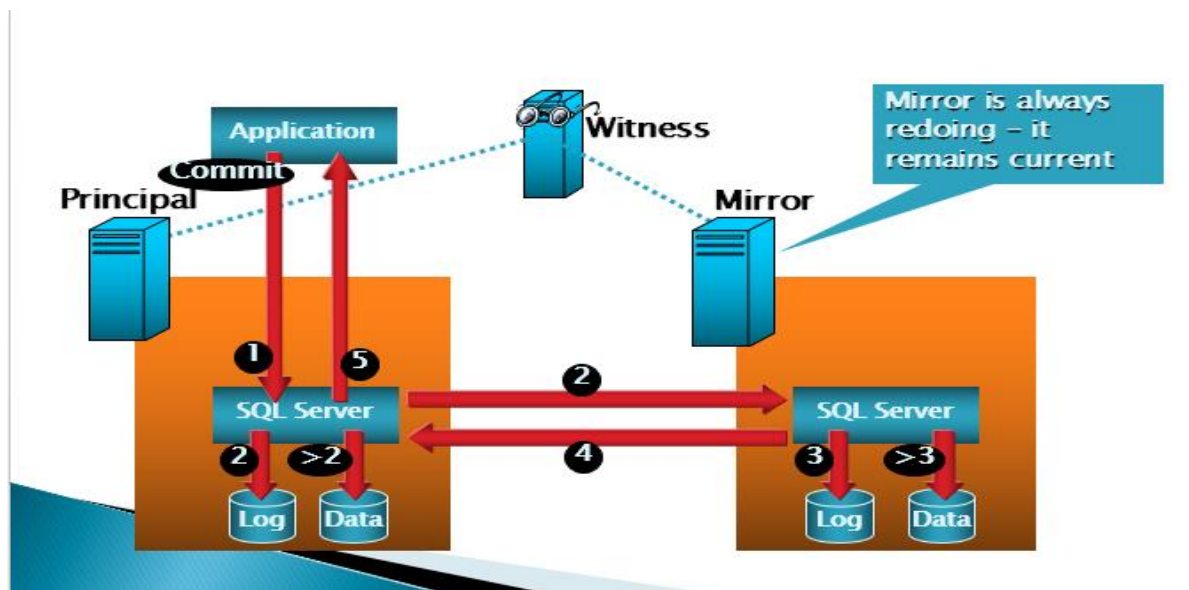
> In this mode when any log record sent from principle to mirror, acknowledgement no need to send back from mirror to principle. Moreover first the transaction should commit @ principle server itself.

> In this mode data loss chances are high

> Another configuration mode (possible) asynchronous mode with no witness server although this is possible to setup it is not recommended because it combines the risk of data loss and split-brain scenario.

Operating Mode	Transaction Safety	Transfer Mechanism	Witness Server	Failover type
High Availability	FULL	Synchronous	Yes	Automatic or Manual
High Protection	FULL	Synchronous	No	Manual
High Performance	OFF	Asynchronous	No	Forced

Mirroring Architecture:



Architecture Points:

1. In mirroring transaction log record write into log file (Log buffer).
2. The same Log records sent from principle to mirror server database
3. The complete data transfer from principle to mirror by ENDPOINTS.
4. Default end points for mirroring used [Principle-5022, Mirror-5023, Witness-5024]
5. Exact copy of the database maintained at mirror server.

Mirroring configuration steps:

1. Take full backup in principle database

2. Restore in mirror server with same db name with no recovery mode
3. Start mirroring configuration by providing endpoints.

Endpoint: Mainly help to communicate and data transfer from principle to mirror server database

Note: We can change the endpoints and can configure mirroring.

Database Mirroring Benefits:

- Database mirroring architecture is more robust and efficient than Database Log Shipping.
- It can be configured to replicate the changes synchronously to minimized data loss.
- It has automatic server failover and client failover mechanism.
- Configuration is simpler than log shipping and replication, and has built-in network encryption support (AES algorithm).
- Does not require special hardware (such as shared storage, heart-beat connection) and cluster ware, thus potentially has lower infrastructure cost

Mirroring Modes or Types:

1. High Availability with automatic failover [Including witness]
2. High performance
3. High protection

1. High Availability with automatic failover [Including witness]:

- > Witness instance should be included
- > The data transfer type in this mode is SYNCHRONOUS. I.e. acknowledgement should sent from mirror to principle
- > Data safety is always FULL
- > Transaction first should commit in mirror server database and then it commit in principle server database.

- > If transaction is not commit in mirror due to any reason again the same transaction sends from principle to mirror for further process.
- > This mode supports both automatic and manual failover methods.

2. High Protection: [SYNCHRONOUS]

- > Witness should not be included
- > The data transfer type in this mode is SYNCHRONOUS. i.e. acknowledgement should sent from mirror to principle
- > Data safety is always FULL
- > Transaction first should commit in mirror server database and then it commit in principle server database.
- > If transaction is not commit in mirror due to any reason again the same transaction sends from principle to mirror for further process.
- > This mode support only manual failover.

3. High performance: [ASYNCHRONOUS]

- > In this mode there is no witness server
- > The data transfer type in this mode is ASYNCHRONOUS. i.e. No acknowledgement should sent from mirror to principle after any transaction reach
- > Data safety is always OFF
- > Transaction first should commit in Principle server database and then it commit in Mirror server database.
- > Principle db does not know whether transactions are reaching mirror or not.
- > In this mode data loss chances are high.
- > This mode support only forceful failover types.

DATA BASE MIRRORING SCENARIOS:

1. MONITORING IN MIRRORING:

Note: In Sql server database mirroring to monitor we have inbuilt tool released called as "LAUNCH DATABASE MIRRORING MONITOR".

> DB> right click> click > mirror monitor> Check the columns

CURRENT ROLE

MIRRORING STATE

UNSENT LOG: VALUSE SHOULD 0 KB

UNRESTORED LOG: VALUE SHOULD BE 0 KB

Then we can say mirroring is in sync.

> **Unsent Log:** When principle database\server is not online then what ever the transactions are running principle should be store in "UNSENT LOG"

> Unsent log size increases more if mirror database not present for longer time.

> This unsent log is part of principle server database.

> **Unrestored log:** After some time mirror server \database present the data which is pending in principle server unsent log will be sent to mirror unrestored log.

> **Mirror commit overhead:** How much time take to commit any transaction in mirror server.

T-SQL Method:

To verify / check Database mirroring status using a system stored procedure

Use msdb

sp_dbmmonitorresults @database name = 'DBNAME'

1. To list all the database mirror endpoints run,

Select * from sys.database_mirroring_endpoints

2. To list all the endpoints

Select * from sys.tcp_endpoints

3. Manually fail over a database mirroring session at the mirror in High Availability Mode by issuing the following command at the principal

ALTER DATABASE SET PARTNER FAILOVER

4. Select * from sys.database_mirroring ---> provides information about principal and mirror

5. Select * from sys.database_mirroring_witnesses ---> provides information about witness server

2. FAILOVER PROCESS IN MIRROR

There are three types of failures:

1> Automatic failover

2> Manual failover

3> Force failover

1> Automatic failover:

Whenever principle database\server down witness come into picture to perform failover to mirror server and brings the mirror database online i.e...Accessible to end user\application.

If no witness then DBA need to perform manual failover operations.

2> Manual failover process:

Whenever principle goes down DBA should perform manual failover if no witness.

> GUI: go to databases> right click> mirror> click failover tab

> Query: **alter database [dbname] set partner failover**

Note: This type of failover supports in “HIGH AVAILABILITY WITH AUTOMATIC FAILOVER + High protection” modes

Post failover:

> Application\end users can auto redirect connection to mirror server. No need to inform DBA specifically to app team.

> Logins need to create in mirror server whenever you create in principle instance...

3> Force failover: [ASYNC] Supports only in high performance mode.

ALTER DATABASE [DBNAME] SET PARTNER FORCE_SERVICE_ALLOW_DATA_LOSS

3. SWITCHOVER PROCESS:

> In mirroring there is no separate switch over process. Why because if we do failover then automatically roles get changes and again no switch over concept is required.

> **After failover in mirroring, Auto client direct happen automatically.**

Principle db: become mirror: restoring state

Mirror db: Become principle: Online state.

Note: In High performance mode, we cannot be able to perform manual failover. If we need to do failover then first need to change the database transfer type from Asynchronous mode to synchronous mode.i.e. please change the mirroring mode from high performance to high protection.

> In High performance mode, only supports FORCE FULL FAILOVER MODE

ALTER DATABASE [DBNAME] SET PARTNER FORCE_SERVICE_ALLOW_DATA_LOSS

> **Manual failover supports only SYNCHRONOUS MODE ONLY**

> **Force Full Failover supports in ASYNCHRONOUS MODE**

4. ADDING A FILE STEPS:

1. Break the mirroring between principle and mirror server.

Alter database dbname set partner off

2. Add the file in principle database.
3. Apply one transactional log backup in principle server database and copy the log backup to mirror server.
4. Take the t-log backup file and restore in mirror database with no recovery by using with move.

RESTORE log testdb FROM Disk='path\testdblog.trn'

with no recovery,

MOVE 'logical file name ' TO 'mirror path\physical file name'

5. Reconfigure the mirroring for the database and check the stats of mirroring monitor.

Note: Generally perform these kind of activities in weekends.

Method 2: Instead of breaking the mirroring we can perform "PAUSE Mirroring" from database properties.

Repeat all steps as like from point 2 to point 4.

After 4th step we should resume the mirroring and check whether mirroring is working or not.

<http://www.mssqltips.com/sqlservertip/2834/how-to-add-a-database-file-to-a-mirrored-sql-server-database/>

5. FILE MOVEMENT IN MIRRORING:

Start activity @ Principle server:

1. Collect logical file names for your databases [User and system]

sp_helpfile

2. Change logical name for user database by using alter command.

```
alter database dbname modify file (name='logical file name', filename='newpath') -----Data file
```

```
alter database dbname modify file (name='logical file name', filename='newpath') -----log file
```

Note: Verify whether logical name is updated with new path by using sp_helpfile again...

3. Take instance offline due to db cannot take offline (Mirroring configured)

> If witness included, then automatic failover occur and mirror become principle.

4. Move physical files to new path and bring the instance online.

Note: While moving physical .mdf, .ldf, .ndf files use always copy & paste method but not cut and paste.

@ Mirror which is current principle:

Repeat the same steps @mirror which is current principle server.

Note: When you are performing @mirror server automatic failover happen and principle become online at principle instance.

Do the validation whether file movement is performed.

Note: If we don't have witness then DBA need to perform manual failover before taking the instance offline in both principle and mirror instance.

6. INSERT \UPDATE\DELETE \SHRINK\TRUNCATE IN PRINCIPLE WHAT HAPPEN TO MIRROR DATABASE?

If we delete, insert, update, truncate, shrink records in principle server then automatically delete, insert, update, truncate, shrink records in mirror databases also without effect mirroring configuration.

7. JOBS IN MIRRORING?

Only 1 job: “Database Mirroring Monitor Job” gets created in both principle and mirror server. But witness does not have the job

8. WHAT IS UNSENT LOG AND UNRESTORED LOG? REASONS TO FULL?

Reasons for unsent log become store:

- Mirror db offline
- Mirror server is not available
- End point issue
- Database full [.MDF, .NDF or .LDF]

9. FIXING \MOVING LOGINS?

> Take a login script in principle server and execute the same script in mirror server to recreate in mirror instance.

> After creation login in disabled status.

> In mirroring users move automatically from principle to mirror server database after synchronization.

> If you want to see then perform failover to mirror.

(Or)

> When create any login @ principle then create the same login @mirror server as well.

> If you map the same login @ principle to any mirror configured database, then same login create as a user in mirror database.

> Automatically same user account replicate from principal db to mirror db.

This point of case generally get "ORPHAN USERS"

Orphan User: A user without having login called "ORPHAN USERS"

How to find:

```
sp_change_users_login @action='report'
```

How to fix:

```
sp_change_users_login 'update_one', 'user name', 'login name'
```

Once fixed, automatically a mapping happen between login and user along with the permissions.

Note: After creating login @mirror server login state in DISABLE status. Ensure you have to enable the login.

Note: In real time whenever you create any login in principle ensure that the same login you create in mirror immediately.

10. MANUAL FULL \TLOG BACKUP?

> If we take manual full or t-log backup for principle database there is no impact to mirror server database and mirroring still works.

11. PATCH MANAGEMENT PROCESS IN MIRRORING?

Process steps:

1. Apply in witness server instance even there is no impact to mirroring [Only impact failover process become manual].
2. Once done, perform patch in mirror server instance only after witness online.
3. Once mirroring patch completed, perform manual FAILOVER to mirror instance before start patch in principle.
4. Apply patch in principle instance and fail back if required after the service pack.

Note: Check whether mirroring is working after service packs.

12. DATABASE SNAPSHOT ?

<http://blog.sqlauthority.com/2010/04/05/sql-server-2008-introduction-to-snapshot-database-restore-from-snapshot/>

Newly introduced in SQL Server 2005 version for database mirroring

Main purpose:

1. We generate snapshot on mirror database (Restoring status) and can perform data reading from snapshot in mirror server.
2. Snapshot is a read_only and static view.
i.e. we cannot update\delete \insert into snapshot database tables due to read_only.
3. If we want to read recent \concurrent changes from snapshot required to generate always RECENT snapshot to read recent updates.

Syntax to create snapshot:

CREATE DATABASE snapshot name ON

(Name ='dbname', Filename='path.ss1')

AS SNAPSHOT OF logical data file name

Case 1: Can we delete a database when snapshot is created?

No. You cannot delete the database until you first delete the snapshot database.

Case 2: Can we create a database by restoring from snapshot file?

Yes possible.

RESTORE DATABASE [DBNAME]

FROM DATABASE_SNAPSHOT = 'SNAPSHOTNAME';

Note: I can generate snapshot even database is in online.

13. MIRRORING FAIL REASONS:

1. Recovery model change
2. Endpoint issue disabled\removed
3. Network issue
4. Principle or mirror db files corrupted
5. Db name is different or change
6. Collation name is different between principle and mirror server.
7. Version and edition differ
8. Adding file will fail
9. Service account issue
10. Principle db log file full
11. Principle instance down
12. Mirror instance down

14. MIRRORING ADVANTAGES AND DISADVANTAGES?

Advantages:

Automatic failover

Auto client re direct

Maintenance activities can perform with less down time

Data safety is always 100% high in synchronous data transfer type.

Disadvantages:

> Direct there is no reporting support from mirror server due to restoring state until you have used Database SNAPSHOT

> Multiple mirror servers does not support. Only 1 mirror server is possible.

> It does not support bulk -logged or simple recovery model.

15. IF WE CHANGE THE RECOVERY MODEL IN PRINCIPLE DB WHAT HAPPEN TO MIRROR?

For mirrored configured database, you cannot be able to change the recovery model from full to any other bulk-logged\simple.

Note: why the reason: mirroring supports only full recovery model. If you change the recovery model mirroring is going to fails.

If you need to change, break the mirroring and then perform the recover model change operation. But again can't configure mirroring.

16. WHAT ARE THE NEW FEATURES IN SQL 2008 MIRRORING ONWARDS?

> If any page is corrupted in mirror server database in 2005 SQL then mirroring is in out of sync and DBA should fix this issue manually by performing auto page recovery process.

> Where as in SQL 2008 onwards AUTO PAGE RECOVERY is possible. If any page is corrupted in MS then get the same copy of the page from PS.

> Data encrypt and sent from Principle Server to Mirror Server

q

- > Data compress and sent from Principle Server to Mirror Server
- > I/O errors on the principle server may be fixed during the mirroring session
- > I/O errors on the mirror server require the mirroring session to be suspended
- > Write-ahead on the incoming log stream on the mirror server
- > Improved use of log-send buffers
- > Manual failover no longer required restart of database.

QUORUM:

1. Quorum contains the information of which instance is currently acting as principle (online) and which instance is acting as a mirror server.

Note: When witness included in mirror configuration, always before perform automatic failover witness reads from QUORUM to know from which instance to failover.

DYNAMIC MANAGEMENT VIEWS: This concept is introduced in SQL Server 2005 version.

The main purpose we can monitor SQL Server without consuming hardware resources like DBCC queries.

Whereas DBCC queries always create slight impact on system.

To know list of DMV's:

```
SELECT name, type, type_desc FROM sys.system_objects WHERE name LIKE 'dm_%' ORDER BY name
```

Output:

List of DMV'S

2005 version of Sql -89..... 2008 version of sql-176.....2012 version of sql-900+

17. MIRRORING DMV'S:

1. Sys.dm_db_mirroring_auto_page_repair: New in 2008 version of SQL Server

1. Returns a row for every automatic page-repair attempt on any mirrored database on the server instance.
2. This view contains rows for the latest automatic page-repair attempts on a given mirrored database, with a maximum of 100 rows per database.

connection_id

transport_stream_id

state

state_desc

connect_time

login_time

authentication_method

principal_name

remote_user_name

last_activity_time

is_accept

login_state

login_state_desc

peer_certificate_id

encryption_algorithm

encryption_algorithm_desc

receives_posted

is_receive_flow_controlled

sends_posted

is_send_flow_controlled

q

total_bytes_sent

total_bytes_received

total_fragments_sent

total_fragments_received

total_sends

total_receives peer_arbitration_id

2. Sys.dm_db_mirroring_connections:

Returns a row for each connection established for database mirroring.

Db id

File id

Page id

Error type

Page status

Modification time

DIFFERENCE BETWEEN LOGSHIPPING AND DB MIRRORING

Log shipping	DB Mirroring
Database level HA	Database level HA
Transaction log backups shipped from PS TO SS	Transaction log records sent from principle to mirror server
Data transfer from PS to SS by using backup, copy and restore jobs	Data transfer from PS TO MS by using ENDPOINTS
Min down time 30 min	Min down time 3 sec
database name may or may not be same still LS support	DB name should be same
Support different editions	Should be same edition
Data loss chances are more	Data loss chanced are very less
Support both bulk and full recovery model	Support only full recovery model
Directly can use secondary server for reporting	Not possible MS for reporting directly
Multiple secondary servers are possible	Only 1 mirror server can possible.
Support only manual failover	Support automatic ,manual, forceful failover types
Data transfer type is always an Asynchronous	Data transfer support both synchronous and Asynchronous types
Failover and switch over supports	Only failover is possible
If primary server is down then users cannot connect automatically to secondary server as no auto client redirect	Supports auto client redirect method.
Monitoring is via jobs or reports	Mirroring monitor tool is now in place to monitor.

End point changes

-- Create endpoint for PRINCIPAL server --

```
CREATE ENDPOINT [EndPoint4DBMirroring1430]
```

```
STATE=STARTED
```

```
AS TCP (LISTENER_PORT = 5022, LISTENER_IP = ALL)
```

```
FOR DATA_MIRRORING (ROLE = PARTNER,
```

```
AUTHENTICATION = WINDOWS NEGOTIATE
```

```
, ENCRYPTION = REQUIRED ALGORITHM RC4)
```

-- Create endpoint for MIRROR server --

```
CREATE ENDPOINT [EndPoint4DBMirroring1440]  
STATE=STARTED  
AS TCP (LISTENER_PORT = 5023, LISTENER_IP = ALL)  
FOR DATA_MIRRORING (ROLE = PARTNER,  
AUTHENTICATION = WINDOWS NEGOTIATE  
, ENCRYPTION = REQUIRED ALGORITHM RC4)
```

-- Create endpoint for WITNESS server --

```
CREATE ENDPOINT [EndPoint4DBMirroring1450]  
STATE=STARTED  
AS TCP (LISTENER_PORT = 5024, LISTENER_IP = ALL)  
FOR DATA_MIRRORING (ROLE = WITNESS,  
AUTHENTICATION = WINDOWS NEGOTIATE  
, ENCRYPTION = REQUIRED ALGORITHM RC4)
```

REPLICATION

What is Replication?

> It is a set of technologies for copying and distributing data and database objects from one database to another and then synchronizing between databases to maintain consistency. Using replication, you can distribute data to different locations and to remote or mobile users over local and wide area networks, dial-up connections, wireless connections, and the Internet.

> This is called as Object Level of High Availability

What can it do?

> Move data from one source to another.

> Manipulation/transformation of data when moving from source to destination.

> E.g. it can map data from a data-type in DB2 to an equivalent in Sybase.

> Provide a warm-standby system.

> Merge data from several source databases into one destination database

Replication Terminologies:

1. ARTICAL: A table, schema, column, rows, indexes, views, store procedure, triggers. Etc. can called as "ARTICAL" in replication.

2. PUBLISHER: Where data coming from application and the article is online.

3. SUBSCRIBER: Where it contains a copy of the publisher article.

4. DISTRIBUTOR: It act as intermediate between publisher and subscriber. i.e. distributing article data from publisher to subscriber(S).

5. LOCAL PUBLICATION: The number of articles included as a part of publisher server called "local publication"

6. LOCAL SUBSCRIPTION: The number of articles included as a part of subscriber server called "local subscription"

> Data transfer in replication: Replication always uses "replication agents" to transfer data.

Replication pre-requisites:

- > Verify that there are no differences in system collation settings between the servers.
- > Verify that the local windows groups and SQL Server Login definitions are the same on both servers.
- > Verify that external software components are installed on both servers.
- > Verify that CLR assemblies deployed on the publisher are also deployed on the subscriber.
- > Verify that SQL agent jobs and alerts are present on the subscriber server, if these are required.
- > Verify that for the certificates and keys used to access external resources, authentication and encryption match on the publisher and subscriber server.

TYPES OF REPLICATIONS:

There are 5 types of replications

1. SNAPSHOT REPLICATION [ONE DIRECTION]
2. TRANSACTION REPLICATION [ONE DIRECTION]
3. TRANSACTIONAL WITH UPDATABLE SUBSCRIPTION [BI DIRECTIONAL] --- Removed from SQL SERVER 2012
4. MERGE REPLICATION [BI DIRECTIONAL]
5. PEER-TO-PEER REPLICATION [BI DIRECTIONAL]

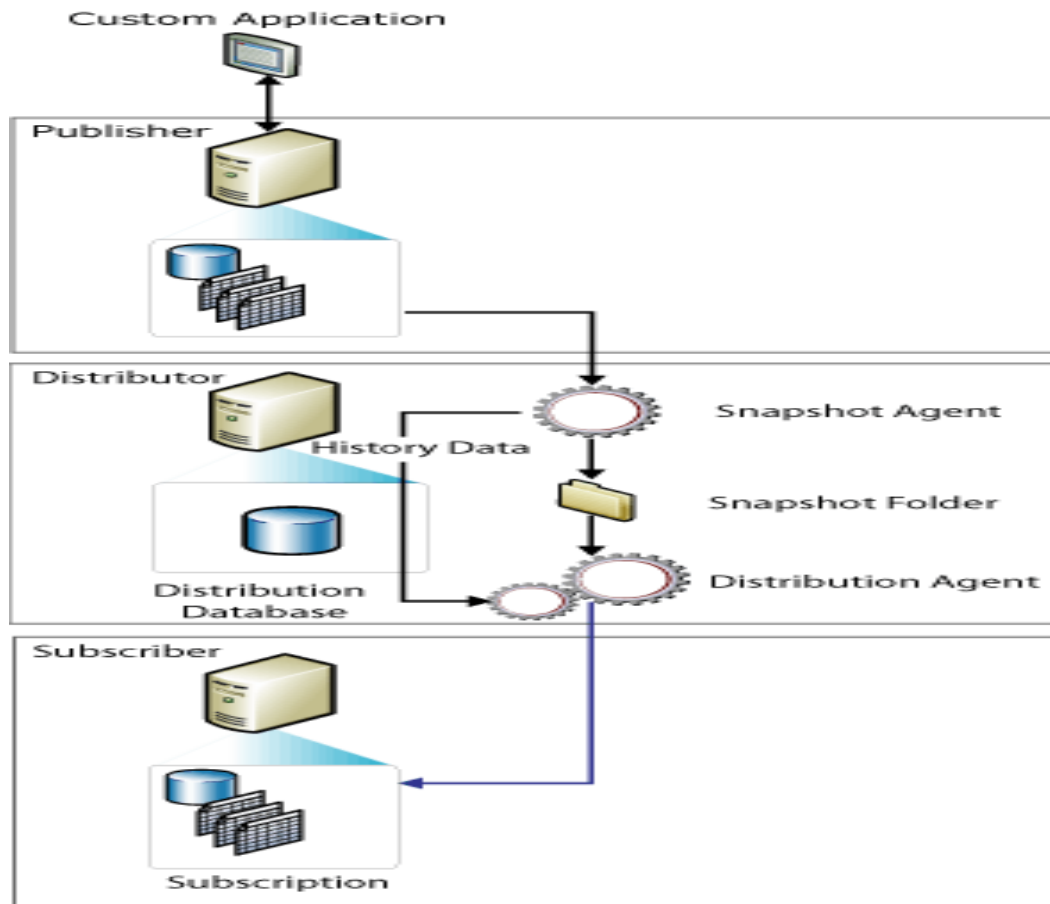
1. SNAPSHOT REPLICATION [ONE DIRECTION]

What is Snapshot Replication?

> Snapshot replication refers to a replication method between databases. During this process, data is infrequently updated at specified times by copying data changes from the original database (publisher) to a receiving database (subscriber).

MINIMUM REQUIREMENTS:

1. Minimum 2 server are required
2. Minimum 1 article is required
3. Snapshot folder is required
4. SQL SERVER AGENT SHOULD BE UP AND RUNNING
5. REPLICATION SUPPORTS ANY TYPE OF RECOVERY MODEL [FULL\BULK-LOGGED\SIMPLE].
6. Same SQL server edition and version is recommended to keep



ARCHITECTURE:

- > Snapshot taken from publisher on article and stored into snapshot folder in distributor server by SNAPSHOT AGENT
- > From snapshot folder collect snapshot by distributor agent and distributes the data to subscriber and store into subscriber server.

CONFIGURATION STEPS:

1. Configure distributor in distribution server changes:
 - > Distribution database creates under system database
 - > Login "distributor_admin" CREATE with SYS admin permissions automatically
 - > Replication 6 Maintainece jobs creates after configured distributor
 - > Provide the password after adding publisher instance and specify the snapshot folder path to store snapshots.
2. Configure publisher instance and select the article table to publish as a part of replication
3. Configure subscriber.

Key points:

- > Takes snap from publisher and keep to snapshot folder.

SNAPSHOT Background: When you generate a snapshot initial schema+data (if any) takes from publisher and store into snapshot folder.

- > The same snapshot replicate to subscriber and apply at subscription article.

SNAPSHOT REPLICATION AGENTS:

1. Snapshot agent

1. The Snapshot Agent is typically used with all types of replication.
2. It prepares schema and initial data files of published tables and other objects, stores the snapshot files.
3. Records information about synchronization in the distribution database.
4. The Snapshot Agent runs at the Distributor.

2. Distributor agent

1. Distributor agent takes snapshot files from snapshot folder, apply to subscriber.

2. Depending on the push and pull type of replication distributor agent operate either from distributor or from subscriber

Agent .exe file location:

Location: C:\Program Files\Microsoft SQL Server\90\COM

SNAP.exe

Distrib.exe

SNAPSHOT FILES IN REPL DATA FOLDER:

- 1. SCH [Schema Script]:** This contains the script of the published tables (schema)
- 2. BCP [Bulk Copy Program]:** Contains data which need to move to subscriber
- 3. PRE:** Contains drop script of the article
- 4. IDX [Index File]:** Contains index of publisher tables which need to apply in subscriber after creating.

PULL SUBSCRIPTION: In this mode distributor always works from subscriber and pulls data from distributor snapshot.

PUSH SUBSCRIPTION: In this mode distributor always works from distributor and sends data to subscriber article.

> By default always the type select as "PULL"

> Pull types always gives good performance compare to PUSH type.

REPLICATION ADVANTAGES AND DISADVANTAGES:**Advantages:**

- > Supports for small table sizes
- > Support any recovery model
- > Pub and sub db names can be different but still support.

Disadvantages:

- > Data loss chances are very high
- > Con current changes in pub are not replicate to sub immediately. Only when after re generating the new snapshot.

Note: When you configure snapshot replication with multiple subscribers then multiple distribution agents will create...

2. TRANSACTION REPLICATION [ONE DIRECTION]

What is Transaction Replication?

> Transactional replication is the automated periodic distribution of changes between databases. Data is copied in (or near) real-time from the primary server (publisher) to the receiving database (subscriber). Thus, transactional replication offers an excellent backup for frequent, daily databases changes.

Transactional replication typically starts with a snapshot of the publication database objects and data. As soon as the initial snapshot is taken, subsequent data changes and schema modifications made at the Publisher are usually delivered to the Subscriber as they occur (in near real time)

ARCHITECTURE:

- > Initial schema and data sends from publisher to subscriber by using snapshot agent method
- > After concurrent changes replicate from publisher to subscriber by using a new agent creates in transaction replication "log reader agent [logread.exe]".

TRANSACTION REPLICATION AGENTS:

- 1. Snapshot agent:** Take initial schema and data in publisher and store to snapshot folder.
- 2. Distributor agent:** Snapshot file and concurrent data should replicate to subscriber
- 3. Log reader agent:** Concurrent data changes in publisher send to distributor to subscriber

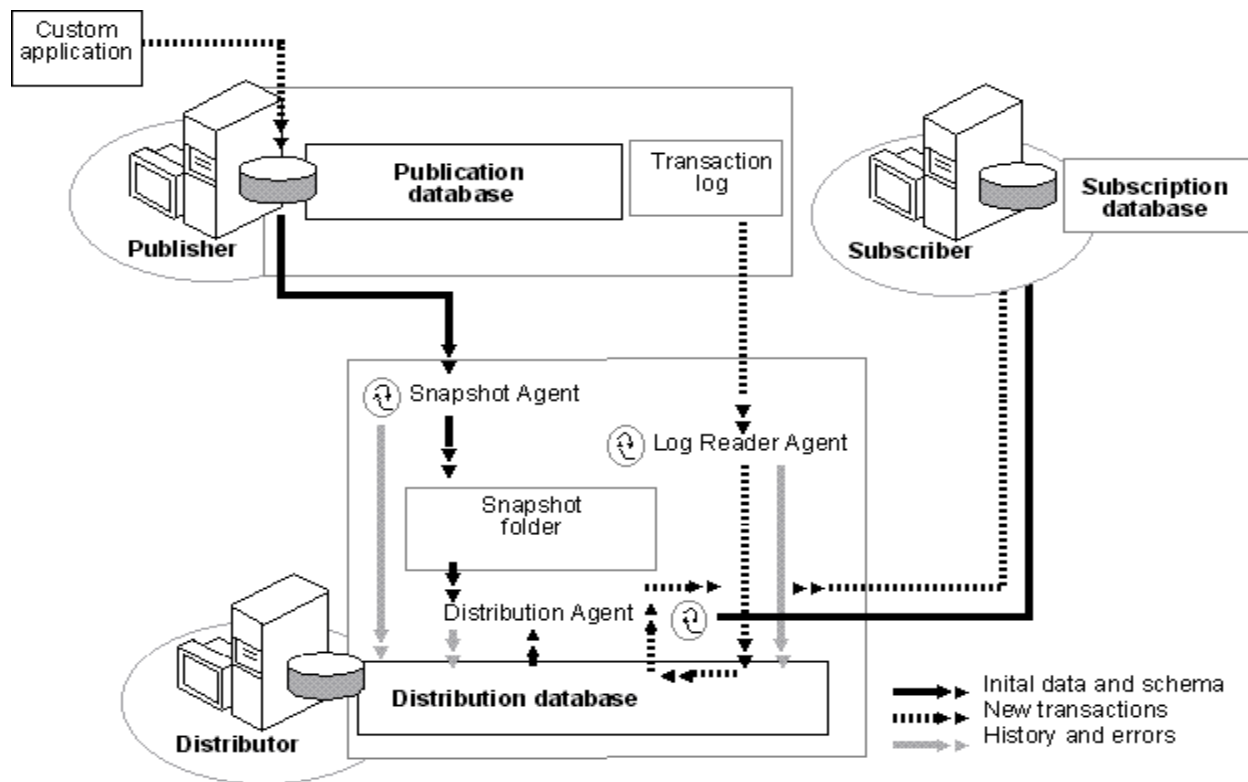
CONFIGURATION STEPS:

1. Configure distributor
2. Configure publisher
3. Configure subscriber

DATA FLOW POINTS:

Note: PRIMARY KEY SHOULD CREATE BEFORE CONFIGURING TRANSACTIONAL REPLICATION.

1. Initial snapshot generates in publisher and send to distributor snapshot folder.
2. From there distributor agent takes the same snap and apply to subscriber.
3. After any concurrent changes [DML operation or any] @ publisher table then those changes replicate immediately to subscriber by LOG READER AGENT via distributor agent
4. Log reader agent collect data from transaction log file @pub and stores into distributor > distribution database. From their distributor agent takes and send to subscriber.
5. Transactional replication is always suitable for any types of critical transaction happen every second.



> In transactional replication is a one directional type of replication

> If you perform DML at publisher only data goes to subscriber

q

- > But if you insert data at sub data won't come to publisher
- > First transaction commit at sub and later at publisher.

Note: How Tran replication works in simple recovery model?

- > Generally in simple recovery model when check point the transaction in VLF's truncated.
- > But in replication, when you perform any transaction as per WAL, should logged into transaction log after replication MARK THOSE TRANSACTIONS AS
- > Replication transaction and will not be deleted from log file in publisher until move to distributor even truncate operation comes.

Replication Monitor:

IN Replication, to monitor the synchronization status we have to use "REPLICATION monitor" tool...

Transactional Replication:

Latency: How much time take to send the transaction from pub to sub

TRACER TOKEN: Measures the latency values from publisher to distributor and distributor to subscriber and overall total latency.

- > Check column "PERFORMANCE" to know whether replication working fine or not.

REPLICATION JOBS:

1. Agent history clean up: distribution: Removes replication agent history from distribution database

Default Schedule: 10 min

2. Distribution clean up: distribution: Removes transactions from distribution database to control the size

Default schedule: 10 MIN

3. Expired subscription clean up: Detect and removed expired subscriptions from publication database.

Default schedule: Runs every day 1: 00 AM

4. Reinitialize subscriptions having data validation failures:

Detect whether any data validation errors then if any errors reinitialize the subscriber with fresh snapshot from publisher.

Schedule: No schedule

Note: This job not recommend to run in business hours which cause performance impact.

5. Replication agents backup: Checks whether replication agents are running or not

Default schedule: Run every 10 min

6. Replication monitoring refresher for distribution:

Refreshes cached queries by using replication monitor

Default schedule: Run continuously

3. TRANSACTIONAL WITH UPDATABLE SUBSCRIPTION [BI DIRECTIONAL]

> This type of replication called as BIDIRECTIONAL replication.

> Updatable subscriptions for transactional replication allow Subscribers to replicate changes to the Publisher. Triggers are added to the published tables in the subscription database, and when a change is made at the Subscriber, the trigger fires:

1. When data DML operation performs at publisher, same data sends to sub by using SNPAHOT and Log reader agents.

2. If you perform any DML operations at subscriber then data sent to publisher by using 2 methods.

- **For immediate updating subscriptions**, the change is propagated directly to the Publisher and applied using Microsoft Distributed Transaction Coordinator (MSDTC).
- **For queued updating subscriptions**, the change is first propagated to a queue and then applied to the Publisher by the Queue Reader Agent.

Transaction with Updatable Subscription Agents:

1. SNAPSHOT AGENT
2. LOG READER AGENT
3. QUEUE READER AGENT
4. DISTRIBUTOR AGENT

CONFIGURATION STEPS

1. Configure distribution
2. Configure publisher.

Changes: - When you configure, the article gets added with new extra column "MSREPL_TRAN_VERSION"

Note: - "MSrepl_tran_version" >this column is used for change tracking and conflict detection.

> Same published table replicate to subscriber by using snapshot.

> Concurrent changes from pub to sub by using log reader agent

3. Configure Subscriber:

Changes:-

1. At the time of configuration required to configure "LINKED SERVER" for the purpose of connectivity from subscription to publisher articles.
2. You have option to select the type of data sending from sub to pub either "MSDTC [IMMEDIATE SYNCRONISATION] OR QUEUE READER CHANGES"

Changes @subscriber:

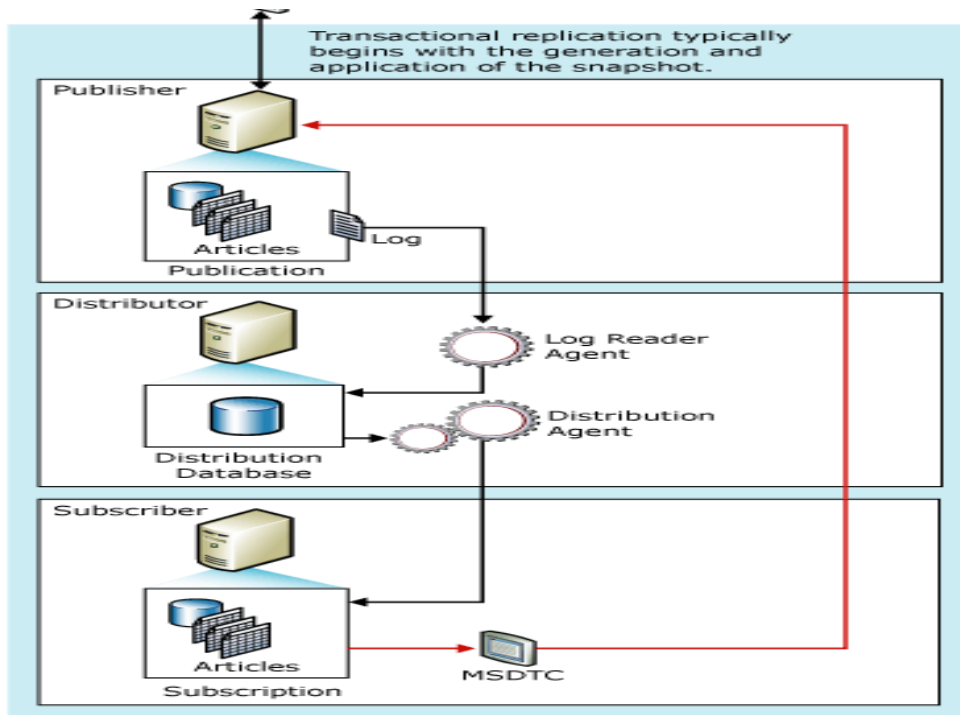
Trigger @ subscriber:-

1. trg_MSsync_del_tab
2. trg_MSsync_ins_tab
3. trg_MSsync_upd_tab

Store procedures @ publisher:-

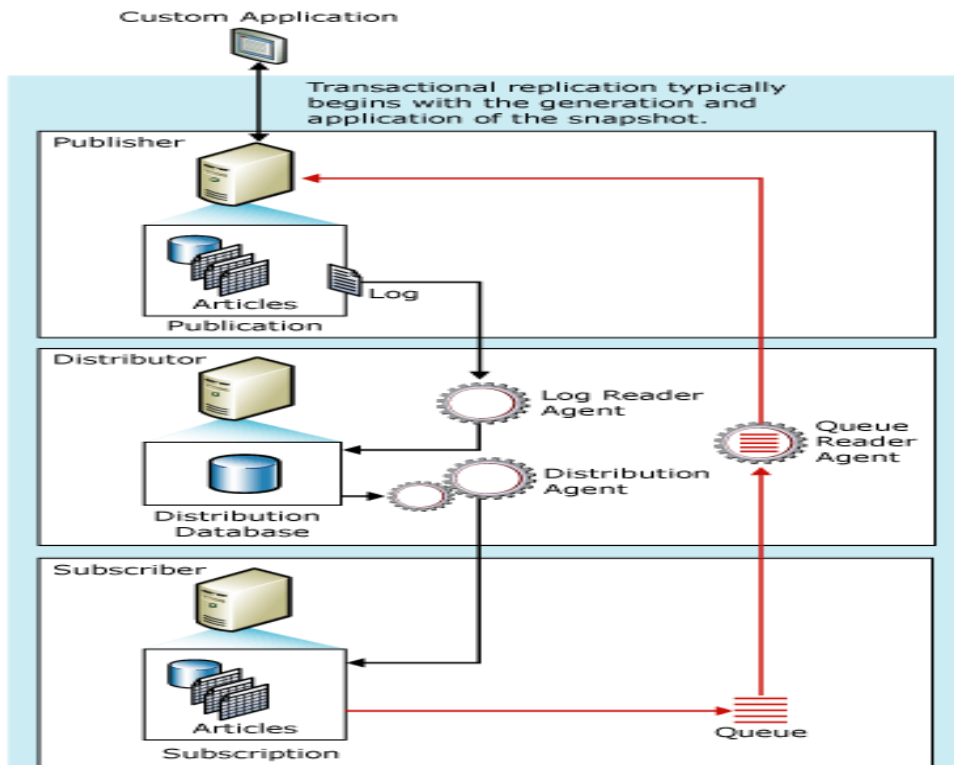
1. Dbo.sp_MSsync_del_tab_1
2. Dbo.sp_MSsync_ins_tab_1
3. Dbo.sp_MSsync_upd_tab_1

Case 1: IF MSDTC



1. A change made at the Subscriber is captured by a trigger on the subscribing table.
2. The trigger calls through MSDTC to the appropriate stored procedure at the Publisher.
3. The stored procedure performs the insert, update, or delete unless there is a conflict. If there is a conflict, the change is rolled back at the Publisher and the Subscriber.

Case: 2 IF QUEUE READER AGENT



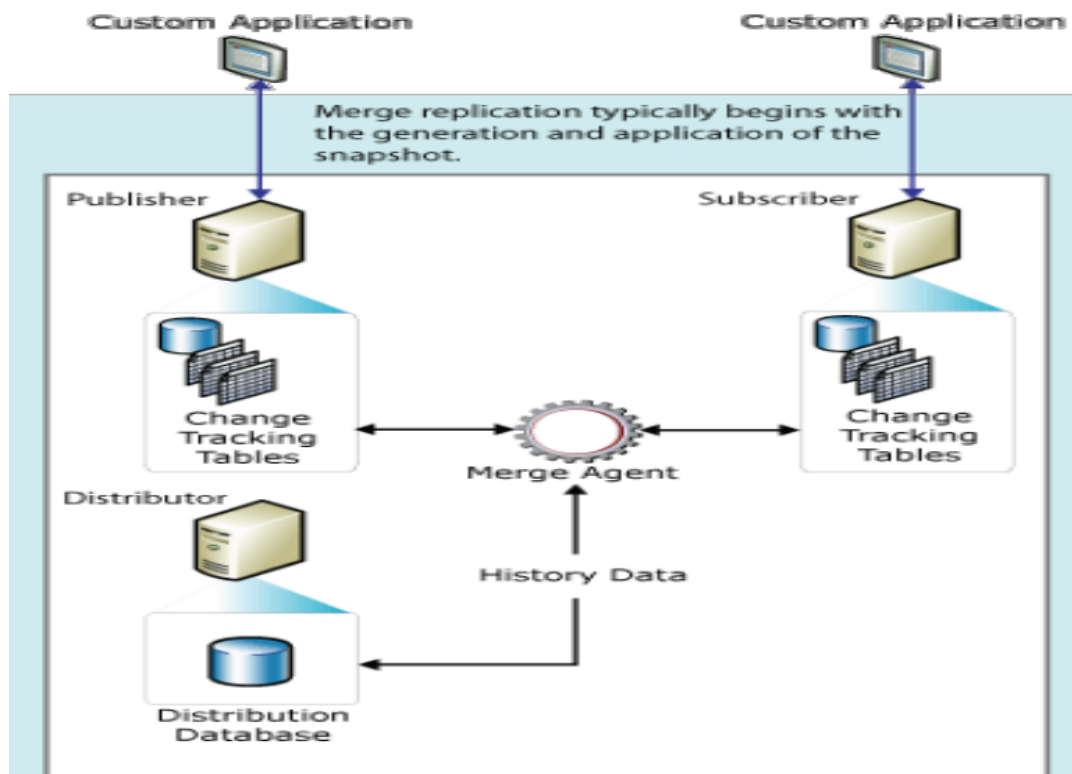
1. Updates made at the Subscriber are captured by triggers on the subscribing tables. The triggers store these updates in MSreplication_queue.
2. The Queue Reader Agent reads from MSreplication_queue, and then applies queued transactions to the appropriate publication using replication stored procedures.
3. While applying the queued transactions, conflicts (if any) are detected and resolved according to a conflict resolution policy that is set when the publication is created.
4. Changes made at the Publisher as a result of changes replicated from a Subscriber are propagated to all other Subscribers according to the Distribution Agent schedule.

Note: - ALWAYS FASTER AND COMMIT happen immediately in MSDTC type of data transfer.

But where as in Queued transaction, data not going to send to sub and first take same transaction into queue "MSREPLICATION_QUEUE" and then store. From there sent to publisher to "QUEUE reader agent"

4. MERGE REPLICATION [BI DIRECTIONAL]

1. Merge replication data can be read or write from any server site ...i.e. from publisher or either subscriber
2. In Merge replication data transfer between multiple server or sites by using agent called "MERGE AGENT " and place a major role.



Architecture Points:

1. Bi-directional replication is possible [update data in publication or subscriber and it will maintain same data to maintain sync.]
2. Rowguid-column [unique identifier column] will help to avoid conflict detection in both publication and subscription.
3. Only one publisher and able to configure multiple subscribers.

4. Primary key is not required. Merge agent collect or distribute all changes at publisher and subscribers then collect all changes at all the locations[publisher, subscriber] and do processing then distribute to all add maintain same data at all locations

Data Flow Points:

Merge replication, like transactional replication, typically starts with a snapshot of the publication database objects and data. Subsequent data changes and schema modifications Publisher and Subscribers are tracked with triggers.

The Subscriber synchronizes with the Publisher when connected to the network and exchanges all rows that have changed and Subscriber since the last time synchronization occurred.

Merge replication is typically used in server-to-client environments. Merge replication is appropriate in any of the following situations:

1. Multiple Subscribers might update the same data at various times and propagate those changes to the Publisher and to other Subscribers.
2. Subscribers need to receive data, make changes offline, and later synchronize changes with the Publisher and other Subscribers.
3. Each Subscriber requires a different partition of data.
4. Conflicts might occur and, when they do, you need the ability to detect and resolve them.
5. The application requires net data change rather than access to intermediate data states. For example, if a row changes five times at a Subscriber before it synchronizes row will change only once at the Publisher to reflect the net data change (that is, the fifth value).

Merge Replication Agents:

1. **Merge agent:** - Entire data processing taken by this agent
2. **Snapshot agent:** - First time initial schema and data
3. **Distributor agent:** - No job for distributor agent (idle).

CONFIGURATION STEPS:**1. First configure distribution database:-****Note:**

1. In merge replication; distribution agent or server job is idle except storing snapshot history at distribution database.
2. Merge agent collect all data from different sites and store into distribution database for processing of data .Once done removes from distributor database and re send to other servers.

2. Configure publisher:

Note: - ROW GUID COLUMN gets added to publisher tables and same replicate to multiple subscribers.

Changes:-

Note: - Below triggers and Store procedures gets created in all servers included as a part of merge replication.

@ Publisher and @ Subscriber

Store Procedures:

dbo.MSmerge_del_sp_80FDDC450D814A6F4A371E97BB9A4ECC

dbo.MSmerge_ins_sp_80FDDC450D814A6F4A371E97BB9A4ECC

dbo.MSmerge_upd_sp_80FDDC450D814A6F4A371E97BB9A4ECC

dbo.MSmerge_sel_sp_80FDDC450D814A6F4A371E97BB9A4ECC

Triggers:

MSmerge_del_EDE22551187F43A2A520C328D819CF51

MSmerge_ins_EDE22551187F43A2A520C328D819CF51

MSmerge_upd_EDE22551187F43A2A520C328D819CF51

Tables:-From these tables merge agent read and gets\collect the data changes.

Merge Tables at both publisher and subscriber:

dbo.MSmerge_contents: Store any insert or update performed at pub or subscriber

dbo.MSmerge_tombstone: Store any delete operation information at pub or subscriber.

dbo. MSmerge_genhistory: Contains one row for each generation. A generation is a collection of changes that is delivered to a publisher or subscriber. Generations are closed each time the Merge Agent runs, Subsequent changes in an are added to one or more open generations.

How Merge Replication Detects and Resolves Conflicts

The Merge Agent detects conflicts by using the lineage column of the **MSmerge_contents system table**.

> If column-level tracking is enabled for an article, the COLV1 column is also used. These columns contain metadata about when a row or column is inserted or updated, and about which nodes in a merge replication topology made changes to the row or column.

Conflict table: dbo.MSmerge_conflicts_info [Insert or update or delete].

sp_showrowreplicainfo (Transact-SQL):

Displays information about a row in a table that is being used as an article in merge replication

> As the Merge Agent enumerates changes to be applied during synchronization, it compares the metadata for each row at the Publisher and Subscriber.

5. PEER-TO-PEER REPLICATION [BI DIRECTIONAL]

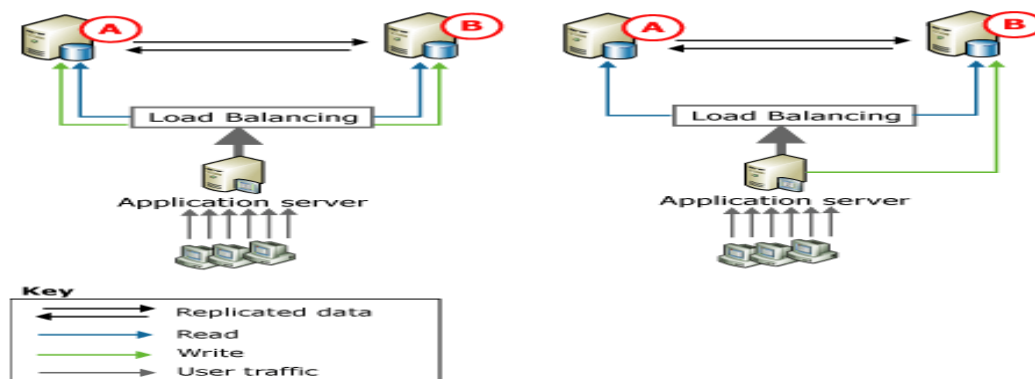
> Peer-to-peer replication provides a scale-out and high-availability solution by maintaining copies of data across multiple server instances, also referred to as *nodes*. Built on the foundation of transactional replication, peer-to-peer replication propagates transnationally consistent changes in near real-time.

> This enables applications that require scale-out of read operations to distribute the reads from clients across multiple nodes. Because data is maintained across the nodes in near real-time, peer-to-peer replication provides data redundancy, which increases the availability of data.

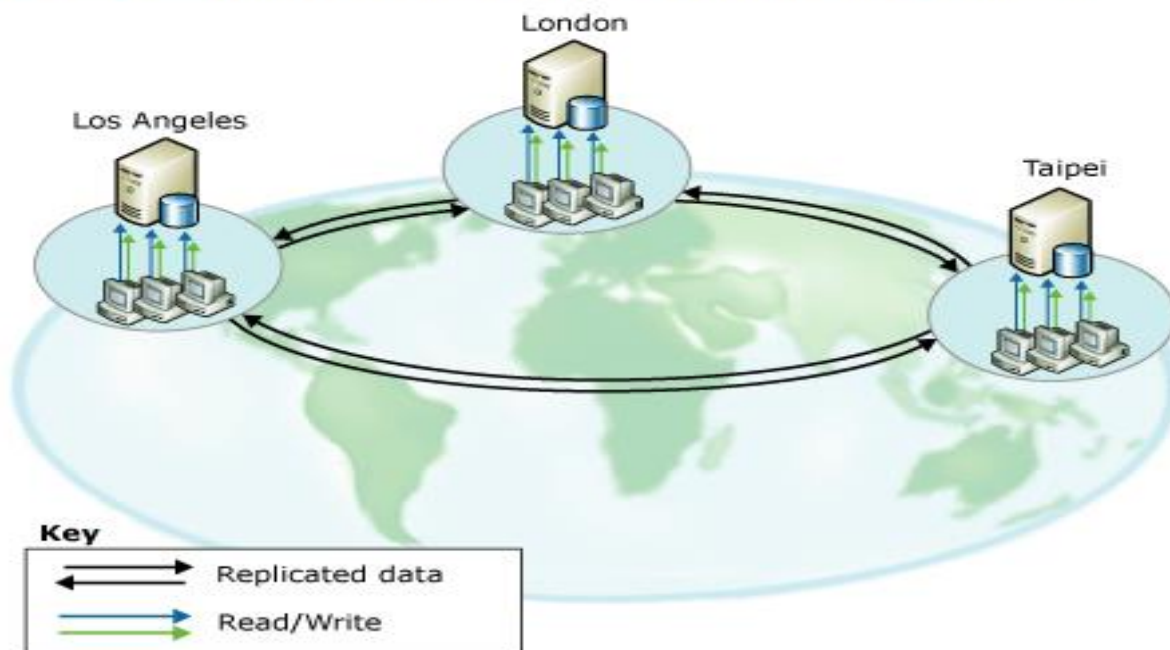
MINIMUM REQUIREMENTS:

- > Enterprise edition of SQL Server should be and minimum 2005 version...
- > Table schema and table name should be identical between all the nodes.
- > No dependency with recovery model
- > SQL Server agent should be up and running

Topology That Has Two Participating Databases



Topologies That Have Three or More Participating Databases



> Every node should perform 3 roles

1. Publisher, 2.Subscriber, 3.Distributor

For the purpose to avoid single point of failure.

2 node peer to peer replication: [2 log reader+2 dist Agent+1 snapshot]

1 pub+2 dist+2 sub

3 Node peer to peer replication: [3 log reader+3 dist Agent+1 snapshot]

3 pub+3 dist+6 subscriber (Each node 2)

4 node peer to peer replication: [4 log reader+4 dist Agent+1 snapshot]

4 pub+4 dist+ 12 subscribers

CONFIGURATION STEPS:

1. Configure distribution database in all the nodes
2. Take database full backup and restore into all other nodes with recovery
3. Start configuring local publication @node1

For 2005\8\8R2:

At the time of local publication configuring need to select "TRANSACTIONAL REPLICATION" due to PEER TO PEER only added in 2012 in GUI configuration

4. Right click > properties on local publication> enable "PEER TO PEER REPLICATION TO TRUE"
5. Start configuring peer to peer replication topology
6. Once done, automatically publication and subscriptions gets created in all the nodes.
7. Verify data is gets distributed between all the nodes.

Feature Restrictions

- > Peer-to-peer replication supports the core features of transactional replication,
- > But does not support the following options:
 - > Initialization and reinitialization with a snapshot.
 - > Row and column filters.
 - > Non-SQL Server Publishers and Subscribers.
 - > Immediate updating and queued updating subscriptions.
 - > Shared Distribution Agents.

Peer-to-Peer Replication Advantages & Disadvantages:**Advantages:**

1. Table level of high availability
2. When business having multiple branches this HA replication is good.
3. SQL Server agent based to distribute the data
4. Immediate synchronization @ other sites depending on the type of replication
5. No recovery model dependency

Disadvantages:

1. Troubleshooting is difficult
2. Agent should always up and running
3. No automatic failover method but the table in sub online state.
4. Application team need to point their application manually to subscriber.
5. End users also should point their connection to sub instance manually...i.e. no auto client redirect method support like mirroring

REPLICATION SCENARIOS:

1. HOW MANY AGENTS ARE IN EACH TYPE OF REPLICATION?

Snapshot Replication: Snapshot agent, distributor agent

Transactional Replication: Snapshot agent, distributor agent, Log reader agent

Transaction with updatable subscription: Snapshot agent, distributor agent, log reader, queue reader agent.

Merge Replication: Snapshot agent, distributor agent, merge agent

Peer to Peer Replication: Snapshot agent, distributor agent, and log reader agent [Per each peer or node]

2. HOW TO ADD AND DELETE ARTICLE IN EXISTING REPLICATION? WHAT IS DIFF BETWEEN 2000, 2005 AND 2008 ADDING ARTICLE PROCESS?

Adding article

1. Take full database backup for publisher database
2. Take the scripts in publisher and subscriber for create and drop scripts.[Purpose: like a backup of existing replication configuration]
3. After go to publisher >>>Publisher database properties>>and article>>and select that article.
4. Renationalize the snap shot again from replication monitor: to replicate newly added article from publisher to subscriber

Through query:

SP_addartical [sp used to add article]

Example: EXEC SP_addartical

```
@publication = Pub_dbAmericasCitrixFarm,
@article = Table_2,
@source object = Table_2,
```

Removal of article:

1. Take full database backup for publisher database
2. Take the scripts in publisher and subscriber for create and drop scripts.
3. After go to publisher >>>Publisher database properties>>>and article>>>and uncheck that article.
4. Renationalize the snap shot again from replication monitor.

Example:**EXEC sp_dropsubscription**

```
@publication = Pub_dbAmericasCitrixFarm,
@article = N'Table_2',
@subscriber = 'FTDCCWPCTRXSQL';
```

3. SCHEMA CHANGES IF WE DO ANY IMPACT TO SUBSCRIBER?

If we perform any schema changes to the existing replicated article, then we need to re initialize snapshot from publisher one more time.

4. REMOVAL OF REPLICATION?

Method 1:

Go to the publication >replication> "Disable publishing and distribution" option>click next and finish

Automatically delete distribution database. After go to local publication delete publication and subscription from local subscriber.

Method 2:

Exec sp_droppublication 'publication name'

Exec sp_dropsubscription'subscriptoon name'

5. DISTRIBUTION DATABASE REMOVAL AND IS IT POSSIBLE TO TAKE DISTRIBUTION DATABASE OFFLINE?

Method 1: Go to distribution server>right click> disable publishing and distribution> remove distribution database.

Method 2: Remove dependent publisher and subscriber

After fire **drop database distribution** (at distribution server)

6. DMV'S [DYNAMIC MANAGEMENT VIEW] USED FOR REPLICATION:

1. Sys.dm_repl_articles: Returns information about database objects published as articles in a replication topology.

2. Sys.dm_repl_schemas: Returns information about table columns published by replication.

3. Sys.dm_repl_tranhash: Returns information about transactions being replicated in a transactional publication.

4. Sys.dm_repl_traninfo: Returns information on each replicated or change data capture transaction.

7. HOW TO PERFORM MONITORING IN REPLICATION?

Undistributed commands: which displays how many commands or queries which are waiting to move from distribution database to subscriber. It will also displays how much take to replicate those commands from distributor to subscriber.

Monitoring Areas: Replication monitor and replication Maintaince jobs

Note: Verify \check the history whether data is moving from "Publisher to distributor" or "Distributor to subscriber" and check "undistributed command"

Note: Perform regular index maintenance, update stats, reindex, on Replication system tables just like you do for user tables.

MSmerge_contents

MSmerge_genhistory

MSmerge_tombstone

MSmerge_current_partition_mappings

MSmerge_past_partition_mappings

8. WHAT ARE THE SP AND TRIGGERS ARE USED IN REPLICATION?

Merge Replication:

Store Procedures:

dbo.MSmerge_del_sp_80FDDC450D814A6F4A371E97BB9A4ECC

dbo.MSmerge_ins_sp_80FDDC450D814A6F4A371E97BB9A4ECC

dbo.MSmerge_upd_sp_80FDDC450D814A6F4A371E97BB9A4ECC

dbo.MSmerge_sel_sp_80FDDC450D814A6F4A371E97BB9A4ECC

Triggers:

MSmerge_del_EDE22551187F43A2A520C328D819CF51

MSmerge_ins_EDE22551187F43A2A520C328D819CF51

MSmerge_upd_EDE22551187F43A2A520C328D819CF51

Transactional Replication With Updatable:**Trigger @ subscriber:-**

1. Trg_MSsync_del_tab
2. Trg_MSsync_ins_tab
3. Trg_MSsync_upd_tab

Store procedures @ publisher:-

1. Dbo.sp_MSsync_del_tab_1
2. Dbo.sp_MSsync_ins_tab_1
3. Dbo.sp_MSsync_upd_tab_1

9. IF I CHANGE RECOVERY MODEL THEN IS THERE ANY IMPACT?

> There will be no impact if we change recovery models due to replication support all type of recovery models.

10. DIFF BETWEEN PUSH AND PULL?

- > Distributor agent resided in distribution server - Push type
- > Distributor agent resided in subscriber server- Pull Type

11. CAN WE MODIFY DISTRIBUTION DATABASE DATA?

Yes we can modify distribution database but it effect replication. Not recommended

12. TRUNCATE TABLE IN PUBLISHER WILL IT'S EFFECT TO SUBSCRIBER?

> Truncate is a non-logged operation and it will not replicate automatically from publisher to subscriber

> In subscriber, need to perform truncate operation again to sync both pub and sub.

13. ERROR NUMBERS?

14. PRIMARY KEY VIOLATION?

PRIMARY KEY VIOLATION:

Error messages:

Violation of PRIMARY KEY constraint 'PK_tablename'. Cannot insert duplicate key in object 'dbo.tablename'. (Source: MSSQLServer, **Error number: 2627**)

Impact: This type of error generally appears in Transaction replication type.

Replication sync gets fail if this error reported. data\transaction sends from publisher to distributor. But from distributor to subscriber it fails and report the error in transactional replication monitor.

Where to find:

> Go to replication monitor> All subscriptions column > Status column> double click> check the information from publisher to distributor and dist> sub

> You can see error of primary key violation in dist to sub history.

q

From the error:

Transaction sequence no

Command ID:

Solution:

1. First find what transaction is inserted into subscriber which is causing this error by using below commands and execute only on distribution database.

EXEC Distribution. Sp_browsereplcmds

```
@xact_seqno_start = '0x00000018000000A1000300000000',
```

```
@xact_seqno_end = '0x00000018000000A1000300000000',
```

```
@command_id =1, @publisher_database_id = 1
```

2. COMMAND with transaction: {CALL [sp_MSins_dboetab] (N'3 ', N'c ', N'ap ')} }

3. Delete manually from subscriber and monitor for some time.

4. Now replication is working with excellent performance with no latency value.

15. SHOULD I SCRIPT MY REPLICATION CONFIGURATION?

Yes. Scripting the replication configuration is a key part of any disaster recovery plan for a replication topology

16. WHY DOES REPLICATION ADD A COLUMN TO REPLICATED TABLES; WILL IT BE REMOVED IF THE TABLE ISN'T PUBLISHED?

> Merge replication adds the column **rowguid** to every table, unless the table already has a column of data type **unique identifier** with the **ROWGUIDCOL** property set (in which case this column is used). If the table is dropped from the publication, the **rowguid** column is removed; if an existing column was used for tracking, the column is not removed.

Rowguid column: added in merge

MSrepl_tran_version: transactional updatable subscription

17. CAN MULTIPLE PUBLICATIONS USE THE SAME DISTRIBUTION DATABASE?

> Yes. There are no restrictions on the number or types of publications that can use the same distribution database. All publications from a given Publisher must use the same Distributor and distribution database.

18. DOES REPLICATION ENCRYPT DATA?

> No. Replication does not encrypt data that is stored in the database or transferred over the network

19. WHY CAN'T I RUN TRUNCATE TABLE ON A PUBLISHED TABLE?

> TRUNCATE TABLE is a non-logged operation that does not fire trigger

20. IF I CHANGE RECOVERY MODEL THEN IS THERE ANY IMPACT?

No

21. HOW SNAPSHOT WILL BE TAKEN?

> In SQL Server replication data transfer happen by using method of “BCP [BULK COPY PROGRAM] - for data, sch –for schema, Ind-for indexes move”

22. CAN WE MODIFY DISTRIBUTION DATABASE DATA?

> Yes we can modify the distribution database tables but not recommended...it will go to create impact.

23. SCHEMA CHANGES IF WE DO ANY IMPACT TO SUBSCRIBER?

Steps:

> After schema changes in publisher article then we require to reinitialize the snapshot agent to generate new snapshot

> Go to subscriber and then renationalize to apply new snapshot into subscriber article.

Note1: In peer –to-peer replication automatically schema change replicate to subscriber without snapshot agent run.

Note2: If we run snapshot then only modified data changes replicate to subscriber.

Snapshot Replication	Transactional Replication	Transactional replication with updatable subscription	Merge Replication	Peer-2-Peer replication
Snapshot and Distribution agents	Snapshot , Distribution agents and Log reader agent	Snapshot , Distribution agents, Queue Reader agent and Log reader agent	Snapshot , Distribution agents and Merge agent	Snapshot , Distribution agents and Log reader agent (each per one node)
one way replication	one way replication - use transactional	BI- Directional	BI- Directional	BI- Directional
Only one publishers are allowed	Only one publishers are allowed	Only one publishers are allowed	Only one publishers are allowed	Multiple publishers are allowed
Multiple subscribers are allowed	Multiple subscribers are allowed Both updateable modes of transactional replication have limited scalability (up to 10 subscribers)	Multiple subscribers are allowed Both updateable modes of transactional replication have limited scalability (up to 10 subscribers)	Multiple subscribers are allowed	Multiple subscribers are allowed
Enterprise limit: 25	Enterprise limit: 25	Enterprise limit: 25	Enterprise limit: 25	Enterprise limit: 25
Not required[conflict detection]	Not required	Updatable replication has rich conflict detection and handling capabilities as well.	Merge replication has rich conflict detection and handling capabilities as well.	Peer 2 peer replication has rich conflict detection and handling capabilities as well from 2008.
Primary key not required for article	Primary Key mandatory	Primary Key mandatory	Primary key not required for article	Primary Key mandatory

Note: All schema changes will replicate automatically without reinitiate snapshot agent from SQL Server 2005

Basically with merge replication when a synchronization occurs, the final state of the rows is what is merged with the other side. So if I have a stock tracking table which each stock is updated thousands of times between synchronizations only the last value of the stock will be replicated.

Essentially when it comes down to it, merge replication is infinitely scalable, merge replication can tolerate the bulk of the DML occurring at either the publisher or subscriber, it has rich conflict handling functionality.

To add article required to generate snapshot as mandatory

To add article required to generate snapshot as mandatory

To add article required to generate snapshot as mandatory

To add article required to generate snapshot as mandatory

To add article required to generate snapshot as mandatory

Merge replication

Is trigger based

It uses only merge agent

It could be set to track column changes instead of row changes

It could optimize the final changes produced at synchronization time.

Merge was designed for loosely connected systems that occasionally synchronize their data

Peer to Peer replication

Is Transaction Log based

It needs log Reader + Distribution agent

It creates at least one procedure call per row affected

It is only available on EE

Peer-to-Peer was simply an extension of bi-directional transactional replication with low latency in mind. It is

useful in redundant server-to-server data propagation.

Diff between Trans updatable replication and merge replication

Transactional updatable: If we update row 1000 times then all 1000 modifications replicate to subscriber.

In merge replication, if we update row 1000 times then last modification only replicate to subscriber

Advantages of Replication

Load balancing: distributes query load among the servers

Offline Processing: manipulate data from your database on a machine that is not always connected to the network.

Redundancy: Allows you to build a fail-over database server that's ready to pick up the processing load at a moment's notice.

CLUSTERING

Clustering:

> Windows clustering: 2 or more independent systems (referred as nodes) working for same purpose which is to provide continuous high availability to the instance

Note:

1. Cluster is a windows feature but it's not a Sql feature
2. Cluster feature is enabled for most of the O\S like win servers 2000, 2003, 2008, 2008 R2, 2012, 2014

Types of windows clusters:

1. NETWORK LOAD BALANCER [NLB]--APPLICATION LEVEL
2. COMPOENET LOAD BALANCER [CLB]
3. MICROSOFT CLUSTER SERVICE [MSCS]-UPTO WINDOWS 2003 VERSION

Note: IN WINDOWS 2008 ONWARDS NAMEING CONVENTION HAS CHANGED TO "FAILOVER CLUSTER"

4. GEO CLUSTER
5. VERITAS CLUSTER

1. NETWORK LOAD BALANCER [NLB] -- APPLICATION LEVEL: FRONT END

1. Network Load Balancing acts as a front-end cluster, distributing incoming IP traffic across a cluster of servers.
2. Up to 32 computers running a member of the Windows Server 2003 family can be connected to share a single virtual IP address.
3. NLB enhances scalability by distributing its client requests across multiple servers within the cluster.
4. As traffic increases, additional servers can be added to the cluster; up to 32 servers are possible in any one cluster.

5. NLB also provides high availability by automatically detecting the failure of a server and repartitioning client traffic among the remaining servers within 10 seconds, while it provides users with continuous service.

2. COMPOENET LOAD BALANCER [CLB]: WEBSERVERS: MIDDLE WARE SERVERS.

1. Component Load Balancing distributes workload across multiple servers running a site's business logic.
2. It provides for dynamic balancing of COM+ components across a set of up to 8 identical servers
3. CLB complements by acting on the middle tier service.

3. MSCS [MICROSOFT CLUSTER SERVICES] \ FAILOVER CLUSTER:

Failover Clustering:

Cluster Service acts as a back-end cluster; it provides high availability for applications such as databases, messaging and file and print services.

MSCS attempts to minimize the effect of failure on the system as any node (a server in the cluster) fails or is taken offline.

MSCS failover capability is achieved through redundancy across the multiple connected machines in the cluster, each with independent failure states.

The Windows Server 2003, supports up to 8 nodes in a cluster.

The Windows Server 2008, supports up to 16 nodes in a cluster.

Hardware Information:

Processor - Minimum of I5 - 1.90 GHz 2 cores and 4 Logical processors

Installed memory (RAM): 4.00 GB

Hard disk: 500GB

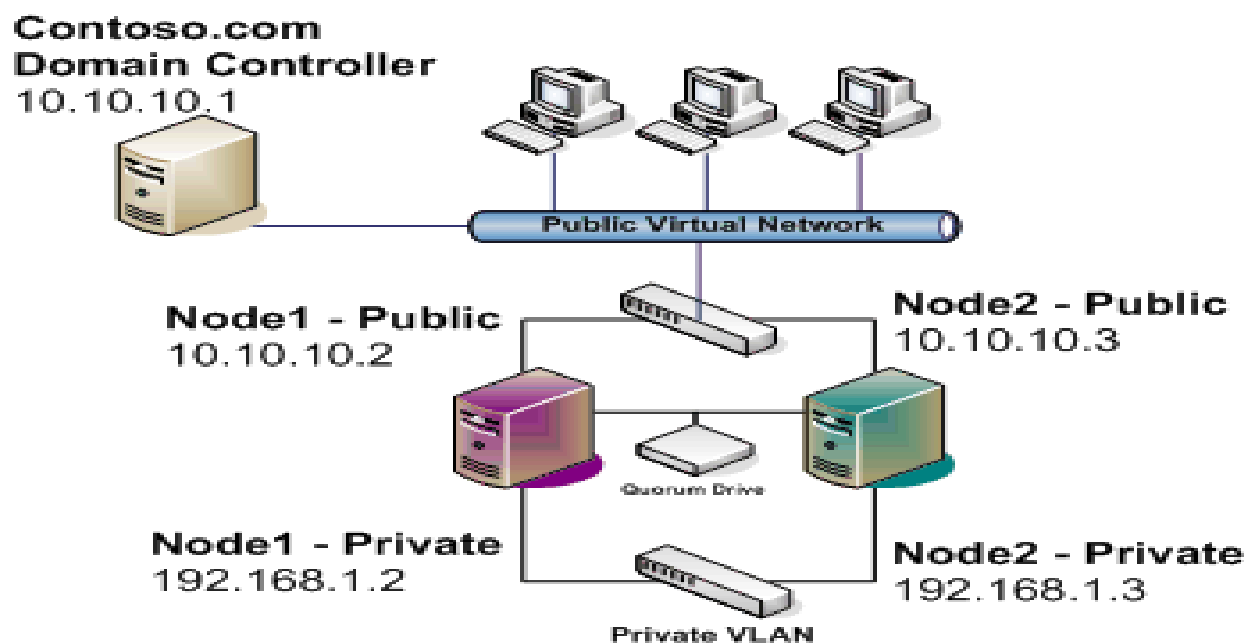
System type: 64-bit Operating System

Operating System: Windows 7 Home Premium

Required Software:

- 1) VMWare Player (free trial)
- 2) Windows Server 2012 R2 (with trial license free for 180 days)
- 3) Starwind Software (with free trial for 30 days)
- 4) Three virtual machines: one setup as Domain Controller and DNS Server (DC), the other two as clustered nodes (WIN1 and WIN2).

How Clustering Works:



q

Cluster Configuration:

1. Ready 3 virtual machines

DC

Win1

Win2

DC Server configuration:

EmkayDC IP: 192. 168.1.1

Gateway IP: 192. 168.1.1

DNS IP: 192. 168.1.1

2. Login to Domain controller and configure IP address

3. Change the computer name and do restart

4. Enable .netframework 3.5 SP1.

Go to command prompt and enter this:

Dism /online /enable-feature /feature name: NetFX3 /all /Source: d:\sources\sxs /LimitAccess

5. AD [Active Directory] Configuration

testcluster.com

6. Configure storage-STARWIND software to configure SCSI disks

Node1 Server configuration:

EmkayDC IP: 192. 168.102.128

Gateway IP: 192. 168.102.2

q

DNS IP: 192. 168.102.128

2. Login to Node1 and configure IP address (Both Public as well as Private)
3. Change the computer name and do restart
4. Enable .netframework 3.5 SP1.
5. Add Extra nic card for internal node1 and node2 communication.
6. Add node to domain.
7. Initialize virtual disks.

Node2 Server configuration:

EmkayDC IP: 192. 168.102.129

Gateway IP: 192. 168.102.2

DNS IP: 192. 168.102.128

2. Login to Node2 and configure IP address (Both Public as well as Private)
3. Change the computer name and do restart
4. Enable .netframework 3.5 SP1.
5. Add Extra nic card for Internal node1 and node2 communication.
6. Add node to domain.
7. Initialize virtual disks.

After above steps please run validation steps:

Validation Report Contains:

1. Cluster configuration
2. Inventory
3. Storage
4. Network

5. System configuration

Note: Validation cluster is a new feature in Windows cluster 2008 onwards...

How we can validate in Windows o\s 2003 or lower versions????

Manual validate is required after windows cluster installation.

1. Check all ip's are working and pinging between all nodes.
2. Check MSDTC is configured
3. Check quorum drive is configured
4. Check storage disks are configured.
5. Cluster service and cluadmin should be configured.

OVERALL WINDOWS CLUSTER IPADDRESS FOR 2 NODE CLUSTER: Total 7 for only windows

- > 1 PUBLIC NODE1+ 1 PUBIC NODE 2
- > 1 PRIVATE NODE 1+ 1 PRIVATE NODE 2
- > 1 DOMAIN CONTROLLER IP
- > 1 Windows cluster IP
- > 1 MSDTC ip address

Windows team handover to DBA team after windows cluster installation for Sql server installation.

DBA Team checks after windows cluster installation:

SQL Server Cluster Installation Steps.

SCSI (Small Computer System Interface):

- SCSI is a faster, more robust technology
- Aside from speed, another great advantage the SCSI card can connect 15 or more devices in a daisy chain.
- Flexibility towards expanding any system.

QUORUM Overview:

A quorum is the cluster's configuration database.

The database resides in a file named \MSCS\quolog.log. The quorum is sometimes also referred to as the quorum log.

Two main very important jobs for quorum:

1. It tells the cluster which node should be active
2. Which node or nodes are in standby?
3. Stores cluster configuration in quorum.

Types of QUORUM:

1. Standard Quorum
2. MNS [Majority node set]

1. Standard Quorum:

> This types generally used in WINDOWS SERVER 2003 O\S, which help to configure windows cluster.

> If Standard quorum is down, then entire windows cluster not works until you reconfigure the quorum...Why..? Due to cluster service always read node information from quorum \MSCS\quolog.log only.

> If quorum is not available then failover stuff not going to happen.

2. MNS [Majority node set] quorum:

This type of quorum is introduced newly in WINDOWS SERVER 2008 O\S.

Note: If quorum disk goes down in Wind 2008\R2\2012 still my cluster work due to maintains a local copy of "quolog.log" in each node. Due to this my cluster service can read from local copy of each MNS quorum .This makes always my cluster to run even quorum down in wind 2008 onwards.

Public IP: Useful for end user connectivity to nodes

Private IP: Internal node communication

Cluster IP: Required for Windows cluster

MSDTC IP: Required for MSDTC

Heart Beat: Very important component in cluster which always sends UDP packs between nodes for every 1.2 sec.

If packets fails for 5 times then the cluster services will initiate failover of your applications.

Cluster groups:

2 Types: Windows group +SQL group

Windows group: MSDTC

SQL Server group: Services and application

Types of Sql cluster:

Upto windows 2003 OS:

ACTIVE-PASSIVE

ACTIVE-ACTIVE

SINGLE NODE CLUSTER [Feature may add another node in same cluster]

Windows 2008 Os onwards:

SINGLE INSTANCE [ACTIVE-PASSIVE]:

> Single instance is a new terminology that started from SQL 2008 onwards. A single instance cluster has only one instance of SQL Server, which is owned by one node and all other nodes are in wait state till the failover occurs.

MULTIPLE INSTANCE [ACTIVE-ACTIVE]:

> Multiple instance replaces the term active/active.

N+1 [N: NO OF NODES ACTIVE +1: PASSIVE]:

> This is equalent to multiple instance cluster, where for example if we have 4 nodes. Each node contains one instance hosted (i.e. 3 instances hosted on three nodes) and the 4th node is maintained as a standby waiting for failover. N+1 means N nodes are hosted with SQL Server instances and 1 node is in waiting state.

N+M [N: NO OF NODES ACTIVE+ M:\ MULTIPLE PASSIVE]:

> This is also a multiple instance cluster where N nodes are hosted with SQL Server instances and M nodes are in waiting state.

SQL Server Services that can be clustered are:

- **SQL Server Main Service, Agent Service and Analysis Services.**

Cluster Service:

- The cluster service manages all the activity that is specific to the cluster. One instance of the cluster service runs on each node in the cluster. The cluster service does the following
- Manages Cluster Objects and Configurations
- Manages the local restart policy
- Coordinates with other instances of the cluster service in the cluster

q

- Handles event notification
- Facilitates communication among other software components
- Performs failover operations

Resource:

A resource is a physical or logical entity, which has below properties:

- Can be bought online and taken offline
- Can be managed in the failover cluster
- Can be owned by only one node at a time

RESOURCES IN CLUSTER:

Disks

QUORUM

MSDTC0

PUBLIC IP

PRIVATE IP

SQL SEREVR SERVICES

WINDOWS CLUSTER NAME

WINDOWS CLUSTER SERVICES

MSDTC: is used by SQL Server and other applications when they want to make a distributed transaction between more than one machines. A distributed transaction is simple a transactions which spans between two or more machines. The basic concept is that machine 1 starts a transaction, and does some work. It then connects to machine 2 and does some work. The work on machine 2 fails, and is canceled. The work on machine 1 needs to then be rolled back.

RESOURCE STATE:

All resources can have following states

Offline

Offline_Pending

Online

Online_Pending

Failed

CLUSTERING TERMS:**Cluster Nodes:**

- A cluster node is a server within a cluster group. A cluster node can be Active or it can be Passive as per SQL Server Instance installation.

Heartbeat:

- Heartbeats are single User Datagram Protocol (UDP) packets exchanged between nodes once every 1.2 seconds to confirm that each node is still available. If a node is absent for five consecutive heartbeats, the node that detected the absence initiates a regroup event to make sure that all nodes reach agreement on the list of nodes that remain available.

Private Network:

- The Private Network is available among cluster nodes only. Every node will have a Private Network IP address, which can be ping from one node to another. This is to check the heartbeat between two nodes.

Public Network:

- The Public Network is available for external connections. Every node will have a Public Network IP address, which can be connected from any client within the network.

Shared Cluster Disk Array:

- A shared disk array is a collection of storage disks that is being accessed by the cluster. This could be SAN or SCSI RAID.
- Windows Clustering supports shared nothing disk arrays. Any one node can own a disk resource at any given time. All other nodes will not be allowed to access it until they own the resource (Ownership change occurs during failover).

Quorum Drive

- This is a physical drive assigned on the shared disk array specifically for Windows Clustering. Clustering services write constantly on this drive about the state of the cluster. Corruption or failure of this drive can fail the entire cluster setup.

Cluster Name

- This name refers to Virtual Cluster Name, not the physical node names or the Virtual SQL Server names. It is assigned to the cluster as a whole.

Cluster IP Address

- This IP address refers to the address which all external connections use to reach to the active cluster node.

Cluster Administrator Account

- This account must be configured at the domain level, with administrator privileges on all nodes within the cluster group. This account is used to administer the failover cluster.

Cluster Resource Types

- This includes any services, software, or hardware that can be configured within a cluster. Ex: Generic Application, Service, Internet Protocol, Network Name, Physical Disk.

Cluster Group

- Conceptually, a cluster group is a collection of logically grouped cluster resources. It may contain cluster-aware application services, such as SQL Server 2000, 2005, 2008.

SQL Server Network Name (Virtual Name)

- This is the SQL Server Instance name that all client applications will use to connect to the SQL Server.

SQL Server IP Address (Virtual IP Address)

- This refers to the TCP/IP address that all client applications will use to connect to SQL +Server; the Virtual Server IP address.

SQL Server 2000 Full-text

- Each SQL Virtual Server has one full-text resource.

Microsoft Distributed Transaction Coordinator (MS DTC)

- Certain SQL Server Components require MS DTC to be up and running. MS DTC is shared for all named / default instances in cluster group.

SQL Server Virtual Server Administrator Account

- This is the SQL Server service account, and it must follow all the rules that apply to SQL Service user accounts in a non-clustered environment.

PRE-REQUISITES FOR SQL SERVER CLUSTER INSTALLATION [2005]

1. Copy SQL Software into number of nodes.
2. Start SQL Server servers from active node always.
3. If 2 node: 29 configuration checks for both the nodes..
4. Component selection

Note: Should select "CREATE SQL SERVER FAOlover CLUSTER"

5. VIRTUAL SERVER NAME [Extra screen in cluster setup]--Application and end user connectivity to SQL Server
6. VIRTUAL SERVER IP Address [Extra screen in cluster setup]--Application and end user connectivity to SQL Server
7. Cluster Node configuration
8. Domain Administrator account and password
9. Domain user and group account should require for SQL Server services.
10. Collation, Errors and reports...etc. as like standalone installation

SQL SERVER CLUSTER INSTALLATION [2008\R2\2012]

1. SQL Server network name
2. SQL Server Network IP
3. Cluster disk selection
4. Cluster network configuration
5. Cluster security policy

Note: INSTALLATION DIFF BETWEEN SQL 2005 AND 2008\R2\12:

SQL Server cluster installation:**SQL Server 2005 extra screens in cluster instance:**

1. Windows cluster mandatory along manual checks [Wind 2003]
2. Virtual name
3. Virtual ip
4. Cluster group selection
5. Domain administrator group
6. 3 domain level groups
7. Cluster network configuration
8. 29 configuration checks

SQL Server 2008\R2\12 extra screens in cluster instance:

- > Network name
- > Network ip
- > Domain level service accounts
- > 14 per node configuration checks.

DIFFERENCES BETWEEN SQL 2005 CLUSTER AND SQL 2008\8 R2\12\14 CLUSTER:

SQL 2005:

1. Always start Sql server installation from active node
2. Start servers+tools in active node but only SERVER replicate automatically into passive node [DATABASE ENGINE SERVICE+ANALYSIS SERVICE+SQL AGENT+SQL FULL TEXT]-Cluster aware services
3. First Sql installation complete in passive node and then complete in active node only cluster aware service or components.
4. After manually install tools+Reporting+integration services in passive node.

Note: Automatically installation replicate by using domain level admin account and password.

Extra screens:

Virtual name

Virtual ip

Cluster group selection

Configuration checks [29 if 2 node cluster]

Domain level admin groups

Domain admin account and password

Cluster disk selection

SQL Server 2008\8 R2\12\14:

1. Always start Sql server installation from active node
2. Start server and tools in Active node but no changes are replicate in passive node.
Only create binary file in active node locally in C:\ drive
3. Once active node installation completed, then manually go to passive node > re run the setup >"ADD NODE" option > then again install servers+tools

q

Again one more set of binary files gets create under node 2 as well.

Extra screens:

Network name

Network ip

Cluster group selection

Configuration checks [14 checks]

Cluster disk selection

Note:

Cluster AWARE components:

1. Database engine services

-SQL Server main service

-SQL Server agent service

-SQL Server full text [Option from 2008 onwards]

2. SQL Server analysis

-SQL Server analysis service

3. Cluster unaware services:

Tools:

Reporting services

Integration services

Notification services.

Note: Only we can see cluster aware services from cluadmin but not unaware services...

CLUSTER SENEARIOS:

1. FAILOVER AND FAILBACK IN CLUSTER:

Moving SQL Server resource from active node to passive node called "FAILOVER".

Manual Failover of SQL Server in cluster:

2 Ways we have

Method 1:

Go to cluadmin.msc> go to services and applications>go to Sql server > right click> move application or services from NODE AAA TO NODE BBB... or best possible node or select node.

> Then automatically all the SQL Server resources will failover to another node.

What will failover?

First Sql services will take offline

Shared disks will take offline in current node.

Automatically cluster resource take bring online in another node in below process.

First get shared disk online

Bring Sql server services online

2. Automatic failover: If node is down then automatically SQL and disks resources move to another node.

Note: When we perform any kind of failover we need to have instance level downtime.

i.e. Sql server services take offline in current node and after comes online into another node

Resource Failover and resource Sequence:

Active node:

1. SQL Server services offline
2. Disks going to offline
3. Cluster network name offline

Passive node:

1. Disks online first
2. Cluster network name
3. SQL server services online.

2. HOW TO RESTART SQL SERVER SERVICES WITH IN THE SAME NODE?

Go to service in cluadmin> right click>

BRING THE RESOURCE ONLINE

TAKE THE RESOURCE OFFLINE

3. CLUSTER MONITORING?

Always do any activities from only CLUSTER ADMINISTRATOR

Ex: Checking Sql server SERVICES, FAILOVER, FAILBACK, OFFLINE, ONLINE OF SQL SERVER RESOURCE... etc.

4. IS ALIVE AND LOOK ALIVE:**Cluster Monitoring:**

Look Alive check: (called as Basic resource health check) verifies that SQL Server is running on the current node.

By default it checks every 5 seconds.

If Look Alive check fails Windows Cluster performs an Is Alive check.

SELECT @@SERVERNAME every 5 seconds.

Is Alive check : (called as Thorough resource health check) runs every 60 seconds and verifies instance is up and running or not using the command in Resource DLL called SELECT @@SERVERNAME every 60 seconds.

q

Resources: Shared disks,

Network ip's [public, private]

Quorum

SQL Services [Main service, agent service. etc.]

Network ip and network name...

MSDTC

Note: How quorum drive update with your cluster resources?

IS\LOOK ALIVE RUNS> use administrator account> writes into quorum> cluster service reads from quorum> required trigger automatic failover of apps.

Note: QUORUM always resides in ACTIVE NODE.

5. PATCH MANAGEMENT PROCESS:

How to apply service packs in SQL Server cluster:

SQL SERVER 2005:

1. ALWAYS Apply the service pack for tools in passive node
2. Restart passive node
3. Once passive node up, please go to active node start applying service pack for server +tools
Automatically same service pack replicate to node 2 servers (passive node).

Note: Until service pack complete, Sql server databases are not accessible to application and downtime is more.

> If service pack fail in active node then it automatically fails in passive node.

> SP always apply to node 1 binary files and node 2 binary files automatically

SQL SERVER 2008 \8 R2\12\14:

1. Always apply the service pack in passive node for Server +tool binary files. Meanwhile my application still access the active node database and can run the business.
2. Once patch completed in passive node reboot the node.
3. Once node up, go to active node > Perform manual failover of SQL resource to passive node.
4. Then apply service pack in previous active node for servers+tools binary files
5. After reboot the previous active node.

Note:

- > Very less down time required for any Maintaince activities
- > Risk is very less...in terms of if sp is failed in passive node no impact to another node and can still my application run the business.

Note: FROM Active NODE DO the failover of SQL Server resources to passive node.

Now the current active node become passive node.

Start patch in node 1 as well for both servers and tools.

Restart the node

Check whether build number is changed or not.

Note: From Sql server 2008 with minimal down time (only for restart) can apply the patch.

ACTIVE-ACTIVE:

> If it is active -active node then need to perform by manual failover to any node and make one node as passive node .Then apply patch for passive after failback again to active node.

1. Failover multiple instance to node1 and now node 2 become passive node.
2. Apply patch for my passive node, for both instance binary files.
3. After do failback of both instance from current active to passive.
4. Apply patch to node 1 passive for both instance binary files.

Note: Restart node by node when it is required.

5. PREFERRED NODE SET:

Go to cluadmin> Services and Applications> Right click > check nodes> apply

Automatic Failover is always based on the preferred node concept

> Preferred nodes are an attribute of a resource group which is outside of SQL Server and is a Windows Resource.

> You can always find out which node owns the resource by using

SERVERPROPERTY ('ComputerNamePhysicalNetBIOS')

6. NODE CONFIGURATION STANDARDS:

Always when you set memory;

Example:

Node 1: SBI Application db: 4 GB of memory [1 GB O\S+ 3GB SQL]

Node 2: ICICI APP DB: 4 GB of Memory [1 GB O\S+ 3GB SQL]

Case 1: If node 1 is down????

Then my node2 need to bare the load of both NODE 1 SBI+NODE2 ICICI instances.

End users may face the performance impact>How to avoid this case?

Solution: node2 set 8 GB of memory [3 GB ICICI INST+ 3 GB SBI INST+ 2 GB O\S]

Your application gives good performance even 2 instances are running in single node.

7. HOW TO BYPASS RESTART COMPUTER POLICY?

Go to registry> hkey_local_mechine> system> current control set> control> session manager> "PENDING FILE RENAME OPERATIONS"> Open file and clear the data...

Rerun the checks again and skips restart policy.

Note: Registry can be backup and restore by using IMPORT \EXPORT methods at windows level

8. HOW TO READ CLUSTER EVENTS:

1. Windows 2003 O\S:

C:\windows\cluster\clusterlog.txt

2. Windows 2008 O\S onwards:

Go to cluadmin> cluster events to check history of cluster.

9. HOW TO ADD THE DEPENDENCY DISK?

> After adding windows team or san team disk into disk management. DBA team check whether disk is showing in my computer or not.

> Now DBA team need to make the added disk as a dependency to SQL Server service.

> Add storage to the SQL Server group and then perform below step.

IN SQL Server 2005:

Note: Adding dependency disk is completely offline operation i.e. need to take SQL Server services offline...Downtime is required

Path to add dependency disk:

Go to cluster administrator> go to Sql server main server> go to properties>go to dependency tab> modify> add disk...> click ok

IN SQL Server 2008:

Note: Adding dependency disk is completely online operation i.e. No need to take SQL Server service offline or downtime is not required.

Path to add dependency disk:

Go to cluster administrator> go to Sql server main server> go to properties>go to dependency tab> modify> add disk...> click ok

10. DIFFERENCE BETWEEN WINDOWS SERVER 2003\2005, WINDOWS SERVER 2008\R2\SQL 2008\R2, WINDOWS SERVER 2012\SQL SERVER 2012

Points	Windows server 20003\SQL 2005	Windows server 2008,2008 r2,\SQL 2008, R2	Windows server 2012
Domain level Groups	3 domain level groups are Should be mandatory.	Optional because we can use local accounts to install SQL Server	Optional because we can use local accounts to install SQL Server
Quorum	Drive letter should be Q	Optional	Optional
MSDTC	MSDTC is mandatory	MSDTC is optional because it uses local dtc services from wind 2008 server o\s	MSDTC is optional because it uses local dtc services from wind 2008 server o\s
Tempdb files	Should keep in shared or iscsi disks	Should keep in shared or iscsi disks	Optional...we can keep temp db files in local disks for each node.
Installation	Always required to start instance from active node installation completed automatically first in passive node then it gets finish in active node. Only required to perform tools installation in passive node...	Start installation from active node but it will not replicate to passive node .Required to run setup.exe in passive node and click add node option "to install in passive node.	Start installation from active node but it will not replicate to passive node .Required to run setup.exe in passive node and click add node option "to install in passive node.
Cluster Events	Monitoring : c:\windows\cluster\cluster.log file need to check if any event which need to check in cluster	But whereas from wind 2008 onwards, cluster events option is included in cluster administrator.	But whereas from wind 2008 onwards, cluster events option is included in cluster administrator.
Quorum down	If quorum goes down , then our entire cluster will be down	But even quorum down, it uses MNS [Majority node set quorum] of each local copy to store information relate to which node is active and passive.	But even quorum down, it uses MNS [Majority node set quorum] of each local copy to store information relate to which node is active and passive.
Naming	Cluster services	Failover cluster	failover cluster
Cluadmin	Cluadmin	cluadmin.msc	cluadmin.msc
Console	All services,nodes,ips all together in cluadmin in wind 2003	But in Windows 2008 onwards, classification with nodes, ip's, disks, services and applications.	But in Windows 2008 onwards , classification with nodes, ip's, disks, services and applications.

Validation report	Manual checks need to perform before SQL Server installation on cluster	Run validation report to validate windows cluster setup.	Run validation report to validate windows cluster setup.
Number of nodes	8	16	16
Dependency disk	Downtime required. Need to restart SQL Server services	Online operation. NO downtime required	Online operation. NO downtime required
Policies	In SQL Server 2005 cluster, if any resource went to offline, we require manually to bring online.	But in SQL Server 2008 Onwards if any resource offline then with in default 15 min of interval cluster try to start the resource into online. ...	But in SQL Server 2008 Onwards if any resource offline then with in default 15 min of interval cluster try to start the resource into online. ...

PERFORMANCE TUNING

LOCKS

Locks:

- > To hold a specific object (tables, database, pages, rows, instance, extent, key.....etc) in SQL Server by using this locking concept.
- > To provide consistence data or right data or correct data to the end user
- > Sql Server cannot lock the resources.

Note: Lock internally managed by lock manager and takes the decision depend on the transaction what lock to be applied.

LOCK RESOURCES:

ROW LEVEL: Row identifier used to lock a single row with in a table

PAGE LEVEL: 8 kilo bytes (kb) data pages or index pages

EXTENT LEVEL: Contiguous group of eight data pages or index pages

TABLE LEVEL: Entire table including all data index

DATABASE LEVEL: Database

KEY LEVEL: row lock with in an index used to product key ranges in serializable transaction

How to find locks:

SP_LOCK

OR

SELECT * FROM SYS.DM_TRAN_LOCKS

q

Output:

Resource type [Database or Page or Object or Row or Extent or table]

Request mode [Lock type]

Request type

Request status [Grant or Wait]

Request Session id

How will u find out which session is doing what work?

SP_WHO2

TYPES OF LOCKS:

1. Shared lock[S]: Multiple users can able to read the data on specific resource. No transaction or query need to wait.

> When transaction starts internally lock manager applies shared lock and once reading completed lock revoked automatically.

2. Exclusive Lock[X]: When we perform any insert, update and delete operations then exclusive lock (X) placed on resource.

> Always lock manager gives the priority to DML operations compare to any select queries.

3. Update Lock [U]: Whenever we perform any update operations then update lock placed in SQL Server.

> Update lock calls most of the time exclusive lock (X) by lock manager.

4. Schema Lock (SCH-L): When performing any locks at schema table then lock manager raise Schema level lock.

5. Bulk Update [BU]: Bulk update lock generally placed by lock manager when there are any bulk transactions.

Ex: Insert into, bulk into, select into

6. Intent lock: Intended to apply desired lock on particular lock.

2 Types:

1. Intent Shared [IS] -- Intended to read the data

2. Intent Exclusive [IX] -- Intended to write the data

3. Shared with Intent Exclusive [IS] --

Lock Escalation: Reducing the number of locks by using this method.

> Instead of multiple row level locks better is table level lock. Which reduces number of locking types and improves the performance by escalating lock.

> Instead of multiple page level of locks better is database level lock.

> This decision of Escalation is taken by SQL Server Engine

> SQL Server supports escalating the locks to the table level. The locks can only be escalated from rows to the table or pages to the table level.

RID --> Pages --> Tables --> DB

Note: In Sql server can be maintain by lock manger users are DBA does not have any consoling locking system

BLOCKINGS

Blockings:

> Blocking occurs when one SPID holds a lock on a specific resource and a second SPID attempts to acquire a conflicting lock type on the same resource.

How to find blocking:

Method 1: SP_WHO2

Method 2: Select * from sys.sysprocesses where spid>50 and blocked<>0

Note: System processes >50 are never involved in a blocking part

Only user defined process (from SPID 51) participates in a blocking.

Method 3: Select * from sys.dm_os_waiting_tasks

Method 4: Go to database > right click> reports> standard reports> all blocking sessions

Method 5: By running the profiler > "ALL BLOCKED TRANSCATIONS"

Solution:

1. Check what is the spid causing the blocking in column "BLKBY" and find the queries which are relate to by using

DBCC INPUTBUFFER (SPID) --- Display upto 250 char max.

2. Share these spid and query information to apps guys and ask confirmation from their side to kill one of the SPID to resolve the blockings.

3. Once apps team confirms then we will proceed for killing the spid. Ensure apps team send email but not verbally.

4. Kill the session by using

KILL SPID

5. Ask apps team to check the query status and keep monitoring from DBA end.

If blocking runs what is the IMPACT?

- Application runs slowly and apps side queries take longer time
- If blockings run long time then it may down the Sql server. ASAP inform after monitoring to apps team and kill after confirmation.

How to avoid blockings in SQL SERVER:

Long-running queries.

Canceling queries without rollback

Changing large numbers of records in a single transaction

Lack of appropriate indexes

Note: BLOCKING information never stores in error log or event viewer. We have to capture when blockings are running.

Blocking:

Create table A

Insert some values

Open--54:

Begin Tran ----- Manual Commit

Insert into TableA values (1,'ad')

Open--56:

Select *from TableA

Open --58:

q

Select * from sys.sysprocesses where spid>50 and blocked<>0

(to find the blocking)

Dbcc inputbuffer (56)

Dbcc inputbuffer(54)

(to find which query is running blocked and blocked by session)

sp_lock

(to find which lock is applied)

to inform app team which query imp based on kill the session

DEAD LOCKS

Deadlock: A deadlock occurs when two or more tasks permanently block each other by each task having a lock on a resource which the other tasks are trying to lock.

ERROR: 1205 is the deadlock error number.

Error message: Transaction (Process ID 55) was deadlocked on lock resources with another process and has been chosen as the deadlock victim. Rerun the transaction.

Impact: If deadlock occurs it impact user queries\connections\application slowness\ and Sql server may down if long time deadlock runs

How to find?

Note: By default deadlock information never capture in SQL Server error log or event viewer or you cannot find by using any query or DMV's.

We have to enable trace or run the profiler to capture the victim transactions for deadlocks

Method: 1

Enable DBCC TRACEON (1222, 1204,-1) then after automatically SQL Server captures deadlock transaction into SQL Server errorlogs.

1222: Output in text formate

1204: Output in XML formate

-1: Global Trace it will check the all the data bases.

How to find how many Traces?

DBCC TRACESTATUS

Method 2:

To run the Profiler what permission need to require?

Sysadmin

Run the profiler by selecting the events

> LOCK: DEADLOCK CHAIN

> LOCK: DEADLOCK

> LOCK: DEADLOCK GRAPH

Once run then automatically deadlock transactions captured into profiler tool in a graph representation.

> Lock will be created in Buffer Pool

Solution: Deadlocks always kill or resolved by deadlock manager based on "DEADLOCK_PRIORITY" parameter gives which SPID need to kill.

DEADLOCK_PRIORITY: It provides which transaction need to kill on basis below factors.

> Which transaction takes less time to roll back?

> When transaction started?

> What type of statement which is?

Pass the final information to dead lock manager and kills victim transaction to fix deadlock.

How to minimize deadlocks:

1. Keep transactions as short as possible.
2. During transactions, don't allow any user input.
3. Avoid cursors.
4. Consider using the NOLOCK hint to prevent locking.

Select query always with no_lock

No_Lock: It will never keep shared lock

5. Access objects in the same order.

INDEXES

Index

> An index is a collection of pages associated with a table (or view) used to speed retrieval of rows from the table or enforce uniqueness

Index designed to improve the query performance of SQL Server

Note: Indexes store data\pages internally in a formate of B-tree structure.

Why Use an Index?

Use of SQL server indexes provide many facilities such as:

- Rapid access of information
- Efficient access of information
- Enforcement of uniqueness constraints

Types of indexes:

1. HEAP TABLE

A table without having any indexes called as HEAP table"

Heap table also called as "TABLE SCAN"

When Use a Heap:

> An index can be an unnecessary storage and maintenance overhead. (Eg: Audit Log)

> Using a table scan to find data can be quicker than maintaining and using an index. (Eg: less rows)

> Frequently rows are added, deleted, and updated. The overhead of index maintenance can be more costly than the benefits.

> Contain predominantly duplicate data rows. A table scan can be quicker than an index lookup.

2. CLUSTER INDEX:

1. A common analogy for a clustered index is a phone book

> Clustered indexes are the cornerstone of good database design.

> A poorly-chosen clustered index lead to high execution times and storage space.

> One clustered per Table. As only one physical way to store data.

Note:

> In cluster index data always stores in databases which are at leaf level.

> Index data always stores at root and intermediate levels where it contains only index pages.

Note:

> Primary Key by default creates a clustered Index. But whereas cluster index never create primary key on table.

If delete primary key cluster index deletes automatically.

How to find list of indexes?

Select * from Sys. Sysindexes

Columns:

Ind id

Index name

Cluster index syntax:

CREATE CLUSTERED INDEX [INDEXNAME] ON [dbo]. [TABLE NAME]

(

[COLUMN NAME]ASC

)

Note: Per table can create only one cluster index. The reason can arrange data physically in 1 way. Multiple physical ways are not possible.

How SQL store Clustered Index Table?

All table data are stored in Data Pages

Create Clustered Index issued?

Data pages are ordered by key

Index Pages are organized as B-Tree by index Key

> Each row in index page contain 2 information

1. Index Key Value

2. 6 byte page pointer [Address reference at bottom level in page for next page]

Limitations:

Which column can be Clustered key?

> Columns in WHERE clause

> To return a range of values by using operators (BETWEEN, >, >=, <, and <=). Once first row is found subsequent indexed values will be physically adjacent.

> Columns used in ORDER BY or GROUP BY clause.

> The data values in the rows are already sorted. This improves query performance.

> Columns used in JOIN

> Return large result sets.

> Columns used frequently in sort operation

> Data retrieved from a table already sorted. (Saves sort cost)

> Are frequently accessed sequentially.

Which columns can't be Clustered key?

- > The data in the indexed columns will change frequently.
- > Changes to a clustered index mean that the entire row of data must be moved to keep the data values of a row in physical order.

3. NON CLUSTER INDEX:

- > In non-cluster index data pages stores in storage location instead of leaf nodes
- > Only pointer address maintain in leaf level and index pages gets created in root, intermediate level and leaf level as well.
- > In this index type performance is not faster compare to cluster index due to multiple levels of search.

Per table:

SQL 2005 ONWORDS: 1 CLUSTER+249 NON CLUSTER INDEX MAX

SQL 2008 ONWARDS: 1 CLUSTER+999 NON CLUSTER INDEX MAX

Syntax:

```
CREATE nonCLUSTERED INDEX [INDEXNAME] ON [dbo].[TABLE NAME]
(
    [COLUMN NAME]ASC
)
```

DIFFERENCE BETWEEN CLUSTER & NON-CLUSTER:

1. Only one clustered index per table, where as you can have more than one non cluster index
2. Cluster index is faster than a non-clustered index, because, the clustered index has to refer back to the table, if the selected column is not present in the index.

3. Clustered index determines the storage order of rows in the table, and hence doesn't require additional disk space, but whereas a Non Clustered index is stored separately from the table, additional storage space is required.

3. UNIQUE INDEX

> This is the type of index where it did not allow duplicate values on created column.

> This type of index allow UNIQUE null value but not multiple null values.

Syntax:

```
CREATE UNIQUE NONCLUSTERED INDEX [INDEXNAME] ON [dbo]. [Table name]
```

```
(
```

```
    [Column name] ASC
```

```
)
```

Note: Can create unique index in cluster index as well .But there is no use as cluster index properties only applies for the tables

4. COMPOSITE INDEX

When you include more than a column at the time of index creation called as COMPOSITE index

Or

Can configure single index more than one column (Multiple) called as composite index

> Can configure cluster index on multiple columns.

> Max 16 columns can include in a part of 1 composite index.

Note: Reduces number of index creation and storage space. Improve performance as well.

5. COVERING INDEX

- > If the query include all the columns in a table with indexes called "COVERING index".
- > All columns should include as a part of the indexes if we miss one column then it became composite index.
- > Reduces DISK I\O contention and improve the query performance.

6. PARTITIONED INDEX: [SQL server 2005]

- > Introduced in SQL server 2005
- > Partitioned Index are created over Partitioned Table.
- > Partitioning can improve performance by distributing data across files, reduce contention, and increase parallel IO operation.
- > Depending on key value index formed on different file group.

Note: This feature Supported only for Enterprise/ Developer Edition

7. XML INDEX: [NEW 2005]

- > This type of index can use when any column data type as a "XML" type.
- > This is a new feature in SQL 2005
- > This type of index improve query performance 50 times faster when retrieving any XML data from the tables.

8. FILTERED INDEX: [NEW 2008]

- > Can this index used on row level on table included in portion.
- > It reduce storage cost, improve query performance, reduce the Maintaince cost and Maintaince is not required until any alter index performed.
- > Allow null values due to this only NON CLUSTER FILTERED INDEX Can create.

9. SPATIAL INDEX: [NEW 2008]

- > New from SQL Server 2008
- > Can be created only on spatial column (Geometric/ Geography)
- > Can create 249 spatial index on a spatial column supported table.
- > Creating more than 1 index on same column useful.
- > Location aware devices and services like GPS/ online mapping needs spatial data support to manage location aware data
- > Max size of index key = 895 bytes.

10. COLUMN STORE INDEX [SQL 2012 NEW FEATURE]

1. Expected- Query could run 100 times faster than ordinary.
 2. Vertipaq – a New Microsoft tech – different way to store columns and compress data in index.
 3. In regular index, all indexed data from each row kept in a single page. And columns of different rows spread across.
 4. In column store index, data from each column are kept together in same page.
 5. Less space enough to store index.
 6. Once u create a column store index the table becomes READONLY.
- > The SQL Server in-memory columns tore index stores and manages data by using column-based data storage and column-based query processing.
 - > Column store indexes work well for data warehousing workloads that primarily perform bulk loads and read-only queries.
 - > Use the column store index to achieve up to 10x query performance gains over traditional row-oriented storage, and up to 7x data compression over the uncompressed data size.

Reason For efficiency:

Column data is compressed.

Few pages. So can fit in Memory (buffer)

SQL needs to retrieve data only from index pages

Syntax:-CREATE COLUMNSTORE INDEX [index name]

ON my Table (col1, col2...)

Index Pub:

FILL FACTOR:

> It tells how much percentage of index data need to fill in index pages [ROOT and intermediate levels]

> A fill factor is reserved free space on each leaf level page which is used for future growth of data or index in table and reduces the page splits.

Assign Fill factor: right click on server> properties > database settings> default index fill factor> provide the value

By default fill factor value is: 0

Note: As per MS recommend fill factor always set 80% and rest of 20% used for index Maintaince.

PAD INDEX:

> This is depends on fill factor percentage value It tells how much percentage of data need to fill in data pages [Leaf levels]

> By default there is no value and it always depends on fill factor parameter.

Note: Without enabling or set fill factor pad index not possible to enable.

WHAT IS INDEX SEEK AND INDEX SCAN?

INDEX SCAN: It scans all the rows in table and can be considered as a table scan. Which takes long time generally to retrieve data.

INDEX SEEK: Only search for qualified rows in SQL Server and it is always faster than index scan.

> If huge insert\update\delete happens what happens to pages and indexes?

The pages get affected with fragmentation due to huge insert\update or delete

FRAGMENTATION

Fragmentation: pages can have empty rows called as fragmentation

> **How to find whether pages are fragmented or not?**

Upto SQL SERVER 2000 version: DBCC SHOWCONTIG

From Sql 2005 onwards: DMV's to find

Select * from sys.dm_db_index_physical_stats (DB_ID ('DB NAME'), Object_id ('tablename'), NULL, NULL, 'Detailed')

Column to verify the fragmentation value:

Avg_fragmentation_in_percent:

>0 to <5: Indexes are good and no fragmentation. No action required.

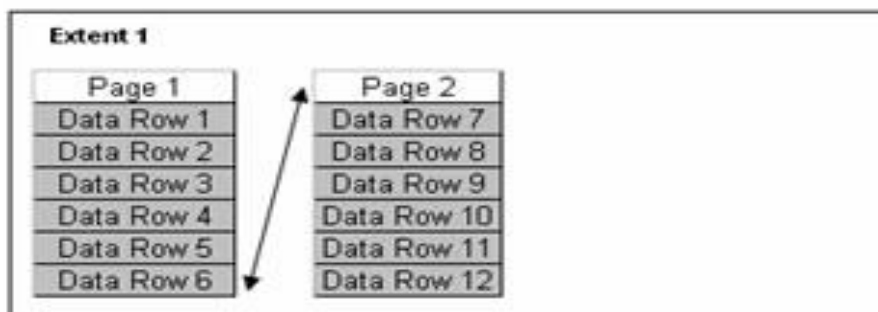
>5 and <30: Reorganize the index

>30: Rebuild the index

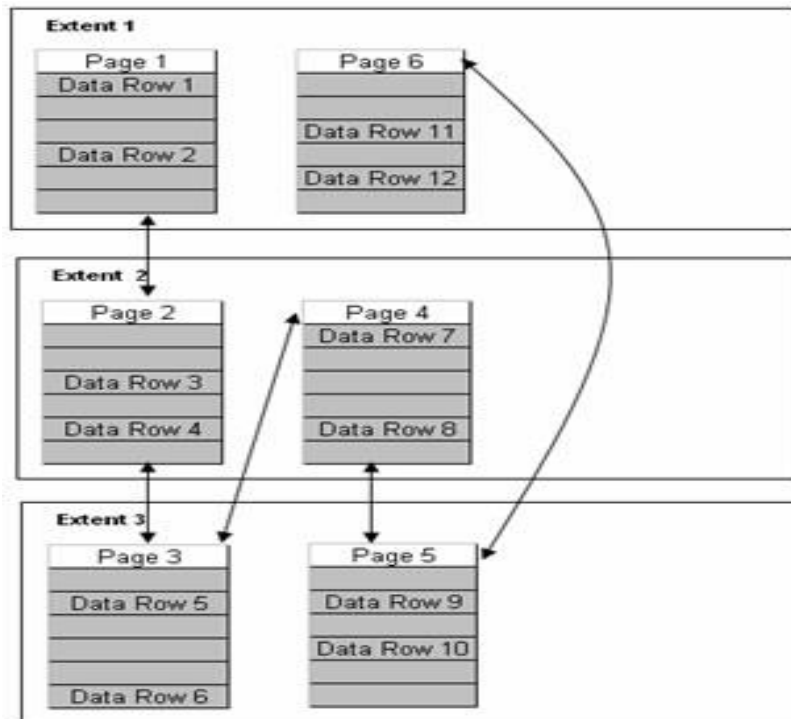
> **How reduce fragmentation?**

By either rebuild or reorganize the index.

Ideal non Fragmented Index :



Fragmented Index:



FRAGMENTATION TYPES

- 1. Internal Fragmentation:** Index pages takes more pages than needed. Data stored in pages in improper order.
- 2. External Fragmentation:** Occurs when pages are not continuous on Extent.

DIFF BETWEEN REBUILD AND REORGANIZE:

REORGANIZE: Internally the data gets rearranged inside the data pages and no effect to existing indexes.

Alter INDEX INDEXNAME ON TABELNAME REORGANIZE

REBUILD: Existing indexes dropped after recreating newly with same name and on same columns. At the time index recreation automatically data gets rearranged.

Alter INDEX INDEXNAME ON TABELNAME REBUILD

Note: When you are creating index\rebuild\reorganize the table loaded into TEMPDB system database and then perform the operations. If Tempdb disk does not have sufficient disk space then index may fail.

In real time index Maintainece is a weekly Maintainece.

Note: Upto SQL Server 2000 version rebuilding the index required down time due to table cannot be accessible to end users. [OFFLINE INDEXING]

From SQL 2005 supports bot offline and online index....

When rebuild index is running still table can be accessible by end users.

Note: Rebuild or reorganize always performs at Tempdb. 120% of free space should require depends on the table size.

If Tempdb drive space is less index fails

MISSING INDEX: Which leads poor performance to particular query.

How to find:

Select * from sys.dm_db_missing_index_details

UNUSED INDEXES: If indexes are not used they should be dropped as Indexes reduces the performance for INSERT/UPDATE statement. Indexes are only useful when used with SELECT statement.

To find whether any unused indexes in SQL Server can get by DMV.

Select * from sys.dm_db_index_usage_stats

Note: if any UN unused indexes in SQL Server then we cannot get good performance. To use indexes drop existing indexes and recreate the same indexes.

How to find SPID percentage completion or estimated time:

Select * from sys.dm_exec_requests

ISOLATION LEVELS

> Isolation levels in SQL Server control the way locking works between transactions.

> SQL Server supports the following isolation levels

1. READ UNCOMMITTED
2. READ COMMITTED [THE DEFAULT]
3. REPEATABLE READ
4. SERIALIZABLE
5. SNAPSHOT [SQL 2005 NEW FEATURE]

> Before I run through each of these in detail you may want to create a new database to run the examples, run the following script on the new database to create the sample data.

Note: You'll also want to drop the Isolation Tests table and re-run this script before each example to reset the data.

```
CREATE TABLE IsolationTests
```

```
(  
    Id INT IDENTITY,  
    Col1 INT,  
    Col2 INT,  
    Col3 INT  
)
```

```
INSERT INTO IsolationTests(Col1, Col2, Col3)
```

```
SELECT 1,2,3
```

```
UNION ALL SELECT 1,2,3
```

```
UNION ALL SELECT 1,2,3
```

```
UNION ALL SELECT 1,2,3
```

q

UNION ALL SELECT 1,2,3

UNION ALL SELECT 1,2,3

UNION ALL SELECT 1,2,3

Also before we go any further it is important to understand these two terms....

1. Dirty Reads – This is when you read uncommitted data, when doing this there is no guarantee that data read will ever be committed meaning the data could well be bad.

2. Phantom Reads – This is when data that you are working with has been changed by another transaction since you first read it in. This means subsequent reads of this data in the same transaction could well be different.

1. READ UNCOMMITTED

> This is the lowest isolation level there is. Read uncommitted causes no shared locks to be requested which allows you to read data that is currently being modified in other transactions. It also allows other transactions to modify data that you are reading.

> As you can probably imagine this can cause some unexpected results in a variety of different ways. For example data returned by the select could be in a half way state if an update was running in another transaction causing some of your rows to come back with the updated values and some not to.

> To see read uncommitted in action let's run Query1 in one tab of Management Studio and then quickly run Query2 in another tab before Query1 completes.

Query1

```
BEGIN TRAN
```

q

```
UPDATE Tests SET Col1 = 2
```

```
--Simulate having some intensive processing here with a wait
```

```
WAITFOR DELAY '00:00:10'
```

```
ROLLBACK
```

Query2

```
SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED
```

```
SELECT * FROM IsolationTests
```

> Notice that Query2 will not wait for Query1 to finish, also more importantly Query2 returns dirty data. Remember Query1 rolls back all its changes however Query2 has returned the data anyway, this is because it didn't wait for all the other transactions with exclusive locks on this data it just returned what was there at the time.

> There is a syntactic shortcut for querying data using the read uncommitted isolation level by using the NOLOCK table hint. You could change the above Query2 to look like this and it would do the exact same thing.

```
SELECT * FROM IsolationTests WITH(NOLOCK)
```

2. READ COMMITTED [THE DEFAULT]

> This is the default isolation level and means selects will only return committed data. Select statements will issue shared lock requests against data you're querying this causes you to wait

if another transaction already has an exclusive lock on that data. Once you have your shared lock any other transactions trying to modify that data will request an exclusive lock and be made to wait until your Read Committed transaction finishes.

> You can see an example of a read transaction waiting for a modify transaction to complete before returning the data by running the following Queries in separate tabs as you did with Read Uncommitted.

Query1

```
BEGIN TRAN
```

```
UPDATE Tests SET Col1 = 2
```

```
--Simulate having some intensive processing here with a wait
```

```
WAITFOR DELAY '00:00:10'
```

```
ROLLBACK
```

Query2

```
SELECT * FROM IsolationTests
```

> Notice how Query2 waited for the first transaction to complete before returning and also how the data returned is the data we started off with as Query1 did a rollback. The reason no isolation level was specified is because Read Committed is the default isolation level for SQL Server. If you want to check what isolation level you are running under you can run “**DBCC useroptions**”. Remember isolation levels are Connection/Transaction specific so different queries on the same database are often run under different isolation levels.

3. REPEATABLE READ

> This is similar to Read Committed but with the additional guarantee that if you issue the same select twice in a transaction you will get the same results both times. It does this by holding on to the shared locks it obtains on the records it reads until the end of the transaction, this means any transactions that try to modify these records are forced to wait for the read transaction to complete.

> As before run Query1 then while its running run Query2

Query1

```
SET TRANSACTION ISOLATION LEVEL REPEATABLE READ  
  
BEGIN TRAN  
  
SELECT * FROM IsolationTests  
  
WAITFOR DELAY '00:00:10'  
  
SELECT * FROM IsolationTests  
  
ROLLBACK
```

Query2

```
UPDATE IsolationTests SET Col1 = -1
```

> Notice that Query1 returns the same data for both selects even though you ran a query to modify the data before the second select ran. This is because the Update query was forced to wait for Query1 to finish due to the exclusive locks that were opened as you specified Repeatable Read.

q

> If you rerun the above Queries but change Query1 to Read Committed you will notice the two selects return different data and that Query2 does not wait for Query1 to finish.

Note:

In Repeatable read: Phantom reads are possible

SELECT – UPDATE: QUERY PRREFERENCE “SELECT”

SELECT –INSERT: QUERY PRREFERENCE “INSERT”

SELECT –DELETE: QUERY PRREFERENCE “SELECT”

> One last thing to know about Repeatable Read is that the data can change between 2 queries if more records are added. Repeatable Read guarantees records queried by a previous select will not be changed or deleted, it does not stop new records being inserted so it is still very possible to get Phantom Reads at this isolation level.

4. SERIALIZABLE

> This isolation level takes Repeatable Read and adds the guarantee that no new data will be added the chance of getting Phantom Reads. It does this by placing range locks on the queried data. This causes any other transactions trying to modify or insert data touched on by this transaction to wait until it has finished.

> You know the drill by now run these queries side by side...

Query1

```
SET TRANSACTION ISOLATION LEVEL SERIALIZABLE
```

```
BEGIN TRAN
```


q

```
SELECT * FROM IsolationTests
```

```
WAITFOR DELAY '00:00:10'
```

```
SELECT * FROM IsolationTests
```

```
ROLLBACK
```

Query2

```
INSERT INTO IsolationTests(Col1,Col2,Col3)
```

```
VALUES (100,100,100)
```

> You'll see that the insert in Query2 waits for Query1 to complete before it runs the chance of a phantom read. If you change the isolation level in Query1 to repeatable read, you'll see the insert no longer gets blocked and the two select statements in Query1 return a different amount of rows.

Serializable: Phantom reads are not possible due to transaction execution performs serially.

SELECT FIRST - INSERT NEXT: QUERY PREFERENCE "SELECT"

INSERT FIRST- SELECT NEXT: QUERY PREFERENCE "INSERT"

5. SNAPSHOT [SQL 2005 NEW FEATURE]

> Well it's more in the way it works, using snapshot doesn't block other Queries from inserting or updating the data touched by the snapshot transaction. Instead it creates its own little snapshot of the data being read at that time, if you then read that data again in the same transaction it reads it from its snapshot, This means that even if another transaction has made

q

changes you will always get the same results as you did the first time you read the data. To use the snapshot isolation level you need to enable it on the database by running the following command

```
ALTER DATABASE IsolationTests  
SET ALLOW_SNAPSHOT_ISOLATION ON
```

Note:

Advantages of Snapshot:

No blockings

No deadlocks

No query need to wait whether it is select or insert or update

Always end user gets committed data

Note: How it works?

> The table data gets mapped into space file in TEMPDB after set the isolation level as SNAPSHOT then what ever the SELECT queries reads from Tempdb sparse file and whatever inserts\updates\deletes performs on actual table in the user database

SWITCHES

/3GB SWITCH:

- > By default SQL Server can be able to utilize 2 GB max memory in 32-bit o/s.
- > If required to use SQL Server more than 2 GB [Max 3 GB] then recommended to enable /3GB switch in BOOT.INI file.

C:\BOOT.INI

- > Note pad will be opened and at last enable /3GB and save the notepad file. After your SQL Server can be able to consume 3 GB max.

Note: No server or Sql reboot is required if we enable \3GB switch

PAE [Physical addressing extension]:

Enable /PAE switch in OS level and

If RAM<4GB:

/3GB

If RAM > 4GB and <16GB:

/3GB /PAE

If RAM > 16GB:

/PAE

Note: After enabling PAE switch at o\s level SQL Server can go and utilize entire memory from O\S .Due to this chance of running Server out of memory which can leads to Server hang. Required to control or restrict or cap memory at SQL Server level by using SQL Server SWITCH "AWE"

AWE [Addressing windowing extension]:

> Enable AWE option in SQL Server level.

Note: As there is a risk involved with memory consumption from OS Mode, it is always recommended to set "MAX SERVER MEMORY" restriction.

Note: In 64-bit O/S no need to enable any switches to use memory by SQL Server applications.

Note:

> IN WINDOWS and SQL 32 BIT O\S Need to enable AWE & PAE switches to use more memory to SQL Server.

> In x 64 bit O\S no need to enable AWE OR PAE switch .Directly we can enable MIN AND MAX memory values at SQL Server level.

SQL 2005 or LOWER VERSIONS: After setting AWE with min and max memory required to restart ONLY SQL SERVER SERVICES.

SQL 2008 ONWARDS: No need to perform ant Sql server service restart after min and max memory.

MEMORY CAPPING: Nothing but restricting memory at SQL Server level.

32-bit	64-bit
Lesser Data Transfer Speeds upto 32bits	Faster (double) performance benefit, speeds upto 64bits
Maximum RAM supported is 4GB	Maximum RAM Supported is 7-8TB
Less memory registers allocated	Has more memory registers allocated
Reserves less space for OS reserved portions when compared to 64 bit	Reserves more space for OS reserved portions
Memory mapped files are more difficult to implement in 32 bit	It's easy to map memory mapped files
32 bit less expensive	64 bit expensive
	Offers special enhanced security features like Kernel Patch Protection, Support for hardware backup data execution protection
More compatible device drivers	Incompatible Device Drivers

LOCK PAGES IN MEMORY

> It is a Windows policy determines which accounts can use a process to keep data in physical memory, preventing the system from paging the data to virtual memory on disk.

Note: Always SQL Server gives good performance when read or write from memory instead of disk. To control IO from Disk better to add SQL Server service account need to add under Group policies.

GPEDIT.MSC -> Computer Configuration -> Windows Settings -> Security Settings -> Local Policies -> User Rights Assignment -> Lock Pages in Memory -> Add Service Account to use the Physical Memory.

> In 32bit it is used to extend memory access beyond 4 GB or to enable the AWE feature. For 64bit systems, it is to possibly gain performance and to “lock pages” for the buffer pool.

DAC (Dedicated Administration Connection)

- Microsoft SQL Server provides a dedicated administrator connection (DAC).
- The DAC allows an administrator to access any emergency issue a running instance of SQL Server Database Engine to troubleshoot problems on the server—even when the server is unresponsive to other client connections.
- The DAC is available through the Sql cmd utility and SQL Server Management Studio [Query analyzer].
- The connection is only allowed from a client running on the server.

How to enable DAC?

- By default DAC is enabled to connect through local connection. That means you can connect to the local instance of SQL server using DAC without making any changes.
- To connect the SQL server using DAC from remote machine using TCP/IP, we have to enable the 'remote admin connections' using sp_configure. We can enable the remote DAC connection using the below query.
- Exec sp_configure 'remote admin connections',1

Sp_configure: This procedure allows you to enable internal configurations of SQL Server .Eg, memory, DAC, Query timeout. Etc.

To use SP CONFIGURE used should have Sysadmin permissions.

Note1:

- By default SQL server listen to DAC connection on port number 1434. If the port number 1434 is not available, SQL server dynamically assign a port number during the start up and this can be found in the SQL server error log as given below.
- **DAC only allows you to connect query analyzer but not object explore SSMS.**
- To enable DAC you should be member of SYS Admin role.

>> Connect by using query analyzer:

Admin: host name ---Default instance

Admin: hostname\instance name ---Named instance

>> By SQLCMD:

C:\>sqlcmd -S localhost -U sa -P dev -d master -A [Default instance]

C:\>sqlcmd -S [named instance name] -U sa -P dev -d master -A [Named instance]

-U: User name

-P: Password

-A: DAC Connection.

Note: If the DAC is already in use, the connection will fail with an error indicating it cannot connect.

Error: 17810, Severity: 20, State: 2.

DAC Limitations:

- Only one DAC connection is allowed per instance. If a DAC connection is already open, new connection request will be denied with error 10053.
- You can't connect SSMS object explorer using the DAC connection, but you can connect a query analyser window.
- SQL server prohibits running parallel queries or commands on DAC connection. Error 3637 is generated if you try to perform a backup or restore operation.
- Only login with Sysadmin rights can establish the DAC connections.
- Cannot run parallel queries in SQL Server instance as only 1 DAC connection is allowed.
- SQL Server services are not possible to restart by using DAC query analyzer.
- SQL Server DAC possible only with EE edition
- DAC never uses port number 1433.

By default always DAC uses 1434 port number but if 1434 is not enabled in server then SQL uses dynamic port at the time of SQL Server start up.

- SA account password should be used while connecting from SQL CMD.

DAC at O/S Level:

- This method help to connect a dedicated admin connection to o\s by using mstsc
 \admin

Max any server RDP allow only 2 connections. If I need to connect for any emergency then should use above method without disconnecting any users.

CDC (CHANGE DATA CAPTURE)

>>What is Change Data capture? [CDC]

1. Microsoft SQL Server 2008 has introduced a very exciting feature for logging DML changes.
2. Change data capture provides information about DML changes on a table and a database.
3. Change data capture records insert, update, and delete activity, that's applied to a SQL Server table and makes a record available of what changed, where, and when, in simple relational 'change tables'.
4. Also stores historical data and COLUMN level changes in SQL Server by using CDC feature.
5. Change data capture is available only on the Enterprise, Developer, and Evaluation editions of SQL Server

>>>How it works>

1. The source of change data for change data capture is the SQL Server transaction log.
2. As inserts, updates, and deletes are applied to tracked source tables, entries that describe those changes are added to the log.
3. The log serves as input to the change data capture process. This reads the log and adds information about changes to the tracked table's associated change table.

>>Permissions required to configure CDC:

EITHER DB_OWNER OR SYSADMIN permissions

>>CDC Configuration steps:

1. Enable CDC on database by using

```
EXEC sys.sp_cdc_enable_db
```

@5 system tables gets created automatically.

q

[cdc].[captured_columns]

[cdc].[change_tables]

[cdc].[ddl_history]

[cdc].[index_columns]

[cdc].[lsn_time_mapping]

[dbo].[systranschemas]

2. Enable CDC on table by using

EXEC sys.sp_cdc_enable_table

@source_schema = N'dbo',

@source_name = N'MyTable',

@role_name = NULL

Note: Few CDC system table and 2 CDC jobs create automatically inside of the SQL Server databases

CDC Default Tables:

cdc.captured_columns: This table returns result for list of captured column.

cdc.change_tables: This table returns list of all the tables which are enabled for capture.

cdc.ddl_history: This table contains history of all the DDL changes since capture data enabled.

cdc.index_columns: This table contains indexes associated with change table.

cdc.lsn_time_mapping: This table maps LSN number (for which we will learn later) and time.

dbo.systranschemas:

q

After enabling CDC on table one more addition tracking table

Ex: CDC.DBO_STAB_CT

List of automatic jobs:

cdc.DBNAME_capture

cdc.DBNAME_cleanup

Findings:

Select * from CDC.DBO_STAB_CT

If the operation column shows value

1: Delete operation

2: Insert operation

3: Before update

4: After update

Along with this data gets captured into CDC defined table.

Note: Enable CDC only with confirmation from apps team or client... If you enable it consumes more hardware resource and additional storage is required.

WAITING TYPES

> When if any query waiting for resources then relevant wait type comes into picture. Which cause high performance impact.

How to find wait type:

Select * from sys.sysprocesses

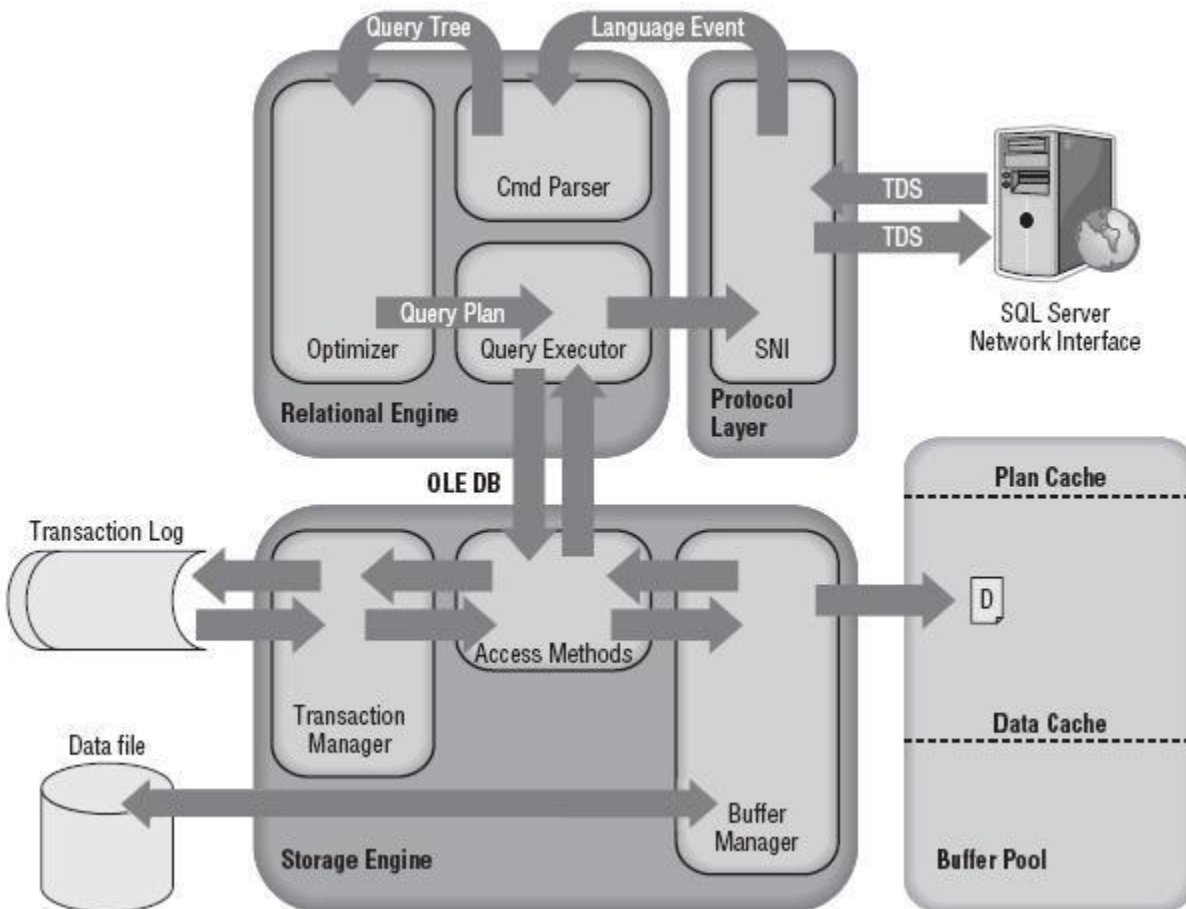
Column "last wait type"

Types of wait types:

1. **LCK_M_S:** Occurs when a task is waiting to acquire a shared lock. [Occurs mostly in blockings]
2. **ASYNC_IO_COMPLETION:** Occurs when a task is waiting for I/Os to finish.
3. **ASYNC_NETWORK_IO:** Occurs on network writes when the task is blocked behind the network. Verify that the client is processing data from the server.

SQL SERVER ARCHITECTURE

> SQL Server architecture is mainly divided into different components i.e. SNI Protocol Layer, Relational Engine, Storage Engine, Buffer Pool. Majorly classified as two main engines: Relational Engine and the Storage engine.



SNI (SQL Server Network Interface)

> The SQL Server Network Interface (SNI) is a protocol layer that establishes the network connection between the client and the server. It consists of a set of APIs that are used by both the database engine and the SQL Server Native Client (SNAC).

SQL Server has support for the following protocols:

► **Shared memory:** Simple and fast, shared memory is the default protocol used to connect from a client running on the same computer as SQL Server. It can only be used locally, has no configurable properties, and is always tried first when connecting from the local machine.

► **TCP/IP:** TCP/IP is the most commonly used access protocol for SQL Server. It enables you to connect to SQL Server by specifying an IP address and a port number. Typically, this happens automatically when you specify an instance to connect to. Your internal name resolution system resolves the hostname part of the instance name to an IP address, and either you connect to the default TCP port number 1433 for default instances or the SQL Browser service will find the right port for a named instance using UDP port 1434.

► **Named Pipes:** TCP/IP and Named Pipes are comparable protocols in the architectures in which they can be used. Named Pipes was developed for local area networks (LANs).

► **VIA:** Virtual Interface Adapter is a protocol that enables high-performance communications between two systems. It requires specialized hardware at both ends and a dedicated connection.

TDS (Tabular Data Stream) Endpoints

TDS is a Microsoft-proprietary protocol originally designed by Sybase that is used to interact with a database server. Once a connection has been made using a network protocol such as TCP/IP, a link is established to the relevant TDS endpoint that then acts as the communication point between the client and the server.

The Relational Engine:

The Relational Engine is also sometimes called the query processor because its primary function is query optimization and execution.

1. Command Parser:

The Command Parser's role is to handle T-SQL language events. It first checks the syntax and returns any errors back to the protocol layer to send to the client. If the syntax is valid, then the next step is to generate a query plan or find an existing plan. A Query plan contains the details about how SQL Server is going to execute a piece of code. It is commonly referred to as an execution plan.

Plan Cache: Creating execution plans can be time consuming and resource intensive, so The Plan Cache, part of SQL Server's buffer pool, is used to store execution plans in case they are needed later.

2. Query Optimizer:

The Query Optimizer is one of the most complex and secretive parts of the product. It is what's known as a "cost-based" optimizer, which means that it evaluates multiple ways to execute a query and then picks the method that it deems will have the lowest cost to execute. This "method" of executing is implemented as a query plan and is the output from the optimizer.

3. Query Executor:

The Query Executor's job is self-explanatory; it executes the query. To be more specific, it executes the query plan by working through each step it contains and interacting with the Storage Engine to retrieve or modify data.

The Storage Engine:

> The Storage engine is responsible for managing all I/O to the data, and contains the Access Methods code, which handles I/O requests for rows, indexes, pages, allocations and a Buffer Manager, which deals with SQL Server's main memory consumer, the buffer pool. It also contains a Transaction Manager, which handles the locking of data to maintain Isolation (ACID properties) and manages the transaction log.

1. Access Methods:

> Access Methods is a collection of code that provides the storage structures for data and indexes as well as the interface through which data is retrieved and modified. It contains all the code to retrieve data but it doesn't actually perform the operation itself; it passes the request to the Buffer Manager.

2. Buffer Manager:

> The Buffer Manager manages the buffer pool, which represents the majority of SQL Server's memory usage. If you need to read some rows from a page the Buffer Manager will check the data cache in the buffer pool to see if it already has the page cached in memory. If the page is already cached, then the results are passed back to the Access Methods. If the page isn't already in cache, then the Buffer Manager will get the page from the database on disk, put it in the data cache, and pass the results to the Access Methods.

Data Cache: The data cache is usually the largest part of the buffer pool; therefore, it's the largest memory consumer within SQL Server. It is here that every data page that is read from disk is written to before being used.

3. Transaction Manager:

> The Transaction Manager has two components that are of interest here: a Lock Manager and a Log Manager.

> The Lock Manager is responsible for providing concurrency to the data. The Access Methods code requests that the changes it wants to make are logged, and the Log Manager writes the changes to the transaction log. This is called Write-Ahead Logging.

Checkpoint Process:

> A checkpoint is a point in time created by the checkpoint process at which SQL Server can be sure that any committed transactions have had all their changes written to disk. This checkpoint then becomes the marker from which database recovery can start.

> The checkpoint process ensures that any dirty pages associated with a committed transaction will be flushed to disk. Unlike the lazy writer, however, a checkpoint does not remove the page from cache; it makes sure the dirty page is written to disk and then marks the cached page as clean in the page header.

Lazy writer:

> The lazy writer is a thread that periodically checks the size of the free buffer list. When it's low, it scans the whole data cache to age-out any pages that haven't been used for a while. If it finds any Dirty pages that haven't been used for a while, they are flushed to disk before being marked as free in memory.

Checkpoint occurs:

Manual checkpoint command fires

Auto check point

Backup command fired

Alter database command

SQL Server shutdown

Cluster failover

Snapshot is created

Commit is issues.

Add file and file group arch

Pages and extents

NDF, MDF and LDF arch

Lazy writer

Check point.

Dirty pages

Dirty data.

SQL PROFILER TOOL

Performance:

Reasons: Below 3 areas will impact your performance.

- **Disk Issues**
- **DB Design**
- **Bad written Queries**

<u>Performance issues</u>	<u>Reason</u>
80%	Poor Coded SQL Statements
15%	Bad Database Design
5%	Inadequate Hardware Configurations

What is the purpose of SQL Profiler in SQL server?

- SQL profiler is a tool to monitor performance of various stored procedures. It is used to debug the queries and procedures. Based on performance, it identifies the slow executing queries. Capture any problems by capturing the events on production environment so that they can be solved.

SQL Profiler captures SQL Server events from a server. The events are saved in a trace file that can be used to analyse and diagnose problem.

The different purposes of using SQL Profiler are:

- It is used to find the cause of the problem by stepping through problem queries.
- It is very useful to analyse the cause of slow running queries.
- It can be used to tune workload of the SQL server.
- It also stores security-related actions that can be reviewed by a security administrator.

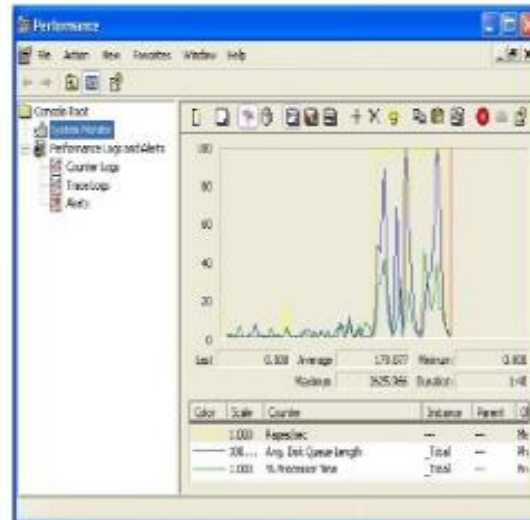
- SQL Profiler also supports auditing the actions performed on instances of SQL Server.

Use SQL Server Profiler:

- Monitor the performance of an instance of the SQL Server Database Engine, Analysis Server, or Integration Services (after they have occurred).
- Identify the cause of a deadlock
- Debug Transact-SQL statements and stored procedures.
- Audit SQL Server activity
- Monitoring T-SQL activity per user
- Analyze performance by identifying slowly executing queries.
- Perform stress testing and quality assurance by replaying traces.
- Collect a sample of events for tuning the physical database design by using database engine tuning advisor
- Replay traces of one or more users.
- Perform query analysis by saving Show plan results.
- Aggregate trace results to allow similar event classes to be grouped and analyzed. These results provide counts based on a single column grouping.
- Correlate performance counters with a trace to diagnose performance problems.
- Configure trace templates that can be used for tracing later.

Performance Monitor

- Monitor overall resource consumption.
- Wonderful tool to create baseline
- Identify resource bottleneck
- Can be scheduled
- etc



Profiler is tool that monitors the events and activity running on a SQL Server. Using profiler, this monitoring can be viewed, saved, and replayed.

SQL Server Profiler

[illegible]

- Identify SQL Statements causing issues.
- Keeps a track of every SQL event.
- Track acc. to your need.
- Can be scheduled
- Etc.

q

Mostly these events we are using in real time:

Trace Event	Events List
1. All Database Events	Log File Auto Shrink Data File Auto Shrink Log File Auto Grow Data File Auto Grow ErrorLog EventLog Exception Hash Warning Execution Warnings Sort Warnings Missing Column Statistics
2.All Errors and Warnings	Missing Join Predicate Exchange Spill Event Blocked process report User Error Message Background Job Error Bitmap Warning Database Suspect Data Page CPU threshold exceeded
3.Lock and Timeout Events	Lock: Acquired

Lock: Cancel

Lock: Deadlock

Lock: Deadlock Chain

Lock: Timeout

Lock: Timeout (timeout > 0)

4.Performance

Auto Stats

Showplan All or **Show plan XML**

Showplan Statistics Profile or **Showplan XML
Statistics Profile**

5.Security Audit:

Audit Login,

Audit Login Failed

Audit Logout

6.Stored Procedures events:

7.TSQL events:

SQL: BatchStarting

SQL: StmtStarting

8.High CPU usage/Long running queries

SP: StmtStarting

SP: Starting

RPC: Starting

To start, stop and delete a trace you use the following commands.

To find traceid

```
SELECT * FROM:: fn_trace_getinfo (default)
```

This will give you a list of all of the traces that are running on the server.

To start a trace

```
Sp_trace_setstatus traceid, 1
```

Traceid would be the value of the trace

To stop a trace

```
Sp_trace_setstatus traceid, 0
```

Traceid would be the value of the trace

To close and delete a trace

```
Sp_trace_setstatus traceid, 2
```

To delete you need to stop the trace first and then you can delete the trace. This will close out the trace file that is written.

DBA Database Check list:

1. DBCC opentran--To check what the current open transactions are running
2. Check blockings
3. Check deadlocks
4. Check indexes
5. Check fragmentation: Accordingly perform reorganize or rebuild
6. Check file growth settings
7. Check isolation levels

8. Check I\o operations.

Development query tuning\checklist:

Joins

Number of columns (eg:: 6 columns to get data but used select * from option then it will use all the columns)

Grouping functions (sum, avg, max, min, aggregate)

Choosing the better index

Perform implicit and explicit operations

Check no_lock functionality is enabled...

What are the events is captured SQL Profiler?

1. T-SQL statements, stored procedures
2. Cursors, locks(deadlock)
3. Database objects and auto growth of size of data & log files
4. Error & Warnings(syntax errors)
5. Performance(show plan)
6. Table scans
7. Security audits(failed logins, password changes)
8. Monitor server control, memory changes(cpu, reads, writes)
9. Sessions, transactions, tuning

PERFORMANCE TOOL

Performance monitor:

This is one of tool in built in SQL Server and the main purpose is to capture resource utilization in SQL Server in terms of CPU,

MEMORY, DISK I\O and network

1. MEMORY COUNTERS:

1. Page life Expectancy:

This counter tells how much time page residing in the buffer pool is nothing but Page life Expectancy.

The standard value should be greater than 300.

2. Lazy Writes/Sec:

This counter tracks how many times a second that the Lazy Writer process is moving dirty pages from the buffer to disk in order to free up buffer space.

Note: Generally speaking, this should not be a high value, say more than 20 per second or so

3. Checkpoint Pages/Sec: When a checkpoint occurs, all dirty pages are written to disk. This is a normal procedure and will cause this counter to rise during the checkpoint process.

Note: Default automatic check point value is 3 sec

4. Page reads/sec: Number of physical database page reads issued. 80 – 90 per second is normal. Anything value more than leads to memory pressure.

5. Page writes/sec: Number of physical database page writes issued. 80 – 90 per second is normal. Anything value more than leads to memory pressure.

6. Free pages: Total number of pages on all free lists.

Standard value is more than 640.

7. Stolen pages: Number of pages used for miscellaneous server purposes (including procedure cache).

Standard value vary depends on the system.

8. Buffer Cache hit ratio: Percentage of pages that were found in the buffer pool without having to incur a read from disk.

Standard value is always >90%.

9. Target Server Memory (KB): Total amount of dynamic memory the server can consume.

10. Total Server Memory (KB): Total amount of dynamic memory (in kilobytes) that the server is using currently

2. DISK COUNTERS:

1. Avg. Disk Sec/Read: Measure of disk latency. Avg. Disk sec/Read is the average time to read data from disk

Excellent < 08 Msec (.008 seconds)

Good < 12 Msec (.012 seconds)

Fair < 20 Msec (.020 seconds)

Poor > 20 Msec (.020 seconds)

2. Avg. Disk sec/Write: Measure of disk latency. Avg. Disk sec/Write is the average time, in seconds, of a write of data to the disk.

< 8ms (non cached) --Means in disk

< 1ms (cached): Means in buffer.

3. Avg. Disk Read Queue Length: Avg. Disk Read Queue Length is the average number of read requests that were queued for the selected disk during the sample interval

Standard value: < 2 * spindles

4. Avg. Disk Write Queue Length: Avg. Disk Read Queue Length is the average number of write requests that were queued for the selected disk during the sample interval

Standard value: < 2 * spindles

5. Avg disk reads\sec: Per second how many number of reads should perform in disk

6. Avg disk writes\sec: Per second how many number of writes should perform in disk

3. CPU OR PROCESSOR COUNTERS:

%Processor Time:

%Privileged Time

User time

Interrupts time

4. NETWORK COUNTERS:

Network Interface: Bytes Sent/sec

Network Interface: Bytes Received/sec

Network Interface: Bytes Total/sec.

Processor: % DPC Time

Processor: DPCs queued/sec.

TCP: Segments Sent/sec.

TCP: Segments Received/sec

TCP: Segments/sec.

TCP: Connections Reset.

TCP: Connections Established

DTA [DATABASE TUNING ADVISOR]

Database tuning advisor: [DTA]

> The main purpose of this tool is to provide recommendations to improve the query performance by analysis the query [work load file] on a table in the database.

> Database tuning advisor can analyze the performance effects of workloads run against one or more databases or a SQL profiler trace (they may contain T-SQL batch or remote procedure call). After analyzing, it recommends to add, remove or modify physical design structure such as clustered and non-clustered indexes, indexed views and partitioning.

Workload: a work load is set of transact-SQL statements that executes against databases you want to tune.

Steps:

1. Open DTA tool
2. Save the query as file
3. Provide the path for work load file
4. Select the database
5. Start analysis.

Note: After analysis DTA provides recommendations and then implement always on test server first. Check the performance and after implement on prod system.

From DTA reports> statement cost report > Value have to check to know how much percentage of query faster.

Example:

Create table [Table] (userID varchar (55), FirstName nvarchar (55), Lastname nvarchar (55))

; With sequence as (

Select N = row_number () over (order by @@spid)

from sys.all_columns c1, sys.all_columns c2

)

Insert into [Table] (UserID, FirstName, Lastname)

q

Select

```
'UserID' + right ('000000' + cast (N as varchar (10)), 6),  
'John' + right ('000000' + cast (N as varchar (10)), 6),  
'Doe' + right ('000000' + cast (N as varchar (10)), 6)  
from sequence where N <= 100000  
Select * from [Table] where Lastname='Doe299999'
```

How to clear the buffers:

DBCC FREEPROCCACHE

GO

DBCC DROPCLEANBUFFERS

GO

DBCC FREESYSTEMCACHE ('ALL')

GO

DBCC FREESESSIONCACHE

Note:

Never clear buffer until proper confirmation from application team.

Statistics: Mainly it contains table updated information and density value information.

Whenever we update table then statistics also updated.

Note: By default AUTO statistics option is always true in each database.

Drop statistics DBO. [Table]. [_WA_Sys_00000003_0F975522]

Drop statistics DBO. [Table]. [_WA_Sys_00000001_0F975522]

ACID PROPERTIES

What is Transaction?

Transaction is a set of logical unit of work and it contains one or more database operations. A valid transaction should be met (ACID) Atomicity, Consistency, Isolation, Durability properties.

Atomicity: A transaction must be an automatic unit of work, either all of its data modifications are performed or none of them is performed.

Consistency: When completed a transaction must leave all data in consistent state. In a relational database, all rules must be applied to transactions modifications to maintain all data integrity. All internal data structures, such as B-tree indexes (or) doubly linked lists must be correct at end of transaction.

Isolation: Modifications made by concurrent transaction must be isolated from the modifications made by the other consequent transactions. A transaction either sees a data in the state it was in before another concurrent transaction modified it, (or) sees the data after second transaction completed.

Durability: After a transaction has completed, its effects are permanently in place in the system. The modification persists even in the event of a system failure.

WINDOWS TASK SCHEDULAR

Task Scheduler backup:

Note: In Sql server express edition we won't have Sql server agent service due to this we cannot be able to configure jobs or either Maintaince plans to automate.

How can we automate backups then?

BY USING WINDOWS LEVEL TASK SCHEDULER:

Steps:

1. Create a batch file with

Instance name

Path

Type of backup

Output file path

Retention period

2. Execute SP in master database for configuration.

3. Go to run > task schedule

Path

Scheduler name

Schedule

Account to run this task

4. Your task scheduler contact batch file and trigger backups via sp.

5. Test the backup status.

QUERY TUNING

Query is running slow or query tuning:

1. Check any open Tran are there by using

DBCC OPENTRAN

2. Check for any blockings are there?

If yes then according to your project process we find blockings and speak to apps team to confirm which SPID required to kill with approvals via email.

KILL SPID

If no blockings then?

3. Check for any deadlocks

Note: By default SQL Server db engine will not capture any deadlock information DBA team need to enable trace flags.

DBCC Traceon (1222 or 1204, -1)

4. Check for any query execution plan or cost based plan.

Execution plan display:

1. Physical operation (Scan type)
2. Logical Operation
3. Estimated I\O COST
4. Estimated CPU COST
5. Estimated number of executions
6. Estimated number of rows.
7. Cache Size

q

8. Rebinds

9. Rewinds

10. Operator cost

Execution plan definitions:

Actual Execution Plan - (CTRL + M) - is created after execution of the query and contains the steps that were performed

Estimated Execution Plan - (CTRL + L) - is created without executing the query and contains an approximate execution plan

> Rebind: Provides number of join operations

> Rewind: Provides number of join operations has changed since.

5. Check whether any lock types are in the tables

SP_LOCK or select * from Sys.dm_tran_locks [DMV]

6. Check whether any indexes are created on the table at or not. If no then inform your dev\apps team to suggest create index which improves performance of query.

7. Check for FRAGMENTATION LEVEL on the tables.

Select * from sys.dm_db_index_physical_stats (DB_ID ('DB NAME'), Object_id ('tablename'), NULL, NULL, 'Detailed') -----(From SQL Server 2005)

or DBCC SHOWCONTIG (Upto 2000 version of SQL Server)

If We FRAGMENTATION level

1. <5: No action and indexes are good.

2. If >5 and <30: Index need to reorganize

Alter index indexname on table name reorganize

3. If >30: Index need to rebuild

Alter index indexname on table name Rebuild

If no FRAGMENTATION

8. Check for any missing indexes by using

Select * from sys.dm_db_missing_index_details

9. Check for any unused indexes in a table

Select * from sys.db_unused_index_details

10. Check CPU and memory utilisation.get Top 10 queries which are consuming more CPU and memory in the Sql then inform share the details to apps team.

11. Run profiler or perfmon tool to capture events or counters depending on type of parameters.

12. Check any disk or I\O or Network related issues.

13. Run the DTA [Database tuning advisor] to get any suggestions to improve the performance

From query side:

1. Check the query (select) contains no_lock functionality or not.

2. Check whether any complicated loops or joins or triggers used or not.

Note: How to find which SQL Server instance is consuming more CPU or memory when multiple instances in a single server.

Go to task manager> view > select columns > Select PID [Process id]. Take PID for SQL Server and compare with configuration manager> Process id for specific instance. Compare both PID for task manager and configuration manager > Then conclude the instance name which takes more memory or CPU.....

How to find SPID level inside the instance?

Use SP_WHO2 or SP_WHO3

HIGH CPU ISSUE

1. Go to task manager> check whether SQL Server .exe (SSMS.EXE) is using more CPU or not?

If using then look further investigation from inside Sql server else if any other resource .exe is using inform to concern team.

Check any long running or any open transactions are running in SQL Server causing high CPU.

2. If it's SQL server then open perfmon and add the counter for

Processor: % Privileged time,

% Processor time

%user time

If you see the %privileged time to be high like covering around 40% and above then it could be some driver issue.

Note: Per processor 156 worker threads can create.

3. First and very important one is to know what's the architecture i.e. 64 bit (X64 or IA 64) or 32 bit(x86):

Check the configuration using command: Xp_msver

Note: -Improper index or missing index or fragmented index which lead to query to run longer .It leads to high CPU.

4. Check if there are any trace flags enabled:

Dbcc tracestatus (-1); It's always good to check the reason of enabling the trace flags, if enabled.

5. Check the below mentioned SQL parameters

```
sp_configure 'advanced', 1
```

```
go
```

```
Reconfigure with override
```

```
go
```

```
sp_configure
```

```
go
```

Priority Boost: - it's not recommended to be enabled as it can cause high CPU utilization. If it's enabled (value =1) please disable (set value to 0) it - it's a static parameter which needs SQL restart

Max degree of parallelism: - Although this parameter doesn't make much of a difference but sometimes I have seen SPID stuck in ACCESS_METHODS_SCAN_RANGE_GENERATOR, cx packet or other parallel wait types. If you see any such waits when you run this query:

Note: - If you have more than 8 processors, please make sure the max degree of parallelism parameter is 8 or below 8.

7. Run CPU related counters by using perfmon tool

8. DBCC FREEPROCACHE: prepares new plan by query optimizer and recompilation starts again...Which may improve the performance and reduce CPU utilization.

9. Check with developer on designing of query or Application design or any further bugs...

SQL Server - Max Degree of Parallelism (MAXDOP):

> Limit the number of processors to use in parallel plan execution

> The Max Degree of Parallelism or MAXDOP is a configuration indicating how the SQL Server optimizer will use the CPUs. This is a server wide configuration that by default uses all of the CPUs to have the available portions of the query executed in parallel. MAXDOP is very beneficial in a number of circumstances, but what if you have a reporting like query that runs in an OLTP system that monopolizes much of the CPU and adversely affects typical OLTP transactions.

Or

> When SQL Server runs on a computer with more than one microprocessor or CPU, it detects the best degree of parallelism, that is, the number of processors employed to run a single statement, for each parallel plan execution. You can use the max degree of parallelism option to limit the number of processors to use in parallel plan execution. To enable the server to determine the maximum degree of parallelism, set this option to 0, the default value. Setting maximum degree of parallelism to 0 allows SQL Server to use all the available processors up to 64 processors. To suppress parallel plan generation, set max degree of parallelism to 1. Set the value to a number greater than 1 to restrict the maximum number of processors used by a single query execution. The maximum value for the degree of parallelism setting is controlled by the edition of SQL Server, CPU type, and operating system. If a value greater than the number of available processors is specified, the actual number of available processors is used. If the computer has only one processor, the max degree of parallelism value is ignored.

What all benefit from parallel execution plan:

❓ **Complex/Long running queries** – During query optimization, SQL Server looks for queries that might benefit from parallel execution. It distinguishes between queries that benefit from parallelism and those that do not benefit, by comparing the cost of an execution plan using a single processor versus the cost of an execution plan using more than one processor and uses the cost threshold for parallelism value as a boundary point to determine short or long query. In a parallel query execution plan, the INSERT, UPDATE, and DELETE operators are executed serially. However, the WHERE clause of an UPDATE or a DELETE statement, or the SELECT part of an INSERT statement may be executed in parallel. The actual data changes are then serially applied to the database.

Index data definition language (DDL) – Index operations that create or rebuild an index (REBUILD only, not applicable to REORGANIZE), or drop a clustered index and queries that use CPU cycles heavily are the best candidates for a parallel plan. For example, joins of large tables, large aggregations, and sorting of large result sets are good candidates.

Other Operations – Apart from the above, this option also controls the parallelism of DBCC CHECKTABLE, DBCC CHECKDB, and DBCC CHECKFILEGROUP.

The degree of parallelism value is set at the SQL server instance level and can be modified by using the sp_configure system stored procedure (command shown below). You can override this value for individual query or index statements by specifying the MAXDOP query hint or MAXDOP index option. Note that this can be set differently for each instance of SQL Server. So if you have multiple SQL Server instances in the same server, it is possible to specify a different Maximum DOP value for each one.

--The max degree of parallelism option is an advanced option

--and let you set only when show advanced options is set to 1

```
sp_configure 'show advanced options', 1;
```

```
GO
```

```
RECONFIGURE WITH OVERRIDE;
```

```
GO
```

--configuring to use 8 processors in parallel

--setting takes effect immediately (without restarting the MSSQLSERVER service)

```
sp_configure 'max degree of parallelism', 8;
```

```
GO
```

```
RECONFIGURE WITH OVERRIDE;
```

```
GO
```

By default, when SQL is installed on a server, the parallelism setting is set to 0 meaning that the optimizer can utilize all the available processors to execute an individual query. This is not necessarily the most optimal setting for the application and the types of queries it is designed to support.

❓ **Scenario 1** – For OLTP application, a typical setting is 1 would help. The reason for this is that in an OLTP environment, most of the queries are expected to be point queries which address one or a relatively small number of records. Such queries do not need parallelized processing for efficient execution. If there are specific queries which have a need for a setting greater than 1, then the source code needs to be examined to see if a MAXDOP hint can be availed.

❓ **Scenario 2** – For OLAP application, the setting should typically be default 0 (up to 8 processors) or be greater than 1, because each queries, such application will use, will typical target thousands of, millions of records and also there might a scenario when you drop the index before ETL operation and re-create it once refreshed data is uploaded in typical data warehousing application. There will definitely be performance advantages in using multiple processors to do these works in parallel fashion.

Note: Using a setting of 0 in these applications is not recommended, especially when there are more than 8 processors in order to keep the coordination costs, context switching down to manageable levels. It is typical to start with a value of 4 and experiment with the reporting queries to see if this needs to be adjusted upwards.

MEMORY ISSUE

How to find SQL Server actual memory utilization?

Steps: 1. Go to task manager> check SQLsrv.exe or SSMS.exe how much memory utilizing.

2. If any other .exe is using more Memory then inform to relevant team. If SQL Server is consuming then go to SSMS> try to check any spid consuming more memory by using

Select * from sys.sysprocesses

3. If example: is Sql server is consuming 20 GB out of total 24 GB physical memory, then how to find actual utilization from Sql end.

By using, DBCC memory status

Find Value or parameter: STOLEN POTENTIAL [Shows free memory from SQL level]

CONVERT VALUE INTO: $\text{VALUE} * 8 / 1024 / 1024 = \text{Actual free memory out of total SQL Server consumed memory.}$

19 GB free out of SQL Server memory 20 GB.

4. Please check whether memory capping or restriction is done at SQL Server level.

5. If not please inform apps team to raise Change record and you provide suggestion how much memory capping to be set dedicatedly for SQL Server.

Always keep 80% of memory Sql server and rest of 20 % of memory to o\s out of total physical memory.

6. Still memory issue, then enable the memory related counters to find the standard values.

7. Verify the execution plans and sometimes query optimizer may prepare wrong plans which leads to high CPU and high memory

To clear the plans

DBCC FREEPROCCACHE---It clear all the plans from buffer pool

Note: plz get approval from application team before performs the cache clear.

UPDATE STATISTICS

> By default, the query optimizer already updates statistics as necessary to improve the query plan in some cases you can improve query performance by using UPDATE STATISTICS

> Updating statistics ensures that queries compile with up-to-date statistics. However, updating statistics causes queries to recompile. We recommend not updating statistics too frequently because there is a performance tradeoff between improving query plans and the time it takes to recompile queries

Note:

1. Update statistics is the one of the DBA Maintaince activity and runs every day after business hours.
2. After rebuild index no need to perform again update statistics. Why because whenever index runs automatically Update statistics also gets updated.
3. Whenever create any index, always first create STATISTICS and the create indexes.

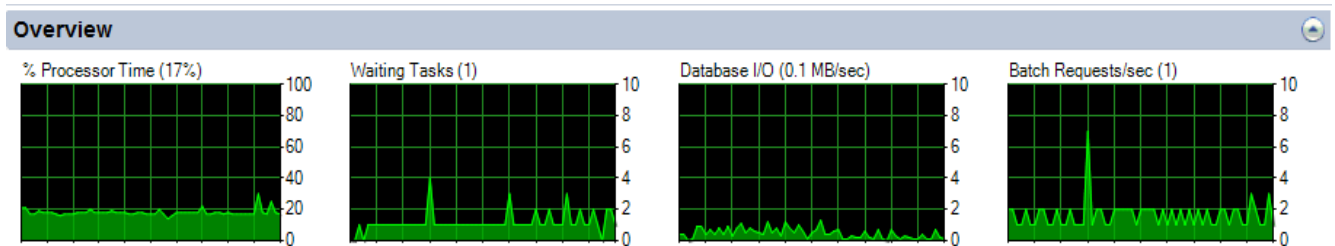
ACTIVITY MONITOR

> Activity monitor is used to get the information about users connections to the database engine and the locks that they hold. Activity monitor is used to troubleshooting database locking issues, and to terminate deadlocked or unresponsive process.

> **To use activity monitor:** VIEW SERVER STATE permission on server and SELECT permission to the sysprocesses & syslocks tables in the master database.

> **To kill process:** Sysadmin and processadmin database roles and permissions are required to KILL a process.

This snapshot is very helpful to get a quick performance snapshot without the need to use other monitoring tool for the same purpose.



% Processor Time – is the percentage of time the processors spend to execute threads that are not idle

Waiting Tasks – is the number of tasks that are waiting for processor, I/O, or memory to be released so the tasks can be processed

Database I/O – is the data transfer rate in MB/s from memory to disk, disk to memory, or disk to disk

Batch Requests/sec – is the number of SQL Server batches received by the instance in a second

The Processes pane:

> The Processes pane shows the information about the currently running processes on the SQL databases, who runs them, and from which application

Processes									
Sessi...	U..	Login	Database	Task State	Com...	Application	Wait Time (ms)	Wait Type	Wait Resource
51	1	NT SERVICE\Rep...	ReportServer\$SQL2...			Report Server		Current wait type (see "sys.dm_os_waiting_tasks.wait_type")	
52	1	NT AUTHORITY\...	ApexSQLCrdb			ApexSQL Comply 9812	0		
53	1	Fujitsu\Milena	msdb			Microsoft SQL Server ...	0		
54	1	NT AUTHORITY\...	tempdb			ApexSQL Comply 9812	0		
55	1	Fujitsu\Milena	master			Microsoft SQL Server ...	0		
56	1	Fujitsu\Milena	tempdb	RUNNING	SELECT	Microsoft SQL Server ...	0		
57	1	Fujitsu\Milena	master			Microsoft SQL Server ...	0		
58	1	NT SERVICE\Rep...	ReportServer\$SQL2...			Report Server	0		
59	1	NT AUTHORITY\...	ApexSQLCrdb			.Net SqlClient Data Pr...	0		

Session ID – is a unique value assigned by Database Engine to every user connection. This is the spid value returned by the sp_who procedure

User Process – 1 for user processes, 0 for system processes. The default filter is set to 1, so only user processes are shown

Login – the SQL Server login that runs the session

Database – the database name on which the process is running

Task State – the task state, blank for tasks in the runnable and sleeping state. The value can also be obtained using the sys.dm_os_tasks view, as the task_state column. The states returned can be:

“PENDING: Waiting for a worker thread.

RUNNABLE: Runnable, but waiting to receive a quantum.

RUNNING: Currently running on the scheduler.

SUSPENDED: Has a worker, but is waiting for an event.

DONE: Completed.

SPINLOOP: Stuck in a spinlock.” [2]

Command – the current command type. The value can also be obtained using the sys.dm_exec_requests view, as the command column

Application – the name of the application that created the connection

Wait Time (ms) – how long in milliseconds the task is waiting for a resource. The value can also be obtained using the sys.dm_os_waiting_tasks view, as the wait_duration_ms column

Wait Type – the last/current wait type. The value can also be obtained using the sys.dm_os_waiting_tasks view, as the wait_type column. The waits can be resource, queue and external waits

Wait Resource – the resource the connection is waiting for. The value can also be obtained using the sys.dm_os_waiting_tasks view, as the resource_description column

Blocked By – the ID of the session that is blocking the task. The value can also be obtained using the sys.dm_os_waiting_tasks view, as the blocking_session_id column

Head Blocker – the session that causes the first blocking condition in a blocking chain

Memory Use (KB) – the memory used by the task. The value can also be obtained using the sys.dm_exec_sessions view, as the memory_usage column

Host Name – the name of the computer where the current connection is made. The value can also be obtained using the sys.dm_exec_sessions view, as the host_name column

Workload Group – the name of the Resource Governor Workload group [3]. The value can also be obtained using the sys.dm_resource_governor_workload_groups view, as the name column

The Resource Waits pane:

Shows information about waits for resources

Resource Waits				
Wait Category	Wait Time (ms/sec)	Recent Wait Time (ms/sec)	Average Waiter Count	Cumulative
Buffer I/O	3	8		0.0
Logging	2	1		0.0
Memory	0	0		0.0
Network I/O	0	0		0.0
Other	0	0		0.0
SQLCLR	0	0		0.0
Buffer Latch	0	0		0.0
Compilation	0	0		0.0
Latch	0	0		0.0

Wait Category – the categories are created combining closely related wait types. The wait types are shown in the Wait Type column of the Processes pane

Wait Time (ms/sec) – the time all waiting tasks are waiting for one or more resources

Recent Wait Time (ms/sec) – the average time all waiting tasks are waiting for one or more resources

Average Waiter Count – is calculated for a typical point in time in the last sample interval and represents the number of tasks waiting for one or more resources

Cumulative Wait Time (sec) – the total time waiting tasks have waited for one or more resources since the last SQL Server restart, or DBCC SQLPERF last execution

The Data File I/O pane:

Shows information about the database files on the SQL Server instance. For each database, all database files are listed – MDF, LDF and NDF, their paths, and names

Data File I/O		
Database	File Name	MB/sec Read
AdventureWorks	C:\Program Files\Microsoft SQL Server\MSSQL11.SQL2012\MSSQL\DATA\AdventureWorks2012_Data.mdf	
AdventureWorks	C:\Program Files\Microsoft SQL Server\MSSQL11.SQL2012\MSSQL\DATA\AdventureWorks2012_log.ldf	
AdventureWorks	d:\file_SecondaryFG.ndf	
ApexSqlMonitor	C:\Program Files\Microsoft SQL Server\MSSQL11.SQL2012\MSSQL\DATA\ApexSqlMonitor.mdf	
ApexSqlMonitor	C:\Program Files\Microsoft SQL Server\MSSQL11.SQL2012\MSSQL\DATA\ApexSqlMonitor_log.ldf	
master	C:\Program Files\Microsoft SQL Server\MSSQL11.SQL2012\MSSQL\DATA\master.mdf	
master	C:\Program Files\Microsoft SQL Server\MSSQL11.SQL2012\MSSQL\DATA\mastlog.ldf	

MB/sec Read – shows recent read activity for the database file

MB/sec Written – shows recent write activity for the database file

Response Time (ms) – average response time for recent read-and-write activity

The Recent Expensive Queries pane:

Expensive queries are the queries that use much resources – memory, disk, and network. The pane shows expensive queries executed in the last 30 seconds. The information is obtained

from the sys.dm_exec_requests and sys.dm_exec_query_stats views. A double-click on the query opens the monitored statement

The context menu for the specific query provides options to open the query in Query Editor, and show the execution plan

Recent Expensive Queries						
Query	Executio...	CPU (ms/sec)	Physical ...	Logical ...	Logical ...	Ave
WITH merged_query_stats AS (SELECT ...		2	177	12	4	460
WITH interval_fingerprint AS (SELECT	WITH interval_fingerprint_stats AS		125	0	0	141
SELECT StartTime, Event	(27	0	0	0
INSERT INTO #am_fing	SELECT		13	0	2	45
SELECT [Session ID]	cur_stats.batch_text,		1	0	0	18
SELECT JobID, StartDat	SUBSTRING (0	0	0	0
	cur_stats.batch_text,					
	(cur_stats.sample_statement_start_offset/2) + 1					
select colid	= s_tcv.colid, na...	0	0	0	0	0
select top 4	-- Notificatio...	6	0	0	0	0
SET @DBVersion = (select top(1) [ServerVersio...		2	0	0	0	0

Query – the SQL query statement monitored

Executions/min – the number of executions per minute, since the last recompilation. The value can also be obtained using the sys.dm_exec_query_stats view, as the execution_count column

CPU (ms/sec) – the CPU rate used, since the last recompilation. The value can also be obtained using the sys.dm_exec_query_stats view, as the total_worker_time column

Physical Reads/sec, Logical Writes/sec, and Logical Reads/sec – the rate of physical reads/logical writes/logical reads per second. The value can also be obtained using the sys.dm_exec_query_stats view, as the total_physical_reads/ total_logical_writes/ total_logical_reads columns

Average Duration (ms) – average time that the query runs. Calculated based on the total_elapsed_time and execution_count columns in the sys.dm_exec_query_stats view

Plan Count – the number of duplicate query plans. A large number requires investigation and potential explicit query parameterization

EXECUTION PLAN

What is execution plan and explain it?

Execution plan graphically displays the data retrieval methods chosen by SQL Server. It represents the execution cost of specific statements and queries in SQL Server. This graphical approach is very useful for understanding the performance of the query.

What is an execution plan? When would you use it? How would you view the execution plan?

An execution plan is basically a road map that graphically or textually shows the data retrieval methods chosen by SQL Server Query optimizer for a stored procedure or ad-hoc query and is a very useful tool for a developer to understand the performance characteristics of a query or stored procedure since the plan is the one that SQL Server will place in its cache and use to execute the stored procedure or query. From within Query Analyzer is an option called "Show Execution Plan" (located on the query drop-down menu). If this option is turned on it will display query execution plan in separate window query is ran again.

Execution plan displays:

1. Physical operations
2. Logical operations
3. Actual Number rows
4. Estimated I/O cost
5. Estimated CPU cost
6. Number of Executions
7. Estimated Number of Executions
8. Estimated Operator cost
9. Estimated Subtree cost
10. Estimated Number of Rows
11. Estimated Row Size
12. Actual rebinds
13. Actual rewinds
14. Key lookup
15. Nested look up
16. Index seek
17. Index scan

RAID LEVELS

RAID Levels: [RAID stands for Redundant Array of Inexpensive or Independent Disks]

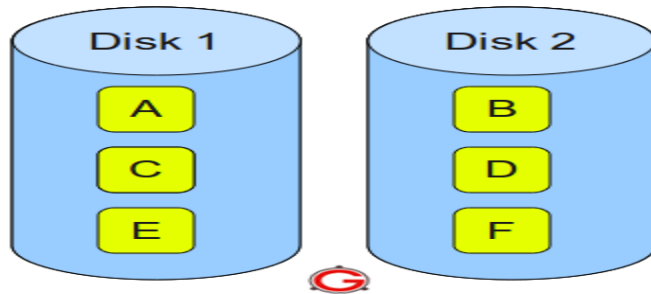
> RAID is a disk system that contains multiple disk drives, called an array, to provide greater performance, fault tolerance, storage capacity, at a moderate cost. While configuring your server system, you typically have to make a choice between hardware RAID and software RAID for the server's internal disk drives



TYPES OF RAID:

1. RAID -0:

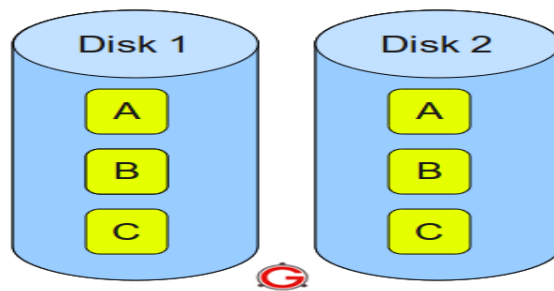
1. Minimum 2.Disks.
2. Excellent performance (as blocks are striped).
3. No redundancy (no mirror, no parity).
4. Don't use this for any critical system



RAID 0 – Blocks Striped. No Mirror. No Parity.

2. RAID-1:

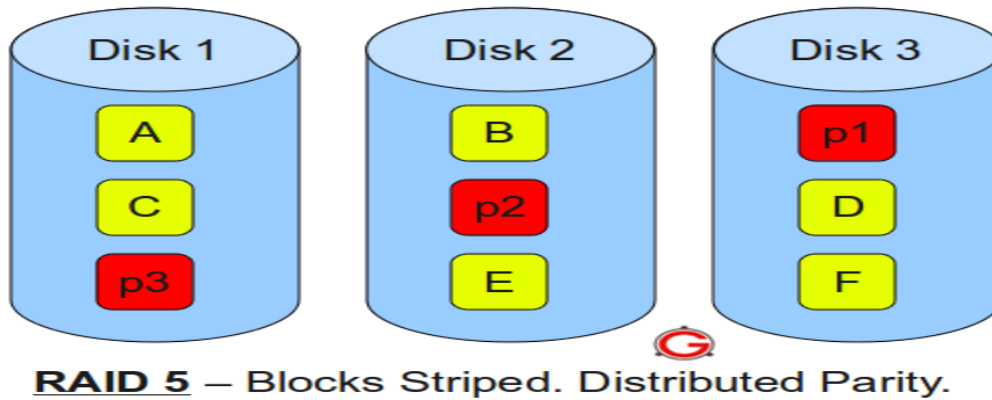
1. Minimum 2 disks.
2. Good performance (no striping. no parity).
3. Excellent redundancy (as blocks are mirrored)



RAID 1 – Blocks Mirrored. No Stripe. No parity.

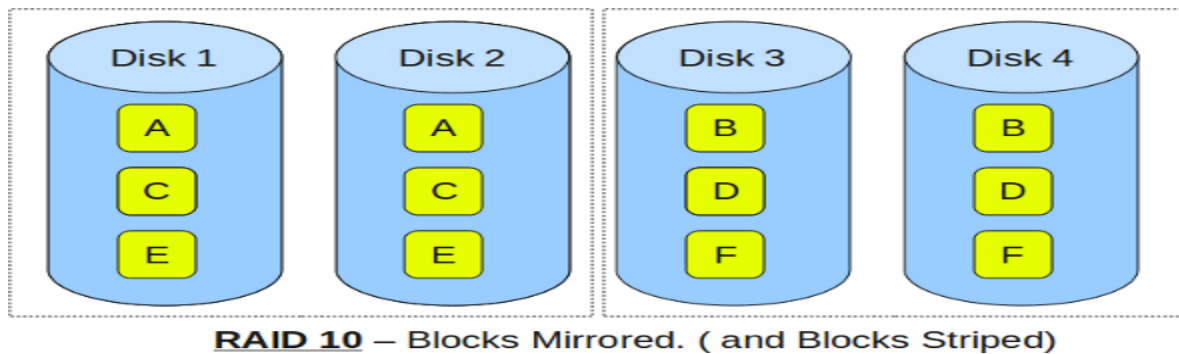
3. RAID-5:

1. Good performance (as blocks are striped).
2. Good redundancy (distributed parity).
3. Best cost effective option providing both performance and redundancy. Use this for DB that is heavily read oriented. Write operations will be slow.



4. RAID-10 [RAID 1 + RAID 0]:

1. Minimum 4 disks.
2. This is also called as “stripe of mirrors”
3. Excellent redundancy (as blocks are mirrored)
4. Excellent performance (as blocks are striped)



If you can afford the dollar, this is the BEST option for any mission critical applications (especially databases)

Note: From disk management you can find whether RAID is configured or not by using "if any 2 or 3 disks letter having same ...That mean raid implemented"

You can configure any RAID from disk management...

TEMPDB ARCHITECTURE

What is Stored in Tempdb?

Tempdb is used to store three different categories of temporary data:

- **User Objects**
- **Internal Objects**
- **Version Stores**

User Objects:

- Local and global temporary tables and indexes
- User-defined tables and indexes
- Table variables
- Tables returned in table-valued functions

Note: These lists are not designed to be all inclusive.

Internal Objects:

- Work tables for DBCC CHECKDB and DBCC CHECKTABLE.
- Work tables for hash operations, such as joins and aggregations.
- Work tables for processing static or keyset cursors.
- Work tables for processing Service Broker objects.
- Work files needed for many GROUP BY, ORDER BY, UNION, and SELECT DISTINCT operations.
- Work files for sorts that result from creating or rebuilding indexes (SORT_IN_TEMPDB).
- Storing temporary large objects (LOBs) as variables or parameters (if they won't fit into memory).

Version Stores:

- The version store is a collection of pages used to store row level versioning of data.
- There are two types of version stores:

1. Common Version Store: Used when:

- Building the inserted and deleted tables in after triggers.
- When DML is executed against a database using **snapshot transactions** or **read-committed row versioning isolation levels**.
- When multiple active result sets (MARS) are used.

2. Online-Index-Build Version Store: Used for online index builds or rebuilds. EE edition only.

Tempdb doesn't act as an other databases:

- Tempdb only uses simply recovery model.
- Manydb options not be able to change.
- Tempdb may not be dropped, attached or detached.
- Tempdb may not backed up, restore, can't implement any HA options.

Types of Tempdb Problems:

- Generally, there are three major problems you run into with Tempdb:
 1. Tempdb is experiencing an I/O bottleneck, hurting server performance.
 2. Tempdb is experiencing DDL and/or allocation contention on various global allocation structures (metadata pages) as temporary objects are being created, populated, and dropped. E.G. Any space-changing operation (such as INSERT) acquires a latch on PFS, SGAM or GAM pages to update space allocation metadata. A large number of such operations can cause excessive waits while latches are acquired, creating a bottleneck, and hurting performance.
 3. Tempdb has run out of space.

SOLUTION:**Use performance Monitor:****And also DMV'S are useful what is going on Tempdb**

- **sys.dm_db_file_space_usage:** Returns one row for each data file in Tempdb showing space usage.
- **sys.dm_db_task_space_usage:** Returns one row for each active task and shows the space allocated and deallocated by the task.
- **sys.dm_db_session_space_usage:** Returns one row for each session, with cumulative values for space allocated and deallocated by the session.

Monitoring Tempdb Space:**Performance Counters:**

- SQL Server: Database: Data File(s) Size (KB): tempdb
- SQL Server: Database: Log File(s) Used Size (KB): tempdb
- SQL Server: Transactions: Free Space in tempdb (KB)

DMV

- sys.dm_db_file_space_usage

Errors in tempdb running slow check in error logs:

- Check the SQL Server error log for these errors:
 - 1101 or 1105: A session has to allocate more space in tempdb in order to continue
 - 3959: The version store is full.
 - 3967: The version store has been forced to shrink because tempdb is full.
 - 3958 or 3966: A transaction is unable to find a required version record in tempdb.

Note: Be sure auto growth is turned on for tempdb, and ensure that you have enough available free disk space.

Operations that cannot be performed on the tempdb database:

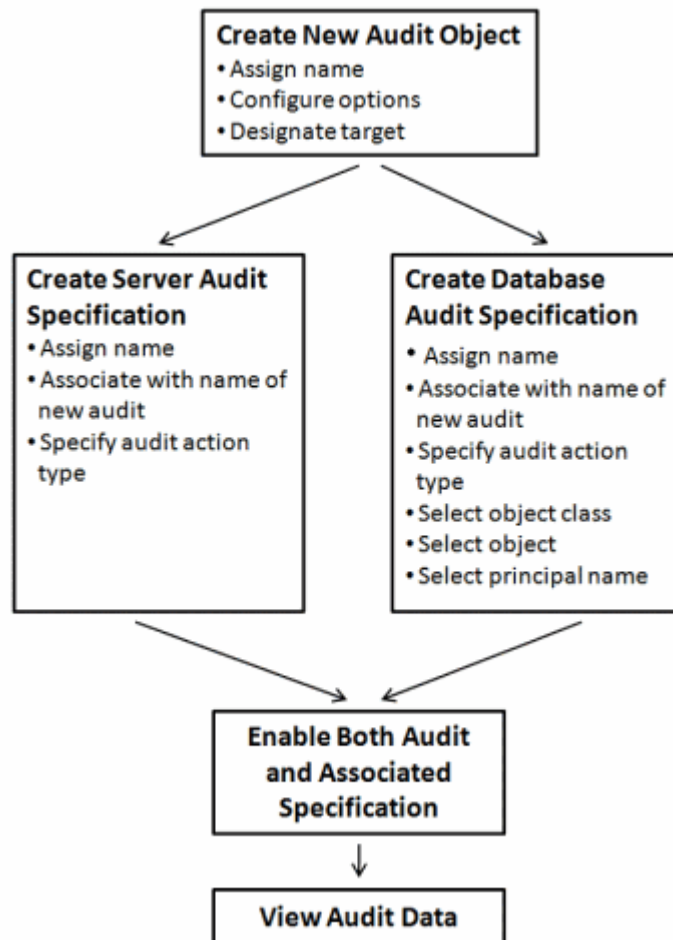
- > Adding filegroups.
- > Backing up or restoring the database.
- > Changing collation. The default collation is the server collation.
- > Changing the database owner. Tempdb is owned by dbo.
- > Creating a database snapshot.
- > Dropping the database.
- > Dropping the guest user from the database.
- > Participating in database mirroring.
- > Removing the primary filegroup, primary data file, or log file.
- > Renaming the database or primary filegroup.
- > Running DBCC CHECKALLOC.
- > Running DBCC CHECKCATALOG.
- > Setting the database to OFFLINE.
- > Setting the database or primary filegroup to READ_ONLY.

SQL SERVER AUDITING

Auditing: Auditing an instance of SQL Server or a SQL Server database involves tracking and logging events that occur on the system.

Working with SQL Server 2008 auditing need to keep four things in mind:

- **Server Audit Specification (Events to capture on the Server Instance Level)**
- **Database Audit Specification (Events to capture on a specific database)**
- **Target (Where would be the events be logged)**



- Auditing is a new feature in SQL Server 2008 which enables database administrators to capture the events. I hope this feature will be light weight compared to other third party audit event collectors.

- The **audit** object provides a manageable auditing framework that makes it easy to define the events that should be logged and the locations where the log should be stored.
- SQL Server helps you to implement a comprehensive auditing solution to secure database your database and meet regulatory compliance requirements.
- Assign all users to meaningful logical groups.
- Assign permissions to groups.
- Always use Windows authentication mode if possible.
- Change the SA account password to a known value if you might ever need to use it. Always use a strong password for the SA account and change the SA account password periodically.
- Do not manage SQL Server by using the SA login account; assign Sysadmin privilege to a known user or group.
- Rename the SA account to a different account name to prevent attacks on the SA account by name.
- User has to perform any activities on the database then DBA need to provide relevant permissions to the user.

Creating an audit, and reviewing audit results using SSMS, is a four-step process, as outlined in the previous section:

1. Creating the Audit object
2. Creating a Server or Database Audit Specification
3. Starting the Audit
4. Reviewing Audit Events

Creating the Audit Object:

The first step is to create a new audit object. To create a new audit object using SSMS, go to the SQL Server instance you want to audit, open up “Security,” and you will see the “Audits” folder, as shown in Figure 2:

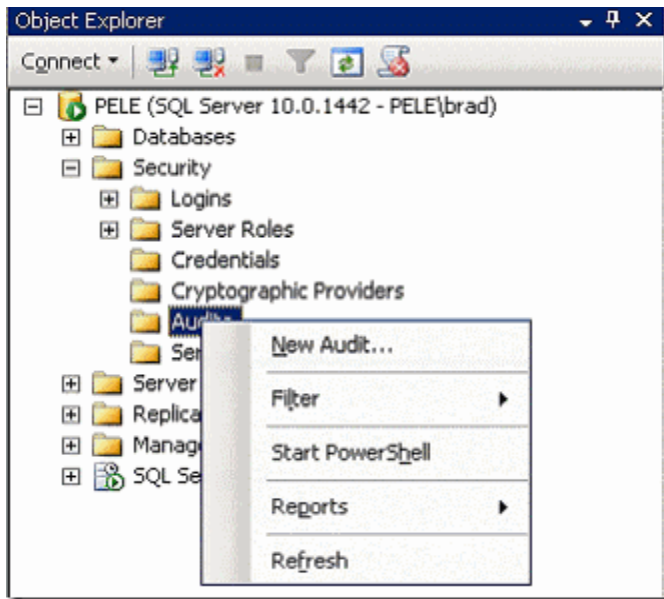


Figure 2: Choose “New Audit” to create an audit from within SSMS.

Right-click on the “Audits” folder and select “New Audit,” and the “Create Audit” dialog box appears, as shown in Figure 3:

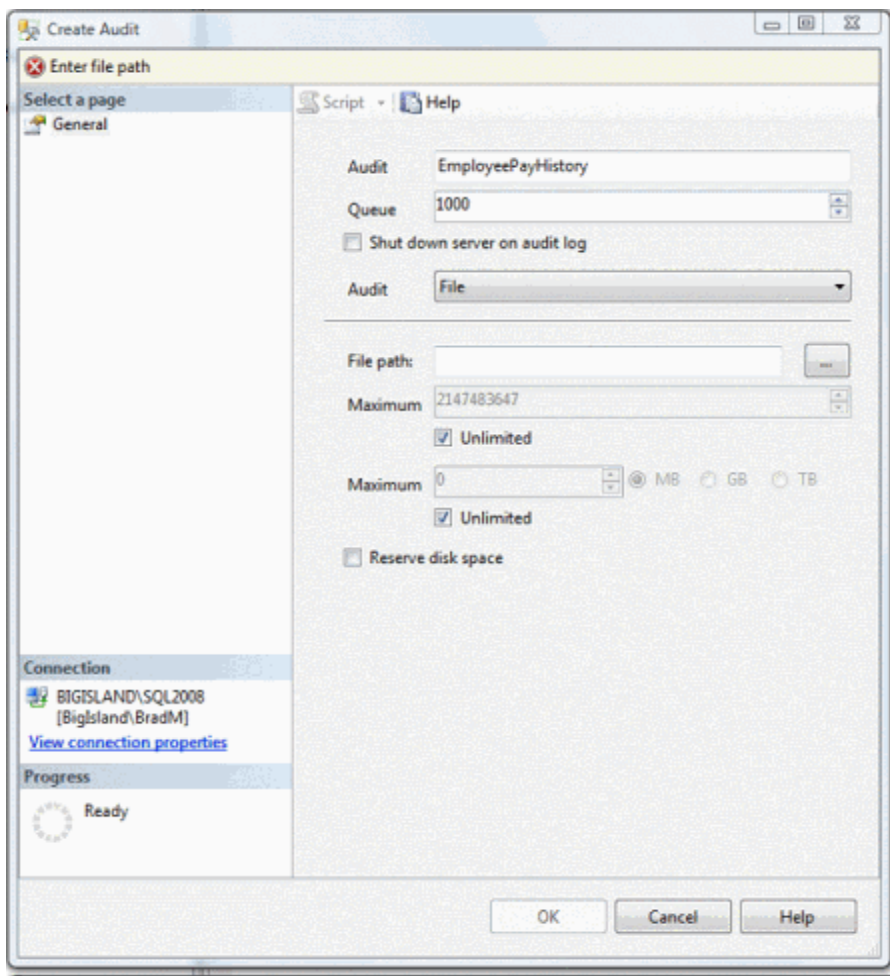


Figure 3: To create an audit, you have to assign it a name and specify where the audit data will reside.

The first thing you need to do is to decide if you want to use the name that is automatically generated for you as the audit object name, or to assign your own name. Since numbers don't mean much to me, I assigned it my own name.

Next, you have to provide a "Queue Delay" number. This refers to the amount of time after an audit event has occurred before it is forced to be processed and written to the log. The default value is 1000 milliseconds, or 1 second. While I am going to accept the default for this demo, you might want to consider increasing this value if you have a very busy server.

The next option on the screen is called "Shut down server on audit log failure". If you select this option, and later SQL Server is restarted, and for whatever reason the audit data can't be logged, then SQL Server will not start, unless you manually start it at the command line using a special parameter. This option should only be used in environments where very tight

auditing standards are followed and you have 24/7 staff available to deal with the problem, should it occur.

Next, beside “Audit,” in the dialog box, there is a drop-down box with “File” selected by default. This option is used to tell SQL Server where you want the audit logs to be stored.

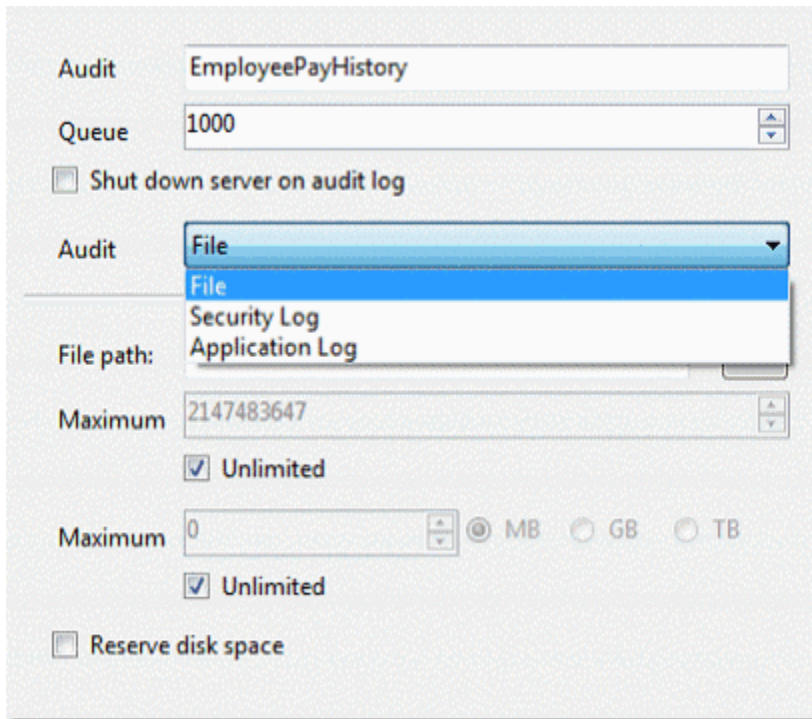


Figure 4: Three are three options where you can store audit data.

SQL Server Audit allows you to store audit data in a file, in the Security Log, or the Application Log. If you choose “File”, then you must also specify the location of the file, along with additional information, such as how much data it can collect, and so on. If you choose Security Log or Application Log, then the audit results are stored in these Windows Operating System Event Logs. I am going to choose “Application Log”. Once this is done, the dialog box should look as shown in Figure 5:

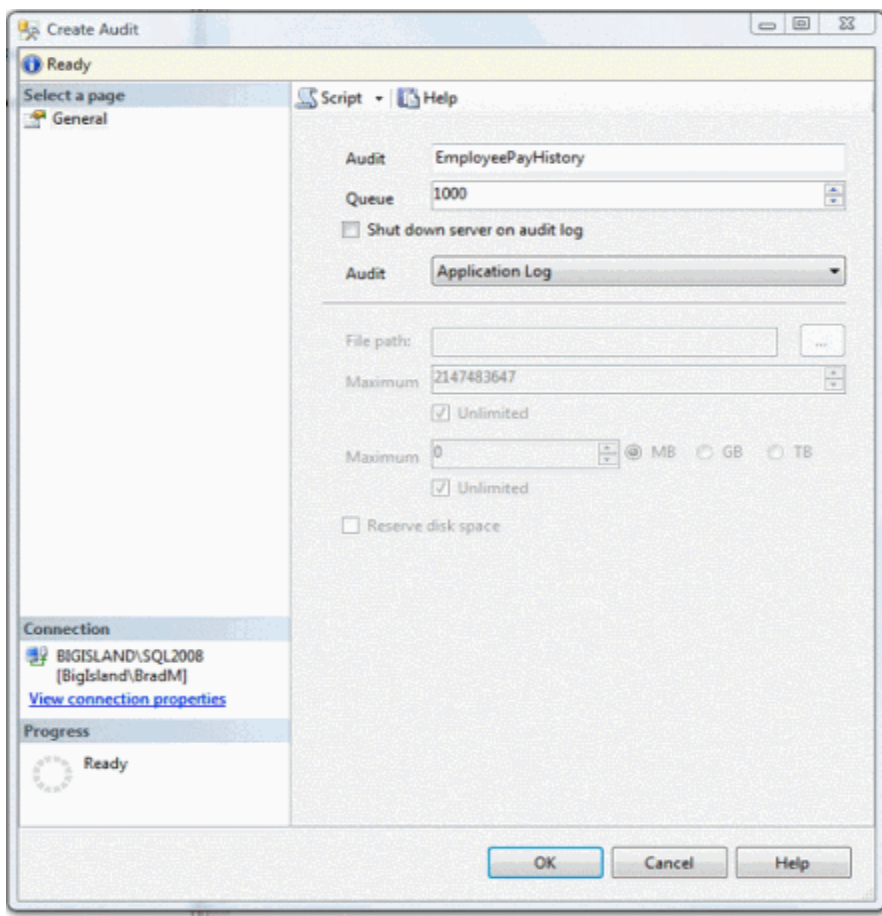


Figure 5: Once all the data has been provided, click “OK” to create the audit.

Now that the audit has been configured, click on “OK” to save it. It should then appear in the SSMS Object Browser, as shown in Figure 6:

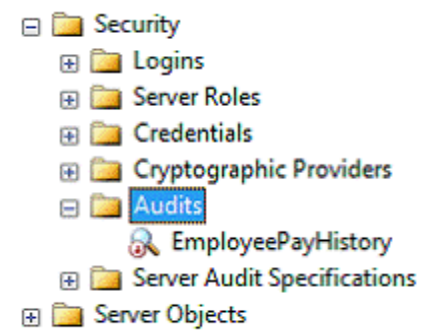


Figure 6: Notice the red arrow next to the newly created audit.

The red arrow next to the audit object means that it is not currently enabled. That’s OK for now, we can enable it later.

Creating a Server or Database Audit Specification

Now that we have created the audit, we need to create the matching audit specification. If we wanted to do an instance-wide audit, we would create a server audit specification. But for this example, where the goal is to audit the SELECT activity on a single table in a single database, a database audit specification is created.

To create a database audit specification using SSMS, open up the database to be audited, then open up the security folder under it. Next, right-click on “Database Audit Specifications” and select “New Database Audit Specification”, as shown in Figure 7:

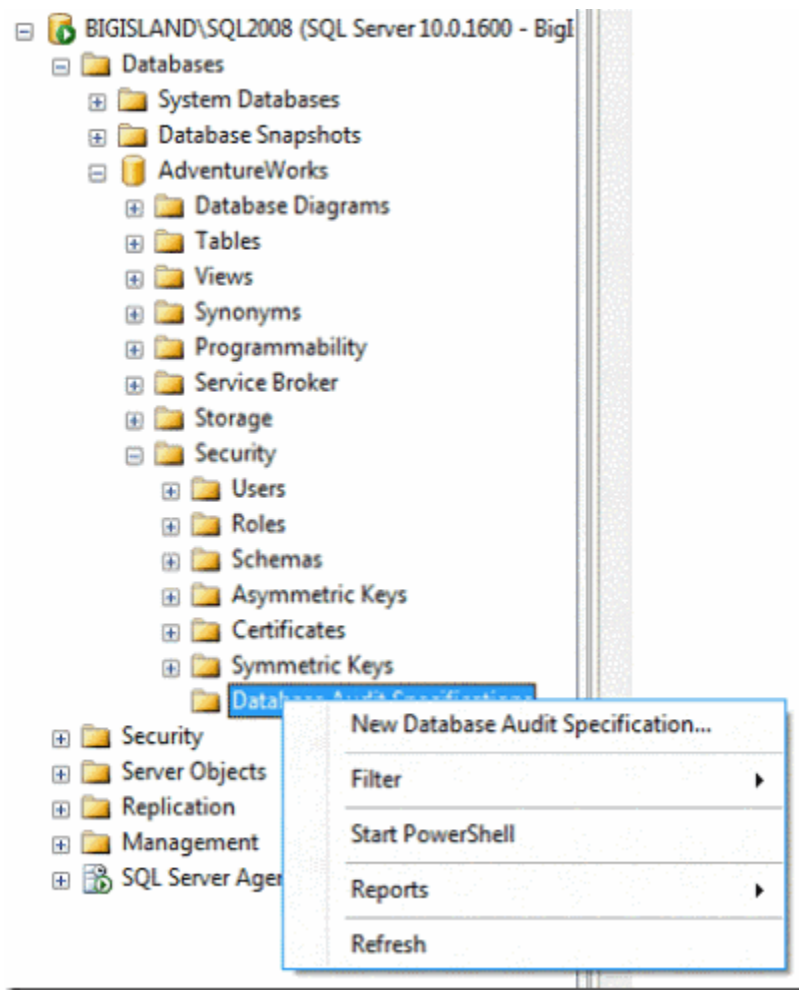


Figure 7: To create a database audit specification, you must do so from within the database you want to audit.

The “Create Database Audit Specification” dialog box appears, as shown in Figure 8:

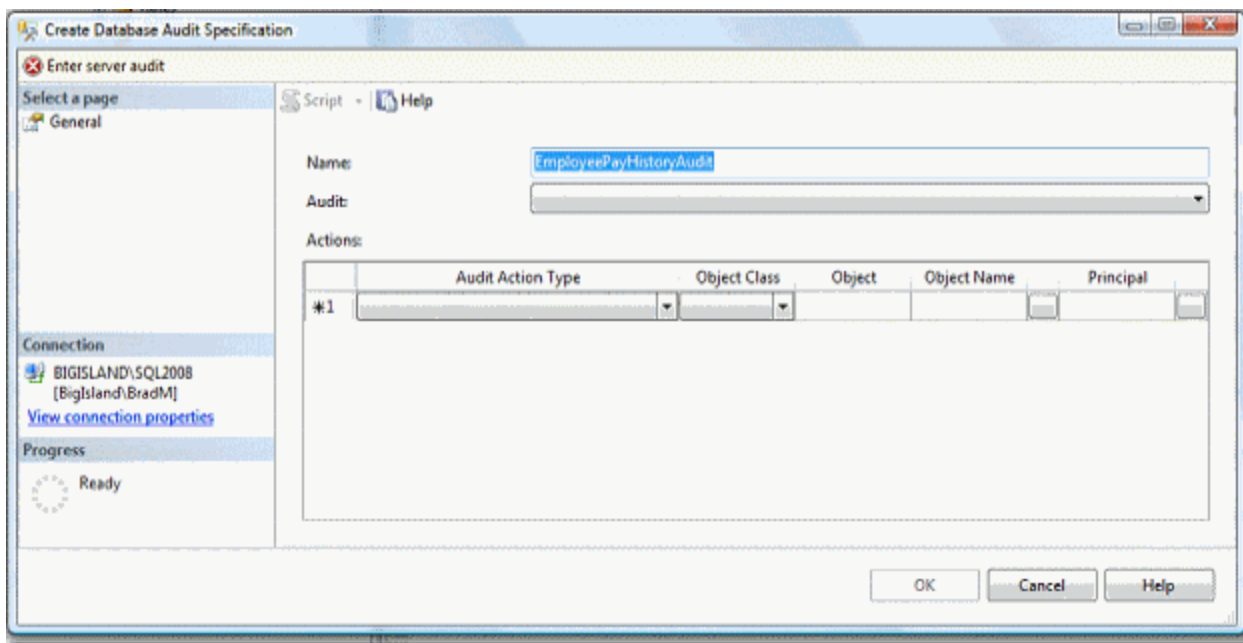


Figure 8: The “Create Database Audit Specification” dialog box has many options to complete.

You can either choose to accept the default name assigned to this database specification, or you can enter your own. Next, select the appropriate audit object from the Audit dropdown box, as shown in Figure 9:

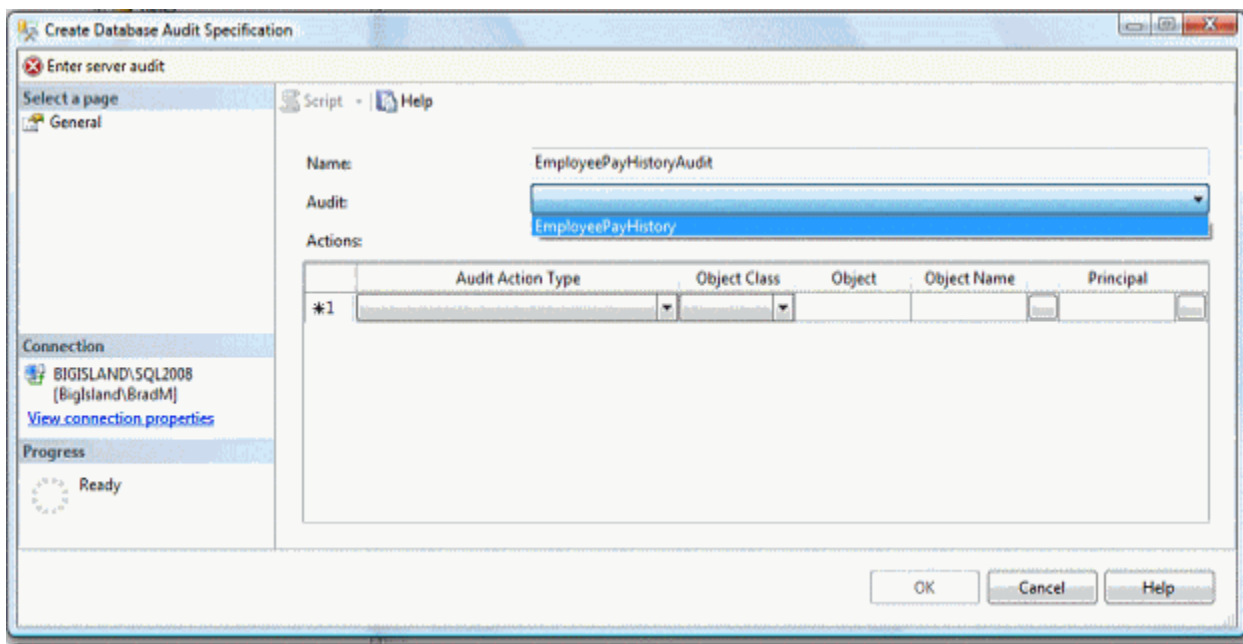


Figure 9: The “Create Database Audit Specification” dialog box has many options to complete.

In this case there is only one audit object, the “EmployeePayHistory”, as this is a newly installed SQL Server and doesn’t have any other audit objects on it.

Next, you must specify the kind of audit activity you want to capture by selecting from the “Audit Action Type” drop-down box, as shown in Figure 10:

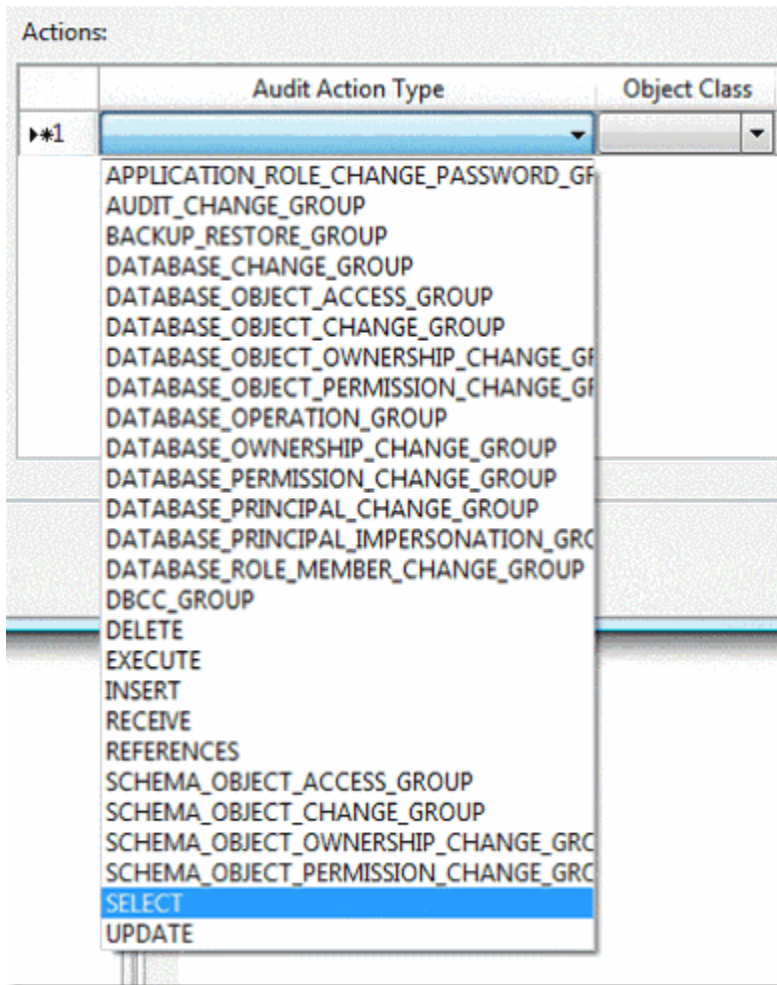


Figure 10: You can select from many pre-defined audit actions.

For this example, I want to choose the “SELECT” “Audit Action Type,” as the goal is to record all SELECT activity for the payroll table. Of course, you can choose any audit action type you want, but you can only choose from those that are listed. You can’t create your own.

Now that the audit action type has been chosen, the “Object Class” must be chosen – see Figure 11:

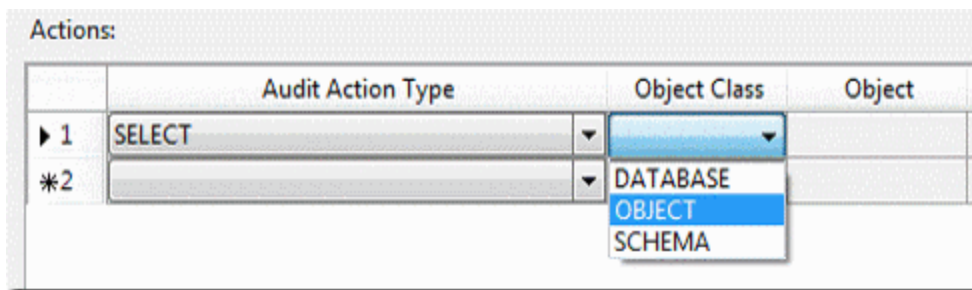


Figure 11: In this case, you can choose from three object classes.

The object class allows us to narrow down the scope of what we want to audit. For this example, because we want to monitor activity on a table, “Object” is selected.

The next step is to specify the object, or the table name, that is to be audited. To do this, click on the browse button under “Object Name,” and the “Select Objects” dialog box appears, as shown in Figure 12:

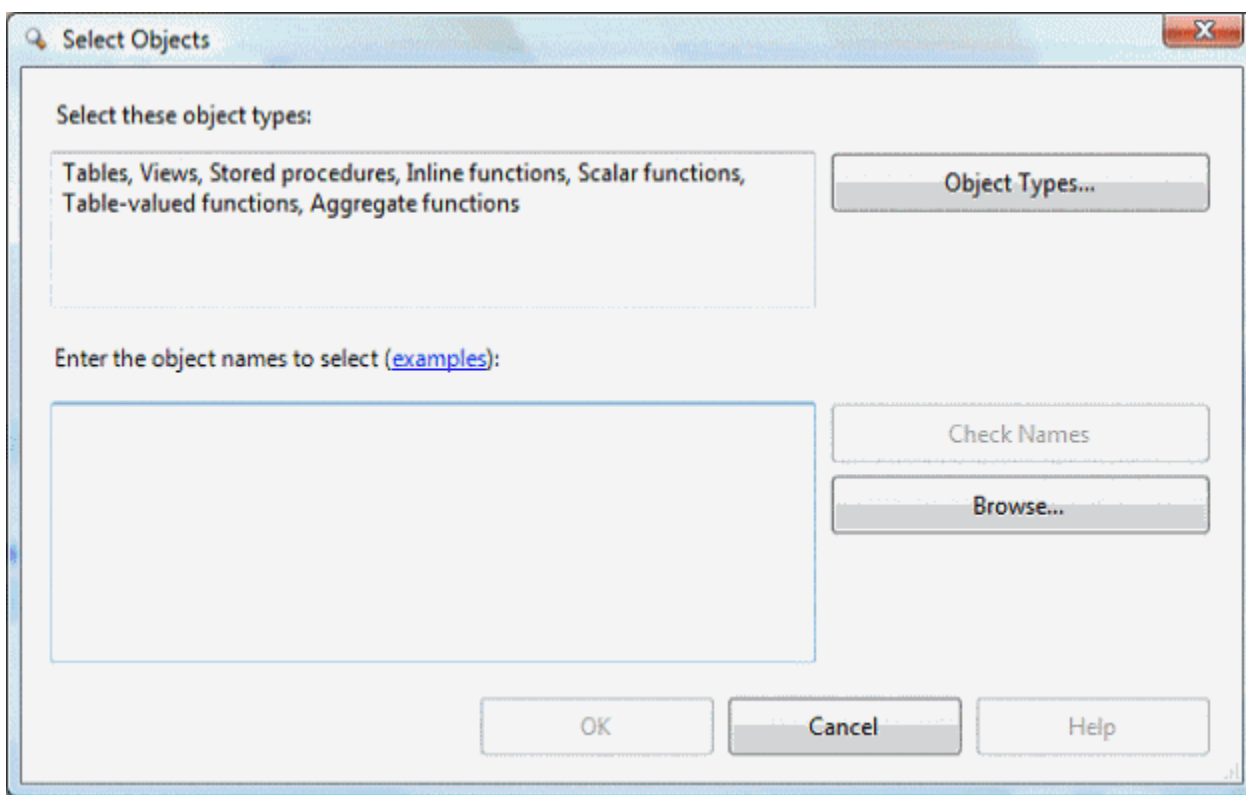


Figure 12: The “Select Objects” dialog box allows you to select which object to audit.

Having clicked on the “Browse” button, the list of available objects will appear, as shown in Figure 13:

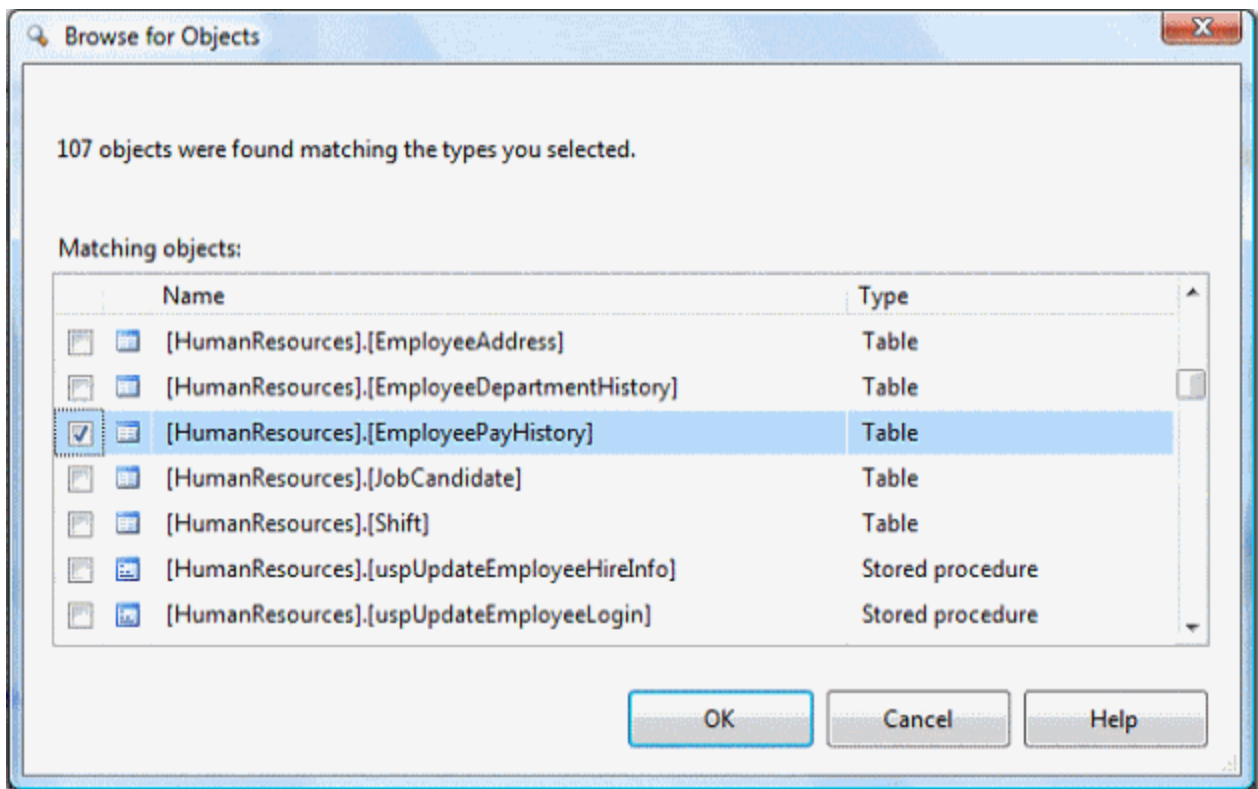


Figure 13: Select the object to be audited from this list.

Browse through the “Browse for Object” dialog box until you find the object or objects you want to audit, then select them. Above, I have selected a single table: HumanResources.EmployeePayHistory.

Once the objects have been selected, click “OK,” and the “Select Object” dialog box reappears, as shown in Figure 14:

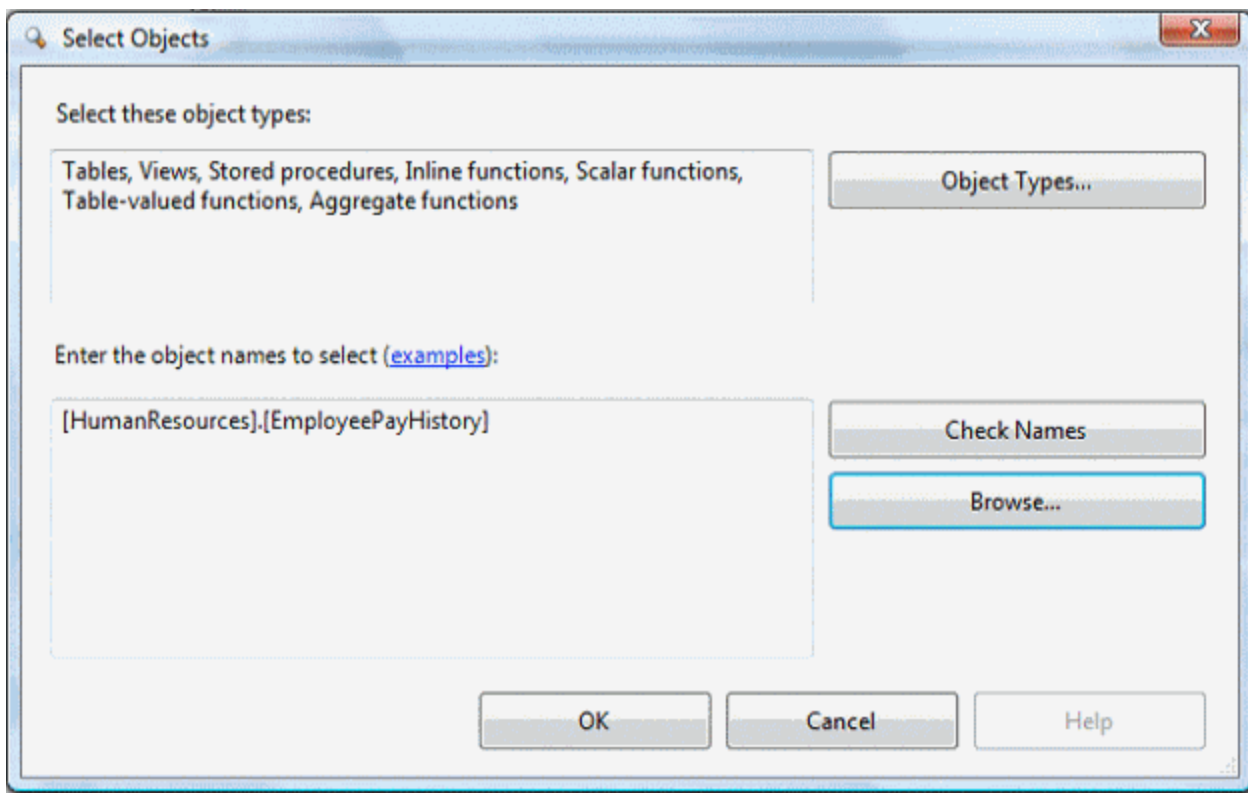


Figure 14: The audited object has been selected.

Now that the object to be audited has been selected, click “OK,” and you are returned to the original “Create Database Audit Specification” dialog box, as shown in Figure 15:

Audit Action Type	Object Class	Object	Object Name	Principal
SELECT	OBJECT	HumanResou...	EmployeePay...	...

Figure 15: We now see all of our actions up to this point.

There is one last step, and that is to specify what security principals (user accounts) that we want to monitor. To do this, click on the browse button under “Principal Name,” and another “Select Object” dialog box appears.

I am going to spare you seeing this screen again, and skip immediately to the “Browse for Object” dialog box, where you can see what principals you can choose from, as shown in Figure 16:

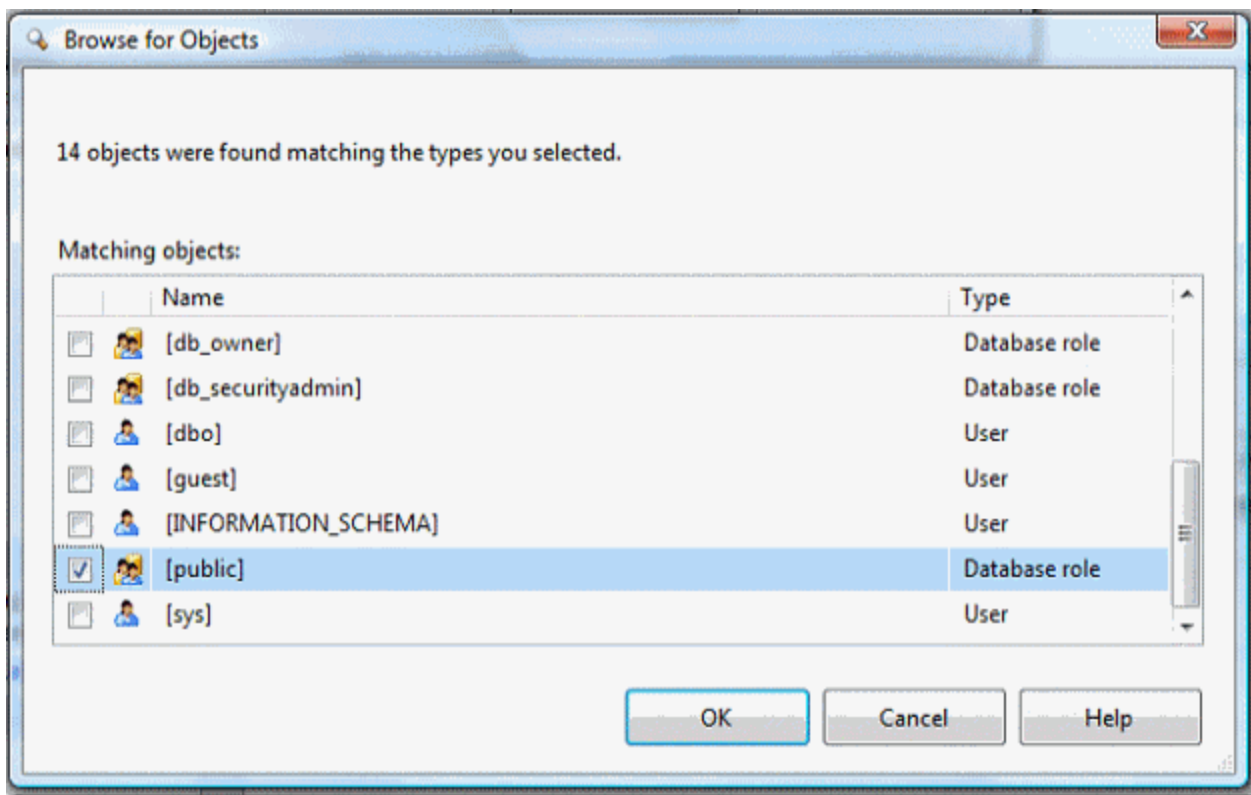


Figure 16: Select the principal you want to audit.

In this case, public is chosen, because the goal of this audit is to identify *anyone* who runs a SELECT against the payroll table. Optionally, you can select on specific users or roles. Click on “OK” for this dialog box, then click on “OK” for the “Select Objects” dialog box, and we reach the final screen, seen on Figure 17:

Name: EmployeePayHistoryAudit

Audit: EmployeePayHistory

Actions:

	Audit Action Type	Object Class	Object	Object Name	Principal
▶ 1	SELECT	OBJECT	HumanResou...	EmployeePay...	public
* 2					

OK Cancel Help

Figure 17: We are finally done creating the database audit specification.

Since we are only interested in auditing this one table for a single action, we will stop now. If you wanted to, you could continue to add additional actions and objects to this audit specification. Click on "OK," and the database Audit Specification will be saved, and you can view it in object explorer, as shown in Figure 18:

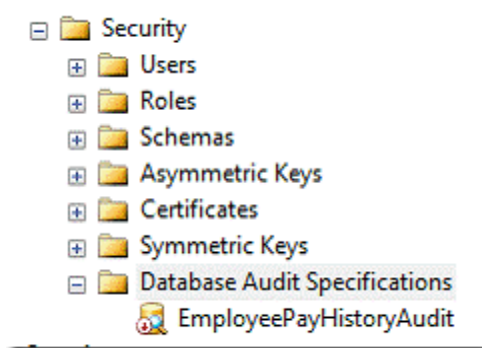


Figure 18: Notice the red arrow next to the specification, which tells us that it is turned off.

Once the new database audit specification has been created, it has a red arrow next to it, indicating that it is turned off. We will turn it on in the next step.

<http://bradmcgehee.com/2010/03/30/an-introduction-to-sql-server-2008-audit/>

NEW FEATURES LIST FOR 2005, 2008, 2008R2, 2012, 2014

SQL Server 2005	SQL Server 2008	SQL Server 2008R2
DB Mirroring	[SQL Server] Audit	Report Builder 3.0
Peer-to-peer replication	Activity Monitor	Master Data Services
Db snapshot	Backup Compression	PowerPivot for SharePoint
Service Broker	Data Collector and Management	SQL Server Utility
SSMS	Data Compression	StreamInsight
DMV's	Resource Governor	Multi Server Dashboards
Snapshot isolation	Policy-Based Management	Data-Tier Application
Database Mail	Transparent Data Encryption (TDE)	SQL Server 2008 R2 Parallel Data Warehouse
Analysis Services	MultiServer Query	SQL Server 2008 R2 Datacenter
Piecemeal restoration	IntelliSense for Query Editing	
Maintaince plans	File stream	
Resource database	Patch uninstantiation-Rolling upgrade	
Reports	Slipstream	
http://www.sqlusa.com/articles2005/top10operation/	http://sqlcat.com/sqlcat/b/top10lists/archive/2009/01/30/top-10-sql-server-2008-features-for-the-database-administrator-dba.aspx	http://www.databasejournal.com/features/mssql/article.php/3857466/Top-10-Features-of-SQL-2008-R2.htm

SQL SERVER 2012**SQL SERVER 2014****AlwaysOn Availability Groups**[Memory-Optimized Tables](#)**Contained Databases**[SQL Server Data Files in Windows Azure](#)**ColumnStore Indexes**[Host a SQL Server Database in a
Windows Azure Virtual Machine](#)**ShowPlan Enhancements**[AlwaysOn Enhancements](#)**User-Defined Server Roles**[Buffer Pool Extension](#)**Enhanced Auditing Features**

Enhancements to Backups[SQL Server 2014 also provides new Windows Azure integration to SQL Server's backup capabilities]

Enhanced PowerShell Support**Your database on Azure****Distributed Replay**

Database Compatibility Level[The 90 compatibility level is not valid in SQL Server 2014]

PowerView**SQL Azure Enhancements**

<http://mcpmag.com/articles/2012/03/14/top-12-features-of-sql-server-2012.aspx>

<http://sqlmag.com/sql-server-2014/sql-server-2014-important-new-features>

DMV'S & SP & DBCC

DMV'S [DYNAMIC MANAGEMENT VIEWS]

- > This concept is introduced in SQL Server 2005 version.
- > The main purpose we can monitor SQL Server without consuming hardware resources like DBCC queries.
- > The DMV'S newly introduced in SQL Server 2005 gives the database administrator information about the current state of the SQL Server machine.
- > These Values will help the administrator to diagnose problems and tune the server for optimal performance.
- > The DMV'S in SQL Server are designed to give you a window into what's going on inside SQL Server
- > They can provide information on what's currently happening inside the server as well as the objects it's strong. They are designed to be used instead of system tables and various functions.
- > DMV'S are stored in sys schema and they start with **dm_in** the name

To know list of DMV'S:

```
SELECT name, type, type_desc FROM sys.system_objects WHERE name LIKE 'dm_%' ORDER BY name
```

Output:

List of DMV'S

2005 version of Sql -89..... 2008 version of sql-176.....2012 version of sql-900+

There are two types of dynamic management views and functions:

Server-scoped dynamic management views and functions: These require VIEW SERVER STATE permission on the server.

Database-scoped dynamic management views and functions: These require VIEW DATABASE STATE permission on the database.

There are multiple categories close to 17... In which these views and functions have been organized

We have 85 of these views and functions. To give a further split, 76 of these are views and 9 of them are functions... Below are the 4 types of DMV which can be used frequently.

1. SQL Server Related [Hardware Resources] DMV'S
2. Database Related DMV'S
3. Index Related DMV'S
4. Execution Related DMV'S
5. Replication Related DMV'S
6. Query notifications Related DMV'S
7. SQL Operating System Related DMV'S
8. I/O Related DMV'S
9. Transaction Related DMV'S

1. SQL Server related [Hardware Resources] DMV'S:

> This section contains the dynamic management views that are associated with the SQL Server Operating System (SQLOS). The SQLOS is responsible for managing operating system resources that are specific to SQL Server.

Locks:

Select * from Sys.dm_tran_locks:

Returns information about locks

Blockings:

Select * from sys.dm_os_waiting_tasks:

> Returns information about the wait queue of tasks that are waiting on some resource.

Sys.dm_os_wait_stats:

> Returns information about all the waits encountered by threads that executed. You can use this aggregated view to diagnose performance issues with SQL Server and also with specific queries and batches.

2. Database Related DMV'S:**Mirroring:****1. Sys.dm_db_mirroring_auto_page_repair:**

1. Returns a row for every automatic page-repair attempt on any mirrored database on the server instance.
2. This view contains rows for the latest automatic page-repair attempts on a given mirrored database, with a maximum of 100 rows per database.

2. Sys.dm_db_mirroring_connections:

Returns a row for each connection established for database mirroring.

3. INDEX related DMV'S

Fragmentation: DMV's to find 2005 onwards

1. Select * from sys.dm_db_index_physical_stats:

To find Column to verify the fragmentation value:

Avg_fragementaion_in_pernt:

Missing Index:**Select * from sys.dm_db_missing_index_details:**

Returns detailed information about missing indexes, excluding spatial indexes.

sys.dm_db_index_usage_stats:

Returns counts of different types of index operations and the time each type of operation was last performed in SQL Server.

4. Execution related DMV'S:**Sys.dm_exec_cached_plans:**

> Returns a row for each query plan that is cached by SQL Server for faster query execution. You can use this dynamic management view to find cached query plans, cached query text, the amount of memory taken by cached plans, and the reuse count of the cached plans.

Sys.dm_exec_connections:

> Returns information about the connections established to this instance of SQL Server and the details of each connection.

Sys.dm_exec_sessions:

> shows information about all active user connections and internal tasks. This information includes client version, client program name, client login time, login user, current session setting, and more.

Sys.dm_exec_cursors:

> Returns information about the cursors that are open in various databases.

5. Replication related DMV'S:**Sys.dm_repl_articles:**

> Returns information about database objects published as articles in a replication topology.

Sys.dm_repl_tranhash:

> Returns information about transactions being replicated in a transactional publication.

Sys.dm_repl_schemas:

> Returns information about table columns published by replication.

Sys.dm_repl_traninfo:

> Returns information on each replicated or change data capture transaction.

STORED PROCEDURES IN SQL SERVER

> A stored procedure is a group of Sql statements that has been created and stored in the database. Stored procedure will accept input parameters so that a single procedure can be used over the network by several clients using different input data. Stored procedure will reduce network traffic and increase the performance. If we modify stored procedure all the clients will get the updated stored procedure

In SQL we are having different types of stored procedures are there

- a) System Stored Procedures
- b) User Defined Stored procedures
- c) Extended Stored Procedures

System Stored Procedures:

System stored procedures are stored in the master database and these are starts with a sp_ prefix. These procedures can be used to perform variety of tasks to support Sql server functions for external application calls in the system tables and use to perform many administrative and informational activities.

Ex: sp_helptext [StoredProcedure_Name]

User Defined Stored Procedures:

User Defined stored procedures are usually stored in a user database and are typically designed to complete the tasks in the user database. While coding these procedures don't use sp_ prefix because if we use the sp_ prefix first it will check master database then it comes to user defined database

Stored procedure are modules or routines that encapsulate code for reuse. A stored procedures can take input parameters, return tabular or scalar results and messages to the client.

Extended Stored Procedures:

Extended stored procedures are the procedures that call functions from DLL files that an instance of Microsoft SQL Server can dynamically load and run. Now a day's extended stored procedures are depreciated for that reason it would be better to avoid using of Extended Stored procedures.

Why use Stored Procedures?

- > Rewriting inline SQL statements as Stored Procedures
- > Compilation and storing of the query execution plan
- > Enabling of conditional and procedural logic
- > Centralized repository for DML and DDL code enabling code reuse
- > Protection from SQL Injection attacks
- > Enabling of strict security model
- > Readability

Sample of creating Stored Procedure

```
USE AdventureWorks2008R2;
GO
CREATE PROCEDURE dbo.sp_who
AS
    SELECT FirstName, LastName FROM Person.Person;
GO
EXEC sp_who;
EXEC dbo.sp_who;
GO
DROP PROCEDURE dbo.sp_who;
GO
```

Advantages of using stored procedures

- a) Stored procedure allows modular programming.

You can create the procedure once, store it in the database, and call it any number of times in your program.

- b) Stored Procedure allows faster execution.

If the operation requires a large amount of SQL code is performed repetitively, stored procedures can be faster. They are parsed and optimized when they are first executed, and a compiled version of the stored procedure remains in memory cache for later use. This means the stored procedure does not need to be reparsed and reoptimized with each use resulting in much faster execution times.

c) Stored Procedure can reduce network traffic.

An operation requiring hundreds of lines of Transact-SQL code can be performed through a single statement that executes the code in a procedure, rather than by sending hundreds of lines of code over the network.

d) Stored procedures provide better security to your data

Users can be granted permission to execute a stored procedure even if they do not have permission to execute the procedure's statements directly.

DBCC (DATABASE CONSOLE COMMANDS) COMMANDS

> DBCC commands are used to check the consistency of the database or database objects. While executing DBCC commands the DB engine creates a database snapshot and then runs the checks against this snapshot. After the DBCC command is completed, this snapshot is dropped.

> It's consuming hardware resources. The DBCC commands are most useful for performance and troubleshooting exercises.

DBCC commands broadly falls into four categories:

- **Maintenance**
- **Informational**
- **Validation**
- **Miscellaneous**

MAINTENANCE COMMANDS:

Perform the maintenance tasks on a database, index, or filegroup.

CLEANTABLE:

> Reclaims space from dropped variable-length columns in tables or indexed views.

DBCC CLEANTABLE ('Database name', 'Table name', size)

DBREINDEX:

> Rebuilds one or more indexes for a table in the specified database.

DBCC DBREINDEX ('Table name', 'Index name', Fill factor)

DROPCLEANBUFFERS:

> Removes all clean buffers from the buffer pool.

DBCC DROPCLEANBUFFERS

FREEPROCCACHE:

> Removes all elements from the plan cache, removes a specific plan from the plan cache by specifying a plan handle or SQL handle, or removes all cache entries associated with a specified resource pool.

DBCC FREEPROCCACHE

INDEXDEFRAG:

> Defragments indexes of the specified table or view.

DBCC INDEXDEFRAG ('Database name', 'Table name', 'index name, partition number')

SHRINKDATABASE:

> Shrinks the size of the data and log files in the specified database.

DBCC SHRINKDATABASE ('Database name', target percentage)

SHRINKFILE:

> Shrinks the size of the specified data or log file for the current database, or empties a file by moving the data from the specified file to other files in the same filegroup, allowing the file to be removed from the database. You can shrink a file to a size that is less than the size specified when it was created. This resets the minimum file size to the new value.

DBCC SHRINKFILE ('File name', target percentage)

INFORMATIONAL COMMANDS:

Performs tasks that gather and display various types of information.

CONCURRENCYVIOLATION:

> Is maintained for backward compatibility. It runs but returns no data.

DBCC CONCURRENCYVIOLAION

UPDATEUSAGE:

> Reports and corrects pages and row count inaccuracies in the catalog views. These inaccuracies may cause incorrect space usage reports returned by the sp_spaceused system stored procedure.

DBCC UPDATEUSAGE ('Database name')

INPUTBUFFER:

> Displays the last statement sent from a client to an instance of Microsoft SQL Server 2005.

DBCC INPUTBUFFER (session id)

OPENTRAN:

> Displays information about the oldest active transaction and the oldest distributed and nondistributed replicated transactions.

DBCC OPENTRAN ('database name')

OUTPUTBUFFER:

> Returns the current output buffer in hexadecimal and ASCII format for the specified session_id.

q

DBCC OUTPUTBUFFER (session id)

PROCCACHE:

> Displays information in a table format about the procedure cache.

DBCC PROCCACHE

SHOW_STATISTICS:

> Displays the current distribution statistics for the specified target on the specified table.

DBCC SHOW_STATISTICS (table or index view name', target)

SHOWCONTIG:

> Displays fragmentation information for the data and indexes of the specified table or view.

DBCC SHOWCONTIG ('table name')

SQLPERF:

> Provides the transaction log space usage statistics for all databases. It can also be used to reset wait and latch statistics.

DBCC SQLPERF (LOGSPACE)

TRACESTATUS:

> Display the status of trace flags.

DBCC TRACESTATUS (Trace number)

USEROPTIONS:

> Returns the SET options active (set) for the current connection.

DBCC USEROPTIONS

VALIDATION COMMANDS:

> Performs validation operations on a database, table, index, catalog, filegroup, or allocation of database pages.

CHECKALLOC:

> Checks the consistency of disk space allocation structures for a specified database.

DBCC CHECKALLOC ('Database name')

CHECKCATALOG:

> Checks for catalog consistency within the specified database.

DBCC CHECKCATALOG ('Database name')

CHECKCONSTRAINTS:

> Checks the integrity of a specified constraint or all constraints on a specified table in the current database.

DBCC CHECKCONSTRAINTS WITH ALL_CONSTRAINTS

CHECKDB:

> Checks the logical and physical integrity of all the objects in the specified database.

DBCC CHECKDB ('Database name')

CHECKFILEGROUP:

> Checks the allocation and structural integrity of all tables and indexed views in the specified filegroup of the current database.

DBCC CHECKFILEGROUP FILEGROUPNAME

CHECKIDENT:

> Checks the current identity value for the specified table and, if it is needed, changes the identity value.

DBCC CHECKIDENT ('table_name')

CHECKTABLE:

> Checks the integrity of all the pages and structures that make up the table or indexed view.

DBCC CHECKTABLE ('table_name')

REAL TIME CLASSES

REAL TIME CLASSES

PROCESS:

- 1 **ITIL** (Information Technology Infrastructure Library) Process
- 2 **DB** Maintains Activities(Daily\weekly\Monthly)
- 3 **DR** (Disaster Recovery) Plan
- 4 **BCP** (Business continuity Plan) Plan
- 5 **RACI** (Responsible Accountable Consult Inform)Matrix
- 6 **RCA** (Root Cause Analysis) Plan
- 7 **SLA** (Service level agreement)
- 8 Capacity planning\management
- 9 Interview Handling
- 10 Day-to-Day Activities
- 11 General Responsibility
- 12 Ticket, Monitoring Tool
- 13 Escalation Matrix
- 14 KT Questions
- 15 On-call & Bridge call

ITIL (Information Technology Infrastructure Library) Process:

- It is a set of good practices and it is not a standard
- ITIL V1, V2 and V3. Present works in V3 and remain all are expired
- ITIL has 5 phases/sections
 - ❖ Service strategy
 - ❖ Service design
 - ❖ Service transactions
 - ❖ **Service operations**
 - ❖ Service improvement

Service operations

1. Incident Management----- job fail
2. Change Management----- service pack,filemovements,adding file, Configuring Highavailbilty --- any planned activity comes under
3. Problem Management----- restarting of Sql

System is not available for business or end users “Down Time”

Incident Management (or) call (or) Alert

- Incident means Issue
- Incident is a unexpected interruption to running service and it can cause major\minor damage to the business
- Incident is sometime bug, ticket, complaint
- Incident has categories i.e. critical(p1), high(p2), medium(p3) and low(p4)

Service Level Agreement: An agreement between client and company

Type	Response	Resolve
P1	15min	4hrs[Platinum]
P2	30min	8hrs [Gold]
P3	30min	2 days [silver]
P4	1 day	3days [bronze]
P5	1 Day	7 Days[plastic]

Office network connectivity tools:

VPN: Virtual Private Network
CITRIX

Servers:

Production: Any OLTP transactions.

Pre-production: Any server before going to the live we can called as pre-production server

Development:

Testing:

Note:

1. You can access this tool by using a web URL
1. Every DBA can have user name and password to login into BCM REMEDY ticketing tools.
2. Alerts calls assigned to DBA queue after L1 team assign\ distribute the calls based on the resource availability.

Progress steps for incident:

New >>Open>>Assign>> in progress [SLA going to start]>>Pending [SLA stop calculation]>>Resolved>>Cancled

Incident Management is unplanned task

- System raised
- User raised

System raised: This type of alerts raised by monitoring tool... i.e. BMC Patrol, and SCOM (System Centre Operation Manager)

How it works:

When any error tracks into SQL server logs or event viewer\agent error logs then patrol agent per each server can monitor and raise alert to BMC remedy to the based on the priority.

User raised: Depending on the user requirement to raise a call based on the impact to select priority in SQL Server.

Note: Always work on any incidents based on the priority and SLA's

How to login into servers?

- Go to MSTC [Microsoft terminal service connection] or remote desktop processing>> provide FQDN [Fully qualified domain name] (or) server ip to login.
- Provide user name and password domain name then connect to login into servers.

Change Management:

Change Management Tool

BMC Remedy

CPMS: Change Process Management Service

- Change Management is introducing modifications to an existing environment
- It's a planned activity
- Its having 3 stages
 - ❖ **Normal Changes (CR)::** Ex: File movement, adding file, changing memory parameters, High availability configurations...
 - ❖ **Standard Changes (S-CR) ::** Patch management, DR Testing
 - ❖ **Emergency Changes (E-CR) ::** disk full ,100% cpu utilization, log file full, Database\Sql service down, Master database down

Change Management Process:

1. Get all the requirements including down time details from apps team
2. Raise a change as per the plan
3. Once change raise, pull all technical steps including pre-installation, implementation, and verification and backout plan.
4. Once completed, get technical side approval
5. Once approved, get SDM Approval.[Delivery manager approval\client approval]
6. After implement the change as per implementation date and finish as per the time lines.
7. Once implement success then Change is success else failed.

Change States:

Raise Change> Impact assessment> File the technical steps> Pass for technical approval> Pass to SDM approval> Once Approved,

Problem Management:

Repetitive or Recurring issues we can fix by using “Problem Management”

- Problem Management handles recurring incidents. Problem Management Naming convention is ‘PRB’
- Identify the causes, resolve if it is from root

- Generally Sr.L3/L4 will be doing problem management
- For every Problem once it is resolved requires a RCA (Root cause analysis)

Note: PB tickets handled by Pb management coordinators by setting up the call, getting updates from all team. On time all members should provide the correct updates.

Capacity planning\management:

- Forecasting the growth of database is called Capacity planning.
- It is nothing but a Disk planning
- Volume reporting is giving 1 month is estimation report and it is given to 1 year to allocated the disk space

Server\Database Hardening rules: Harding is to protect the server from threats

Harding is to protect the server from threats

Rules:

1. **Remove un necessary roles to the users\logins**
2. **Provide strong sa password**
3. **Disable built-in administrator account**
4. **Remove unnecessary users\logins**
5. **Always create instead of individual logins create groups**

BCP (Business continuity Plan) Plan:

- BCP is to defines and perform the day-to-day operations at multiple locations.
- If one server is down then another server is do the process
- Every year once one resource move to another location and work from there, and end of the same day has to provide a clear BCP team and again this is ITIL process.

RCA (Root Cause Analysis) Plan:

- This is related to Problem management
- In root cause analysis
 - ❖ Issue/problem
 - ❖ Solution
 - ❖ Measurement
 - ❖ Mitigate not to happen again the same issue in server.

RACI Matrix: Defines Roles and responsibilities from each team which relate to each team.

- ❖ R- Responsibility
- ❖ A-Accountable

q

- ❖ C-Consult
- ❖ I-Inform
- RACI is a simple Excel sheet

DBA Maintains Activities:

Daily Maintainece: Differential backup, Transaction log backup, Blocking jobs, CPU Utilization

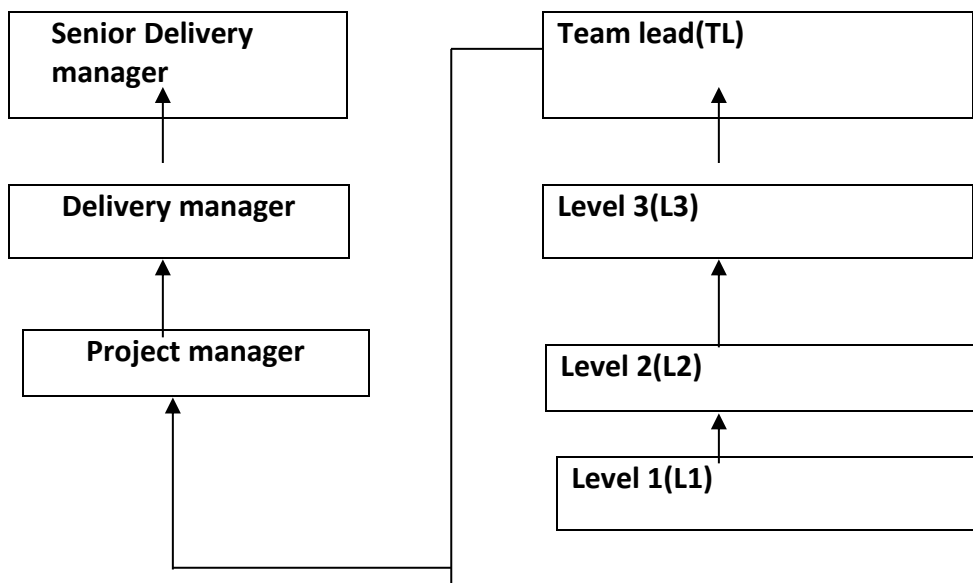
History cleanup job

Weekly Maintainece: Full backup, Rebuild indexes, Reorganize index, DBCC Checkdb, Purging jobs,

Monthly Maintainece: Patch management, Backup file testing

Yearly Maintainece: DR Testing, BCP plan

Escalation Matrix (or) Reporting Structure:



When you complaint against a person ensure you have follow process as per reporting structure.

Note: Direct and final compliant can be given HR

SLA [Service Level Agreement]: This is the agreement between client and company

SLA:

Type	Response	Resolve
P1	15min	4hrs
P2	30min	8hrs
P3	30min	1 days
P4	1 day	2days
P5	1 Day	5 Day

Disaster Recovery Plan:

> This defines a business which needs to run continuously without having any service interruption.

Ex: By implementing mirroring, replication, and clustering.

Real time: DR test will happen every 1 year or 6 months.

Team size: 12

24*7

L1→ 3

L2→4

L3→ 3

Team lead: 1

Manager: 1

Shift Management:

8 Hr: 6:00 AM- 4:30 PM: 1 L1+1 L2+ 1 L3

8 Hr: 12:00 PM-9:30 PM: 1 L1+2 L2+ 1 L3

8 Hr: 9:00 PM -06:30 AM: 1 L1+1 L2+ 1 L3

Shift handover:

To handover to next shift person,

1. What are the new requests came into our queue
2. What are the tasks are pending
3. What are the calls need to attend
4. How many tickets are closed?
5. How many critical tickets are came in shift?

How many number servers in your project support? 500 +

How many prod and dev\uat\pre-prod servers? 350+ Prod and UAT\PRE-PROD\DEV-150+

How many Sql instances are installed? 700+ instance

How many database are? 5000+ databases

What is the max database size: 4 TB?

Backup strategy:

Full backup: Daily [After business hours]

Log backup: Daily [Every 15 min once]

How many database\instance configured with HA solutions?

Logshipping: 100 +

Mirroring: 200+

Replication: 2000+

Clustering: 100+ Instance

Backup file retention period?

In disk: 7 days

In Tape: 2 months

Old backups automatically deleted from disk or tape by using clean up job.

DBA Daily Maintaine Activities:

Output File Cleanup

Cycle Error Log

Database Backup - USER_DATABASES – FULL

Database Integrity Check - SYSTEM_DATABASES

Database Backup - USER_DATABASES – LOG

DatabaseIntegrityCheck - SYSTEM_DATABASES

DatabaseIntegrityCheck - User_DATABASES

DBA Weekly maintaine activities:

OS LEVEL SNAP Backup

Index Optimize - USER_DATABASES

Servers reboot on request basis

DBA monthly maintaine activities:

Database growth report by monthly job as per schedule every month 1st.

Database refresh

q

Do we have a client interaction?

Yes, daily we have call with client

What you discuss?

How many P1 or P2 reported today or last week or last month?

How many changes are implemented today or last week or last month?

What are the current on-going issues?

Any improvements need to perform in technical side or process side?

Any client comments?

Retention period Of SQL Server agent and Sql server logs?

1 month SQL Server logs can be maintained

10 Days Sql server agent error log can be maintained.

ROLES AND RESPONSIBILITIES:

1. I am currently supporting client "PROJECT NAME" with 1000 servers including
 - 700 production servers
 - 100 development servers
 - 200 pre-production, staging or testing servers.
2. The project belongs to US\UK with 24*7 supporting mode.
3. I am dividing my roles and responsibilities into 2 categories.
 1. Technical Roles
 2. Process roles
4. **In technical roles**, Performing SQL Server various version installation of 2005, 2008, 2008 R2 and 2012 along the configurations.
5. Applying service packs, hot fixes and cumulative updates for various versions and if any failures trouble shoots accordingly.
6. Whenever if we have any issues with file and file groups, to resolve adding the over flow files to database.
7. As a part of security, creation of users, login and providing relevant Permissions to the requestor.
8. Implementing best security hardening rules in production servers.
9. Configured different types of backups for various versions and involved in recovery scenarios if database crashes or depending on the requirement.
10. Defining the best backup strategy especially for production servers.
11. Suggest to application team on recovery models for database to resolve log file related issues especially on production servers.
12. Performed system and user database file movements whenever if we have any disk space crunch.
13. Involve recovering of system and user database if any corruption happened especially MASTER REBUILDING, model, msdb and tempdb.
14. Performed attach\detach, import \export, copy database techniques.
15. Configured jobs, maintainece plans, and dB mail to send any notification to end user on production servers.
16. As a part high availability and disaster recovery solutions, configured log shipping, db mirroring, and replication and clustering.
17. Involved in UPGRADATION and Migration activities and resolved if any ORPAN user issues.
18. As a part of performance tuning,
 - Resolving blockings
 - Resolving deadlocks
 - Maintain indexes
 - Query tuning

q

- Running Sql server profiler to capture event classes
- Running performance monitor tool to capture counters

These all my technical roles and responsibilities

Process roles:

We need to write

Daily Responsibilities:

- 1) Check the Shift Handover.
- 2) Check for any task continuity support needed.
- 3) Verify the Ticket Queue
- 4) Check the mails
- 5) Perform Health Checks (Critical Servers)
- 6) Perform any Change Requests during your shift (if any scheduled).
- 7) In an Offshore-Onsite model, attending Status Calls.

Health Checks:

- 1) Instance running or not
- 2) Agent running or not
- 3) Reading Error/Agent Logs
- 4) Checking Backups
- 5) Checking disk space
- 6) HA Synchronization/failure checks
- 7) Event Viewer
- 8) SQL Server settings
- 9) Database settings
- 10) High availability checks (failing (or) not)
- 11) SQL Server Error logs
- 12) Storage

These are the task list that must be done by a DBA on daily basis.

- 1) Check System Event Logs and SQL Server Error Logs for unusual events.
- 2) Verify that all scheduled jobs have run successfully.
- 3) Confirm that backups have been made successfully and moved to the backup storage or secure location.
- 4) Monitor disk space to make sure SQL Servers will not run out of disk space.
- 5) Periodically monitor performance of database and system using both System Monitor and SQL Server Profiler.
- 6) Monitor and identify blocking issues.
- 7) Keep a log of any changes you make to servers, instance setting, database setting and monitoring script.
- 8) Create SQL Server alerts to notify you of problems, and e-mailed to you. Take action as needed.
- 9) Regularly restore backups to a test server in order to verify that you can restore them.

Monthly Check list?

1. Make a list of all SA passwords for each server it in secure place.
2. Make a list of all the passwords for each login crated on the production boxes.
3. We save the SQL Servers and Windows configuration information in secure place. This information is needed to rebuild an NT& SQL Server box in case of a disaster
4. Perform a test restore of a database backup. This is done in order to prepare for unforeseen situations.
5. We save information about any changes made to a server – hardware or software.
6. Maintain system logs in a secure fashion. Keep records of all service packs installed for both Microsoft Windows NT Server and Microsoft SQL Server. Keep records of network libraries used, the security mode, SA passwords and service accounts.
7. Valid that no error messages are generated during the restore process.
8. Set the MSSQL and SQL Server Agent services to Auto-Start when the server starts.

What kind of Tickets will we get?**P1 Tickets:**

- 1) Instance Crashed (Prod)
- 2) Corruptions (Disk)
- 3) Hardware Failures
- 4) Suspect State of Database

P2 Tickets:

- 1) Application Performance Slow
- 2) Long Running Queries
- 3) Blockings
- 4) Deadlocks
- 5) Log Space Used 95%
- 6) Tempdb Usage 95%
- 7) Space Issues
- 8) Connectivity Issues (User Raised Ticket)
- 9) Permission issues
- 11) Connections are full
- 12) Page Corruptions
- 13) CPU Utilization 100%
- 14) Memory Utilization High
- 15) Buffer Cache Hit Ratio < 50%

