SQL Server Architecture

Can you explain SQL Server Architecture?

There are total 4 major components in SQL Server Architecture:

- Protocol Layer
- Query Processor/Relational Engine
- Storage Engine
- SQLOS

Each and every instruction must be interact with these 4 components. High level SQL Server Architecture can be explained in 4 steps as below:

- The Protocol Layer receives the request from the end client and translates it into the form where SQL Server Relational Engine can understand and work on it.
- The Query Processor accepts the T-SQL batches processes and executes the T-SQL batch. If data is required the request is passed to Storage Engine
- The Storage Engine manages the data access and service the requested data.
- The SQLOS takes responsibility of operating system and manages locks, synchronization, buffer pool, memory, thread scheduling, I/O etc.

What is a Page?

- The fundamental unit of data storage in SQL Server is the page.
- Page size is 8kb and 128 pages = 1 MB.
- Page starts with the header of 96 bytes that is used to store page number, page type, the amount of free space on the page, and the object id that owns the page. The maximum size of a single row on a page is 8060KB.

What is ROW_OVERFLOW_DATA allocation unit?

When the total row size of all fixed and variable columns in a table exceeds the 8,060 byte limitation, SQL Server dynamically moves one or more variable length columns to pages in the ROW_OVERFLOW_DATA allocation unit.

This is done whenever an insert or update operation increases the total size of the row beyond the 8060 byte limit. When a column is moved to a page in the ROW_OVERFLOW_DATA allocation unit, a 24-byte pointer on the original page in the IN_ROW_DATA allocation unit is maintained. If a subsequent operation reduces the row size, SQL Server dynamically moves the columns back to the original data page.

What is an EXTENT?

- An extent is eight physically contiguous pages or 64 KB.
- That means 16 Extents = 1 MB.
- There are two types of Extents. Uniform Extents and Mixed Extents.
- Uniform extents are owned by a single object;
- Mixed extents are shared by up to eight objects

What are the different types of Pages?

PFS (Page Free Space): Percentage of free space available in each page in an extent. The first page on any database data file is always a PFS page.

GAM and SGAM (Global Allocation Map & Shared GAM):

- GAM: Extents have been allocated: 1 – Free space 0 – No space
- SGAM: Mixed Extents have been allocated: 1 – Free Space + Mixed Extent and 0 – No space
- Each GAM/SGAM covers 64000 extents - 4 GB.
- The second page on any database data file is always a GAM page
- The third page on any database data file is always a SGAM page.

DCM (Differential Changed Map): This tracks the extents that have changed since the last BACKUP DATABASE statement. 1 – Modified, 0 – Not modified
BCM (Bulk Changed Map): This tracks the extents that have been modified by bulk logged operations since the last BACKUP LOG statement. 1 – Modified, 0 – Not modified (Used only in bulk logged Recovery model)
In each data file pages are arranged like below
File Header → PFS → GAM → SGAM → BCM → DCM → Data and Index Pages
Along with that we have three different data pages

- Data
- Index
- Text/Image (LOB, ROW_OVERFLOE, XML)

What is Instant File Initialization and how it works?

On Windows systems when SQL Server needs to write something on disk, first it verify that the hard disk space is trustworthy means the space is readable. The verification process is:

- SQL Server writes zeros to the file
- This process is known as zeroing process
- SQL Server uses single thread for zeroing process
- The actual operation that triggers this verification should wait until this verification process completed.

If Instant File Initialization is enabled for SQL Server, it skips the zeroing process for data files and reduces the wait time.

What do you know about Checkpoints?

For performance reasons, the Database Engine performs modifications to database pages in memory, in the buffer cache, and doesn't write these pages to disk after every change. Rather, the Database Engine periodically issues a checkpoint on each database. A checkpoint writes the current in-memory modified pages (known as dirty pages) and transaction log information from memory to disk, and also records the information in the transaction log.

What is Write Ahead Transaction Log?

SQL Server uses a write-ahead log (WAL), which guarantees that no data modifications are written to disk before the associated log record is written to disk.
The internal process that actually goes on:

- Copy of the data pages are pulled and placed in buffer cache
- Applied the operation on the pages that are on buffer cache
- Write the log record details (Pages modified) to Disk
- Write/flush/apply the page to the disk

SQL Commands:

- DDL – Data Definition Language:
  CREATE, DROP, ALTER, TRUNCATE
- DML – Data Manipulation Language:
  INSERT, UPDATE, DELETE, CALL, EXPLAIN CALL, LOCK
- TCL – Transaction Control Language:
  COMMIT, SAVEPOINT, ROLLBACK, SET TRANSACTION, SET CONSTRAINT
- DQL – Data Query Language:
  SELECT
- DCL – Data Control Language:
  GRANT, REVOKE

SQL Server Installations and Upgrades

Do you know how licensing happening in SQL Server?

Prior to SQL Server 2012: Microsoft SQL Server could be purchased as either a Server/CAL (Client Access License) licensing model or a (physical) Processor licensing model.
Server/CAL model: The server was licensed and each user (person or device) needed a CAL.
Processor licensing model: When number of users is large we can go for Processor License which is licensed based on number of physical processors.
From SQL Server 2012: SQL Server Introduced Core Licensing Model which is licensing number of cores for each processor.

- Let’s say I have 6 Processors and each processor is having 8 cores.
- SQL Server 2008 R2 Enterprise: 6 Processor License required
- SQL Server 2012 R2 Enterprise: 6X8 = 48 Core License required

Note: Each Core License price is 1⁄4 of one Processor License. So when we are running ratio of 4 cores per processor end cost doesn’t change and it’s almost same for both 4 Processor License and 16 Core License. This is a very basic difference and it needs a lot of patience to understand SQL Server Licensing models.

What is a service account?

Based on the selected components while doing the installation we will find respective service to each component in the Windows Services. E.g. SQL Server, SQL Server Agent, SQL Analysis Services, SQL Server integration Services etc. There will be a user for each and every service through which each service will run. That use is called Service Account of that service.

Do we need to grant Administrator permissions on the Windows server to SQL Service account to run the services?

No, it is not required. It’s not mandatory to grant Administrator permissions to the service account.

What are the different releases of SQL Server? We usually talk about RTM, RC, SP, CU etc. right, what are the different releases of a SQL Server product and what do you know about them?

- **CTP (Community Technology Preview):** It’s Beta release of SQL Server product.
- **RC (Release Candidate):** A release candidate (RC) is a beta version with potential to be a final product, which is ready to release unless significant bugs emerge.
- **RTM (Released to Manufacturing):** It is the original, released build version of the product. This is the actual product software that we get on DVD or ISO file from MSDN.
- **CU (Cumulative Update):** Cumulative updates contain the bug fixes and enhancements up to that point in time that have been added since the previous Service Pack release and will be contained in the next service pack release.
- **SP (Service Pack):** Larger collection of hotfixes that have been fully regression tested. In some cases delivers product enhancements.
- **GDR (General Distribution Release):** GDR packages contain only security and critical stability issue fixes. GDR fixes should not contain any of the CU updates.
- **LDR/QFE/Hot Fix (Limited Distribution Release/Quick Fix Engineering):** LDR packages contain “other” fixes that have not undergone as extensive testing, and resolve issues that only a fraction of the millions of users might ever encounter. QFE updates include CU fixes.

What is a SQL Server Instance?

An instance of the Database Engine is a copy of the sqlservr.exe executable that runs as an operating system service. Each instance manages its own system databases and one or more user databases. An instance is a complete copy of an SQL Server installation.

Type of Instance and maximum no. of instances which can be installed on a server.

There are two types of Instances.

- Default instance
- Named Instance
- Each computer can run maximum of 50 standalone instances of the Database Engine. One instance can be the default instance.
- The default instance has no name. If a connection request specifies only the name of the computer, the connection is made to the default instance or with IP address and the port number.
- A named instance is one where you specify an instance name when installing the instance. A connection request must specify computer name/IP Address and instance name/Port number in order to connect to the instance.

Can we install multiple instances on the same disk drive?

Yes, we can install multiple instances on the same disk drive because each installation creates its own folder with the below format MSSQL11.INSTANCENAME.

what is the default collation?

Default collation: SQL_Latin1_General_CP1_CI_AS

What is the default port of a SQL Server instance?

SQL Server default instance by default listen on 1433 port.

What are ways of migrating SQL Server from lower version to higher version?

If you want to upgrade a SQL Server instance from SQL Server 2008 R2 to SQL Server 2012/2014, below are the different ways you can do this migration.

- In-Place Upgrade: In this method, existing instance of SQL Server will be upgraded to higher version, thus we end up with one instance of SQL Server with higher version i.e., SQL Server 2012. Here the instance name remains same, so application connection string remains the same, only change that may be required is to have latest connectivity drivers installed.
- Side-By-Side Upgrade: In this method a new instance of SQL Server 2012/2014 is installed on same server or a different server and them all User databases, Logins, Jobs, configuration settings need to be configured or created on the new SQL Server instance.

What’s the practical approach of installing Service Pack?

Steps to install Service pack in Production environments:

- First of all raise a change order and get the necessary approvals for the downtime window. Normally it takes around 45–60 minutes to install Service pack if there are no issues.
- Once the downtime window is started, take a full backup of the user databases and system databases including the Resource database.
- List down all the Start-up parameters, Memory Usage, CPU Usage etc. and save it in a separate file.
- Install the service pack on SQL Servers.
- Verify all the SQL Services are up and running as expected.
- Validate the application functionality.

What is Slipstreaming? And when it is introduced?

It is a term used to describe merging original source media with updates in memory and then installing the updated files. Slipstreaming has been supported by Windows Operating systems for a while but has been added to SQL Server 2008 service pack 1. Slipstream allows you to go to the latest and greatest, initially it was introduced with SQL Server 2008 service pack 1 and a CU for service pack 1.

When we got a request to install SQL Server 2012 with SP1 we can do it in a one go by using slipstreaming procedure that means we can include both SQL Server 2012 and SP1 media files on one folder and can be installed in a single go. Before SQL Server 2008 SP1 we had to install service packs or cumulative updates separately.

Which is the environment you suggest to apply first when a new service pack/cumulative update released?

I always suggest applying first on staging environment which is a Pre-PROD. PROD we can’t directly apply but also remember development environment also lot of dependencies and if something goes wrong it holds/delays the deliverables.

From your experience can you describe the top reasons that cause SQL Server installation failures?

- Missing supporting software. Ex: DOTNET Framework
- Missing Supporting Files
- Corrupted Binaries/Files
- Access dined
- PowerShell not installed
- Partial Uninstallation of previous instance
- Unsupported versions
- Registry corruption
- Using Wrong Services Accounts

What are the various Editions available in SQL Server 2012 version?

Below are the various editions available in Microsoft SQL Server 2012

- SQL Server 2012 Standard Edition
- SQL Server 2012 Enterprise Edition
- SQL Server 2012 Business Edition
- SQL Server 2012 Express Edition
- SQL Server 2012 Web Edition
- SQL Server 2012 Developer Edition

SQL Server Security

If password policy is enforced; can you tell me the new password rules and policies to make sure the password is strong?

Password Complexity: When password complexity policy is enforced, new passwords must meet the following guidelines.

- Password should not contain the login name or computer name
- Password should not be “password,” “admin,” “administrator,” “SA,” “SYSADMIN”
- Password minimum length 8 and maximum 128
- Password contains characters from three of the following four categories:
- Latin uppercase letters (A through Z)
- Latin lowercase letters (a through z)
- Base 10 digits (0 through 9)
- Non-alphanumeric characters such as “!,” “$,” “#,””%”

What are the fixed server level roles?

- SYSADMIN: Can perform any activity
- ServerAdmin: Can change server configuration, restart, shutdown server
- SecurityAdmin: Can manage server level logins, also can manage database level if they have permission on Database
  - Granted: ALTER ANY LOGIN
- ProcessAdmin: Can kill a process on an instance
  - Granted: ALTER ANY CONNECTION, ALTER SERVER STATE
- DiskAdmin: Can manage the disk files
  - Granted: ALTER RESOURCES
- BulkAdmin: Can perform BULK INSERT
  - Granted: ADMINISTER BULK OPERATIONS
- SetupAdmin: Can add and remove linked servers
  - Granted: ALTER ANY LINKED SERVER
- Dbcreator: Can create, alter, drop and restore any database on the instance
  - Granted: CREATE ANY DATABASE
- Public: Default role for newly created login

What are the Database roles available?

- db_accessadmin: Granted: ALTER ANY USER, CREATE SCHEMA, Granted with
  - Grant option - Connect
- db_backupoperator: Granted: BACKUP DATABASE, BACKUP LOG, CHECKPOINT
- db_datareader: Granted - SELECT
- db_datawriter: Granted – INSERT, UPDATE and DELETE
- db_ddladmin: Granted – Any DDL operation
- db_denydatareader: Denied - SELECT
- db_denydatawriter: Denied - INSERT, UPDATE and DELETE
- db_owner: Granted with GRANT option: CONTROL
- db_securityadmin: Granted ALTER ANY APPLICATION ROLE, ALTER ANY ROLE, CREATE SCHEMA, VIEW DEFINITION
- dbm_monitor: Granted: VIEW most recent status in Database Mirroring Monitor

sp_helprolemember: List out the members mapped with the server roles
Note: Fixed database roles are not equivalent to their database-level permission.
For example, the db_owner fixed database role has the CONTROL DATABASE permission. But granting the CONTROL DATABASE permission does not make a user a member of the db_owner fixed database role.

What is Transparent Data Encryption?

TDE is designed to protect data by encrypting the physical files of the database, rather than the data itself. Its main purpose is to prevent unauthorized access to the data by restoring the files to another server. With Transparent Data Encryption in place, this requires the original encryption certificate and master key.

- TDE is introduced with SQL Server 2008 Enterprise edition
- TDE purpose is secure by encrypting database physical files
    - Data File (.MDF/.NDF)
    - Log File (.LDF)
    - Backup File (.bak)
- Data is not encrypted while transferring between databases or when communicating with application. Connection encryption has to be done using Secure Socket Layer (SSL) or Internet Protocol Security (IPSec). There are new features (always encrypted) added in SQL Server 2016 to encrypt data during the network transfer.
- TEMPDB physical file is also encrypted when TDE is initially enabled for the first time in that instance.

How to implement Transparent Data Encryption?

- Create a Database master key for master database
- Create a certificate that’s protected by the master key.
- Create a special key that’s used to protect the database. This key is called the database encryption key (DEK) and we secure it using the certificate.
- Backup Keys and Certificate
- Enable encryption for the user database

How to resolve the orphan user problem?

A database user can become orphaned if the corresponding SQL Server login is dropped. Also, a database user can become orphaned after a database is restored or attached to a different instance of SQL Server. Orphaning can happen if the database user is mapped to a SID that is not present in the new server instance.

To find out the orphan users

```
USE <database_name>;
GO;
sp_change_users_login @Action=’Report’;
GO;
```

To resolve the orphan user problem

```
USE <database_name>;
GO
sp_change_users_login @Action=’update_one’,
@UserNamePattern=’<database_user>’,
@LoginName=’<login_name>’;
GO
```

SQL Server Backup & Restore

How does the database recovery model impact database backups?

The database recovery model is responsible for the retention of the transaction log entries. So based on the setting determines if transaction log backups need to be issued on a regular basis. As we know each and every transaction occurs on database will be logged into transaction log file irrespective of the database recovery model. But the recovery model determines when these logged transactions can be released/removed from log file.

- Simple: Committed transactions are automatically removed from the log when the check point process occurs.
- Bulk Logged: Committed transactions are only removed when the transaction log backup process occurs.
- Full: Committed transactions are only removed when the transaction log backup process occurs.

How do you know if your database backups are restorable?

- Issue the RESTORE VERIFYONLY command to validate the backup. For validating LiteSpeed backups use XP_restore_verifyonly
- Randomly retrieve tapes from off site and work through the restore process with your team to validate the database is restored in a successful manner.

What are some common reasons why database restores fail?

- Sufficient space not available on drive
- User may not have sufficient permissions to perform the restore
- Unable to gain exclusive use of the database.
- LSN's are out of sequence so the backups cannot be restored.
- Syntax error such as with the WITH MOVE command.
- Version problem
- Might be wrong backup location specified
- Service account may not have permissions on backup folder

What are some common post restore processes?

- Sync the logins and users
- Validate the data is accurate by running DBCC commands
- Notify the team\user community
- If the restore is from PROD to Pre-PROD, cleanse the data to remove sensitive data i.e. SSN's, credit card information, customer names, personal information, etc.
- Change database properties i.e. recovery model, read-only, etc.

Can we be able to take the backup for "RESOURCEDB"?

No! We can't perform backup for resource DB. But we can take the physical file backup for "RESOURCEDB" MDF and LDF files.

How to rebuild the system databases?

- To rebuild the master database we have to use setup.exe from command prompt. There is no much difference between 2005 and 2008 except few command line switches.
- Find the setup.exe file (C:...........................\100\Setup BootStrap\Release\setup.exe)
- Run the below command from dos prompt c:\Program Files\Microsoft SQL Server\100\Setup Bootstrap\Release>setup.exe
  /QUIET
  /ACTION=REBUILDDATABASE

```
/INSTANCENAME=<Default/Named>
/SQLSYSADMINACCOUNTS= <Service Account>
[/SAPWD=<Mandatory when using Mixedmode>]
[/SQLCOLLATION=<Give new collation if you want to change default>]
```

- When setup has completed rebuilding the system databases, it will return to the command prompt with no messages (It always first prints out the version). Examine the "Summary" log file ([100\110\120]\setup bootstrap\logs) to verify the completion status.

Can you quickly tell me the process for creating an encrypted backup on SQL Server 2014?

- Create a database master key on master database
- Create a Certificate on master database
- Create a Backup and mention all required parameters for encryption
  - Keyword ENCRYPTION
  - Algorithm ex: AES_256
  - Certificate to be used; the one that we created on master.

Can we perform a tail log backup if .MDF file is corrupted?

Yes! We can perform a tail log as long as the LDF if not corrupted and no bulk logged changes.
A typical tail log backup is having two options, 1. WITH NORECOVERY 2.Continue After Error.
WITH NORECOVERY: To make sure no transactions happens after the tail log backup
CONTINUE AFTER ERROR: Just to make sure log backup happens even though some meta data pages corrupted.

How can full backups be issued without interrupting the LSN's?

Issue the BACKUP command with the COPY_ONLY option

How is a point in time recovery performed?

Point-In-Time recovery means we need to recover data till the specific given time. It depends on which backup types are issued. We have a database and we are performing full, differential and transaction log backups. Due to data corruption we need to recover/restore our database till 2:18 PM. We have the latest transaction log file available at 2:30 PM.

- Restore the most recent full backup with the NORECOVERY clause
- Restore the most recent differential backup with the NORECOVERY clause
- Restore all of the subsequent transaction log backups with the NORECOVERY
- clause except the last transaction log backup
- Restore the last transaction log backup (2:30 PM) with the RECOVERY clause and a STOPAT (2:18 PM) statement.

See I have an environment, Sunday night full backup, everyday night diff backup and every 45 min a transactional backup. Disaster happened at 2:30 PM on Saturday. You suddenly found that the last Sunday backup has been corrupted. What's your recovery plan?

- When you find that the last full backup is corrupted or otherwise un-restorable, making all differentials after that point useless. You then need to go back a further week to the previous full backup (taken 13 days ago), and restore that, plus the differential from 8 days ago, and the subsequent 8 days of transaction logs (assuming none of those ended up corrupted!).
- If you're taking daily full backups, a corrupted full backup only introduce an additional 24 hours of logs to restore.

What are all of the backup\Restore options and their associated value?

Backup Options:

- Full – Online operation to backup all objects and data in a single database
- Differential – Backup all extents with data changes since the last full backup
- Transaction log – Backup all transaction in the database transaction log since the last transaction log backup
- File – Backup of a single file to be included with the backup when a full backup is not possible due to the overall database size
- File group – Backup of a single file group to be included with the backup when a full backup is not possible due to the overall database size
- Cold backup – Offline file system backup of the databases
- Partial Backup – When we want to perform read-write FILEGROUPS and want to exclude read-only FILEGROUPS from backup. It will be useful for huge databases (Data warehousing)
- Third party tools – A variety of third party tools are available to perform the operations above in addition to enterprise management, advanced features, etc.

Restore Options:

- Restore an entire database from a full database backup (a complete restore).
- Restore part of a database (a partial restore).
- Restore specific files or FILEGROUPS to a database (file restore).
- Restore specific pages to a database (page restore).
- Restore a transaction log onto a database (transaction log restore).
- Revert a database to the point in time

Is it possible to restore a Database backup of SQL Server 2012 to SQL Server 2008/2008 R2?

No it’s not possible to restore the upper version database backup to lower version.

SQL Server Failover Clustering

What is Windows Cluster?

Clustering is a technology that automatically allows one physical server to take over the tasks and responsibilities of another physical server that has failed. The goal is to ensure that users running mission-critical applications will have very less downtime when a failure occurred. A cluster is a group of independent computer systems, referred to as nodes, working together as a unified computing resource. A cluster provides a single name for clients to use and a single administrative interface, and it guarantees that data is consistent across nodes.

What is a Cluster Node?

A cluster node is a server within the cluster, and it has Windows Server and the Cluster service installed.

What is Heartbeat in Windows cluster?

Heartbeats are messages that Cluster Service regularly sends between the instances of Cluster Service that are on each node to manage the cluster.

What is a Quorum?

Quorum for a cluster is the number of elements that must be online for that cluster to continue running. Each element in can cast one “vote” to determine whether the cluster continues running.

The full function of a cluster depends on:

- Quorum
- Capacity of each node to support the services and applications that fail over to that node
  Ex: A cluster with 7 Nodes can still have a Quorum (4 Nodes Online) after 3 node failed as majority wins
  But remember it’s not just depends on Quorum also other 4 nodes should have capacity to server clients.

Case 1: On 7 Node Cluster 3 Failed and 4 Online and these 4 are capable to handle the load Cluster continue serving

Case 2: On 7 Node Cluster 3 Failed and 4 Online and these 4 are not capable to handle the load Cluster makes all nodes offline

Case 3: On 7 Node Cluster 3 Failed and 4 Online and these 4 are not capable to handle the load Cluster makes all nodes offline

Case 4: On 7 Node Cluster 4 Failed and 3 Online Cluster makes all nodes offline as majority votes are offline

What are the various Quorum modes available?

Quorum Mode: Strategy to define the elements in cluster which can cast vote to make Quorum

Node Majority: Each node that is available and in communication can vote.

Node and Disk Majority: Each node plus a designated disk in the cluster storage (the “disk witness”) can vote, whenever they are available and in communication.

Node and File Share Majority: Each node plus a designated file share created by the administrator (the “file share witness”) can vote, whenever they are available and in communication.

No Majority: Disk Only: The cluster has quorum if one node is available and in communication with a specific disk in the cluster storage.

What Failover and Failback terms mean in Windows Cluster?

Failover: Failover is the process of moving a group of resources from one node to another in the case of a failure. For example, in a cluster where Microsoft SQL Server is running on node A and node A fails, SQL Server automatically fails over to node B of the cluster.
Failback: Failback is the process of returning a resource or group of resources to the node on which it was running before it failed over. For example, when node A comes back online, SQL Server can fail back from node B to node A.

What is Split Brain situation in Cluster?

Cluster nodes communicate with each other over the network (port 3343). When nodes are unable to communicate with each other, they all assume the resources of the other (unreachable) nodes have to be brought online. Because the same resource will be brought online on multiple nodes at the same time, data corruption may occur. These results in a situation called “Split Brain.”

What is the difference between Active\Passive and Active\Active cluster?

Active – Passive: Active Passive cluster is a failover cluster configured in a way that only one cluster node is active at any given time. The other node, called as Passive node is always online but in an idle condition, waiting for a failure of the Active Node, upon which the Passive Node takes over the SQL Server Services and this becomes the Active Node, the previous Active Node now being a Passive Node.
Active – Active: It is a failover cluster configured in a way that both the cluster nodes are active at any given point of time. That is, one Instance of SQL Server is running on each of the nodes always; when one of the nodes has a failure, both the Instances run on the only one node until the failed node is brought up (after fixing the issue that caused the node failure). The instance is then failed over back to its designated node.

On windows Cluster what are validation tests?

Validation test is a mechanism of verifying that all the components which are participating in the Windows cluster are fine and failover is happening between the nodes.

How to Configure Failover Cluster?

Failover Clustering Configuration Link - YouTube

SQL Server ALWAYSON

What is ALWAYSON in SQL Server?

ALWAYSON Availability Groups feature is a high-availability and disaster-recovery solution that provides an enterprise-level alternative to database mirroring. Introduced in SQL Server 2012, ALWAYSON Availability Groups maximizes the availability of a set of user databases for an enterprise. An availability group supports a failover environment for a discrete set of user databases, known as availability databases that fail over together. An availability group supports a set of read-write primary databases and one to four sets of corresponding secondary databases. Optionally, secondary databases can be made available for read-only access and/or some backup operations.

What are Availability Groups?

A container for a set of databases, availability databases, that fails over together. Let's consider a scenario where a set of 3 databases are interlinked based on application requirement. Now we need to setup HA for these 3 databases. If we choose mirroring we need to have a separate mirroring setup for these 3 databases where as in ALWAYSON Availability Groups easier the job by grouping all these 3 databases.

What are Availability Databases?

It's a database that belongs to an availability group. For each availability database, the availability group maintains a single read-write copy (the primary database) and one to four read-only copies (secondary databases).

How many replicas can I have in an ALWAYSON Availability Group?

SQL Server 2012: 1 Primary and up to 4 Secondary's
SQL Server 2014: 1 Primary and up to 8 Secondary's

How many databases can be configured in an ALWAYSON Availability Group?

Up to 100 is the recommendation, but it's not enforced

What are the minimum requirements of a database to be part of the Always ON Availability Group?

- Availability groups must be created with user databases. Systems databases can't be used.
- Databases must be read-write. Read-only databases aren't supported.
- Databases must be multiuser databases.
- Databases can't use the AUTO_CLOSE feature.
- Databases must use the full recovery model, and there must be a full backup available.
- A given database can only be in a single availability group, and that database can't be configured to use database mirroring.

Is it possible to setup Log Shipping and Replication on a database which is part of Availability Group?

Yes, it can be configured.

Can system database participate in AG?

No.

What version of Windows do I need for ALWAYSON AGs?

We highly recommend Windows Server 2012R2 and above.

What is Availability mode in Always ON?

The availability mode is a property of each availability replica. The availability mode determines whether the primary replica waits to commit transactions on a database until a given secondary replica has written the transaction log records to disk (hardened the log). ALWAYSON supports below modes:
Asynchronous-commit mode: Primary replica commits the transaction on a database without waiting for the conformation from the secondary replica.
Synchronous-commit mode: Primary replica does not commit the transaction on a database until it gets the confirmation (written the transaction log records to disk on secondary) from secondary replica.

What is called Primary replica?

The availability replica that makes the primary databases available for read-write connections from clients is called Primary Replica. It sends transaction log records for each primary database to every secondary replica.

What is called Secondary replica?

An availability replica that maintains a secondary copy of each availability database, and serves as a potential failover targets for the availability group. Optionally, a secondary replica can support read-only access to secondary databases can support creating backups on secondary databases.

What is Availability Group listener?

Availability Group Listener is a server name to which clients can connect in order to access a database in a primary or secondary replica of an ALWAYSON availability group. Availability group listeners direct incoming connections to the primary replica or to a read-only secondary replica.

What are Readable Secondary Replicas?

The ALWAYSON Availability Groups active secondary capabilities include support for read-only access to one or more secondary replicas (readable secondary replicas). A readable secondary replica allows read-only access to all its secondary databases. However, readable secondary databases are not set to read-only. They are dynamic. A given secondary database changes as changes on the corresponding primary database are applied to the secondary database.

What is “Failover” in ALWAYSON?

Within the context of a session between the primary replica and a secondary replica, the primary and secondary roles are potentially interchangeable in a process known as failover. During a failover the target secondary replica transitions to the primary role, becoming the new primary replica. The new primary replica brings its databases online as the primary databases, and client applications can connect to them. When the former primary replica is available, it transitions to the secondary role, becoming a secondary replica. The former primary databases become secondary databases and data synchronization resumes.

What is the use of ALWAYSON Dashboard?

Database administrators use the ALWAYSON Dashboard to obtains an at-a-glance view the health of an ALWAYSON availability group and its availability replicas and databases in SQL Server 2012. Some of the typical uses for the ALWAYSON Dashboard are:

- Choosing a replica for a manual failover.
- Estimating data loss if you force failover.
- Evaluating data-synchronization performance.
- Evaluating the performance impact of a synchronous-commit secondary replica

Suppose primary database became in suspect mode. Will AG have failover to secondary replica?

Issues at the database level, such as a database becoming suspect due to the loss of a data file, deletion of a database, or corruption of a transaction log, do not cause an availability group to failover.

Does AG support automatic page repair for protection against any page corruption happens?

Yes, it automatically takes care of the automatic page repair.

What is session timeout period?

Session-timeout period is a replica property that controls how many seconds (in seconds) that an availability replica waits for a ping response from a connected replica before considering the connection to have failed. By default, a replica waits 10 seconds for a ping response. This replica property applies only the connection between a given secondary replica and the primary replica of the availability group.

Can a database belong to more than one availability group?

No. It's not allowed.

If a DBA expands a data file manually on the primary; will SQL Server automatically grow the same file on SECONDARIES?

Yes! It will be automatically expanded on the Secondary replica.

What are the benefits of Always on feature?

- Utilizing database mirroring for the data transfer over TCP/IP
- providing a combination of Synchronous and Asynchronous mirroring
- providing a logical grouping of similar databases via Availability Groups
- Creating up to four readable secondary replicas
- Allowing backups to be undertaken on a secondary replica
- Performing DBCC statements against a secondary replica
- Employing Built-in Compression & Encryption

Can we take Transaction log backups on the secondary replicas?

Yes, we can take transaction log backups on the secondary replicas without COPY_ONLY option.

Replica is in "resolving" status? What does it mean?

A replica is into "RESOLVING" state when an auto failover is not successful. Additionally the availability databases reports that they are in non-synchronizing state and not accessible.

What are the top reasons that cause an unsuccessful failover?

- Auto failovers in a specific period may crossed the value "Maximum Failures in the Specified Period"
- Insufficient NT Authority\SYSTEM account permissions
- The availability databases are not in a SYNCHRONIZED state

How to Configure Always On AG?

[Always On AG Configuration Link - YouTube]()

SQL Server Replication

Can you define replication?

Replication 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.

What are the various components involved in replication?

- Publisher
- Distributor
- Subscribers
- Article
- Publication
- Subscription

What are the agents involved in replication?

Replication uses a number of standalone programs, called agents, to carry out the tasks associated with tracking changes and distributing data.

- SQL Server Agent
- Snapshot Agent
- Log Reader Agent
- Distribution Agent
- Merge Agent
- Queue Reader Agent

How to monitor latency in replication?

There are three methods.

- Replication monitor
- Replication commands
- Tracer Tokens

Replication Monitor: In replication monitor from the list of all subscriptions just double click on the desired subscription. There we find three tabs.

- Publisher to Distributor History
- Distributor to Subscriber History
- Undistributed commands

What are the different types in replication implementation?

There are mainly types of methodologies available in replication

- Snapshot Replication
- Merge Replication
- Transactional Replication
- Peer to Peer

Can you brief about how Snapshot Replication works?

The publisher simply takes a snapshot of the entire replicated database and shares it with the subscribers. There are two scenarios where snapshot replication is commonly used.

- It is used for databases that rarely change.
- It is used to set a baseline to establish replication between systems while future updates are propagated using transactional or merge replication.

Can you explain how Transactional Replication works?

Transactional replication is implemented by the SQL Server Snapshot Agent, Log Reader Agent, and Distribution Agent.

- Snapshot Agent: Prepares snapshot files containing schema and data of published tables and database objects, stores the files in the snapshot folder, and records synchronization jobs in the distribution database on the Distributor.
- Log Reader Agent: Monitors the transaction log of each database configured for transactional replication and copies the transactions marked for replication from the transaction log into the distribution database, which acts as a reliable store-and-forward queue.
- Distribution Agent: Copies the initial snapshot files from the snapshot folder and the transactions held in the distribution database tables to Subscribers. Incremental changes made at the Publisher flow to Subscribers according to the schedule of the Distribution Agent, which can run continuously for minimal latency, or at scheduled intervals.

Can you explain how Merge Replication works?

- Merge Replication allows the publisher and subscriber to independently make changes to the database.
- Both entities can work without an active network connection.
- When they are reconnected, the merge replication agent checks for changes on both sets of data and modifies each database accordingly.
- If changes conflict with each other, it uses a predefined conflict resolution algorithm to determine the appropriate data.
- Merge replication is commonly used by laptop users and others who cannot be constantly connected to the publisher.
- Merge replication is implemented by the SQL Server Snapshot Agent and Merge Agent.
- Snapshot Agent: If the publication is unfiltered or uses static filters, the Snapshot Agent creates a single snapshot. If the publication uses parameterized filters, the Snapshot Agent creates a snapshot for each partition of data.
- Merge Agent: The Merge Agent applies the initial snapshots to the Subscribers. It also merges incremental data changes that occurred at the Publisher or Subscribers after the initial snapshot was created, and detects and resolves any conflicts according to rules you configure.

Can you quickly explain how to configure replication in a generic way?

- Identify and configure Distributer
- Identify and configure Publisher
- Publish Data and Database Objects:
  - Creating a publication and defining the data and database objects in the publication, setting options, and applying filters, if necessary.
- Subscribing to Publications:
  - Creating PUSH and PULL subscriptions; Specify synchronization schedules and set other options.
- Initializing Subscription: Describes how to initialize the Subscriber
- Synchronizing Data: Specify options for synchronization, which occurs when the Distribution Agent or Merge Agent runs and updates are propagated between the Publisher and Subscribers

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.

Have you ever heard the word “sync with backup” in transactional replication?

Yes! This can be set on the distribution database and the publication database. When this option enabled transactions in the log will not be truncated until they have been backed up. The sync with backup option ensures consistency between the publication database and the distribution database, but the option does not guarantee against data loss. For example, if the transaction log is lost, transactions that have been committed since the last transaction log backup will not be available in the publication database or the distribution database. This is the same behavior as a non-replicated database.

How to Add an Article to an existing publication?

For adding and dropping articles from an existing publication, you must create a new snapshot for the publication. This is the recommended practice. To avoid generating a snapshot for all articles when adding a new article, publication property immediate_sync must be set to 0 and then call sp_addarticle, followed by sp_addsubscription. If it is pull subscription, you must call sp_refreshsubscriptions. Then generate a snapshot and this process will yield to generate snapshot only for the newly added articles.

As invalidating the snapshot is not a recommended option, but still, this is more important when we deal with large replication database and we don’t want to reinitialize snapshot agent because of the database size and other considerations in mind. In order to do this; we need to set the publisher properties allow_anonymous and Immediate_sync to False. Let us do this using the following T-SQL.

- Turn off 'allow_anonymous',
- Turn off 'immediate_sync',
- Add Article,
- Now, start Snapshot Agent using Replication monitor
- You should notice that bulk-insert statements are created only for the specific article instead of all articles,
- Next, start log reader agent
- Re-enable the disabled properties, first, immediate_sync and then Allow_anonymous options
- Now you can verify the article on all your subscribers

How to Configure Replication?

Replication Configuration Link - YouTube

SQL Server Concurrency & Locking

What is Locking, Blocking and Deadlocks?

Locking: Locking is a mechanism used by SQL Server to control concurrent access to data and ensure data integrity and consistency. Locks are acquired on data when a transaction starts and released when it ends.

Blocking: Blocking problems when one transaction holds a lock on a piece of data that another transaction wants to access. The second transaction has to wait for the first to release the lock before it can continue. This can lead to performance issues and delay the completion of transactions.

Deadlocks: Deadlocks occur when two or more transactions are blocked and each is waiting for the other to release a lock, resulting in a circular wait. This can lead to an indefinite wait situation and requires manual intervention to resolve. Deadlocks can be identified and resolved using SQL Server's Dynamic Management View (DMV) or by using SSMS, performance monitor or custom code as discussed below.

What are the properties of a transaction?

There are 4 properties called ACID.

Atomicity: All changes to data are performed as if they are a single operation. That is, all the changes are performed, or none of them are.

Example: In an application that transfers funds from one account to another, the atomicity property ensures that, if a debit is made successfully from one account, the corresponding credit is made to the other account.

Consistency: Data is in a consistent state when a transaction starts and when it ends.

Example: In an application that transfers funds from one account to another, the consistency property ensures that the total value of funds in both the accounts is the same at the start and end of each transaction.

Isolation: The intermediate state of a transaction is invisible to other transactions. As a result, transactions that run concurrently appear to be serialized.

Example: in an application that transfers funds from one account to another, the isolation property ensures that another transaction sees the transferred funds in one account or the other, but not in both, nor in neither.

Durability: After a transaction successfully completes, changes to data persist and are not undone, even in the event of a system failure.

Example: in an application that transfers funds from one account to another, the durability property ensures that the changes made to each account will not be reversed.

What are the other statements that can’t be included inside an explicit transaction?

You can use all Transact-SQL statements in an explicit transaction, except for the following statements:

- ALTER DATABASE
- CREATE FULLTEXT INDEX
- ALTER FULLTEXT CATALOG
- DROP DATABASE
- ALTER FULLTEXT INDEX
- DROP FULLTEXT CATALOG
- BACKUP
- DROP FULLTEXT INDEX
- CREATE DATABASE
- RECONFIGURE
- CREATE FULLTEXT CATALOG
- RESTORE

Can we use UPDATE STSTISTICS inside a transaction?

Yes and No! Yes we can use UPDATE STSTISTICS inside a transaction but it doesn't follow ACID properties means a rollback can't impact statistics update.

Demonstrate Isolation levels?

- **Read Uncommitted:** This is the lowest isolation level. It only isolates transactions and activities to ensure that physically corrupt data is never read. It allows dirty reads, non-repeatable reads, and phantom reads.
- **Read Committed:** This isolation level does not permit dirty reads, but does allow non-repeatable reads and phantom reads. This is the default isolation level for SQL Server, and is used for each connection to SQL Server unless one of the other isolation levels has manually been set for a connection.
- **Repeatable Read:** This isolation level does not permit dirty reads or non-repeatable reads, but does allow phantom reads.
- **Serializable Read:** This is the highest isolation level and ensures that all transactions and statements are completely isolated from each other. It does not allow dirty reads, non-repeatable reads, or phantom reads.
- **READ_COMMITTED_SNAPSHOT (statement level):** READ_COMMITTED_SNAPSHOT is a variation of the default READ_COMMITTED isolation level. It uses row versioning, instead of locking, to provide read consistency at the SQL Server statement level.
- **ALLOW_SNAPSHOT_ISOLATION (transaction level):** ALLOW_SNAPSHOT_ISOLATION is similar to READ_COMMITTED_SNAPSHOT, but it is based at the transaction level, not the statement level.

What are the Pessimistic and Optimistic ISOLATION levels?

- Serializable: Pessimistic
- Repeatable Read: Pessimistic
- Read Uncommitted: Pessimistic
- Read Committed: It supports both Optimistic and Pessimistic (Default)
- Snapshot: Optimistic

Any idea about row versioning?

The concept of row versioning is not new to SQL Server, as SQL Server has been using it for years with triggers. For example, when a DELETE trigger is executed for a row, a copy of that row is stored in the "deleted table" just in case the trigger is rolled back and the deleted row needs to be "undeleted." In a sense, the row is versioned, and if need be, can be reused.

What is Update Statistics?

You can update query optimization statistics on a table or indexed view in SQL Server by using SQL Server Management Studio or Transact-SQL. 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 or the stored procedure sp_updatestats to update statistics more frequently than the default updates.

SQL Server Performance Tuning

What are the different indexes available in SQL Server?

- **Clustered Index:** A clustered index defines the order in which data is physically stored in a table. Table data can be sorted in only way, therefore, there can be only one clustered index per table. In SQL Server, the primary key constraint automatically creates a clustered index on that particular column.
- **Non Clustered Index:** The non-Clustered Index is similar to the index of a book. The index of a book consists of a chapter name and page number, if you want to read any topic or chapter then you can directly go to that page by using the index of that book. No need to go through each and every page of a book.
- **ColumnStore Index:** Columnstore indexes are the standard for storing and querying large data warehousing fact tables. This index uses column-based data storage and query processing to achieve gains up to 10 times the query performance in your data warehouse over traditional row-oriented storage.

What is fill factor? How to choose the fill factor while creating an index?

The Fill Factor specifies the % of fullness of the leaf level pages of an index. When an index is created or rebuilt the leaf level pages are written to the level where the pages are filled up to the fill factor value and the remainder of the page is left blank for future usage. This is the case when a value other than 0 or 100 is specified. For example, if a fill factor value of 70 is chosen, the index pages are all written with the pages being 70 % full, leaving 30 % of space for future usage.

What is Online Indexing?

Online indexing means performing index maintenance operations (reorganizing or rebuilding) in background while the data still available for access.

What are the primary differences between index reorganization and an index rebuild?

- Reorganization is an “online” operation by default; a rebuild is an “offline” operation by default
- Reorganization only affects the leaf level of an index
- Reorganization swaps data pages in-place by using only the pages already allocated to the index; a rebuild uses new pages/allocations
- Reorganization is always a fully-logged operation; a rebuild can be a minimally-logged operation
- Reorganization can be stopped mid-process and all completed work is retained; a rebuild is transactional and must be completed in entirety to keep changes

What is the Heap Table?

- A table without any index created on it
- Data is not stored in any particular order
- Specific data cannot be retrieved quickly, unless there are also non-clustered indexes

What is fragmentation?

Fragmentation can be defined as any condition that cause more than optimal amount of disk I/O to be performed in accessing a table or cause the longer disk I/O. Optimal performance comes for the SELECT queries when the data pages of tables are contiguous or fully packed. Fragmentation can happen in two levels, file system level fragmentation which is called as Logical/Physical Disk Fragmentation and Index level fragmentation.

Can we create a Non-Clustered index on a Clustered indexed column? What is the benefit?

Yes! It is possible. You can create it if you clearly understand how B-Tree forms for clustered and Non-Clustered indexes. However from my knowledge here are the benefits by having Clustered and Non-Clustered indexes on same column:

- Scanning Non-Clustered index leaf nodes is easier than clustered index leaf nodes as Clustered Index leaf node is having actual rows and Non-clustered index leaf nodes are having only Index Columns. So it can help in improving the queries like SELECT COUNT(<Indexed Column>) as it simply scans the non-clustered index instead of clustered index.
- The other case where you have queries that require only indexed columns, here again scanning the non-clustered index is faster than clustered index.

Note: Having Non-Clustered on Clustered index column might help you in improving the performance for some queries but it will surely increase the Index maintenance which in-turn to more I/O and disk space usage

What methods are available for removing fragmentation of any kind on an index in SQL Server?

Before SQL Server 2005:

```
        DBCC INDEXDEFRAG
        DBCC DBREINDEX
        CREATE INDEX...DROP EXISTING (cluster)
        DROP INDEX; CREATE INDEX
```

SQL Server 2005, 2008, 2012, 2014, 2016:

```
        ALTER INDEX...REORGANIZE
        ALTER INDEX...REBUILD
        CREATE INDEX...DROP EXISTING (cluster)
        DROP INDEX; CREATE INDEX
```

What is PLE?

Page Life Expectancy: Number of seconds a page is staying on buffer cache. Usually we do calculate based on the Memory allocated to SQL Server Instance.

What is the option "Lock Pages in Memory"?

Lock Pages in Memory is a setting that can be set on 64-bit operating systems that essentially tell Windows not to swap out SQL Server memory to disk. By default, this setting is turned off on 64-bit systems, but depends on various conditions this option needs to be turned on.

What is "refreshing a database" means?

It is a process of migrating or moving the live data from the production server to a testing or development server.

How to Rebuild an index?

We can use the Command called "DBCC DBREINDEX".

The following example rebuilds the Employee_EmployeeID clustered index with a fill factor of 80 on the Employee table in the AdventureWorks2022 database.

```
USE AdventureWorks2022;
GO
DBCC DBREINDEX ('HumanResources.Employee', PK_Employee_BusinessEntityID, 80);
GO
```

What is the process of tuning the Performance?

- **Identification:** Use native tools like Execution Plans, DMV, Extended Events, Query Tuning Advisor, Activity Monitor, system stored procedures, DBCC, custom stored procedures or third party tools etc.
- **Analysis:** Analyze the data to determine the core problems
- **Providing Solution:**
  - Creating new index on appropriate columns
  - Altering the complex quires to make them use the existing indexes.
  - By Updating Statistics for Tables and Views.
  - By rebuilding or reorganizing indexes.
  - By Resolving blocking problems.
  - By preventing Deadlocks
  - Resolving parameter sniffing issues
  - Using the proper SET options
  - Changing the Database/Server configurations
  - Recoding to troubleshoot the performance bottleneck
  - Enabling trace flags
- **Testing:** Test the various options to ensure they perform better and do not cause worse performance in other portions of the application
- **Knowledge sharing:** Share your experience with the team to ensure they understand the problem and solution, so the issue does not occur again, also document the problem and the possible solutions on central repository.

What is the table partitioning in SQL Server?

- Table partitioning allows tables or indexes to be stored in multiple physical locations.
- A partitioned index is made of multiple small indexes that hold the range values of a same column.
- Partitioning is a transparent feature, means it doesn’t require the code change in the calling application.

What is MAXDOP?

The Microsoft SQL Server max degree of parallelism (MAXDOP) configuration option controls the number of processors that are used for the execution of a query in a parallel plan. This option determines the computing and threads resources that are used for the query plan operators that perform the work in parallel.

SQL Server DBA General

What are DBCC commands in SQL Server?

Microsoft SQL Server Database Console Commands (DBCC) are used for checking database integrity; performing maintenance operations on databases, tables, indexes, and filegroups; and collecting and displaying information during troubleshooting issues.

What is PHYSICAL_ONLY option in DBCC CHECKDB?

This command limits the checking to the integrity of the physical structure of the page and record headers, but it can also detect torn pages, checksum failures, and common hardware failures. Using the PHYSICAL_ONLY option may cause a much shorter run- time for DBCC CHECKDB on large databases and is recommended for frequent use on production systems. Specifying PHYSICAL_ONLY causes DBCC CHECKDB to skip all checks of FILESTREAM data.
DBCC CHECKDB (‘TestDB’) WITH NO_INFOMSGS, PHYSICAL_ONLY

What are the most common issues a SQL DBA should deal with as a part of DBA daily job?

- Backup Failure
- Restore Failure
- Log Full Issues
- Blocking Alerts
- Deadlocks Alerts
- TEMPDB full issues
- Disk Full Issues
- SQL Connectivity Issues
- Installation and Upgrade Failures
- SQL Agent Job failures
- Performance Issues
- Resource (Memory/IO/CPU etc.) Utilization Alerts
- High-Availability and Disaster Recovery related issues

How to recover a database that is in suspect stage?

- ALTER DATABASE test_db SET EMERGENCY
- After you execute this statement SQL Server will shut-down the database and restart it without recovering it. This will allow you to view/query database objects, but the database will be in read-only mode. Any attempt to modify data will result in an error similar to the following:
  Msg 3908, Level 16, State 1, Line 1 Could not run BEGIN TRANSACTION in database ‘test’ .....Etc.
- Now set the database in single user mode
  ALTER DATABASE test SET SINGLE_USER
  GO
- Execute DBCC CHECKDB
  DBCC CHECKDB (‘test’, REPAIR_ALLOW_DATA_LOSS) GO

If DBCC CHECKDB statement above succeeds the database is brought back online (but you’ll have to place it in multi-user mode before your users can connect to it). Before you turn the database over to your users you should run other statements to ensure its transactional consistency. If DBCC CHECKDB fails then there is no way to repair the database - you must restore it from a backup.

You got an alert by saying that a SQL Agent job has been failed on one of the server. You need to find the root cause. What is approach?

There are certain things we need to quickly check when a job failed:

- Job History
- SQL Server Error Log
- Log file – If you configure any log file at Job Step advanced properties
- Window Event Log
- Job Execution Time Delta – Time Difference between the current and last execution

The above checklist will give you maximum information that causes the job failure. To do further RCA, note down the job execution time and capture below details in that particular time period:

- CPU usage, Memory usage, I/O usage, Blocking and Deadlocks if any, Any dump file created, Log full issues if any, Any other suspecting errors.

In your environment how do you handle the old history and backups?

For the current client we designed maintenance plans to take care of History and Maintenance cleanup. We use tasks History Cleanup and Maintenance Cleanup tasks from maintenance plans.
Database Full backups (Weekly): Delete full backups older than 2 weeks
Database Diff and Log (Daily/Hourly): Delete all backups older than 2 weeks
Note: For backups even though we delete from the disk those files will be maintained on archive for 6 months

What's the difference between DELETE TABLE and TRUNCATE TABLE commands?

Truncate:

- De-allocates the data pages in a table and only this deallocation is stored in transaction log.
- It acquires only table and page locks for the whole table. Since no row locks are used less memory is required.
- Resets identity column if available
- Removes all pages. NO empty pages are left behind in a table
- Fast(er)
- Doesn't fire delete triggers

Delete:

- Removes one row at the time and every deleted row is stored in the transaction log It acquires table and/or page and row locks for the whole table
- Leaves identity column alone
- Can leave empty pages in a table since empty page removal requires a table lock which doesn't necessarily happen
- Slow(er)
- Fires delete triggers.

What is the quickest way to get SQL Server memory usage?

DBCC MEMORYSTATUS

On a production server TEMPDB is getting full and how do you fix it?

We usually don't try to fix issues on production TEMPDB without a maintenance window. When we see TEMPDB is getting full we identify the active queries and from those active queries we'll identify queries/SPID causing the TEMPDB full and then kill that SPID. When we need to give a permanent fix on TEMPDB we take a maintenance window and then apply the required fix that may include increasing the TEMPDB data file size, creating a new data file with the same size, creating filter indexes based on problematic queries to reduce load on Temdb etc.

One of the disk drive is 95% full within a very less time. Now you started getting disk full alerts and you need to free up the space. What are the different options you try to clear the space? And what are the possible reasons that cause the sudden disk full issues?

Possible Reasons:

- Huge data load happens and it increased the data file size as auto growth is enabled
- An open transaction causes to increase a database log file increased when it's auto grow option is on
- A big transaction log/differential backup generated
- TEMPDB might filled the disk due to a huge query sorting and maintenance
- Huge number of SQL Mini DUMP files created on log folder
- It's may not always SQL Server causes the DISK FULL issue, check the huge files on disk manually and find if any other application or OS causing the issue.
- Disk size was gradually increased as per the process but the alert was disabled and just it got enabled and it started sending alert messages

Resolutions:

- We quickly identify if there are any files that can be moved to other drive
- Remove if you find any old/unnecessary backups, SQL Server logs or crashed logs
- If it is log file full issues handle it properly and shrink the log file
- If you find any low risk or small databases located on that drive then try to move those database files to other drive using detach and attach method
- If you identify any open transaction which is causing the disk full then collect all possible details of that transaction and kill it.
- Talk to the server owner and ask for more space on that drive or request for a new drive, take a maintenance window and increase the drive space or attach a new drive.

Any idea about boot page?

In every database there is a page available which stores about the most critical information about that database.
This page is called boot page. Boot Page is page 9 in first file on primary file group.
We can examine the BOOTPAGE using DBCC PAGE or DBCC DBINF

We are not able to connect to SQL Server. Can you list out what are all the possible issues and resolutions?

- Service Down/Issue
- Network Access/Firewall Issue
- Authentication and Login issue
- SQL Server configuration Issue
- Application Driver or Connection String Issue

Possible Problems:

- Using a wrong instance name/IP or port
- Using a wrong user name or password
- User access might be revoked
- Trying to access from outside organization VPN
- SQL Server is down
- SQL Server is not responding due to high CPU/Memory/Disk I/O
- Might be a disk full issue
- Master database might be corrupted
- User default database may not be online
- SQL Server port might be blocked
- We are using named instance name and SQL Browser service is down
- Using the wrong network protocol
- Remote connections may not be enabled

- Network issue with the host windows server
- Using a wrong client driver (32 bit – 64 bit issues or Old driver using for new version)
- Version Specific issues, for example an application cannot connect to a contained database when connection pooling is enabled. This issue got fixed in SQL Server 2014 CU1

Resolutions:

The error message itself can tell you how to proceed ahead with the resolution:

- If possible first thing should be done is, check SQL Server and Windows error log as it can tell us the exact problem and based on that we can determine the possible best resolution.
- Please cross check connection string information before complaining
- Cross check hosted windows server and SQL Server are up and running
- Make sure the SQL login default database is online and accessible
- Make sure the user access is not revoked
- Make sure all system databases are up and running
- Cross check all resource usage that includes Memory, CPU, Disk I/O, Disk Space etc.
- Try to use IP address and port number instead of instance name, also try with FQDN
- Try to connect from different possible places/systems to make sure the source system has no issues
- Check windows server is reachable from remote location using PING
- Check SQL Server listening on the given port using TELNET <IP> <Port>. Try both from local and remote
- If the port is blocked add this port to exception list in windows firewall INBOUND rules
- Make sure SQL Server is configured to allow remote connections
- If you are also not able to connect then try to connect using DAC and fix the issue by running DBCC commands
- Try if you can connect using SQLCMD
- Cross check if there is any recent changes happened in Active Directory security policy
- Make sure you are using the correct driver to connect to application
- Cross check if there is any blocking on system process

My database is in full recovery mode and log file is reaching 95%. Recently huge log backups generated due to a bulk data import followed by an index maintenance. Now the problem is we don’t have space available to either to increase the log size or to take the next log backup. What we can do to reduce the log size to normal?

I have faced some situations like this. When we don’t have any option available to handle a log file growth then there is only way to change the database recovery model to SIMPLE.

- SET Recovery model to simple – It break the log chain
- Perform SHRINK operation and CHECKPOINT
- Once log file size reduced then make database recovery mode to FULL
- Immediately perform a full backup
- Then we can delete the old huge transaction log files to get some more free space
- As per the schedule log backups should starts

Let’s say our database BOOTPAGE is corrupted. How do you recover the DB?

When BOOTPAGE corrupted:

- We cannot run DBCC
- We cannot run repair
- We cannot put database in EMERGENCY mode
- There is no way that we can get recover data except restoring available backups

How to size the LDF file?

There is no specific formula to size a log file. We can do that based on log file usage:
Simple Recovery: We usually keep LDF file size as 10 to 15 percent of its MDF file size. But still monitor the file usage and make sure the file is properly sized to handle the longest transaction.
Full Recovery: We usually size LDF file size as 80 to 90 percent of its MDF file size. It essentially depends on three factors. 1. MDF file size; 2. Log backup interval and 3.Largest Transaction.

How do you troubleshoot transaction log full issue?

Columns log_reuse_wait and log_reuse_wait_desc of the sys.databases CATALOG view describes what is the actual problem that causes log full/delay truncation.

- Backing up the log.
- Freeing disk space so that the log can automatically grow.
- Moving the log file to a disk drive with sufficient space.
- Increasing the size of a log file.
- Adding a log file on a different disk.
- Completing or killing a long-running transaction.

How do you define High Availability, Disaster Recovery?

High Availability (HA): This is a technology solution to make sure server/databases are available 24X7 and 365 days. The principal goal of a high availability solution is to minimize or mitigate the impact of downtime. Ex: ALWAYSON AG, Failover Clustering, Database Mirroring, Log Shipping etc.
Disaster Recovery (DR): A disaster can be numerous that includes power failure, hardware failure, virus attack, natural disaster, human error, etc. A SQL Server disaster recovery plan (DRP) is a process to have SQL Server up and running, and to overcome data loss after a disaster. It refers to restoring your systems and data to a previous acceptable state in the event of partial or complete failure of computers due to natural or technical causes. Ex: Backup & Restore, Replication, Failover Clustering, ALWAYSON AG, Database Mirroring, Log Shipping etc.

What is Disaster Recovery Test?

A disaster recovery test (DR test) is the examination of each step in a disaster recovery plan as outlined in an organization's business continuity/disaster recovery (BCDR) planning process. Evaluating the DR plan helps ensure that an organization can recover data, restore business critical applications and continue operations after an interruption of services.
Types of disaster recovery tests
There are three basic types of disaster recovery testing. These include a plan review, tabletop exercise and simulation test.
Plan review: Here, the DR plan owner and other members of the team behind its development and implementation closely review the plan, examining it in detail to find any inconsistencies and missing elements.
Tabletop exercise: These are exercises where stakeholders gather to walk step by step through all the components of a disaster recovery plan. This helps determine if everyone knows what they are supposed to do in case of an emergency and uncovers any inconsistencies, missing information or errors.
Simulation: Simulating disaster scenarios is a good way to see if the procedures and resources -- including backup systems and recovery sites -- allocated for disaster recovery and business continuity work in a situation as close to the real-world as possible. A simulation runs a variety of disaster scenarios to see if the teams involved in the DR process can restart technologies and business operations in a timely manner. This process can determine if there are enough people on staff to get the DR job done properly.