Category/Task Dictionary for DBA Service Offerings

(101 DBA tasks that are relevant to every organization)

CATEGORIES:

INFRASTRUCTURE: INSTALLS, MAINTENANCE, AND ADMINISTRATION

Databases, New, Migrate:

- Create new databases, new tables in SQL Server. Migrate data to new databases, tables.
- Configure recovery models (FULL, SIMPLE, BULK LOAD) in accordance with the organization's recovery plan and backup plan.
- Model databases for OLTP or OLAP transactions to determine how data is stored and accessed.
- Create an Entity Relationship Diagram (ERD) to analyze the three principal components-entities, attributes, and relationships.
- Configure table constraints, default values, check values, primary keys, and foreign keys.
- Create PROD, DEV, QA, TEST and UAT environments.
- Migrate databases and transition schemas from their current state to a newly desired state.
- Migrate databases to fulfill development and reporting environment needs.

Server - Cumulative Updates and Patching:

- Apply Updates and Patching in SQL Server
- SQL Server cumulative updates (KB articles), service packs and hot fixes include security enhancements and should be installed as soon as they become available.
- Assist with Host OS service packs.

Latest updates and history for SQL Server are available here:
https://learn.microsoft.com/en-us/troubleshoot/sql/releases/download-and-install-latest-updates

Server Version Upgrades:

- Apply SQL Version upgrades. Upgrade to SQL Server 2022.
- Run the SQL Server Upgrade Advisor before upgrading and make changes.
- Consider your data server environment – backups from new edition servers cannot be restored to older version database servers.
- Test any attached applications with the new data server version on a test server.
- Always backup before making and version changes.

- Don’t upgrade SQL Server clusters in place, rebuild them on new hardware.
- Statistics are not automatically updated during the upgrade process, update all of the statistics in all your databases.

Database Maintenance:

- Maintain databases, data files, growth, availability.
- Use sp_cycle_errorlog or DBCC ERRORLOG to prevent error log from growing excessively large – restarting the SQL Server service also refreshes the SQL Server log – this process should be set up in a weekly job.
- Shrink a database to reclaim file space after large amounts data is deleted.
- Monitor and proactively alter database size for specific circumstances and configure AUTOGROWTH to deal with unexpected growth.

Database Consistency, DBCC Check Table:

- In combination with a well-designed backup plan, the regular scheduling of database consistency checks (DBCC) is essential to monitoring database corruption.
- DBCC Operations include DBCC_CHECKTABLE, DBCC_CHECKCATALOG, DBCC_CHECLALLOC, service broker data check, and indexed view content check. These consistency checks should be automated in your environment and scheduled to run after every FULL backup.
- Since DBCC commands are I/O intensive and can adversely affect performance, schedule for off-peak usage.

Offline, Suspect or Recovery Pending databases:

- Attempt to rebuild a database with a corrupted or deleted SQL Server Transaction Log File.
- Assist with space issues and recovery and returning databases to an online state.

Manage SQL Server System Databases (master, model, msdb, tempdb):

- Backing up the system databases
- Moving system databases
- Single user mode

- Restore the master database
- Configure the model database
- Configure tempdb location, size, and auto growth

Memory management, Parallel processing:

- If SQL Server is one of several server applications running on a single computer, the system administrators may need to control the amount of memory allocated to SQL Server, applications, and the OS.
- Setting max degree of parallelism (MAXDOP) to 0 (default) allows SQL Server to use all the available processors up to 64 processors. However, this isn't the recommended value for most cases.
- A single query request can spawn multiple tasks up to the MAXDOP limit, and each task will use one worker and one scheduler.
- SQL Server 2022 (16.x) introduced Degree of Parallelism (DOP) Feedback, a new feature to improve query performance by identifying parallelism inefficiencies for repeating queries, based on elapsed time and waits.

Data Type Selection & Changes:

- SQL Server data types impact storage size and costs as well as affect query performance.
- Optimize data types to limit unnecessary memory usage.
- Convert deprecated data types before they retire.

Clustered Instances, Log Shipping:

- Make your database Highly Available (HA, HADR) and keep your applications online more of the time with Windows Failover Clustering (WFC). Standard architecture includes one Domain Controller, two Windows Server host nodes and one shared storage, usually SAN. Azure configurations provide additional storage options.
- SQL Server Log Shipping utilizes native SQL Server features and provides an alternative Disaster Recovery solution. It includes the migration of transaction log files between separate (offsite) locations.
- Log shipping remains a viable business continuity solution for Azure VM (IaaS) deployment models while Azure automated backups and failover groups are available

to support business continuity for SQL Managed Instances (Paas) and Azure SQL Database (Paas) models.

Transaction Log Maintenance:

- SQL Server Transaction Logs work in conjunction with full recovery models to provide point in time recovery.
- Without transaction log backups the transaction log file will grow continuously until it runs out of free space.
- Regular transaction log and backup chain maintenance includes tail-log backups and backing up, truncating, and shrinking transaction log files.
- One popular data recovery process used to recover data that had been inadvertently deleted is to restore the transaction log chain to a specific point in time before the DELETE statement was executed.

New SQL Server Installation and Configuration:

- Install SQL Server instances consistently following an agreed upon organization standard.
- Only deploy the features you need; others can be added later.
- Don't install Full-text indexing, Reporting Services, Analysis Services if you don't need them.
- For best performance in Windows, turn off any operating system services that aren't needed.
- Pay attention to Collation; you have to delete and recreate the server to change it.
- Document all server features, configuration, and security configuration to enable you to recreate the server in an emergency.
- For optimal SQL Server performance dedicate your physical servers to a single instance of SQL Server along with no other applications.
- Do not install SQL Server on a Domain Controller.
- Be sure SQL Server is installed on an NTFS partition.
- Do not use NTFS data file encryption and compression on SQL database and log files.
- Only install the required network libraries and network protocols on your SQL Server instances.
- Ensure SQL Servers are behind a firewall and are not exposed directly to the internet.
- Use the Surface Area Configuration Facet in SSMS to manually implement changes on each individual server database engine.

- For a new install, SQL Server configuration settings should remain at their ‘default’ values – they should be changed only after all the pros and cons have been evaluated.

Configuration Manager:

- Configuration Manager is a tool used to manage the Windows services associated with SQL Server, to configure the network protocols used by SQL Server, to manage the network connectivity configuration, to assign service accounts selected to control services and to configure the automation default setting available for each service.

Capacity Planning:

- When planning data environments cost and durability are key considerations.
- Capacity planning is a critical exercise we can assist with to help understand the life of your existing hardware and software platforms as you need to scale to meet business needs within a planned expenditure scope.
- Typical considerations are CPU and memory utilization, transactions per second, disk contention, numbers of users, database size, and overall availability.

SQL Server Hardware Optimization (CPU, Mem, Drives, Space):

- There is no general practice for how much a CPU and memory a SQL Server instance needs to be efficient. Some factors to consider include the SQL Server version and edition of the instance, host OS version, other applications running on the host server, the type of transactions OLTP vs OLAP, the number of concurrent users accessing the data, the size of the databases, the quality of the I/O drives attached, etc.
- Upon initial SQL Server install, the recommended minimum hardware requirements are: 6GB of hard drive space, a minimum of 2GB RAM, 2.0 GHz x64 processor, Windows 10 / Windows Server 2016 or greater, and internet access.
- From the initial setup resources should be tuned as database size and user traffic increase. To evaluate the current resource usage and tune SQL Server to the current database size and usage pattern SQL Server components should be monitored with applications such as Activity Monitor, Windows Performance Monitor, Database Engine Tuning Advisor, Extended Events, Dynamic Management Views, etc.
- Best practices for hardware tuning include establishing performance baselines, isolating performance problems, and identifying bottlenecks.

Host Server Drive Layout Best Practices:

- One of the most important best practices advised for the installation of a SQL Server instance is the layout pattern of the host drives, particularly for production instances.
- Proper drive configuration ensures the proper redundancy to provide fault tolerance to meet business needs. It facilitates system availability, durability and provides the best environment for performance.
- Improper drive configuration can lead to data loss when system interruptions occur.
- If your host drives don't follow these specifications, it's time to redesign your server's drive layouts to avoid potential issues. Improper drive layout can starve operating system resources, create resource contention, create I/O bottlenecks, and resultantly impede database performance.
- These drive configuration practices apply to on-prem hardware infrastructure, virtual installations, and Azure SQL VMs. Azure managed instances and Azure SQL databases do not apply as the storage resources are fully managed by Microsoft.
- For modern storage solutions, particularly VMs, separate drives do not necessarily mean different physical hardware.
- In the case of modern SANs and cloud configurations the number of controllers assigned will play an important role in performance.
- Throughput and I/O capabilities are important considerations when designing for performance.
- Configure single drives and SANs (data and log) with Raid 1 arrays for OLTP processing and Raid 10 arrays for OLAP processing. In Raid 1 data is written to multiple disks. Redundancy exists as long as one drive in the mirrored set is functioning. In Raid 10 Mirrored disk sets are striped providing high redundancy as well as high performance.
- Put the SQL Server binarys and program files on a separate drive, D: works.
- Put the SQL Server system databases (master, model, msdb) (I:) on their own drive.
- Put data (E:) and log files (F:) on their own drives, apart from your operating system. You don't want your data operations competing with operating system resources and if any OS related issue occurs you are in danger of your databases entering Recovery Mode and potentially losing data.
- Put the tempdb (G:)files on its own drive as tempbd access patterns, primarily write intensive, differ from the typical data reads and writes common on the data drives.
- Ensure backup files (H:)are stored on their own drive. If you lose a data file because of a failed drive, you don't want to lose the backup files necessary to restore the data as well.
- You might also consider copying backup files to an offsite/off server resource. Offsite backup storage adds additional protection against ransomware attacks.

SQL Server Advanced Options - Enabling XP_CMDSHELL and SQLCMD, DAC:

- In SQL Server Advanced Options are available to manage services, assist with configuration settings, network protocols, network connectivity, registry values and to view failover cluster instance information. *See XP_CMDSHELL and SQLCMD.*
- All of these options are only available through scripting the advanced server configurations.
- In Azure the Dedicated Administrator Connection (DAC) is available to connect to the system during resource bottlenecks for diagnostic and troubleshooting purposes.

Azure SQL Server Databases and Configuration:

- Azure SQL is a family of managed, secure, and intelligent products that use the SQL Server database engine in the Azure Cloud.
- The three products in the Azure SQL family are:
  - **Azure SQL Database**, a managed database service that includes serverless compute,
  - **Azure SQL Managed Instance**, which includes a fully managed instance as a service and a SQL database engine.
  - **SQL Server on Azure VM's**, offering a full virtual machine and a full SQL Server instance, most recognizable to an on-prem SQL Server sitting on a Windows or Linux host.
- Each product has its own configuration interface with the SQL Server on Azure VM configuration offering the most granular configuration possibilities.
- Azure SQL databases and Azure Managed Instances do not allow third party tools to access the file system and do not provide backups which can be restored to non-Aure SQL Database/Managed Instances, limiting the portability of these databases.

MySQL, PostGreSQL Server, Flex Server (Paas):

- Azure cloud offers all versions of MySQL, MySQL Workbench and PostGreSQL as fully managed pay-as-you-go database services in the cloud.
- Each can be configured on top of Windows or Linux operating systems, and each has their own configuration parameters available at the server level in the Azure portal.
- Flexible servers are available for MySQL and PostgreSQL instances in three computing tiers: Burstable, General Purpose, and Memory Optimized.

ODBC/JDBC Administration:

- To configure database agnostic software connectivity our consultants may install ODBC, JDBC, or manufacturer developed data connectors.
- Installing and configuring a data connector driver enables the ability to query, edit, browse, and manage virtually all major database systems.

Health Checks and Troubleshooting:

- Our SQL Server Instance and Database Health Check diagnostic and troubleshooting package is a proactive, KPI driven, automated reporting package which returns details about system design, relevant OS and SQL Server performance counters, query performance, index usage, status and performance, database settings and information, database usage and trends, memory consumption, database I/O response times and trends, SQL Agent job information and trends, backup and restore history, ownership information, instance update status, and instance security checks.
- Our SQL Server Instance and Database Health Check diagnostic and troubleshooting package will be custom designed for your environment.

SQL Development and ETL Environments:

- If you need a development to production environment to optimize the use of your data infrastructure resources, we can build resources.
- DEV, TEST, QA, UAT to PROD and REPORTING instances can be built with migration processes to keep your team building and grooming data and providing key analytics resources to support your business needs.
- ODBC and JDBC drivers are available for alternate RDBMS systems.
- Consider enterprise service bus architecture using TIBCO Business Works as a framework to integrate disparate sources, complex data mapping, multiple RDBMS systems, legacy systems, and Rest APIs.

Windows .bat file, Powershell Automation:

- Windows batch files and Powershell cmdlets are tools that provide task administration and configuration management in SQL Server and can be used to automate the administration of SQL Server outside of SSMS, execute SQL scripts, backup, configure multiple SQL instances, copy and move data, configure application connections, execute event driven processes, monitor OS performance, execute SQL Agent jobs, and integrate file system tasks with database server operations.

- Windows .bat files and Powershell scripts are integral methods used to fully integrate SQL Server through the operating system to fully automate your data solutions.

Multi Server Configuration:

- Managing multiple SQL Server instances across enterprise domains requires vigilant service with the databases, database servers, and the data itself not to mention backups, jobs, maintenance plans, reporting processes, connectivity, authentication, monitoring database server health, and troubleshooting queries all of which impact applications and the business.
- Our consultants are ready to support with any or all of the above tasks. We are also ready to implement Multi Server Administration configurations such as establishing ‘Master Servers’ and selecting ‘Target Servers’ from which jobs, procedures and scripts can be deployed across the organization.

MS SQL Server RDBMS Licensing:

- Microsoft SQL Server subscription rates are computed by the Server + CAL method (per user method) or by the physical processing Core method (unlimited users per physical hardware cores/processors).
- From these models, costs are computed by SQL Server version and edition.
- Numerous environment related alternatives do apply to arrive at a business use cost projection. Physical server hardware, virtual servers and containers all require SQL Server licensing under the per-Core model. Azure IAAS SQL VM server platforms will require licensing under the per-Core model. Azure PAAS platforms such as Managed Instances and SQL Databases rates are computed on a Pay-as-you-go subscription model.

  Further SQL Server subscription information can be found at: https://microsoft.com/en-us/sql-server/sql-server-2022-pricing.

- The cost of your Microsoft SQL Server subscriptions is a key factor in designing you data processing environment and our consultants are prepared to assist with data environment cost projections as well as preparing your data environment design.

SSMS, Azure Data Studio Upgrade:

- SQL Server Management Studio (SSMS) is an integrated environment for managing any SQL infrastructure from SQL Server to Azure SQL Database. It is used where detailed control and extensive database management is required. Use cases include huge databases, performance tuning and complex data reporting.
- SSMS 19.3 is the latest general availability version and supports SQL Server 2014 versions and later.
- Azure Data Studio is a cross platform data management and development tool with connectivity to cloud and on-prem databases. Use cases include cross-platform development, Azure first environments and customization through the use of its extensive marketplace extensions library.
- The latest general availability release of Azure Data Studio is release number 1.48.0.

Azure Optimization for Databases:

- With the latest versions of Azure SQL Database Microsoft has introduced a number of new applications to help optimize workloads. These applications include Automatic tuning, Automatic Index Management and Adaptive Query Processing.
- These processes can help optimize performance, efficiency, and scalability in your database environment.

Azure Serverless Databases:

- SQL Database Serverless (Paas) optimizes price performance and simplifies performance management for databases with intermittent and unpredictable usage.
- Serverless is a compute tier that allows for auto scaling and auto pausing based on workload demand, thus providing an opportunity to manage your database service cost.
- The serverless database compute tier is measured in a minimum and maximum vCore parameter which you provide, and which proportionally manages the memory and IO limits required.
- Azure SQL Database can serve a number of use cases including: using it as a DR target to ensure business continuity, replicating or migrating data for use in dev/test environments, backup and restore with the use of the automatic backups configured in the service tier, analytics using Azure's advanced analytics platforms such as Azure Storage Blob and Azure Data lake Store, and scaling out read only workloads with the

use of active geo-replication to off load read only workloads such as reporting jobs to secondary copies.

Azure Cost Optimization:

- Our consultants are prepared to assist with designing your Azure cloud data strategy to utilize cloud resources efficiently, minimizing the waste of cloud infrastructure resources.
- Starting steps in governing your Azure SQL cloud resources include
    - right sizing resources to balance between performance and spending
    - clean-up cloud overallocated resources created as a transitional step and are no longer necessary to maintain
    - taking advantage of Azure reservations and savings plans
    - optimizing database and application performance to use less resources enabling you to downsize the original infrastructure
    - enabling stop, pause or offline settings for your Azure resource thereby halting the billing of those resources when not in use.

RECOVERY PLANNING, BACKUPS, SECURITY, AUDITING, AND DISASTER RECOVERY

Database Recovery Models:

- A Recovery Model is a database configuration option that determines the type of backup and the ability to restore the data or recover it from a failure. Selecting a database Recovery Model is a business decision based on what data the database holds and what amount of data loss is acceptable.
- Every database requires a Recovery Model during configuration and the choice impacts your hardware and software resources in addition to data preservation. SQL Server offers three Recovery Models Full, Simple and Bulk Logged. If the business requires full point-in-time recovery, a Full Recovery Model may be appropriate. Databases with acceptable amounts of data loss between backups eliminate the use of the transaction log backups and are configured with the Simple Recovery Model.
- The use of the Full Recovery Model may not be viable for all databases for obvious reasons of cost and complexity.
- The Bulk Logged Recovery Model, similar to the Full Recovery Model, allows for minimal logging in the transaction log but does not allow data to be restored to a specific point in time. It can only restore to the last saved bulk logged activity.
- Ensuring the availability of the data assets in your organization relies on the correct planning and application of Recovery Models.
- In Azure, the default Recovery Model of an Azure SQL Database is Full and can't be changed to any other Recovery Model as in on-prem or IaaS database Recovery Models.

Database Log Shipping for Recovery:

- SQL Server Log Shipping can be configured to provide a DR solution. This is extremely helpful in preventing data loss in case of a failure.
- The process is automated and completely contained within SQL Server and the logic is quite simple. The source database is the primary instance, and a full backup is taken and copied over to the secondary instance which resides on a different destination and restored. Transaction Log backups are then taken on the primary as some time goes by to capture additional current transactions. The logs are then copied over to the secondary instance and restored.

- Log shipping is a database replication method but not without limitations. It does not provide real time synchronization as there is always a delay between the primary and secondary databases due to the backup, copy, and restore steps.

Database Backup and Restore, Recovery:

- Backup planning uses FULL, DIFFERENTIAL, and LOG backups with hourly, daily, weekly, and monthly scheduling. Important backup steps include:
    - Verify backup jobs are completing successfully and the files are accessible.
    - Manage backup file storage and disk space consumption.
    - Ensure backups are stored offsite and in a secure location.
    - Don't forget to script and save stored procedures, views, functions, SQL logins, SQL jobs, etc. These objects are not included in the database backup.
    - Encrypt all backup files in a SQL job in case they become copied or lost.
    - When using SQL Server encryption ensure the appropriate service master keys, database master keys and certificates are also backup and stored.

Backing up the Transaction Log:

- Differential and transaction log backups differ from a full backup in that they only backup the data that has changed since the last full backup.
- Because they are backing up a much smaller data set, they take a much less amount of time to complete, normally a few minutes.
- Differential backups will backup all of the data since the last full backup.
- Transaction log backups will backup all the new data from the moment of any backup.
- The system truncates the transaction log only after a new successful transaction log backup (not a full backup). If you don't take regular transaction log backups the log file continues to grow until it runs out of space.

Testing backups:

- Backup files should be regularly restored on a test server to verify backups are good and can be successfully restored.

Database Level Security:

- Give users the least amount of privileges needed to perform their job.
- Use stored procedures or views to allow users to access data instead of direct access to tables.
- Use Windows Authentication whenever possible instead of SQL Server logins.
- SQL Server backups can be encrypted for additional security.

Server Level Security:

- Limit the number of sysadmins allowed to access the SQL Server.
- Create dual accounts for those with sysadmin access, one to be used when sysadmin access is not absolutely necessary.
- Give users the least amount of privileges needed to perform their job.
- Do not grant permissions to the public database role.
- Disable the Guest User account from each user database by using REVOKE CONNECT FROM GUEST.
- Create an organization process to remove login IDs when they no longer require access.
- Configure login auditing to monitor users who succeed to login and users who fail to login.
- Do not use cross database ownership chaining if not required.
- Never grant permissions to xp_cmdshell to non-sysadmins.
- Do not use SA/sysadmin accounts as accounts used to access SQL Server from applications.
- Remove the BUILTIN/Administrators network group to prevent local server administrators from being able to access SQL Server.
- Limit permissions for SQL Server service accounts to minimum rights, local administrator rights are rarely required, domain administrator rights are never required.
- Do not browse the web from a production SQL Server instance.
- Do not install virus/antispyware on a production SQL Server instance, instead perform scans from a remote server during off peak hours.
- Enable SSL or IPSEC for SQL Server network connections.

Disaster Recovery:

- Ensure your entire IT infrastructure is redundant. It is only as strong as your weakest link.
- Make Disaster Recovery an integral component of your Change Management process.
- Fully test disaster recovery plans once a year.

Windows, Active Directory Users and Groups:

- Create Groups in Active Directory for users with similar permissions.
- Limit Active Directory groups to local access, only use Universal or Domain access when absolutely necessary.
- Audit Active Directory Groups and remove those no longer requiring access.
- A member with an excessive number of Active Directory Groups will fail SQL Login with the MAXTOKENSIZE error.

C2 (Common Criteria) Auditing:

- Using sp_configure advanced option. Will configure the server to record both failed and successful attempts to access statements and objects.
- This information can help you profile system activity and track possible security policy violations.
- This option adds significant performance overhead.

Database Snapshots:

- A database snapshot is a read only, static, transactionally consistent view of a database in the same state as when the snapshot was created.
- Users can query a database snapshot making it useful for reporting purposes, mirroring a database for availability purposes to offload reporting and to safeguard data against administrative or user error or to manage a test database.
- Database snapshots are dependent on the source database and must remain on the same server instance as the source database.
- If the source database becomes unavailable for any reason the snapshot will also be unavailable.

VEEAM/VM Snapshot:

- VEEAM is a third-party application that can be used to produce transactionally consistent backups for all SQL Server backup operations or VEEAM can be used to produce snapshots of your entire virtual machine.
- VM snapshots are not transactionally consistent and may not meet your Recovery Point Objectives (RPO).
- A nightly snapshot of a VM means you are exposed to an 'all or nothing' recovery model and data loss all the way back to when the last VM snapshot was taken.
- A VM snapshot also does not provide a restorable datafile, eliminating many migration and replication opportunities.

Data Center, Cloud, Offsite Migration for a Backup Retention Strategy:

- **The 3-2-1 rule is a longstanding disaster recovery backup strategy that states in order to protect mission critical data organizations must maintain three copies of their data using at least two types of media and at least one copy must be stored off site.**
- Backup files stored within the organization must be on different storage media.
- Storing backups in the cloud is sufficient for one copy of database backup files.
- Backup files must not be overwritten after a new backup is taken. A chain of backup files provides point in time retention enabling the recovery of a past data state.

Microsoft Entra ID for Azure:

- Microsoft Entra ID is an identity and access solution that helps organizations secure hybrid and multi cloud environments. It is similar to Active Directory in a network environment.
- Microsoft describes EntraID as an Identity as a Service solution (IDaaS) for applications and services in the cloud as well as for on-prem solutions.

SQL Server Authentication:

- SQL Server offers two methods of authentication: Windows authentication and mixed-mode authentication.

- Windows authentication enables Active Directory authentication and disables SQL Server login authentication. Mixed-mode enables both Windows authentication and SQL Server login authentication.
- Windows and Active Directory authentication is always available and cannot be disabled.
- SQL Server login authentication is considered less secure than Windows authentication because the password is stored in a system database on the SQL Server and can be backed up and restored with a database backup.

Database User Roles and Permissions:

- Database roles are a set of predefined permissions related to specific tasks or responsibilities.
- Group role users according to their job functions, streamlining permission management.
- Permissions are an authorization to perform specific actions within the system such as viewing data, modifying records or executing commands.
- It is incumbent upon the organization to endorse and apply the use of a set of roles and permissions according to its business needs to protect data assets.
- Managing orphaned users, which are users that retain access to database data without an active owner, is a primary concern in database environments for security administrators.

Row Level Security:

- Row level security in SQL Server is used to restrict users at the database level rather than handling restrictions at the application itself.
- A primary advantage of row level security is you can store data that has different security requirements in the same tables or databases instead of segregating that data into separate tables or databases.
- Row level security is a tool that can be used to protect sensitive data as users can have access to a table without having access to all of the rows on that table.

Database Encryption, Data at rest, in-transit:

- Encryption is a process that can make data useless without the corresponding decryption key.

- Encryption doesn’t solve access control problems however it enhances security by limiting data loss even if the access controls are bypassed.
- The use of encryption should also include a maintenance strategy for the storage of passwords, keys, and certificates.
- Some SQL Server encryption processes to consider include:
    - SSL Transport Encryption to secure data in transit,
    - Transparent Data Encryption (TDE) to protect data at rest,
    - Backup encryption to protect SQL backup files and
    - Column/Cell level encryption to protect sensitive data within columns.

Orphaned Users:

- Orphaned users are user accounts found at the database level but do not have a matching account at the server level.
- This condition often exists as a result of a user’s network security account being dropped but the user was not removed at the database level or as a result of a database backup and restore from a different server as a native SQL database backup includes the users accounts maintaining access at the time of the backup.
- To resolve orphaned users, remove any user account no longer requiring database access. Users requiring continued database access will need to have a new login created and mapped to their Windows SID.

PII Auditing:

- Understanding the importance of data security for PII there are a few initial steps that can be taken to protect PII data.
- PII includes any data that contributes to the unique identification of a person and includes name, address age, SSN, DOB, birth location, credit card details and transactions, medical records, etc..
- The first task is to identify and separate PII data within an organization’s data environment.
- Once identified, PII data can be stored safely with encryption. Roles and permissions can then be adjusted to restrict access to the encrypted PII data assets.
- Particular care should be given to provide protection for protected data replicated during copying, development, migration and backup processes.
- These steps are an initial approach to securing protected data and are supplemented by applying the principle of least privilege in assigning access, strong RDBMS security measures, and maintaining strong network security and physical security.

SOC Auditing:

- SOC or SOX auditing is a financial control audit mandated by the Sarbanes-Oxley Act and is applicable for publicly traded companies.
- SOX specifies four key aspects of controls for IT: access, IT security, data backup, and change management.
- Penalties for violations found in the results of a SOX audit can be significant and warrant particular attention towards maintaining compliance.

HIPPA Compliant Database:

- HIPPA law applies to healthcare professionals, healthcare companies (including healthcare SaaS) as well as cloud service providers that process Protected Health Information (PHI).
- To be HIPPA compliant a database must follow administrative, physical, and technical safeguards of the HIPPA Security Rule.
- The basic requirements for a HIPPA Compliant Database include: Encrypted PHI, Unique User ID's, Database authentication, Database authorization, Database security logging and monitoring, Database backups, Patching, and updating database management software and a PHI disposal capability.

Internal Auditing:

- Organizations may require a defined audit solution to track and log changes or events that are performed on their databases and SQL Instances used to house critical data.
- Some of these audit requirements may be a regulatory requirement such as audits supporting HIPPA, PCI and SOX business requirements.
- SQL Server 2012+ offers several options of internal auditing technology.
- Without using any of the SQL audit options the SQL Server Error Logs, Extended Events, and SQL Triggers are available to review and search for significant event data.
- Within SQL Server a server level auditing option is available in SSMS and can include DML changes to include Change Data Capture, Change Tracking, or System versioned Temporal tables.
- With the SQL Server Enterprise Edition, you are able to broaden your audit scope to the database level

SQL Server Trace Flags, Profiler, Extended Events:

- SQL Traces are used in SQL Server to track and audit certain events occurring in your SQL instance, events are recorded in a trace file stored in the instance log directory.
- SQL Flags are used to adjust SQL Server parameters when required for configuring and tuning an instance.
- Traces and flags should be used sparingly on your SQL instance and for a specific purpose.
- A flag or trace that works well with a particular application or workload may have a totally opposite effect on other processes.
- Many trace flags are enabled for a particular diagnostic event and then are disabled when complete.
- The Extended Events architecture available in SQL Server version 2016+ is the new tool Microsoft provides to collect and monitor the database engine actions to diagnose problems in SQL Server, Azure SQL Database, and Azure SQL Managed Instances. Extended Events is the replacement tool for SQL Profiler which had been deprecated.

DATA DELIVERY, MIGRATION, MONITORING, AND PERFORMANCE

Database, Table Migration (Import/Export):

- Data is frequently migrated in data environments to support delivery, reporting, development, and DR requirements.
- There exists multiple processes to support data migration in SQL Server on-prem and in Azure. Each process has its own characteristics for speed, ease of use, reliability, and level of automation.
- Some migration processes include connectors or drivers capable of negotiating transformation between server types such as SQL Server to Oracle, MySQL, PostGreSQL, MariaDB, and others.
- It is necessary to evaluate the methods and choose which is most appropriate for your migration task.
- Some of the popular tools available for data migration in SQL Server include: Transactional Replication, Bulk Copy, Backup and Restore, Detach and Attach, Import-Export Wizard, Log Shipping, Azure Database Migration Service, Azure Migrate, Azure SQL Migration Extension for Azure Data Studio, Database Experimentation Assistant, Data Migration Assistant, Data Migration Service, SQL Server Migration Assistant, Azure Data Factory, SQL Server Data Tools, a plethora of third party tools, Transact SQL Script, and many more.

Replication (Internal/External):

- Plan replication topology.
- Create distributors, publishers, and subscribers on separate physical hardware.
- Create, document, and test a backup and restore strategy for replication.
- Script replication topology as part of a disaster recovery plan for easy recreation.
- Validate data between publishers and subscribers.
- Monitor replication jobs and processes, performance tune as necessary, and ensure alerts are configured.

Log Shipping for Migration:

- SQL Server Log Shipping can be configured and provides a solution to replicate databases to support multiple reporting activities or alternate use cases.

- The process is automated and completely contained within SQL Server and the logic is quite simple. The source database is the primary instance, and a full backup is taken and copied over to the secondary instance which resides on a different destination and restored. Transaction Log backups are then taken on the primary as some time goes by to capture additional current transactions. The logs are then copied over to the secondary instance and restored.
- Log shipping is a database replication method but not without limitations. It does not provide real time synchronization as there is always a delay between the primary and secondary databases due to the backup, copy and restore steps.

Data Modeling:

- Data Modeling for relational databases is the process of creating a model to organize and store data in a database which conceptually represents data objects, associations, business rules, regulatory compliances, and government policies.
- The modeling process begins with conceptual or semantic modeling and may include Entity Relationship Diagrams (ERD).
- Conceptual modeling recognizes entities, entity names, and relationships between entities.
- The Logical Model comes after the conceptual model and adds attributes, primary keys, and foreign keys.
- Physical Model design completes the data modeling process and adds table names, column names, and column data types.
- The result of the physical model is the final database design and can also include schema definition, indexes, and column constraints.

Performance Optimization:

- Database performance is measured in processing speed vs, resource consumption and optimization begins with a survey of the hardware infrastructure supporting your SQL Server environment.
- Hardware infrastructure can bottleneck workloads within any of the compute, storage and network resources.
- The configuration of the installed SQL Server instance can be modified to meet specific workload demands as the case with minimum and maximum amounts of memory available to the server. parallelism settings, traces and flags enabled and deadlock priority.

- At the database level the design of storage devices can be optimized to reduce contention and speed I/O processes.
- The growth sizes and patterns for your database files, including tempdb, can impact your disk writing performance and statistics and indexing are necessary for data read optimization.
- The design of the database and type of environment it exists in can have a large impact on performance as the case with simplifying OLTP databases with a normalized structure to reduce redundancy and designing a denormalized database structure for databases supporting an OLAP environment.
- Finally, the query used to retrieve data must be evaluated to reveal bottlenecks using the query execution plan. To develop a plan for performance optimization:
    - Establish baselines
    - Use execution plans
    - Use DMV's
    - Test
    - Establish alerts and notifications
    - Monitor performance through the System Monitor and Profiler/SQL Trace
    - Monitor and identify blocking issues

Transaction Isolation Levels:

- Transaction isolation levels provide data integrity and define the degree to which a transaction must be isolated from the data modifications made by any other transaction in the database system.
- The SQL standard defines four transaction levels: READ UNCOMMITTED, READ COMMITTED, REPEATABLE READ, and SERIALIZABLE.
- A transaction level specified for a transaction defines at what level dirty reads, non-repeatable reads, and phantom reads are acceptable and controls the interaction between two concurrent transactions.
- Higher isolation levels offer stronger data consistency but can also result in longer lock times and increased contention.
- Lower isolation levels provide more concurrency but can result in data inconsistencies.

Create Indexes:

- SQL Server indexing is by far the one best way to improve the performance of queries and applications.
- Clustered indexes determine in what order data is written to disk, thereby providing a default format for data retrieval.
- Clustered indexes can be supplemented by non-clustered indexes which are user defined and support alternate data retrieval patterns to speed data retrieval when queries are known to make requests that do not closely match the data order found in the clustered index.
- All indexes are subject to fragmentation through normal insert and delete processes occurring in the database as such indexes regularly decline in performance.
- Regular maintenance, rebuilding, and reorganization of indexes is required to bring the index back up to optimal performance.

Index Statistics:

- The SQL Server Database Engine Query Optimizer uses statistics to create query plans that improve query performance.
- In some cases, you need to create additional statistics or modify the query design for best results.
- All database statistics go ‘stale’ when the number of unique values in an index has changed significantly, this is the result of frequent insert, update, and delete operations.
- To refresh database statistics, you can configure ‘auto create statistics’ and ‘auto update statistics’ to run in the background or manually update statistics in each database.

Index Maintenance:

- Rebuild or reorganize indexes on a schedule, daily-weekly-monthly.
- Rebuilding or reorganizing removes logical fragmentation and wasted space.
- Run the Database Tuning Advisor to identify potential missing indexes.
- Remove indexes that are rarely used.
- Ensure every table has one clustered index, ideally the primary key or on a column that monotonically increases.
- Only add indexes if you know they will be used by queries run against the table.
- Ensure indexes are unique.

- High fill factors with a value of 100 are used for seldom changed data, highly modified data requires a lower fill factor to reduce page splitting.
- Over indexing hurts performance and increases the time it takes to perform INSERTS, UPDATES, and DELETES.
- Use covering indexes for repeating queries and include all columns referenced in the SELECT, JOIN, and WHERE clauses of the query.
- Ensure the first column of a composite index is the most selective column.
- Reduce the width of indexes to reduce the number of I/O reads required, boosting performance.
- Index all columns that are frequently accessed by the JOIN, WHERE, ORDER BY, GROUP BY, TOP, and DISTINCT clauses.

Execution Plans/Query Analysis:

- SQL Server provided three types of execution plans which graphically represent the way which the Query Optimizer has chosen to provide the final query result they are:
    - Estimated Plan
    - Actual Plan
    - Cached Plan
- An analysis of the execution plan provides an opportunity for the user to alter the process SQL Server Query Optimizer has chosen to read and deliver query results. Essentially, you alter the Query Optimizer's process by changing or adding information in your query about the data which will cause the Query Optimizer to select a more performant process.
- Execution plans are completely dependent upon the actual query provided and should be considered for all data retrieval requests.

Managing Locking, Deadlocks:

- Locking, blocking, and deadlocking are essential concepts in database management systems and are necessary to maintain data consistency and integrity while blocking is a natural consequence of locking.
- Deadlocking is an undesirable situation that can lead to severe performance issues.
- By following best practices such as keeping transactions short, acquiring locks in a consistent order, using appropriate lock types, and implementing deadlock detection and resolution mechanisms you can minimize the impact of these issues.

Advanced Table Joins:

- Beyond the basic JOIN options available for relational data, it may be necessary to analyze the join structure, particularly in a multiple join scenario, to prevent data fanning, duplicate data, missing values, and incorrect results from aggregate functions.
- Many-to-one relationships, not joining on unique values and incorrect left join order can all contribute to incorrect results in multiple join scenarios.

Referential Integrity:

- To maintain consistency between tables when defining primary and foreign keys it is necessary to define ON DELETE and ON UPDATE cascading actions.
- These options govern what happens to the data in child (foreign key) tables when the data in the primary key table changes.
- Without these definitions inserts, updates and deletes will fail or selecting the incorrect cascade action will result in inconsistent data between the two tables where the referential relationships are defined.
- There are five options for ON DELETE and ON UPDATE to consider, they are:
  - ON DELETE CASCADE
  - ON DELETE SET NULL
  - ON DELETE SET DEFAULT
  - ON DELETE RESTRICT
  - ON DELETE NO ACTION

Data Warehouse:

- Data warehouses are created to provide strategic information to organizations.
- This strategic information is not available within OLTP resources and requires an analytical transformation to restructure operational data into a dimensional view of business data.
- The organization of a data warehouse typically is either top-down, where the data warehouse becomes a central repository for the entire organization or bottom up, where a collection of departmental data marts are combined into a data warehouse.
- A data warehouse environment within an organization begins with the transformation of operational data into strategic information and ends with an analytical environment capable of producing knowledge for business consumption.

Normalization – Denormalization in a Data Warehouse:

- Normalized databases (OLTP) are designed to reduce the occurrence of pieces of data to eliminate redundancy and improve referential integrity. A data warehouse or data mart may include a strategy to minimize the running time of specific database queries that unite data from many tables into one often include the denormalization of OLTP data to satisfy a particular use case.

Table Views:

- A View in SQL Server is a virtual table that contains data from one or more physical tables.
- A view is created as the result of a query and refreshes its data every time it is executed, and the data does not persist when the view is closed and therefore requires no storage.
- Views are used to:
    - Simplify data access
    - Improve data readability
    - Improve query performance
    - Can be indexed
    - Can be used to provide a data source for reporting applications
- Views can also be used as security mechanisms by allowing users access to data without granting permissions to the physical database table.

Materialized Views:

- A Materialized View in SQL Server is a view that that persists the data returned from the view definition query and automatically gets updated as data changes in the underlying tables.
- It is possible to execute DML (INSERT, UPDATE, and DELETE) statements on a materialized view, but not on a regular view.

Stored Procedures:

- Stored procedures in SQL Server are a group of prepared statements stored in a database that you can save and reuse over and over again.

- The use of stored procedures minimizes the use of slow networks, reduces network traffic, improves round trip response time, is easier to maintain, and is more secure option to deploy code.
- Stored procedures can also be used to manage transactions, ensuring the consistency and integrity of the database. You can explicitly begin, commit, or rollback transactions within a stored procedure.

Merge Operations:

- The SQL Merge statement modifies an existing table based on the result of comparison between key fields with another table in the context.
- It is a popular clause that can handle inserts, updates, and deletes all in a single transaction without having to write separate logic for each.
- When using the optional OUTPUT clause, the Merge statement you can additionally output a table of all the affects rows by type of INSERT, UPDATE, or DELETE action.

DML, DDL, CLR, LOGON Triggers:

- A Trigger is a special type of stored procedure that automatically runs when a particular event occurs in the database server.
- In SQL Server we can create four types of triggers:
    - **Data Definition Language (DDL) Triggers**
        - DDL triggers can be created on CREATE, ALTER and DROP statements.
    - **Data Manipulation Language (DML) Triggers**
        - DML triggers can be created on INSERT, UPDATE and DELETE statements.
    - **Common Runtime Language (CLR) Triggers**
        - CLR triggers allow for triggers to be coded in one of the .NET languages like C#, Visual Basic And F#.
    - **Logon Triggers**
        - Logon triggers fire when a LOGON event of SQL Server is raised.

Cloud Migration, Hybrid management:

- Azure SQL objects include SQL Server in Azure VM, SQL Managed Instance, and the Azure SQL Database.

- There are six primary methods to migrate SQL Server on-prem databases to the Azure cloud.
    - The Backup and Restore method is available for SQL Server in Azure VM and SQL Managed Instances only.
    - The Detach/Attach and Mirroring/Availability Group and Log Shipping methods are only possible in a SQL Server in Azure VM.
    - Replication, Data Migration Assistant, and Bulk Load methods are available in all Azure SQL objects.
- Hybrid configurations with some SQL resources remaining on-prem and some SQL resources migrating to Azure are possible.

Data Quality Services (DQS):

- Data Quality Services is an application that ships with SQL Server and greatly helps in ensuring data quality by data profiling, matching, cleansing, correcting data and monitoring the overall status of these processes.
- It is a knowledge-based tool that allows us to create a knowledge base about the data and use the results to improve the integrity and quality of the data.
- DQS is designed to improve the completeness, accuracy, conformity, and consistency of the data identified for inclusion in the DQS data cleansing process.
- A matching process in DQS can also be configured to remove duplicates.
- The DQS process classifies data to different categories as an output, they are:
    - Correct
    - Corrected
    - Not Corrected
    - Auto Suggested and New

Master Data Services (MDS):

- Master Data Management (MDM) aims at managing a single inventory of master data in an enterprise using a standard set of tools and processes.
- Each arm of the enterprise in each geography would have its own HR, Finance, Sales, and other organizations. As the scale of the diverse forms of data in a business increases, the need to bind these data under a common umbrella becomes even more compelling. These requirements highlight the need for a common set of master data to be maintained in a centralized repository that can be accessed by every section of the organization.

- In general, the process of governing as well as managing master data using a set of standard processes, tools and business rules can be termed as master data management (MDM).
- MDM considerations include identifying master data, collecting master data, data governance, implement MDM tools, and maintenance.

Incident Management, Troubleshooting and Documentation:

- For a DBA incident, management can quickly become a crisis management exercise. Here is an incident event checklist from Brent Ozar to help provide a process for handling incidents and provide a formatted record by which events can be tracked and referenced to help resolve repeat occurrences:

1. **Can You Connect?**
    a. SELECT * FROM sys.Databases
    b. Check returned columns for expected values, especially user_access_desc, state_desc, and log_reuse_wait.
    c. Connected successfully?, Anything notable in returned columns?

Connect with the dedicated admin connection if necessary: http://BrentOzar.com/go/DAC

2. **Who is active?**
    a. Gather a few key pieces of information using Adam Machanic's sp_whoisactive stored procedure: EXEC master.dbo.sp_whoisactive
    b. Did you run this successfully? How many rows did it return? Was blocking present? (See the blocking_session_id column)

More info: http://BrentOzar.com/go/active

3. **What's in the SQL Server Error Log?**
    a. Check with SQL Server Management Studio using 'Management' > 'SQL Server Logs'. Alternately, query the log with this command:
    EXEC xp_readerrorlog @p1=1 /*Error Log Number*/, @p2=1 /*Error Log Type- 1= SQL Server*/
    b. Check all logs since at least the last startup and just prior (Use @p1=2, @p1=3, etc to access prior logs). Are there any recent errors/login failures recorded? When was the last startup date and time? Was the last restart part of planned maintenance?

4. **What's your Quick Assessment?** Check all which appear to be involved:
    a. Access failure (security related), Availability loss (multiple reboots), Performance problem.
    b. Is this an access failure? Verify if this is a security issue, a network issue, an application tier problem, etc.

5. **What's the Windows Event Log Got to Say?**

a. Are there events in the windows logs at the same time or just before the problem periods?

b. System log, Application log, Security log. Also check Configuration Manager Logs, Application Logs

6. Capture SQL Server Activity to a Table

a. Run sp_whoisactive in a loop to log activity to a table. This gathers data while you keep looking.

More info: http://BrentOzar.com/go/activehistory

7. SQL Server Overall Waits

a. Run a query to find the top three SQL Server waits in sys.dm_os_wait_stats by percent since the last restart:
Get a solid query for sys.dm_os_wait_stats for your version of SQL from Glenn Berry: http://BrentOzar.com/go/glenn. Once you open the file search for 'os_wait'.

SQL Server Database Mail:

- Your SQL Server instance can be configured to send reports, query results, job results, notifications, and files using an external SMTP server.
- Applications can be used to send email messages to users.
- SQL Server Database Mail is available in all versions of the SQL Database Engine and in Azure SQL Managed Instance.

Monitoring (incl Azure DB):

- Database monitoring is a set of tools and techniques to track resource consumption for a precise time or in real time.
- Aside from ad hoc monitoring and third-party monitoring tools Microsoft provides the following monitoring tools to assist with performance monitoring in SQL Server on-prem:
  - System Monitor
  - Extended Events
  - Activity Monitor
  - Windows Resources Monitor
  - Database Engine Tuning Advisor (DTA)
  - Performance Dashboard and Query Store
- In Azure you can monitor performance in the Metrics View found in the database Monitoring View.

Oracle Data Migration:

- There are many third-party solutions for migrating Oracle databases on-prem to SQL Server on-prem (see opensource KNIME).
- For native solutions these five solutions are effective:
  - MS SQL Server Migration Assistant for Oracle
  - Import using Oracle Client and SQL Server Management Studio
  - Linked Server on SQL Server pointing to the Oracle database
  - SQL Server Integration Services with the Attunity connector
  - Export to .csv file and import to SQL Server via bulk copy
- In Azure, the SQL Server Migration Assistant (SSMA) is a comprehensive environment that helps you migrate Oracle databases to SQL Server, Azure SQL Database, or Azure SQL Datawarehouse.
- Azure Data Factory is also a potential solution for migrating Oracle data to SQL Server.

MySQL Data Migration:

- There are many third-party solutions for migrating MySql databases on-prem to SQL Server on-prem (see opensource KNIME).
- For native solutions:
  - MySql Workbench has a migration wizard
  - A MySql connector is available in SQL Server Integration Services
  - In SSMS a Linked Server with an ODBC Data Source can be used to migrate data
- In Azure, the SQL Server Migration Assistant (SSMA) is a comprehensive environment that helps you migrate MySQL databases to SQL Server, Azure SQL Database, or Azure SQL Datawarehouse.
- Azure Data Factory is also a potential solution for migrating MySQL data to SQL Server.

PostGreSQL Data Migration:

- There are many third-party solutions for migrating PostGreSQL databases on-prem to SQL Server on-prem (see opensource KNIME).
- For native solutions:
  - CopyTo command in PostGreSQL, good for small migrations, can be used to manually export PostGreSQL data into .csv files for import into SQL Server.
  - A PostGreSQL connector is available in SQL Server Integration Services.

  - In SSMS a Linked Server with an ODBC Data Source can be used to migrate data.
- In Azure, the SQL Server Migration Assistant (SSMA) is a comprehensive environment that helps you migrate PostGreSQL databases to SQL Server, Azure SQL Database, or Azure SQL Datawarehouse.
- Azure Data Factory is also a potential solution for migrating PostGreSQL data to SQL Server.

DATA REPORTING AND ANALYTICS, DATABASE AUTOMATION

SQL Agent Jobs, Azure Elastic jobs:

- SQL Server Agent is a Windows Service that executes scheduled administrative tasks called jobs. SQL Server Agent can run jobs on a schedule, on event, or on demand.
- SQL Agent Job Monitor is used to view the current activity of SQL Server Agent jobs and job steps.
- Create and monitor successful vs failed jobs, operators, proxies, certificates, notifications, and alerts.
- Jobs should be scheduled to run separately at different times, avoid overlapping jobs.
- SQL Agent jobs should include error-trapping, job activity and alerts to notify immediately when a job fails.
- Create a special SQL Server login account whose sole purpose is to run jobs, assign it to all jobs.
- Ensure jobs that contain TSQL code are optimized to run efficiently.
- Script all SQL jobs and save in a secure area in case they are needed to rebuild servers.
- SQL Agent jobs are not included in the SQL Server database backup.
- Azure SQL databases do not have the SQL Server Agent service however there are some work arounds:
  - Create a SQL job on an on-prem SQL instance to connect to and run on the Azure SQL Database
  - Use Azure Automation to schedule jobs in Azure and automate manual tasks
  - Use the Elastic Database Jobs service in Azure cloud to execute ad hoc tasks
  - Use Powershell on-prem or in an Azure SQL VM to automate a task and schedule script execution

Data Maintenance Jobs, Maintenance Plans:

- The SQL Server maintenance Plan Wizard creates an Integration Services package which is run by SQL Server Agent.
- Can be used to schedule and execute maintenance jobs on local and remote SQL Server instances. Best practices include jobs for:
  - DBCC Checkdb
  - SP_updatestats
  - Backup Database FULL
  - Copy Backup file to another server IAW backup strategy

SQL Server Integration Services (SSIS), SQL Server Data Tools (SSDT):

- SSIS is an ETL platform and development environment for building enterprise level data integration and data transformation solutions.
- It is an essential data migration and data warehousing tool.
- Installing SSDT creates a fully functional Integrated Development Environment for Windows and includes MS Visual Studios 2019 (and now VS2022) as well as the SSIS development extension.
- Microsoft offers an Azure-SSIS Integration Runtime (Azure-SSIS IR) to deploy, configure, run, monitor, schedule, and manage packages on premises and on Azure and/or with Powershell.
- The Azure-SSIS IR is available in Azure Data Factory and only needs to be installed once.

SSISDB Catalog:

- The SSISDB Catalog is created in SQL Server Management Studio (SSMS) and is the central point for the control and execution of the SSIS packages created on the SQL Server Integration Server. In the SSIS Catalog you can configure environments to specify run time values, set project and package parameters, store, execute, receive operational history reports, and troubleshoot executions from SSIS packages.

Incremental Loading, SSIS:

- Incremental loading is a technique that compares the source and destination data and identifies the new, updated, or deleted records. Then it applies only the necessary changes to the destination instead of loading the entire data set every time.
- Incremental loads are accomplished in SSIS by configuring a special Data Flow Task which includes a series of lookup tables to compare the source with the destination determining records with no match (inserts), records that change (updates), and records that are removed (deletes).

Change Data Capture:

- Change Data Capture (CDC) allows you to capture data modifications to CDC enabled databases and tables. The SQL Server CDC feature is important to incremental load scenarios such as incrementally inserting changed data to a data warehouse from an OLTP environment.
- CDC requires the SQL Agent service which is not available in Azure SQL Database however you can use the temporal table, SSIS, or Azure Data Factory to implement CDC.

Databases for SharePoint:

- SharePoint uses a web application, content database, and site collection to manage its data.
- The data stored in the content database can be housed in a SQL Server and usually consists of sites, permissions, documents, lists, etc.
- SharePoint administrators use the Central Administration web application to configure and set the properties for content databases. Use the Central Administration web application to associate web applications with content databases and site collection containers to content databases. You will also manage application pool accounts, mirrored databases and failover servers, authentication, and database ready and offline statuses.
- Maintenance tasks, backups, indexing, etc. is managed in the SQL Server used to store the SharePoint databases and can be managed through SQL Server Management Studio (SSMS).

Linked Servers (Internal/External):

- Linked servers are a method by which a SQL Server instance can communicate with other ODBC compliant databases (ie., Oracle, MySQL) with a direct TSQL OPENQUERY() thus creating a distributed database.
- Linked server ODBC connectors also include providers for CSV files, speadsheet files, some NoSQL databases, Salesforce, etc…
- Linked server configuration will require a local and remote server login with necessary permissions.
- SQL Server linked services are able to be configured to connect to Azure SQL Managed Instances and Azure SQL Databases.

Data Extraction, Transformation, and Loading (ETL):

- ETL is the process of acquiring your organization's disparate transactional data (OLTP), staging, cleansing, and optimizing that data into an analytical format (OLAP) or analytics database then delivering that data to a storage area where it can be consumed by a Business Intelligence (BI) application.
- There are numerous integration applications which specialize in on-prem and cloud ETL processes including SSIS, TIBCO, Azure Data Factory, Amazon Redshift and Google BigQuery.
- Modern analytics architecture includes a shift from ETL to ELT where on-premise data is shifted to cloud-native data warehouses where it is prepared for transformation for analytics consumption. New tools in the modern marketplace are being used to move and access this data include API's and webhooks.

Azure Data Factory, Rest API's, Data Pipelines:

- Azure Data Factory (ADF) is an Azure cloud based ETL and integration service that supports data migration and transformation activities between cloud-to-cloud locations and cloud to on-premises locations. Within ADF you can move and transform data, schedule pipelines, transform through a computing service then publish to an application or storage location.
- API's allow applications to extend and reuse business logic, data, and processes in the form of a service. Within ADF you can consume 3rd party/HTTP API data resources by configuring a data pipeline to connect to the API, make a data request then move the result to storage.

Postman, API Testing:

- Postman is an open source tool used for the initial programming and testing of the REST API/HTTP methods GET, POST, PUT, PATCH, DELETE, headers and cookies dealing with the uploads or authentication with API keys, tokens, Oath and more.
- The validated results of a successful Postman API test are easily inputted into a more complex ADF data pipeline to provide additional configuration and transformation of the data uploaded or downloaded in the initial API request.

EXCEL Services:

- There are numerous methods to import Excel spreadsheet data into SQL Server or to Azure SQL database. Some of these methods require conversion to flat files (.csv) prior to importing the data.
- Methods to import data directly from Excel for on-prem SQL instances include:
  - SQL Server Import Export Wizard
  - SQL Server Integration Services (SSIS)
  - TSQL using the OPENROWSET function
- Methods using export to text for SQL Server and Azure SQL database include:
  - Import Flat File Wizard
  - Bulk Insert Statement
  - BCP
  - Copy Wizard (ADF)
  - Azure Data Factory

Reporting Services (SSRS) Server:

- Reporting Services (SSRS) is a server-based report generating application that is a component of SQL Server.
- SSRS consists of four main components the Reporting Services Service, the Web portal, the Report Designer, and the Report Builder.
- SSRS allows organizations to create enterprise level, paginated, decision making information in a variety of formats based on the information stored in SQL Server databases and other data sources.

SSRS Installation and Configuration:

- Since SQL Server 2016, the SSRS server is available from Microsoft through a separate downloadable installer. SSRS is often installed on a separate Windows host server to allow configuration for multiple read processes, report retrieval, and report delivery.
- The Azure SQL Server Managed Instance can be used as a backend for SSRS server databases and reduces the need for an IaaS installation to host SSRS in Azure.

Migrate Legacy Reports:

- Migrating legacy reporting tools to a new reporting tool for reports that are still valid but the data source or the reporting tool has essentially become obsolete is a complex task that requires planning.
- First users should vet the older reports or sections of the report to ensure they are still required, backup and retire the old data as appropriate. The report itself can contain extensive calculations and formulas that will also require vetting, backup, and retirement if appropriate.
- Data sources to be carried forward have to be analyzed for data quality and data compatibility with newer enterprise data resources. Data types also have to be looked at to ensure compatibility. Then an ETL process needs to be created to perform the data migration.
- The reports themselves will require redesign and the calculations and formulas recreated in the new reporting tool.
- This process is different than a regular migration of data from old RDBMS format to new or from on-prem to cloud. Sometimes a third-party tool can be located to perform the task but often it becomes a slow manual process.

Data Strategy:

- DBA Services can be likened to a broad assembly floor with thousands of individual components that have only limited interaction awaiting assembly into a performance automobile. What's needed to put this vehicle together is a corporate plan, a Data Strategy.
- Data Strategy is an enterprise plan for managing an organization's data and for investing in the organization's capabilities to effectively use the data for benefit. Data Strategy **IS** Corporate Strategy. Don't embark on data and analytics initiatives without a strong tie to your organization's strategic direction.
- Analytics initiatives must be managed as a portfolio of ROI generating projects leveraging targeted investment in new capabilities. Our consultants have proven methodologies for managing this process.

Power BI:

- Considering the PowerBI platform to power your analytics? Our Data Engineering consultants offer a one-day instruction class on PowerBI which covers the complete installation to visualization process. This class is suitable for individual or group sessions and is a fantastic tool for your development team to break the ice and become familiar with PowerBI.

Analytics Platforms (Tableau, PowerBI, TIBCO Spotfire, MS Analytics Platform Service, others):

- Grooming data sources for consumption by analytics platforms requires a comprehensive enterprise data strategy which has been prepared with leadership input to provide the business goals, user interaction to scope the data, possibly a data scientist to develop statistical models and algorithms to collect and organize data, as well as a data engineer who designs and creates an ETL process to deliver data to the analytics platform. No matter which analytics platform you select the data strategy and data preparation tasks will be very similar.
- Database administrators often assist analytics processes with preparing organizational data sources for the Big Data/Analytics data strategy processes but often these tasks are more appropriately driven by business analysts, data architects, data engineers and data warehouse developers.

Dashboard Creation, KPI's, Business Intelligence:

- With a selected analytics platform or reporting tool you can create dashboards and meaningful data visualizations using data and metrics derived from your SQL database.

- Key Performance Indicators (KPI) are a favorite tool used to communicate the amount of progress made toward a goal. KPI's are business goal defined and are a great way to answer:
    - What am I ahead or behind on?
    - How far ahead or behind am I?
    - What minimum amounts did I complete?
- Dashboards and KPI's are combined to create executive level, data-driven, actionable, decision-making reporting visualizations. Server based platform tools such as PowerBI, Tableau, TIBCO Spotfire, and SSRS are used to deliver these business intelligence analytics visualizations.

Knime Analytics:

- The Knime Analytics Tool is an open source analytics platform use to access, blend, transform, visualize, and analyze data from any source.
- With Knime, you can enrich your data sets and create data science solutions by integrating Python, R, and Javascript components into your data sets.
- Knime workflows are a great way to evaluate and wireframe analytics workflows before engaging in a larger BI project.

TASK CHARTS FOR DBA SERVICE OFFERINGS

Infrastructure (Installs, Maintenance, and Administration)

Service Code 1	
	Databases New, Migrate
	Server - Cumulative Updates and Patching
	Server Version Upgrades
	Database Maintenance
	Database Consistency, DBCC Check Table
	Offline, Suspect or Recovery Pending databases
	Manage SQL Server System Databases (master, model, msdb, tempdb)
	Memory management, Parallel processing
	Data Type Selection & Changes
	Clustered Instances, Log Shipping
	Transaction Log Maintenance
	New SQL Server Installation and Configuration
	Configuration Manager
	Capacity Planning
	SQL Server Hardware Optimization (CPU, Mem, Drives, Space)
	Host Server Drive Layout Best Practices
	SQL Server Advanced Options - Enabling XP_CMDSHELL and SQLCMD, DAC
	Azure SQL Server Databases and Configuration
	MySQL, PostGreSQL Server, Flex Server (Paas)
	ODBC/JDBC Administration
	Health Checks and Troubleshooting
	SQL Development and ETL Environments
	Windows .bat file, Powershell Automation
	Multi Server Configuration
	MS SQL Server RDBMS Licensing
	SSMS, Azure Data Studio Upgrade
	Azure Optimization for Databases
	Azure Serverless Databases
	Azure Cost Optimization

Recovery Planning, Backups, Security, Auditing, Disaster Recovery

Service Code 2	
	Database Recovery Models
	Database Log Shipping for Recovery
	Database Backup and Restore, Recovery
	Backing up the Transaction Log
	Testing backups
	Database Level Security
	Server Level Security
	Disaster Recovery
	Windows, Active Directory Users and Groups
	C2 (Common Criteria) Auditing
	Database Snapshots
	VEEAM/VMWare Snapshot
	Data Center, Cloud, Offsite Migration for a Backup Retention Strategy
	Microsoft Entra ID for Azure
	SQL Server Authentication
	Database User Roles and Permissions
	Row Level Security
	Database Encryption, Data at rest, in-transit
	Orphaned Users
	PII Auditing
	SOC Auditing
	HIPPA Compliant Database
	Internal Auditing
	SQL Server Trace Flags, Profiler, Extended Events
	Database Recovery Models
	Database Log Shipping for Recovery
	Database Backup and Restore, Recovery
	Backing up the Transaction Log
	Testing backups

Data Delivery, Monitoring, Performance

Service Code 3	
	Database, Table Migration (Import/Export)
	Replication (Internal/External)
	Log Shipping for Migration
	Data Modeling
	Performance Optimization
	Transaction Isolation Levels
	Create Indexes
	Index Statistics
	Index Maintenance
	Execution Plans/Query Analysis
	Managing Locking, Deadlocks
	Advanced Table Joins
	Referential Integrity
	Data Warehouse
	Normalization – Denormalization in a Data Warehouse
	Table Views
	Materialized Views
	Stored Procedures
	Merge Operations
	DML, DDL, CLR, LOGON Triggers
	Cloud Migration, Hybrid management
	Data Quality Services (DQS)
	Master Data Services (MDS)
	Incident Management, Troubleshooting and Documentation
	SQL Server Database Mail
	Monitoring (incl Azure DB)
	Oracle Data Migration
	MySQL Data Migration
	PostGreSQL Data Migration

Data Reporting and Analytics, Database Automation

Service Code 4	
	SQL Agent Jobs, Azure Elastic jobs
	Data Maintenance Jobs, Maintenance Plans
	SQL Server Integration Services (SSIS), SQL Server Data Tools (SSDT)
	SSISDB Catalog
	Incremental Loading, SSIS
	Change Data Capture
	Databases for Sharepoint
	Linked Servers (Internal/External)
	Data Extraction, Transformation, and Loading (ETL)
	Azure Data Factory, Rest API's, Data Pipelines
	Postman, API Testing
	EXCEL Services
	Reporting Services (SSRS) Server
	SSRS Installation and Configuration
	Migrate Legacy Reports
	Data Strategy
	PowerBI
	Analytics Platforms (Tableau, PowerBI, TIBCO Spotfire, MS Analytics Platform Service, others)
	Dashboard Creation, KPI’s, Business Intelligence
	Knine Analytics