Cornerstone

Administering a SQL Server Database

T2764

Table of contents

T2764

Administering a SQL Database

Facilities

- Class hours
- Building hours
- Restrooms
- Meals

About This Course

- Audience
- Course Prerequisites
- Course Objectives
- What You Can Expect

Course Outline

- **Module 1**, Authentication and authorization
- **Module 2**, Server roles and database roles
- **Module 3**, Granting permissions
- **Module 4**, Auditing and encryption
- **Module 5**, Backup strategies and recovery models
- **Module 6**, Performing backup in SQL Server
- **Module 7**, Restore
- **Module 8**, SQL Server Agent jobs
- **Module 9**, SQL Server Agent security
- **Module 10**, SQL Server error messages, Database Mail and Agent alerts
- **Module 11**, Managing SQL Server Using PowerShell
- **Module 12**, Extended Events
- **Module 13**, Monitoring SQL Server
- **Module 14**, Troubleshooting
- **Module 15**, Data transfer

Module 1

Authentication and authorization

Module Overview

- Authenticating connections
- Authorizing logins to connect to databases
- Linked servers and orphaned users
- Contained databases

Lesson: Authenticating connections

- Overview of security in SQL Server
- Authentication
- Managing logins and login policies

Overview of security in SQL Server

- Securables
  - Objects to which access should be secured
- Principals
  - Security identities that access and perform actions against securables
- Permissions
  - The actions principals is allowed to perform on a securable
- Security Hierarchies:
  - Securables can contain other securables
  - Principals can contain other principals
  - Permissions are inherited unless overridden

Authentication

- Two types of *logins*
  - SQL Server login
    - Specify a name and password
    - Name is stored in clear-text, password is hashed
  - Windows login
    - You “allow” a Windows user or group to authenticate/connect
    - SQL Server store name and SID
- Two security modes
  - Allow only Windows authentication
  - Allow both types
  - Read from the registry at start-up
  - Set at install time, can change using SSMS

Choose an Authentication Mode

https://docs.microsoft.com/en-us/sql/relational-databases/security/choose-an-authentication-mode

Managing logins and login policies

- Logins:
  - Create in SQL Server Management Studio
  - Create a login using the CREATE LOGIN statement

```
CREATE LOGIN [MYDOMAIN\Controllers] FROM WINDOWS
WITH DEFAULT_DATABASE = AdventureworksDW
```

  - Create and set security policy for a SQL Server Login

```
CREATE LOGIN Bob
WITH PASSWORD = 'Th3Passw0rd.', CHECK_POLICY = ON
```

  - Disable a logins

```
ALTER LOGIN Bob DISABLE
```

  - Delete a login

```
DROP LOGIN Bob
```

Demo...

CREATE LOGIN (Transact-SQL)

https://docs.microsoft.com/en-us/sql/t-sql/statements/create-login-transact-sql

ALTER LOGIN (Transact-SQL)

https://docs.microsoft.com/en-us/sql/t-sql/statements/alter-login-transact-sql

DROP LOGIN (Transact-SQL)

https://docs.microsoft.com/en-us/sql/t-sql/statements/drop-login-transact-sql

Lesson: Authorizing logins to connect to databases

- Granting access to databases
- The database owner, dbo and guest user

Granting access to databases

- Logins cannot access a database to which they have not been granted access
  - Except if the login has the sysadmin server role
    - A sysadmin accesses all databases as the dbo user
- Grant access to a login by creating a database user for the login

```
CREATE USER [MYDOMAIN\Controllers]
CREATE USER [MYDOMAIN\Controllers] FOR LOGIN [MYDOMAIN\Controllers]
CREATE USER Controllers FOR LOGIN [MYDOMAIN\Controllers]

CREATE USER Bob
```

Demo...

CREATE USER (Transact-SQL)

https://docs.microsoft.com/en-us/sql/t-sql/statements/create-user-transact-sql?

ALTER USER (Transact-SQL)

https://docs.microsoft.com/en-us/sql/t-sql/statements/alter-user-transact-sql

The database owner, dbo and guest user

- The owner of a database is registered
  - In the master database, as seen in sys.databases
  - In the user database, the login that the dbo user refers to
  - Possibility for inconsistency when you move databases
  - Consider having:
    - The same login owning all databases
    - Or having a login for each database with the sole purpose of owning that database
- The dbo user
  - The owner of the database seen from that database's perspective.
  - Member of the sysadmin role operates as dbo in all databases.
- guest database user:
  - Enables logins without user accounts to access a database
  - Disabled by default in user databases
  - Enabled by using the GRANT CONNECT statement

Principals (Database Engine)

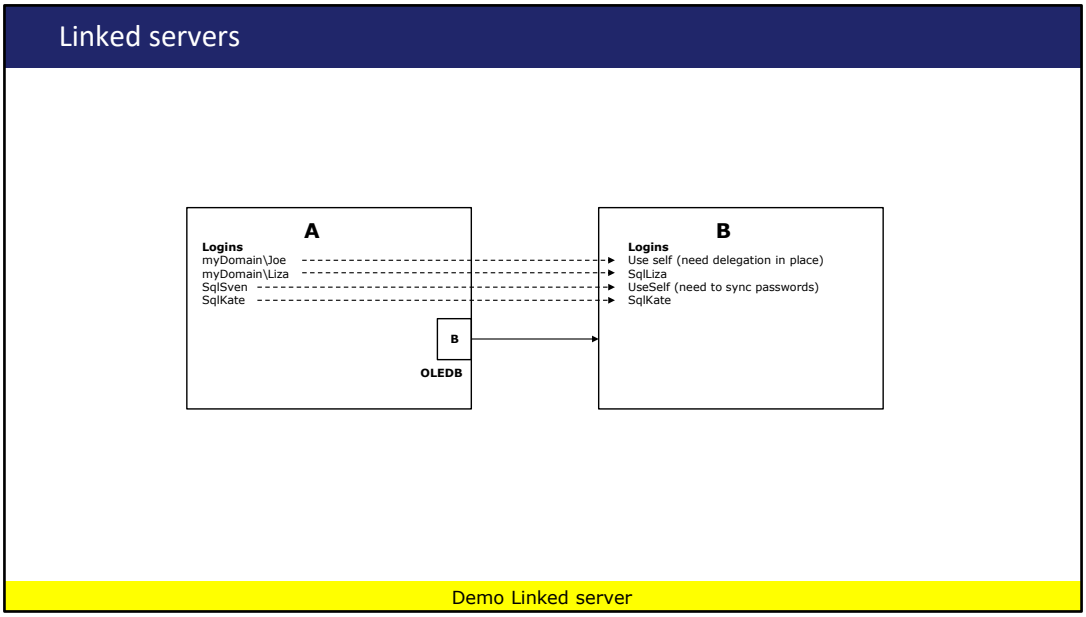
https://docs.microsoft.com/en-us/sql/relational-databases/security/authentication-access/principals-database-engine

Among other things, this article discusses why it might be a good idea to have one login per database, where the sole purpose for if the login is to be an owner of that database

https://www.sommarskog.se/grantperm.html

Lesson: Linked servers and orphaned users

- Linked servers
- Linked servers security
- Linked servers and delegation
- The EXECUTE AS command
- Orphaned users



Linked Servers (Database Engine)

https://docs.microsoft.com/en-us/sql/relational-databases/linked-servers/linked-servers-database-engine

Create linked servers (SQL Server Database Engine)

https://docs.microsoft.com/en-us/sql/relational-databases/linked-servers/create-linked-servers-sql-server-database-engine

sp_addlinkedserver (Transact-SQL)

https://docs.microsoft.com/en-us/sql/relational-databases/system-stored-procedures/sp-addlinkedserver-transact-sql

Linked servers security

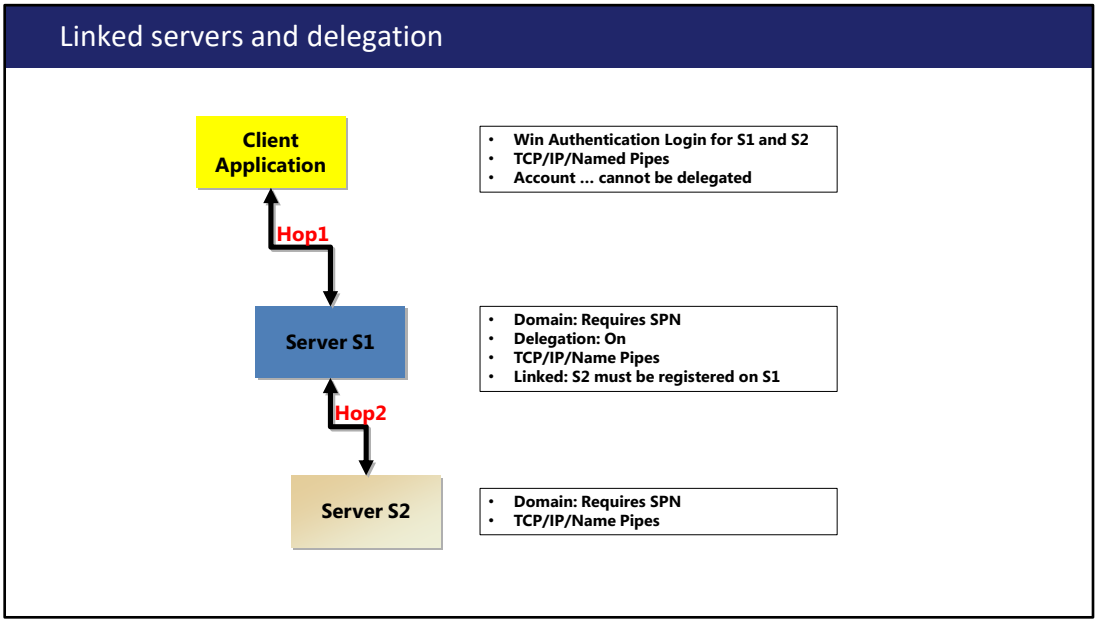
- Authenticated access to external data sources
- Link server objects:
  - Provider and data sources
- Configuration:
  - Client, server, database server tiers
  - Definitions

```
EXEC sp_addlinkedserver
@server='NORTH\TEST',
@srvproduct='SQL Server',
```

- Security

Mapping logins between linked servers:

https://docs.microsoft.com/en-us/sql/relational-databases/system-stored-procedures/sp-addlinkedsrvlogin-transact-sql



Service Principal Name (SPN) Support in Client Connections

https://docs.microsoft.com/en-us/sql/connect/oledb/features/service-principal-name-spn-support-in-client-connections

How to link two SQL Server instances with Kerberos

https://www.sqlshack.com/how-to-link-two-sql-server-instances-with-kerberos/

The EXECUTE AS command

- EXECUTE AS LOGIN
  - Changes the session's security context to that login's context
- EXECUTE AS USER
  - Changes only the user context
  - Sandbox the session to that database
- REVERT
  - Make you "yourself" again
- Use the ORIGINAL_LOGIN() function
  - For auditing and similar

EXECUTE AS (Transact-SQL)

https://docs.microsoft.com/en-us/sql/t-sql/statements/execute-as-transact-sql

ORIGINAL_LOGIN (Transact-SQL)

https://docs.microsoft.com/en-us/sql/t-sql/functions/original-login-transact-sql

Orphaned users

- Orphaned users created by mismatched SIDs
  - Search for logins in the database that do not exist on the server
  - Avoid using CREATE LOGIN … WITH SID…
  - Or resolve using ALTER USER name WITH LOGIN = name
  - Consider Windows authenticated accounts

Demo Orphaned users

Troubleshoot orphaned users (SQL Server)

https://docs.microsoft.com/en-us/sql/sql-server/failover-clusters/troubleshoot-orphaned-users-sql-server

Moving a database between two SQL Server instances

https://karaszi.com/moving-a-database-between-two-sql-server-instances

Lesson 4: Contained databases

- Introduction to contained databases
- Considerations when using contained databases

Introduction to contained databases

- Contained databases do not have a dependency to logins
- Use contained databases to:
  - Move databases between different SQL Server instances without having to migrate some server-level dependencies
  - Develop databases when the developer does not know which instance will ultimately host the database
  - Enable failover in a high-availability scenario without having to synchronize server-level logins
- Users in a contained database include:
  - Users mapped to Windows accounts (users or groups)
  - Users with passwords

Demo Contained Database

Contained Databases

https://docs.microsoft.com/en-us/sql/relational-databases/databases/contained-databases

Contained Database Users - Making Your Database Portable

https://docs.microsoft.com/en-us/sql/relational-databases/security/contained-database-users-making-your-database-portable

Security Best Practices with Contained Databases

https://docs.microsoft.com/en-us/sql/relational-databases/databases/security-best-practices-with-contained-databases

Considerations when using contained databases

- Benefits:
  - Migration
  - Failover, including Always On Group Availability
  - Administration
  - Development
- Considerations:
  - CDC, CT, replication not allowed
  - Numbered procedures not supported
  - Collation
  - Password policy not optional, CREATE USER
  - ALTER DATABASE CURRENT
  - Connection strings must be explicit
  - Cross database queries

Lab 1: Authenticating Users

- Ex 1. Verify Windows groups and accounts
- Ex 2. Set SQL server to mixed security mode
- Ex 3. Create logins
- Ex 4. Create users for your logins
- Ex 5. If time permits, manage an orphaned user

Estimated Time: 40 minutes

Module 2

Server roles and database roles

Module Overview

- Server level roles
- Database level roles

Lesson: Server level roles

- Server-scoped permissions
- Examples of server-scoped permissions
- User-defined server roles
- The built-in server roles
- New fixed server roles in 2022

Server-scoped permissions

- Control access to server resources
- Organized as a hierarchy
  - CONTROL SERVER at the top of the hierarchy
  - Granting a permission to a server principal implicitly grants its child permissions
- Can only be granted to server principals (not to database principals)

Permissions (Database Engine)
https://docs.microsoft.com/en-us/sql/relational-databases/security/permissions-database-engine

Examples of server-scoped permissions

- ADMINISTER BULK OPERATIONS
- ALTER ANY LINKED SERVER
- ALTER ANY LOGIN
- ALTER ANY DATABASE
- ALTER SETTINGS
- ALTER SERVER STATE
- CONTROL SERVER
- CREATE ANY DATABASE
- VIEW SERVER STATE

Demo Server permissions

New granular permissions for SQL Server 2022 and Azure SQL to improve adherence with PoLP
https://techcommunity.microsoft.com/t5/sql-server-blog/new-granular-permissions-for-sql-server-2022-and-azure-sql-to/ba-p/3607507

User-defined server roles

- Managing User-Defined Roles
  - CREATE SERVER ROLE
  - DROP SERVER ROLE
- Managing Permissions
  - GRANT, DENY and REVOKE
- Managing Membership
  - ALTER SERVER ROLE

CREATE SERVER ROLE (Transact-SQL)
https://docs.microsoft.com/en-us/sql/t-sql/statements/create-server-role-transact-sql

ALTER SERVER ROLE (Transact-SQL)
https://docs.microsoft.com/en-us/sql/t-sql/statements/alter-server-role-transact-sql

The built-in server roles

- Fixed:
  - Sysadmin
    - Can do everything
  - Securityadmin
    - Can manage logins. Be careful, can (almost) make itself sysadmin.
  - Serveradmin
    - Can run sp_configure and execute SHUTDOWN command.
  - Processadmin
    - Can execute KILL command.
  - Setupadmin
    - Can configure linked servers.
  - Bulkadmin
    - Can execute the BULK INSERT command.
  - Diskadmin
    - Manage backup devices
  - Dbcreator
    - CREATE, ALTER and DROP DATABASE
- Public
  - All logins are member of public. No default permissions (except for CONNECT and VIEW ANY DATABASE).

Server-Level Roles
https://docs.microsoft.com/en-us/sql/relational-databases/security/authentication-access/server-level-roles

New fixed server roles in 2022

- ##MS_DatabaseConnector##
  - Connect to any database without requiring a User-account in the database to connect to
- ##MS_DatabaseManager##
  - Create and delete databases
- ##MS_PerformanceDefinitionReader##
  - Read all catalog views that are covered by VIEW ANY PERFORMANCE DEFINITION, and respectively has VIEW PERFORMANCE DEFINITION permission on any database on which the member of this role has a user account.
- ##MS_SecurityDefinitionReader##
  - Read all catalog views that are covered by VIEW ANY SECURITY DEFINITION, and respectively has VIEW SECURITY DEFINITION permission on any database on which the member of this role has a user account.
- ##MS_DefinitionReader##
  - Read all catalog views that are covered by VIEW ANY DEFINITION, and respectively has VIEW DEFINITION permission on any database on which the member of this role has a user account.
- ##MS_LoginManager##
  - Create and delete logins
- ##MS_ServerPerformanceStateReader##
  - Read all dynamic management views (DMVs) and functions that are covered by VIEW SERVER PERFORMANCE STATE, and respectively has VIEW DATABASE PERFORMANCE STATE permission on any database on which the member of this role has a user account.
- ##MS_ServerSecurityStateReader##
  - Read all dynamic management views (DMVs) and functions that are covered by VIEW SERVER SECURITY STATE, and respectively has VIEW DATABASE SECURITY STATE permission on any database on which the member of this role has a user account.
- ##MS_ServerStateReader##
  - Read all dynamic management views (DMVs) and functions that are covered by VIEW SERVER STATE, and respectively has VIEW DATABASE STATE permission on any database on which the member of this role has a user account.
- ##MS_ServerStateManager##
  - Members of the ##MS_ServerStateManager## fixed server role have the same permissions as the ##MS_ServerStateReader## role. Also, it holds the ALTER SERVER STATE permission, which allows access to several management operations, such as: DBCC FREEPROCCACHE, DBCC FREESYSTEMCACHE ('ALL'), DBCC SQLPERF().

Server-Level Roles
https://docs.microsoft.com/en-us/sql/relational-databases/security/authentication-access/server-level-roles

Lesson: Database level roles

- Database scoped permissions
- Assigning database principals to database roles
- User defined database roles
- Fixed database roles
- Application roles

Demo Database permissions

Database scoped permissions

- Control access to database resources
- Organized as a hierarchy
  - CONTROL at the top of the hierarchy
  - Granting a permission to a database principal implicitly grants it child permissions
- Can only be granted to database principals
  - Not to a login
- Some server permissions implicitly grant database permissions

Permissions (Database Engine)
https://docs.microsoft.com/en-us/sql/relational-databases/security/permissions-database-engine

Assigning database principals to database roles

- Add and remove members from a role using the ALTER ROLE statement
- A role member may be a user or a user-defined role
- Fixed database roles cannot be members of other roles
- Membership of database roles can only be assigned to database principals.
  - Server principals cannot be assigned membership of database roles

ALTER ROLE (Transact-SQL)
https://docs.microsoft.com/en-us/sql/t-sql/statements/alter-role-transact-sql

User defined database roles

- Managing User-Defined Roles
  - CREATE ROLE
  - DROP ROLE
- Managing Permissions
  - GRANT, DENY and REVOKE
- Managing Membership
  - ALTER ROLE

CREATE ROLE (Transact-SQL)
https://docs.microsoft.com/en-us/sql/t-sql/statements/create-role-transact-sql

Fixed database roles

- Fixed roles common to all databases:

db_owner	**db_datawriter**
db_securityadmin	**db_datareader**
db_accessadmin	**db_denydatawriter**
db_backupoperator	**db_denydatareader**
db_ddladmin	**public**

- **msdb** has additional pre-defined roles for managing jobs, SSIS etc.

Database-level roles
https://docs.microsoft.com/en-us/sql/relational-databases/security/authentication-access/database-level-roles

Application roles

- Security context of the user is replaced by the application role
- Creating Application Roles
  - Use CREATE APPLICATION ROLE
  - Password must meet Windows password policy
- Using Application Roles
  - Use sp_setapprole
  - Use a secure network connection to avoid leaking the application role password
  - Exit application role by closing connection or using sp_unsetapprole (requires a stored cookie)
  - Limited to guest access to other databases

Demo Application roles

Application Roles
https://docs.microsoft.com/en-us/sql/relational-databases/security/authentication-access/application-roles

CREATE APPLICATION ROLE (Transact-SQL)
https://docs.microsoft.com/en-us/sql/t-sql/statements/create-application-role-transact-sql

Lab 2: Assigning Server and Database Roles

- Ex 1: Server roles and server permissions
- Ex 2: Database roles and database permissions
- Ex 3: Test the roles and permission

Estimated Time: 40 minutes

Module 3

Granting permissions

Module Overview

- Securables, principals and granting privileges
- Object level security
- Schema level security

Lesson: Securables, principals and granting privileges

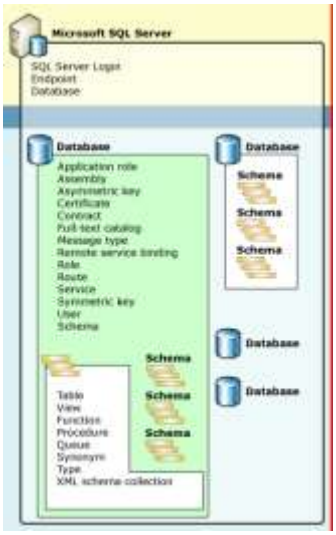
- Principals
- Securables
- GRANT, DENY and REVOKE
- WITH GRANT OPTION

Principals

- Instance / Server Principals:
  - SQL Server login
  - Server role
- Database Principals:
  - User
  - Database role
  - Application role

Principals (Database Engine)
https://docs.microsoft.com/en-us/sql/relational-databases/security/authentication-access/principals-database-engine

Securables



Securables
https://docs.microsoft.com/en-us/sql/relational-databases/security/securables

Permissions Hierarchy (Database Engine)
https://docs.microsoft.com/sv-se/sql/relational-databases/security/permissions-hierarchy-database-engine

GRANT, REVOKE, DENY

- GRANT assigns a permission
- DENY explicitly denies a permission:
  - Use to deny inherited permissions
  - Use only in exceptional circumstances
- REVOKE removes both GRANT and DENY permissions

Permissions (Database Engine)
https://docs.microsoft.com/en-us/sql/relational-databases/security/permissions-database-engine

Permissions Hierarchy (Database Engine)
https://docs.microsoft.com/en-us/sql/relational-databases/security/permissions-hierarchy-database-engine

WITH GRANT OPTION

- Use the WITH GRANT OPTION to enable the principal to grant the same permissions to other users

```
GRANT UPDATE ON Production.Product
      TO myGroup
      WITH GRANT OPTION
```

- Use CASCADE to revoke or deny these permissions from the principal and the other users

```
REVOKE UPDATE ON Production.Product
      FROM myGroup
      CASCADE
```

GRANT (Transact-SQL)
https://docs.microsoft.com/en-us/sql/t-sql/statements/grant-transact-sql

Lesson: Object level security

- Securing tables and views
- Column-level security
- Row-level security (RLS)
- Securing stored procedures
- Securing user-defined functions (UDFs)
- Ownership Chains

Securing tables and views

- Several object permissions apply to tables and views:
  - SELECT
  - INSERT, UPDATE, DELETE

```
GRANT SELECT ON OBJECT::Production.Products TO myGroup

GRANT SELECT ON Production.Products TO Stan
```

Demo Grant table permissions

Column-level security

- Can assign permissions at column level
  - SELECT
  - UPDATE
- Can assign permissions for multiple columns in one GRANT or DENY statement
- Column-level GRANT statements override table-level DENY statements

```
GRANT SELECT
      ON Marketing.Salesperson (SalespersonID, Name)
      TO User1

DENY SELECT ON
      Marketing.Salesperson (Salary, Bonus)
      TO User2
```

Row-level security (RLS)

- Control access to rows in a table, for example:
  - Salesperson accessing customer data in their region
  - Employee accessing data relevant to their department
- Advantages:
  - Logic held with data, reduces risk of errors and simplifies security
- Implement by adding a security predicate defined as an In-Line Table-Valued Function (IL TVF)
- Introduced in SQL Server 2016

Demo Row level security

Row-Level Security
https://docs.microsoft.com/en-us/sql/relational-databases/security/row-level-security

Introduction to Row-Level Security in SQL Server
https://www.sqlshack.com/introduction-to-row-level-security-in-sql-server/

Securing stored procedures

- EXECUTE: enables users to call stored procedures
- ALTER: enables users to modify stored procedures
- VIEW DEFINITION: enables users to access the code definition

```
GRANT EXECUTE
ON GetCustomerDetails
TO User1
```

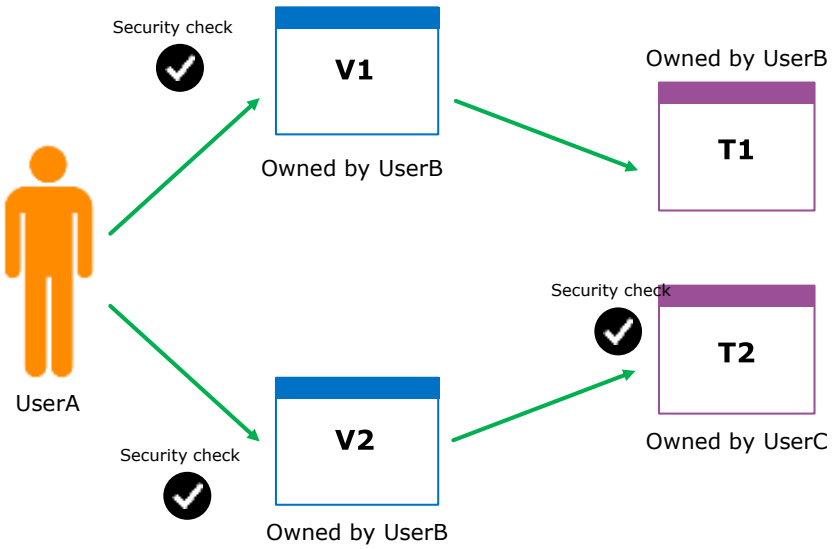
Demo Grant module permissions

Securing user-defined functions (UDFs)

- Scalar UDFs require EXECUTE permissions
- TVFs require SELECT permissions

```
GRANT EXECUTE
ON FormatPhoneNumber
TO public
```

Ownership Chains



Understanding SQL Server Ownership Chaining
https://www.mssqltips.com/sqlservertip/6394/understanding-sql-server-ownership-chaining/

Packaging Permissions in Stored Procedures
https://www.sommarskog.se/grantperm.html

Lesson: Schema level security

- Separation of users and schema
- Object name resolution
- Granting permissions at schema level

Separation of users and schema

- A schema is a container for database objects
  - Not something physical
  - The schema name is part of the object's name space
  - Similar to a folder in the file system
- You can see schemas in sys.schemas
  - Under the "Security" folder in SSMS
- A user has a default schema
- Built-in schemas includes
  - Dbo, guest, sys, INFORMATION_SCHEMA

Ownership and user-schema separation in SQL Server
https://docs.microsoft.com/en-us/sql/relational-databases/security/authentication-access/ownership-and-user-schema-separation

Create a database schema
https://docs.microsoft.com/en-us/sql/relational-databases/security/authentication-access/create-a-database-schema

CREATE SCHEMA (Transact-SQL)
https://docs.microsoft.com/en-us/sql/t-sql/statements/create-schema-transact-sql

Object name resolution

- Objects in different schemas can have the same name
- SQL Server resolves names without schema qualification by
  - 1: Look in the user's default schema
  - 2. Looking in the dbo schema
- Avoid ambiguity by using two-part names

Granting permissions at schema level

- Apply permissions to all relevant objects in the schema
- Can simplify permission management

```
GRANT EXECUTE ON SCHEMA::Sales TO Bob

GRANT SELECT ON SCHEMA::Sales TO Liza
```

Demo Permissions schema level

Lab 3: Granting permissions

- Ex 1: Grant and deny permissions at object and schema level
- Ex 2: Test above permissions

Estimated Time: 30 minutes

Module 4

Auditing and encryption

Module Overview

- Options for auditing
- The Server Audit feature
- Options for data encryption and masking

Lesson: Options for auditing

- Auditing data access
- Common Criteria compliance
- Auditing with triggers

Discussion: Auditing data access

- Why is auditing required?
- What methods have you used for auditing?
- What are the limitations of the methods you have used?
- Which standards that require auditing does your organization need to comply with?

Various techniques to audit SQL Server databases
https://www.sqlshack.com/various-techniques-to-audit-sql-server-databases/

SSMS, security page configuration

- Common Criteria Compliance:
  - Ratified as an international standard in 1999
  - Supersedes C2 rating
  - ISO standard 15408
  - Enable common criteria compliance enabled configuration option by using sp_configure:
    - Residual Information Protection (RIP)
    - Ability to view login statistics
    - Column GRANT does not override DENY
    - Additional script must be run to comply with Common Criteria Evaluation Assurance Level 4+ (EAL4+)
- Login auditing
  - On by default for failed logins

Common Criteria Compliance Enabled Server Configuration
https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/common-criteria-compliance-enabled-server-configuration-option

C2 audit mode Server Configuration Option
https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/c2-audit-mode-server-configuration-option

Auditing with triggers

- Triggers can provide part of an auditing solution:
  - DML triggers for data modification
  - Logon triggers for tracking logins
  - DDL triggers for schema modification
- Watch out for:
  - Performance impact (DML triggers)
  - A sufficiently powerful users can disable triggers
  - No DML trigger for SELECT statement
    - Only INSERT, UPDATE and DELETE
  - Complexity regarding DML trigger firing order

Demo Audit with triggers

Create a Simple SQL Server Trigger to Build an Audit Trail
https://www.mssqltips.com/sqlservertip/4055/create-a-simple-sql-server-trigger-to-build-an-audit-trail/

Lesson: The Server Audit feature

- Introduction to SQL Server Audit
- Defining a Server Audit
- Audit Actions and Action Groups
- Creating Server Audit Specifications
- Creating Database Audit Specifications
- Audit-Related Dynamic Management Views and System Views
- Custom Audit Events
- Retrieving Audit Data
- Working with the Audit Record Structure
- Potential SQL Server Audit Issues

Introduction to SQL Server Audit

- Required Enterprise Edition prior to 2012
- 2012 and 2014: Enterprise Edition
  - Limited to Server Audit specifications if lower than Enterprise Edition
- 2016 and higher: all editions full functionality
- Terminology:
  - Server Audit
  - Server Audit Specification
  - Database Audit Specification
  - Actions and Action Groups
  - Target

Demo Server audit

SQL Server Audit (Database Engine)
https://docs.microsoft.com/en-us/sql/relational-databases/security/auditing/sql-server-audit-database-engine

Defining a Server Audit

- Specify:
  - Target
  - Queue delay
  - Action on failure
- Set STATE = ON to enable

```
CREATE SERVER AUDIT mySecurityAudit
TO FILE
( FILEPATH = '\\servername\sharename\', MAXSIZE = 500 MB
, MAX_ROLLOVER_FILES = 10 ,RESERVE_DISK_SPACE = OFF)
WITH
(QUEUE_DELAY = 1000, ON_FAILURE = FAIL_OPERATION)
GO

ALTER SERVER AUDIT SecurityAudit
WITH (STATE = ON)
```

Create a Server Audit and Server Audit Specification
https://docs.microsoft.com/en-us/sql/relational-databases/security/auditing/create-a-server-audit-and-server-audit-specification

CREATE SERVER AUDIT (Transact-SQL)
https://docs.microsoft.com/en-us/sql/t-sql/statements/create-server-audit-transact-sql

Audit Actions and Action Groups

- Action Groups
  - Server level
  - Database level
  - Audit level
- Actions
  - Database level

- Actions and action groups are linked to an audit with an audit specification

SQL Server Audit Action Groups and Actions
https://docs.microsoft.com/en-us/sql/relational-databases/security/auditing/sql-server-audit-action-groups-and-actions

Creating Server Audit Specifications

- Specify:
  - Audit
  - Action groups to be included
  - State

```
CREATE SERVER AUDIT SPECIFICATION AuditLogins
FOR SERVER AUDIT mySecurityAudit
ADD (FAILED_LOGIN_GROUP),
ADD (SUCCESSFUL_LOGIN_GROUP)
WITH (STATE = ON)
```

Create a Server Audit and Server Audit Specification
https://docs.microsoft.com/en-us/sql/relational-databases/security/auditing/create-a-server-audit-and-server-audit-specification

CREATE SERVER AUDIT SPECIFICATION (Transact-SQL)
https://docs.microsoft.com/en-us/sql/t-sql/statements/create-server-audit-specification-transact-sql

Creating Database Audit Specifications

- Specify:
  - Audit
  - Action Groups
  - Actions on specific securable objects
    - May be filtered by specific database principals
  - State

```
CREATE DATABASE AUDIT SPECIFICATION DBSecurity
FOR SERVER AUDIT mySecurityAudit
ADD (DATABASE_PRINCIPAL_CHANGE_GROUP),
ADD (SELECT ON SCHEMA::HumanResources BY public)
WITH (STATE = ON)
```

Create a server audit and database audit specification
https://docs.microsoft.com/en-us/sql/relational-databases/security/auditing/create-a-server-audit-and-database-audit-specification

CREATE DATABASE AUDIT SPECIFICATION (Transact-SQL)
https://docs.microsoft.com/en-us/sql/t-sql/statements/create-database-audit-specification-transact-sql

Audit-Related Dynamic Management Views and System Views

- DMVs
  - sys.dm_audit_actions
  - sys.dm_audit_class_type_map
  - sys.dm_server_audit_status

- System Views
  - sys.server_audits
  - sys.server_file_audits
  - sys.server_audit_specifications
  - sys.server_audit_specifications_details
  - sys.database_audit_specifications
  - sys.audit_database_specification_details

Custom Audit Events

- Allows you to create custom audit entries:
  - Add USER_DEFINED_AUDIT_GROUP to an audit specification
  - Call sp_audit_write from Transact-SQL code

```
CREATE TRIGGER HR.CheckBonus ON HR.EmployeeBonus
AFTER INSERT, UPDATE
AS
IF EXISTS (SELECT * FROM inserted WHERE bonus > 1000)
BEGIN
    EXEC sp_audit_write @user_defined_event_id = 12,
                        @succeeded = 1,
                        @user_defined_information = N'Employee bonus over $1000'
END
```

sp_audit_write (Transact-SQL)
https://docs.microsoft.com/en-us/sql/relational-databases/system-stored-procedures/sp-audit-write-transact-sql

Retrieving Audit Data

- Event log targets:
  - Use Event Viewer to view Windows event logs
- Binary file targets:
  - Retrieve file-based audits by using the sys.fn_get_audit_file function

```
SELECT *
FROM sys.fn_get_audit_file('X:\AuditFiles\*',default,default)
```

SQL Server Audit Records
https://docs.microsoft.com/en-us/sql/relational-databases/security/auditing/sql-server-audit-records

Working with the Audit Record Structure

- Work with the results of sys.fn_get_audit_file as with any other result set
- Large audit records
  - To comply with Windows event log rules, values for character fields with greater than 4,000 characters are split into multiple audit records
  - sequence_number column indicates the sequence needed to join split records together

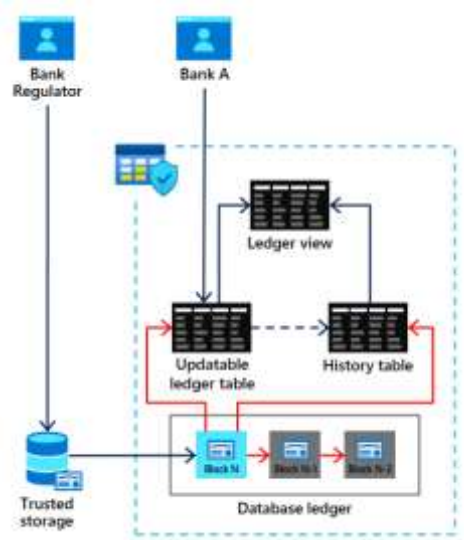
sys.fn_get_audit_file (Transact-SQL)
https://docs.microsoft.com/en-us/sql/relational-databases/system-functions/sys-fn-get-audit-file-transact-sql

Potential SQL Server Audit Issues

- Enable and disable auditing
  - Change the STATE property to ON or OFF to enable or disable server audits and audit specifications
- Considerations for SQL Server Audit:
  - Performance impact of audit writes
  - If audit configuration prevents the instance from starting, use the -f switch
  - If a database is restored to an instance that does not support database audits, the audit is ignored

Ledger (SQL Server 2022)

- Keep a ledger of modifications
- Protected by blockchain
  - Makes it "tamper-proof"
- Two types of tables
  - Ledger tables, modifiable
    - You can see modifications, like an audit
  - Append-only table; keep a log that cannot be modified
- Views added to the database



Ledger documentation
https://learn.microsoft.com/en-us/sql/relational-databases/security/ledger/ledger-landing-sql-server

SQL Ledger: Protecting Data in Azure SQL
https://www.sqlservercentral.com/articles/sql-ledger-protecting-data-in-azure-sql

Ledger (Continued)

- Digest can be seen as a checksum of the database
- Can be used to verify that no tampering has been made
- Generate
  - Manually using sp_verify_database_ledger
  - Automatically to Azure storage account
    - Database scoped configuration
    - Every 30 seconds
- Verification
  - Scans ledger and history tables
  - Recomputes hashes
  - Compares to digest
  - Requires ALLOW_SNAPSHOT_ISOLATION

Ledger documentation
https://learn.microsoft.com/en-us/sql/relational-databases/security/ledger/ledger-landing-sql-server

SQL Ledger: Protecting Data in Azure SQL
https://www.sqlservercentral.com/articles/sql-ledger-protecting-data-in-azure-sql

Lesson: Options for data encryption and masking

- Transparent Data Encryption, TDE
- Moving, copying or restoring an encrypted database
- Extensible Key Management
- Always Encrypted
- Dynamic Data Masking

Transparent Data Encryption, TDE

- Required Enterprise Edition prior to 2019
- Keys:
  - Service master key
  - Database master key
  - Server certificate
  - Database encryption key
- To enable TDE:
  1. Create a DMK
  2. Create a server certificate
  3. Create a DEK
  4. Encrypt the database

Transparent data encryption (TDE)
https://docs.microsoft.com/en-us/sql/relational-databases/security/encryption/transparent-data-encryption

Encryption Hierarchy
https://docs.microsoft.com/en-us/sql/relational-databases/security/encryption/encryption-hierarchy

Free Stuff, including e-book on implementing TDE
https://sqldownunder.com/free-stuff

Moving, copying or restoring an encrypted database

1. Detach the source database
2. Copy/move database files
3. Create new SMK in the master database of the target server
4. Generate a new server certificate from a backup of the server certificate on the source server, and its associated private key
5. Attach the database

Extensible Key Management

- EKM enables encryption keys to be stored securely in third-party hardware security modules, or external EKM providers
  - For instance Azure Key Vault
- Requires additional SQL Server configuration:
  - The EKM provider enabled option must be on
  - Credentials must be created to enable SQL Server to access keys in the EKM provider

Extensible Key Management (EKM)
https://docs.microsoft.com/en-us/sql/relational-databases/security/encryption/extensible-key-management-ekm

Extensible Key Management Using Azure Key Vault (SQL Server)
https://docs.microsoft.com/en-us/sql/relational-databases/security/encryption/extensible-key-management-using-azure-key-vault-sql-server

Always Encrypted

- Typical Always Encrypted Use Cases
  - Protect sensitive data from access by DBAs
- Encryption Types
  - Deterministic
  - Randomized
- Always Encrypted Keys
  - Column master key
  - Column encryption key
- Always Encrypted Driver
  - Transparent to application, more or less

Demo Always encrypted

Always Encrypted
https://docs.microsoft.com/en-us/sql/relational-databases/security/encryption/always-encrypted-database-engine

Overview of key management for Always Encrypted
https://docs.microsoft.com/en-us/sql/relational-databases/security/encryption/overview-of-key-management-for-always-encrypted

Always Encrypted with secure enclaves
https://docs.microsoft.com/en-us/sql/relational-databases/security/encryption/always-encrypted-enclaves

Encryption by DBMSs
https://blog.cryptographyengineering.com/2019/02/11/attack-of-the-week-searchable-encryption-and-the-ever-expanding-leakage-function/

Dynamic Data Masking

- Mask formats:
  - Default
  - Email
  - Custom String
  - Random
- Viewing masked data:
  - SELECT permission will see masked data
  - UNMASK permission will see unmasked data
    - Only at database level unless Azure SQL/SQL Server 2022
- Restrictions
  - Always Encrypted
  - FILESTREAM
  - COLUMN_SET
  - Calculated columns

Demo Dynamic data masking

Dynamic Data Masking
https://docs.microsoft.com/en-us/sql/relational-databases/security/dynamic-data-masking

Optional Lab 4: Auditing and encryption

- Ex 1. Implement Server Audit
- Ex 2. Implement TDE
- Ex 3. Implement Always Encrypted

- The exercises are independent of each other
  - You will not likely manage to do all three
  - Do whichever you want to do, in whatever order you feel like

Estimated Time: 60 minutes

Module 5

Backup strategies and recovery models

Module Overview

- Backup and restore strategies
- Understanding and managing the transaction log
- Backup types

Lesson: Backup and restore strategies

- Determining an Appropriate Backup Strategy
- Types of Backup Media
- SQL Server Backup with Azure Blob Storage

Determining an Appropriate Backup Strategy

- Determine safety levels:
  - How long can recovery take? (RTO)
  - How much data is it acceptable to lose? (RPO)
  - Is it possible to recover the data from other sources?
- Backup strategy should map to requirements:
  - Types and frequency of backups
  - Backup media to use
  - Retention period for backups and for media
  - Backup testing policy
    - Regular disaster recovery practices
    - After each backup was performed (RESTORE VERIFYONLY)
    - Verify the environment so that nothing shady is going on
    - Practice the restore GUI and commands

The Accidental DBA (Day 6 of 30): Backups: Understanding RTO and RPO
https://www.sqlskills.com/blogs/paul/the-accidental-dba-day-6-of-30-backups-understanding-rto-and-rpo/

Why RPO and RTO Are Actually Performance Metrics Too
https://www.brentozar.com/archive/2015/09/why-rpo-and-rto-are-actually-performance-metrics-too/

Types of Backup Media

- Disk
- URL
- TAPE
  - Hardly ever used
- We can address the device:
  - Directly, as in:
    - DISK = 'F:\Sqlbackups\mybackup.bak'
  - Indirectly using predefined backup devices
    - Acts like pointers to the physical media

Demo Backup devices

Media Sets, Media Families, and Backup Sets (SQL Server)
https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/media-sets-media-families-and-backup-sets-sql-server

Backup to URL

- Two options
  - Azure blob storage
  - AWS S3 compatible devices (2022)
- Potential benefits:
  - Unlimited storage
  - Offsite backup solution without the need for tapes and transport
  - No backup hardware to purchase or maintain
  - Offsite backups available instantly

Quickstart: SQL backup and restore to Azure Blob Storage
https://docs.microsoft.com/en-us/sql/relational-databases/tutorial-sql-server-backup-and-restore-to-azure-blob-storage-service

SQL Server Backup and Restore with Azure Blob Storage
https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/sql-server-backup-and-restore-with-microsoft-azure-blob-storage-service

SQL Server backup to URL for Microsoft Azure Blob Storage
https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/sql-server-backup-to-url

SQL Server backup and restore with S3-compatible object storage
https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/sql-server-backup-and-restore-with-s3-compatible-object-storage

Lesson: Understanding and managing the transaction log

- Overview of SQL Server Transaction Logs
- Transaction Log File Structure
- Virtual Log Files (VLFs)
- Working with Recovery Models
- Capacity Planning for Transaction Logs
- Working with Checkpoint Options

Overview of SQL Server Transaction Logs

Transaction logs provide a history of actions executed by a database management system to guarantee atomicity and durability of transactions:

1. Data modification is sent by the application
2. Data pages are located in or read into the buffer cache and then modified
3. Modification is recorded in the transaction log on disk
4. Later, checkpoint writes dirty pages to data files

Demo Automatic recovery

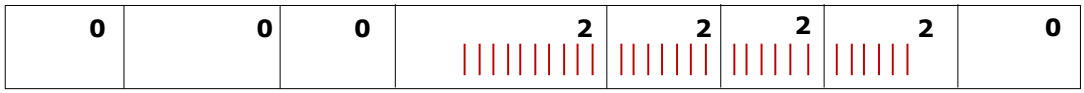
The Transaction Log (SQL Server)
https://docs.microsoft.com/en-us/sql/relational-databases/logs/the-transaction-log-sql-server

Transaction Log File Structure

- Sufficient information is logged to be able to:
  - Roll back transactions if requested
  - Recover the database in case of failure
  - The transaction log is also used by other features such as
    - Always On Availability Groups
    - Transactional Replication
- Write-Ahead Logging is used to create log entries:
  - Transaction logs are written in chronological order in a circular way through out the log file(s)
  - Truncation of transaction log is based on the recovery model

Virtual Log Files (VLFs)

- An ldf file is divided into several Virtual Log Files, VLFs
  - The transaction log cannot be shrunk past the highest used VLF
  - DBCC LOGINFO show us the VLFs and if each VLF is empty/free or in in-use
    - sys.dm_db_log_info is a more modern and supported way to get the same information. Introduced in SQL Server 2017.
    - Status = 0 means free (available to re-use)
    - Status = 2 mean in use
    - If all VLFs are in use then the log file has to grow as log records are produced
  - Many VLFs (several hundreds) causes some operations to become slow
    - Recovery
    - Rollback
    - …



Large transaction log file
https://karaszi.com/large-transaction-log-file

Why you want to be restrictive with shrink of database files
https://karaszi.com/why-you-want-to-be-restrictive-with-shrink-of-database-files

Working with Recovery Models

- Simple – If we **don't** perform log backups
  - Does not permit or require log backups
  - Automatically truncates the log to keep space requirements small
- Full – If we **do** perform log backups
  - Requires log backups (or log will never be emptied)
  - Allow you to avoid data loss due to a damaged or missing data file
  - Permits recovery to a specified point in time
- Bulk logged
  - Requires log backups (or log will never be emptied)
  - Allow SELECT INTO, index management and bulk loading go in minimally logged mode, as in simple recovery mode
  - Such log backup will include data pages a as well

Demo Empty log

Recovery Models (SQL Server)
https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/recovery-models-sql-server

View or Change the Recovery Model of a Database (SQL Server)
https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/view-or-change-the-recovery-model-of-a-database-sql-server

Capacity Planning for Transaction Logs

- Capacity needs are based on several factors:
  - Recovery model used for the database
  - Transaction log backup frequency in full and bulk logged recovery models
  - Number and size of transactions in the database
- Examine log behavior during predeployment testing

Manage the size of the transaction log file
https://docs.microsoft.com/en-us/sql/relational-databases/logs/manage-the-size-of-the-transaction-log-file

Working with Checkpoint Options

- Types of checkpoint operations:
  - Automatic
  - Indirect
  - Manual
  - Internal
- CHECKPOINT statement configures the target recovery duration

Database Checkpoints (SQL Server)
https://docs.microsoft.com/en-us/sql/relational-databases/logs/database-checkpoints-sql-server

CHECKPOINT (Transact-SQL)
https://docs.microsoft.com/en-us/sql/t-sql/language-elements/checkpoint-transact-sql

Lesson: Backup types

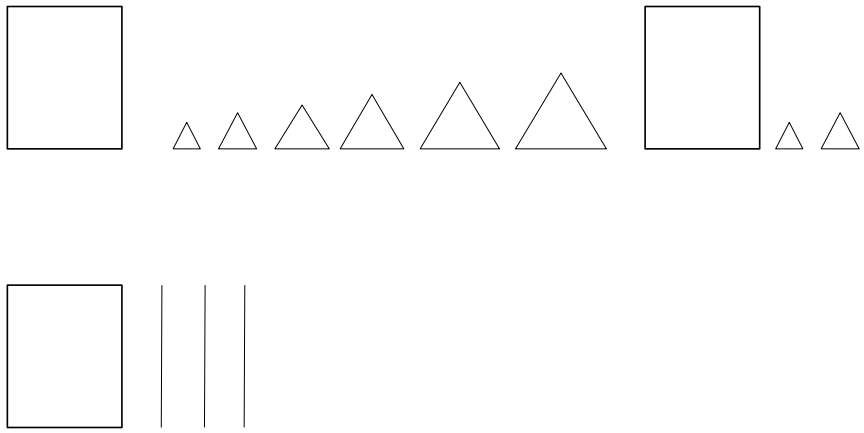
- Overview of Microsoft SQL Server Backup Types
- Full Database Backup Strategies
- Transaction Log Backup Strategies
- Differential Backup Strategies
- Partial and Filegroup Backup Strategies

Overview of backup types

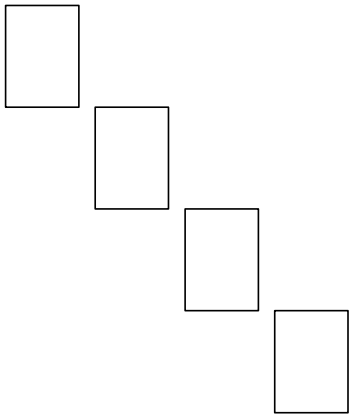
Backup type	Description
Full	All extents from all data files and the active part of the transaction log
Differential	The extents that have changed since the last full backup
Partial	The primary filegroup, every read/write filegroup, and any specified read-only filegroups
Transaction Log	Any database changes recorded in the log files
Tail-log	Log backup taken just before a restore operation. “The last log backup before restore".
File/File Group	Specified files or filegroups
Copy Only	A full or log (without affecting differential or log backups)

Backup Overview (SQL Server), https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/backup-overview-sql-server

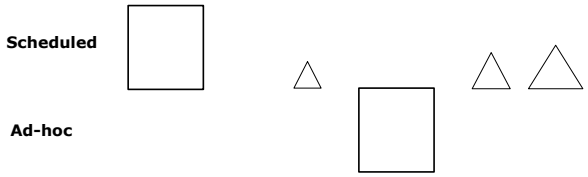
Backup Types, supporting picture 1: Full, diff and log backups

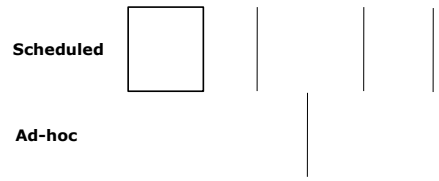


Backup Types, supporting picture 2: File and filegroup backups

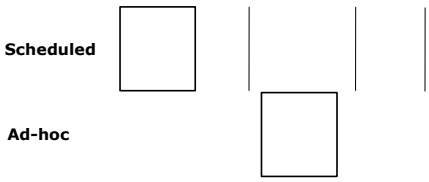


Backup Types, supporting picture 3: Copy only discussion





Backup Types, supporting picture 4: Is copy only required here?



Full Database Backup Strategies

A full database backup strategy:

- Involves taking a full backup of the primary data file
- In simple mode, the database can only be recovered to the point that the last backup was taken
- Can be an optimal solution where data is modified infrequently or is used in a test environment

Full Database Backups (SQL Server)
https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/full-database-backups-sql-server

Transaction Log Backup Strategies

A database and transaction log backup strategy:

- Involves at least full and transaction log backups
- Enables point-in-time recovery
- Allows the database to be fully restored in the case of data file loss

Transaction Log Backups (SQL Server)
https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/transaction-log-backups-sql-server

Differential Backup Strategies

A differential backup strategy:

- Involves performing full and differential database backups
- Includes differential backups with only changed data
- Is useful if only a subset of a database is modified more frequently than the rest of the database
- Use modified_extent_page_count to determine whether to perform a differential or full database backup

Differential Backups (SQL Server)
https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/differential-backups-sql-server

Partial and Filegroup Backup Strategies

Partial and filegroup backups:

- Faster backup and restore for very large databases
- Can be complex to set up and manage
- If possible consider using snapshot backup at storage level instead

Partial Backups (SQL Server)
https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/partial-backups-sql-server

Full File Backups (SQL Server)
https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/full-file-backups-sql-server

No lab for this module

Module 6

Performing backup in SQL Server

Module Overview

- The backup command and the GUI in SSMS
- Managing backups
- Backup options

Lesson: The backup command and the GUI in SSMS

- Introduction to SQL Server Backups
- Media Sets and Backup Sets
- Performing Database Backups
- Performing Transaction Log Backups
- Performing Partial and Filegroup Backups

Introduction to SQL Server Backups

- BACKUP Transact-SQL statement

```
BACKUP { DATABASE | LOG } <database_name>
TO <backup_device>, [, …n]
WITH <general_options>
```

BACKUP (Transact-SQL)
https://docs.microsoft.com/en-us/sql/t-sql/statements/backup-transact-sql

Media Sets and Backup Sets

- Media sets consist of one or more disk backup devices
  - Data can be striped across multiple devices
- A backup set represents one backup of any type
- Backup sets are written to media sets
  - A media set can contain multiple backup sets
- Backup devices and media sets are created at first use:
  - Use FORMAT to overwrite an existing media set
  - Use INIT to overwrite existing backup sets in a media set
  - Use the FORMAT option with caution

Media Sets, Media Families, and Backup Sets (SQL Server)
https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/media-sets-media-families-and-backup-sets-sql-server

Performing Database Backups

- Full backup:
  - Entire database
  - Active portion of log

```
BACKUP DATABASE AdventureWorks
TO DISK = 'D:\Backups\AW.bak'
WITH INIT
```

- Differential backup:
  - Extents modified since the last full backup
  - Active portion of log file

```
BACKUP DATABASE AdventureWorks
TO DISK = 'D:\Backups\AW.bak'
WITH DIFFERENTIAL, NOINIT
```

BACKUP (Transact-SQL)
https://docs.microsoft.com/en-us/sql/t-sql/statements/backup-transact-sql

Performing Transaction Log Backups

- Backs up only the transaction log
- Backs up the log from the last successfully executed log backup to the current end of the log
- Truncates inactive log part of the transaction log

```
BACKUP LOG AdventureWorks
TO DISK = 'D:\Backups\AW.bak'
WITH NOINIT
```

Note: Database must be in full or bulk-logged recovery model

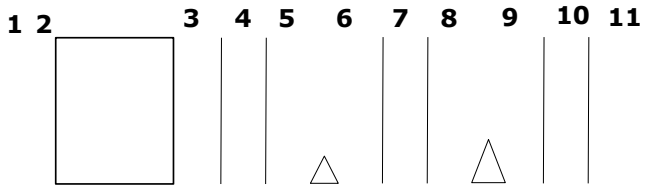
- Perform a log-tail backup before restoring

```
BACKUP LOG AdventureWorks
TO DISK = 'D:\Backups\AW.bak'
WITH NORECOVERY
```

Demo Tail log backup on crashed database

BACKUP (Transact-SQL)
https://docs.microsoft.com/en-us/sql/t-sql/statements/backup-transact-sql

Backups supporting picture



Performing Partial and Filegroup Backups

- Partial backup:
  - Primary filegroup
  - Read/Write filegroups

```
BACKUP DATABASE LargeDB
READ_WRITE_FILEGROUPS
TO DISK = 'D:\Backups\LrgRW.bak'
```

- File or filegroup backup:
  - Specific files or filegroups

```
BACKUP DATABASE LargeDB
FILEGROUP = 'FG2'
TO DISK = 'D:\Backups\LrgFG2.bak'
```

Lesson: Managing backups

- Determining a Retention and Testing Policy for Backups
- Options for backup integrity
- Viewing backup history
- Retrieving Backup Metadata

Determining a Retention and Testing Policy for Backups

- Planning for backup retention must be part of the strategy and form part of the test plan to ensure accuracy
- Several considerations:
  - Combination of backups needed for a database recovery
  - Archival requirements
  - Synchronization with database checks
  - Available secure storage location
  - Hardware required for restoring backups
  - Completeness of backups

Options for backup integrity

- Mirrored media sets:
  - Can mirror a backup set to up to four media sets
  - Mirrors require the same number of backup devices
  - Requires Enterprise Edition
- CHECKSUM backup option:
  - Available for all backup types
  - Generates a checksum over the backup stream
  - Use to verify the backup
- Backup verification:
  - Can use RESTORE VERIFYONLY for backup verification
  - Useful when combined with CHECKSUM option

BACKUP (Transact-SQL)
https://docs.microsoft.com/en-us/sql/t-sql/statements/backup-transact-sql

Viewing backup history

- SQL Server tracks all backup activity in a set of tables in the msdb database

```
SELECTbs.media_set_id, bs.backup_finish_date, bs.type,
 bs.backup_size,
bs.compressed_backup_size,mf.physical_device_name
FROM dbo.backupset AS bs
 INNER JOIN dbo.backupmediafamily AS mf
  ON bs.media_set_id = mf.media_set_id
WHERE database_name = 'AdventureWorks'
ORDER BY backup_finish_date DESC
```

- The Backup and Restore Events report in SSMS displays detailed backup history information

Demo view backup information

Backup and Restore Tables (Transact-SQL)
https://docs.microsoft.com/en-us/sql/relational-databases/system-tables/backup-and-restore-tables-transact-sql

Retrieving Backup Metadata

- RESTORE LABELONLY returns information about the backup media on a specified backup device
- RESTORE HEADERONLY returns all the backup header information for all backup sets on a particular backup device
- RESTORE FILELISTONLY returns a list of data and log files contained in a backup set

RESTORE Statements - LABELONLY (Transact-SQL)
https://docs.microsoft.com/en-us/sql/t-sql/statements/restore-statements-labelonly-transact-sql

RESTORE Statements - HEADERONLY (Transact-SQL)
https://docs.microsoft.com/en-us/sql/t-sql/statements/restore-statements-headeronly-transact-sql

RESTORE Statements - FILELISTONLY (Transact-SQL)
https://docs.microsoft.com/en-us/sql/t-sql/statements/restore-statements-filelistonly-transact-sql

Lesson: Backup options

- Copy-Only Backups
- Compressing Backups
- Encrypting Backups

Copy-Only Backups

- Back up the database without changing the restore order
- Copy-only transaction log back ups do not truncate the log
- Copy-only full database backups do not affect the differential base

```
BACKUP DATABASE AdventureWorks
TO DISK = 'D:\Backups\AWCopy.bak'
WITH COPY_ONLY
```

Copy-Only Backups
https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/copy-only-backups-sql-server

Compressing Backups

- Standard Edition or higher
- Reduces size of backup on device
- Reduces I/O requirements, increases CPU usage
- Increases speed of backup and restore
- Some restrictions:
  - Cannot share media with uncompressed backups
  - Cannot share media with Windows backups

Demo Backup compression

Backup Compression (SQL Server)
https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/backup-compression-sql-server

Encrypting Backups

1. Create a database master key for master
2. Create a certificate or asymmetric key
3. Back up the database, specifying the algorithm and key

```
BACKUP DATABASE AdventureWorks
TO DISK = 'D:\Backups\AW_Encrypt,bak'
WITH FORMAT, INIT,
ENCRYPTION(ALGORITHM=AES_128,
SERVER CERTIFICATE = myBackupCert)
```

Backup Encryption
https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/backup-encryption

Lab 6: Backing Up Databases

- Exercise 1: Setup default backup options
- Exercise 2: Perform different backup types

Estimated Time: 30 minutes

Module 7

Restore

Module Overview

- The restore process
- Restoring from different backup types
- Advanced restore scenarios
- Point-in-time restore and 3rd party backup software

Start backup scripts if required for demos in this module

Lesson: The restore process

- Phases of the Restore Process
- Controlling the recovery process during restore
- Preparations for Restoring Backups

Phases of the Restore Process

- The restore process of a SQL Server database consists of three phases:

Phase	Description
Create database	…if it doesn't exist, with same file layout as the one to restore
Data copy	Copies data to the database files
Redo	Applies committed transactions from restored log entries
Undo	Rolls back transactions that were uncommitted at the recovery point

- Redo and undo are also known as recovery

Restore and Recovery Overview (SQL Server)

https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/restore-and-recovery-overview-sql-server

Controlling the recovery process during restore

- These options controls:
  - A: Recovery after the restore, the UNDO phase (REDO is always performed)
  - B: Accessibility of database after restore
  - C: Ability to restore further backups after the restore
- Only one of below can be specified
- RECOVERY (default)
  - A is performed. B, database is accessible. C: No further restore operations possible.
- NORECOVERY
  - A is not performed. B, database is not accessible. C: Further restore operations are possible.
- STANDBY
  - A is performed. B: Database is accessible, but in a readonly, standby state. C: Further restore operations are possible.

Preparations for Restoring Backups

- Perform a tail-log backup if using full or bulk-logged recovery model
- Identify the backups to restore:
  - Last full, file, or filegroup backup
  - Last differential backup, if exists
  - Log backups, if exists

Investigating your backup devices

- RESTORE HEADEORNLY
  - Lots of information about each backup in a backup device
- RESTORE FILELISTONLY
  - What database files the database had for a backup in a backup device
- Above two give you all information needed to perform the restore operation
- RESTORE LABELONLY
  - Mainly used to see if a backup device is part of a striped backup

```
RESTORE HEADERONLY FROM DISK = 'R:\Adventureworks.bak'

RESTORE FILELISTONLY FROM DISK = 'R:\Adventureworks.bak' WITH FILE = 1

RESTORE LABELONLY FROM DISK = 'R:\Adventureworks.bak'
```

RESTORE Statements - HEADERONLY (Transact-SQL)

https://learn.microsoft.com/en-us/sql/t-sql/statements/restore-statements-headeronly-transact-sql

RESTORE Statements - FILELISTONLY (Transact-SQL)

https://learn.microsoft.com/en-us/sql/t-sql/statements/restore-statements-filelistonly-transact-sql

RESTORE Statements - LABELONLY (Transact-SQL)

https://learn.microsoft.com/en-us/sql/t-sql/statements/restore-statements-labelonly-transact-sql

Lesson: Restoring from different backup types

- Restoring a Full Database Backup
- Restoring a Differential Backup
- Restoring Transaction Log Backups

Restoring a Full Database Backup

- Use SQL Server Management Studio
- Use the RESTORE DATABASE statement:
  - Use WITH REPLACE to overwrite an existing database
  - Use WITH MOVE to relocate the database files

```
RESTORE DATABASE Adventureworks FROM DISK = 'R:\Adventureworks.bak'
```

RESTORE Statements (Transact-SQL)

https://docs.microsoft.com/en-us/sql/t-sql/statements/restore-statements-transact-sql

Restoring a Differential Backup

1. Restore the (most recent) full database backup WITH NORECOVERY or STANDBY
2. Restore the most recent differential backup
   1. WITH RECOVERY if you don't have log backups
   2. WITH NORECOVERY (or STANDBY) if you ***do*** have log backups

```
RESTORE DATABASE Adventureworks FROM DISK = 'R:\Adventureworks.bak'

RESTORE DATABASE Adventureworks FROM DISK = 'R:\AW_Diff.bak'
```

Restoring Transaction Log Backups

- Restore transaction logs by using the RESTORE LOG statement
- Restore the log chain chronologically:
  - Use NORECOVERY (or STANDBY) for all but the last backup
  - Use RECOVERY for the last backup

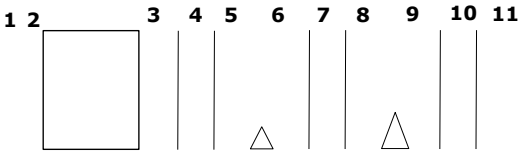
```
RESTORE LOG Adventureworks FROM DISK = 'R:\AW_Log.bak' WITH FILE = 1, NORECOVERY

RESTORE LOG Adventureworks FROM DISK = 'R:\AW_TailLog.bak' WITH RECOVERY
```

Demo Restore

Apply Transaction Log Backups (SQL Server)

https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/apply-transaction-log-backups-sql-server

Supporting picture restore demo



Lesson: Advanced restore scenarios

- Restoring File or Filegroup Backups
- Restoring an Encrypted Backup
- Restoring Data Pages
- Recovering System Databases

Restoring File or Filegroup Backups

Restoring an individual file or filegroup:

1. Create a tail-log backup (if possible)
2. Restore each damaged file or filegroup
3. Restore any differential file backups
4. Restore transaction log backups in sequence
5. Recover the database

Performing a piecemeal restore:

1. Restore read/write filegroups with PARTIAL
2. Restore any differential or log backups and recover the database
3. Restore read-only filegroups

Restore Files and Filegroups over Existing Files (SQL Server)

https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/restore-files-and-filegroups-over-existing-files-sql-server

File Restores (Simple Recovery Model)

https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/file-restores-simple-recovery-model

File Restores (Full Recovery Model)

https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/file-restores-full-recovery-model

Online Restore (SQL Server)

https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/online-restore-sql-server

Piecemeal Restores (SQL Server)

https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/piecemeal-restores-sql-server

Restoring Data Pages

Online Page Restore:

1. Restore pages from a full backup with NORECOVERY
2. Restore latest differential backup with NORECOVERY
3. Restore log backups with NORECOVERY
4. Back up the log
5. Restore the log with RECOVERY

Offline Page Restore:

1. Back up the tail-log with NORECOVERY
2. Restore pages from a full backup with NORECOVERY
3. Restore latest differential backup with NORECOVERY
4. Restore log backups with NORECOVERY
5. Restore the tail-log with RECOVERY

Restore Pages (SQL Server)

https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/restore-pages-sql-server

Restoring an Encrypted Backup

- If the encryption key exists on the server instance:
  - Restore the database as normal
- Otherwise:
  1. Create or restore a database master key for the master database
  2. Create the encryption certificate or key from a backup
  3. Restore the database

Backup Encryption

https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/backup-encryption

Recovering System Databases

System database	Description
master	Should we do backup? Yes Recovery model: simple Restore using single user mode (-m startup switch)
model	Should we do backup? Yes Recovery model: full by default, user configurable Restore like any database, might have to start SQL Server with–T3608 trace flag
msdb	Should we do backup? Yes Recovery model: simple by default, user configurable Restore like any user database
tempdb/resource	No backups can be performed **tempdb** is created during instance startup Restore **resource** using file restore or setup

Backup & restore: system databases (SQL Server)

https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/back-up-and-restore-of-system-databases-sql-server

Restore the master Database (Transact-SQL)

https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/restore-the-master-database-transact-sql

The master database contains information such as:

- Logins
- Linked server configuration
- User-defined error messages (rarely used)
- A combination of all database's sys.database_files (which for tempdb is the template for the tempdb files at startup)
- Sp_configure settings
- Server audit
- Backup devices (created using sp_addumpdevice)
- Endpoints (you can set permissions on network libraries, basically)
- Credentials
- Server-level DDL triggers

Recovering from failure to start SQL Server

- The SQL Server boot process
  - 1. Recover master database, file paths are in the registry
  - 2. Recover model database, file paths are in the master database
  - 3. Create tempdb, as defined in master database
  - Recover the rest of the databases
- SQL Server fails to start if any of 1-3 fails
- Check errorlog file and eventlog for reason
- Try starting SQL server using –f
  - ”Fail safe”
  - If some config option is out of whack
  - If SQL server can’t create tempdb
    - Tempdb creation will be done in default database folder

Rebuild System Databases

https://docs.microsoft.com/en-us/sql/relational-databases/databases/rebuild-system-databases

Performing rebuild of system databases

- If SQL server refuses to start because of problems with system databases
  - And –f doesn't help
- Perform rebuild
  - SETUP /… /…
  - Start SQL Server in single user mode, using –m switch
  - Restore master database
  - Start SQL Server normally
  - Restore msdb
  - Restore model, if required

Rebuild System Databases

https://docs.microsoft.com/en-us/sql/relational-databases/databases/rebuild-system-databases

Lesson: Point-in-time restore and 3rd party backup software

- Overview of Point-in-time restore
- The STOPAT Option
- 3rd party backup software

Overview of point-in-time restore

- Enables recovery of a database up to any arbitrary point in time that is contained in the transaction log backups
- Point in time can be defined by:
  - A datetime value
  - A named transaction
- Database must be in FULL recovery model
  - Or BULK_LOGGED if no minimally logged operations were performed for the last log backup where STOPAT is specified

Restore a SQL Server Database to a Point in Time (Full Recovery Model)

https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/restore-a-sql-server-database-to-a-point-in-time-full-recovery-model

Minimizing data loss when accidents happens

https://karaszi.com/minimizing-data-loss-when-accidents-happens

The STOPAT Option

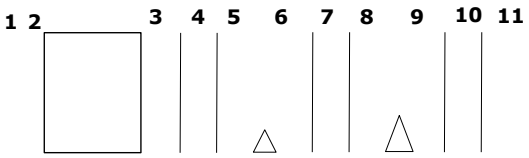
- Provide the STOPAT and WITH RECOVERY options as part of all RESTORE statements in the sequence:
  - No need to know in which transaction log backup the requested point in time resides
  - If the point in time is after the time included in the backup, a warning will be issued and the database will not be recovered after the restore completes
  - If the point in time is before the time included in the backup, the RESTORE statement fails
  - If the point in time provided is within the time frame of the backup, the database is recovered up to that point

Demo Restore with STOPAT

RESTORE Statements (Transact-SQL)

https://docs.microsoft.com/en-us/sql/t-sql/statements/restore-statements-transact-sql

Supporting picture restore demo



Performing a Point-in-time Recovery by Using SQL Server Management Studio

Demo point in time restore using GUI

Restore a SQL Server Database to a Point in Time (Full Recovery Model)

https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/restore-a-sql-server-database-to-a-point-in-time-full-recovery-model

3:rd party backup software

- A 3:rd party backup software might have support to also backup your SQL Server databases
- Two types of SQL Server integration:
- VIRTUAL_DEVICE
  - Same backup commands as we've already discussed
  - Backup data is streamed back to backup software
- Snapshot
  - Snapshots are performed, synchronizing with below services
    - Traditionally uses Volumes Shadow Services and the VDI API
    - SQL Server VSS Writer
      - It is imperative that this is involved
        - Or be prepared to get corrupt database if restoring from such a snapshot
      - Seen by SQL Server as full backup – allow the backup software to do log backups on top of this
      - Make sure it doesn't mess with your diff backups
      - Not online
      - Freezing and resuming I/O messages in errorlog
      - Make sure this is involved when you snap a VM

SQL Server Backup Applications - Volume Shadow Copy Service (VSS) and SQL Writer

https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/sql-server-vss-writer-backup-guide

Virtual device interface (VDI) reference

https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/vdi-reference/reference-virtual-device-interface

SQL Server Backup and Restore in a Veeam environment

https://www.veeam.com/wp-sql-server-backup-with-veeam.html

Other backup enhancements in SQL server 2022

- Hardware backup compression, 2022
  - `ALTER SERVER CONFIGURATION SET HARWARE_OFFLOAD ON (QAT);`
  - BACKUP DATABASE mydb
    TO DISK='F:\SQLBACKUPS\mydb.bak'
    WITH COMPRESSION (ALGORITHM = 'QAT-DEFLATE');
- Expose snapshot backups freeze and thaw in T-SQL
  - Doesn't use VDI/VSS
  - Still need 3:rd party support
    - Easier for 3:rd party vendor to implement
    - Database files are locked even though they are frozen
    - You can start SQL Server with trace flag 3661 to avoid the file system locks
      - Allow you to COPY/XCOPY the database files
      - Unsupported and undocumented
      - No ***not*** use in production

Lab 7: Restoring SQL Server Databases

- Ex 1. Restore of an existing, broken, database
- Ex 2. Restore of a non-existing database
- Ex 3. Take a backup. Do something bad. Perform restore.
- Ex 4. If time permits: add log backups to above exercise

Estimated Time: 60 minutes

Module 8

SQL Server Agent jobs

Module Overview

- SQL Server Agent
- Creating Agent jobs
- Managing Agent jobs
- Multi-server job management (MSX)

Lesson: SQL Server Agent

- Overview of SQL Server Agent

Overview of SQL Server Agent

- A service in Windows
  - Included also in Express Edition
    - But it is not possible to start the service for Express Edition
  - Setup by default set to be manually started
  - Connects to its SQL Server instance at startup
    - Using Windows authentication
    - Verifies that it is sysadmin
- A job can perform a number of tasks
- A schedule can be defined to run one or more jobs
- An alert can be used to respond to system events
- An operator can be notified by jobs or alerts

Demo Agent config

SQL Server Agent
https://docs.microsoft.com/en-us/sql/ssms/agent/sql-server-agent

Lesson: Creating Agent jobs

- Jobs, job types, and job categories
- Job steps
- Scheduling jobs
- Scripting jobs
- Agent tokens

Jobs, job types, and job categories

- Jobs consist of a number of job-steps
- Job step types include:
  - T-SQL
  - CMD (bat, cmd and exe file)
  - PowerShell script
  - Executing an SSIS package
  - SSAS commands and queries
  - Various types of Replication job step types, created when configuring Replication
- A job can be assigned a job category

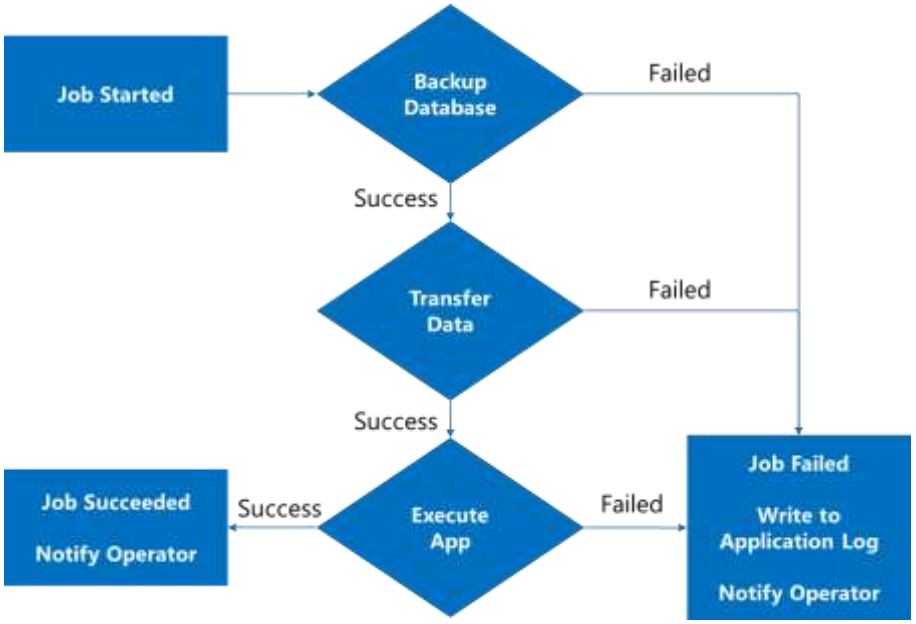
Demo Create agent job

Create a Job
https://docs.microsoft.com/en-us/sql/ssms/agent/create-a-job

Create a Transact-SQL Job Step
https://docs.microsoft.com/en-us/sql/ssms/agent/create-a-transact-sql-job-step

Create a CmdExec Job Step
https://docs.microsoft.com/en-us/sql/ssms/agent/create-a-cmdexec-job-step

Job steps


Job Started
Backup Database
Failed
Success
Transfer Data
Failed
Success
Job Succeeded
Notify Operator
Success
Execute App
Failed
Job Failed
Write to Application Log
Notify Operator

Scheduling jobs

- Type of schedules
  - Recurring
  - One time
  - When SQL Server Agent starts
  - Whenever the CPU is idle
- One job can have multiple schedules
- Multiple jobs can share a schedule

Create a Schedule
https://docs.microsoft.com/en-us/sql/ssms/agent/create-a-schedule

Scripting jobs

- You can script a dialog in SSMS to create a job
- Generate scripts for existing jobs

Agent tokens

- Agent tokens are replaced at run-time
- Examples of what can be picked up
  - Machine name
  - Instance name
  - Job name
  - Start date and time for job
- Can be useful for instance
  - Output file name
  - MSX environment
  - When scripting job and create the same on a different instance
- Example:
  - `PRINT '$(ESCAPE_SQUOTE(SRVR))'`

Use Tokens in Job Steps
https://docs.microsoft.com/en-us/sql/ssms/agent/use-tokens-in-job-steps

Lesson: Managing Agent jobs

- Viewing Job History
- Querying SQL Server Agent-related System Tables and Views
- Troubleshooting Failed Jobs

Viewing Job History

- Information about job history is written to tables in the msdb database
  - Can be viewed via a Transact-SQL query
  - Can be viewed in SSMS by using the Log File Viewer
  - Optionally, the information can be written to jobstep output files
- SSMS also has the Job Activity Monitor that shows the current job activity, and the schedules for active jobs

View the Job History
https://docs.microsoft.com/en-us/sql/ssms/agent/view-the-job-history

Querying SQL Server Agent-related system tables

- Tables for configuration and history information is in the
  - msdb database
  - dbo schema
- The dbo.sysjobhistory table contains a row for each job with a step_id = 0
  - This contains the result for the whole job
  - And also rows for step_id > 0, which are the job step details information
    - Which includes the step output and error messages

Demo View job history

SQL Server Agent Tables (Transact-SQL)
https://learn.microsoft.com/en-us/sql/relational-databases/system-tables/sql-server-agent-tables-transact-sql

dbo.sysjobhistory (Transact-SQL)
https://learn.microsoft.com/en-us/sql/relational-databases/system-tables/dbo-sysjobhistory-transact-sql

Troubleshooting Failed Jobs

- SQL Server Agent status:
  - Is the service account valid?
  - Is the msdb database online?
- Job history:
  - Job outcome identifies the last step to execute
  - Job step outcome identifies why the step failed
- Job execution:
  - Is the job enabled?
  - Is the job scheduled?
  - Is the schedule enabled?

Lesson: Multiserver job management (MSX)

- Multiserver Concepts
- Considerations for multiserver management
- Running Jobs on Target Servers
- Automating Multiserver Maintenance

Multiserver Management Concepts

Master server (MSX)

- Holds central copy of job definitions
- Distributes jobs to target servers
- Receives messages and alerts from target servers

Target servers (TSX)

- Assigned to one master server
- Connect to master server to update schedules of jobs
- Downloads new jobs from master server

Make a Master Server
https://docs.microsoft.com/en-us/sql/ssms/agent/make-a-master-server

Make a Target Server
https://docs.microsoft.com/en-us/sql/ssms/agent/make-a-target-server

Considerations for Multiserver Management

- A job can be defined to execute
  - Locally (default, just as in a non-MSX environment)
  - On any number on target servers
  - But not both above
- A target can have only one master
- You cannot change a target server name while enlisted
- The target server's Agent must use domain account
- The target's Agent service connects to its master SQL Server instance using Windows authentication
- Create certificates for authentication
  - Or set the MsxEncryptChannelOptions registry entry on target server to 0 or 1

Demo Create MSX environment

Set Encryption Options on Target Servers
https://docs.microsoft.com/en-us/sql/ssms/agent/set-encryption-options-on-target-servers

Lab 8: SQL Server Agent jobs

- Ex 1: Create a job
- Ex 2: Run and troubleshoot the job
- Ex 3: Schedule the job
- Ex 4: If time permits: create an MSX environment

Estimated Time: 40 minutes

Module 9

SQL Server Agent security

Module Overview

- SQL Server Agent security
- Credentials in SQL Server
- Agent Proxies

Lesson: Understanding SQL Server Agent Security

- SQL Server Agent security
- SQL Server Agent Roles
- Assigning Security Contexts to SQL Server Agent Job Steps
- Troubleshooting Security in SQL Server Agent

Overview of Security in SQL Server Agent

- SQL Server Agent is a Windows service, so a service account is required
- Agent service account:
  - A dedicated Windows domain account is recommended
  - Local System and Network Service are supported, but are not recommended
  - By default, job steps that interact with the operating system execute under the security context of the service account

Select an account for the SQL Server Agent service
https://docs.microsoft.com/en-us/sql/ssms/agent/select-an-account-for-the-sql-server-agent-service

Set the Service Startup Account for SQL Server Agent (SQL Server Configuration Manager)
https://docs.microsoft.com/en-us/sql/ssms/agent/set-service-startup-account-sql-server-agent-sql-server-configuration-manager

SQL Server Agent Roles

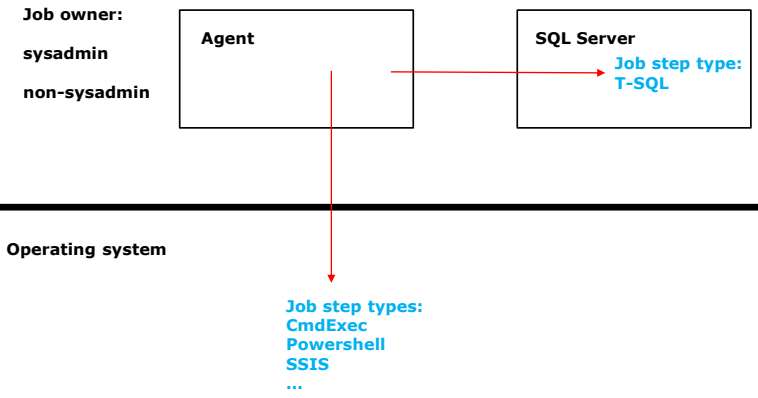
- SQLAgentUserRole
  - Manage own jobs
- SQLAgentReaderRole
  - Manage own jobs
  - View definitions for jobs owned by other users
- SQLAgentOperatorRole
  - Manage own jobs
  - View definitions for jobs owned by other users
  - Enable and disable jobs owned by other users

SQL Server Agent Fixed Database Roles
https://docs.microsoft.com/en-us/sql/ssms/agent/sql-server-agent-fixed-database-roles

Assigning Security Contexts to SQL Server Agent Job Steps

- Transact-SQL job steps
  - Typically executed in the security context of the job owner
  - Members of sysadmin impersonate the SQL Server Agent service account, or can impersonate other database users
- Other job step types
  - Executed by sysadmin using the service account
  - Other logins must use a proxy account
- Proxy accounts
  - Enable a job step to impersonate a Windows identity
  - Are associated with one or more job step subsystems

Who is executing your job step commands?



SQL Server Agent jobs and user contexts
http://sqlblog.karaszi.com/sql-server-agent-jobs-and-user-contexts/

Troubleshooting Security in SQL Server Agent

- Confirm that the job is running
- Check the security account
- Check the job owner
- Check the securable objects accessed that each task accesses
- Check permissions used by each failing step

Lesson: Credentials in SQL Server

- Overview of Credentials
- Configuring Credentials
- Managing Credentials

Overview of Credentials

- Authentication for a resource or system outside the database engine instance
  - Typically Windows user name and password
  - Third-party cryptographic providers are also supported
- Some system credentials are created automatically during SQL Server installation (## prefix)

Credentials (Database Engine)
https://docs.microsoft.com/en-us/sql/relational-databases/security/authentication-access/credentials-database-engine

Configuring Credentials

- Configure credentials by using the CREATE CREDENTIAL command or through SSMS
- Passwords are encrypted by using the master server encryption key
- When the master server encryption key is changed, stored password are automatically re-encrypted for the new key

```
CREATE CREDENTIAL myCredential
  WITH IDENTITY = 'MyDomain\myAccount',
  SECRET = 'myS3cret.'
```

CREATE CREDENTIAL (Transact-SQL)
https://docs.microsoft.com/en-us/sql/t-sql/statements/create-credential-transact-sql

Managing Credentials

- sys.credentials catalog view
- ALTER CREDENTIAL
  - Both the identity and the secret are always updated

```
ALTER CREDENTIAL myCredential
  WITH IDENTITY = 'MyDomain\myAccount',
  SECRET = `myOtherS3cret.'
```

- DROP CREDENTIAL

ALTER CREDENTIAL (Transact-SQL)
https://docs.microsoft.com/en-us/sql/t-sql/statements/alter-credential-transact-sql

Lesson: Agent Proxies

- Overview of Agent Proxies
- Managing Agent Proxies

Overview of Agent Proxies

- Job step subsystems:
  - Proxy accounts can be associated with one or more of the SQL Server Agent job step subsystems
  - A proxy account cannot be used to run a job step using a subsystem it does not have an association with
- Proxy account permissions:
  - Being referenced as a proxy account does not change the permissions of the credential
  - Only members of sysadmin can create and use proxy accounts by default
  - Permission to use proxy accounts can be granted to members of the SQL Server Agent fixed roles

Create a SQL Server Agent Proxy
https://docs.microsoft.com/en-us/sql/ssms/agent/create-a-sql-server-agent-proxy

Managing Agent Proxies

- Proxy account configuration stored in msdb
- Proxy account catalog views:
  - dbo.sysproxies
  - dbo.sysproxylogin
  - dbo.sysproxyloginsubsystem
  - dbo.syssubsystems
- Manage proxy accounts through SSMS, or by using system stored procedures in msdb

Lab 9: SQL Server Agent security

- Ex 1: Create an Agent job with a security issue
- Ex 2: Fix the problem using a Credential and an Agent Proxy

Estimated Time: 40 minutes

Module 10

SQL Server error messages, Database Mail and Agent alerts

Module Overview

- SQL Server error messages
- Database Mail
- Operators and Agent Alerts

Lesson: SQL Server errors messages

- SQL Server error messages
- Error Severity Levels
- Configuring the SQL Server Error Log

SQL Server error messages

- An error that is triggered by the database engine have the following properties:

Property	Description
Error number	Unique identifying number.
Error message	String describing the cause of the error.
Severity	Integer describing the seriousness of the error. This is a bit of a mess so don't expect too much consistency.
State	Tinyint that might give additional information.
Procedure name	Name of the stored procedure or trigger where the error occurred.
Line number	Integer containing the line number at which the error occurred.

Demo SQL Server error messages

Errors and Events Reference (Database Engine)
https://docs.microsoft.com/en-us/sql/relational-databases/errors-events/errors-and-events-reference-database-engine

Understanding Database Engine Errors
https://docs.microsoft.com/en-us/sql/relational-databases/errors-events/understanding-database-engine-errors

Error Severity Levels

- The severity of an error indicates the type of problem that SQL Server encounters.
- You sometimes see classifications in the 11-19 range.
  - This design has been worked on since the late eighties
  - Don't expect too much consistency for such sub-classification

Range	Description
0 to 9	Informational messages
10	Can be either informational message or error message
11 to 19	"Traditional" errors
20-24	SQL Server terminates the session

Database Engine Error Severities
https://docs.microsoft.com/en-us/sql/relational-databases/errors-events/database-engine-error-severities

Configuring the SQL Server Error Log

- SQL Server writes severe errors to the Event Log, Application, and also to the Errorlog file.
- SQL Server creates a new error log file when it starts:
  - No max size by default
    - Configurable
  - Retains six log files by default
    - Configurable
  - You can force a new error log file using sp_cycle_errorlog
    - Use with caution
    - Having a tail of error log information for a few months can be very valuable
      - Questionable if scheduling this for periodical execution is a good idea

Monitoring the Error Logs
https://docs.microsoft.com/en-us/sql/tools/configuration-manager/monitoring-the-error-logs

Analyze SQL Server and Agent errorlogs
https://karaszi.com/analyze-sql-server-and-agent-errorlogs

Lesson 2: Configuring Database Mail

- Overview of Database Mail
- Database Mail Profiles
- Database Mail Security
- Database Mail Logs and Retention
- SQL Server Agent Operator

Overview of Database Mail

- SQL Server acts as an SMTP client, can send mail for instance:
  - As part of a job
  - In response to an alert
  - Using the *sp_send_dbmail* stored procedure
- Set up using
  - The Database Mail Configuration Wizard
  - Or through msdb.dbo.sysmail… stored procedures

Database Mail
https://docs.microsoft.com/en-us/sql/relational-databases/database-mail/database-mail

How to Configure Database Mail
https://www.brentozar.com/blitz/database-mail-configuration/

Database Mail Profiles

- Collection of one or more accounts
- Defines configuration for sending mail
- You can have more than one SMTP account for a profile

Profile	Description
Private	Accessible only to members of **sysadmin** role or those granted permission by members of **sysadmin** role.
Public	Accessible to any member of the **sysadmin** role or the **DatabaseMailUserRole** database role in **msdb**.

- A default private profile takes precedence over the default public profile

Demo Configure database mail

Create a Database Mail Profile
https://docs.microsoft.com/en-us/sql/relational-databases/database-mail/create-a-database-mail-profile

Database Mail Security

- Control access to Database Mail by:
  - SQL Server service account
  - Global enable/disable—disabled by default
  - Membership of msdb.DatabaseMailUserRole
  - Users' access to private profiles
- You can prohibit the use of specific file extensions and set file attachment size limits

Database Mail Logs and Retention

- Database Mail logs event messages at a level:
  - Normal
  - Extended
  - Verbose
- Database Mail stores all email and attachments
- Need to plan a retention policy

```
DECLARE @DeleteOlder datetime

SET @DeleteOlder = DATEADD(MONTH, -1, GETDATE())

EXECUTE msdb.dbo.sysmail_delete_mailitems_sp @sent_before = @DeleteOlder
EXECUTE msdb.dbo.sysmail_delete_log_sp @logged_before = @DeleteOlder
```

Check the Status of E-Mail Messages Sent With Database Mail
https://docs.microsoft.com/en-us/sql/relational-databases/database-mail/check-the-status-of-e-mail-messages-sent-with-database-mail

Database Mail Log and Audits
https://docs.microsoft.com/en-us/sql/relational-databases/database-mail/database-mail-log-and-audits

sysmail_delete_mailitems_sp (Transact-SQL)
https://docs.microsoft.com/en-us/sql/relational-databases/system-stored-procedures/sysmail-delete-mailitems-sp-transact-sql

sysmail_delete_log_sp (Transact-SQL)
https://docs.microsoft.com/en-us/sql/relational-databases/system-stored-procedures/sysmail-delete-log-sp-transact-sql

Do you clean up your Database Mail log tables?
http://sqlblog.karaszi.com/do-you-clean-up-your-database-mail-log-tables/

Sending email from Agent

- Configure the Database Mail profiler to be used by agent
  - SSMS, Properties for Agent
  - `sp_set_sqlagent_properties`

Demo Configure Agent to use profile, create operator and send email from job

Operators
https://docs.microsoft.com/en-us/sql/ssms/agent/operators

Create an Operator
https://docs.microsoft.com/en-us/sql/ssms/agent/create-an-operator

Lesson 3: Operators and Agent Alerts

- Operators
- Agent Alerts
- Creating Agent Alerts
- Configuring Alert Actions

Operators

- An operator has a name and an email address
- The operator can be notified
  - When a job has executed
  - When an Agent Alert occurs

Operators
https://docs.microsoft.com/en-us/sql/ssms/agent/operators

MailAfterJob - send email after Agent job, include ouput files
https://karaszi.com/mailafterjob-send-email-after-agent-job-include-ouput-files

Agent Alerts

- An Alert can be triggered by
  - Errors from database engine, written to the Event Log
  - SQL Server performance counter condition
  - WMI events
- An Alert can
  - Email an Operator
  - Start an Agent job

Alerts
https://docs.microsoft.com/en-us/sql/ssms/agent/alerts

Agent Alerts Management Pack
https://karaszi.com/agent-alerts-management-pack

Creating Agent Alerts

- Create alerts:
  - SSMS
  - sp_add_alert
- Alerts are only triggered if the error message is written to the Windows Event Log
  - Check sys.messages.is_event_logged to determine which errors are automatically logged
  - Modify the error definition (sp_altermessage) to force an error to be logged
    - This setting is respected by most error messages, but not all

Create an Alert Using an Error Number
https://docs.microsoft.com/en-us/sql/ssms/agent/create-an-alert-using-an-error-number

Agent Alerts Management Pack
https://karaszi.com/agent-alerts-management-pack

Configuring Alert Actions

- Actions:
  - Execute a job
  - Notify one or more operator
- When using notifications, you can optionally use tokens to add more detail

Demo Configure Agent alert

Lab 10: Monitoring SQL Server with Alerts and Notifications

- Ex 1: Create and test an Agent Operator
- Ex 2: Investigate an SQL Server error message
- Ex3: Create an Agent Event Alert

Estimated Time: 40 minutes

Module 11

Managing SQL Server Using PowerShell

Lesson: PowerShell and SQL Server

- What is PowerShell
- Getting help in PowerShell
- Getting started with PowerShell
- Modules for managing SQL Server
- SQL Server Management Objects, SMO
- PowerShell providers

What Is PowerShell?

- Combination of command prompt and scripting language
- Can be used to automate Windows tasks and configuration settings
  - Including SQL Server
- Commands, or cmdlets
  - Verb+prefixnoun
  - For instance GetChildItem
- Includes aliases, or shortcuts, such as cd, cls, and dir

Getting help in PowerShell

- Get-Help is a cmdlet that displays help
  - Get-Help <cmdlet>, such as Get-Help Get-Item
- Get-Help Get-Help
  - Above displays help about help
- Wildcards
  - Get-Help *Azure*
- Tab completion
  - Start a cmdlet and then press TAB repeatedly
- Get-verb
  - A cmdlet that displays all PowerShell verbs

Getting started with PowerShell

- PowerShell cmdlets are made available through modules
- Run cmdlets individually or in a script
- Use the Windows PowerShell Console to run cmdlets
- Use the Windows PowerShell ISE to develop and debug scripts
- Denote variables with $—can be strongly or loosely typed

SQL Server PowerShell
https://docs.microsoft.com/en-us/sql/powershell/sql-server-powershell

Modules for managing SQL Server

- SQLPS
  - Installed by SQL Server setup
  - Legacy, no new development
- SQLSERVER
  - Can be installed from https://www.powershellgallery.com/
  - Superset over SQLPS
  - Can be installed on database engine, so both are available
- Dbatools
  - Community project with many CmdLets
  - https://dbatools.io/

SQLSERVER module
https://www.powershellgallery.com/packages/SqlServer

DbaTools
https://dbatools.io/

SQL Server Management Objects, SMO

- Expose SQL Server objects to .NET programming languages
- Classes are grouped into namespaces
  - SMO namespace includes the Database class
  - Other namespaces include SMO.Mail and SMO.Agent
- Use Database class to retrieve and change settings
  - Can be used to create a new database

SQL Server Management Objects (SMO) Programming Guide
https://docs.microsoft.com/en-us/sql/relational-databases/server-management-objects-smo/sql-server-management-objects-smo-programming-guide

Load the SMO Assemblies in Windows PowerShell
https://docs.microsoft.com/en-us/sql/powershell/load-the-smo-assemblies-in-windows-powershell

PowerShell providers

- Provide a way of navigating to a location and getting access to objects and data
- Use Get-psProvider to list all the PowerShell providers
- Use Get-PSDrive to see the PowerShell drives
- Use Set-Location or cd to change to the PowerShell drives

SQL Server PowerShell Provider
https://docs.microsoft.com/en-us/sql/powershell/sql-server-powershell-provider

Dbatools
https://dbatools.io/

No lab for this module

Module 12

Extended Events

Module overview

- Extended Events concepts
- Working with Extended Events

Lesson: Extended Events concepts

- Extended Events, SQL Trace, and SQL Server Profiler
- Packages
- Events
- Predicates
- Actions
- Targets
- Sessions
- Types and Maps

Extended Events, SQL Trace, and SQL Server Profiler

- SQL Trace and Profiler are tools for collecting information about activity on a SQL Server instance
- Extended Events is the successor to SQL Trace
  - SQL Trace and Profiler has been deprecated since SQL Server 2012
  - Extended Events is richer and more powerful SQL Trace
    - But also slightly more complex

SQL SERVER – SQL Profiler vs Extended Events
https://blog.sqlauthority.com/2016/06/22/sql-server-sql-profiler-vs-extended-events/

Extended Events vs SQL Trace
http://andreas-wolter.com/en/extended-events-vs-sql-trace/

Quickstart: Extended events in SQL Server
https://docs.microsoft.com/en-us/sql/relational-databases/extended-events/quick-start-extended-events-in-sql-server

Extended events overview
https://docs.microsoft.com/en-us/sql/relational-databases/extended-events/extended-events

Packages

- A package is a container for other object types:
  - Events
  - Predicates
  - Actions
  - Targets
  - …

SQL Server Extended Events Packages
https://docs.microsoft.com/en-us/sql/relational-databases/extended-events/sql-server-extended-events-packages

Events

- Events are logging points in executable code
  - When an event fires, it indicates that the associated code has been executed
  - Returns data in a fixed schema
  - Events are compatible with Event Tracing for Windows

Actions

- Actions provide supplementary information about an event
  - Each event might be linked to one or more actions
  - Actions are triggered synchronously after an associated event has fired

Actions In Extended Events
https://www.sqlservercentral.com/blogs/actions-in-extended-events

Predicates

- Allow the construction of rules to filter event capture
  - Made up of two subcategories:
    - Comparisons—logical operators (=, <, > and so on)
    - Sources—values that might be used as inputs to comparisons
  - Complex predicates might be constructed. For instance
    - Every x events
    - Every y seconds

Predicate Order Matters in Extended Events
https://sqlperformance.com/2014/06/extended-events/predicate-order-matters

Targets

- Targets collect data from Extended Events sessions
  - A session may write to multiple targets
  - Commonly used targets are
    - Event counter
    - Event file
    - Histogram
    - Ring buffer
      - There's no GUI to "decode" the XML for this in SSMS
      - Avoid this unless you want to parse this your self, using T-SQL for example

Targets for Extended Events in SQL Server
https://docs.microsoft.com/en-us/sql/relational-databases/extended-events/targets-for-extended-events-in-sql-server

Sessions

- A session links events to targets
  - Events may include actions
  - Events may be filtered with predicates
  - Sessions are isolated from one another
  - A session has a state (started or stopped)
  - A session has a buffer to hold event data as it is captured

SQL Server Extended Events Sessions
https://docs.microsoft.com/en-us/sql/relational-databases/extended-events/sql-server-extended-events-sessions

Lesson: Working with Extended Events

- Configuring sessions
- The system_health session
- Best practices

Configuring sessions

- Session configuration options:
  - MAX_MEMORY
  - EVENT_RETENTION_MODE
  - MAX_DISPATCH_LATENCY
  - MAX_EVENT_SIZE
  - MEMORY_PARTITION_MODE
  - STARTUP_STATE
  - TRACK_CAUSALITY

CREATE EVENT SESSION (Transact-SQL)
https://docs.microsoft.com/en-us/sql/t-sql/statements/create-event-session-transact-sql

Tips for getting started with Extended Events
http://sqlblog.karaszi.com/tips-for-getting-started-with-extended-events/

The system_health session

- Created by default on all database engines
  - Starts at instance startup
  - Captures events useful for troubleshooting
  - Ring buffer and file targets

Using sp_HumanEvents

- A more user-friendly way to run a focused XE session for a while
  - And then look at the output
- Get information about things such as
  - Blocking
  - Query performance
  - Compiles
  - Recompiles
  - Wait stats

sp_HumanEvents
https://www.erikdarlingdata.com/sp_humanevents/

sp_HumanEvents: Usage and Guidelines
https://www.erikdarlingdata.com/sp_humanevents-usage-and-guidelines/

Best practices

- Run Extended Events sessions only when you need them
- Use the SSMS GUI to browse available events
- Understand the limitations of the ring buffer target
  - No GUI
  - Limited to 4 MB when you view the data
- Consider the performance impact of collecting query execution plans
- Understand the deadlock graph format

Lab 12: Extended Events

- Ex 1: Capturing SQL commands from an application
- Ex 2: Modify the trace definition
- Ex 2: If time permits: capturing strange events

Estimated Time: 40 minutes

Module 13

Monitoring SQL Server

Module Overview

- Monitoring activity
- Concurrency
- Popular tools

Lesson: Monitoring Activity

- Dynamic Management Views and Functions
- Viewing Activity by Using DMVs
- Activity Monitor in SQL Server Management Studio
- Performance Monitor
- SQL Server Counters
- Data Collection

Dynamic Management Views and Functions

- Commonly referred to as “DMVs”
  - You sometime see DMVs and DMFs or DMOs
- DMVs return server and database state information:
  - Can be used to monitor the health of a server instance, diagnose problems, and tune performance
- Two types:
  - Server-scoped—require VIEW SERVER STATE permission
  - Database-scoped—require VIEW DATABASE STATE permission
- The naming convention categorizes the DMV:
  - sys.dm_<*category*>_something
- Approx 280 DMVs (in SQL Server 2019)
  - Many of them exists in the Azure SQL offerings

System Dynamic Management Views
https://docs.microsoft.com/en-us/sql/relational-databases/system-dynamic-management-views/system-dynamic-management-views

Viewing Activity by Using DMVs

- DMVs return two types of information:
  - Real-time state information
  - Recent historical information:
    - The information is aggregated since the instance was started
- Useful resources, largely using DMVs
  - Glenn Berry's diagnostic queries
  - Brent Ozar's First Responder Kit

Demo DMVs

Glenn Berry's diagnostic queries
https://glennsqlperformance.com/resources/

Bren't Ozar's First Responder Kit
https://www.brentozar.com/first-aid/

Activity Monitor in SSMS

- Activity Monitor displays:
  - Information about SQL Server processes
  - Information about how these processes affect the current instance of SQL Server
- It consists of tabbed panes, each for different information
  - When expanded, panes query the instance for information
  - When collapsed, all queries for that pane stop
  - You can rearrange, reorder, and filter pane contents

Demo Activity Monitor

Explaining Activity Monitor
https://sqlblog.karaszi.com/explaining-activity-monitor/

How to collect performance and system information in SQL Server
https://www.sqlshack.com/how-to-collect-performance-and-system-information-in-sql-server/

Why You Shouldn't Use SQL Server's Activity Monitor
https://www.brentozar.com/archive/2014/10/why-i-dont-use-sql-server-activity-monitor/

Performance Monitor

- Performance Monitor collects performance counters from objects
- SQL Server-specific Performance Monitor counters are added when you install SQL Server
  - The short service name is included in the object name
- Resources:
  - SQL documentation, Monitoring Resource Usage:
    - https://docs.microsoft.com/en-us/sql/relational-databases/performance-monitor/monitor-resource-usage-system-monitor
  - PAL, on Github
    - https://github.com/clinthuffman/PAL

Monitor Resource Usage (Performance Monitor)
https://docs.microsoft.com/en-us/sql/relational-databases/performance-monitor/monitor-resource-usage-system-monitor

15 SQL Server Performance Counters to Monitor In 2022
https://www.sentryone.com/blog/allenwhite/sql-server-performance-counters-to-monitor

SQL Server Perfmon (Performance Monitor) Best Practices
https://www.brentozar.com/archive/2006/12/dba-101-using-perfmon-for-sql-performance-tuning/

PAL, on Github
https://github.com/clinthuffman/PAL

Working with SQL Server Counters

- SQL Server provides objects that can be used to monitor activity
- These have associated counters and statistics
- Counters can have multiple instances of objects if there are multiple instances of associated resources
- Some objects have only one instance
- Statistics can be displayed for any counter
  - Server statistics only displayed when instance is running

Data Collection

- Also known as Management Data Warehouse
- “Dead” technology
- No improvements since SQL Server 2008 (when it was introduced)
- High overhead
- “Nobody” is using it
- Two wizards
  - One that creates the tables
  - Another that creates the agent jobs that collects information
    - Either writes directly to the tables
    - Or store in a file and a separate job writes that information to the tables

Data Collection
https://docs.microsoft.com/en-us/sql/relational-databases/data-collection/data-collection

Lesson: Concurrency

- Transactions, concurrency, locks and blocking
- Isolation levels
- Deadlocks

Transactions, concurrency, locks and blocking

- Definition of a Transaction
  - Atomicity
  - Consistency
  - Isolation
  - Durability
- Lock types
  - Shared (read data)
  - Exclusive (modify data)
  - And others…
- Two sessions (or more) trying to access a resource using non-compatible locks results in blocking
  - Not the same thing as deadlock

Demo Blocking

Transactions in SQL Server for beginners
https://www.sqlshack.com/transactions-in-sql-server-for-beginners/

All about locking in SQL Server
https://www.sqlshack.com/locking-sql-server/

Transaction locking and row versioning guide
https://docs.microsoft.com/en-us/sql/relational-databases/sql-server-transaction-locking-and-row-versioning-guide

Isolation levels

- Specified using SET TRANSCTION ISOLATION LEVEL
  - READ UNCOMMITTED
  - READ COMMITTED
  - REPEATABLE READ
  - SERIALIZABLE
  - SNAPSHOT
    - Requires that ALLOW_SNAPSHOT_ISOLATION is enables for the database
- We can change the implementation of READ COMMITTED
  - From the blocking implementation to row-versioning /snapshot implementation
  - Set using the READ_COMMITTED_SNAPSHOT database option
  - Can make a big difference
  - Make sure that your application supports this before changing the option

SET TRANSACTION ISOLATION LEVEL (Transact-SQL)
https://docs.microsoft.com/en-us/sql/t-sql/statements/set-transaction-isolation-level-transact-sql

Isolation Levels in SQL Server
https://www.brentozar.com/isolation-levels-sql-server/

Deadlocks

- Two or more are blocking each other
- SQL Server selects a victim
  - Rollback
  - Error 1205
- Capture information using Extended Events trace
  - sqlserver.xml_deadlock_report event
- Avoid or fix by
  - Add or remove indexes
  - Change order of doing things
  - Use row versioning instead of blocking behavior
  - …

Demo Deadlock

What are SQL Server deadlocks and how to monitor them
https://www.sqlshack.com/what-are-sql-server-deadlocks-and-how-to-monitor-them/

How to fix SQL Server deadlocks
https://www.red-gate.com/products/dba/sql-monitor/resources/articles/monitor-sql-deadlock

Lesson: Popular tools

- Glenn Berry's diagnostic queries
- First responder kit
- Other 3:rd party tools

Glenn Berry's diagnostic queries

- A bunch of queries
- Gives you an insight into the instance and database(s)
- Go over the script and execute those that are of interest to you
- See Glenn's comments and recommendations
- Take notes as you are doing this
  - Revisit those notes after
  - Take appropriate actions

Glenn berry's diagnostic queries
https://glennsqlperformance.com/resources/

First responder kit

- A handful of stored procedures
- sp_blitz
  - "Free SQL Server Health Check Script“
- sp_BlitzFirst
  - "Helps You Troubleshoot Slow SQL Servers“
- sp_BlitzIndex
  - "SQL Server’s Index Sanity Test“
- sp_BlitzCache
  - "Find Your Worst-Performing Queries“
- sp_BlitzWho
  - “Tells you what’s really happening“
- sp_BlitzQueryStore
  - Look for periods with high activity and expensive queries

First responder kit download
https://www.brentozar.com/first-aid/

Documentation
https://github.com/BrentOzarULTD/SQL-Server-First-Responder-Kit/tree/main#sp_blitz-overall-health-check

List of blog posts, including some that give tips on using sp_Blitz procedures for various purposes
https://www.brentozar.com/archive/2022/10/erik-darlings-month-of-free-tools-training/

Other 3:rd party tools

- sp_whoisactive
- sp_PressureDetector
  - Diagnose CPU or memory pressure
- sp_HumanEvents
  - Create XE trace for you, based on your interest
  - Run the trace for a while
  - Show you the result

sp_whoisactive Downloads
http://whoisactive.com/downloads/

sp_whoisactive Documentation
https://whoisactive.com/docs/

5 common SQL Server Problems to Troubleshoot with sp_WhoIsActive
https://straightpathsql.com/archives/2023/01/5-common-sql-server-problems-to-troubleshoot-with-sp_whoisactive/

sp_PressureDetector
https://erikdarlingdata.com/sp_pressuredetector/

sp_HumanEvents
https://erikdarlingdata.com/sp_humanevents/

sp_HumanEvents: Usage and Guidelines
https://erikdarlingdata.com/sp_humanevents-usage-and-guidelines/

Lab 13: Monitoring SQL Server

- Ex 1: Monitor and resolve a blocking situation
- Ex 2: Use Glenn Berry's diagnostic queries to diagnose your instance
- Ex 3: Use sp_blitz to check for best practices on your instance

- The exercises are independent of each other
  - Do whichever you want to do, in whatever order you feel like and as time permits

Estimated Time: 30 minutes

Module 14

Troubleshooting

Module Overview

- Resolving service-related issues
- Resolving connectivity and login issues

Lesson: Resolving Service-Related Issues

- Troubleshooting Service-Related Issues
- SQL Server Error Log
- Windows Event Logs

Troubleshooting Service-Related Issues

- Check Windows and SQL error logs
- If SQL Server can be started but not accessed:
  - Check for network-related issues
  - Try to access SQL Server via the DAC
    - ADMIN:servername in SSMS (cannot connect Object Explorer, though)
    - /A in SQLCMD
- If SQL Server doesn't start:
  - Check the Windows system log
  - Check master and model databases for corruption
  - Check that the paths to tempdb files are accessible
  - Try to start the service from the command prompt
  - Try start SQL Server using the –f switch

Demo DAC and failed SQL Server startup

SQL Server Error Log

- Contains a record of critical errors and important events
- By default, the current log, plus copies of the six most recent log files, are retained
- View with Log File Viewer or by using a text editor such as Notepad
  - It can be large, so having an editor that reads segments into memory can be a good idea.
- Review all the available log files; a problem may not have started in the current logging period

Monitoring the Error Logs
https://docs.microsoft.com/en-us/sql/tools/configuration-manager/monitoring-the-error-logs

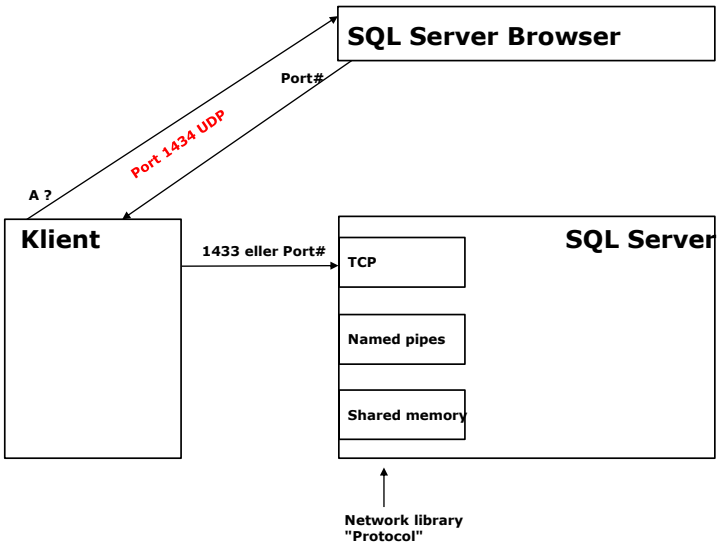
Windows Event Logs

- System log to review Windows-related information
- Application log for application-related messages
- SQL Server writes to the Application log.

Lesson: Resolving Connectivity and Login Issues

- Communication architecture
- Troubleshooting Connectivity Issues
- Troubleshooting Login Failures
- Useful resources

Communication architecture



Network Protocols and Network Libraries
https://docs.microsoft.com/en-us/sql/sql-server/install/network-protocols-and-network-libraries

SQL Client config, 32 and 64 bit
http://sqlblog.karaszi.com/sql-client-config-32-and-64-bit/

Troubleshooting Connectivity Issues

- Try to access using Shared Memory on the server:
  - If no access via Shared Memory, troubleshoot the login and the service
- Test the network connectivity
  - Can the server name be resolved?
  - Can the network and the server be reached?
  - Is the client attempting to connect using the correct network interface?
  - Is the client configured to use the right protocol and settings?
  - Is the Browser Service running for named instances that are not using fixed ports?
  - Are instance aliases correctly configured?
  - Is a firewall blocking connectivity?
- Client aliases are in the client's registry
  - HKLM\SOFTWARE\Microsoft\MSSQLServer\Client
  - HKLM\SOFTWARE\WOW6432Node\Microsoft\MSSQLServer\Client

Troubleshoot connectivity issues in SQL Server
https://docs.microsoft.com/en-US/troubleshoot/sql/connect/resolve-connectivity-errors-overview

A network-related or instance-specific error occurred while establishing a connection to SQL Server
https://docs.microsoft.com/en-us/troubleshoot/sql/connect/network-related-or-instance-specific-error-occurred-while-establishing-connection

How to Add a Hostname Alias for a SQL Server Instance (several named instance, all listening on port 1433
https://docs.microsoft.com/en-us/archive/blogs/dbrowne/how-to-add-a-hostname-alias-for-a-sql-server-instance

Troubleshooting Login Failures

- Windows logins:
  - Is the domain controller available?
  - Can SQL Server communicate with the domain controller?
- SQL Server logins:
  - Is SQL Server configured for mixed mode authentication?
  - Is the password correct?
  - Is the login locked or is there a requirement to change the password?
- General considerations:
  - Is the login enabled and does it have CONNECT permission?
  - Is the default/requested database available and is access permitted?
- The *reason* for the failed login is in the errorlog file

Troubleshooting "Login failed for user" errors
https://docs.microsoft.com/en-us/troubleshoot/sql/connect/login-failed-for-user

Useful resources

- Interactive troubleshooting guide
  - https://support.microsoft.com/en-us/help/4009936/solving-connectivity-errors-to-sql-server
- General article on troubleshooting communication problems
  - https://msdn.microsoft.com/library/mt750266.aspx
- How to have several instances, all listening to port 1433
  - https://docs.microsoft.com/en-us/archive/blogs/dbrowne/how-to-add-a-hostname-alias-for-a-sql-server-instance

Interactive troubleshooting guide
https://support.microsoft.com/en-us/help/4009936/solving-connectivity-errors-to-sql-server

General article on troubleshooting communication problems
https://msdn.microsoft.com/library/mt750266.aspx

How to have several instances, all listening to port 1433
https://docs.microsoft.com/en-us/archive/blogs/dbrowne/how-to-add-a-hostname-alias-for-a-sql-server-instance

Lab 14: Troubleshooting

- Ex 1: Handle a failed connection situation
- Ex 2: Handle an unstable SQL Server instance situation

Estimated Time: 45 minutes

Module 15

Data transfer

Module Overview

- Overview of data import and export
- Linked servers and SSIS
- Bcp, BULK INSERT and OPENROWSET
- DACPAC and BACPAC

Lesson: Overview of data import and export

- Overview of import and export
- Data transfer tools
- Considerations when importing data to SQL server
- Handling indexes during import
- Disabling and enabling constraints

Overview of import and export

- There are various scenarios for export and import
  - Ranging in complexity from export to comma-separated file to populating a data warehouse
- More complex data transfer routines is often referred to as
  - ETL: Extract, Transform, Load
  - ELT, Extract, Load, Transform
- Scenarios:
  - Copying from a table to a comma-separated file
  - Exporting query result to a file
  - Importing file data to a table
  - Populating a data warehouse, including data transformations

Data transfer tools

- Since the usage cases and complexity of a data transfer scenario varies, we have several options
  - SQL Server Integration Services
    - Import and Export Wizard
  - Bcp.exe (bulk copy program)
  - BULK INSERT
  - OPENROWSET (BULK)
- We will look at these, and other tools in this module

Considerations when importing data to SQL server

- Disable triggers
  - Do not carry the overhead of firing the triggers
  - But they will not do the job they were designed to do
- Minimize locking
  - Use of TABLOCK or similar to speed up import and reduce memory footprint
- Minimize logging
  - Do not have the database in FULL recovery model
  - Might affect your backup and restore routines
- Specify a reasonable transaction size

Handling indexes during import

- Disable an index
  - ALTER INDEX … DISABLE
  - Prevents user access to the index
  - Prevents access to the data for a clustered index (pretty meaningless option)
  - Keeps index definition in metadata
- Enable an index
  - ALTER INDEX … REBUILD
  - Is easy to automate because the definition is available from metadata
- Above is alternative to DROP and CREATE index

Disabling and enabling constraints

- Disabling PRIMARY KEY and UNIQUE constraints:
  - Achieved by disabling the associated index
  - Causes associated indexes to be rebuilt when enabled
  - Can cause failures when re-enabled if data that violates the constraint exists
  - Causes associated foreign key constraints to be disabled
- Disabling FOREIGN KEY and CHECK constraints:
  - Performed directly on the constraint
  - Constraint must be enabled WITH CHECK to verify existing data, otherwise it is untrusted

Lesson: Linked server and SSIS

- Linked servers
- Overview of SQL Server Integration Services
- The SQL Server Import and Export Wizard

Linked Servers

- Execute commands against remote data sources
- Managing Linked Servers:
  - From SSMS or using Transact-SQL
  - Define a data source and a security context for the connection
- Querying Linked Servers:
  - Four-part name:
    - server.database.schema.table
  - OPENQUERY:
    - Pass-through query—must be in linked server's query language
    - If the provider supports it, you can run DML operations against results returned by OPENQUERY

Overview of SQL Server Integration Services

- The SSIS Service
  - A platform for ETL operations
  - Optionally installed as a feature of SQL Server
  - Control flow engine
    - Runtime resources and operational support for data flow
  - Data flow engine
    - Pipeline architecture for buffer-oriented rowset processing
- SSIS Projects
  - Organize related packages together
- SSIS Packages
  - Control flow definition and data flow definition
  - Executed using for instance dtexec.exe, dtexecui.exe or the Agent SSIS jobstep
- SQL Server Data Tools
  - The tool with which you develop packages (among other things)

SQL Server Integration Services
https://docs.microsoft.com/en-us/sql/integration-services/sql-server-integration-services

SQL Server Data Tools
https://learn.microsoft.com/en-us/sql/ssdt/sql-server-data-tools

The SQL Server Import and Export Wizard

- Simplified interface for creating SSIS packages for data import and export
  - Limited support for transformations
- Packages may be executed immediately, or saved for later execution

Import and Export Data with the SQL Server Import and Export Wizard
https://docs.microsoft.com/en-us/sql/integration-services/import-export-data/import-and-export-data-with-the-sql-server-import-and-export-wizard

Lesson: Bcp, BULK INSERT and OPENROWSET

- The bcp Utility
- The BULK INSERT Statement
- The OPENROWSET Function

The bcp Utility

- Command-line tool to import and export data

```
bcp AdventureWorks.Sales.Currency out D:\Currency.csv -S MIA-SQL -T -c -t , -r \n
```

- Use format files to define data schema:
  - Create a format file with the format nul direction

```
bcp AdventureWorks.Sales.Currency format nul -S MIA-SQL -T -c -t , -r \n -x -f
D:\CurrencyFmt.xml
```

  - Use a format file:

```
bcp AdventureWorks.Sales.Currency out D:\Currency.csv -S MIA-SQL -T -f
D:\CurrencyFmt.xml
```

Import and export bulk data using bcp (SQL Server)
https://docs.microsoft.com/en-us/sql/relational-databases/import-export/import-and-export-bulk-data-by-using-the-bcp-utility-sql-server

bcp Utility
https://docs.microsoft.com/en-us/sql/tools/bcp-utility

The BULK INSERT Statement

- Transact-SQL command that provides similar options to bcp in
- Runs inside the database engine process
- Can be part of a user-defined transaction

```
BULK INSERT AdventureWorks.Sales.OrderDetail
   FROM 'F:\orders\neworders.txt'
   WITH
      (
         FIELDTERMINATOR ='|',
         ROWTERMINATOR ='\n'
      );
```

Use BULK INSERT or OPENROWSET(BULK...) to import data to SQL Server
https://docs.microsoft.com/en-us/sql/relational-databases/import-export/import-bulk-data-by-using-bulk-insert-or-openrowset-bulk-sql-server

The OPENROWSET Function

- SELECT rows from a data file based on a format file
  - Data may then be used in other Transact-SQL statements
- Import rows from any OLE DB provider
  - ad hoc distributed queries server level setting must be enabled
  - OLE DB provider must be configured to allow ad hoc access
- Allow you to import a file as a BLOB into a single column/row

Use BULK INSERT or OPENROWSET(BULK...) to import data to SQL Server
https://docs.microsoft.com/en-us/sql/relational-databases/import-export/import-bulk-data-by-using-bulk-insert-or-openrowset-bulk-sql-server

Lesson: DACPAC and BACPAC

- Data-Tier Application Overview
- Deploying Data-Tier Applications
- Performing In-Place Upgrades of Data-Tier Applications
- Extracting Data-Tier Applications

Data-Tier Application Overview

- DACPAC
  - Zip-file describing the schema for a database
- Creating a DAC
  - Application developers (Visual Studio Database Project)
  - DBAs (generate from existing database)
- DAC registration
  - Versioning metadata stored by SQL Server
- BACPAC
  - Like a DACPAC that also includes data

Data-tier Applications
https://docs.microsoft.com/en-us/sql/relational-databases/data-tier-applications/data-tier-applications

Deploying Data-Tier Applications

- SSMS Deploy Data-Tier Application Wizard
  - Minimal customization available
- SqlPackage.exe
  - Command-line tool
- Windows PowerShell
  - Microsoft.SqlServer.Dac namespace
- Azure Management Portal
  - Deploy a DACPAC as an Azure SQL Database

SqlPackage.exe
https://docs.microsoft.com/en-us/sql/tools/sqlpackage/sqlpackage

Microsoft.SqlServer.Dac Namespace
https://docs.microsoft.com/en-us/dotnet/api/microsoft.sqlserver.dac

Performing In-Place Upgrades of Data-Tier Applications

- A DACPAC can be used to carry out an in-place upgrade of an existing DAC
  - Differences are identified
  - A script is generated to make the schema of the DAC match the schema defined in the DACPAC
- Upgrade tools:
  - SSMS Upgrade Data-Tier Application Wizard
  - SqlPackage.exe
  - PowerShell
- Upgrade behavior can be controlled using settings

Extracting Data-Tier Applications

- A database can be extracted to a DACPAC
  - The source database might only contain objects supported by DAC
  - No contained users
- Export tools:
  - SSMS Export Data-Tier Application Wizard
  - SqlPackage.exe
  - PowerShell

Optional lab 15: Data transfer

- Ex 1: Use BCP to export to comma-separated file
- Ex 2: Create a BACPAC file

Estimated Time: 30 minutes

Course Evaluation

Your evaluation of this course is important to us!

Thank you!