**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**KHOA AN TOÀN THÔNG TIN**

**A logo of a padlock

Description automatically generated**

**BÁO CÁO BÀI TẬP LỚN**

**HỌC PHẦN: AN TOÀN ỨNG DỤNG WEB VÀ CƠ SỞ DỮ LIỆU**

**MÃ HỌC PHẦN: INT14105**

**ĐỀ TÀI: TÌM HIỂU VỀ HỆ QUẢN TRỊ CSDL MICROSOFT SQL SERVER 2016**

Các sinh viên thực hiện:

B21DCAT227 Nguyễn Quốc Vượng

Nguyễn Hữu Tài

Đặng Thế Việt

Ngô Anh Quý

Tên nhóm: 05

Tên lớp: 01

Giảng viên hướng dẫn: ThS.Ninh Thị Thu Trang

**HÀ NỘI 2024**

**PHÂN CÔNG NHIỆM VỤ NHÓM THỰC HIỆN**

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Công việc / Nhiệm vụ** | **SV thực hiện** | **Thời hạn  hoàn thành** |
| 1 | Phụ trách **kiến trúc hệ thống** trong Chương 1, phân tích các thành phần cơ bản và cách chúng tương tác với nhau. | Ngô Anh Quý | 8/12/2024 |
| 2 | Nghiên cứu và trình bày về **các thành phần của SQL Server 2016**, bao gồm các công cụ và module hỗ trợ trong hệ thống. | Nguyễn Quốc Vượng | 8/12/2024 |
| 3 | Tìm hiểu và trình bày về **các tính năng nổi bật** trong Chương 2, tập trung vào các cải tiến về hiệu năng, tích hợp dữ liệu lớn, hỗ trợ phân tích và mở rộng trên đám mây. | Đặng Thế Việt | 10/12/2024 |
| 4 | Phụ trách nội dung về **cơ chế bảo mật** của SQL Server 2016 trong Chương 2, bao gồm Always Encrypted, quản lý quyền truy cập và các công cụ giám sát bảo mật. | Nguyễn Hữu Tài | 10/12/2024 |

**NHÓM THỰC HIỆN TỰ ĐÁNH GIÁ**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **TT** | **SV thực hiện** | **Thái độ tham gia** | **Mức hoàn thành CV** | **Kỹ năng giao tiếp** | **Kỹ năng hợp tác** | **Kỹ năng lãnh đạo** |
| 1 | Nguyễn Quốc Vượng | 5 | 5 | 5 | 5 | 4 |
| 2 | Nguyễn Hữu Tài | 5 | 5 | 5 | 5 | 4 |
| 3 | Ngô Anh Quý | 5 | 5 | 5 | 5 | 4 |
| 4 | Đặng Thế Việt | 5 | 5 | 5 | 5 | 4 |

***Ghi chú***:

* Thái độ tham gia: Đánh giá điểm thái độ tham gia công việc chung của nhóm (từ 0: không tham gia, đến 5: chủ động, tích cực).
* Mức hoàn thành CV: Đánh giá điểm mức độ hoàn thành công việc được giao (từ 0: không hoàn thành, đến 5: hoàn thành xuất sắc).
* Kỹ năng giao tiếp: Đánh giá điểm khả năng tương tác, giao tiếp trong nhóm (từ 0: không hoặc giao tiếp rất yếu, đến 5: giao tiếp xuất sắc).
* Kỹ năng hợp tác: Đánh giá điểm khả năng hợp tác, hỗ trợ lẫn nhau, giải quyết mâu thuẫn, xung đột
* Kỹ năng lãnh đạo: Đánh giá điểm khả năng lãnh đạo (từ 0: không có khả năng lãnh đạo, đến 5: có khả năng lãnh đạo tốt, tổ chức và điều phối công việc trong nhóm hiệu quả).

MỤC LỤC

[MỤC LỤC 3](#_Toc184632061)

[DANH MỤC CÁC HÌNH VẼ 4](#_Toc184632062)

[DANH MỤC CÁC BẢNG BIỂU 4](#_Toc184632063)

[DANH MỤC CÁC TỪ VIẾT TẮT 5](#_Toc184632064)

[MỞ ĐẦU 6](#_Toc184632065)

[CHƯƠNG 1. CHƯƠNG 1. Tổng quan về SQL Server 2016 7](#_Toc184632066)

[1.1 Kiến trúc 7](#_Toc184632067)

[1.1.1 Kiến trúc chung - General 7](#_Toc184632068)

[1.1.2 Kiến trúc bộ nhớ - Memory 8](#_Toc184632069)

[1.1.3 Kiến trúc file dữ liệu - Data file 9](#_Toc184632070)

[1.1.4 Kiến trúc file nhật ký - Log file 10](#_Toc184632071)

[1.2 Thành phần 12](#_Toc184632072)

[1.2.1 Database Engine (Công Cụ Cơ Sở Dữ Liệu) 12](#_Toc184632073)

[1.2.2 Machine Learning Services (MLS) - Dịch Vụ Học Máy 13](#_Toc184632074)

[1.2.3 Integration Services (SSIS) - Dịch Vụ Tích Hợp 13](#_Toc184632075)

[1.2.4 Analysis Services (SSAS) - Dịch Vụ Phân Tích 13](#_Toc184632076)

[1.2.5 Reporting Services (SSRS) - Dịch Vụ Báo Cáo 14](#_Toc184632077)

[1.2.6 Replication - Sao Chép 14](#_Toc184632078)

[1.2.7 Data Quality Services (DQS) - Dịch Vụ Chất Lượng Dữ Liệu 15](#_Toc184632079)

[1.2.8 Master Data Services (MDS) - Dịch Vụ Dữ Liệu Chính 15](#_Toc184632080)

[CHƯƠNG 2. Tính năng và cơ chế bảo mật 17](#_Toc184632081)

[2.1 Tính năng 17](#_Toc184632082)

[2.1.1 Always Encrypted 17](#_Toc184632083)

[2.1.2 In-Memory OLTP 17](#_Toc184632084)

[2.1.3 Temporal Tables 18](#_Toc184632085)

[2.1.4 Stretch Database 18](#_Toc184632086)

[2.1.5 Row-Level Security 19](#_Toc184632087)

[2.1.6 Dynamic Data Masking 20](#_Toc184632088)

[2.1.7 Hỗ trợ JSON 20](#_Toc184632089)

[2.1.8 PolyBase 21](#_Toc184632090)

[2.1.9 Query Store 22](#_Toc184632091)

[2.1.10 Live Query Statistics 23](#_Toc184632092)

[2.1.11 Master Data Services 23](#_Toc184632093)

[2.1.12 Data Quality Services 24](#_Toc184632094)

[2.2 Cơ chế bảo mật 25](#_Toc184632095)

[2.2.1 Chế độ xác thực 25](#_Toc184632096)

[2.2.2 Quản lý quyền 26](#_Toc184632097)

[2.2.3 Mã hóa 29](#_Toc184632098)

[CHƯƠNG 3. DEMO 32](#_Toc184632099)

[3.1 Tạo cơ sở dữ liệu 32](#_Toc184632100)

[3.2 Thêm bảng 32](#_Toc184632101)

[3.3 Xóa bảng 32](#_Toc184632102)

[3.4 Thêm, sửa, xóa và xem dữ liệu 32](#_Toc184632103)

[3.5 Tạo sao lưu cơ sở dữ liệu 33](#_Toc184632104)

[MỤC LỤC 34](#_Toc184632105)

[DANH MỤC CÁC HÌNH VẼ 35](#_Toc184632106)

[DANH MỤC CÁC BẢNG BIỂU 35](#_Toc184632107)

[DANH MỤC CÁC TỪ VIẾT TẮT 36](#_Toc184632108)

[MỞ ĐẦU 37](#_Toc184632109)

[CHƯƠNG 4. TỔNG QUAN VỀ HỆ THỐNG MOODLE LMS 38](#_Toc184632110)

[4.1 Giới thiệu 38](#_Toc184632111)

[4.2 Kiến trúc và tính năng của Moodle LMS 38](#_Toc184632112)

[4.2.1 Kiến trúc và các thành phần của Moodle LMS 38](#_Toc184632113)

[4.3 Kết chương 38](#_Toc184632114)

[CHƯƠNG 5. THIẾT KẾ VÀ CÀI ĐẶT CÁC PLUGIN CHO MOODLE LMS 39](#_Toc184632115)

[5.1 Khái quát 39](#_Toc184632116)

[5.2 Thiết kế và cài đặt các plugin hỗ trợ quản trị ngân hàng câu hỏi 39](#_Toc184632117)

[5.2.1 Các yêu cầu 39](#_Toc184632118)

[5.3 Kết chương 39](#_Toc184632119)

[CHƯƠNG 6. THỬ NGHIỆM VÀ ĐÁNH GIÁ 40](#_Toc184632120)

[6.1 Triển khai và thử nghiệm trên hệ thống Moodle LMS cục bộ 40](#_Toc184632121)

[6.1.1 Triển khai trên hệ thống Moodle LMS cục bộ 40](#_Toc184632122)

[6.2 Kết chương 40](#_Toc184632123)

[KẾT LUẬN 41](#_Toc184632124)

[TÀI LIỆU THAM KHẢO 42](#_Toc184632125)

DANH MỤC CÁC HÌNH VẼ

[Hình 1 8](#_Toc184633441)

[Hình 2 Hình ảnh minh họa các quyền được gán cho các vai trò máy chủ (SQL Server 2019 trở về trước) 29](#_Toc184633442)

[Hình 3 Các quyền được gán cho các vai trò cơ sở dữ liệu cố định 31](#_Toc184633443)

[Hình 4 33](#_Toc184633444)

[Hình 5 Câu lệnh SQL tạo cơ sở dữ liệu mới 34](#_Toc184633445)

[Hình 6 Câu lệnh SQL tạo bảng mới 34](#_Toc184633446)

[Hình 7 Câu lệnh SQL xóa bảng và dữ liệu bên trong bảng 34](#_Toc184633447)

[Hình 8 Câu lệnh SQL thêm dữ liệu vào bảng 35](#_Toc184633448)

[Hình 9 Câu lệnh SQL sửa dữ liệu trong bảng 35](#_Toc184633449)

[Hình 10 Câu lệnh SQL xóa dữ liệu trong bảng 35](#_Toc184633450)

[Hình 11 Câu lệnh SQL xem dữ liệu trong bảng 35](#_Toc184633451)

[Hình 12 Câu lệnh SQL tạo bản sao lưu cơ sở dữ liệu 35](#_Toc184633452)

[Hình 13 Câu lệnh SQL khôi phục cơ sở dữ liệu từ một bản sao lưu 35](#_Toc184633453)

DANH MỤC CÁC BẢNG BIỂU

[Bảng 1. 30](#_Toc184634085)

[Bảng 2. 32](#_Toc184634086)

DANH MỤC CÁC TỪ VIẾT TẮT

|  |  |  |
| --- | --- | --- |
| **Từ  viết tắt** | **Thuật ngữ tiếng Anh/Giải thích** | **Thuật ngữ tiếng Việt/Giải thích** |
| LMS | Learning Management System | Hệ thống quản lý học tập |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

MỞ ĐẦU

Trong thời đại công nghệ số, việc quản lý và khai thác dữ liệu hiệu quả đã trở thành yếu tố cốt lõi để các tổ chức và doanh nghiệp cạnh tranh và phát triển. SQL Server 2016, một hệ quản trị cơ sở dữ liệu quan hệ (Relational Database Management System - RDBMS) do Microsoft phát triển, đã ra đời nhằm đáp ứng nhu cầu ngày càng cao trong việc xử lý dữ liệu phức tạp, cung cấp hiệu năng vượt trội và đảm bảo an toàn thông tin.

SQL Server 2016 không chỉ kế thừa các tính năng mạnh mẽ từ các phiên bản trước mà còn mang đến nhiều cải tiến đáng kể về kiến trúc, hiệu năng và khả năng bảo mật. Với sự hỗ trợ từ công nghệ In-Memory OLTP, SQL Server 2016 cho phép tăng tốc độ xử lý giao dịch trực tuyến. Bên cạnh đó, tính năng PolyBase giúp tích hợp và phân tích dữ liệu lớn một cách hiệu quả, trong khi Always Encrypted đảm bảo an toàn dữ liệu ngay cả trong môi trường phức tạp.

Mục tiêu của báo cáo này là phân tích toàn diện các khía cạnh nổi bật của SQL Server 2016, bao gồm kiến trúc, tính năng và cơ chế bảo mật. Qua đó, báo cáo sẽ đánh giá khả năng ứng dụng của hệ quản trị cơ sở dữ liệu này trong các tình huống thực tế.

Nội dung của báo cáo được chia thành 3 chương chính như sau:

* **Chương 1**: Tổng quan về SQL Server 2016, tập trung vào việc phân tích kiến trúc hệ thống và các thành phần cơ bản, giúp người đọc hiểu rõ cách thức hoạt động của hệ quản trị này.
* **Chương 2**: Phân tích chi tiết các tính năng nổi bật của SQL Server 2016, bao gồm các cải tiến về hiệu năng, khả năng tích hợp, hỗ trợ phân tích dữ liệu nâng cao và bảo mật.
* **Chương 3**: Thực hiện demo cài đặt và sử dụng SQL Server 2016, tập trung vào việc kiểm chứng hiệu quả của các tính năng thông qua các kịch bản thử nghiệm thực tế.

1. CHƯƠNG 1. Tổng quan về SQL Server 2016
   1. Kiến trúc
      1. Kiến trúc chung - General

**A computer screen shot of a computer

Description automatically generated**



 Kiến trúc của SQL Server gồm các phần:

**Client**: Nơi khởi tạo yêu cầu.

**Query**: Truy vấn SQL là ngôn ngữ bậc cao.

**Logical Units**: Keyword, biểu thức, toán tử,...

**N/W Packets**: Code liên quan đến mạng.

**Protocols**: Trong SQL Server ta có 4 giao thức:

**Shared memory**: Dành cho các kết nối cục bộ và mục đích khắc phục sự cố.

**Named pipes**: Dành cho những kết nối trong mạng LAN.

**TCP/IP:** Dành cho các kết nối với mạng WAN.

**VIA-Virtual Interface Adapter**: Yêu cầu phần cứng đặc biệt được thiết lập bởi nhà cung cấp và cũng không được hỗ trợ từ bản SQL 2016.

**Server**: Nơi SQL Services được cài đặt và có database.

**Relational Engine**: Đây là nơi sự thực hiện (execution) thực sự sẽ được hoàn thành. Nó chứa bộ phân tích Query, tối ưu hóa Query và bộ thực thi Query.

**Query Parser (Command Parser)** và **Compiler (Translator)**: chịu trách nhiệm kiểm tra cú pháp của truy vấn và chuyển đổi truy vấn sang ngôn ngữ của máy.

**Query Optimizer:** Nó sẽ chuẩn bị output là Execution Plan bằng cách lấy input là truy vấn, các thống kê và cây Algebrizer.

**Execution Plan**: Giống như một bản đồ chỉ đường, chứa thứ tự các bước thực hiện như là một phần của việc thực hiện các truy vấn.

**Query Executor**: Đây là nơi truy vấn được thực hiện từng bước một, với sự giúp đỡ của Execution Plan và cũng là nơi Storage Engine sẽ được liên lạc.

**Storage Engine**: Chịu trách nhiệm lưu trữ, truy xuất dữ liệu trong hệ thống lưu trữ (ổ đĩa, SAN,...), thao tác dữ liệu, khóa và quản lý các transaction.

**SQL OS:** Nằm giữa máy host (Windows OS) và SQL Server. Tất cả các hoạt động được thực hiện trên engine cơ sở dữ liệu được "chăm sóc" bởi SQL OS. SQL OS cung cấp các dịch vụ hệ điều hành khác nhau, chẳng hạn như hoạt động quản lý bộ nhớ với buffer pool, log buffer, phát hiện deadlock (khóa chết) bằng cách sử dụng cấu trúc block và lock.

**Checkpoint**: Checkpoint là một tiến trình nội bộ, ghi tất cả các trang đã sửa đổi (gọi là Dirty Page) từ Buffer Cache vào ổ đĩa vật lý. Ngoài ra, nó cũng ghi các bản log từ Log Buffer vào file vật lý. Việc ghi lại các Dirty Page vào ổ đĩa còn được biết đến như là Hardening of dirty pages (cứng hóa các Dirty Page).

**Lazy Writer**: Lazy Writer sẽ đẩy các Dirty Page và ổ cứng vì một lý do hoàn toàn khác, đó là giải phóng bộ nhớ trong Buffer Pool. Điều này xảy ra khi SQL Server đang bị thiếu bộ nhớ. Tiến trình này được kiểm soát bởi một tiến trình Internal và không có thiết lập cho nó. SQL Server liên tục giám sát việc sử dụng bộ nhớ để đánh giá tính khả dụng và mức cạnh tranh tài nguyên, giúp đảm bảo luôn có sẵn một dung lượng trống nhất định. Khi phát hiện bất kỳ sự xung đột tài nguyên nào, nó sẽ kích hoạt Lazy Writer để chuyển một số Dirty Page vào ổ đĩa và giải phóng bộ nhớ. Nó sử dụng thuật toán Least Recently Used (LRU) để quyết định trang nào sẽ được đẩy vào ổ cứng. Nếu Lazy Writer luôn hoạt động, nó có thể tạo ra nút thắt cổ chai với bộ nhớ.

* + 1. Kiến trúc bộ nhớ - Memory

     Bộ nhớ của SQL Server có những tính năng nổi bật:

Một trong những mục tiêu thiết kế cơ bản của tất cả phần mềm cơ sở dữ liệu là giảm thiểu I/O ổ đĩa vì quá trình đọc và ghi đĩa là một trong những hành động sử dụng nhiều tài nguyên nhất.

Bộ nhớ trong Windows có thể được gọi với Virtual Address Space, được chia sẻ bởi chế độ Kernel (chế độ OS) và User (ứng dụng như SQL Server).

User address space của SQL Server được chia thành 2 phần: MemToLeave và Buffer Pool.

Kích thước của MemToLeave (MTL) và Buffer Pool (BPool) được quyết định bởi SQL Server trong quá trình khởi động.

Buffer Management là một thành phần quan trọng nếu muốn đạt được hiệu suất I/O cao. Nó bao gồm 2 cơ chế: Buffer Manager để truy cập và cập nhật các trang cơ sở dữ liệu và Buffer Pool để cắt giảm I/O file vào database.

Buffer Pool được chia thành nhiều phần, quan trọng nhất là Buffer Cache và Procedure Cache. Buffer Cache giữ các trang dữ liệu trong bộ nhớ để những dữ liệu thường xuyên truy cập có thể trích xuất từ bộ nhớ cache. Quá trình thay thế sẽ đọc các trang dữ liệu từ ổ đĩa. Đọc dữ liệu từ cache sẽ tối ưu hóa hiệu suất bằng cách giảm thiểu số lượng các thao tác I/O, vốn chậm hơn so với truy xuất dữ liệu từ bộ nhớ.

Procedure Cache giữ các thủ tục được lưu trữ và các Execution Plan để tối ưu hóa số lần Execution Plan được tạo. Lập trình viên có thể tìm thấy thông tin về dung lượng và hoạt động trong Procedure Cache sử dụng lệnh DBCC PROCCACHE.

Các phần khác của Buffer Pool bao gồm:

Các cấu trúc dữ liệu mức hệ thống: Chứa dữ liệu mức Instance về cơ sở dữ liệu, khóa.

Log Cache: Dành riêng cho việc đọc và ghi các trang transaction.

Connection Context: Mỗi kết nối với Instance có một vùng nhỏ bộ nhớ để ghi trạng thái hiện tại của kết nối. Thông tin này bao gồm thủ tục được lưu trữ và các thông số hàm do người dùng xác định, vị trí con trỏ chuột và nhiều hơn nữa.

Stack Space: Windows phân bổ stack space cho mỗi luồng được bắt đầu với SQL Server.

* + 1. Kiến trúc file dữ liệu - Data file

Kiến trúc này có các thành phần sau:

**File Group:**

Các file cơ sở dữ liệu có thể nhóm lại với nhau thành các nhóm file để phân bổ và quản lý theo mục đích. Một file chỉ có thể là thành viên của một nhóm file. Các file log không thể nhóm vào File Group vì dung lượng file log được quản lý riêng biệt với dung lượng dữ liệu.

Có hai loại File Group trong SQL Server là Primary và User-defined. Primary chứa các file dữ liệu chính và bất kỳ file nào không được gán cụ thể cho File Group khác. Tất cả các trang cho bảng hệ thống được cấp phát trong Primary. User-defined là các nhóm file do người dùng định nghĩa, nó được chỉ định bằng cách sử dụng từ khóa file group trong lệnh tạo cơ sở dữ liệu hoặc xóa cơ sở dữ liệu.

**File:**

Cơ sở dữ liệu có 3 loại file Primary (file dữ liệu chính), Secondary (file dữ liệu phụ) và Log (file nhật ký). Primary là điểm bắt đầu của cơ sở dữ liệu và trỏ đến các file khác trong cơ sở dữ liệu.

Mỗi cơ sở dữ liệu có một Primary. Lập trình viên có thể đặt phần mở rộng cho các file dữ liệu chính là gì cũng được, nhưng khuyến nghị là nên để .mdf. File dữ liệu phụ là file khác file dữ liệu chính. Một cơ sở dữ liệu có thể có nhiều hoặc chỉ có một file dữ liệu phụ. Phần mở rộng cho file dữ liệu phụ nên đặt là .ndf.

Các file log giữ tất cả thông tin được sử dụng để phục hồi cơ sở dữ liệu. Cơ sở dữ liệu phải có ít nhất một file log. Chúng ta có thể có nhiều file log cho một cơ sở dữ liệu. Phần mở rộng nên đặt là .ldf.

File có 2 tên là Logical và Physical. Logical được sử dụng để tham chiếu đến file trong tất cả các lệnh T-SQL. Tên Physical là OS\_file\_name, nó phải tuân theo quy tắc của hệ điều hành. File dữ liệu và file log có thể được đặt trên hệ thống file FAT hoặc NTFS, nhưng không thể đặt trên các hệ thống file nén. Có thể có tối đa 32.767 file trong một cơ sở dữ liệu.

**Extent:**

Extent là một đơn vị cơ bản trong đó không gian được phân bổ cho mỗi bảng, chỉ mục. Mỗi Extent là 8 trang liền kề hoặc 64KB. SQL Server có 2 loại Extent là Uniform và Mixed. Uniform được tạo thành từ một object duy nhất, Mixed được tạo thành từ tối đa 8 object.

**Page:**

Page (trang) là đơn vị cơ bản trong lưu trữ dữ liệu của SQL Server. Kích thước của một trang là 8KB. Bắt đầu mỗi trang là 96 byte tiêu đề, được sử dụng để lưu trữ thông tin hệ thống như loại trang, số lượng không gian trống trên trang và ID của đối tượng sở hữu trang. Có 9 loại trang dữ liệu trong SQL Server:

Data: Các hàng dữ liệu với tất cả dữ liệu từ text, ntext và ảnh.

Index: Các mục chỉ mục.

Text\Image: Dữ liệu text, ntext và ảnh.

GAM: Thông tin về extent được chỉ định.

SGAM: Thông tin về extent được cấp phát ở mức hệ thống.

Page Free Space (PFS): Thông tin về không gian trống hiện có trên các trang.

Index Allocation Map (IAM): Thông tin về extent được sử dụng bởi bảng hoặc chỉ mục.

Bulk Changed Map (BCM): Thông tin về extent được sửa đổi bởi hoạt động hàng loạt kể từ lệnh ghi sao lưu cuối cùng.

Differential Changed Map (DCM): Thông tin về extent đã thay đổi kể từ lệnh sao lưu cơ sở dữ liệu gần nhất.

* + 1. Kiến trúc file nhật ký - Log file

Các log transaction trên SQL Server hoạt động hợp lý khi nó là chuỗi các bản ghi log. Mỗi bản ghi log được xác định bởi Log Sequence Number (LSN), chứa ID của transaction mà nó thuộc về.

Log ghi lại những sửa đổi dữ liệu hoặc các hoạt động được thực hiện hay lấy hình ảnh trước và sau khi dữ liệu bị chỉnh sửa. Hình ảnh trước là bản sao của dữ liệu trước khi thao tác được thực hiện, hình ảnh sau là bản sao của dữ liệu sau khi thao tác đã được thực hiện.

Các bước để phục hồi một hoạt động phụ thuộc vào loại bản ghi log**:**

Thao tác logic được log.

Để đi tới thao tác logic ở trước, thao tác sẽ được thực hiện lại.

Để quay lại thao tác logic phía sau, thao tác logic đảo ngược sẽ được thực hiện.

Ảnh trước và sau được log.

Để đi tới thao tác ở trước, ảnh sau sẽ được áp dụng.

Để quay lại thao tác phía sau, ảnh trước sẽ được áp dụng.

Các thao tác khác nhau đã được ghi lại trong bản log transaction. Những thao tác sau sẽ có trong đó:

Bắt đầu và kết thúc mỗi transaction.

Mọi sửa đổi dữ liệu (chèn, cập nhật, xóa), bao gồm các thay đổi các thủ tục lưu trữ hệ thống hoặc lệnh ngôn ngữ định nghĩa dữ liệu (DDL) đến bảng, bao gồm cả bảng hệ thống.

Mọi extent và phân bổ, hủy phân bổ trang.

Tạo hoặc xóa bảng, chỉ mục.

Các thao tác rollback cũng được log lại. Mỗi transaction sẽ giữ một khoảng không gian trong bản log để chắc chắn rằng có đủ không gian log cần thiết cho rollback thực hiện lệnh hoặc thông báo lỗi. Không gian này sẽ được giải phóng khi transaction hoàn tất.

Phần của file log từ bản log đầu tiên (bắt buộc phải có để khôi phục lại toàn bộ cơ sở dữ liệu thành công) đến bản log cuối cùng được gọi là phần hoạt động của log hay log hoạt động. Đây là phần bản log bắt buộc để có thể phục hồi cơ sở dữ liệu đầy đủ. Không có phần nào trong log hoạt động được cắt xén. LSN của bản ghi log đầu tiên được gọi là LSN phục hồi tối thiểu (Min LSN).

SQL Server Database Engine chia mỗi file log Physical thành một số file log ảo. File log ảo không có kích thước cố định và không có số lượng file log ảo cố định cho mỗi file log Physical.

Database Engine chọn dung lượng cho file log ảo một cách tự động khi nó tạo hoặc mở rộng file log. Database Engine cố duy trì số lượng file ảo nhỏ. Kích thước của file log ảo không thể cấu hình hay thiết lập bởi quản trị viên. Duy nhất chỉ có khi file log Physical được xác định kích thước nhỏ và giá trị growth\_increment thì file log ảo mới ảnh hưởng đến hiệu suất hệ thống.

Giá trị kích thước là kích thước khởi tạo cho file log và growth\_increment là lượng không gian được thêm cho file mỗi khi file yêu cầu thêm không gian mới. Khi file log đạt đến kích thước lớn vì có nhiều sự gia tăng nhỏ, chúng sẽ có nhiều file log ảo. Điều này có thể làm chậm quá trình khởi động database và các hoạt động sao lưu, phục hồi log.

Lời khuyên là lập trình viên nên gán cho file log giá trị kích thước gần với kích thước cuối cùng được yêu cầu và giá trị growth\_increment tương đối lớn. SQL Server sử dụng write-ahead log (WAL), đảm bảo rằng không có sự sửa đổi dữ liệu nào được ghi vào ổ đĩa trước khi bản log liên quan được ghi vào ổ đĩa. Điều này giúp duy trì các thuộc tính ACID cho transaction.

**Algebrizer trong SQL**

Algebrizer là một tiến trình trong quá trình thực hiện truy vấn. Nó bắt đầu làm việc sau Parser. Khi Query Parser tìm thấy một truy vấn đúng cú pháp, nó sẽ chuyển đến cho Algebrizer và công việc của Algebrizer bắt đầu: Algebrizer chịu trách nhiệm xác minh các object và tên cột. Ví dụ, nếu tên cột bị viết sai trong truy vấn, Algebrizer phải có trách nhiệm xác nhận điều đó và tạo ra lỗi. Algebrizer cũng xác định tất cả các loại dữ liệu đang được xử lý trong một truy vấn nhất định. Algebrizer xác minh xem GROUP BY và những cột đã gộp có được đặt đúng nơi hay không. 

**Checkpoint trong SQL Server**

Trong SQL Server 2016 có 4 loại checkpoint:

Automatic: Loại này là checkpoint phổ biến nhất, chạy dưới dạng một tiến trình nền để chắc chắn SQL Server Database có thể được phục hồi trong khoảng thời gian được xác định bởi Recovery Interval trong Server Configuration Option.

Indirect: Loại checkpoint này mới có trên SQL Server 2016. Nó cũng là tiến trình chạy nền nhưng chỉ dành cho một số user cụ thể xác định thời gian phục hồi cho database cụ thể trong tùy chọn cấu hình. Khi Target\_Recovery\_Time cho database cụ thể được chọn, nó sẽ ghi đè lên Recovery Interval được chỉ định cho máy chủ, tránh Automatic checkpoint trên database đó.

Manual: Checkpoint này chạy giống như bất kỳ lệnh SQL nào khác, khi lập trình viên tạo lệnh checkpoint, nó sẽ chạy để hoàn thành. Checkpoint này chỉ chạy trên database hiện tại. Lập trình viên có thể xác định Checkpoint\_Duration trong tùy chọn để chỉ ra khoảng thời gian họ muốn checkpoint được hoàn thành.

Internal: Khi là một user, bạn không thể kiểm soát loại checkpoint này trong những hành động cụ thể như:

Ngắt khởi động một hành động checkpoint trên tất cả các database trừ khi việc shutdown bị lỗi, không bình thường (dùng lệnh Shutdown with nowith).

Khi mô hình phục hồi bị thay đổi từ Full\Bulk-logged sang Simple.

Trong khi đang sao lưu cơ sở dữ liệu.

Nếu database đang ở trong kiểu phục hồi Simple, quá trình checkpoint sẽ tự động thực hiện hoặc khi bản log đã đầy 70%, hoặc dựa trên tùy chọn Recovery Interval của Server.

Lệnh ALTER DATABASE để thêm hoặc xóa file log/dữ liệu cũng khởi tạo một checkpoint.

* 1. Thành phần
     1. Database Engine (Công Cụ Cơ Sở Dữ Liệu)

**Kiến trúc:**

Database Engine là trái tim của SQL Server, chịu trách nhiệm lưu trữ, xử lý và bảo mật dữ liệu. Nó bao gồm nhiều thành phần con, mỗi thành phần đảm nhận một vai trò cụ thể:

* **Relational Engine:** Quản lý dữ liệu theo mô hình quan hệ, bao gồm các bảng, cột, kiểu dữ liệu, ràng buộc, v.v.
* **Query Processor:** Phân tích và tối ưu hóa các truy vấn SQL, tạo kế hoạch thực thi truy vấn hiệu quả.
* **Storage Engine:** Quản lý cách dữ liệu được lưu trữ vật lý trên đĩa, bao gồm các tệp dữ liệu, nhóm tệp, các trang dữ liệu, v.v.
* **Transaction Manager:** Đảm bảo tính toàn vẹn của dữ liệu bằng cách quản lý các giao dịch, đảm bảo ACID properties (Atomicity, Consistency, Isolation, Durability).
* **Buffer Manager:** Quản lý bộ nhớ đệm cho dữ liệu và các trang chỉ mục, giúp tăng tốc độ truy cập dữ liệu.

**Cơ chế hoạt động:**

Khi một ứng dụng gửi truy vấn đến SQL Server, Query Processor sẽ phân tích truy vấn, tạo kế hoạch thực thi tối ưu, sau đó chuyển cho Storage Engine để truy xuất dữ liệu. Storage Engine đọc dữ liệu từ đĩa và đưa vào bộ nhớ đệm (Buffer Manager), sau đó trả kết quả cho Query Processor. Transaction Manager đảm bảo các thay đổi dữ liệu được thực hiện một cách an toàn và nhất quán.

**Tùy chọn cấu hình:**

Database Engine có rất nhiều tùy chọn cấu hình, ảnh hưởng đến hiệu năng, khả năng mở rộng và tính bảo mật. Ví dụ:

* **Memory configuration:** Cấu hình lượng RAM mà SQL Server có thể sử dụng.
* **Max degree of parallelism:** Giới hạn số lượng CPU cores mà một truy vấn có thể sử dụng đồng thời.
* **Compatibility level:** Xác định phiên bản SQL Server mà cơ sở dữ liệu tương thích.

**Kịch bản ứng dụng:**

* Quản lý dữ liệu khách hàng cho một ứng dụng web.
* Lưu trữ dữ liệu giao dịch cho một hệ thống bán hàng.
* Xây dựng một kho dữ liệu doanh nghiệp (data warehouse).
  + 1. Machine Learning Services (MLS) - Dịch Vụ Học Máy

**Kiến trúc:**

MLS tích hợp các ngôn ngữ lập trình Python và R vào SQL Server, cho phép thực hiện các tác vụ học máy trực tiếp trong cơ sở dữ liệu. Nó sử dụng một framework mở rộng (extensibility framework) để thực thi các script Python và R.

**Cơ chế hoạt động:**

Người dùng có thể viết các stored procedures T-SQL gọi đến hàm sp\_execute\_external\_script để thực thi script Python hoặc R. Các script này có thể truy cập dữ liệu trực tiếp trong SQL Server, thực hiện các tác vụ như xử lý dữ liệu, huấn luyện mô hình, dự đoán, v.v.

**Tùy chọn cấu hình:**

* **Python/R runtime version:** Chọn phiên bản Python/R được sử dụng bởi MLS.
* **Packages:** Cài đặt thêm các gói Python/R cần thiết cho các tác vụ học máy.

**Kịch bản ứng dụng:**

* Dự đoán doanh số bán hàng dựa trên dữ liệu lịch sử.
* Phân loại khách hàng theo hành vi mua sắm.
* Phát hiện gian lận trong giao dịch tài chính.
  + 1. Integration Services (SSIS) - Dịch Vụ Tích Hợp

**Kiến trúc:**

SSIS là một nền tảng để xây dựng các giải pháp tích hợp và chuyển đổi dữ liệu hiệu năng cao. Nó sử dụng một kiến trúc dựa trên luồng dữ liệu (data flow) và các thành phần (components) để thực hiện các tác vụ ETL (Extract, Transform, Load).

**Cơ chế hoạt động:**

Người dùng tạo các gói SSIS (packages) bằng cách kết nối các thành phần nguồn (source), chuyển đổi (transformation) và đích (destination). Dữ liệu được trích xuất từ nguồn, chuyển đổi theo các quy tắc nghiệp vụ, sau đó tải vào đích.

**Tùy chọn cấu hình:**

* **Data sources and destinations:** Cấu hình các kết nối đến các nguồn và đích dữ liệu.
* **Transformations:** Chọn các phép biến đổi dữ liệu cần thiết (ví dụ: sắp xếp, lọc, gộp, tách).
* **Package execution options:** Cấu hình cách thức thực thi gói SSIS (ví dụ: tuần tự, song song, theo lịch trình).

**Kịch bản ứng dụng:**

* Tích hợp dữ liệu từ nhiều nguồn khác nhau vào kho dữ liệu.
* Làm sạch và chuẩn hóa dữ liệu trước khi tải vào hệ thống đích.
* Tự động hóa các quy trình xử lý dữ liệu.
  + 1. Analysis Services (SSAS) - Dịch Vụ Phân Tích

**Kiến trúc:**

SSAS là một công cụ cơ sở dữ liệu phân tích (analytical data engine) được sử dụng trong các ứng dụng hỗ trợ quyết định và phân tích kinh doanh. Nó hỗ trợ cả mô hình dữ liệu đa chiều (multidimensional) và dạng bảng (tabular).

**Cơ chế hoạt động:**

Người dùng tạo các mô hình dữ liệu trong SSAS, bao gồm các chiều (dimensions), độ đo (measures), và các phép tính (calculations). Các mô hình này được sử dụng để phân tích dữ liệu theo nhiều chiều khác nhau, giúp người dùng hiểu rõ hơn về xu hướng và các mối quan hệ trong dữ liệu.

**Tùy chọn cấu hình:**

* **Model type:** Chọn mô hình đa chiều hoặc dạng bảng.
* **Data sources:** Cấu hình kết nối đến các nguồn dữ liệu.
* **Calculations:** Xác định các phép tính và KPI (Key Performance Indicators).

**Kịch bản ứng dụng:**

* Phân tích doanh số bán hàng theo khu vực, sản phẩm, và thời gian.
* Theo dõi hiệu suất kinh doanh theo các chỉ số KPI.
* Dự báo xu hướng thị trường.
  + 1. Reporting Services (SSRS) - Dịch Vụ Báo Cáo

**Kiến trúc:**

SSRS cung cấp các công cụ và dịch vụ để tạo, triển khai, và quản lý các báo cáo phân trang (paginated reports). Nó bao gồm một cổng web (web portal) để truy cập và quản lý các báo cáo.

**Cơ chế hoạt động:**

Người dùng tạo các báo cáo SSRS bằng cách sử dụng Report Builder hoặc Report Designer. Các báo cáo này có thể kết nối đến nhiều nguồn dữ liệu khác nhau và hiển thị dữ liệu dưới dạng bảng, biểu đồ, bản đồ, v.v.

**Tùy chọn cấu hình:**

* **Data sources:** Cấu hình kết nối đến các nguồn dữ liệu.
* **Report layouts:** Thiết kế bố cục và định dạng cho báo cáo.
* **Report parameters:** Cho phép người dùng nhập các tham số để lọc dữ liệu báo cáo.

**Kịch bản ứng dụng:**

* Tạo báo cáo doanh thu hàng tháng.
* Báo cáo phân tích hiệu suất nhân viên.
* Báo cáo tổng hợp dữ liệu khảo sát khách hàng.
  + 1. Replication - Sao Chép

**Kiến trúc:**

Replication là một tập hợp các công nghệ để sao chép và phân phối dữ liệu và các đối tượng cơ sở dữ liệu từ một cơ sở dữ liệu này sang cơ sở dữ liệu khác. Nó bao gồm ba loại sao chép: transactional replication, merge replication, và snapshot replication.

**Cơ chế hoạt động:**

Replication sử dụng các agent (tác nhân) để sao chép và đồng bộ dữ liệu. Log Reader Agent đọc các thay đổi từ transaction log, Distributor Agent phân phối dữ liệu đến các Subscriber, và Merge Agent hoặc Distribution Agent thực hiện đồng bộ dữ liệu tại Subscriber.

**Tùy chọn cấu hình:**

* **Replication type:** Chọn loại sao chép phù hợp với nhu cầu.
* **Publications:** Xác định dữ liệu cần sao chép.
* **Subscriptions:** Cấu hình các máy chủ đích (subscriber) nhận dữ liệu sao chép.

**Kịch bản ứng dụng:**

* Phân phối dữ liệu đến các chi nhánh của công ty.
* Tạo các bản sao dự phòng cho mục đích khôi phục sau thảm họa.
* Sao chép dữ liệu sang các hệ thống phân tích.
  + 1. Data Quality Services (DQS) - Dịch Vụ Chất Lượng Dữ Liệu

**Kiến trúc:**

DQS cung cấp giải pháp làm sạch dữ liệu dựa trên tri thức. Nó bao gồm Data Quality Server và Data Quality Client.

**Cơ chế hoạt động:**

Người dùng tạo cơ sở tri thức (knowledge base) trong DQS, bao gồm các quy tắc và tham chiếu để xác định và sửa chữa các lỗi dữ liệu. Sau đó, họ sử dụng Data Quality Client để thực hiện các tác vụ làm sạch dữ liệu, chẳng hạn như sửa lỗi, chuẩn hóa, và loại bỏ trùng lặp.

**Tùy chọn cấu hình:**

* **Knowledge bases:** Tạo và quản lý cơ sở tri thức.
* **Data quality projects:** Xác định các nguồn dữ liệu, quy tắc làm sạch, và các tùy chọn xử lý.

**Kịch bản ứng dụng:**

* Chuẩn hóa dữ liệu địa chỉ khách hàng.
* Loại bỏ các bản ghi khách hàng trùng lặp.
* Kiểm tra tính hợp lệ của dữ liệu trước khi nhập vào hệ thống.
  + 1. Master Data Services (MDS) - Dịch Vụ Dữ Liệu Chính

**Kiến trúc:**

MDS là giải pháp quản lý dữ liệu chính (master data management) của SQL Server. Nó cho phép người dùng tạo một kho lưu trữ tập trung cho dữ liệu chính, đảm bảo tính nhất quán và chính xác của dữ liệu.

**Cơ chế hoạt động:**

Người dùng tạo các mô hình dữ liệu (models) trong MDS, bao gồm các thực thể (entities) và thuộc tính (attributes). Họ sau đó nhập dữ liệu chính vào MDS, sử dụng các quy tắc nghiệp vụ (business rules) để đảm bảo tính hợp lệ của dữ liệu.

**Tùy chọn cấu hình:**

* **Models:** Xác định cấu trúc dữ liệu chính.
* **Entities:** Tạo các bảng chứa dữ liệu chính.
* **Attributes:** Xác định các cột dữ liệu.
* **Business rules:** Áp dụng các quy tắc để kiểm tra và chuẩn hóa dữ liệu.

**Kịch bản ứng dụng:**

* Quản lý danh mục sản phẩm cho toàn bộ công ty.
* Lưu trữ thông tin khách hàng chính xác và nhất quán.
* Quản lý danh sách nhà cung cấp.

1. Tính năng và cơ chế bảo mật
   1. Tính năng
      1. Always Encrypted
         1. Tổng quan

Always Encrypted bảo vệ dữ liệu nhạy cảm (số thẻ tín dụng, số nhận dạng cá nhân) bằng cách mã hóa ngay từ ứng dụng, đảm bảo các khóa mã hóa không rời khỏi môi trường ứng dụng, cung cấp lớp bảo vệ mạnh mẽ cho cơ sở dữ liệu.

* + - 1. Cấu trúc mã hóa
* **Column Encryption Keys:** Mã hóa dữ liệu trong các cột.
* **Column Master Keys:** Bảo vệ Column Encryption Keys, lưu trữ trong các kho khóa an toàn (Azure Key Vault, HSM).
  + - 1. Lợi ích
* Bảo vệ dữ liệu từ ứng dụng.
* Cho phép dùng cùng lúc bảng mã hóa và không mã hóa.
* Sao lưu, phục hồi dữ liệu tiêu chuẩn.
  + - 1. Hạn chế
* Không hỗ trợ một số phép toán trực tiếp trên dữ liệu mã hóa.
* Lệnh FOR XML, FOR JSON không hỗ trợ cột mã hóa.
* Chỉ số và một số tính năng không hoạt động với cột mã hóa.
  + 1. In-Memory OLTP
       1. Tổng quan

In-Memory OLTP là tính năng tối ưu hóa hiệu suất giao dịch bằng cách lưu trữ dữ liệu trong bộ nhớ thay vì đĩa, giảm độ trễ và cải thiện tốc độ xử lý.

* + - 1. Lợi ích chính
* **Hiệu suất cao**: Tăng tốc truy cập và xử lý, giảm tắc nghẽn do khóa và latch.
* **Khả năng phục hồi**: Hỗ trợ sao lưu và phục hồi tiêu chuẩn.
* **Tương thích**: Cho phép sử dụng cả bảng bộ nhớ và bảng đĩa trong cùng một cơ sở dữ liệu.
  + - 1. Kịch bản sử dụng
* **Xử lý giao dịch có thông lượng cao:** Thích hợp cho các lĩnh vực yêu cầu phản hồi nhanh và khối lượng giao dịch lớn (tài chính, cá cược).
* **Quản lý bảng tối ưu hóa bộ nhớ:** Lưu trữ dữ liệu quan trọng cần truy cập nhanh.
  + - 1. Hạn chế
* **Không phù hợp với mọi công việc**: Không lý tưởng cho truy vấn phức tạp hoặc xử lý dữ liệu lớn.
* **Sao lưu và phục hồi**: Có thể gặp sự cố với cơ sở dữ liệu dùng bảng tối ưu hóa bộ nhớ.
* **Hạn chế tính năng**: Một số tính năng SQL Server không hoạt động với bảng tối ưu hóa bộ nhớ.
  + 1. Temporal Tables
       1. Tổng quan

Temporal Tables được giới thiệu trong SQL Server 2016, cho phép tự động lưu trữ và quản lý lịch sử dữ liệu, giúp theo dõi thay đổi của dữ liệu theo thời gian mà không cần tạo bảng lưu trữ riêng.

* + - 1. Lợi ích chính
* **Theo dõi lịch sử**: Tự động lưu trữ các phiên bản dữ liệu trước đó, dễ dàng truy xuất thông tin quá khứ.
* **Quản lý đơn giản**: Không cần viết mã để lưu trữ lịch sử, SQL Server tự động xử lý.
* **Truy vấn linh hoạt**: Dễ dàng truy vấn dữ liệu hiện tại và lịch sử bằng cú pháp SQL.
  + - 1. Kịch bản sử dụng
* **Quản lý thay đổi dữ liệu:** Phù hợp với các hệ thống cần theo dõi thay đổi dữ liệu (tài chính, nhân sự, quản lý thông tin).
* **Phân tích dữ liệu theo thời gian:** Hỗ trợ phân tích xu hướng và biến đổi dữ liệu qua thời gian.
  + - 1. Hạn chế
* **Kích thước dữ liệu:** Lưu trữ lịch sử có thể làm tăng nhanh kích thước cơ sở dữ liệu.
* **Hạn chế kiểu dữ liệu:** Một số kiểu dữ liệu không hoạt động với Temporal Tables.
* **Yêu cầu phiên bản:** Chỉ hỗ trợ trên SQL Server 2016 trở lên.
  + 1. Stretch Database
       1. Tổng quan

Stretch Database cho phép mở rộng cơ sở dữ liệu lên đám mây Azure, giúp lưu trữ dữ liệu ít truy cập trên đám mây và giữ dữ liệu thường xuyên truy cập tại chỗ, từ đó tiết kiệm chi phí và cải thiện hiệu suất.

* + - 1. Lợi ích chính
* **Tiết kiệm chi phí:** Lưu trữ dữ liệu ít sử dụng trên đám mây giảm chi phí lưu trữ nội bộ.
* **Tăng cường hiệu suất:** Cơ sở dữ liệu nhỏ hơn giúp truy vấn nhanh hơn.
* **Dễ dàng truy cập:** Dữ liệu trên Azure có thể truy cập như dữ liệu tại chỗ mà không cần thay đổi ứng dụng.
  + - 1. Kịch bản sử dụng
* **Lưu trữ lâu dài**: Phù hợp cho lưu trữ dữ liệu ít truy cập hoặc dữ liệu lịch sử.
* **Phân tích dữ liệu**: Dữ liệu có thể chuyển sang Azure để phân tích mà không ảnh hưởng đến hiệu suất của ứng dụng chính.
  + - 1. Hạn chế
* **Kết nối Internet**: Cần kết nối Internet ổn định để truy cập dữ liệu trên Azure.
* **Hiệu suất**: Truy cập dữ liệu từ đám mây có thể chậm hơn so với dữ liệu tại chỗ, đặc biệt với nhiều truy vấn đồng thời.
* **Ngừng hỗ trợ**: Tính năng này không còn hỗ trợ trong SQL Server 2022.
  + 1. Row-Level Security
       1. Tổng quan

Row-Level Security (RLS) là tính năng trong SQL Server cho phép quản lý quyền truy cập dữ liệu ở cấp độ dòng trong bảng, đảm bảo người dùng chỉ truy cập các dòng dữ liệu mà họ có quyền xem, từ đó nâng cao bảo mật.

* + - 1. Lợi ích chính
* **Bảo mật nâng cao**: Kiểm soát chi tiết quyền truy cập vào từng dòng dữ liệu, ngăn ngừa truy cập trái phép.
* **Quản lý quyền truy cập đơn giản**: Giảm bớt việc tạo nhiều bản sao bảng và sử dụng phương pháp phức tạp.
* **Dễ dàng tích hợp**: Có thể tích hợp vào ứng dụng hiện có mà không cần thay đổi logic nhiều.
  + - 1. Kịch bản sử dụng
* **Quản lý dữ liệu nhạy cảm:** Thích hợp cho các ứng dụng yêu cầu bảo mật cao (tài chính, y tế, thông tin nhạy cảm).
* **Phân quyền cho người dùng:** Sử dụng RLS để phân chia quyền truy cập cho các nhóm người dùng khác nhau.
  + - 1. Hạn chế
* **Ảnh hưởng đến hiệu suất:** RLS có thể làm chậm truy vấn, cần cân nhắc khi thiết kế.
* **Khó khăn trong gỡ lỗi**: RLS có thể làm phức tạp việc gỡ lỗi và kiểm tra dữ liệu.
* **Không hỗ trợ một số lệnh SQL:** Một số lệnh và thao tác có thể không hoạt động chính xác khi RLS được áp dụng.
  + 1. Dynamic Data Masking
       1. Tổng quan

Dynamic Data Masking (DDM) là tính năng trong SQL Server cho phép che giấu dữ liệu nhạy cảm trong các kết quả truy vấn, bảo vệ thông tin mà không ảnh hưởng đến ứng dụng hay yêu cầu thay đổi cấu trúc cơ sở dữ liệu. Tính năng này cải thiện bảo mật bằng cách chỉ cho phép người dùng truy cập dữ liệu rõ ràng mà họ có quyền.

* + - 1. Lợi ích chính
* **Bảo vệ dữ liệu nhạy cảm:** Che giấu thông tin nhạy cảm như số thẻ tín dụng, số an sinh xã hội mà không thay đổi dữ liệu trong cơ sở dữ liệu.
* **Đơn giản hóa bảo mật:** Dễ dàng triển khai mà không cần thay đổi ứng dụng hoặc cấu trúc dữ liệu.
* **Quản lý quyền truy cập linh hoạt:** Quản trị viên có thể kiểm soát cách thức dữ liệu hiển thị cho người dùng khác nhau.
  + - 1. Kịch bản sử dụng
* **Ứng dụng yêu cầu bảo mật cao**: Phù hợp cho các ứng dụng cần bảo vệ dữ liệu nhạy cảm nhưng vẫn yêu cầu truy cập nhanh chóng và dễ dàng.
* **Môi trường phát triển và kiểm thử**: Bảo vệ dữ liệu nhạy cảm trong các môi trường phát triển và kiểm thử mà không tiết lộ thông tin thật.
  + - 1. Hạn chế
* **Không che giấu dữ liệu trong mọi truy vấn**: DDM chỉ hoạt động trên các truy vấn sử dụng các cột có mặt nạ; nếu không tham chiếu cột này, dữ liệu sẽ không bị che giấu.
* **Tương thích với các ứng dụng**: Một số ứng dụng có thể không tương thích tốt với DDM, cần xem xét kỹ trước khi triển khai.
* **Khả năng truy cập có thể bị giới hạn**: Người dùng có thể bị hạn chế trong việc truy cập dữ liệu thực tế cần thiết cho một số tác vụ.
  + 1. Hỗ trợ JSON
       1. Tổng quan

Tính năng hỗ trợ JSON trong SQL Server cho phép người dùng lưu trữ, truy vấn và thao tác với dữ liệu định dạng JSON một cách dễ dàng. Tính năng này giúp tích hợp dữ liệu JSON vào cơ sở dữ liệu quan hệ mà không cần chuyển đổi sang cấu trúc bảng truyền thống, đồng thời cung cấp các chức năng để xử lý và truy vấn dữ liệu JSON trực tiếp.

* + - 1. Lợi ích chính
* **Tích hợp dễ dàng**: Làm việc với dữ liệu JSON mà không cần thay đổi cấu trúc cơ sở dữ liệu hiện có.
* **Truy vấn linh hoạt**: Cung cấp các hàm và toán tử để truy vấn dữ liệu JSON hiệu quả.
* **Tối ưu hóa hiệu suất**: Cho phép lưu trữ dữ liệu JSON cùng với dữ liệu quan hệ, cải thiện hiệu suất truy vấn và giảm thiểu độ phức tạp.
  + - 1. Kịch bản sử dụng
* **Ứng dụng web và dịch vụ API:** Hỗ trợ tích hợp dữ liệu từ các dịch vụ web và API JSON một cách dễ dàng.
* **Quản lý dữ liệu không cấu trúc:** Thích hợp cho các ứng dụng cần lưu trữ và quản lý dữ liệu không có cấu trúc hoặc dữ liệu bán cấu trúc.
  + - 1. Hạn chế
* **Kiểu dữ liệu và hiệu suất:** Dữ liệu JSON được lưu trữ dưới dạng văn bản, có thể ảnh hưởng đến hiệu suất truy vấn nếu dữ liệu lớn hoặc phức tạp.
* **Khả năng kiểm tra dữ liệu:** Dữ liệu JSON không có ràng buộc kiểu dữ liệu như các bảng quan hệ, điều này có thể dẫn đến khó khăn trong việc kiểm tra tính hợp lệ của dữ liệu.
* **Thiếu hỗ trợ cho một số tính năng SQL:** Một số tính năng của SQL Server có thể không hoạt động tốt khi làm việc với dữ liệu JSON.
  + 1. PolyBase
       1. Tổng quan

PolyBase là một tính năng trong SQL Server cho phép người dùng truy cập và xử lý dữ liệu từ các nguồn ngoài như Hadoop, Azure Blob Storage và các cơ sở dữ liệu khác mà không cần phải di chuyển dữ liệu vào cơ sở dữ liệu SQL Server. Tính năng này cung cấp khả năng tích hợp dữ liệu đa dạng, giúp thực hiện các truy vấn phân tán và phân tích dữ liệu hiệu quả hơn.

* + - 1. Lợi ích chính
* **Truy cập dữ liệu đa nguồn:** Cho phép truy cập dữ liệu từ nhiều nguồn khác nhau mà không cần di chuyển dữ liệu, giúp tiết kiệm thời gian và tài nguyên.
* **Tích hợp dễ dàng:** Cung cấp giao diện đơn giản để tích hợp dữ liệu từ các nền tảng Big Data như Hadoop và Azure.
* **Tối ưu hóa hiệu suất:** Hỗ trợ các truy vấn phân tán hiệu quả, giúp cải thiện hiệu suất của các tác vụ phân tích dữ liệu.
  + - 1. Kịch bản sử dụng
* **Phân tích dữ liệu lớn**: Thích hợp cho các tổ chức cần phân tích và xử lý dữ liệu lớn từ nhiều nguồn khác nhau.
* **Quản lý dữ liệu phân tán**: Sử dụng trong các môi trường hybrid nơi dữ liệu có thể nằm rải rác ở nhiều vị trí và hệ thống khác nhau.
  + - 1. Hạn chế
* **Yêu cầu tài nguyên**: Việc truy cập và xử lý dữ liệu từ các nguồn ngoài có thể yêu cầu tài nguyên mạng và tính toán cao.
* **Hỗ trợ một số định dạng dữ liệu**: PolyBase hỗ trợ một số định dạng dữ liệu cụ thể; người dùng cần kiểm tra tính tương thích với các nguồn dữ liệu.
* **Độ phức tạp trong quản lý**: Việc tích hợp dữ liệu từ nhiều nguồn có thể dẫn đến độ phức tạp trong việc quản lý và bảo trì.
  + 1. Query Store
       1. Tổng quan

Query Store là một tính năng trong SQL Server 2016 cho phép theo dõi và lưu trữ thông tin về các truy vấn và hiệu suất của chúng. Tính năng này giúp quản trị viên cơ sở dữ liệu (DBA) phân tích các vấn đề về hiệu suất và thực hiện các điều chỉnh cần thiết để tối ưu hóa hiệu suất truy vấn.

* + - 1. Lợi ích chính
* **Giám sát hiệu suất**: Query Store lưu trữ thông tin về truy vấn, bao gồm thời gian thực hiện, số lần gọi và các kế hoạch thực hiện, giúp người dùng theo dõi hiệu suất một cách dễ dàng.
* **Khôi phục kế hoạch**: Khi có sự thay đổi trong kế hoạch thực hiện truy vấn, Query Store cho phép khôi phục kế hoạch trước đó nếu cần, giúp duy trì hiệu suất ổn định.
* **Phân tích lịch sử**: Cung cấp khả năng phân tích lịch sử truy vấn, cho phép người dùng phát hiện các vấn đề hiệu suất qua thời gian.
  + - 1. Kịch bản sử dụng
* **Tối ưu hóa truy vấn:** Thích hợp cho các DBA và nhà phát triển muốn tối ưu hóa truy vấn và cải thiện hiệu suất của cơ sở dữ liệu.
* **Giám sát thay đổi:** Sử dụng Query Store để theo dõi các thay đổi trong truy vấn và kế hoạch thực hiện, giúp xác định những yếu tố ảnh hưởng đến hiệu suất.
  + - 1. Hạn chế
* **Tài nguyên sử dụng:** Query Store có thể sử dụng tài nguyên hệ thống, vì vậy cần theo dõi việc sử dụng tài nguyên để tránh ảnh hưởng đến hiệu suất.
* **Khả năng lưu trữ:** Dữ liệu trong Query Store có thể bị giới hạn về dung lượng lưu trữ, và người dùng cần quản lý để không vượt quá giới hạn.
* **Tính tương thích:** Một số tính năng có thể không tương thích với tất cả các phiên bản SQL Server, cần kiểm tra trước khi triển khai.
  + 1. Live Query Statistics
       1. Tổng quan

Live Query Statistics là một tính năng trong SQL Server 2016 cho phép người dùng theo dõi và phân tích hiệu suất của các truy vấn SQL trong thời gian thực. Tính năng này cung cấp cái nhìn sâu sắc về cách mà truy vấn được thực hiện, từ đó hỗ trợ người dùng tối ưu hóa hiệu suất và khắc phục các vấn đề phát sinh.

* + - 1. Lợi ích chính
* **Theo dõi hiệu suất tức thì:** Người dùng có thể xem các thông tin về hiệu suất truy vấn ngay khi chúng đang thực thi, giúp nhận diện vấn đề kịp thời.
* **Tối ưu hóa truy vấn:** Dữ liệu trực quan hóa cho phép người dùng phân tích và tối ưu hóa các truy vấn, từ đó nâng cao hiệu suất cơ sở dữ liệu.
* **Giảm thiểu thời gian phân tích:** Với khả năng theo dõi trực tiếp, người dùng có thể nhanh chóng phát hiện các điểm nghẽn trong quy trình thực hiện truy vấn.
  + - 1. Kịch bản sử dụng
* **Phân tích hiệu suất truy vấn**: Tính năng này thường được sử dụng trong các tình huống cần phân tích và tối ưu hóa hiệu suất truy vấn phức tạp.
* **Giám sát hệ thống**: Live Query Statistics có thể giúp các quản trị viên cơ sở dữ liệu theo dõi hoạt động của hệ thống và phát hiện các vấn đề ngay khi chúng phát sinh.
  + - 1. Hạn chế
* **Tác động đến hiệu suất:** Việc theo dõi trực tiếp có thể gây ảnh hưởng đến hiệu suất của các truy vấn đang chạy, nên cần sử dụng một cách thận trọng.
* **Không hỗ trợ mọi truy vấn:** Một số loại truy vấn hoặc tình huống không thể sử dụng Live Query Statistics, cần kiểm tra tính tương thích trước khi triển khai.
* **Yêu cầu về tài nguyên:** Tính năng này có thể tiêu tốn tài nguyên hệ thống, do đó cần theo dõi hiệu suất tổng thể của cơ sở dữ liệu.
  + 1. Master Data Services
       1. Tổng quan

Master Data Services (MDS) là một giải pháp quản lý dữ liệu chính trong SQL Server 2016, cho phép tổ chức quản lý và duy trì dữ liệu chính (master data) một cách hiệu quả. Tính năng này cung cấp khả năng tạo, lưu trữ và quản lý dữ liệu chính trong một môi trường an toàn và có thể mở rộng.

* + - 1. Lợi ích chính
* **Quản lý dữ liệu chính**: MDS giúp đảm bảo rằng dữ liệu chính trong tổ chức được đồng bộ hóa và chính xác, giảm thiểu các sai sót và xung đột dữ liệu.
* **Tăng cường tính chính xác**: Với khả năng kiểm tra và phê duyệt dữ liệu, MDS giúp cải thiện độ tin cậy và chính xác của dữ liệu.
* **Tích hợp dễ dàng**: Hỗ trợ tích hợp dữ liệu từ nhiều nguồn khác nhau, cho phép tổ chức có cái nhìn tổng thể về dữ liệu chính.
  + - 1. Kịch bản sử dụng
* **Quản lý dữ liệu khách hàng**: Thích hợp cho các tổ chức cần theo dõi và quản lý thông tin khách hàng một cách nhất quán.
* **Quản lý sản phẩm:** Sử dụng MDS để quản lý thông tin về sản phẩm, đảm bảo rằng dữ liệu sản phẩm được cập nhật và chính xác.
  + - 1. Hạn chế
* **Khó khăn trong triển khai**: Việc thiết lập và cấu hình MDS có thể phức tạp, đòi hỏi kiến thức chuyên sâu về quản lý dữ liệu.
* **Yêu cầu về tài nguyên**: MDS có thể tiêu tốn tài nguyên hệ thống, do đó cần theo dõi hiệu suất để đảm bảo hoạt động ổn định.
* **Đào tạo người dùng**: Người dùng cần được đào tạo để sử dụng MDS một cách hiệu quả, tránh những sai sót trong quá trình nhập liệu và quản lý dữ liệu.
  + 1. Data Quality Services
       1. Tổng quan

Data Quality Services (DQS) là một giải pháp trong SQL Server 2016 giúp tổ chức đảm bảo chất lượng dữ liệu thông qua các công cụ và quy trình kiểm tra, làm sạch, và duy trì dữ liệu. DQS cung cấp khả năng phát hiện và khắc phục các vấn đề về dữ liệu, từ đó nâng cao độ chính xác và tin cậy của dữ liệu.

* + - 1. Lợi ích chính
* **Cải thiện chất lượng dữ liệu:** DQS giúp phát hiện và loại bỏ dữ liệu không chính xác, trùng lặp, và không đầy đủ, từ đó nâng cao chất lượng dữ liệu tổng thể.
* **Dễ sử dụng:** Giao diện trực quan cho phép người dùng dễ dàng thực hiện các tác vụ kiểm tra và làm sạch dữ liệu mà không cần kiến thức lập trình.
* **Tích hợp với SQL Server:** DQS có thể dễ dàng tích hợp vào các quy trình ETL và hệ thống dữ liệu khác trong SQL Server.
  + - 1. Kịch bản sử dụng
* **Làm sạch dữ liệu khách hàng**: DQS thường được sử dụng để làm sạch và chuẩn hóa thông tin khách hàng, giúp tổ chức quản lý thông tin một cách hiệu quả.
* **Quản lý dữ liệu sản phẩm**: DQS có thể giúp tổ chức đảm bảo rằng thông tin về sản phẩm được cập nhật và chính xác.
  + - 1. Hạn chế
* **Yêu cầu tài nguyên:** Quá trình làm sạch dữ liệu có thể tiêu tốn tài nguyên, cần theo dõi hiệu suất để đảm bảo hoạt động ổn định.
* **Khó khăn trong quy trình:** Việc thiết lập quy trình làm sạch và kiểm tra dữ liệu có thể phức tạp, đòi hỏi kiến thức chuyên sâu về dữ liệu.
* **Đào tạo người dùng:** Người dùng cần được đào tạo để sử dụng DQS hiệu quả, tránh những sai sót trong quá trình làm sạch dữ liệu.
  1. Cơ chế bảo mật
     1. Chế độ xác thực
        1. Windows Authentication

Chế độ xác thực Windows Authentication cho phép SQL Server sẽ xác thực tài khoản của người dùng bằng mã thông báo chính của tài khoản người dùng Windows. Nghĩa là SQL Server không yêu cầu mật khẩu và danh tính người dùng được Windows xác nhận. Xác thực Windows sử dụng Trình quản lý LAN Công nghệ mới (NTLM) hoặc giao thức bảo mật Kerberos, cung cấp khả năng thực thi chính sách mật khẩu như:

* Xác thực độ phức tạp cho mật khẩu mạnh
* Cung cấp hỗ trợ khóa tài khoản
* Mật khẩu có thời hạn

**Ưu điểm:**

* An toàn hơn vì sử dụng cơ chế bảo mật của Windows.
* Hỗ trợ Single Sign-On (SSO), cho phép người dùng đăng nhập một lần và truy cập nhiều ứng dụng mà không cần nhập lại thông tin xác thực.

**Nhược điểm:**

* Phức tạp hơn trong việc thiết lập và quản lý, đặc biệt cho các môi trường không sử dụng Active Directory.
  + - 1. SQL Server Authentication

Chế độ xác thực SQL Server sử dụng các thông tin đăng nhập được tạo và lưu trữ trong SQL Server mà không dựa trên tài khoản Windows của người dùng. Người dùng kết nối bằng Xác thực SQL Server phải cung cấp thông tin xác thực của họ (tên đăng nhập và mật khẩu) mỗi khi kết nối. Chế độ xác thực SQL Server yêu cầu người dùng phải đặt mật khẩu mạnh cho tất cả các tài khoản SQL Server.

Chế độ xác thực SQL Server cung cấp ba chính sách mật khẩu mà người dùng có thể tùy chọn:

* Người dùng phải thay đổi mật khẩu khi đăng nhập lần sau
* Mật khẩu có thời hạn
* Áp dụng chính sách mật khẩu của tài khoản Windows cho thông tin đăng nhập SQL Server

**Ưu điểm:**

* Đơn giản cho các ứng dụng độc lập không sử dụng Active Directory.
* Có thể tạo tài khoản người dùng riêng biệt cho từng ứng dụng.

**Nhược điểm:**

* Mật khẩu và thông tin đăng nhập có thể dễ bị tấn công nếu không được bảo vệ đúng cách.
* Không tích hợp với các chính sách bảo mật của Windows.
  + - 1. Mixed authetication (chế độ hỗn hợp của hai loại trên)
    - Microsoft SQL Server chỉ hỗ trợ 2 phương pháp xác thực là Windows authetication và Mixed authetication (Windows Authentication luôn khả dụng và không thể tắt).
    1. Quản lý quyền

Mỗi tài khoản người dùng được cấp quyền truy nhập thông qua việc gán vào một hoặc một số nhóm vai trò (role). Trong đó, các vai trò được chia thành hai nhóm lớn: *vai trò cấp máy chủ* và *vai trò cấp cơ sở dữ liệu*.

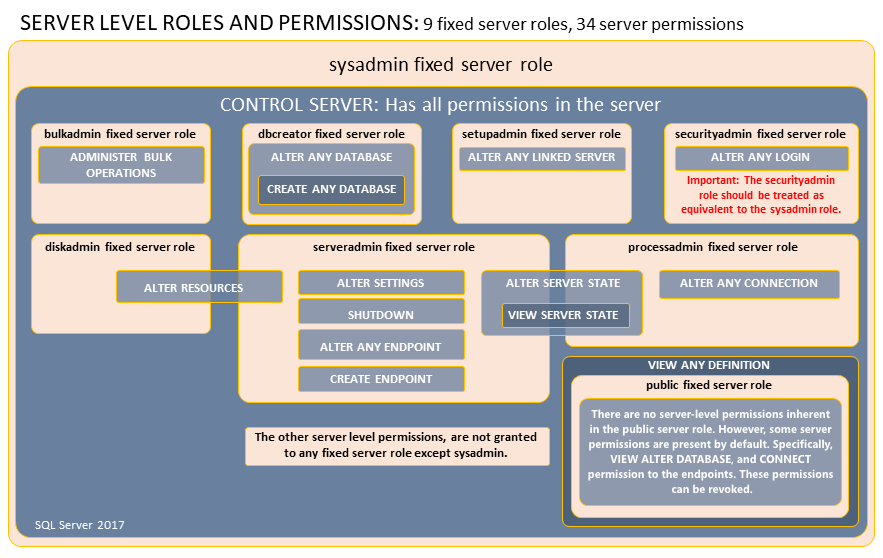
* + - 1. Vai trò cấp máy chủ (Server-level role)

SQL Server cung cấp các vai trò cấp máy chủ để giúp người quản trị quản lý các quyền trên máy chủ. Các vai trò này là các nguyên tắc bảo mật nhóm các nguyên tắc khác. Các vai trò cấp máy chủ có phạm vi quyền trên toàn máy chủ.

Bảng sau hiển thị các vai trò máy chủ cố định và khả năng của chúng:

|  |  |
| --- | --- |
| **Vai trò máy chủ cố định** | **Miêu tả** |
| sysadmin | Các thành viên mang vai trò này có thể thực hiện bất kỳ hoạt động nào trên máy chủ. |
| serveradmin | Các thành viên mang vai trò này có thể thay đổi tùy chọn cấu hình toàn máy chủ và tắt máy chủ. |
| securityadmin | Các thành viên mang vai trò này có thể:   * Quản lý thông tin đăng nhập và thuộc tính của chúng. * Trao, từ chối và thu hồi các quyền cấp máy chủ, quyền cấp cơ sở dữ liệu (nếu có quyền truy cập vào cơ sở dữ liệu). * Đặt lại mật khẩu cho thông tin đăng nhập SQL Server   **Quan trọng:** Vai trò này được coi tương đương với vai trò **sysadmin**. |
| processadmin | Các thành viên mang vai trò này có thể kết thúc các tiến trình đang chạy trong phiên bản SQL Server. |
| setupadmin | Các thành viên mang vai trò này có thể thêm và xóa các máy chủ liên kết. |
| bulkadmin | Vai trò này có khả năng thực hiện các hoạt động liên quan đến việc nhập dữ liệu hàng loạt (bulk operations). |
| diskadmin | Vai trò này được sử dụng để quản lý các tệp đĩa. |
| dbcreator | Các thành viên mang vai trò này có thể tạo, thay đổ, xóa và khôi phục bất kỳ cơ sở dữ liệu nào. |
| public | Mỗi lần đăng nhập SQL Server đều thuộc về vai trò máy chủ **public** . Khi một chủ thể máy chủ chưa được cấp hoặc từ chối các quyền cụ thể trên một đối tượng có thể bảo mật, người dùng sẽ thừa hưởng các quyền được cấp cho **public** trên đối tượng đó. |





1. Hình ảnh minh họa các quyền được gán cho các vai trò máy chủ (SQL Server 2019 trở về trước)
   * + 1. Vai trò cấp cơ sở dữ liệu (Database-level role)

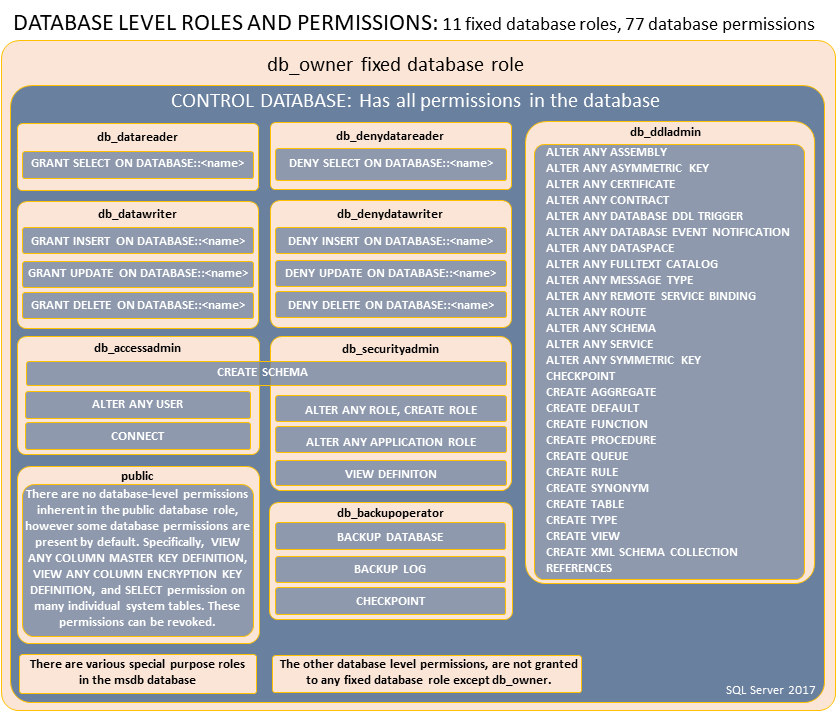
Các vai trò cấp cơ sở dữ liệu có phạm vi quyền trên toàn cơ sở dữ liệu. Các vai trò cấp cơ sở dữ liệu được chia ra làm ba loại:

* Vai trò cơ sở dữ liệu cố định (fixed database role)
* Vai trò cơ sở dữ liệu do người dùng xác định (user-defined database role)
* Vai trò cơ sở dữ liệu công cộng (public database role)
  + - 1. Vai trò cơ sở dữ liệu cố định

Các *vai trò cơ sở dữ liệu cố định* được định nghĩa ở cấp cơ sở dữ liệu và tồn tại trong mỗi cơ sở dữ liệu. Các thành viên mang vai trò **db\_owner** có thể quản lý các thành viên mang *vai trò cơ sở dữ liệu cố định* khác. Quyền được gán cho các *vai trò cơ sở dữ liệu cố định* không thể thay đổi.

|  |  |
| --- | --- |
| **Vai trò cơ sở dữ liệu cố định** | **Miêu tả** |
| db\_owner | Các thành viên mang vai trò này có thể thực hiện mọi hoạt động cấu hình và bảo trì trên cơ sở dữ liệu và cũng có thể thực hiện xóa bỏ cơ sở dữ liệu trong SQL Server. |
| db\_securityadmin | Các thành viên mang vai trò này có thể sửa đổi vai trò của thành viên đối với các vai trò tùy chỉnh và quản lý quyền của chúng.  **Chú ý:** Thành viên mang vai trò này có thể nâng cao đặc quyền của họ nên cần được theo dõi |
| db\_accessadmin | Các thành viên mang vai trò này có thể thêm  hoặc xóa quyền truy cập vào cơ sở dữ liệu cho thông tin đăng nhập Windows, nhóm Windows và thông tin đăng nhập SQL Server. |
| db\_backupoperator | Các thành viên mang vai trò này có thể sao lưu cơ sở dữ liệu. |
| db\_ddladmin | Các thành viên mang vai trò này có thể chạy bất kỳ lệnh Data Definition Language (DDL) trong cơ sở dữ liệu. |
| db\_datawriter | Các thành viên mang vai trò này có thể thêm, xóa hoặc thay đổi dữ liệu trong tất cả các bảng người dùng. |
| db\_datareader | Các thành viên mang vai trò này có thể đọc tất cả dữ liệu từ tất cả các bảng và chế độ xem của người dùng. |
| db\_denydatawriter | Các thành viên mang vai trò này không thể thêm, sửa đổi hoặc xóa bất kỳ dữ liệu nào trong các bảng người dùng trong cơ sở dữ liệu. |
| db\_denydatareader | Các thành viên mang vai trò này không thể đọc bất kỳ dữ liệu nào từ các bảng và chế độ xem của người dùng trong cơ sở dữ liệu. |





1. Các quyền được gán cho các vai trò cơ sở dữ liệu cố định
   * + 1. Vai trò cơ sở dữ liệu do người dùng xác định

*Vai trò cơ sở dữ liệu do người dùng xác định* cho phép người dùng tạo ra các vai trò tùy chỉnh để quản lý quyền truy cập và phân quyền cho người dùng theo nhu cầu cụ thể của tổ chức thông qua các câu lệnh SQL: **GRANT**, **DENY** và **REVOKE.**

* + - 1. Vai trò cơ sở dữ liệu công cộng

Mỗi người dùng cơ sở dữ liệu đều mang *vai trò cơ sở dữ liệu công cộng*. Khi người dùng chưa được cấp hoặc bị từ chối các quyền cụ thể trên một đối tượng bảo mật, người dùng sẽ kế thừa các quyền được cấp cho *vai trò cơ sở dữ liệu công cộng* trên đối tượng đó. Người dùng cơ sở dữ liệu không thể bị xóa khỏi *vai trò cơ sở dữ liệu công cộng*.

* + 1. Mã hóa

SQL Server sử dụng một cơ chế mã hóa dữ liệu tiên tiến dựa trên cơ sở hạ tầng quản lý khóa và mã hóa phân cấp để bảo vệ dữ liệu. Cơ chế này đảm bảo sự an toàn thông qua các lớp mã hóa được thiết kế chặt chẽ, sử dụng kết hợp chứng chỉ, khóa đối xứng và khóa bất đối xứng.

SQL Server sử dụng cơ chế mã hóa phân cấp kết hợp với Mô-đun Quản lý Khóa Mở rộng (EKM) để đảm bảo bảo mật dữ liệu. Dưới đây là các nội dung chính:

**Lưu trữ và quản lý khóa:**

* + Khóa bất đối xứng và đối xứng có thể được lưu trữ bên ngoài SQL Server thông qua EKM, tăng cường bảo mật và tách biệt quản lý.
  + EKM có thể bảo vệ hoặc thay thế các khóa lưu trữ nội bộ của SQL Server.

**Hiệu suất mã hóa:**

* + Để đạt hiệu suất tối ưu, nên sử dụng khóa đối xứng thay vì chứng chỉ hoặc khóa bất đối xứng để mã hóa dữ liệu.

**Hệ thống phân cấp mã hóa:**

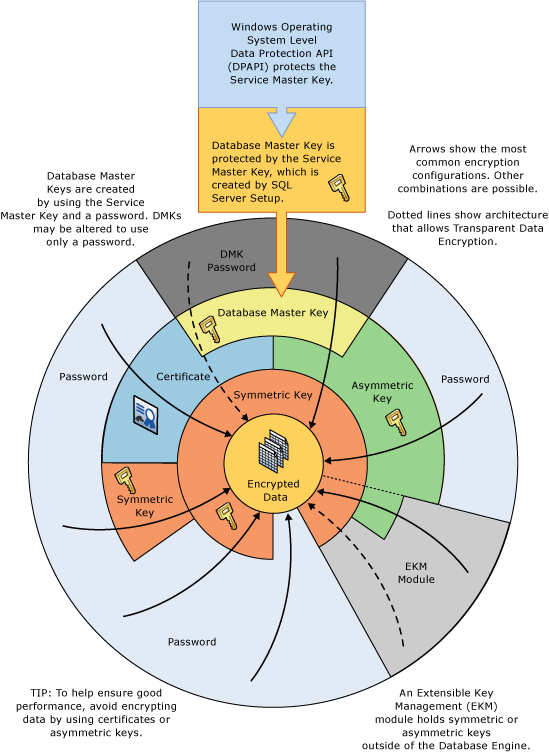
* + Khóa chính dịch vụ (Service Master Key - SMK) bảo vệ khóa chính cơ sở dữ liệu (Database Master Key - DMK). SMK được mã hóa bằng API Bảo vệ Dữ liệu Windows (DPAPI).
  + Cả SMK và DMK đều là khóa đối xứng.

**Mã hóa dữ liệu trong suốt (TDE):**

* + TDE sử dụng khóa đối xứng đặc biệt gọi là Khóa Mã hóa Cơ sở dữ liệu (Database Encryption Key - DEK).
  + DEK được bảo vệ bằng chứng chỉ hoặc khóa bất đối xứng lưu trong EKM.

**Khả năng phân cấp mở rộng:**

* + SQL Server hỗ trợ xếp chồng nhiều lớp hệ thống mã hóa, cho phép linh hoạt trong quản lý và tăng cường bảo mật dữ liệu.



2. DEMO
   1. Tạo cơ sở dữ liệu

* Tạo một cơ sở dữ liệu mới có tên *Store*.



1. Câu lệnh SQL tạo cơ sở dữ liệu mới
   1. Thêm bảng

* *USE Store*: Chuyển sang cơ sở dữ liệu vừa tạo (ở phần 1)
* Tạo bảng mới *Customers* với các cột:
  + Cột *customer\_id*: Khóa chính, tự động tăng, kiểu dữ liệu INT.
  + Cột *customer\_name* và cột *phone*: Chuỗi bắt buộc (không để trống).
  + Cột *email* và cột *address*: Chuỗi tùy chọn.

A computer screen shot of a computer code

Description automatically generated

1. Câu lệnh SQL tạo bảng mới
   1. Xóa bảng

* Xóa bảng có tên *Customers.*

A close-up of a white background

Description automatically generated

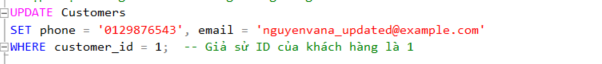
1. Câu lệnh SQL xóa bảng và dữ liệu bên trong bảng
   1. Thêm, sửa, xóa và xem dữ liệu

* Thêm dữ liệu vào bảng *Customers*:
  + *customer\_id*: giá trị này tự động tăng
  + *customer\_name*: 'Pham Thi D'
  + *phone*: '0981234567'
  + *email*: 'phamthid@example.com'
  + *address*: '123 Tran Phu, Da Nang'



1. Câu lệnh SQL thêm dữ liệu vào bảng

* Cập nhật thông tin của khách hàng có *customer\_id* = 1 trong bảng *Customers* với các thay đổi:
  + *phone*: được cập nhật thành '0129876543'.
  + *email*: được cập nhật thành 'nguyenvana\_updated@example.com'



1. Câu lệnh SQL sửa dữ liệu trong bảng

* Xóa dữ liệu khách hàng có *customer\_id* = 3 trong bảng *Customers.*



1. Câu lệnh SQL xóa dữ liệu trong bảng

* Xem tất cả dữ liệu danh sách khách hàng trong bảng *Customers.*



1. Câu lệnh SQL xem dữ liệu trong bảng
   1. Tạo sao lưu cơ sở dữ liệu

* Tạo bản sao lưu toàn bộ cơ sở dữ liệu *Store* file *Store.bak* tại đường dẫn *C:\Backup*



1. Câu lệnh SQL tạo bản sao lưu cơ sở dữ liệu

* Phục hồi cơ sở dữ liệu *Store* từ file sao lưu *Store.bak* tại *C:\Backup*, ghi đè nếu cơ sở dữ liệu đã tồn tại.

A close up of a computer screen

Description automatically generated

1. Câu lệnh SQL khôi phục cơ sở dữ liệu từ một bản sao lưu