

1. Khái niệm về DBMS

Hệ thống quản lý cơ sở dữ liệu (DBMS) là phần mềm giúp quản lý và tổ chức dữ liệu một cách hiệu quả. DBMS cung cấp các tính năng quan trọng như:

- Truy cập dữ liệu nhanh chóng và hiệu quả
- Tính độc lập dữ liệu
- Đảm bảo tính toàn vẹn dữ liệu và bảo mật
- Hỗ trợ truy cập đồng thời
- Khả năng khôi phục dữ liệu khi gặp sự cố

2. Lợi ích của DBMS so với lưu trữ tệp thông thường

So với việc lưu trữ dữ liệu trực tiếp trong tệp hệ điều hành, sử dụng DBMS mang lại nhiều lợi ích như:

- Tính độc lập dữ liệu
- Truy cập dữ liệu hiệu quả
- Đảm bảo tự động tính toàn vẹn và bảo mật dữ liệu

3. Mô hình dữ liệu và lược đồ trong DBMS

Dữ liệu trong DBMS được tổ chức theo một **mô hình dữ liệu**, trong đó phần mô tả dữ liệu được gọi là **lược đồ (schema)**.

- **Mô hình quan hệ** hiện là mô hình phổ biến nhất.
- DBMS cung cấp ba cấp độ trừu tượng hóa dữ liệu:
 - **Lược đồ bên ngoài:** Cách người dùng nhìn thấy dữ liệu
 - **Lược đồ khái niệm:** Cấu trúc logic của toàn bộ cơ sở dữ liệu
 - **Lược đồ vật lý:** Cách dữ liệu được lưu trữ thực tế trên bộ nhớ

Nhờ ba cấp độ này, DBMS đảm bảo **tính độc lập dữ liệu vật lý và logic**, giúp người dùng không bị ảnh hưởng bởi thay đổi trong cách tổ chức dữ liệu bên trong hệ thống.

4. Truy vấn và thao tác dữ liệu

DBMS cung cấp **ngôn ngữ truy vấn** và **ngôn ngữ thao tác dữ liệu**, giúp người dùng dễ dàng truy cập và thay đổi dữ liệu một cách hiệu quả.

5. Giao dịch trong DBMS

- **Giao dịch (Transaction)** là một đơn vị thao tác dữ liệu trong DBMS.
- DBMS đảm bảo tính toàn vẹn của giao dịch theo nguyên tắc **tất cả hoặc không có gì**.
- Để tối ưu hiệu suất, DBMS có thể xử lý nhiều giao dịch **đồng thời**, nhưng vẫn đảm bảo kết quả tương đương với việc thực hiện từng giao dịch một cách tuần tự.
- Hệ thống lưu trữ **nhật ký (log)** để có thể hoàn tác các thay đổi chưa hoàn tất và phục hồi dữ liệu khi gặp sự cố.
- **Checkpointing** là một cơ chế giúp rút ngắn thời gian khôi phục khi hệ thống gặp lỗi.

6. Cấu trúc của DBMS

Mã nguồn của DBMS được chia thành nhiều mô-đun, bao gồm:

- **Trình quản lý không gian đĩa:** Quản lý lưu trữ dữ liệu trên ổ đĩa
- **Trình quản lý bộ đệm:** Điều phối dữ liệu giữa bộ nhớ và ổ đĩa
- **Lớp trừu tượng hóa tệp và chỉ mục:** Hỗ trợ quản lý tệp và truy vấn nhanh
- **Lớp thực thi toán tử quan hệ:** Xử lý các truy vấn SQL
- **Lớp tối ưu hóa truy vấn:** Cải thiện hiệu suất truy vấn bằng cách chọn chiến lược tốt nhất

7. Vai trò của các thành phần trong hệ thống DBMS

- **Quản trị viên cơ sở dữ liệu (DBA):**
 - Thiết kế và quản lý lược đồ dữ liệu
 - Đảm bảo bảo mật dữ liệu
 - Khôi phục hệ thống khi gặp sự cố
 - Tối ưu hóa hiệu suất hệ thống
- **Lập trình viên ứng dụng:**

- Xây dựng ứng dụng sử dụng DBMS để quản lý và thao tác dữ liệu
- **Người dùng cuối:**
 - Sử dụng các ứng dụng do lập trình viên phát triển để truy cập và thao tác dữ liệu

EXERCISES

1.1 Sử dụng hệ thống cơ sở dữ liệu (DBMS) có nhiều lợi ích so với việc lưu trữ dữ liệu trong tệp hệ điều hành, bao gồm:

- **Tính toàn vẹn dữ liệu:** DBMS đảm bảo dữ liệu không bị lỗi hoặc mâu thuẫn thông qua các ràng buộc.
- **Bảo mật:** DBMS cung cấp cơ chế kiểm soát quyền truy cập dữ liệu.
- **Truy vấn hiệu quả:** DBMS có các công cụ tối ưu hóa để truy vấn dữ liệu nhanh hơn.
- **Hỗ trợ truy cập đồng thời:** DBMS quản lý xung đột khi nhiều người cùng truy cập dữ liệu.
- **Khả năng phục hồi:** DBMS có thể khôi phục dữ liệu khi hệ thống gặp sự cố.

Khi nào thì không dùng DBMS?

- **Ứng dụng có ràng buộc thời gian thực nghiêm ngặt:** DBMS có thể không đủ nhanh để xử lý dữ liệu ngay lập tức.
- **Ứng dụng chỉ có một số ít thao tác quan trọng, được xác định rõ ràng:** Trong trường hợp này, có thể viết mã tùy chỉnh hiệu quả hơn.
- **Ứng dụng cần thao tác dữ liệu theo cách không được ngôn ngữ truy vấn của DBMS hỗ trợ:** Khi đó, mô hình dữ liệu trừu tượng của DBMS có thể gây trở ngại thay vì giúp ích.

1.2. **Tính độc lập dữ liệu logic** là khả năng thay đổi **lược đồ khái niệm** (conceptual schema) mà không ảnh hưởng đến **lược đồ bên ngoài** (external schema) hoặc các ứng dụng sử dụng dữ liệu.

Tại sao tính độc lập dữ liệu logic quan trọng?

1. **Giảm ảnh hưởng của thay đổi dữ liệu:** Khi cấu trúc dữ liệu thay đổi, ứng dụng vẫn có thể hoạt động bình thường.
2. **Tăng tính linh hoạt:** CSDL có thể được điều chỉnh mà không làm gián đoạn các ứng dụng.
3. **Dễ dàng mở rộng và bảo trì:** Giúp quản trị viên có thể tái cấu trúc CSDL mà không ảnh hưởng đến người dùng cuối.

1.3 Sự khác biệt giữa tính độc lập dữ liệu logic và vật lý

1. Tính độc lập dữ liệu logic (Logical Data Independence)

- **Khái niệm:** Khả năng thay đổi lược đồ khái niệm (**conceptual schema**) mà không ảnh hưởng đến lược đồ bên ngoài (**external schema**) hoặc các ứng dụng.
- **Ví dụ:**
 - Ban đầu có bảng **Faculty(fid, fname, sal, office)**.
 - Sau đó, để bảo mật thông tin lương giảng viên, bảng được tách thành **Faculty_public(fid, fname, office)** và **Faculty_private(fid, sal)**.
 - Các ứng dụng vẫn truy vấn thông qua view **Courseinfo**, không cần thay đổi.
- **Lợi ích:**
 - Che giấu thay đổi về cấu trúc logic của dữ liệu.
 - Giúp ứng dụng tiếp tục hoạt động mà không cần chỉnh sửa khi cơ sở dữ liệu thay đổi.

2. Tính độc lập dữ liệu vật lý (Physical Data Independence)

- **Khái niệm:** Khả năng thay đổi **cách dữ liệu được lưu trữ vật lý (internal schema)** mà không ảnh hưởng đến **lược đồ khái niệm (conceptual schema)**.
- **Ví dụ:**
 - Thay đổi cách lưu trữ dữ liệu trên đĩa, cấu trúc tệp hoặc chỉ mục nhưng vẫn giữ nguyên lược đồ khái niệm.
 - Ví dụ, đổi từ lưu trữ tuần tự sang lưu trữ theo chỉ mục mà không làm thay đổi bảng Faculty.
- **Lợi ích:**
 - Cho phép tối ưu hóa hiệu suất mà không ảnh hưởng đến cách truy vấn dữ liệu.
 - Cơ sở dữ liệu có thể thay đổi cách lưu trữ mà không làm gián đoạn hệ thống.

1.4 Sự khác biệt giữa lược đồ bên ngoài, lược đồ khái niệm và lược đồ vật lý

- **Lược đồ bên ngoài (External Schema):** Cách người dùng nhìn thấy dữ liệu (các giao diện, báo cáo, chế độ xem).
- **Lược đồ khái niệm (Conceptual Schema):** Cấu trúc logic của toàn bộ cơ sở dữ liệu (bảng, quan hệ, ràng buộc).
- **Lược đồ vật lý (Internal Schema):** Cách dữ liệu được lưu trữ thực tế trong hệ thống (cấu trúc tệp, chỉ mục).

Mối quan hệ với tính độc lập dữ liệu

- **Tính độc lập dữ liệu logic:** Thay đổi lược đồ khái niệm mà không ảnh hưởng đến lược đồ bên ngoài.
- **Tính độc lập dữ liệu vật lý:** Thay đổi lược đồ vật lý mà không ảnh hưởng đến lược đồ khái niệm.

Ví dụ:

- Nếu dữ liệu được lưu trữ bằng cấu trúc khác nhưng quan hệ giữa các bảng không thay đổi → đảm bảo **tính độc lập dữ liệu vật lý**.
- Nếu thêm một trường mới vào bảng mà ứng dụng hiện tại không sử dụng → đảm bảo **tính độc lập dữ liệu logic**.

2.

