**NGUYỄN THẾ DŨNG**

# NHẬP MÔN
# CƠ SỞ DỮ LIỆU

**Trường Đại học Sư phạm - Đại học Huế**
**Huế, tháng 7 năm 2011**

Giáo trình này được viết bởi Nguyễn Thế Dũng, giảng viên Khoa Tin học, Trường ĐHSP - Đại học Huế. Giáo trình này được dùng để giảng dạy và học tập học phần: Nhập môn cơ sở dữ liệu. Mã số: TINS4373.

# Lời nói đầu

Giáo trình này nhằm góp phần nâng cao chất lượng giảng dạy và học tập môn học Cơ sở dữ liệu. Cuối các chương mục, chúng tôi đưa vào phần ôn tập chương cùng các câu hỏi và bài tập nhằm giúp sinh viên dễ học tập và có một cái nhìn rộng hơn về thực tiễn hay các vấn đề mở mà giáo trình chưa đề cập đến do giới hạn khuôn khổ. Phần hướng dẫn trả lời câu hỏi và bài tập xin dành trong một cuốn sách bài tập mà chúng tôi dự định hoàn thành trong thời gian đến.

Chúng tôi cũng sẽ thiết kế một giáo trình điện tử tương ứng với sách này, cùng các tư liệu học tập khác như hệ thống slide bài giảng, hệ thống câu hỏi trắc nghiệm, hệ thống các module hỗ trợ thiết kế cơ sở dữ liệu, mô phỏng câu truy vấn cơ sở dữ liệu ...

Giáo trình gồm các chương sau:

*Chương 1. Khái quát về các hệ cơ sở dữ liệu.*

*Chương 2. Mô hình cơ sở dữ liệu.*

*Chương 3. Mô hình cơ sở dữ liệu quan hệ.*

*Chương 4. Ngôn ngữ cơ sở dữ liệu.*

*Chương 5. Ràng buộc toàn vẹn.*

*Chương 6. Phụ thuộc hàm và Khoá*

*Chương 7. Phân tách.*

*Chương 8. Chuẩn hoá.*

*Chương 9. Phụ thuộc đa trị.*

Có một chút đặc biệt trong giáo trình, đó là:

Không như trong một số tài liệu về môn hệ cơ sở dữ liệu khác, thường trình bày các vấn đề liên quan đến phụ thuộc hàm, khoá, phân tách và chuẩn hoá... thành lý thuyết thiết kế cơ sở dữ liệu quan hệ, như vậy chương này sẽ rất quá dài hay phải phân chia thành một phần tách biệt. Ở đây chúng tôi chia các vấn đề về lý thuyết thiết kế nói trên thành 3 chương, gồm Chương 6. Phụ thuộc hàm và Khoá, Chương 7. Lý thuyết phân tách và Chương 8. Chuẩn hoá, để vấn đề được rõ ràng hơn. Tách riêng vấn đề phân tách và chuẩn hóa, theo chúng tôi sẽ giúp ta có cái nhìn tốt trong khi nghiên cứu cơ sở dữ liệu phân tán.

Giáo trình là kết quả của nhiều niên khoá dạy bộ môn này, có thể xem như

là một tài liệu trong học tập cho các sinh viên học ngành Công nghệ thông tin, cũng như sách tham khảo cho các bạn đọc khác. Do hạn chế về thời gian cũng như trình độ, nên giáo trình chắc chắn có nhiều sai sót. Chúng tôi rất mong sự chỉ bảo của các bạn đọc, nhằm nâng cao chất lượng của giáo trình.

Trong quá trình biên soạn giáo trình này, chúng tôi có tham khảo một số tài liệu của một số tác giả khác, nhằm mang lại những kiến thức phong phú, hay nhất cho sinh viên, nhưng có thể chưa kịp liên hệ được với chính các tác giả ấy. Mong các Thầy, cô vì sự học của các sinh viên mà niệm tình bỏ quá.

Tác giả chân thành cảm ơn Th.S Nguyễn Đức Nhuận, TS. Hoàng Quang và TS Hoàng Lan Giao đã cho nhiều ý kiến đóng góp quý giá, cũng như đã tận tình sửa chữa bản thảo cho giáo trình, chúng tôi cũng gửi lời cảm ơn đến rất nhiều bạn sinh viên đã giúp chúng tôi đánh máy giáo trình, ghi chép bài vở đầy đủ, sưu tầm đề thi, bài tập mà thời điểm trước 2006 còn rất thiếu, làm cơ sở cho chúng tôi viết giáo trình này. Tác giả gửi lời cảm ơn đến anh Dương Phương Hùng là cựu sinh viên của Khoa Tin học – ĐHSP Huế khóa 1996 – 2000, từ năm 1999 đã bước đầu đánh máy cho bản thảo cuốn giáo trình này, cho đến hôm nay tác giả mới hoàn thành được bước đầu, nhằm đáp lại sự giúp đỡ ấy của các Thầy cô, anh chị em sinh viên và đặc biệt là anh Dương Phương Hùng.

***Huế - 7/2011***

***Nguyễn Thế Dũng***

***Khoa Tin học - ĐHSP Huế***

# DANH MỤC CÁC TỪ VIẾT TẮT

| | |
|---|---|
| CSDL | Cơ sở dữ liệu |
| QTCSDL | Quản trị cơ sở dữ liệu |
| DBMS | Hệ quản trị cơ sở dữ liệu |
| DM | Hệ quản trị dữ liệu |
| E/R | Thực thể liên kết/quan hệ thực thể |
| CBH | Chuyên biệt hóa |
| TQH | Tổng quát hóa |
| PTH | Phụ thuộc hàm |
| LĐQH | Lược đồ quan hệ |
| RBTV | Ràng buộc toàn vẹn |

# BẢNG ĐỐI CHIẾU THUẬT NGỮ VIỆT – ANH

| | |
|---|---|
| Cơ sở dữ liệu | Database |
| Hệ Quản trị cơ sở dữ liệu | Database Management System |
| Hệ quản trị dữ liệu | Data Management. |
| Thực thể liên kết/quan hệ thực thể | Entity Relationship |
| Sơ đồ mối quan hệ thực thể | Entity Relationship Diagram |
| Phụ thuộc hàm | Function Dependency |
| Lược đồ quan hệ | Relation Schema |
| Khóa | Key |
| Khóa ngoại | Foreign key |
| Siêu khóa | Supper Key |
| Khóa dự tuyển | Candidate key |
| Ràng buộc toàn vẹn | Integrity constraint |
| Dạng chuẩn | Dạng chuẩn |
| Dư thừa dữ liệu | Redundant Information |
| Bản ghi ảo | Spurious tuples |
| Bất thường khi chèn/xóa/cập nhật bộ | Insertion/delete/udate Anomalies |

# Mục lục

# CHƯƠNG 1. KHÁI QUÁT VỀ CÁC HỆ CƠ SỞ DỮ LIỆU

## Mục đích

*Khái quát các nguyên lý của hệ cơ sở dữ liệu (CSDL) gồm CSDL, hệ quản trị CSDL, con người và các trang thiết bị lưu trữ xử lý dữ liệu.*

*Trình bày về quá trình quản lý dữ liệu bao gồm định nghĩa các cấu trúc lưu trữ thông tin, cung cấp các cơ chế cho việc thao tác thông tin, đảm bảo an toàn cho các sự cố và truy cập không được phép, đảm bảo các dị thường dữ liệu khi có nhiều người cùng chia sẻ dữ liệu.*

*Kiến trúc 3 mức của một hệ CSDL nhằm thể hiện các mức trừu tượng dữ liệu, giúp cho đa số người sử dụng tránh phải quan tâm đến chi tiết về lưu trữ và bảo trì dữ liệu.*

*Ba mức thiết kế CSDL cùng các sản phẩm tương ứng là các lược đồ ngoài, lược đồ khái niệm, lược đồ trong được trình bày, khái niệm độc lập dữ liệu cũng được đề cập.*

*Các chức năng và thành phần chủ yếu của hệ quản trị CSDL (DBMS).*

*Vai trò và chức năng của con người trong mối quan hệ tương tác với hệ CSDL.*

## Yêu cầu

*Hiểu và cho ví dụ minh họa các khái niệm cơ bản của hệ CSDL.*

*Cho ví dụ minh họa về các chức năng của DBMS (đã học ở các năm dưới) và thấy được vai trò của mình trong tương lai với hệ CSDL.*

*Cho ví dụ minh họa nêu lên ý nghĩa của kiến trúc 3 mức.*

## 1.1. Cơ sở dữ liệu là gì? Tại sao cần tới các hệ cơ sở dữ liệu?

Trước khi các hệ CSDL ra đời (khoảng đầu những năm 60 của thế kỷ 20), mỗi chương trình ứng dụng đều có một tệp dữ liệu tương ứng và mỗi khi chương trình ứng dụng cần được sửa đổi hoặc mở rộng thì tệp dữ liệu tương ứng cũng phải thay đổi theo. Việc lưu giữ thông tin của một tổ chức trong một hệ xử lý tệp như vậy (thường hệ này được hỗ trợ bởi một hệ điều hành truyền thống) có những nhược điểm chính như sau:

- *Dư thừa dữ liệu và không nhất quán* (cùng một dữ liệu có thể có được lưu trữ trong nhiều tệp khác nhau; khi tiến hành cập nhật có thể bỏ sót và dẫn tới không nhất quán).
- *Khó khăn trong việc truy cập dữ liệu* (các môi trường xử lý tệp truyền thống không cho phép dữ liệu được tìm kiếm theo cách thức thuận tiện và hiệu quả).
- *Sự cô lập của dữ liệu* (dữ liệu nằm rải rác trong nhiều tệp và các tệp có thể có khuôn dạng khác nhau, nên khó viết các chương trình ứng dụng mới để tìm các dữ liệu thích hợp).
- *Các vấn đề toàn vẹn* (khi có thêm những ràng buộc mới, khó thay đổi các chương trình để có thể tuân thủ chúng).
- *Các vấn đề về tính nguyên tố của các giao tác* (với hệ xử lý tệp truyền thống khó có thể đảm bảo được tính chất "hoặc thực hiện hoàn toàn hoặc không thực hiện gì" và khó đưa được hệ thống trở về trạng thái nhất quán trước khi xảy ra sự cố).
- *Các dị thường của truy cập tương tranh* (để tăng tính hiệu quả và trả lời nhanh hơn, nhiều hệ thống cho phép nhiều người dùng cập nhật dữ liệu đồng thời và như vậy có thể dẫn đến dữ liệu không nhất quán).
- *Các vấn đề an toàn*. Thường thì mỗi người dùng của hệ CSDL chỉ được phép truy cập một phần của CSDL và điều đó cũng là một biện pháp giữ cho dữ liệu trong CSDL được an toàn. Còn với hệ xử lý – tệp truyền thống, các chương trình ứng dụng được thêm vào hệ thống an toàn một cách thức không tiên liệu trước nên khó đảm bảo được các ràng buộc an toàn như vậy.

Các hệ CSDL ra đời nhằm giải quyết những vấn đề nêu trên.

*Một* **cơ sở dữ liệu (CSDL)** *là một tập hợp các dữ liệu có liên quan với nhau được lưu trữ trên các thiết bị nhớ thứ cấp (như băng từ, đĩa từ ...) để đáp ứng nhu cầu khai thác thông tin của nhiều người sử dụng với nhiều mục đích khác nhau.*

Trong cách hiểu trên, trước hết CSDL phải phản ánh thông tin về hoạt động của nhiều sự vật hiện tượng, nghĩa là biểu thị một "góc" của thế giới thực tại, nó phải là một tập hợp các thông tin mang tính hệ thống chứ không thể là một tập

hợp dữ liệu tùy tiện chứa những thông tin rời rạc không có mối quan hệ với nhau. Thông tin lưu trữ trong CSDL được chia sẻ cho nhiều người sử dụng và nhiều ứng dụng khác nhau. Từ đó có thể thấy việc xây dựng và khai thác một CSDL liên quan đến một số vấn đề như đảm bảo tính nhất quán và toàn vẹn dữ liệu, tính bảo mật và quyền khai thác thông tin của người sử dụng, tính an toàn cho dữ liệu khi xảy ra sự cố nào đó…

*Phần mềm cho phép người dùng giao tiếp với CSDL, cung cấp một môi trường thuận lợi và hiệu quả để tìm kiếm và lưu trữ thông tin của CSDL được gọi là* ***hệ quản trị cơ sở dữ liệu*** *(hệ QTCSDL).*

Người ta thường dùng thuật ngữ ***hệ cơ sở dữ liệu*** để chỉ sự kết hợp của các yếu tố sau:

- Cơ sở dữ liệu

- Hệ QTCSDL để truy cập CSDL đó.

- Con người và trang thiết bị để lưu trữ dữ liệu.

Mục đích chính của một hệ CSDL là cung cấp cho người dùng một cách nhìn trừu tượng về dữ liệu. Điều đó có nghĩa là hệ thống che dấu những chi tiết phức tạp về cách thức dữ liệu được lưu trữ và bảo trì. Chính vì vậy, trong cuộc sống hiện đại ngày nay, việc sử dụng các CSDL trở nên phổ biến, quen thuộc đến nỗi nhiều lúc chúng ta coi đó là điều tự nhiên. Khi đến thư viện tìm mượn sách, nhờ máy tính ít nhất chúng ta có thể biết được thông tin chi tiết về sách của thư viện, thông tin về sách đã có người xếp hàng đặt mượn… Khi chúng ta muốn đặt chỗ cho chuyến bay sắp tới của mình, nhân viên đại lí bán vé hàng không sẽ nhanh chóng cung cấp những thông tin cần thiết giúp chúng ta như một thông tin cập nhật vào tập hợp dữ liệu được lưu trữ. Sự phát triển mạnh mẽ của Internet ở thập kỉ cuối của thế kỷ 20 đã làm số người truy cập và khai thác thông tin trong các cơ sở dữ liệu tăng lên  rất nhanh chóng. Với các giao diện Web, người ta có thể đăng kí các khóa học ở một trường đại học, có thể xem số dư trong tài khoản của mình ở một ngân hàng, có thể tìm hiểu chi tiết về một mặt hàng nào đó, … Càng ngày việc truy xuất thông tin trong các cơ sở dữ liệu càng trở thành một bộ phận thiết yếu trong cuộc sống của mỗi người. Vì nhiều người sử dụng CSDL không thuộc giới chuyên tin, những người phát triển hệ thống đã che dấu không

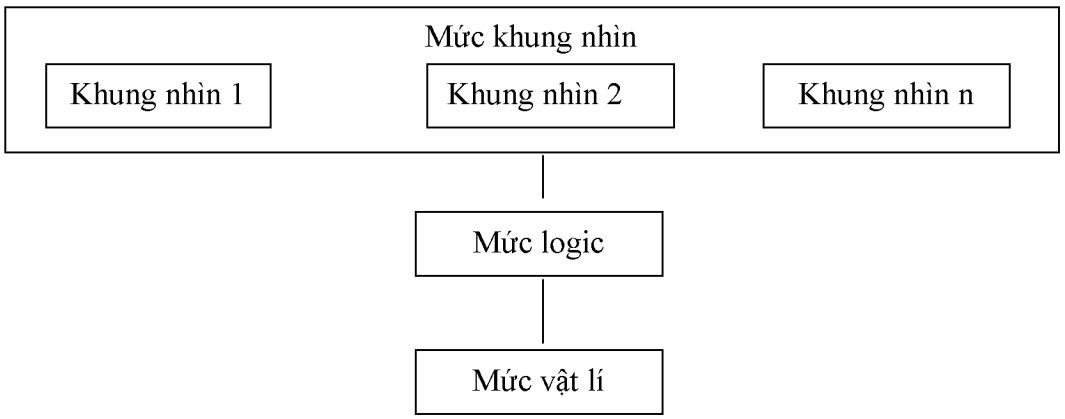
cho người dùng biết sự phức tạp đó thông qua nhiều mức biểu diễn, nhằm làm đơn giản những tương tác của người dùng với hệ thống.

## 1.2. Kiến trúc ba mức của một hệ CSDL

Theo ANSI-PARC (American National Standard Institue – Planning and Requirements Committee: Viện tiêu chuẩn quốc gia Mỹ - Ủy ban nhu cầu và kế hoạch Mỹ) có 3 mức biểu diễn một CSDL.

**Mức vật lí** (còn gọi là mức trong). Mức biểu diễn thấp nhất này mô tả dữ liệu được thực sự lưu trữ như thế nào trong CSDL. Đây là mức thể hiện các cài đặt có tính chất vật lí của CSDL để đạt được tối ưu trong các lần thực hiện các thao tác tìm kiếm và lưu trữ, để tận dụng được các vùng nhớ còn trống. Mức vật lý cũng là mức phản ánh các cấu trúc dữ liệu, các tổ chức tệp được dùng cho việc lưu trữ dữ liệu trên các thiết bị nhớ thứ cấp. Điều đó cũng có nghĩa là mức này tiếp xúc với các phương thức truy nhập của hệ điều hành để đặt dữ liệu vào các thiết bị nhớ, xây dựng các tập chỉ mục, truy xuất dữ liệu,… liên quan đến các vấn đề như sự cấp phát vùng nhớ cho dữ liệu và các chỉ mục, các mô tả bản ghi để lưu trữ, các kĩ thuật nén dữ liệu và giải mã dữ liệu.

**Mức logic** (còn gọi là mức khái niệm). Đây là mức mô tả những dữ liệu nào được lưu trữ CSDL và có những mối quan hệ nào giữa chúng. Nói một cách cụ thể hơn, mức logic biểu diễn các thực thể (trong thế giới thực tại), các thuộc tính và các mối quan hệ giữa các thực thể đó; mức logic cũng cho thấy các ràng buộc trên dữ liệu, các thông tin về ngữ nghĩa của dữ liệu, các thông tin về an ninh và toàn vẹn của dữ liệu. Tuy nhiên mức biểu diễn này chỉ quan tâm đến cái gì được lưu trữ trong CSDL chứ không quan tâm đến cách thức lưu trữ.




***Hình 1.1. Ba mức của sự biểu diễn dữ liệu***

**Mức khung nhìn** (còn gọi là mức ngoài). Mức biểu diễn cao nhất này mô tả chỉ một phần của toàn bộ CSDL, phần thích hợp với một người sử dụng nhất định. Mức này gồm một số khung nhìn của những người sử dụng đặt vào CSDL. Mỗi người dùng có thể không quan tâm đến toàn bộ thông tin của hệ CSDL mà chỉ cần một phần thông tin nào đó. Họ có một cách nhìn thế giới thực tại theo cách nhìn gần gũi và phù hợp với họ. Khung nhìn dành cho người sử dụng đó chỉ gồm những thực thể cùng những thuộc tính, những mối quan hệ của những thực thể mà họ quan tâm. Các khung nhìn khác nhau cũng có thể trình bày cùng một dữ liệu nhưng ở những khuôn dạng khác nhau. Ví dụ người sử dụng này có thể nhìn thấy thông tin ngày theo kiểu (họ, tên) còn người sử dụng khác lại ở kiểu (tên, họ). Một số khung nhìn có thể chứa các dữ liệu suy dẫn ra được hay tính toán được, những dữ liệu này vốn không được thực sự lưu trữ trong CSDL nhưng được tạo ra khi cần đến.

Tóm lại, mức khung nhìn là cách cảm nhận của người dùng về dữ liệu, mức vật lý là cách nhìn nhận của hệ QTCSDL và hệ điều hành về dữ liệu. Mức logic nằm giữa mức khung nhìn và mức vật lý, có thể coi đây là cách cảm nhận của toàn thể cộng đồng người dùng về dữ liệu. Tại mức logic tồn tại cả hai ánh xạ đến hai mức còn lại, tạo nên một sự độc lập đối với nhau của hai mức đó.

Có thể thấy *mục đích của kiến trúc ba mức* nêu trên chính là sự tách biệt quan niệm về CSDL của nhiều người sử dụng với những chi tiết biểu diễn về vật lý của CSDL . Điều đó dẫn đến những thuận lợi sau:

Đối với một CSDL, mỗi người dùng có một khung nhìn riêng của mình. Họ có thể thay đổi khung nhìn của họ và sự thay đổi này không làm ảnh hưởng đến những khung nhìn dữ liệu của người dùng khác đang dùng chung CSDL này.

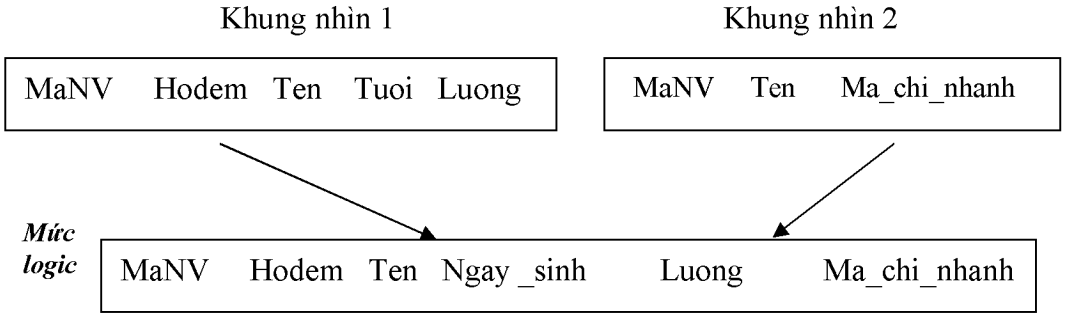
Những tương tác của người dùng với CSDL không phụ thuộc vào những vấn đề chi tiết trong lưu trữ dữ liệu (chẳng hạn như vấn đề chỉ mục hóa hay bảng băm).

Người quản trị CSDL (Database Administrator, thường viết tắt là DBA) có thể thay đổi cấu trúc khái niệm của CSDL mà không làm ảnh hưởng đến tất cả người dùng.

## 1.3. Lược đồ và thể hiện của CSDL

Toàn bộ mô tả CSDL được gọi là **lược đồ CSDL** (database schema). Tương ứng với ba mức biểu diễn dữ liệu nói trên có ba loại lược đồ. Ở mức cao nhất chúng ta có nhiều lược đồ ngoài (còn gọi là lược đồ con) cho những cách nhìn dữ liệu khác nhau của những người sử dụng khác nhau. Ở mức logic chúng ta có lược đồ logic. Ở mức thấp nhất chúng ta có lược đồ vật lí. Thường thì các hệ CSDL hỗ trợ một lược đồ vật lí, một lược đồ logic và nhiều lược đồ con.

Một điều quan trọng là cần phân biệt mô tả của CSDL (tức là lược đồ CSDL) với bản thân CSDL. Lược đồ được xác định trong quá trình thiết kế CSDL và người ta không muốn nó thay đổi thường xuyên. Trong khi đó bản thân CSDL sẽ thay đổi theo thời gian do dữ liệu được thêm vào, xóa đi hay sửa đổi. Toàn bộ dữ liệu lưu trữ trong CSDL tại một thời điểm nhất định được gọi là một *thể hiện* của CSDL (database instance). Như vậy nhiều thể hiện của CSDL có thể tương ứng với cùng một lược đồ CSDL . Đôi khi lược đồ còn được gọi là *nội hàm* (instension) của CSDL và một thể hiện còn được gọi là một *mở rộng* hay một *trạng thái* (extension hay state) của CSDL .




```
struct NHAVIEN{
int MaNV
int Ma_chi_nhanh;
char Hodem [15] ;
        char Ten [15]
    struct date Ngay_sinh;
    float Luong;
```

*Mức vật lý*

struct NHANVIEN *next; /*con trỏ đến bản ghi tiếp của tệp NHANVIEN*};

index MaNV; index Ma_chi_nhanh; /*xác định các chỉ mục cho tệp NHANVIEN*/

### 1.4. Sự độc lập của dữ liệu

Có thể nói một cách khác về mục đích của kiến trúc ba mức của CSDL mà chúng ta vừa nói ở trên (ở mục 2), đó là sự độc lập dữ liệu (data independence), hiểu theo nghĩa các lược đồ ở mức trên không bị ảnh hưởng khi có sự thay đổi các lược đồ ở các mức dưới. Có hai loại độc lập dữ liệu.

*Độc lập dữ liệu vật lí* là khả năng sửa đổi lược đồ vật lí mà không làm thay đổi lược đồ khái niệm và như vậy cũng không đòi hỏi viết lại các trình ứng dụng. Để tăng tính hiệu quả, nhiều khi cần có những thay đổi ở mức vật lí. Chẳng hạn như sử dụng các tổ chức tệp khác trước, dùng thiết bị nhớ khác, thay đổi các chỉ mục hay thay đổi thuật toán băm.

*Độc lập dữ liệu logic* là khả năng sửa đổi lược đồ logic mà không làm thay đổi các khung nhìn (các lược đồ ngoài), cũng có nghĩa là không đòi hỏi viết lại các trình ứng dụng. Các sửa đổi ở mức logic là cần thiết mỗi khi cấu trúc logic của CSDL cần phải thay đổi, chẳng hạn cần thêm hay bớt các thực thể nào đó, các thuộc tính hay các mối quan hệ của chúng. Dĩ nhiên những người dùng có chạm đến những thông tin đã thay đổi này sẽ được thông báo về sự thay đổi nhưng điều quan trọng là những người dùng khác sẽ không bị ảnh hưởng gì.

Độc lập dữ liệu logic khó thực hiện hơn độc lập dữ liệu vật lí vì các chương trình ứng dụng phụ thuộc nhiều vào cấu trúc logic của dữ liệu mà chúng truy cập.

Khái niệm độc lập dữ liệu trong nhiều mặt tương tự với khái niệm *kiểu dữ liệu trừu tượng* trong các ngôn ngữ lập trình hiện đại. Cả hai đều che dấu người dùng những chi tiết cài đặt, cho phép người dùng tập trung vào cấu trúc chung hơn là tập trung vào các chi tiết cài đặt ở mức thấp.

### 1.5. Những cách tiếp cận một CSDL

Trên thực tế, một lược đồ được viết trong ngôn ngữ định nghĩa dữ liệu của một hệ quản trị CSDL cụ thể. Để mô tả các yêu cầu dữ liệu của một tổ chức sao cho mô tả đó dễ hiểu đối với nhiều người sử dụng khác nhau thì ngôn ngữ này lại

ở mức quá thấp. Như vậy cần phải có một mô tả lược đồ ở mức cao hơn, nói cách khác cần phải có một mô hình dữ liệu.

*Mô hình dữ liệu là một tập các khái niệm và kí pháp dùng để mô tả dữ liệu, các mối quan hệ của dữ liệu, các ràng buộc trên dữ liệu của một tổ chức.*

Như vậy có thể xem như một mô hình dữ liệu có ba thành phần:

1. Phần mô tả cấu trúc của CSDL.
2. Phần mô tả các thao tác, định nghĩa các phép toán được phép trên dữ liệu.
3. Phần mô tả các ràng buộc toàn vẹn để đảm bảo sự chính xác của dữ liệu.

Khi dùng mô hình dữ liệu chúng ta có thể biểu diễn dữ liệu theo một cách dễ hiểu và vì vậy mô hình cũng được sử dụng trong việc thiết kế CSDL.

Đã có nhiều mô hình dữ liệu được đề xuất và có thể chia thành ba nhóm theo các cách tiếp cận mô hình hóa như sau:

+ Mô hình (dữ liệu) logic trên cơ sở đối tượng.

Chẳng hạn mô hình quan hệ thực thể, mô hình hướng đối tượng… Các mô hình thuộc nhóm này thường được dùng trong việc mô tả dữ liệu ở mức logic và khung nhìn. Chúng cung cấp các khả năng cấu trúc rất mềm dẻo và cho phép các ràng buộc được đặc tả tường minh. Chúng ta sẽ bàn bạc kỹ hơn về các mô hình này trong các chương sau.

+ Mô hình (dữ liệu) logic trên cơ sở bản ghi.

Các mô hình này thường được dùng trong việc mô tả dữ liệu ở mức khung nhìn và logic. Chúng được dùng mô tả cấu trúc logic tổng thể của CSDL, đồng thời cung cấp một mức cao hơn của sự cài đặt. Trong các mô hình này CSDL được cấu trúc thành các bản ghi có khuôn dạng cố định gồm một số trường (thuộc tính) có thể thuộc nhiều kiểu dữ liệu.

Các mô hình logic quen thuộc được dùng là: Mô hình quan hệ (relational data model), mô hình mạng, mô hình phân cấp. Các mô hình này cũng sẽ được trình bày kỹ trong chương sau.

+ Mô hình (dữ liệu) vật lí.

Các mô hình logic tập trung vào bản chất logic của biểu diễn dữ liệu, tập trung vào cái được biểu diễn trong CSDL, còn được gọi là các mô hình dữ liệu bậc cao. Các mô hình vật lí tập trung vào những chi tiết cho biết dữ liệu được lưu trữ thế nào, còn được gọi là các mô hình dữ liệu bậc thấp.

Các mô hình dữ liệu vật lý mô tả dữ liệu được lưu trữ thế nào trên máy tính, mô tả cấu trúc bản ghi, thứ tự các bản ghi và con đường truy cập. Hai mô hình dữ liệu vật lý quen dùng là mô hình hợp nhất và mô hình bộ nhớ-khung. Mô hình này thường chỉ phù hợp với các chuyên gia lập trình chứ không cần thiết đối với đa số người sử dụng.

## 1.6. Hệ quản trị cơ sở dữ liệu (DBMS)

*Hệ chương trình được xây dựng để giúp người sử dụng định nghĩa, tạo lập, xử lý, bảo trì các CSDL và cung cấp các truy cập có điều khiển đến các CSDL này gọi là hệ quản trị cơ sở dữ liệu (DataBase Management System-DBMS), viết tắt là DBMS.*

DBMS cung cấp cho người sử dụng các ***phương tiện*** sau:

*a. Cung cấp ngôn ngữ cơ sở dữ liệu*

Một hệ cơ sở dữ liệu cung cấp hai kiểu ngôn ngữ khác nhau: một để xác định sơ đồ cơ sở dữ liệu, một để biểu diễn các vấn tin cơ sở dữ liệu và cập nhật.

Ngôn ngữ định nghĩa dữ liệu (Data Definition Language: DDL) cho phép định nghĩa sơ đồ cơ sở dữ liệu. Kết quả biên dịch các lệnh của DDL là tập hợp các bảng được lưu trữ trong một file đặc biệt được gọi là từ điển dữ liệu (data dictionary) hay thư mục dữ liệu (data directory). Tự điển dữ liệu là một file chứa metadata. File này được tra cứu trước khi dữ liệu hiện hành được đọc hay sửa đổi. Cấu trúc lưu trữ và phương pháp truy cập được sử dụng bởi hệ cơ sở dữ liệu được xác định bởi một tập hợp các định nghĩa trong một kiểu đặc biệt của DDL được gọi là ngôn ngữ định nghĩa và lưu trữ dữ liệu (data storage and definition language). Kết quả biên dịch của các định nghĩa này là một tập hợp các chỉ thị xác định sự thực hiện chi tiết của các sơ đồ cơ sở dữ liệu (thường được che dấu).

Ngôn ngữ thao tác dữ liệu (Data manipulation language: DML) là ngôn ngữ cho phép người sử dụng truy xuất hoặc thao tác dữ liệu. Có hai kiểu ngôn ngữ thao tác dữ liệu: DML thủ tục (procedural DML) yêu cầu người sử dụng đặc tả dữ

liệu nào cần và làm thế nào để nhận được nó. DML không thủ tục (Nonprocedural DML) yêu cầu người sử dụng đặc tả dữ liệu nào cần nhưng không cần đặc tả làm thế nào để nhận được nó. Một vấn tin (query) là một lệnh yêu cầu tìm lại dữ liệu (information retrieval). Phần ngôn ngữ DML liên quan đến sự tìm lại thông tin được gọi là ngôn ngữ vấn tin (query language).

*b. Các kiểm soát, các điều khiển đối với truy cập vào CSDL*

Bao gồm:

+ Quản trị giao dịch

Thông thường, một số thao tác trên cơ sở dữ liệu tạo thành một đơn vị logic công việc. Ta hãy xét ví dụ chuyển khoản, trong đó một số tiền x được chuyển từ tài khoản A ( A:=A-x ) sang một tài khoản B ( B:=B+x ). Một yếu tố cần thiết là cả hai thao tác này hoặc cùng xảy ra hoặc không hoạt động nào xảy ra cả. Việc chuyển khoản phải xảy ra trong tính toàn thể của nó hoặc không. Đòi hỏi toàn thể-hoặc-không này được gọi là tính nguyên tử (atomicity). Một yếu tố cần thiết khác là sự thực hiện việc chuyển khoản bảo tồn tính nhất quán của cơ sở dữ liệu: giá trị của tổng A + B phải được bảo tồn. Đòi hỏi về tính chính xác này được gọi là tính nhất quán (consistency). Cuối cùng, sau khi thực hiện thành công hoạt động chuyển khoản, các giá trị của các tài khoản A và B phải bền vững cho dù có thể có sự cố hệ thống. Đòi hỏi về tính bền vững này được gọi là tính lâu bền (durability).

Một giao dịch là một tập các hoạt động thực hiện chỉ một chức năng logic trong một ứng dụng cơ sở dữ liệu. Mỗi giao dịch là một đơn vị mang cả tính nguyên tử lẫn tính nhất quán. Như vậy, các giao dịch phải không được vi phạm bất kỳ ràng buộc nhất quán nào: Nếu cơ sở dữ liệu là nhất quán khi một giao dịch khởi động thì nó cũng phải là nhất quán khi giao dịch kết thúc thành công. Tuy nhiên, trong khi đang thực hiện giao dịch, phải cho phép sự không nhất quán tạm thời. Sự không nhất quán tạm thời này tuy là cần thiết nhưng lại có thể dẫn đến các khó khăn nếu xảy ra sự cố.

Trách nhiệm của người lập trình là xác định đúng đắn các giao dịch sao cho mỗi một bảo tồn tính nhất quán của cơ sở dữ liệu.

Đảm bảo tính nguyên tử và tính lâu bền là trách nhiệm của hệ cơ sở dữ liệu nói chung và của thành phần quản trị giao dịch (transaction-management component) nói riêng. Nếu không có sự cố, tất cả giao dịch hoàn tất thành công và