

# CƠ SỞ DỮ LIỆU

Học kỳ 20211

Chương 1, 2

# Nội dung môn học

- Chương 1: Đại cương về các hệ CSDL.
- Chương 2: Các mô hình dữ liệu.
- Chương 3: Ngôn ngữ định nghĩa và thao tác dữ liệu đối với mô hình quan hệ.
- Chương 4: Lý thuyết thiết kế CSDL quan hệ.
- Chương 5: Tối ưu hóa câu truy vấn
- Chương 6: An toàn và toàn vẹn dữ liệu.
- Chương 7: Tổ chức dữ liệu vật lý

# Thông tin

- Thời lượng: 3 tín chỉ
- Yêu cầu: nghe giảng, bài tập lớn, kiểm tra, thi cuối kỳ
- Giáo trình:

Nguyễn Kim Anh, Nguyên lý của các hệ cơ sở dữ liệu, NXB Đại học Quốc gia, Hà Nội, 2004

# Chương 1 - Đại cương về các hệ cơ sở dữ liệu

---

- 1.1. Các hệ thống xử lý tệp truyền thống và những hạn chế của nó.
- 1.2. Các hệ CSDL: khái niệm, khả năng, kiến trúc, người dùng của một hệ quản trị CSDL.
- 1.3. Sự phân loại các hệ CSDL.

# 1.1. Các hệ thống xử lý tệp truyền thống

- Bước khởi đầu của quá trình tin học hóa doanh nghiệp.
- Tập trung vào nhu cầu xử lý dữ liệu của các phòng riêng lẻ trong tổ chức mà không xem xét tổng thể tổ chức này.
- Viết một chương trình mới đối với mỗi ứng dụng đơn lẻ, không có kế hoạch, không có mô hình hướng đến sự tăng trưởng.



# Các hệ thống xử lý tệp truyền thống

- Mỗi chương trình ứng dụng định nghĩa và quản lý các tệp dữ liệu của riêng nó.
- Trước khi xuất hiện các phần mềm hệ quản trị CSDL, trong quá khứ các hệ thống trên cơ sở tệp đã được tạo lập để xử lý một số lượng lớn dữ liệu.

# Hạn chế của các hệ thống xử lý tệp truyền thống

- Dư thừa và không nhất quán dữ liệu
- Khó khăn trong truy nhập dữ liệu
- Cô lập và hạn chế chia sẻ dữ liệu
- Các vấn đề về an toàn và toàn vẹn
- Các vấn đề về độ tin cậy
- Sự phụ thuộc dữ liệu của các chương trình ứng dụng

## 1.2. Các hệ cơ sở dữ liệu

- CSDL (database) là gì ?
- Tại sao phải sử dụng CSDL ?
- Tại sao phải tìm hiểu về các hệ CSDL (database systems) ?



# Ví dụ về quản lý đào tạo

- Thông tin cần quan tâm
  - Khóa học, lớp học, sinh viên, giáo viên, môn học,...
  - Thông tin về sinh viên: thông tin cá nhân, thông tin học tập,...
  - Thông tin về môn học: khối lượng, giáo viên, lịch học,...
- Cần *lưu trữ* những thông tin đa dạng

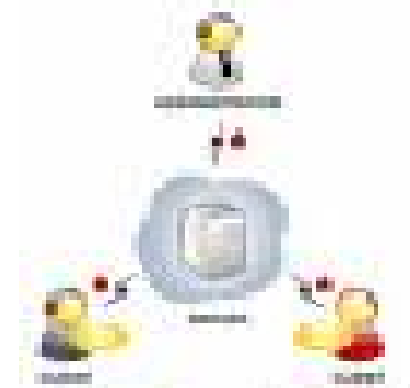
➔ **Cơ sở dữ liệu**



# Ví dụ: khai thác thông tin

- Sinh viên
  - Các môn học trong CTĐT?
  - Điểm thi môn "Cơ sở dữ liệu"?
- Giáo viên
  - Danh sách sinh viên lớp 111565 ?
  - Thời khóa biểu của lớp HTTT-K62 ?
- Giáo vụ
  - Danh sách sinh viên K59 tốt nghiệp loại giỏi ?...

➔ **Phần mềm ứng dụng**



# "Hình dung" về xây dựng một CSDL

- Yêu cầu
  - Lưu trữ thông tin **cần thiết** một cách **chính xác**
  - Truy xuất thông tin **hiệu quả**
- Thực hiện
  - Xác định yêu cầu nghiệp vụ
  - Xác định thông tin cần lưu trữ
  - Xác định cách thức lưu trữ
- Cần công cụ trợ giúp xây dựng một CSDL

**➔ Phần mềm quản trị CSDL**



# Cơ sở dữ liệu (database)

- Là một tập hợp các dữ liệu
  - Biểu diễn một vài khía cạnh nào đó của thế giới thực
  - Có liên hệ logic thống nhất
  - Được thiết kế và bao gồm những dữ liệu phục vụ một mục đích nào đó.
- Là một bộ sưu tập các dữ liệu tác nghiệp được lưu trữ lại và được các hệ ứng dụng của một xí nghiệp cụ thể nào đó sử dụng.

# Hệ quản trị cơ sở dữ liệu (Database Management System-DBMS)

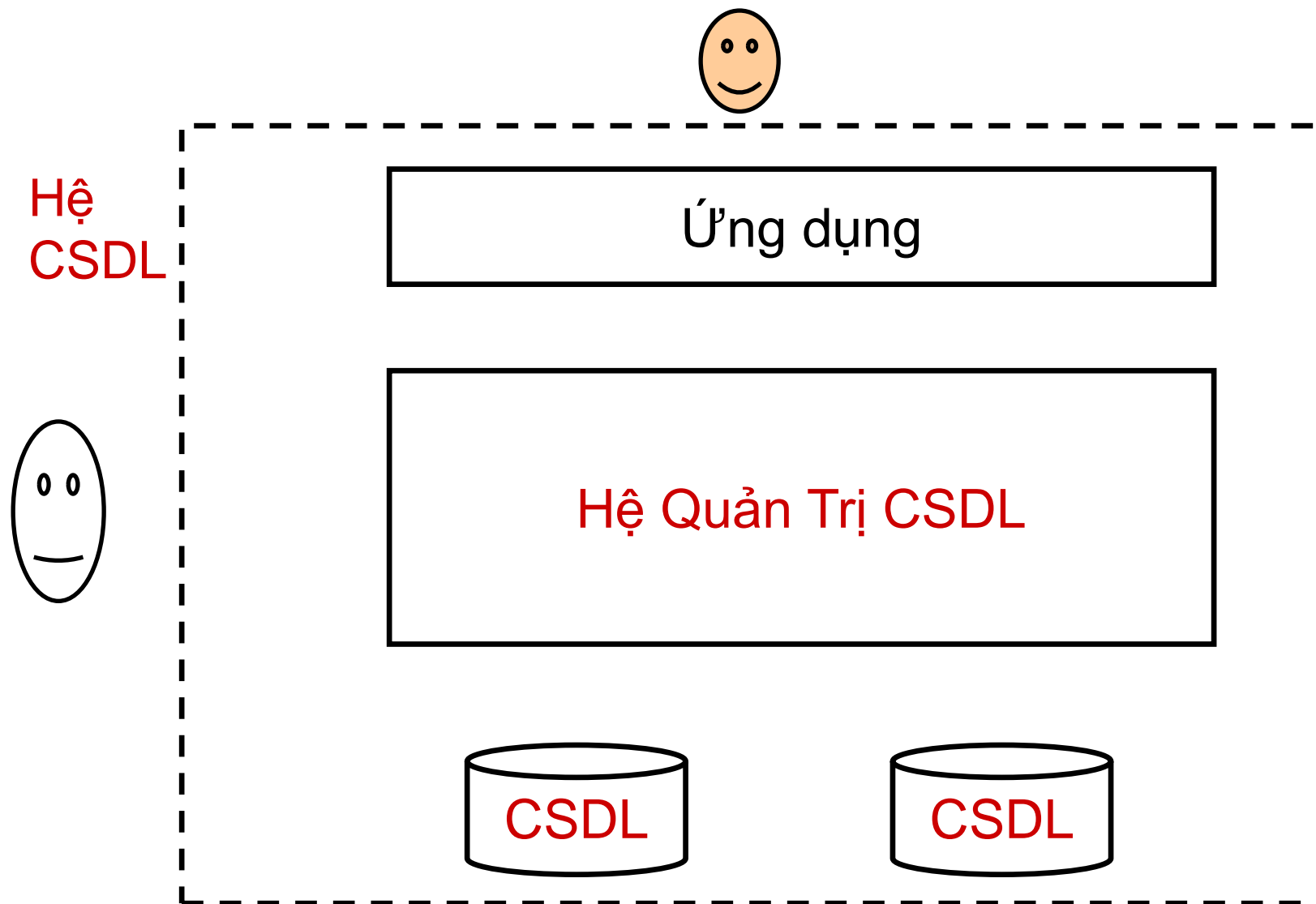
- Là một hệ thống phần mềm cho phép
  - Định nghĩa, tạo lập: xác định kiểu, cấu trúc, ràng buộc dữ liệu, lưu trữ dữ liệu trên các thiết bị nhớ.
  - Thao tác: truy vấn, cập nhật, kết xuất,... các CSDL cho các ứng dụng khác nhau
- Ví dụ: MS SQL Server, mySQL, PostgreSQL, DB2, MS Access, Oracle, FoxPro,...



# Hệ cơ sở dữ liệu

- Là một hệ thống gồm 4 thành phần
  - Hệ quản trị CSDL
  - Phần cứng
  - CSDL và phần mềm ứng dụng
  - Những người sử dụng
- Ví dụ: Hệ quản lý đào tạo, hệ quản lý nhân sự, hệ quản lý kinh doanh,...





# Các tính năng của hệ quản trị CSDL

- Quản lý dữ liệu tồn tại lâu dài
  - Định nghĩa dữ liệu
  - Quản lý lưu trữ
- Truy xuất dữ liệu một cách hiệu quả
  - Biểu diễn các thao tác dữ liệu
  - Xử lý câu hỏi
  - Quản trị giao dịch



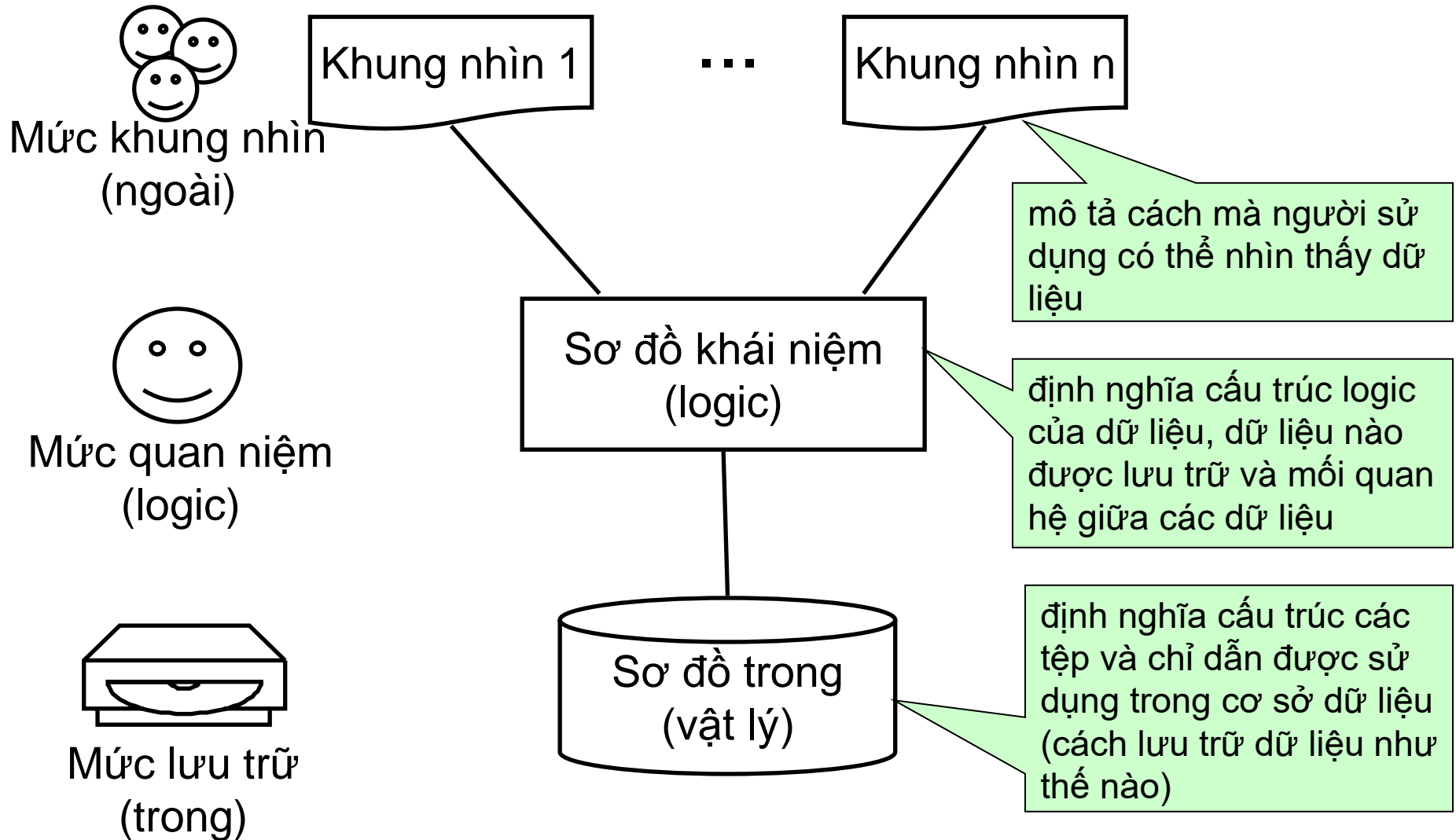
# Các tính năng của hệ quản trị CSDL

- Hỗ trợ ít nhất một mô hình dữ liệu
- Đảm bảo tính độc lập dữ liệu
- Hỗ trợ các ngôn ngữ cấp cao nhất định cho phép người sử dụng định nghĩa cấu trúc của dữ liệu, truy nhập và thao tác dữ liệu
- Điều khiển truy nhập
- Phục hồi dữ liệu

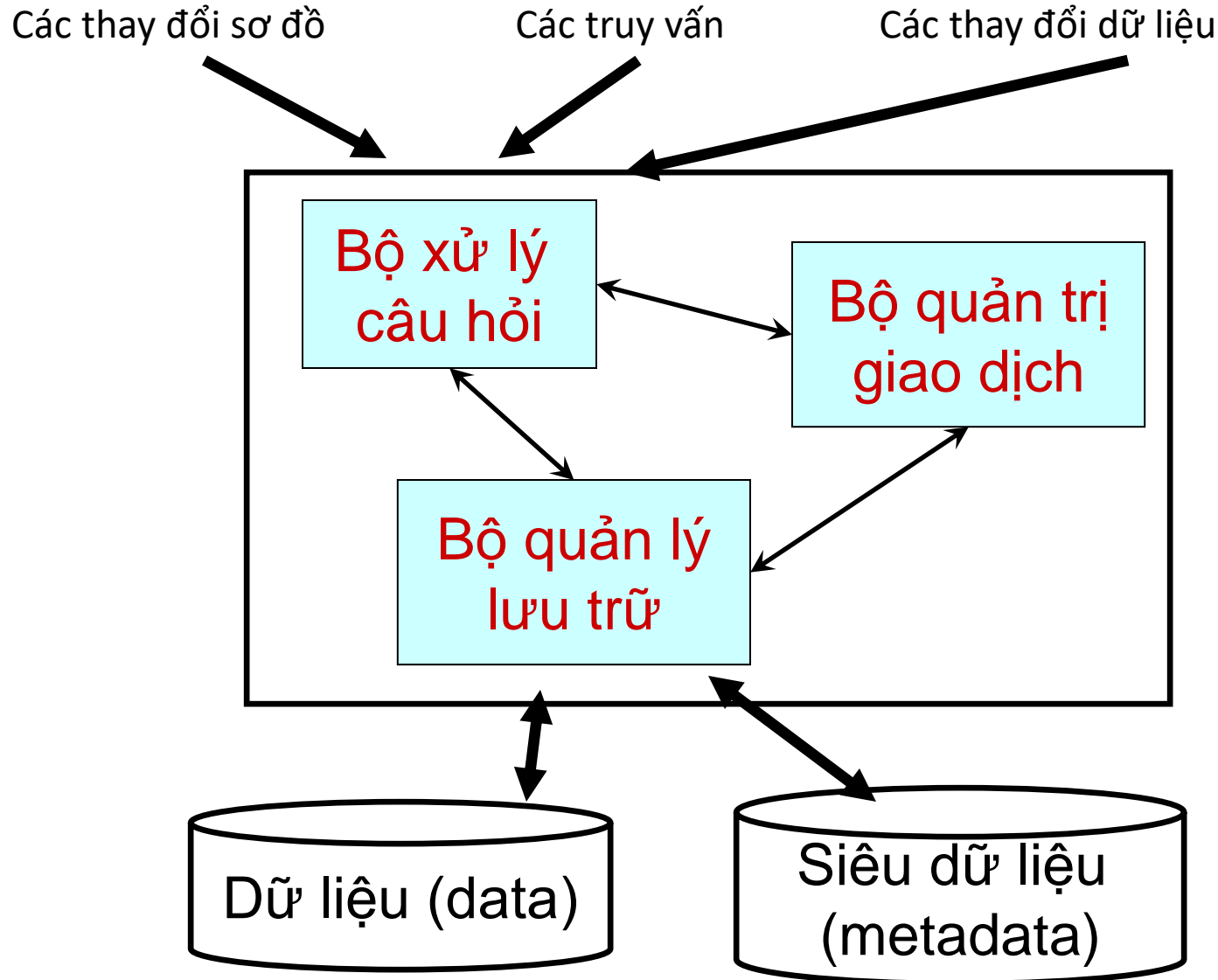
# Các ngôn ngữ

- Ngôn ngữ định nghĩa dữ liệu (**D**ata **D**efinition **L**anguage - DDL)
  - Cấu trúc dữ liệu
  - Mối liên hệ giữa các dữ liệu và quy tắc, ràng buộc áp đặt lên dữ liệu
- Ngôn ngữ thao tác dữ liệu (**D**ata **M**anipulation **L**anguage - DML)
  - Tìm kiếm, thêm, xóa, sửa dữ liệu trong CSDL
- Ngôn ngữ điều khiển dữ liệu (**D**ata **C**ontrol **L**anguage - DCL)
  - Thay đổi cấu trúc của các bảng dữ liệu
  - Khai báo bảo mật thông tin
  - Quyền hạn của người dùng trong khai thác CSDL

# Sự trừu tượng hóa dữ liệu

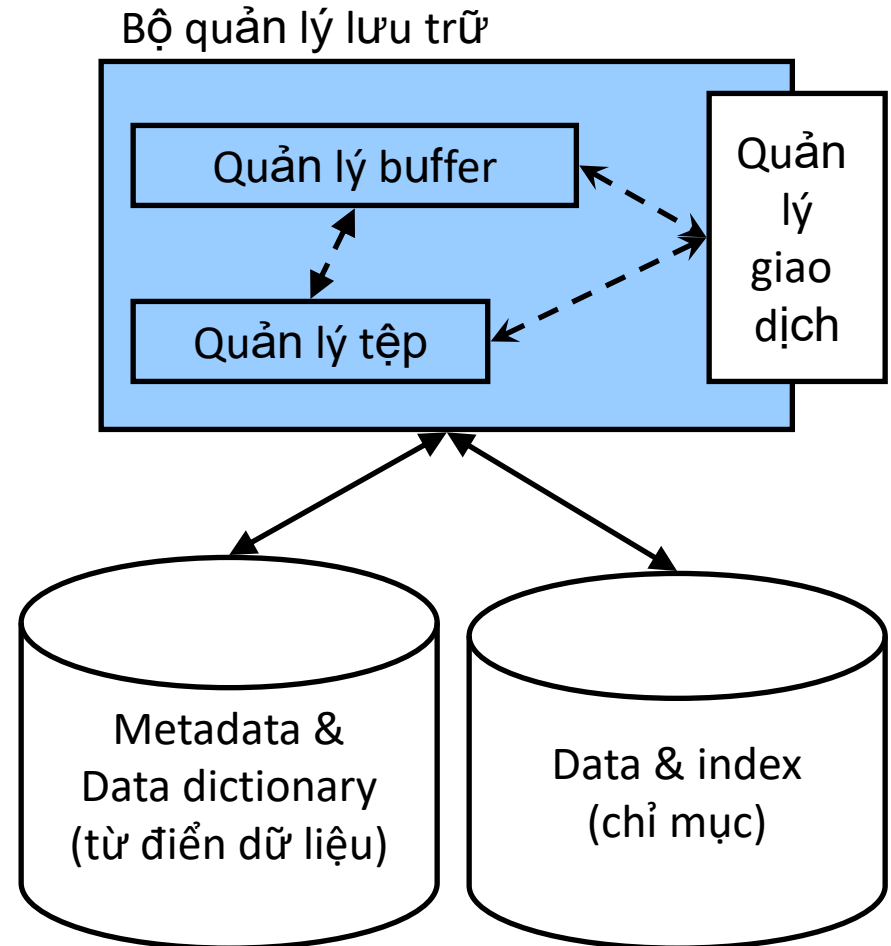


# Kiến trúc của một hệ quản trị CSDL



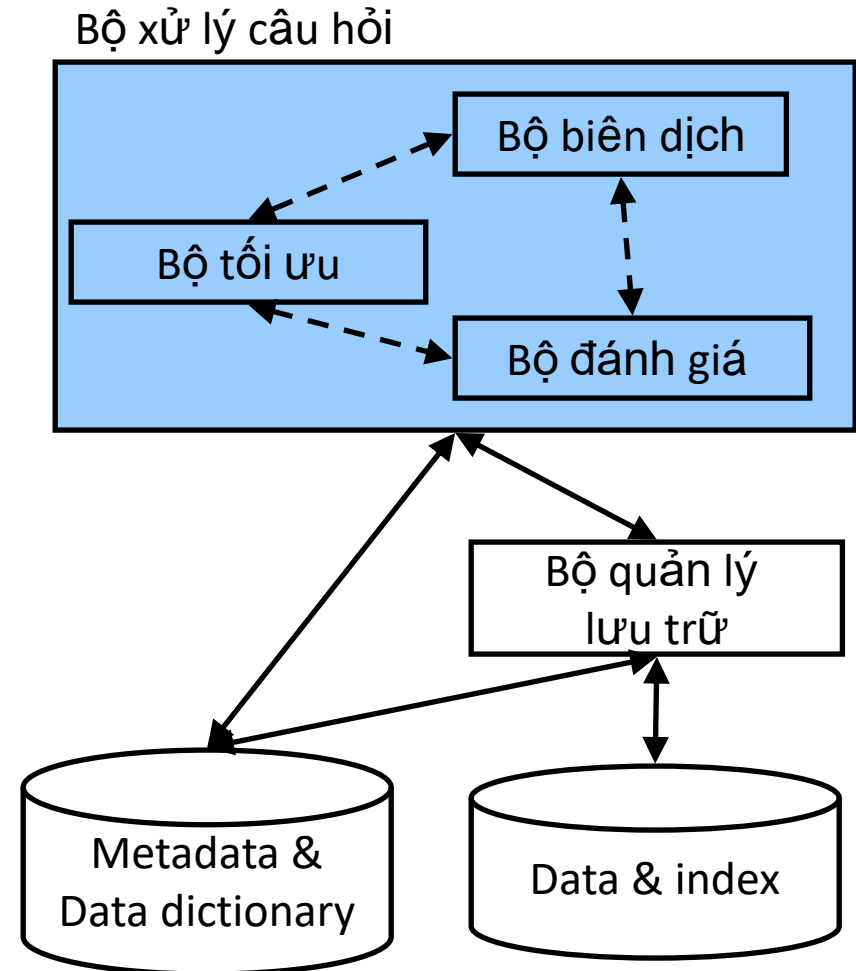
# Quản lý lưu trữ

- Yêu cầu
  - lưu trữ và truy xuất dữ liệu trên các thiết bị nhớ
- Thực hiện
  - Tổ chức tối ưu dữ liệu trên thiết bị nhớ
  - Tương tác hiệu quả với bộ quản lý tệp



# Xử lý câu hỏi

- Yêu cầu
  - Tìm kiếm dữ liệu trả lời cho một yêu cầu truy vấn.
- Thực hiện
  - Biến đổi truy vấn ở mức cao thành các yêu cầu có thể hiểu được bởi hệ CSDL.
  - Lựa chọn một kế hoạch tốt nhất để trả lời truy vấn này.



# Quản trị giao dịch

- Yêu cầu
  - Định nghĩa giao dịch: một tập các thao tác được xử lý như một đơn vị không chia cắt được.
  - Đảm bảo tính đúng đắn và tính nhất quán của dữ liệu.
- Thực hiện
  - Quản lý điều khiển tương tranh.
  - Phát hiện lỗi và phục hồi CSDL

# Người dùng

- **Người thiết kế và cài đặt hệ QTCSDL:** chịu trách nhiệm thiết kế và cài đặt các module của hệ QTCSDL và các giao diện dưới hình thức các gói phần mềm
- **Người phát triển công cụ:** chịu trách nhiệm thiết kế và cài đặt các gói phần mềm hỗ trợ cho việc thiết kế, sử dụng cũng như tăng cường hiệu năng của các hệ CSDL.



# Người dùng

- **Người phân tích hệ thống và phát triển ứng dụng:** chịu trách nhiệm xác định yêu cầu của người dùng cuối, xác định các giao dịch cần thiết để đáp ứng các yêu cầu người dùng. Người lập trình ứng dụng cài đặt những yêu cầu này trong chương trình, kiểm thử, gỡ rối, lập tài liệu cho chương trình
- **Người thiết kế CSDL:** chịu trách nhiệm xác định dữ liệu lưu trữ trong CSDL và cấu trúc biểu diễn và lưu trữ những dữ liệu này

# Người dùng

- **Người sử dụng cuối:** là người khai thác các hệ CSDL
- **Người quản trị CSDL:** chịu trách nhiệm cho phép truy nhập CSDL, điều phối và kiểm tra sử dụng CSDL, quản lý tài nguyên phần cứng và phần mềm khi cần thiết
- **Người bảo trì hệ thống:** là những người quản trị hệ thống chịu trách nhiệm việc hoạt động và bảo trì môi trường (phần cứng và phần mềm) cho hệ CSDL

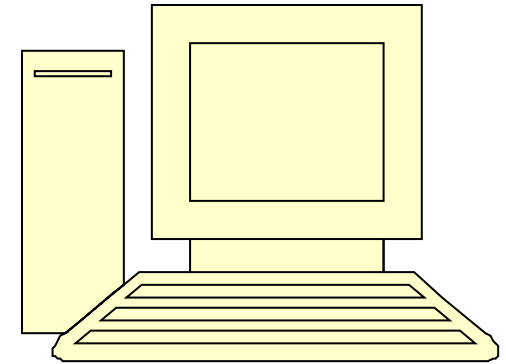
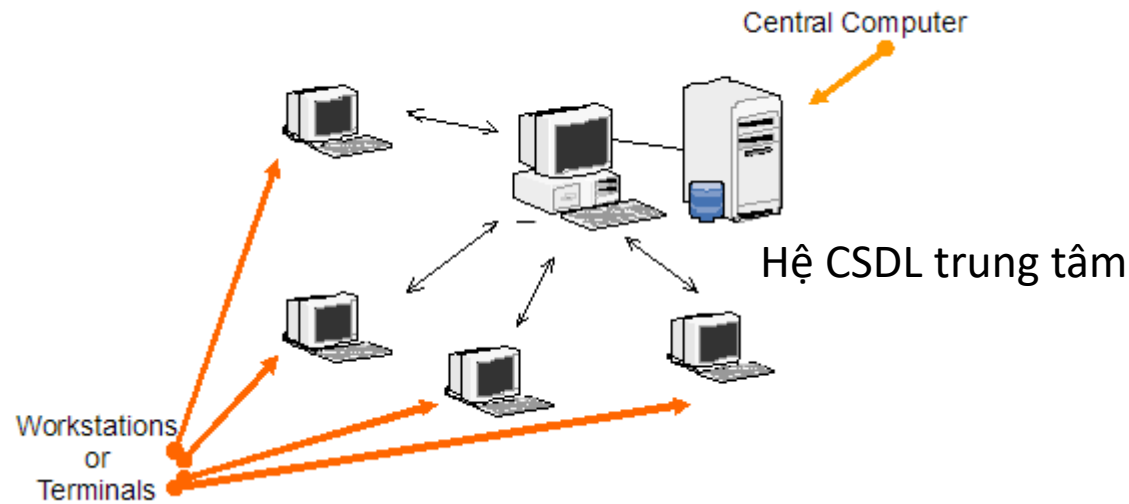
# 1.3. Phân loại các hệ CSDL

- Mô hình dữ liệu
  - MH Mạng, MH phân cấp, MH quan hệ, MH hướng đối tượng, ...
- Số người sử dụng
  - Một người dùng, nhiều người dùng
- Tính phân tán của CSDL
  - Tập trung, Phân tán
- Tính thống nhất của dữ liệu
  - Đồng nhất, Không đồng nhất
- ...

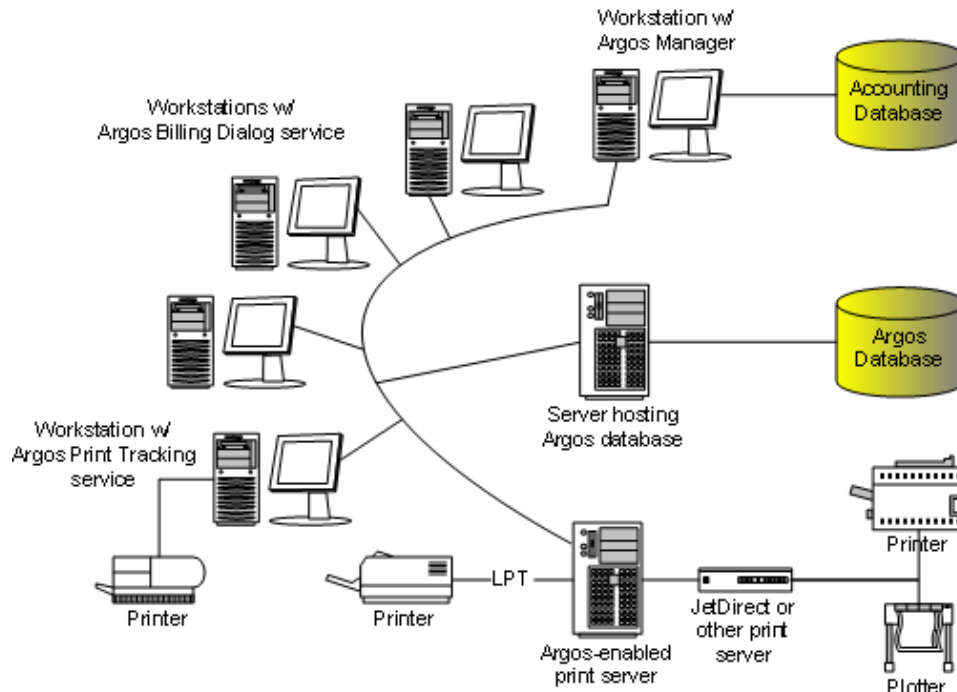
# Các hệ CSDL tập trung

- Hệ CSDL cá nhân: một người sử dụng đơn lẻ vừa thiết kế, tạo lập CSDL, cập nhật, bảo trì dữ liệu, lập và hiển thị báo cáo.
  - đảm nhiệm vai trò: người quản trị CSDL, người viết chương trình ứng dụng, end-user.
- Hệ CSDL trung tâm: dữ liệu được lưu trữ trên một máy tính trung tâm.
- Hệ CSDL khách-chủ:
  - Các máy tính trung tâm lớn → đắt so với các máy nhỏ và máy trạm.
  - Các ứng dụng máy khách truy nhập dữ liệu được quản lý bởi máy chủ.

# Các hệ CSDL tập trung (tiếp)



Hệ CSDL cá nhân

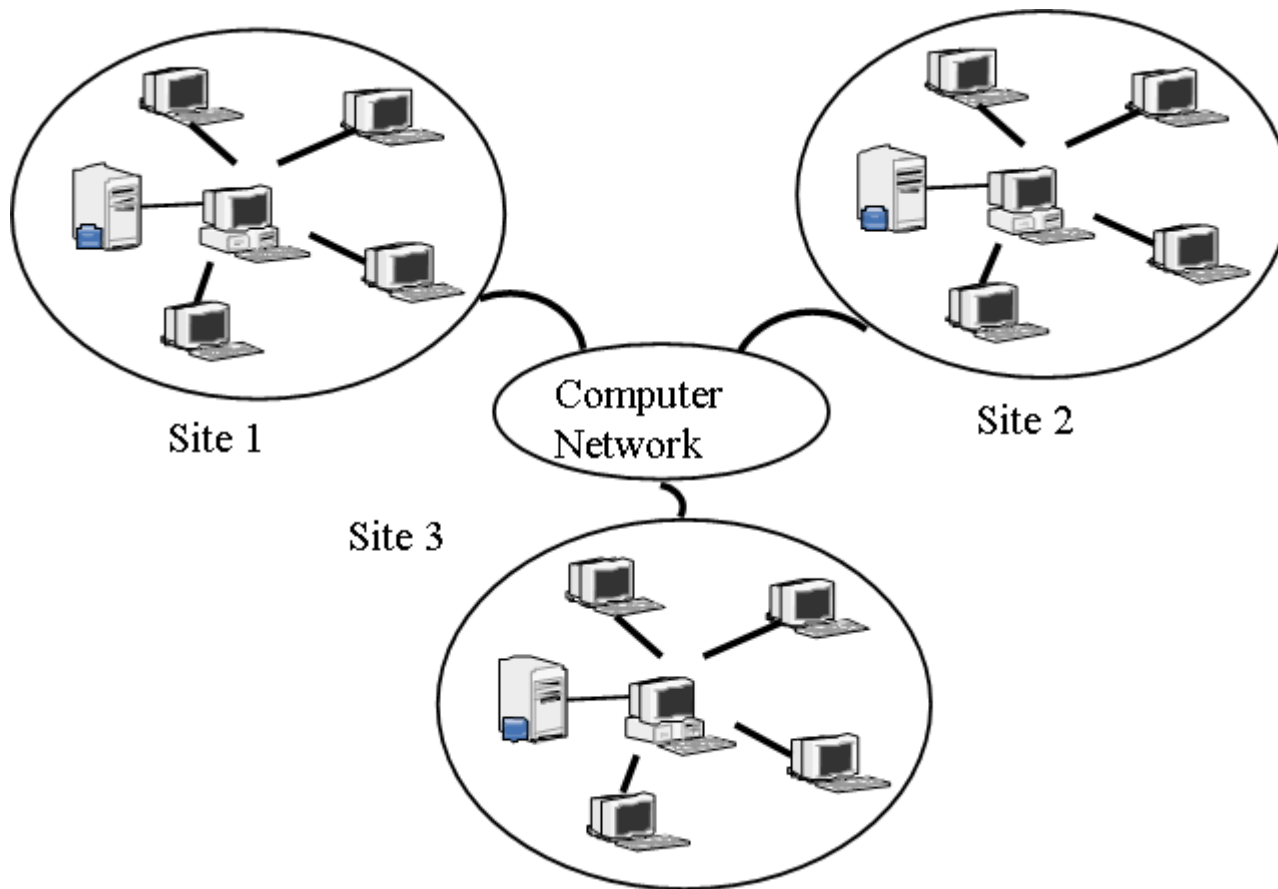


Hệ CSDL khách-chủ

# Các hệ CSDL phân tán

- CSDL phân tán? Là một tập các CSDL có quan hệ logic với nhau nhưng được trải ra trên nhiều trạm làm việc của một mạng máy tính.
- Có 2 tính chất: quan hệ logic và phân tán
- Hệ QTCSDL phân tán: Là một hệ thống phần mềm cho phép tạo lập CSDLPT và điều khiển các truy nhập đối với CSDLPT này.
- Chia ra 2 loại: CSDLPT thuần nhất và không thuần nhất

# Các hệ CSDL phân tán (tiếp)



# Kết luận Chương 1

- CSDL cho phép lưu trữ và khai thác dữ liệu một cách thống nhất và hiệu quả (đặc biệt trong trường hợp khối lượng dữ liệu lớn).
- Sự trừu tượng về dữ liệu và tính độc lập dữ liệu cho phép phát triển ứng dụng “dễ dàng hơn”.
- Hệ quản trị CSDL cung cấp các công cụ hữu hiệu trợ giúp việc tạo lập CSDL và phát triển ứng dụng





# Chương 2 – Các mô hình dữ liệu

---

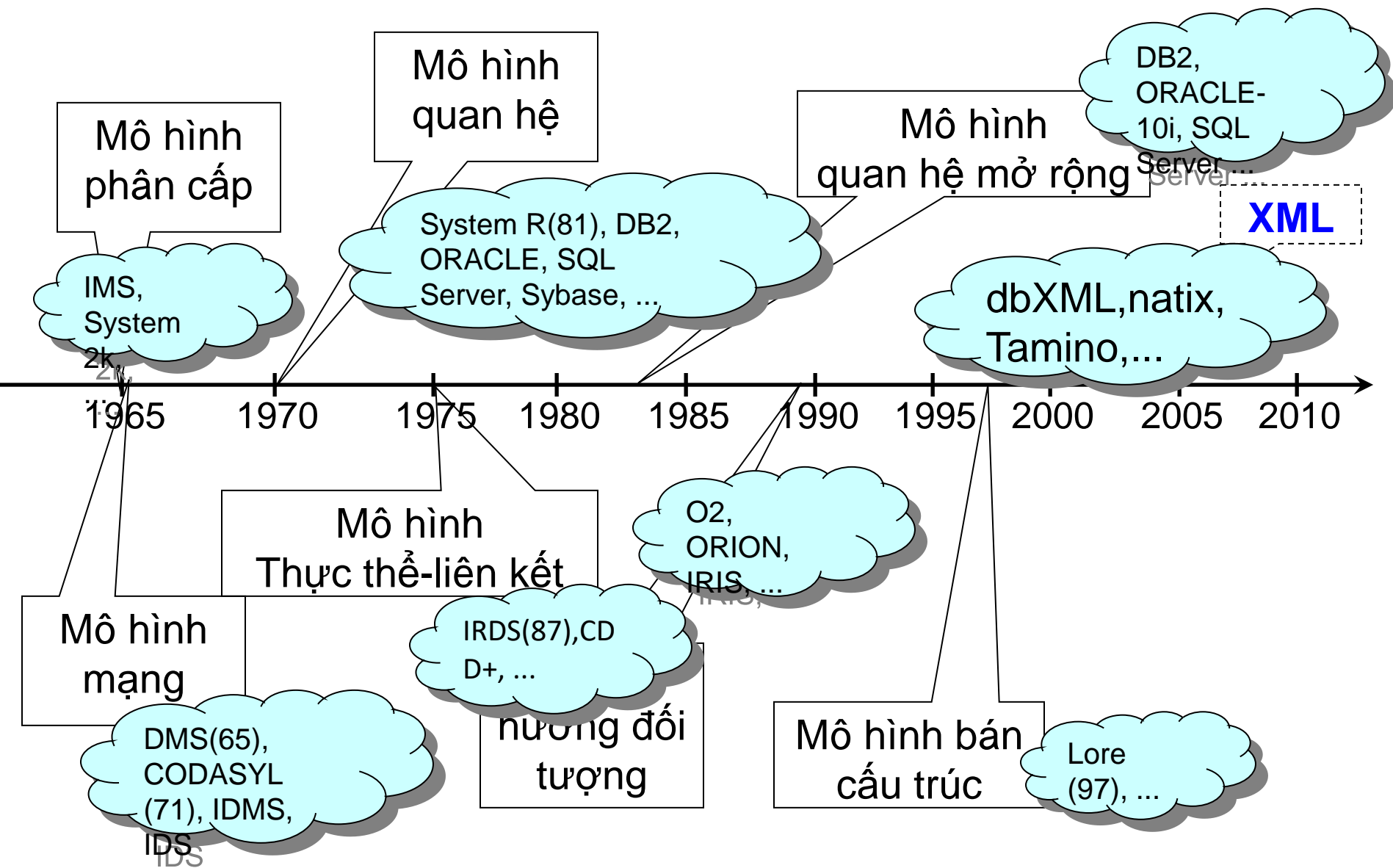
- Tổng quan về mô hình dữ liệu
- Mô hình phân cấp
- Mô hình mạng
- Mô hình quan hệ
- Mô hình thực thể liên kết
- Mô hình hướng đối tượng

# Tổng quan về mô hình dữ liệu

- Mô hình dữ liệu [Codd, 1980] gồm:
  - Một tập hợp các cấu trúc của dữ liệu
  - Một tập hợp các phép toán để thao tác với các dữ liệu
  - Một tập hợp các ràng buộc về dữ liệu
- Mô hình dữ liệu là một tập hợp các khái niệm dùng để mô tả:
  - Dữ liệu
  - Ngữ nghĩa của dữ liệu
  - Các mối quan hệ trong dữ liệu
  - Các ràng buộc dữ liệu

# Tổng quan về mô hình dữ liệu

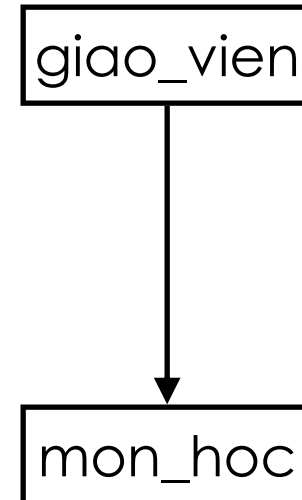
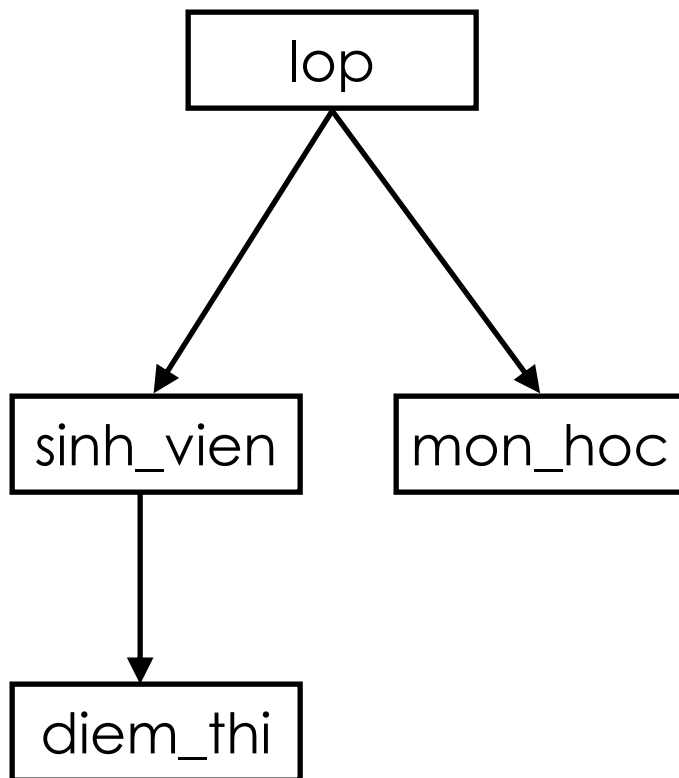
- Nhiều mô hình còn bao gồm cả một tập các phép toán để thao tác các dữ liệu
- Mô hình thuộc dạng ngữ nghĩa: tập trung về ngữ nghĩa của dữ liệu như mô hình thực thể liên kết, sử dụng để hỗ trợ người dùng có cái nhìn khái quát về dữ liệu
- Mô hình thuộc dạng khái niệm: tập trung vào cách thức tổ chức dữ liệu tại mức khái niệm như mô hình mạng, mô hình liên kết, mô hình quan hệ, độc lập với DBMS và hệ thống phần cứng để cài đặt cơ sở dữ liệu



# Mô hình dữ liệu phân cấp (*Hierarchical data model*)

- Ra đời những năm 60-65
- Biểu diễn bằng cây
  - Quan hệ cha-con
  - Mỗi nút có 1 cha duy nhất
  - 1 CSDL = 1 tập các cây = 1 rừng
- Các khái niệm cơ bản
  - Bản ghi
  - Móc nối
  - Các phép toán: GET, GET UNIQUE, GET NEXT, GET NEXT WITHIN PARENT,...

# Ví dụ



- Ưu điểm
  - Dễ xây dựng và thao tác
  - Tương thích với các lĩnh vực tổ chức phân cấp
  - Ngôn ngữ thao tác đơn giản: duyệt cây.
- Nhược điểm:
  - Sự lặp lại của các kiểu bản ghi  $\rightarrow$  dữ liệu dư thừa và không nhất quán.
    - Giải pháp: bản ghi ảo
  - Hạn chế trong biểu diễn ngữ nghĩa của các mối nối giữa các bản ghi (chỉ cho phép quan hệ 1-n)

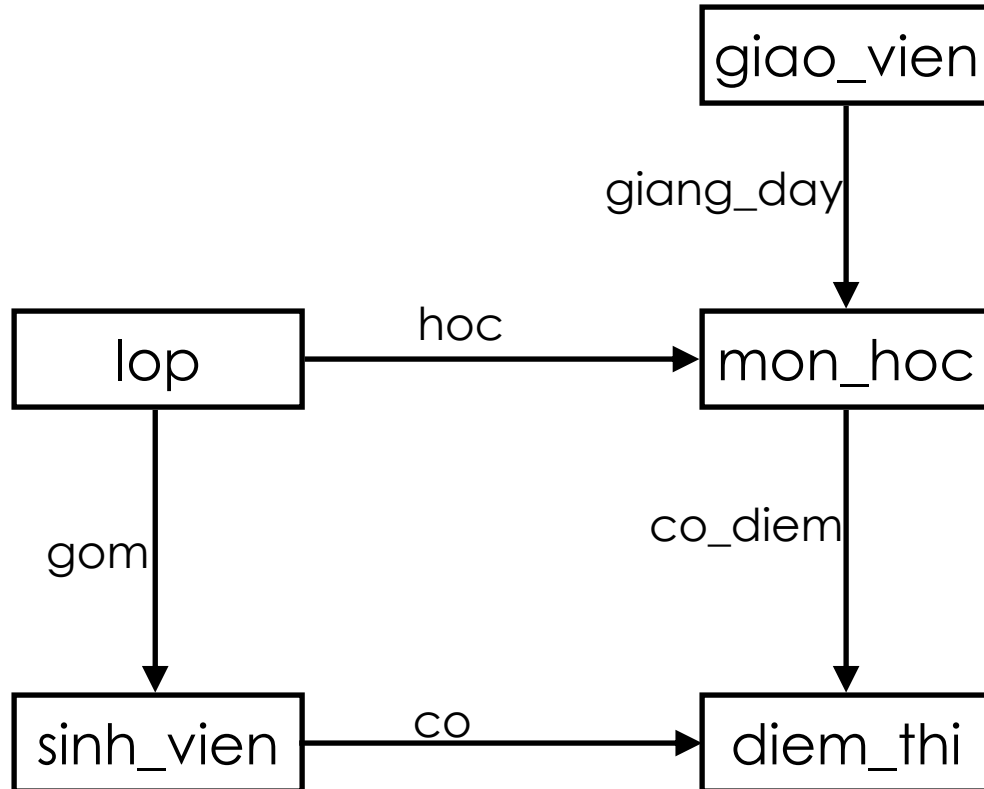


# Mô hình dữ liệu mạng

## *(Network data model)*

- Sự ra đời: Sử dụng phổ biến từ những năm 60, được định nghĩa lại vào năm 71
- Biểu diễn bằng đồ thị có hướng
- Các khái niệm cơ bản
  - Tập bản ghi (record): Kiểu bản ghi (record type); Các trường (field)
  - Móc nối: Tên của móc nối; Chủ (owner) – thành viên (member): theo hướng của móc nối; Kiểu móc nối: 1-1, 1-n, đệ quy
  - Các phép toán
    - Duyệt: FIND, FIND member, FIND owner, FIND NEXT
    - Thủ tục: GET

# Ví dụ



- Ưu điểm
  - Đơn giản
  - Có thể biểu diễn các ngữ nghĩa đa dạng với kiểu bản ghi và kiểu móc nối
  - Truy vấn thông qua phép duyệt đồ thị (navigation)
- Nhược điểm:
  - Số lượng các con trỏ lớn
  - Hạn chế trong biểu diễn ngữ nghĩa của các móc nối giữa các bản ghi

# Mô hình dữ liệu quan hệ

- Sự ra đời: vào năm 1970[Codd, 1970]
- Dữ liệu được biểu diễn dưới dạng bảng
- Là mô hình dữ liệu khái niệm phổ biến cho đến tận thời điểm hiện tại
- Dựa trên lý thuyết toán học, đồng thời cũng gần với cấu trúc tệp và cấu trúc dữ liệu nên có hai loại thuật ngữ liên quan:
  - Thuật ngữ toán học: quan hệ, bộ, thuộc tính
  - Thuật ngữ hướng dữ liệu: bảng, bản ghi, trường

# Ví dụ mô hình dữ liệu quan hệ

## MON\_HOC

maHP	tenHP	soTC
CNTT01	Nhập môn CSDL	4
CNTT02	Truyền DL và mạng	4
CNTT03	Phân tích và thiết kế hệ thống	4
HTTT01	Quản lý dự án	3

## LOP

malop	lop	khoa	GVCN	loptruong
IT4	Tin 4	CNTT	Ng. V. Anh	Trần T. Bình
IT5	Tin 5	CNTT	Lê A. Văn	Ng. Đ. Trung
IT6	Tin 6	CNTT	Ng. T. Thảo	Trần M. Quế
IT7	Tin 7	CNTT	Ng. V. Quý	Ng. T. Phương

## SINH\_VIEN

maSV	tenSV	ngaysinh	gt	diachi	malop
SV0011	Trần T. Bình	1/4/1981	0	21 T. Q. B	IT4
SV0025	Ng. Đ. Trung	3/2/1980	1	56 Đ. C. V	IT5
SV0067	Trần M. Quế	26/3/1982	0	45 H. B. T	IT6
SV0034	Ng. T. Phương	29/2/1980	0	86 L. T. N	IT7

# Mô hình dữ liệu quan hệ – Các khái niệm

- *Thuộc tính* (~trường): là các đặc tính của một đối tượng
- Mỗi thuộc tính được xác định trên một miền giá trị nhất định gọi là *miền thuộc tính*
- Ví dụ:
  - Sinhviên (MãSV, TênSV, Nămsinh, GiớiTính, ĐịaChỉ)
  - $\text{dom}(\text{MãSV}) = \{\text{char}(5)\}$
  - $\text{dom}(\text{TênSV}) = \{\text{char}(30)\}$
  - $\text{dom}(\text{Nămsinh}) = \{\text{date}\}$
  - $\text{dom}(\text{GiớiTính}) = \{0, 1\}$
  - $\text{dom}(\text{ĐịaChỉ}) = \{\text{char}(50)\}$

- **Quan hệ** ( $\sim$ bảng): Cho  $n$  miền giá trị  $D_1, D_2, \dots, D_n$  không nhất thiết phân biệt,  $r$  là một quan hệ trên  $n$  miền giá trị đó nếu  $r$  là một tập các  $n$ -bộ  $(d_1, d_2, \dots, d_n)$  sao cho  $d_i \in D_i$
- Một **quan hệ** có thể được biểu diễn dưới dạng một **bảng** trong đó một **dòng** trong bảng tương đương với một **bộ**, một **cột** trong bảng tương đương với một **thuộc tính** của quan hệ
- **Bậc** của một quan hệ là số các thuộc tính trong quan hệ
- **Lực lượng** của một quan hệ là số các bộ trong quan hệ

# Mô hình dữ liệu quan hệ – Các định nghĩa

- **Định nghĩa:** Cho  $U = \{A_1, A_2, \dots, A_n\}$  là một tập hữu hạn các thuộc tính trong đó  $\text{dom}(A_i) = D_i$ ,  $r$  là quan hệ trên tập thuộc tính  $U$  ký hiệu là  $r(U)$  nếu:

$$r \subseteq D_1 \times D_2 \times \dots \times D_n$$

- $U$  được gọi là sơ đồ quan hệ (lược đồ quan hệ)



- Định nghĩa **Khoá** của quan hệ  $r$  trên tập thuộc tính  $U = \{A_1, A_2, \dots, A_n\}$  là một tập  $K \subseteq U$  sao cho với bất kỳ 2 bộ  $t_1, t_2$  thuộc  $r$  đều tồn tại một thuộc tính  $A$  thuộc  $K$  mà  $t_1[A] \neq t_2[A]$
- Một quan hệ có thể có nhiều khoá
- Nếu  $K$  là khoá của  $r$  thì mọi  $K'$  sao cho  $K \subseteq K'$  đều là khoá của  $r$ .  $K'$  được gọi là **siêu khoá** của  $r$
- **Định nghĩa:**  $K$  là **khoá tối thiểu** của  $r$  nếu  $K$  là một khoá của  $r$  và bất kỳ tập con thực sự nào của  $K$  đều không phải là khoá của  $r$
- **Định nghĩa:** Một tập con  $K \subseteq U$  được gọi là **khoá ngoài** của quan hệ  $r(U)$  tham chiếu đến một quan hệ  $r'$  nếu  $K$  là khoá chính của  $r'$

# Ví dụ

- Quan hệ: SinhViên, Lớp
- Siêu khoá: {MãSV, HọTên};
- Khoá tối thiểu: {MãSV}; {HọTên, NămSinh}
- Khoá ngoài: TênLớp nếu coi nó là khoá chính của quan hệ Lớp

MãSV	HọTên	NămSinh	GiớiTính	TênLớp
001	Nguyễn Văn An	1999	1	KHMT1
002	Nguyễn Văn An	1998	1	KHMT2
003	Lê Văn Cường	1997	1	KHMT2
004	Nguyễn Thùy Linh	1997	0	KTMT1

Tên Lớp	SiSố
KHMT1	60
KHMT2	65
KTMT1	58

# Nhận xét

- Ưu điểm
  - Dựa trên lý thuyết tập hợp
  - Khả năng tối ưu hoá các xử lý phong phú
- Nhược điểm
  - Hạn chế trong biểu diễn ngữ nghĩa
  - Cấu trúc dữ liệu không linh hoạt

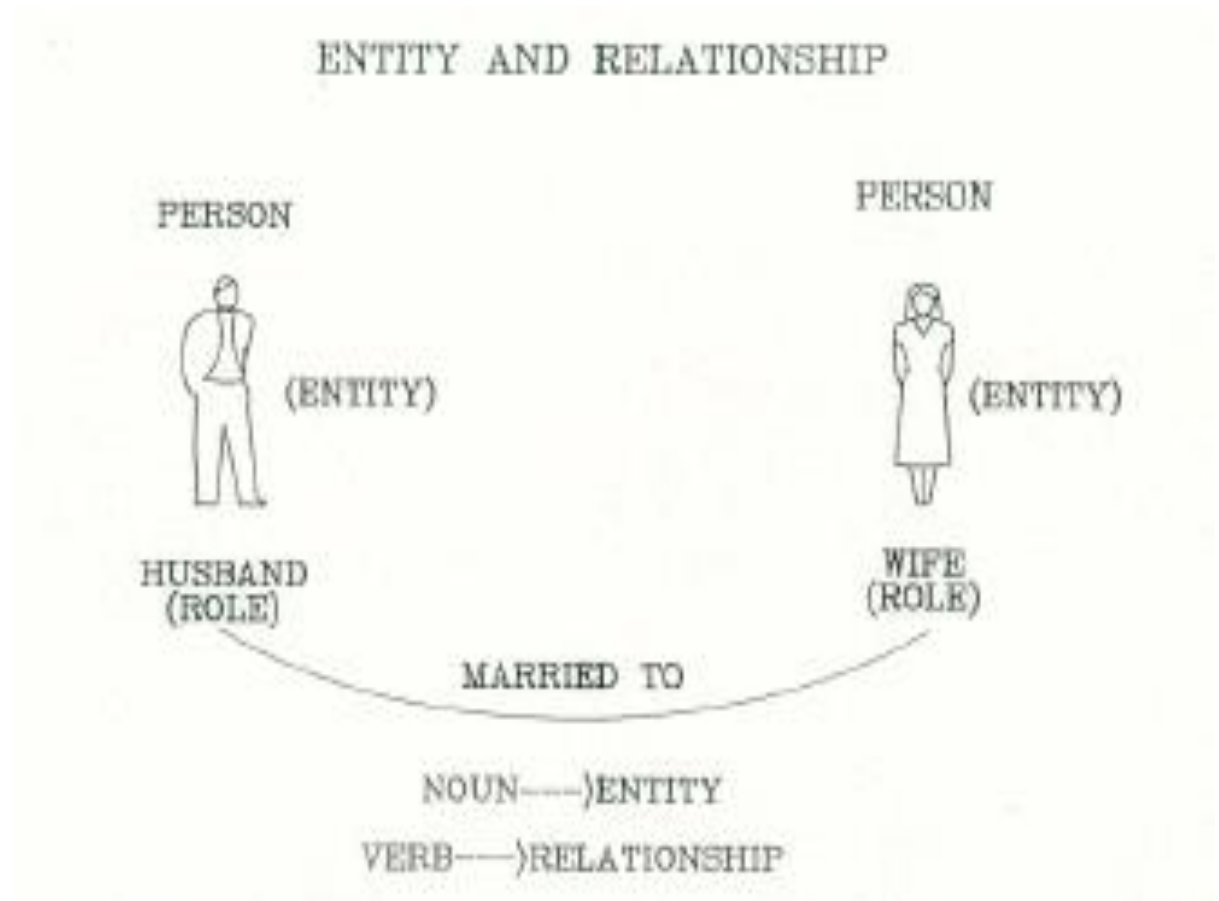
# Mô hình thực thể liên kết

## *(Entity-Relationship data model)*

- Cho phép mô tả các dữ liệu có liên quan trong một xí nghiệp trong thế giới thực dưới dạng các đối tượng và các mối quan hệ của chúng.
- Được sử dụng cho bước đầu thiết kế CSDL, làm nền tảng để ánh xạ sang một mô hình khái niệm nào đó mà Hệ quản trị CSDL sẽ sử dụng
- Trong mô hình thực thể liên kết, CSDL được mô hình hóa như là:
  - Một tập hợp các thực thể
  - Liên hệ giữa các thực thể này

# Mô hình thực thể liên kết – Các khái niệm

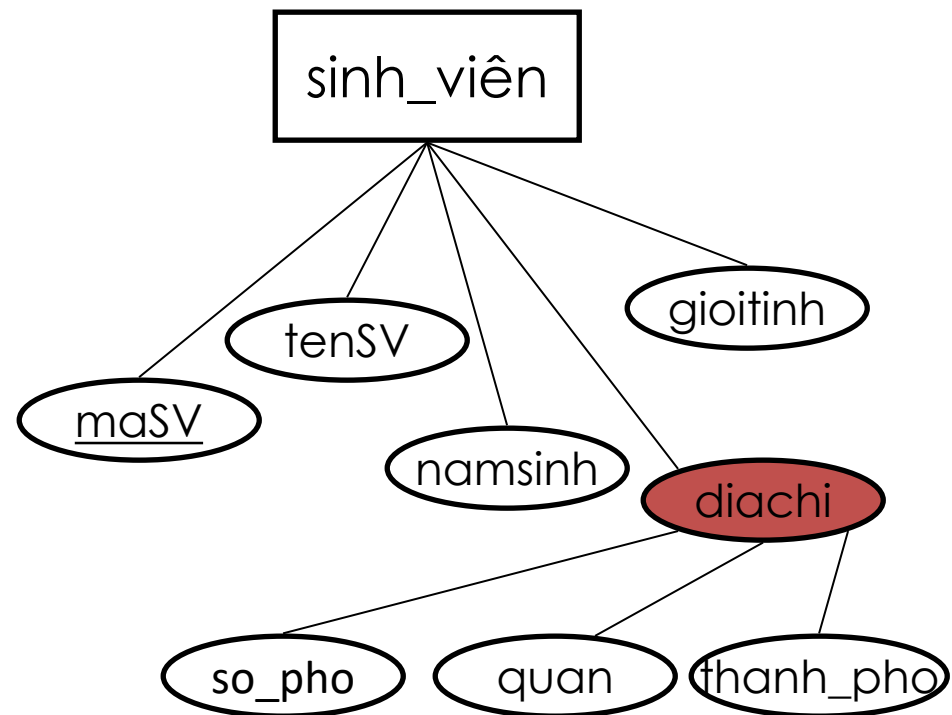
- Thực thể, tập thực thể
- Thuộc tính
- Khoá
- Liên kết, tập liên kết



- **Thực thể**: một đối tượng trong thế giới thực, tồn tại độc lập và phân biệt được với các đối tượng khác
- **Tập thực thể**: một tập hợp các thực thể có tính chất giống nhau
- Ví dụ: – Thực thể: một sinh viên, một lớp  
– Tập thực thể: toàn thể sinh viên của 1 lớp, toàn thể các lớp của 1 khoa
- **Thuộc tính** là đặc tính của một tập thực thể  
– Tập thực thể **SinhViên** có các thuộc tính như: **TênSV**, **NămSinh**,...
- Mỗi thực thể trong tập thực thể có một giá trị đặc tính nằm trong **miền giá trị của thuộc tính**  
– Sinh viên 1 có: Họ tên là Nguyễn Hải Anh, Năm sinh 1980

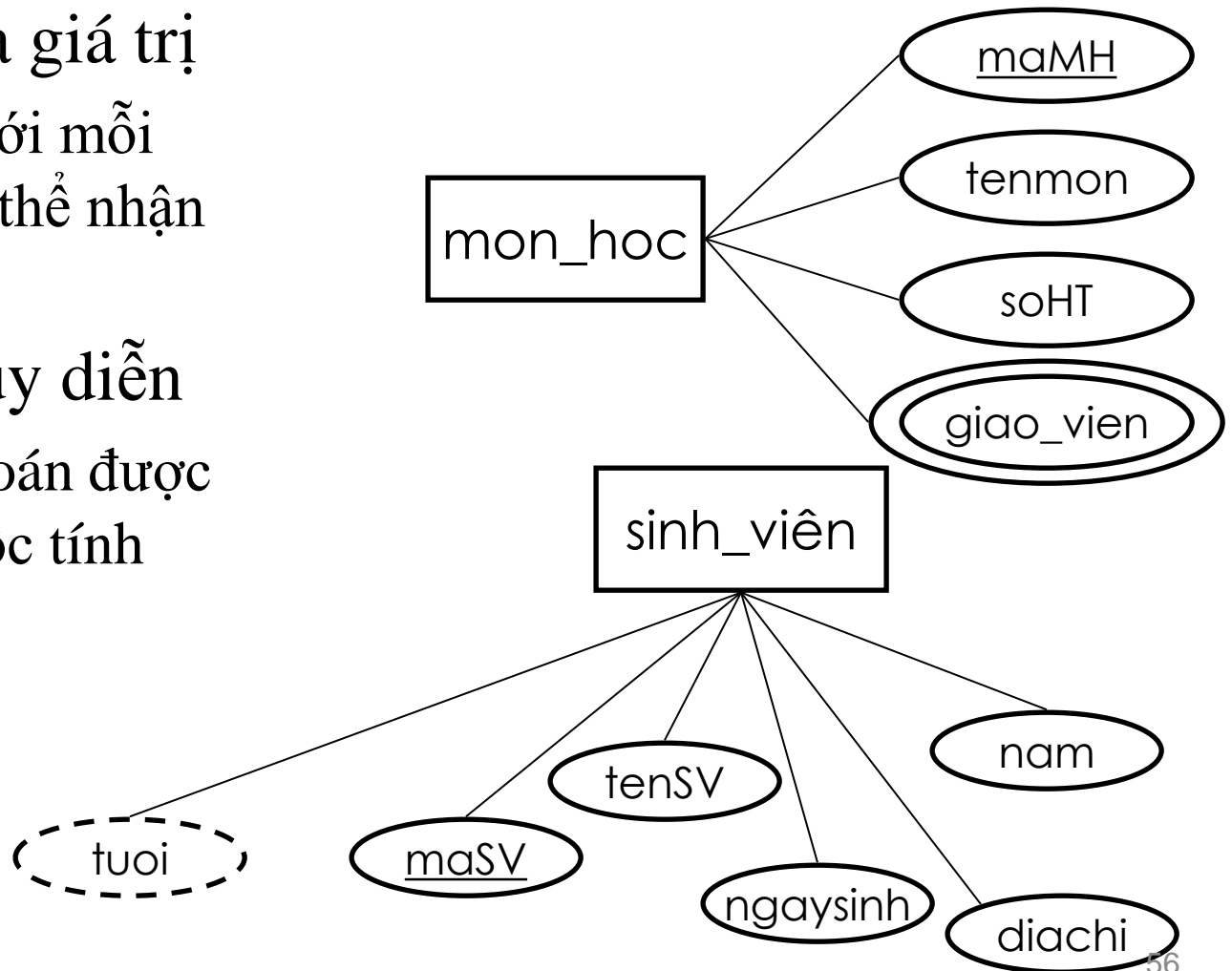
# Kiểu thuộc tính

- Thuộc tính đơn giản (thuộc tính nguyên tố)
  - có kiểu dữ liệu nguyên tố
- Thuộc tính phức
  - có kiểu phức, định nghĩa bởi các thuộc tính khác



# Kiểu thuộc tính

- Thuộc tính đa giá trị
  - tương ứng với mỗi thực thể, có thể nhận nhiều giá trị
- Thuộc tính suy diễn
  - có thể tính toán được từ (các) thuộc tính khác



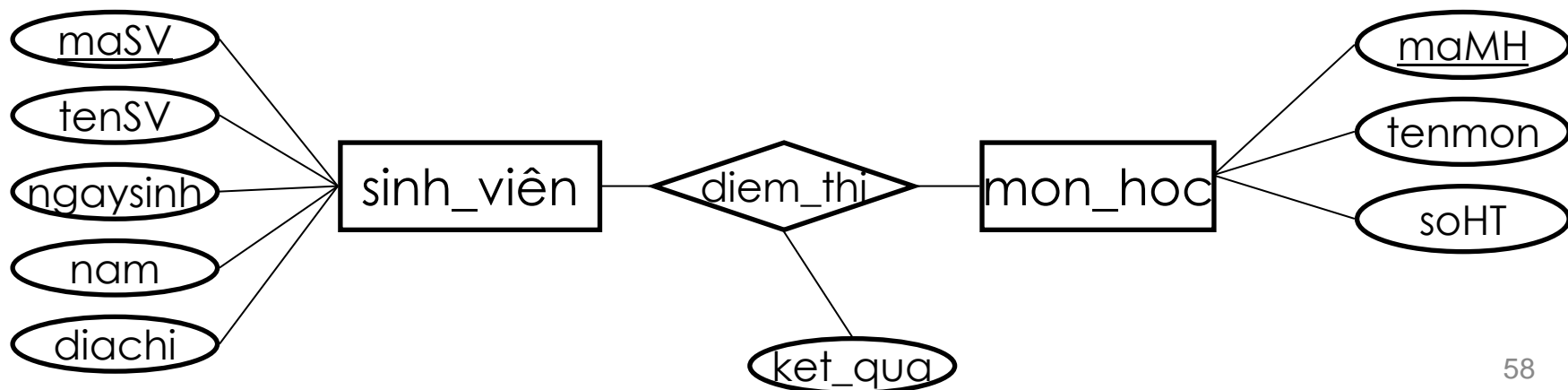


# Khóa

- Một hay một tập thuộc tính mà giá trị của chúng có thể xác định duy nhất một thực thể trong tập thực thể
  - Tập thực thể Sinh Viên có thể dùng MãSV làm khóa
- Khóa gồm nhiều thuộc tính thì gọi là **khóa phức**
- Một tập thực thể có thể có nhiều khóa nhưng chỉ một trong số các khóa được chọn làm **khóa chính**
- Trong sơ đồ ER, thuộc tính nào được chọn làm khóa chính sẽ được **gạch chân**

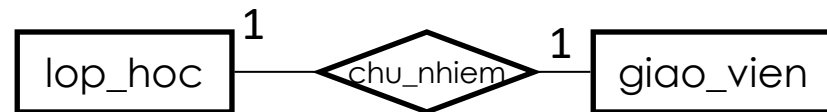
# Liên kết - Tập liên kết

- Một **liên kết** là một **mối liên hệ có nghĩa** giữa nhiều thực thể
  - Cho một thực thể SinhViên1 và LớpA, liên kết ThànhViên chỉ ra rằng SinhViên1 là 1 thành viên của LớpA
- **Tập liên kết** là một tập hợp các liên kết cùng kiểu
  - Giữa tập thực thể SinhViên và Lớp có 1 tập liên kết ThànhViên, chỉ ra rằng mỗi sinh viên đều là thành viên của 1 lớp nào đó
- Một liên kết có thể có thuộc tính

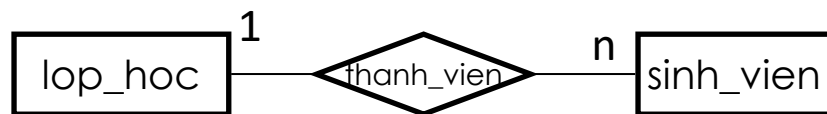


# Ràng buộc của kết nối

- **1-1**: Liên kết 1 thực thể của một tập thực thể với nhiều nhất 1 thực thể của tập thực thể khác



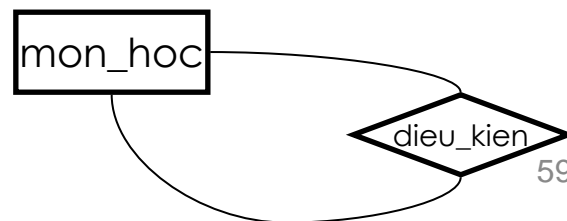
- **1-n**: Liên kết 1 thực thể của một tập thực thể với nhiều thực thể của tập thực thể khác



- **n-n**: Liên kết 1 thực thể của một tập thực thể với nhiều thực thể của tập thực thể khác và ngược lại

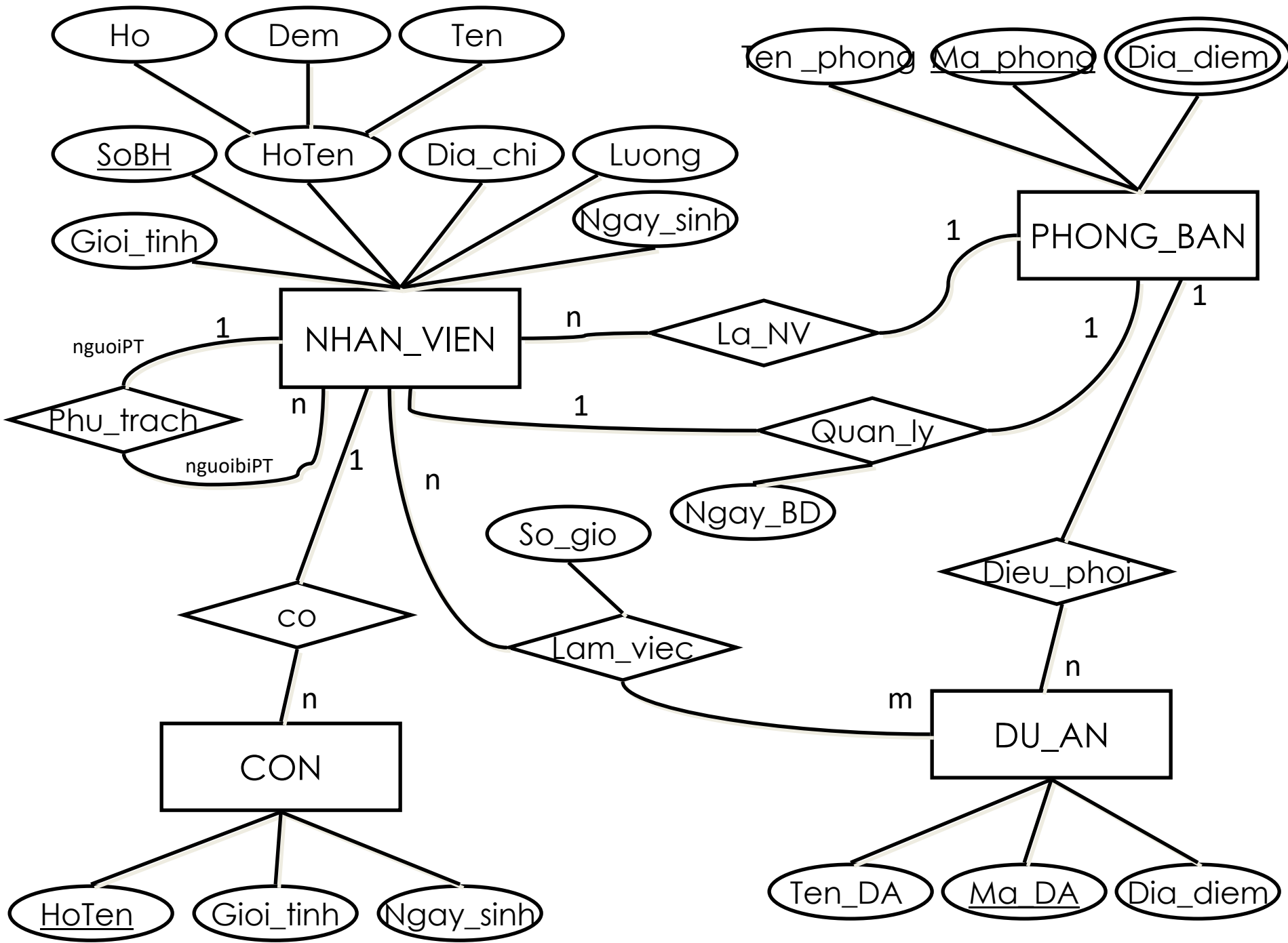


- **đệ quy**: Liên kết giữa các thực thể cùng kiểu



# Cách thiết lập sơ đồ thực thể - liên kết

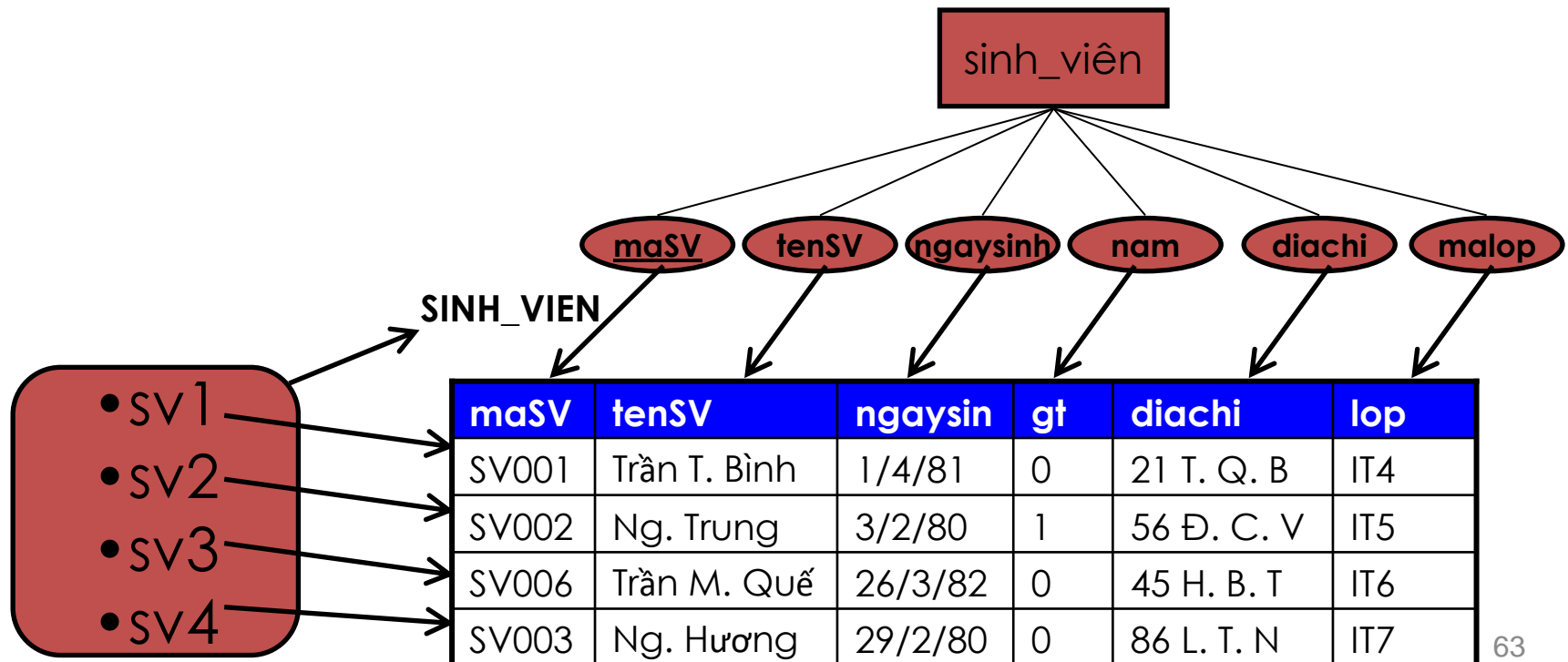
- Bước 1: Xác định các thực thể
- Bước 2: Xác định các liên kết giữa các thực thể
  - Bậc của liên kết
  - Ràng buộc (1-1, 1-n, n-n, đệ quy)



# Biến đổi sơ đồ thực thể liên kết sang sơ đồ quan hệ

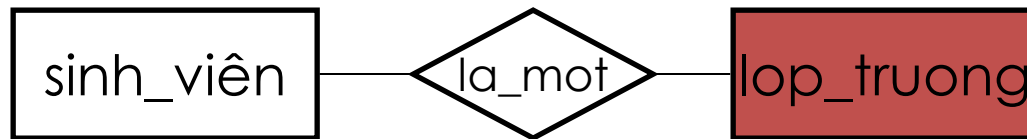
- Biến đổi tập các thực thể
- Biến đổi các liên kết
- Các khoá của các sơ đồ quan hệ
- Các sơ đồ quan hệ với khoá chung

- **Bước 1:** (biến đổi các tập thực thể) 1 tập thực thể  $\rightarrow$  1 quan hệ
  - thuộc tính  $\rightarrow$  thuộc tính (trường)
  - 1 thực thể  $\rightarrow$  1 bộ
  - khoá của tập thực thể  $\rightarrow$  khoá của quan hệ



- **Bước 2:** (biến đổi các tập thực thể) 1 tập thực thể xác định từ tập thực thể khác (E) qua 1 liên kết  $\rightarrow$  1 quan hệ chứa khoá của E:

LOPTRUONG(maSV)





- **Bước 3:** (Biến đổi các liên kết) Liên kết 1-1

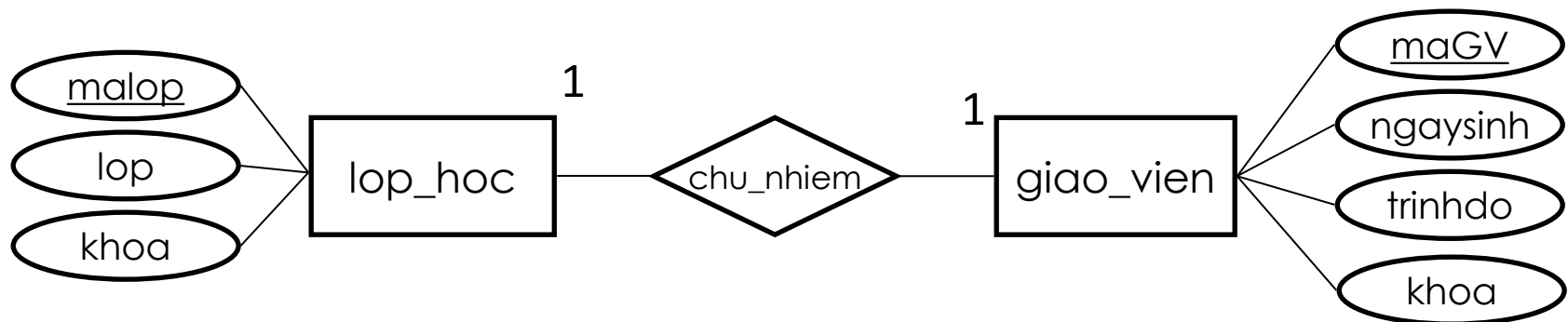
- Thêm 1 quan hệ mới xác định bởi các thuộc tính nằm trong khoá của các thực thể có liên quan

$\text{CHU\_NHIEM\_LOP}(\underline{\text{malop}}, \underline{\text{maGV}})$

*hoặc*

- Dùng khoá ngoài

$\text{LOP\_HOC}(\underline{\text{malop}}, \text{lop}, \text{khoa}, \text{maGV})$



- **Bước 4:** Liên kết 1-n

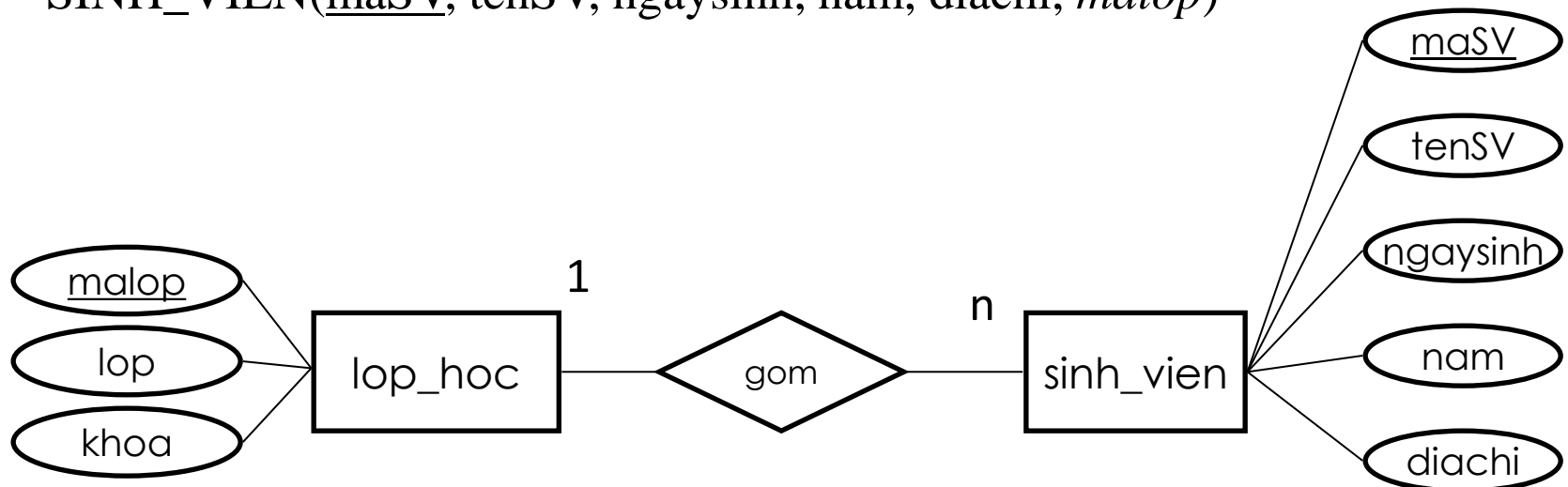
- Thêm 1 quan hệ mới xác định bởi các thuộc tính nằm trong khoá của các thực thể có liên quan

SINHVIEN\_LOP(malop, maSV)

hoặc

- Dùng khoá ngoài: thêm khoá chính của quan hệ bên 1 vào quan hệ bên n làm khoá ngoài

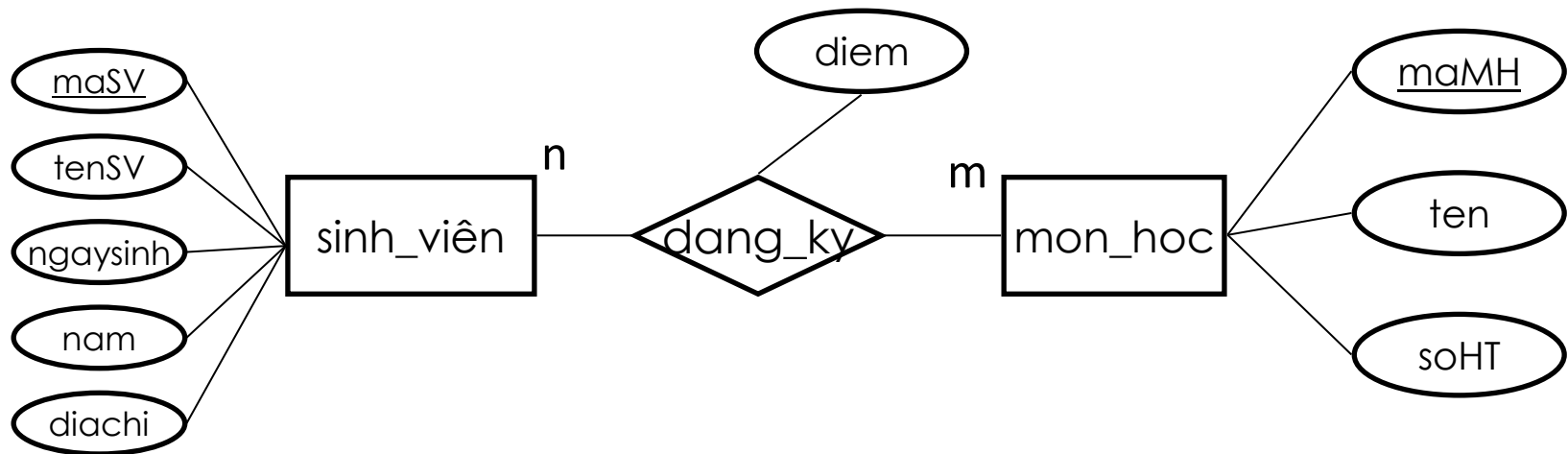
SINH\_VIEN(maSV, tenSV, ngaysinh, nam, diachi, *malop*)



- **Bước 5:** Liên kết n-n

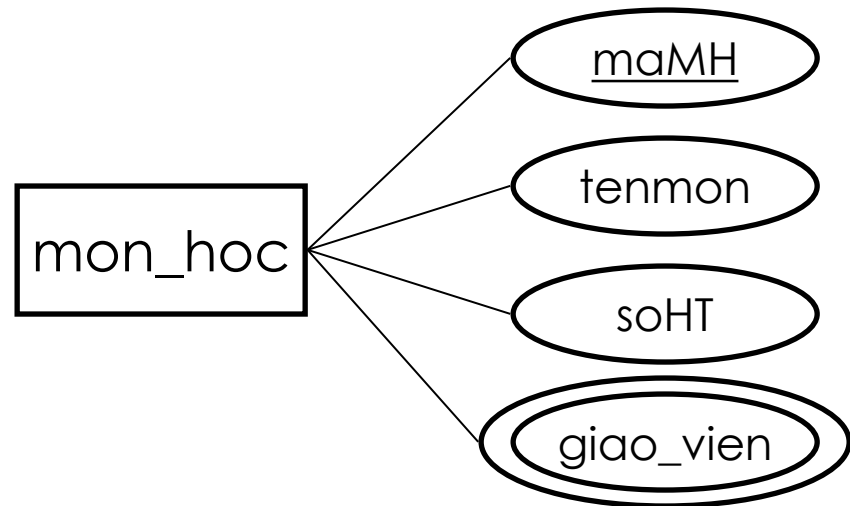
➤ Thêm 1 quan hệ mới xác định bởi các thuộc tính nằm trong khoá của các thực thể có liên quan và các thuộc tính của liên kết

DANG\_KY(maSV, maMH, diem)



- **Bước 6:** Với mỗi thuộc tính đa trị
  - Thêm 1 quan hệ mới xác định bởi thuộc tính đa trị và khoá của tập thực thể tương ứng

MH\_GV(maMH, giao\_vien)



# Mô hình dữ liệu hướng đối tượng (Object-oriented data model)

- Sự ra đời
  - Khoảng đầu những năm 90
- Biểu diễn: sơ đồ lớp
- Các khái niệm cơ bản
  - **Đối tượng**: một đối tượng trong thế giới thực, được xác định bởi một định danh duy nhất
  - **Thuộc tính**: biểu diễn một đặc tính của đối tượng,
  - **Phương thức** : thao tác được thực hiện trên đối tượng.
    - Tất cả các truy nhập vào thuộc tính của đối tượng đều phải được thực hiện thông qua các phương thức này.
  - **Lớp**: một cách thức để khai báo một tập các đối tượng có chung một tập thuộc tính và phương thức

# Ví dụ

```
class sinh_vien {  
    string maSV;  
    string tenSV;  
    date ngaysinh;  
    boolean nam;  
    string diachi;  
    string lop;  
  
    string ten();  
    string ngay_sinh();  
    string dia_chi();  
    string lop();  
    void gan_DC(string DC_moi);  
    void gan_lop(string lop);  
}
```

# Nhận xét

- Ưu điểm
  - Cho phép định nghĩa kiểu đối tượng phức tạp
  - Tính chất: bao đóng (*encapsulation*), kế thừa (*heritage*), đa hình (*polymorphism*)
- Nhược điểm
  - Cấu trúc lưu trữ phức tạp và có thể sử dụng nhiều con trỏ
  - Khả năng tối ưu hoá các xử lý bị hạn chế trong nhiều trường hợp

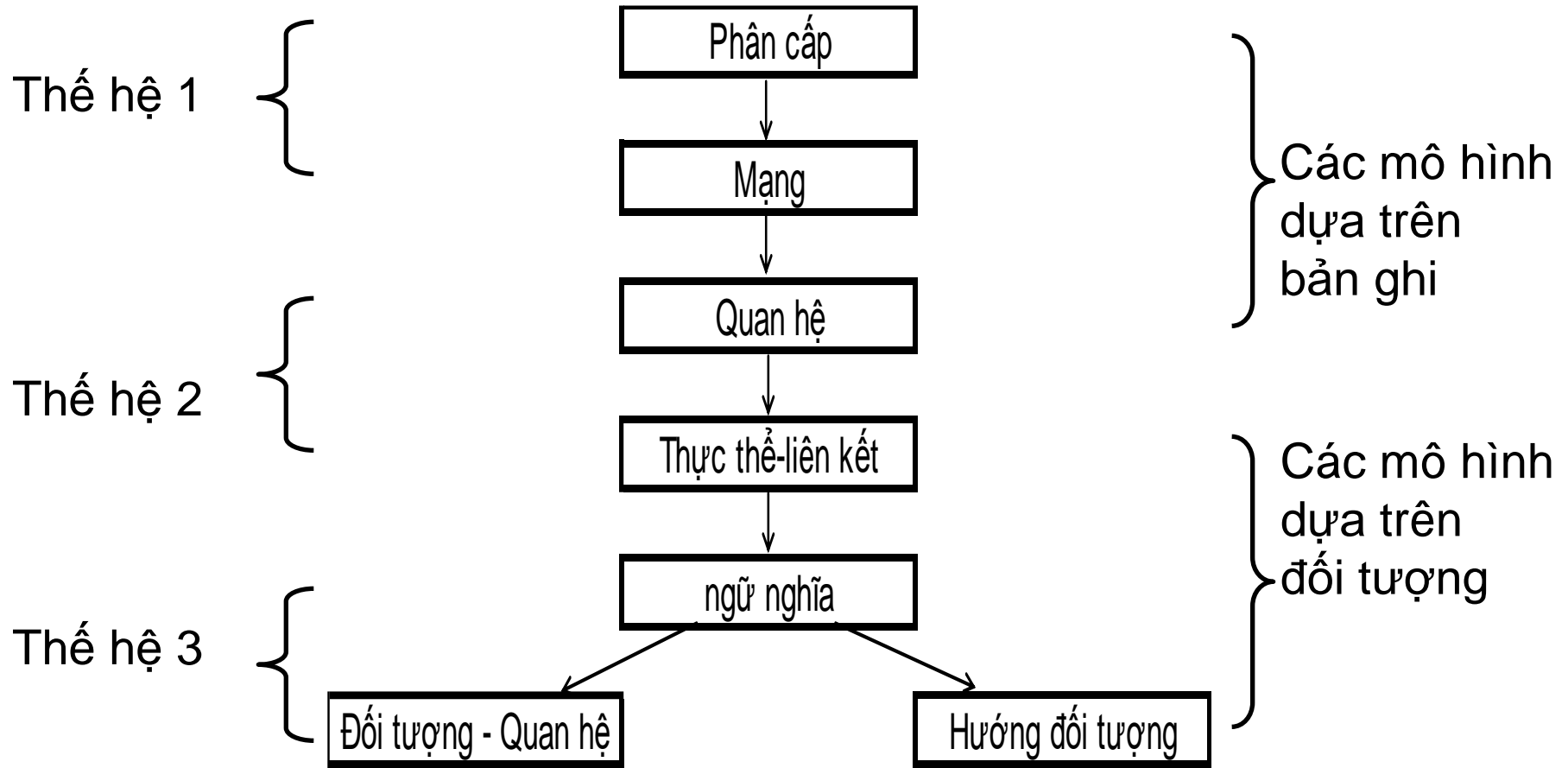
# So sánh và đánh giá

Nhắc lại: Mô hình dữ liệu là một tập hợp các khái niệm dùng để mô tả cấu trúc của một CSDL

	Mô hình mạng	Mô hình phân cấp	Mô hình quan hệ	Mô hình TT-LK	Mô hình HĐT
<b>biểu diễn ngữ nghĩa DL</b>	hạn chế	hạn chế	<i>tương đối đa dạng</i>	đa dạng	đa dạng
<b>lưu trữ DL</b>	s/d nhiều con trỏ	dữ liệu lặp lại	dễ dàng và hiệu quả	khó lưu trữ	<i>cấu trúc phức tạp</i>
<b>khả năng truy vấn</b>	đơn giản	đơn giản	đa dạng		đa dạng
<b>hiệu quả của truy vấn</b>	ít khả năng tối ưu	ít khả năng tối ưu	tối ưu hoá tốt	không được xem xét (không hiệu quả)	<i>không h/q khi s/d nhiều con trỏ</i>



# Phân loại các mô hình

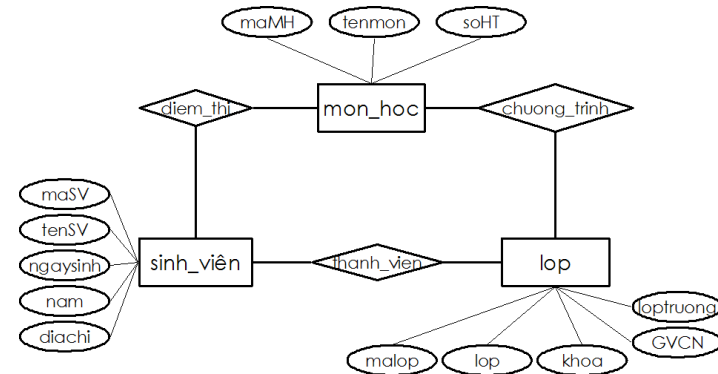


# Các bước xây dựng một hệ CSDL



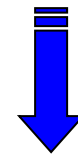
Mô tả ứng dụng

## 1: PHÂN TÍCH

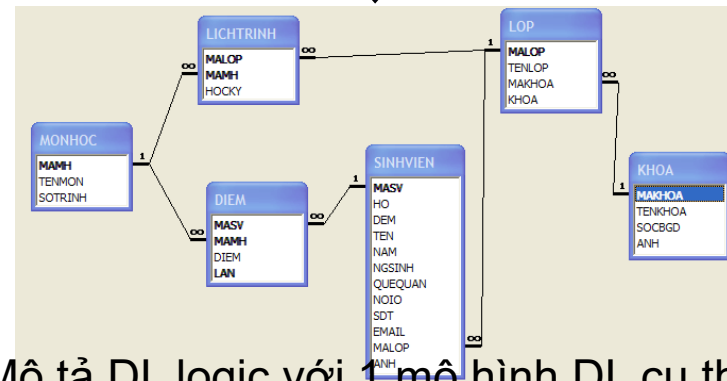


Mô hình hoá DL (vd: Sơ đồ thực thể-liên kết)

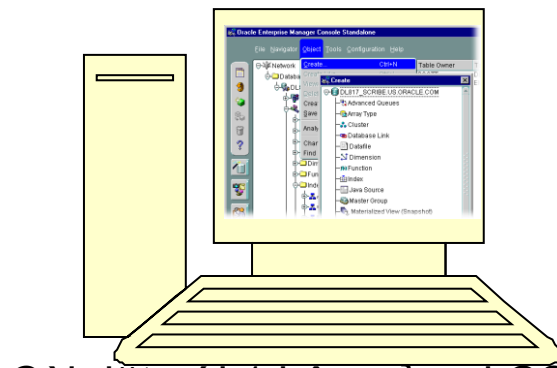
## 2: THIẾT KẾ



## 3: CÀI ĐẶT



Mô tả DL logic với 1 mô hình DL cụ thể (vd: Sơ đồ quan hệ)



Cài đặt với 1 hệ quản trị CSDL (vd: ORACLE)

- Cho sơ đồ thực thể liên kết bên dưới, hãy biến đổi sang mô hình quan hệ:

