



Chương 2

MÔ HÌNH THỰC THỂ KẾT HỢP (ENTITY-RELATIONSHIP)

Bộ môn Hệ thống thông tin
Khoa CNTT – Đại học Khoa học tự nhiên, TpHCM

Ref: TS. HỒ Bảo Quốc

Các slides này dựa trên các sách tham khảo

Database Systems by Connolly & Begg &

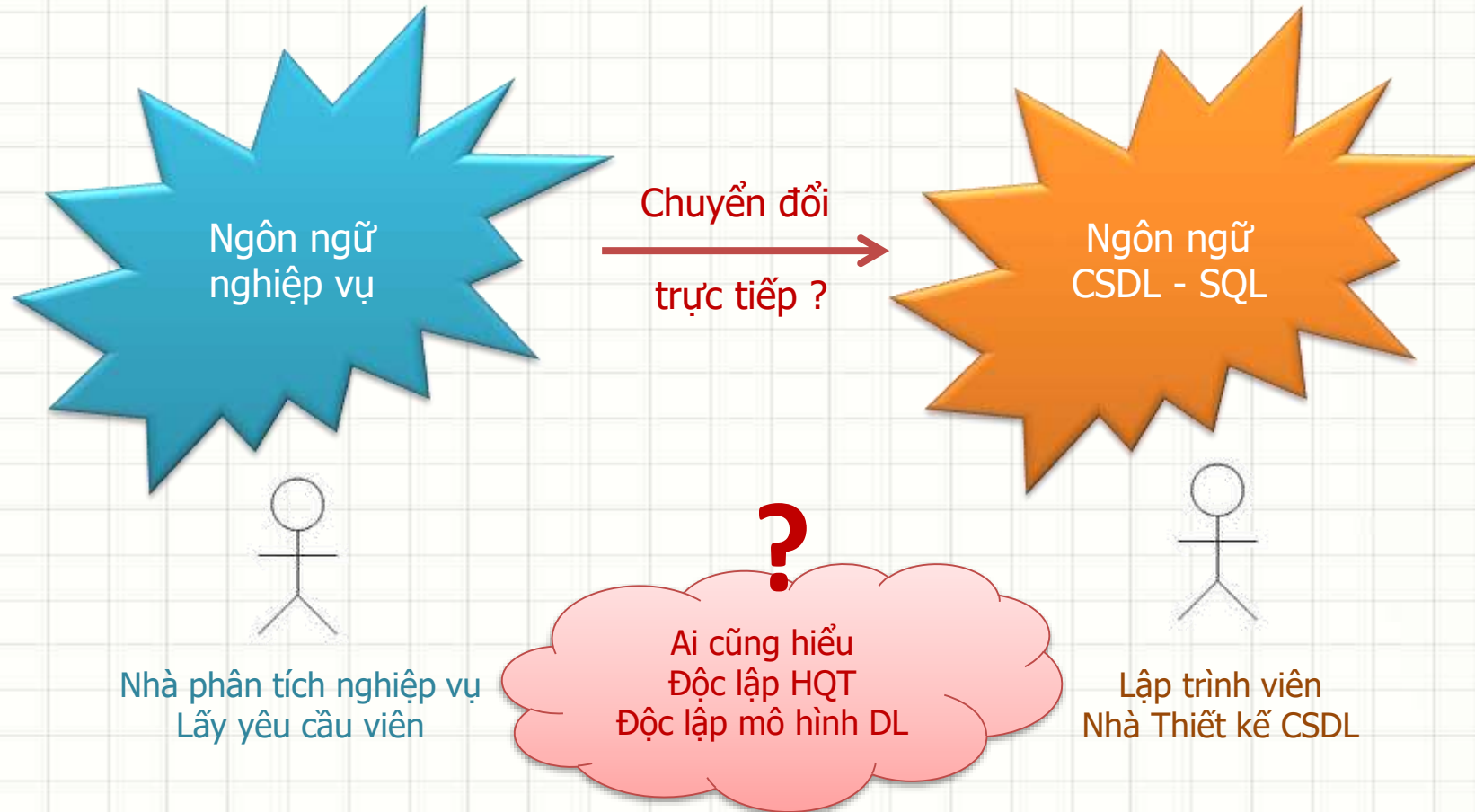
Fundamentals of Database Systems by Elmasri & Navathe



Mô hình hóa nhiều cấp

Thực tế					
Mô hình quan niệm	<ul style="list-style-type: none">♦ Độc lập với mô hình dữ liệu♦ Độc lập với DBMS				
Mô hình logic	<ul style="list-style-type: none">♦ Phụ thuộc mô hình dữ liệu♦ Phụ thuộc DBMS	Codasyl	Relational	Object	XML
Mô hình vật lý	<ul style="list-style-type: none">♦ Phụ thuộc mô hình dữ liệu♦ Phụ thuộc DBMS	<ul style="list-style-type: none">♦ Tổ chức vật lý của dữ liệu♦ Cấu trúc lưu trữ dữ liệu♦ Các cấu trúc chỉ mục (index)			

Tại sao phải có mô hình quan niệm ?



Nội dung

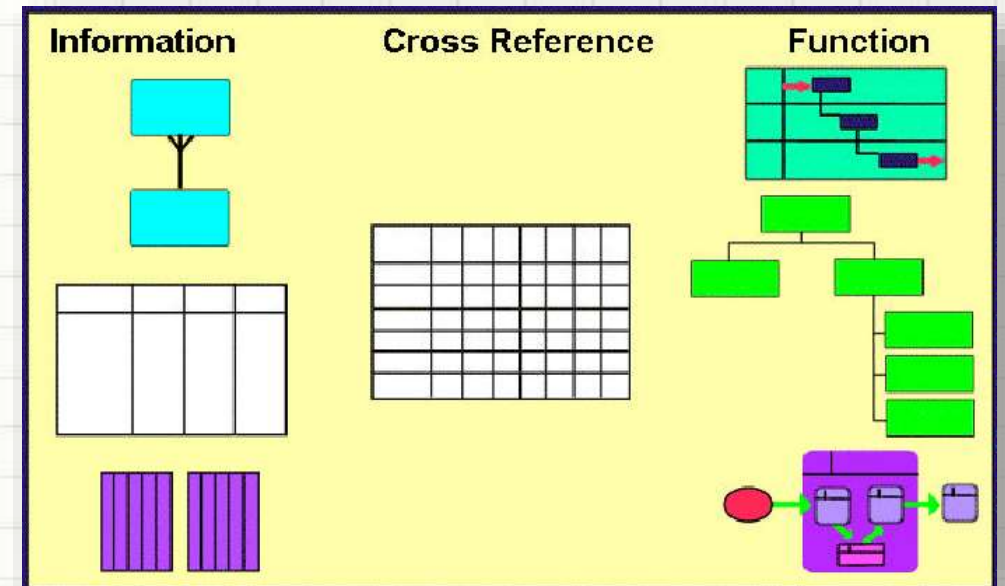
- Quá trình thiết kế CSDL
- Mô hình E/R
- Thiết kế
- Ví dụ



QUÁ TRÌNH THIẾT KẾ CSDL

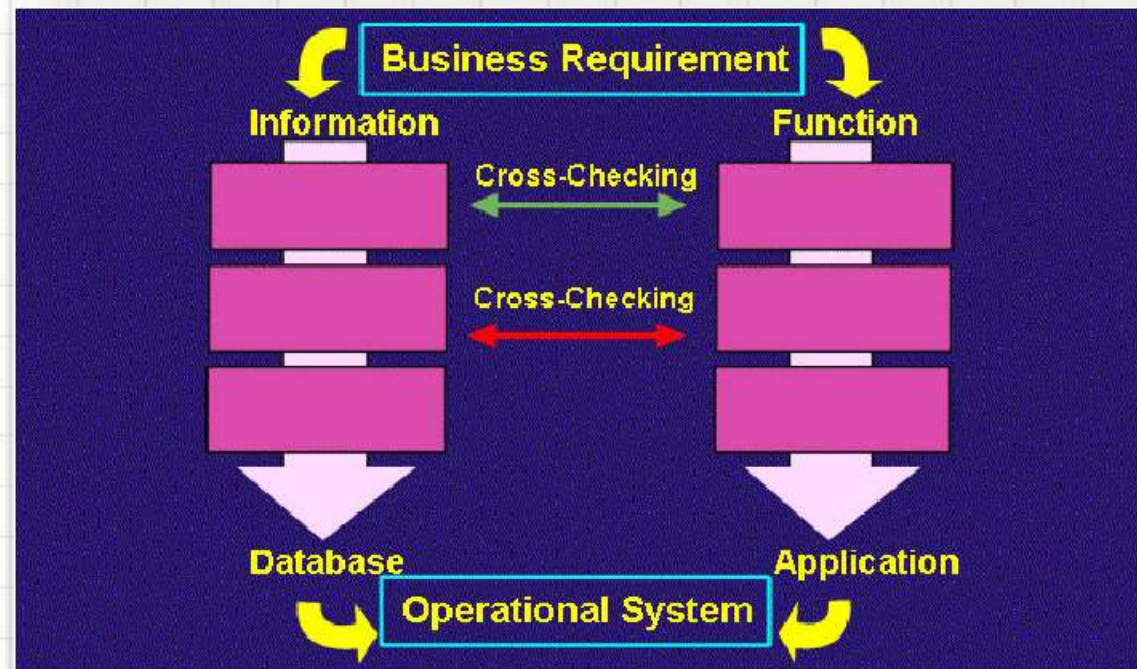
Các kỹ thuật phát triển hệ thống

- Có nhiều kỹ thuật để phát triển hệ thống
- Các kỹ thuật được chia thành ba loại:
 1. Mô hình hóa chức năng
 2. Mô hình thông tin
 3. Tham chiếu chéo (mô hình hóa các liên kết giữa các yếu tố thông tin)



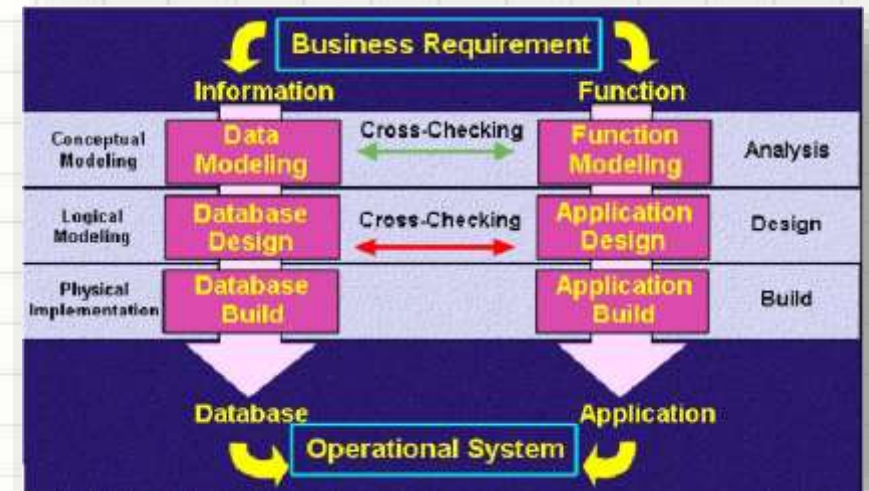
Mô hình Quan niệm, Logic, Vật lý

- Thông tin và xử lý được mô hình hóa tách biệt và sau đó sẽ kiểm tra chéo.
- Kết quả phát triển là CSDL lưu trữ dữ liệu và các ứng dụng truy xuất chúng.



Mô hình Quan niệm, Logic, Vật lý

- **Mô hình mức Quan niệm:**
 - Các mô hình hình thức, trừu tượng của thông tin và các yêu cầu xử lý
- **Mô hình mức Logic:**
 - Chuyển các mô hình quan niệm sang các đặc tả cho môi trường kỹ thuật
- **Xây dựng mức vật lý:**
 - Sử dụng câu lệnh của ngôn ngữ Structured Query Language (SQL) để tạo các đối tượng CSDL vật lý



Mô hình hóa dữ liệu (Data modeling)

- Cố gắng để tạo **một biểu diễn** của **thế giới thực**
 - Bỏ qua một ít sự phức tạp của thế giới thực
 - Sự đơn giản dựa vào một tập nhỏ các ký hiệu
- Cố gắng để rút gọn tổ chức dữ liệu thành **sự mô tả của các thực thể** và các **mối liên hệ** giữa chúng
 - Sự mô tả đơn giản của các yêu cầu thông tin mà máy tính có thể sử dụng
- Tiến trình mô hình hóa **độc lập** với nền tảng phát triển (hay phần mềm)
- Mô hình được sử dụng để **trao đổi** giữa người thiết kế CSDL và người dùng trong giai đoạn phân tích thiết kế

Quá trình thiết kế CSDL



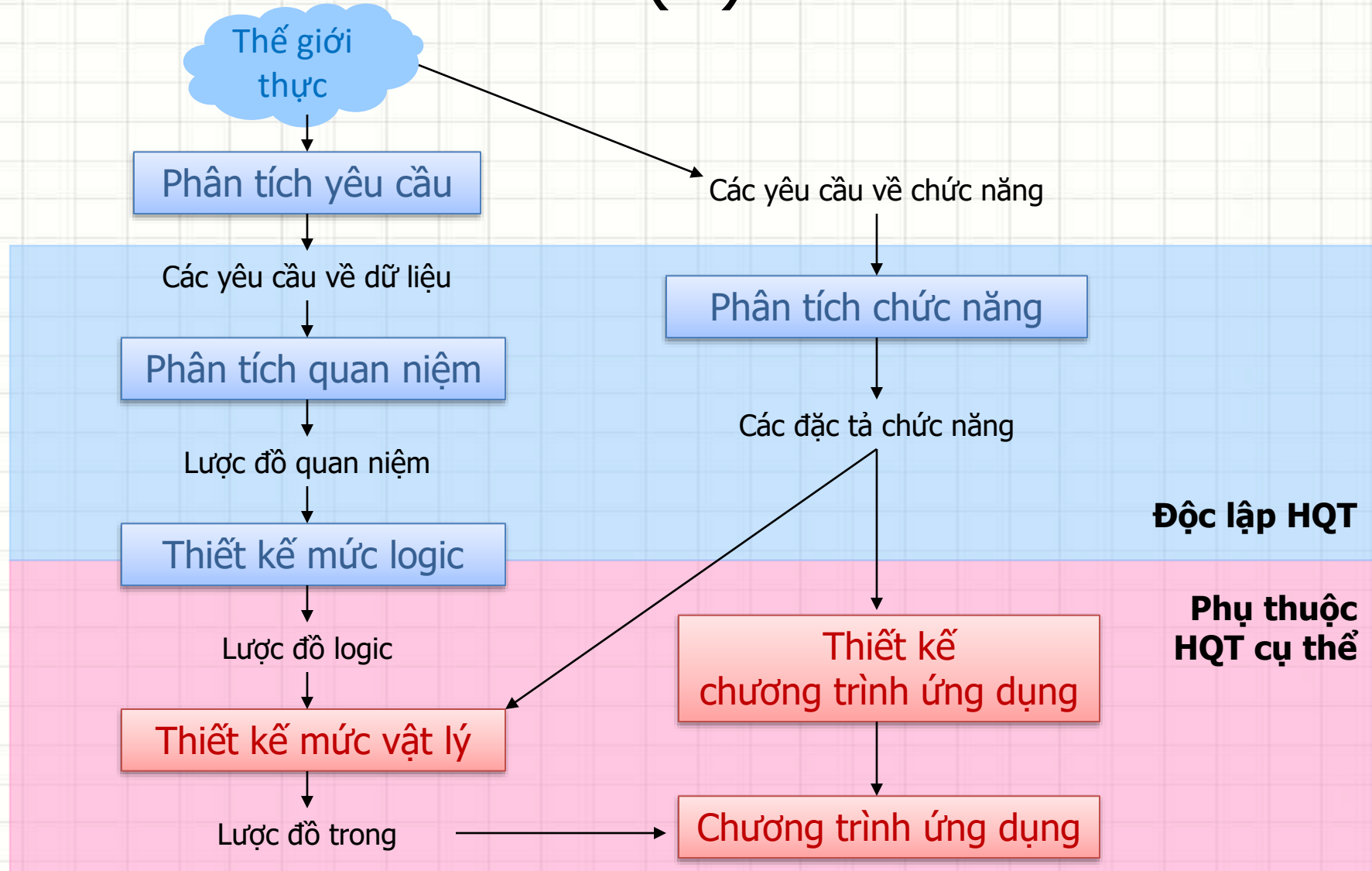
Các bước thiết kế CSDL

- Thu thập và phân tích nhu cầu
 - Các yêu cầu về CSDL
 - Các yêu cầu chức năng (các thao tác lên CSDL)
- Thiết kế quan niệm và phân tích chức năng
 - Tạo một sơ đồ quan niệm (*cấp cao*), ví dụ: ERD
 - Đặc tả giao tác cấp cao tương ứng với các thao tác lên CSDL
- Thiết kế Logic
 - Ánh xạ lược đồ quan niệm đến lược đồ Logic, ví dụ : mô hình quan hệ
- Thiết kế vật lý
 - Cấu trúc lưu trữ và tổ chức tập tin
- Thiết kế chương trình ứng dụng &
- Cài đặt giao tác



***Song song với
thiết kế logic***


Quá trình thiết kế CSDL (tt)



Các hướng tiếp cận CSDL

- **Tiếp cận từ trên xuống (3 bước) – Top Down**


- Xác định các thực thể dữ liệu
- Xác định các thuộc tính của các thực thể
- Xác định các mối liên hệ giữa các thực thể



Tính trừu tượng cao - Khó

- **Tiếp cận từ dưới lên (3 bước) – Bottom Up**

- Tập hợp các yếu tố dữ liệu từ thực tế
- Gom nhóm thành các thực thể
- Xác định các mối quan hệ



Xuất phát từ thực tế – Dễ



MÔ HÌNH THỰC THỂ KẾT HỢP

Mô hình thực thể kết hợp

- Được dùng để thiết kế CSDL ở mức quan niệm
- Biểu diễn trừu tượng cấu trúc của CSDL
- Lược đồ thực thể - kết hợp bao gồm 3 thành phần chính:
 - Tập thực thể (Entity Sets)
 - Thuộc tính (Attributes)
 - Mỗi kết hợp (Relationship)

Tên tập thực thể

Tên thuộc tính

Tên mỗi kết hợp

Thực thể - Tập thực thể

- Một thực thể là một đối tượng của thế giới thực, có thể cụ thể hoặc trừu tượng và tồn tại độc lập
- Tập hợp các thực thể giống nhau tạo thành 1 tập thực thể

- **Chú ý**

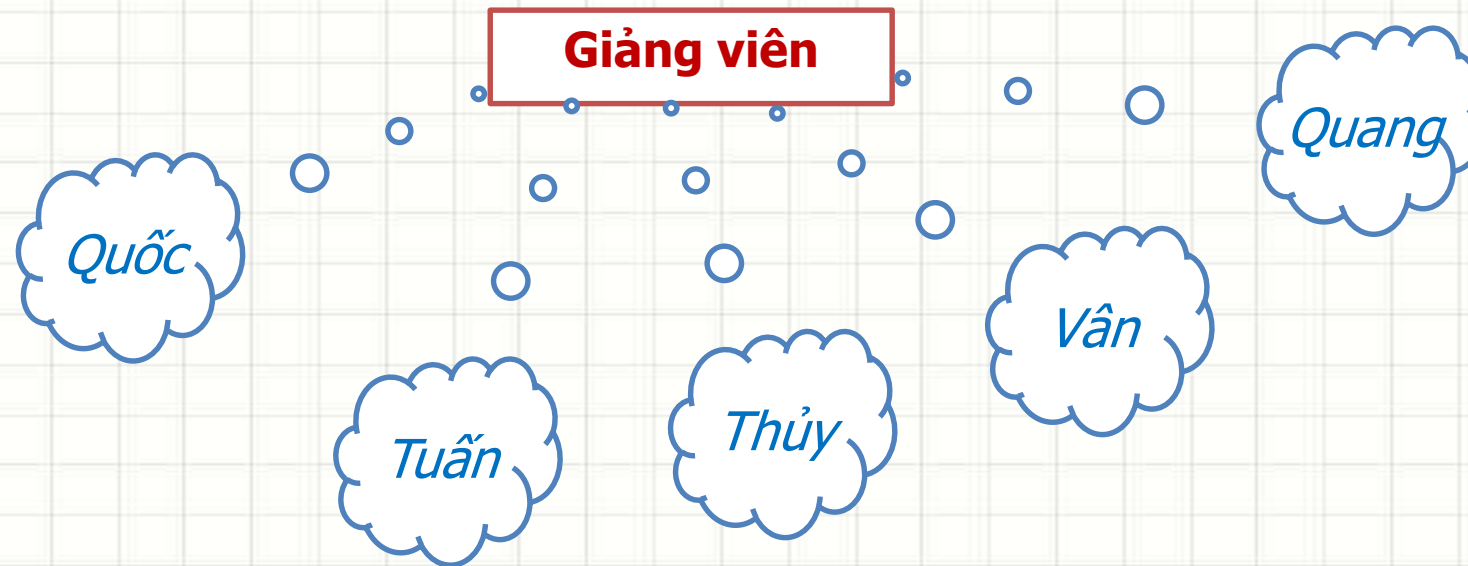
Mô hình ER	Mô hình OOP
Thực thể (Entity)	Đối tượng (Object)
Tập thực thể (Entity Set)	Lớp đối tượng (Class of Object)
Thành phần: <ul style="list-style-type: none">- Cấu trúc dữ liệu (Attribute)— Thao tác trên dữ liệu (Method)	Thành phần: <ul style="list-style-type: none">- Cấu trúc dữ liệu (Attribute)- Thao tác trên dữ liệu (Method)

Cấu trúc của dữ liệu

~~Thao tác trên dữ liệu~~

Thực thể - Tập thực thể (Ví dụ)

- Ví dụ "Quản lý giáo viên"



Thuộc tính của tập thực thể

- Là những **đặc tính riêng biệt** của tập thực thể
- Tính chất của thực thể **cần được quản lý**
- Chỉ quan tâm đến các tính chất **có liên quan** đến ứng dụng
 - **Ví dụ:** tập thực thể GIANGVIEN có các thuộc tính
 - Họ tên
 - Ngày sinh
 - Địa chỉ
 - Học hàm, học vị
 - ...
- Nên có một mô tả ngắn gọn của một thuộc tính
 - Bản số xe: "Là một số duy nhất được gán cho một giấy đăng ký xe"

Thuộc tính của tập thực thể - Phân loại

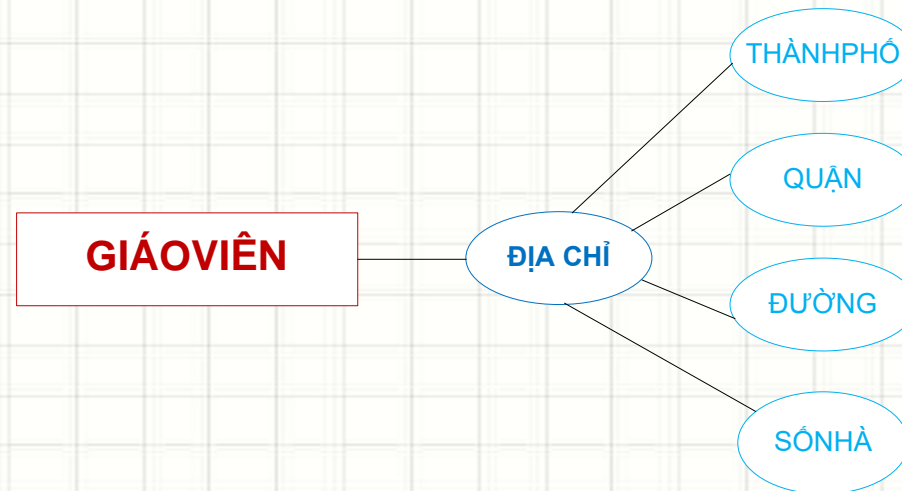
- Là những giá trị nguyên tố (không thể chia nhỏ hơn nữa về mặt ngữ nghĩa)
 - Kiểu chuỗi
 - Kiểu số nguyên
 - Kiểu số thực

Gồm có :

- Thuộc tính đơn <> Thuộc tính kết hợp
- Thuộc tính đơn trị <> Thuộc tính đa trị
- Thuộc tính suy diễn

Thuộc tính kết hợp

- **Thuộc tính kết hợp** là thuộc tính được kết hợp từ nhiều thuộc tính khác
- Ví dụ: thuộc tính **ĐỊA CHỈ** của tập thực thể **GIÁOVIÊN** có thể chia nhỏ thành các thành phần: **SỐNHÀ, ĐƯỜNG, QUẬN, THÀNH PHỐ**



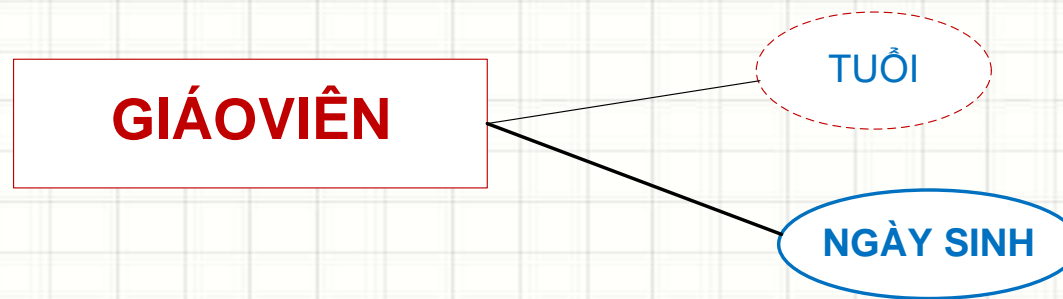
Thuộc tính đa trị

- Thuộc tính đa trị là thuộc tính nhận nhiều giá trị đối với một thực thể cụ thể.
- Ví dụ: thuộc tính **ĐIỆN THOẠI** của tập thực thể **GIÁO VIÊN** là thuộc tính đa trị. Mỗi giáo viên có thể có nhiều số điện thoại.



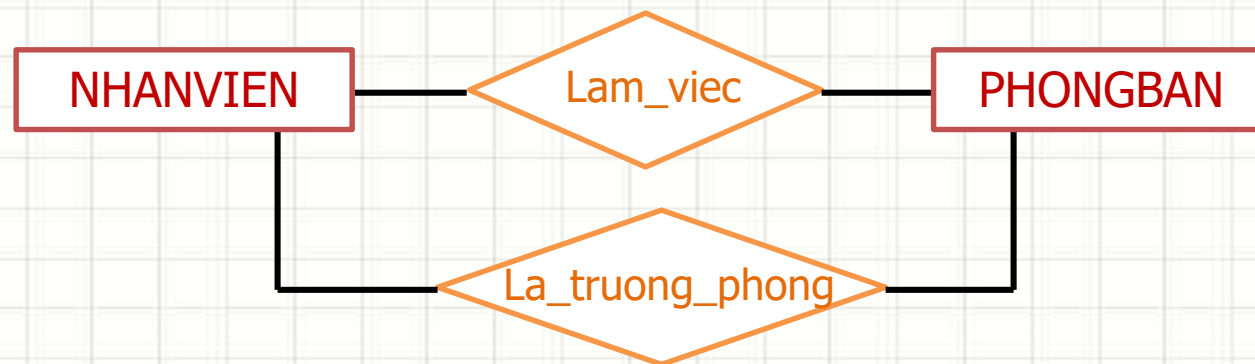
Thuộc tính suy diễn

- Thuộc tính suy diễn là thuộc tính mà giá trị của nó được tính toán từ giá trị của các thuộc tính khác.
- Ví dụ: thuộc tính **TUỔI** của GIÁOVIÊN có thể tính toán từ thuộc tính **NGÀY SINH** của GIÁOVIÊN



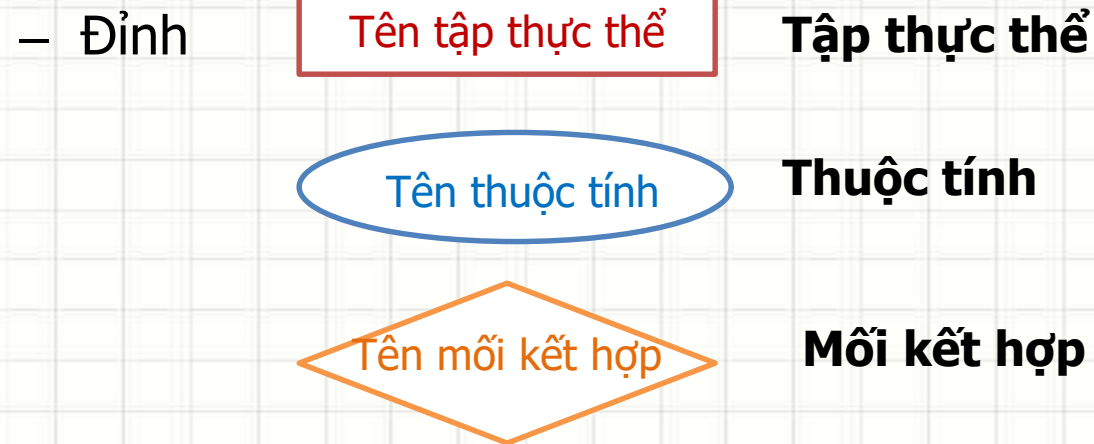
Mối kết hợp - quan hệ

- Mỗi quan hệ = **Mỗi kết hợp**
- Là sự liên kết giữa 2 hay nhiều tập thực thể **hoặc** trong nội bộ của một tập thực thể
- **Ví dụ:** giữa tập thực thể NHANVIEN và PHONGBAN có các liên kết
 - Một nhân viên thuộc một phòng ban nào đó
 - Một phòng ban có một nhân viên làm trưởng phòng



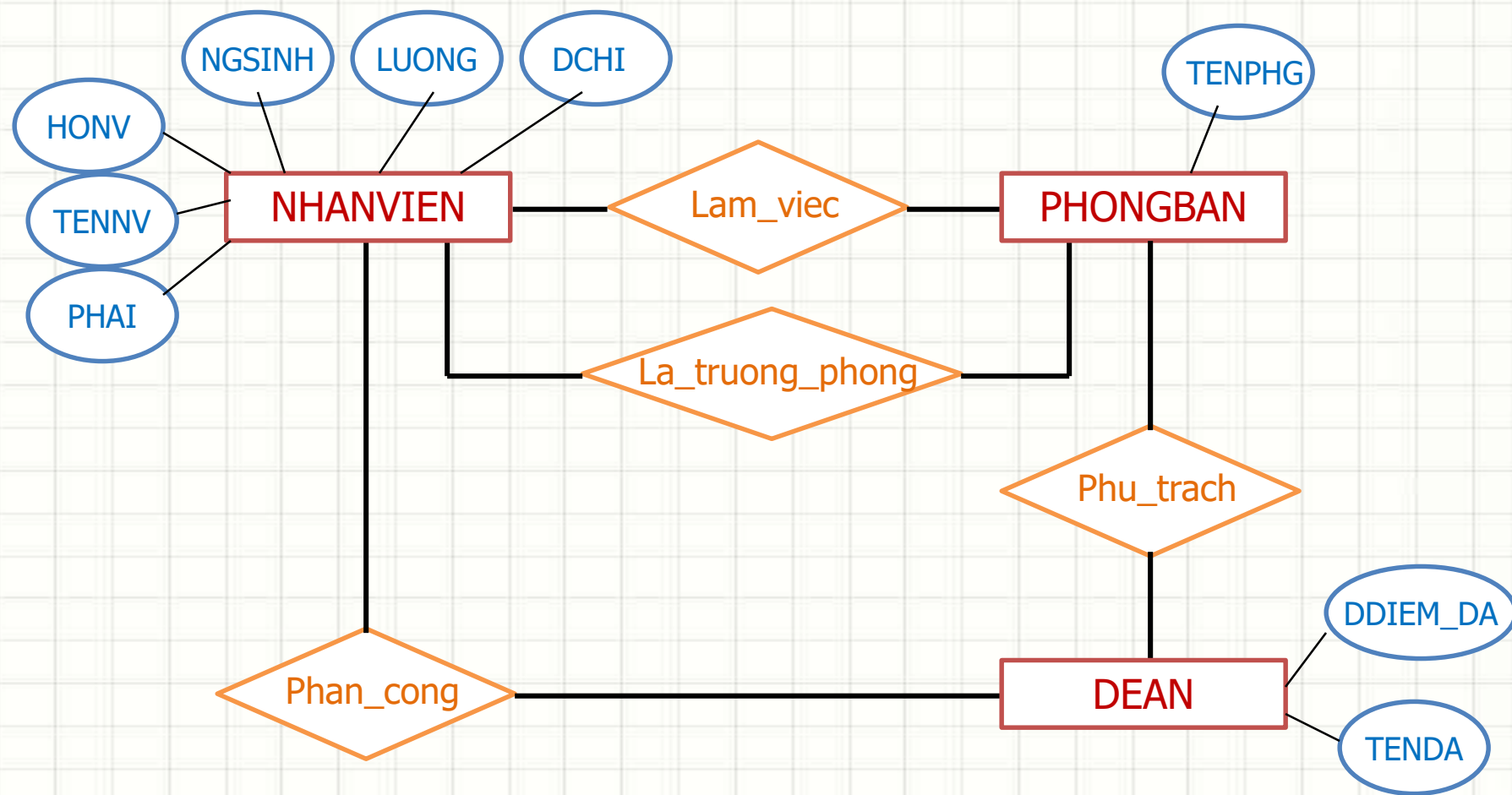
Lược đồ E/R

- Là đồ thị biểu diễn các tập thực thể, thuộc tính và mối quan hệ



- Cạnh là đường nối giữa
- Tập thực thể và thuộc tính
 - Mối quan hệ và tập thực thể
 - Mối kết hợp và thuộc tính

Ví dụ lược đồ E/R

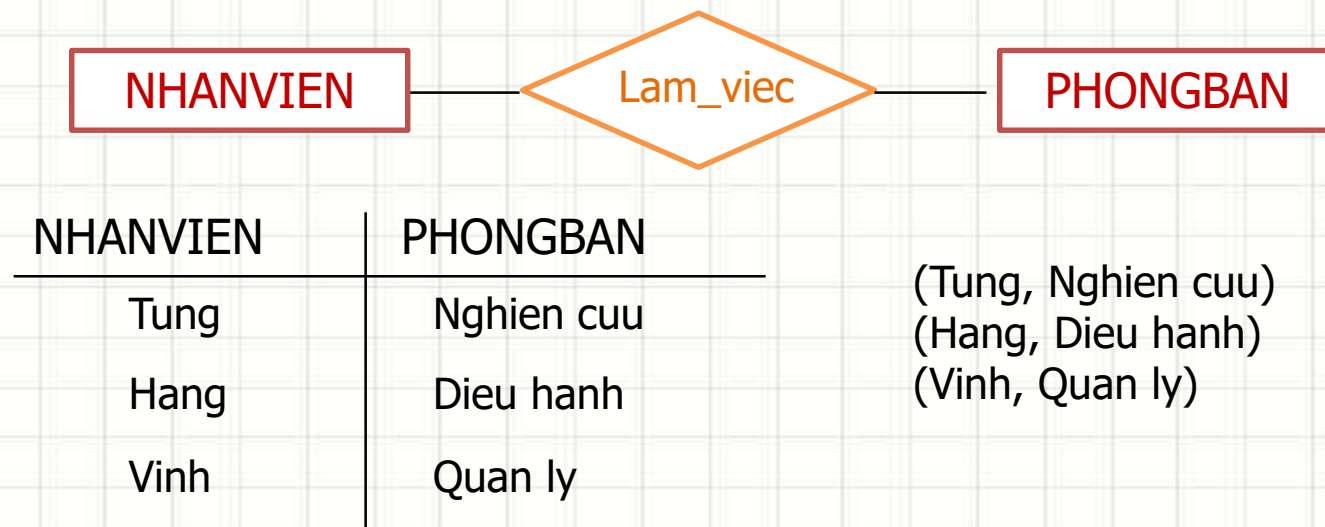


Thể hiện của lược đồ E/R

- Một CSDL được mô tả bởi lược đồ E/R sẽ chứa đựng những dữ liệu cụ thể gọi là thể hiện CSDL
 - Mỗi tập thực thể sẽ có tập hợp hữu hạn các thực thể
 - Giả sử tập thực thể NHANVIEN có các thực thể như NV_1, NV_2, \dots, NV_n
 - Mỗi thực thể sẽ có 1 giá trị cụ thể tại mỗi thuộc tính
 - NV_1 có TENNV="Tung", NGSINH="08/12/1955", PHAI="Nam"
 - NV_2 có TENNV="Hang", NGSINH="07/19/1966", PHAI="Nu"
- **Chú ý**
 - Không lưu trữ lược đồ E/R trong CSDL
 - Khái niệm trừu tượng
 - Lược đồ E/R chỉ giúp ta thiết kế CSDL trước khi chuyển các quan hệ và dữ liệu xuống mức vật lý

Mối kết hợp - Thể hiện

- Thể hiện CSDL còn chứa các mối kết hợp cụ thể
 - Cho mỗi kết hợp R kết nối n tập thực thể E_1, E_2, \dots, E_n
 - Thể hiện của R là tập hữu hạn các danh sách (e_1, e_2, \dots, e_n) , trong đó e_i là các giá trị được chọn từ các tập thực thể E_i
- Xét mối kết hợp “Lam_viec” sau:



Mối kết hợp - Multiplicity

- Xét mối kết hợp nhị phân R (binary relationship) giữa 2 tập thực thể E và F, tính multiplicity bao gồm

- Một-Nhiều

- Một E có thể kết hợp với nhiều F
- Một F có thể kết hợp với một E



- Một-Một

- Một E có thể kết hợp với một F
- Một F có thể kết hợp với một E



- Nhiều-Nhiều

- Một E có thể kết hợp với nhiều F
- Một F có thể kết hợp với nhiều E



Mối Kết hợp - Multiplicity (tt)

- (min, max) chỉ định mỗi thực thể $e \in E$ tham gia **ít nhất** và **nhiều nhất** vào thể hiện của R



- $(0,1)$ – không hoặc 1
- $(1,1)$ – duy nhất 1
- $(0,n)$ – không hoặc nhiều
- $(1,n)$ – một hoặc nhiều

Mối Kết hợp - Multiplicity (tt)

- Ví dụ

- Một phòng ban có nhiều nhân viên



- Một nhân viên chỉ thuộc 1 phòng ban



- Một nhân viên có thể được phân công vào nhiều đề án hoặc không được phân công vào đề án nào

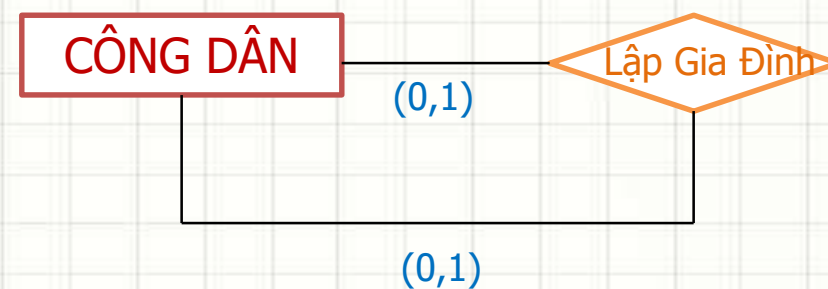
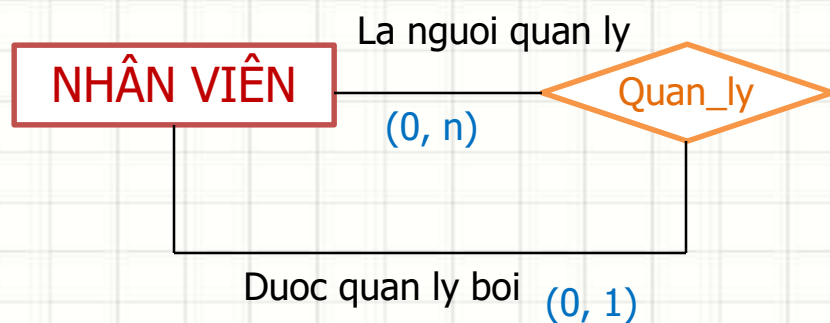


- Một nhân viên có thể là trưởng phòng của 1 phòng ban nào đó



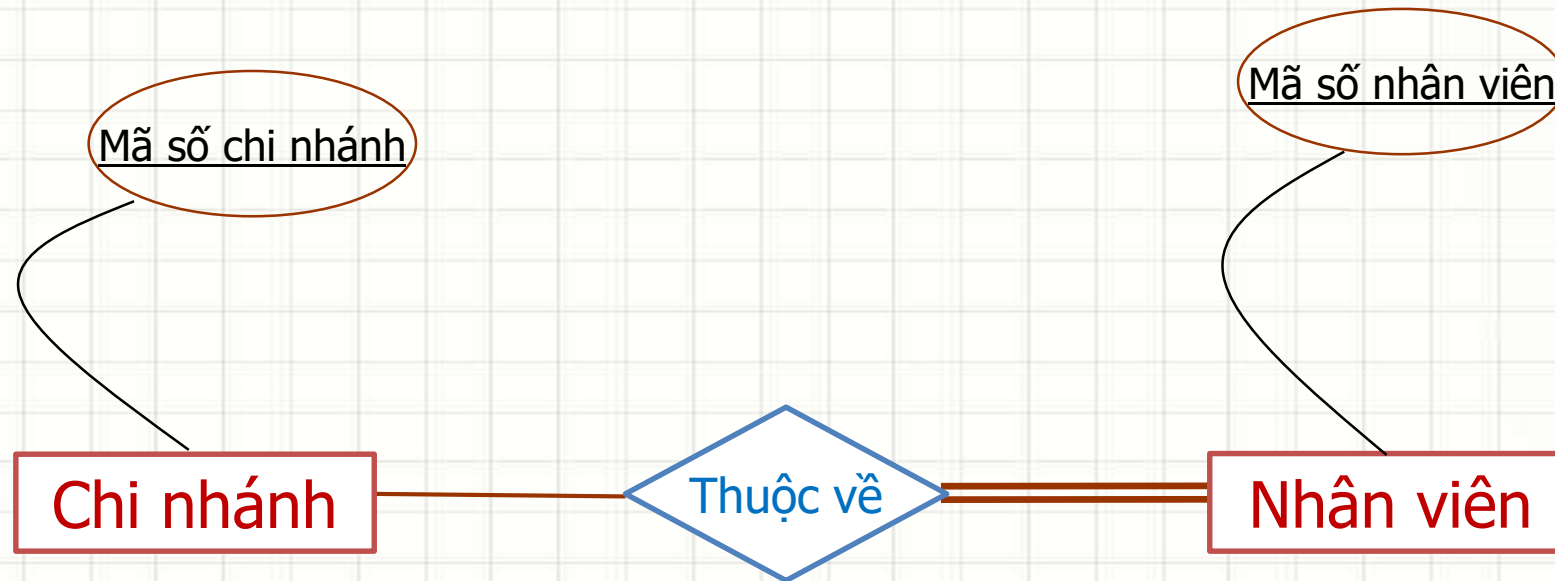
Mối Kết hợp - Vai trò

- Còn gọi là Mối kết hợp phản xạ
- Một loại thực thể có thể tham gia nhiều lần vào một kết hợp **với nhiều vai trò khác nhau**



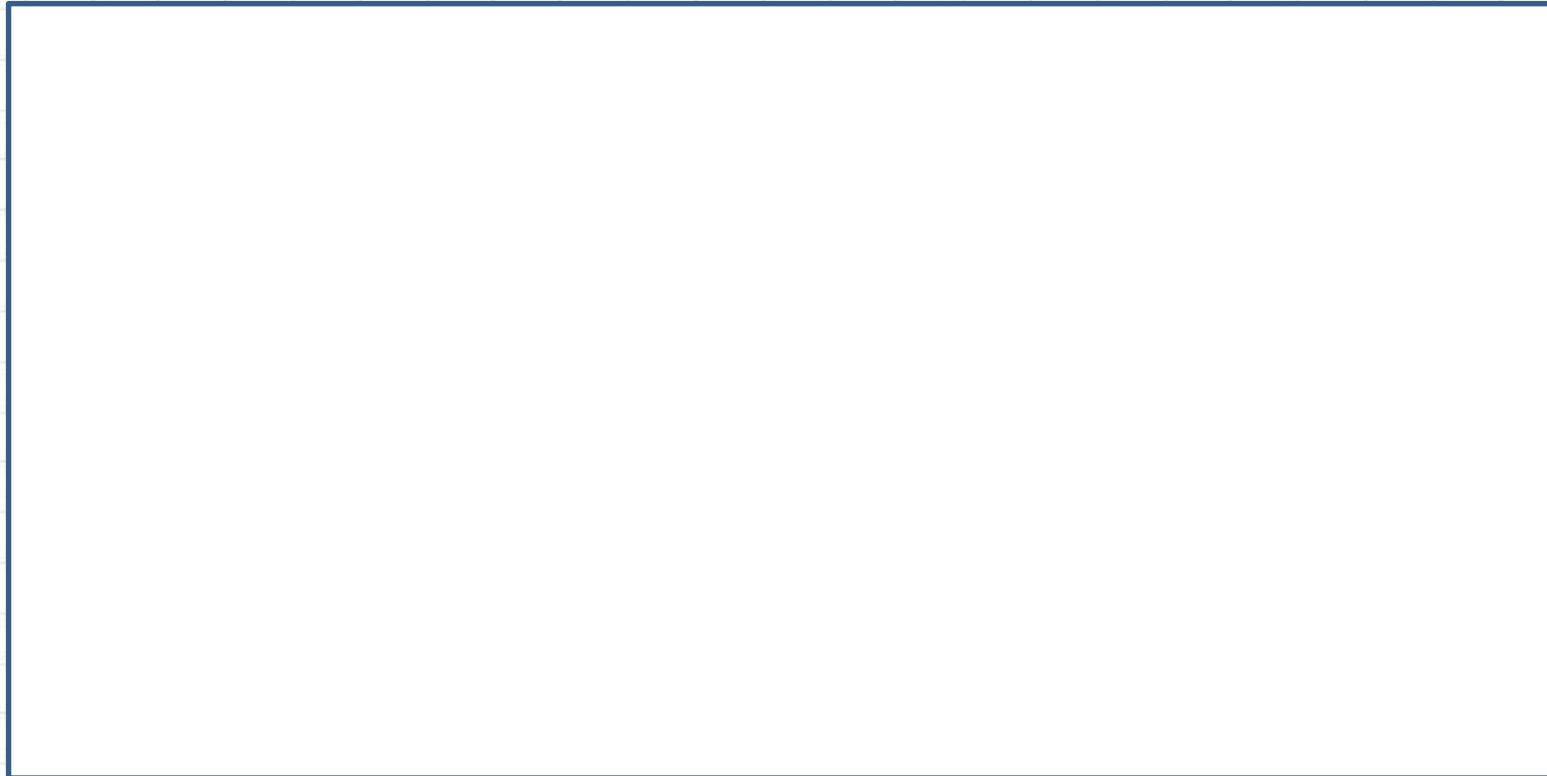
Mối Kết hợp - ràng buộc tham gia

- Một nhân viên bắt buộc phải thuộc về một chi nhánh nào đó



Mối Kết hợp

- Xác định mối quan hệ và bản số cho các tập thực thể sau cho trường hợp CLC

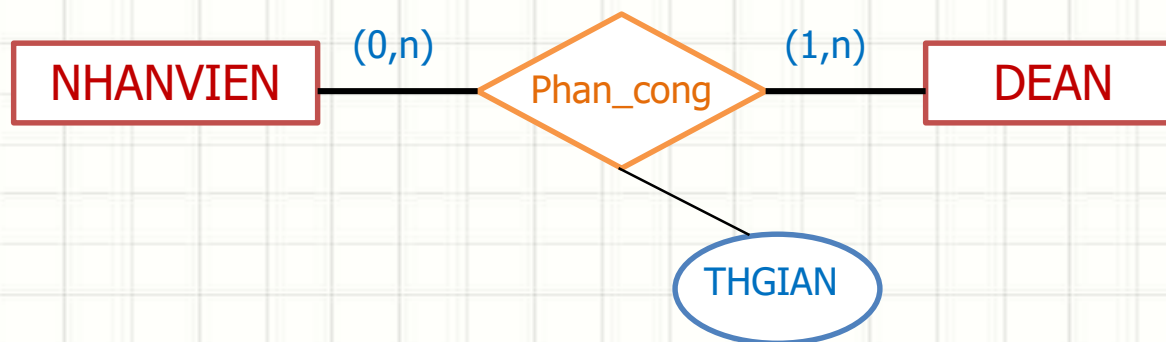


Chấm

1. Thông tin nhóm đúng, đủ, ko dư: 0.5đ – 1đ
2. Trình bày rõ ràng, đẹp: 0.5đ – 1đ
3. Không có sử dụng ký hiệu sai: 0.5đ-2đ
4. Các bản số hợp lý: 0.5đ-3đ
5. Số lượng mỗi kết hợp đủ: 1đ-3đ (cho tối thiểu 8 mỗi kết hợp)

Thuộc tính trên mỗi Kết hợp

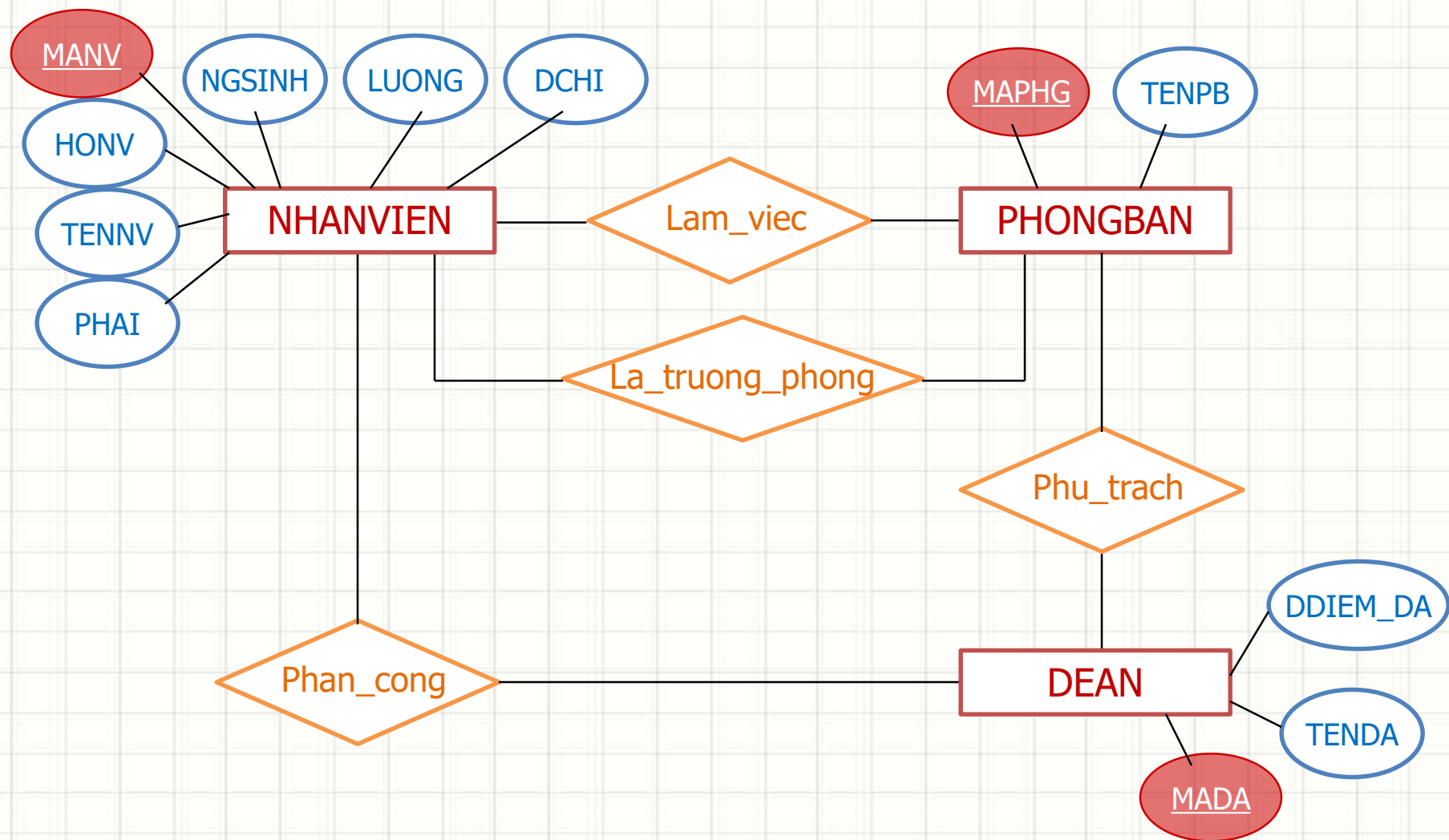
- Thuộc tính trên mỗi kết hợp mô tả tính chất cho mỗi quan hệ đó
- Thuộc tính này không thể gắn liền với những thực thể tham gia vào mỗi kết hợp



Thuộc tính khóa

- Còn được gọi là thuộc tính định danh của tập thực thể (identifier)
- Dùng để **phân biệt** giữa **các thực thể** trong tập thực thể
- **Khóa K** của tập thực thể E là **một hay nhiều thuộc tính** sao cho
 - Lấy ra 2 thực thể bất kỳ e_1 , và e_2 trong E
 - Thì e_1 và e_2 **không** thể có các giá trị giống nhau tại **bộ thuộc tính** trong K
- **Chú ý**
 - Mỗi tập thực thể phải có 1 khóa
 - Một khóa có thể **có 1** hay **nhiều** thuộc tính (composite identifier)
 - Có thể có nhiều khóa trong 1 tập thực thể, ta sẽ chọn ra 1 khóa làm khóa chính cho tập thực thể đó
 - Thuộc tính khóa có thể **đã có thực** ngoài thực tế HOẶC **chưa tồn tại**

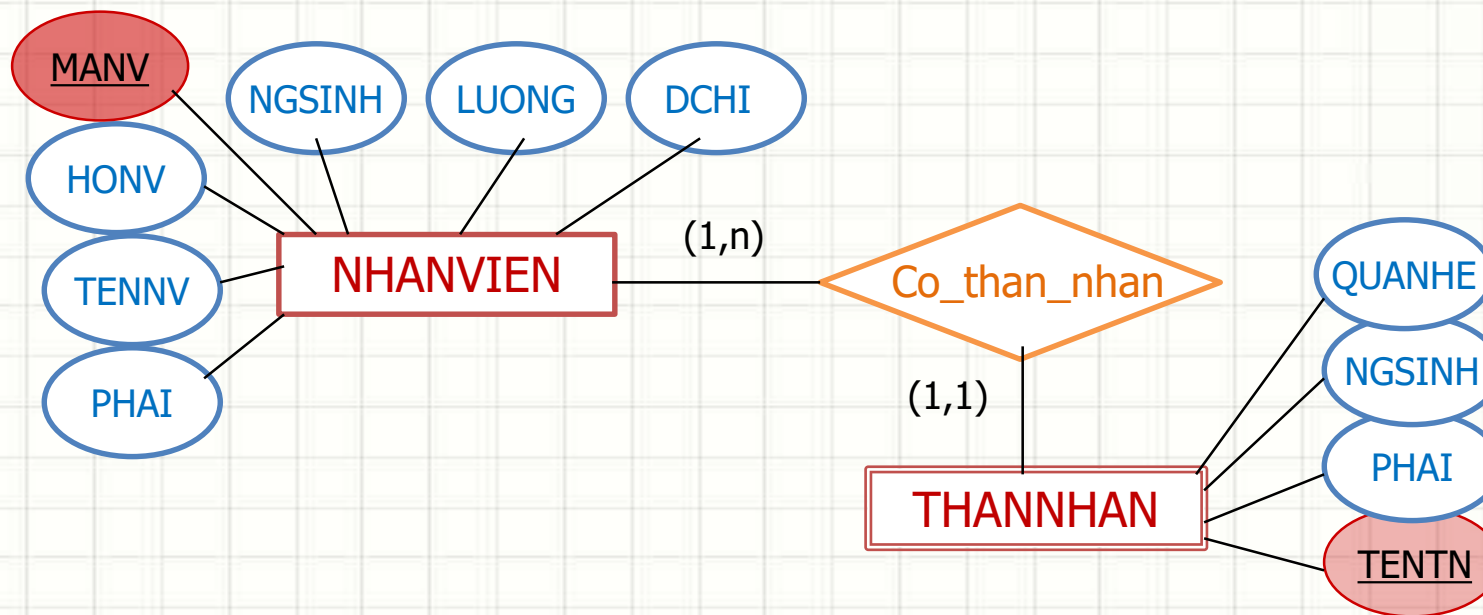
Ví dụ thuộc tính khóa



Tập thực thể yếu

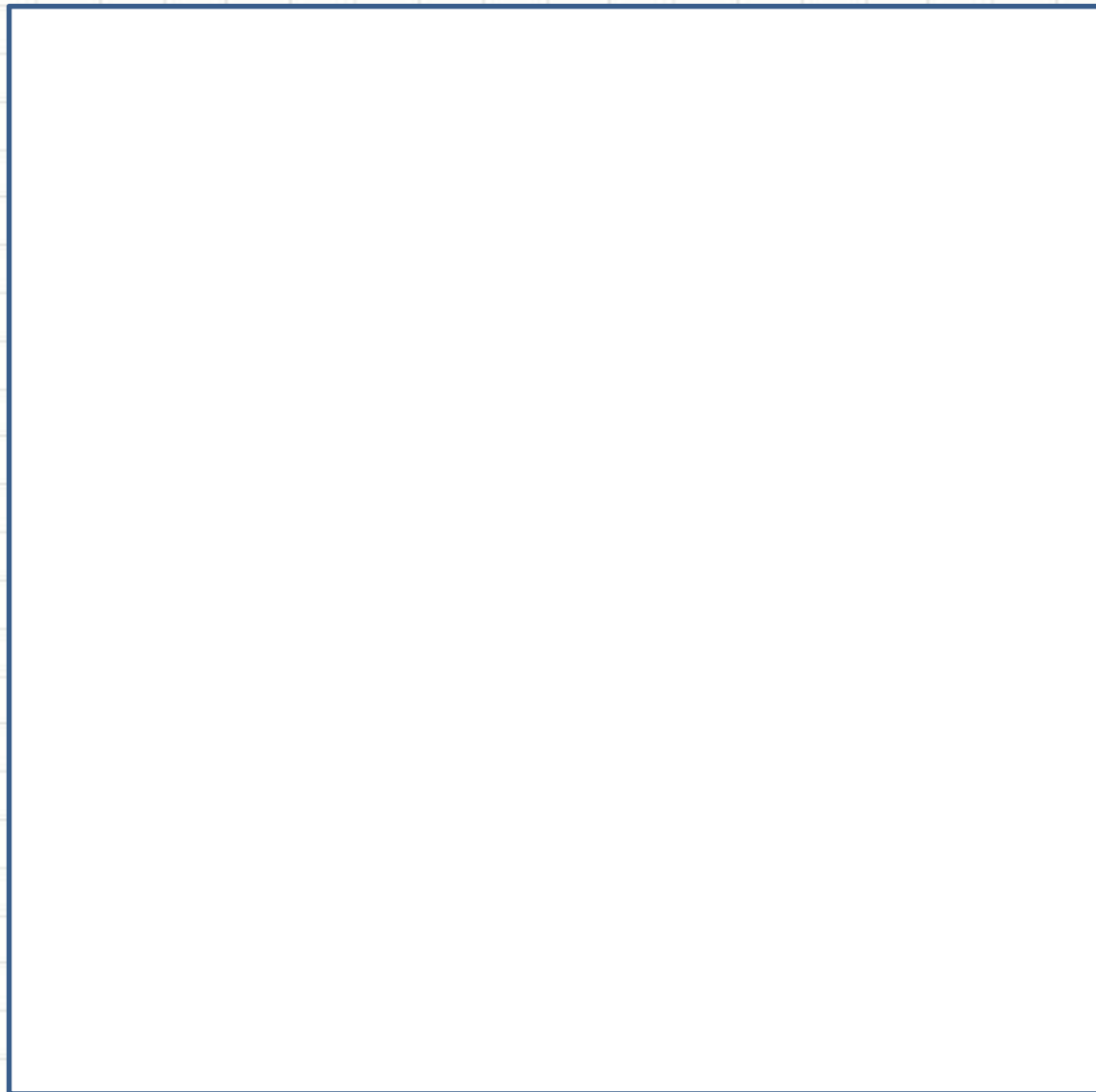
- Là tập thực thể **có khóa** được tổ hợp từ những thuộc tính của tập thực thể khác
- Tập thực thể yếu (**weak entity set**) phải tham gia vào mỗi kết hợp mà trong đó có một tập thực thể chính

- Ví dụ 1



Tập thực thể yếu (tt)

- Ví dụ 2



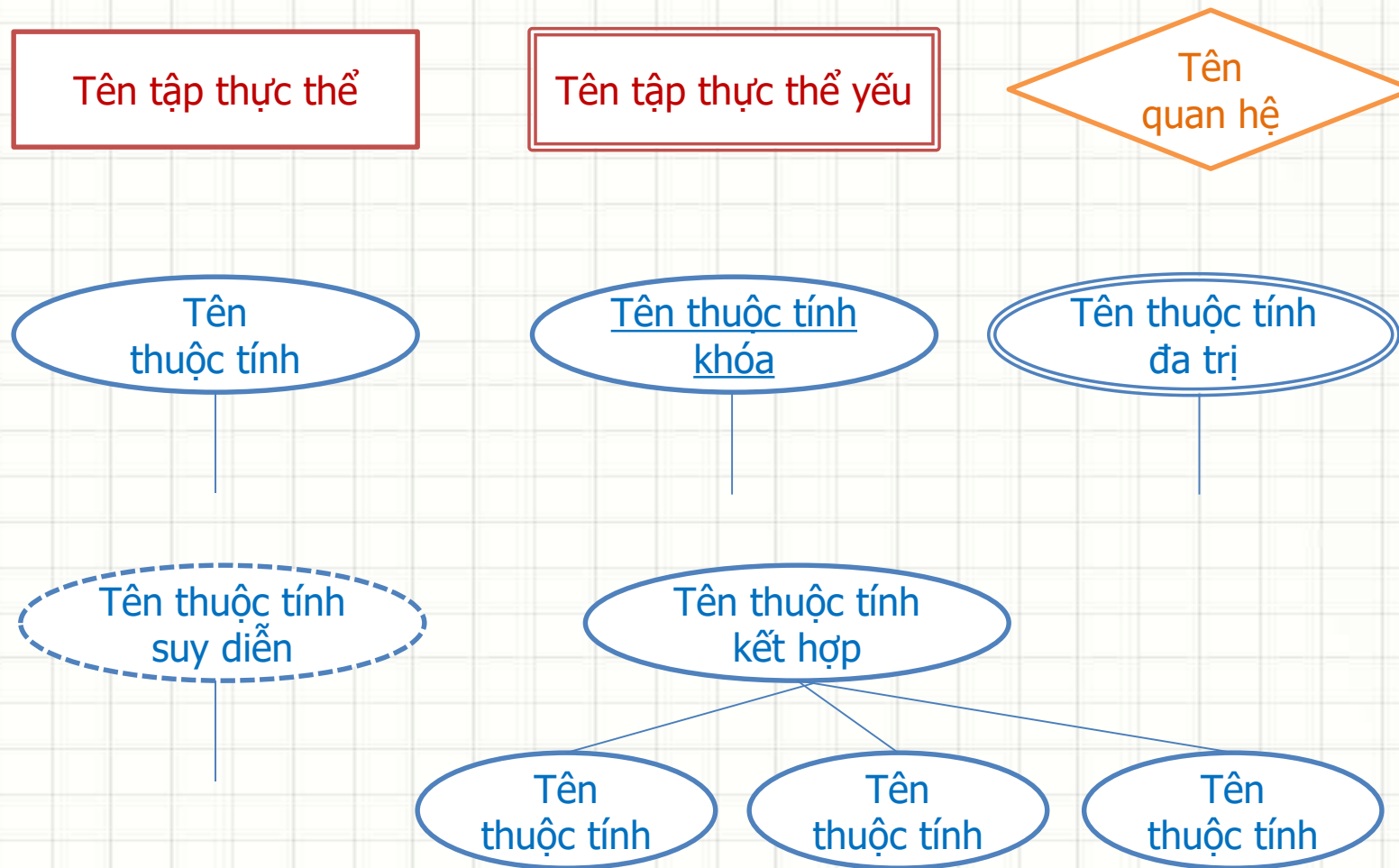
Các bước thiết kế

- (1) Xác định tập thực thể
- (2) Xác định mỗi kết hợp
- (3) Quyết định (min, max) cho mỗi kết hợp
- (4) Xác định thuộc tính và gán thuộc tính cho tập thực thể và mỗi kết hợp
- (5) Quyết định miền giá trị cho thuộc tính
- (6) Quyết định thuộc tính khóa

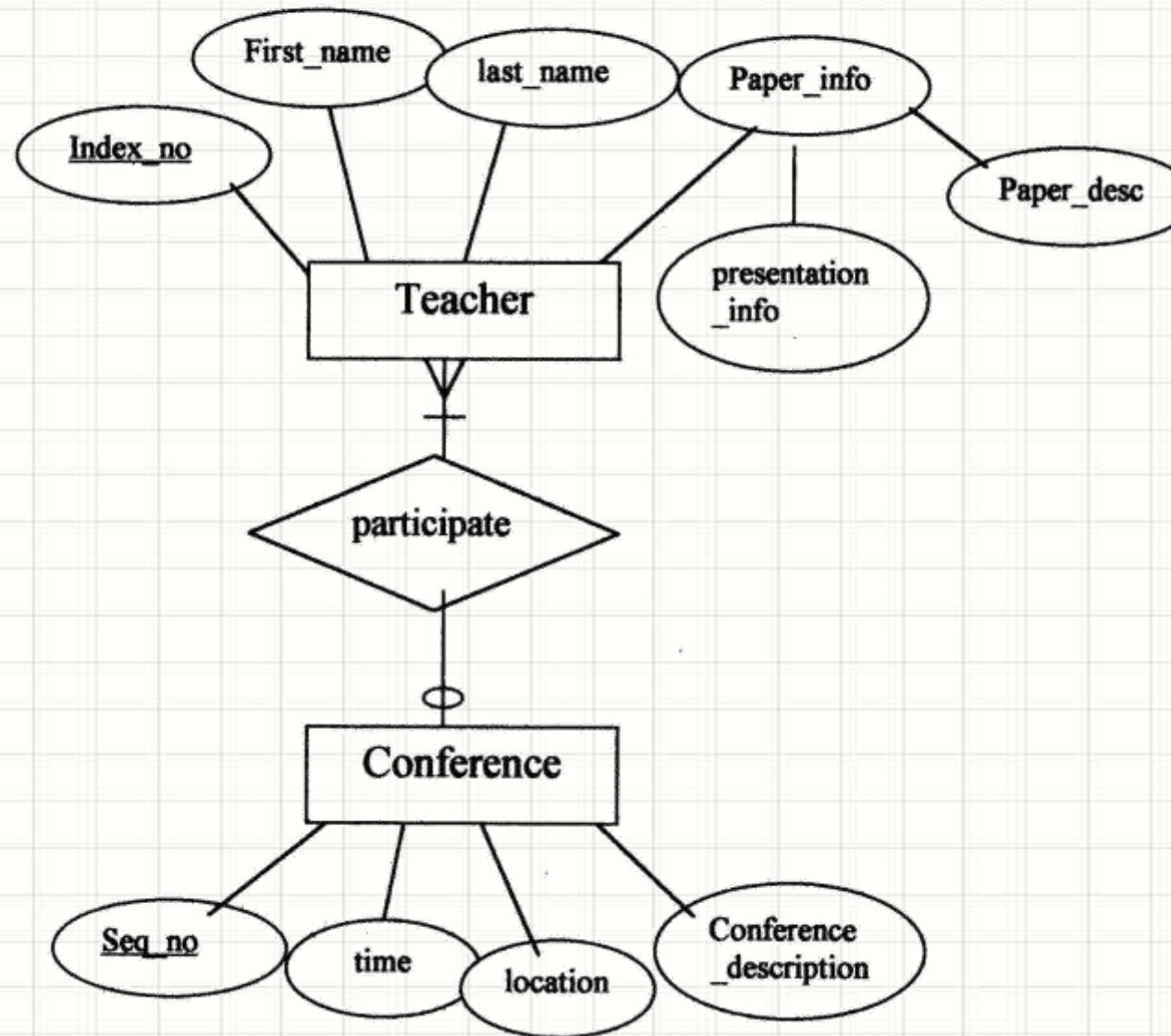
Qui tắc thiết kế

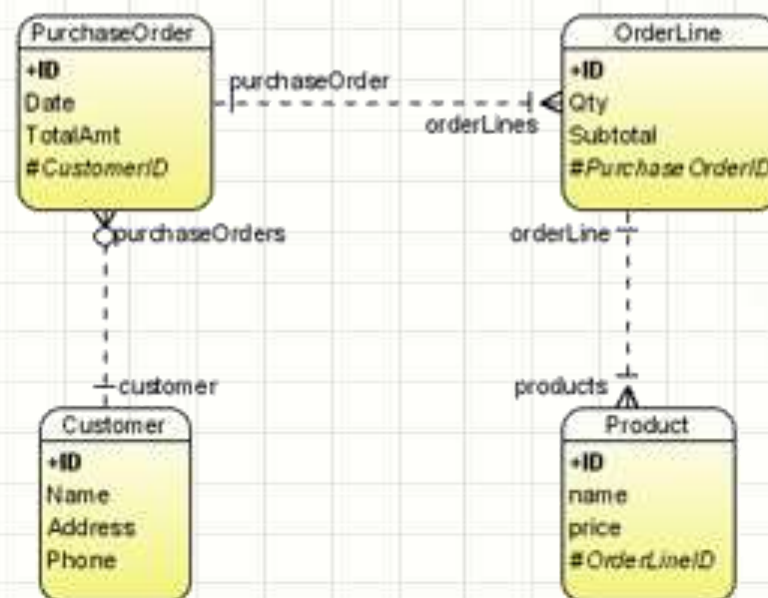
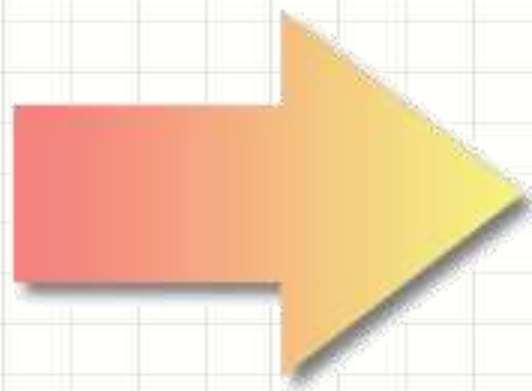
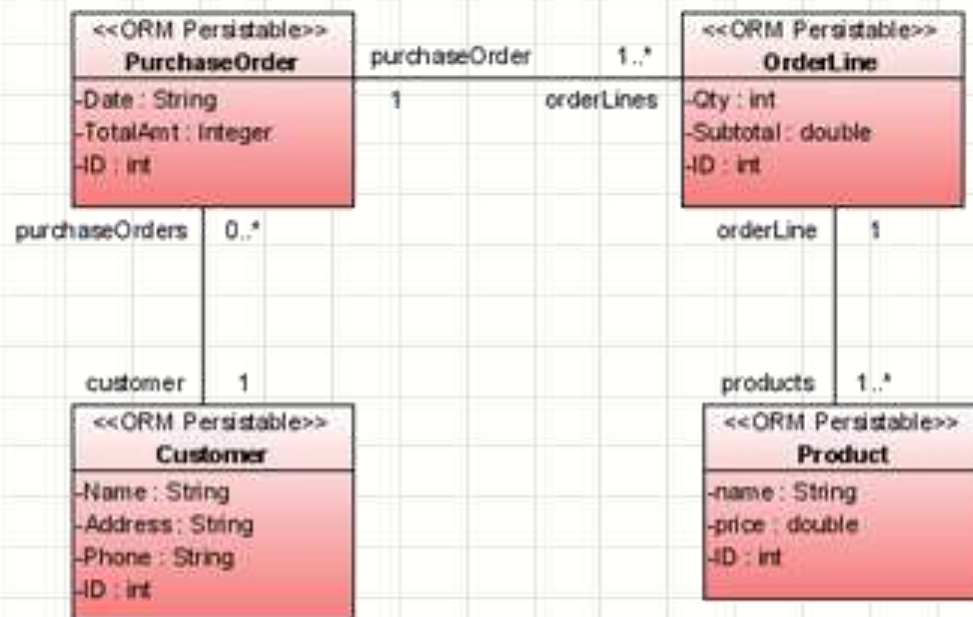
- Chính xác
- Tránh trùng lặp
- Dễ hiểu
- Chọn đúng mối quan hệ
- Chọn đúng kiểu thuộc tính

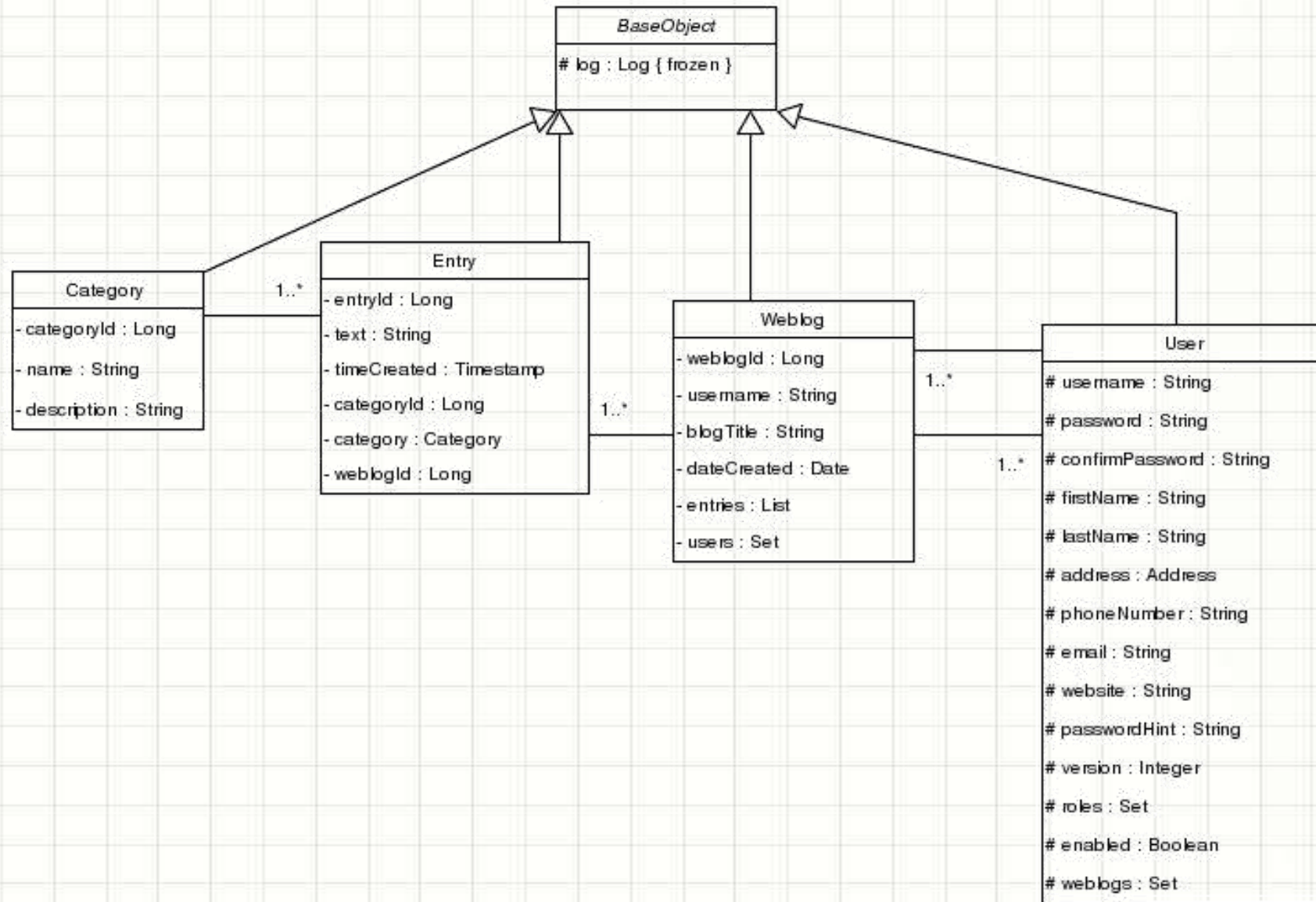
Một số ký hiệu chuẩn của Chen



Michigan Conference Management ER Diagram:











VÍ DỤ QUẢN LÝ ĐỀ ÁN CÔNG TY

Ví dụ 'Quản lý đề án công ty'

