

Đại học Khoa Học Tự Nhiên  
Khoa Công Nghệ Thông Tin

# PHÂN TÍCH THIẾT KẾ HỆ THÔNG THÔNG TIN

BM Hệ Thống Thông Tin  
Khoa Công Nghệ Thông Tin  
ĐH Khoa Học Tự Nhiên TPHCM  
GV. Ths. Nguyễn Trần Minh Thư

# Thiết kế dữ liệu

# Nội dung chi tiết

- Thiết kế luận lý dữ liệu
  - ◆ Cấp cao
  - ◆ Cấp thấp
- Thiết kế mã
- Thiết kế dữ liệu vật lý

# Thiết kế luận lý dữ liệu

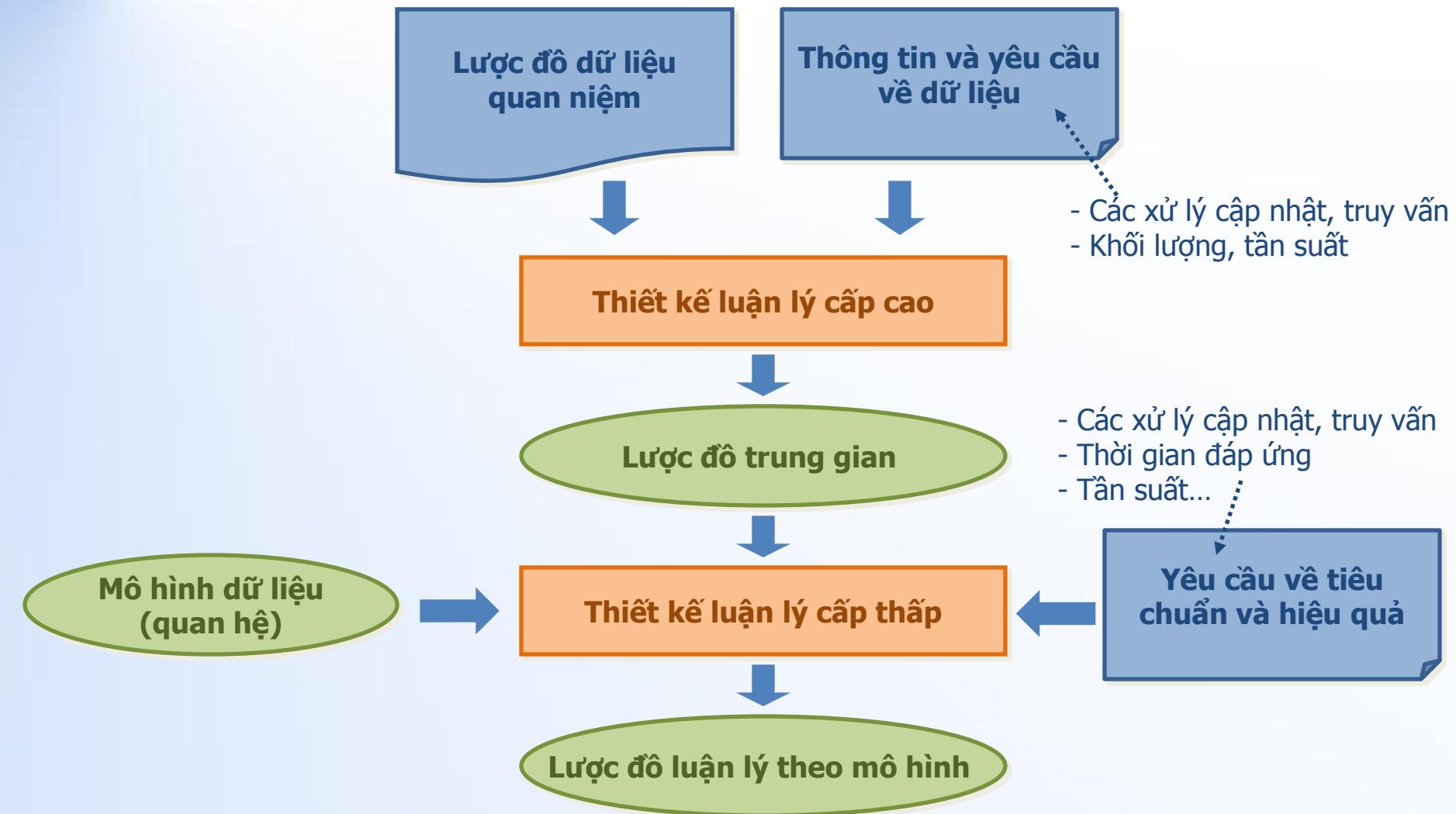
## ■ Cấp cao

- ◆ Độc lập với mô hình cài đặt
- ◆ Dùng chung cho nhiều loại mô hình dữ liệu

## ■ Cấp thấp

- ◆ Chuyển đổi lược đồ kết quả của thiết kế luận lý cấp cao sang 1 mô hình dữ liệu nhất định

# Thiết kế luận lý dữ liệu (tt)



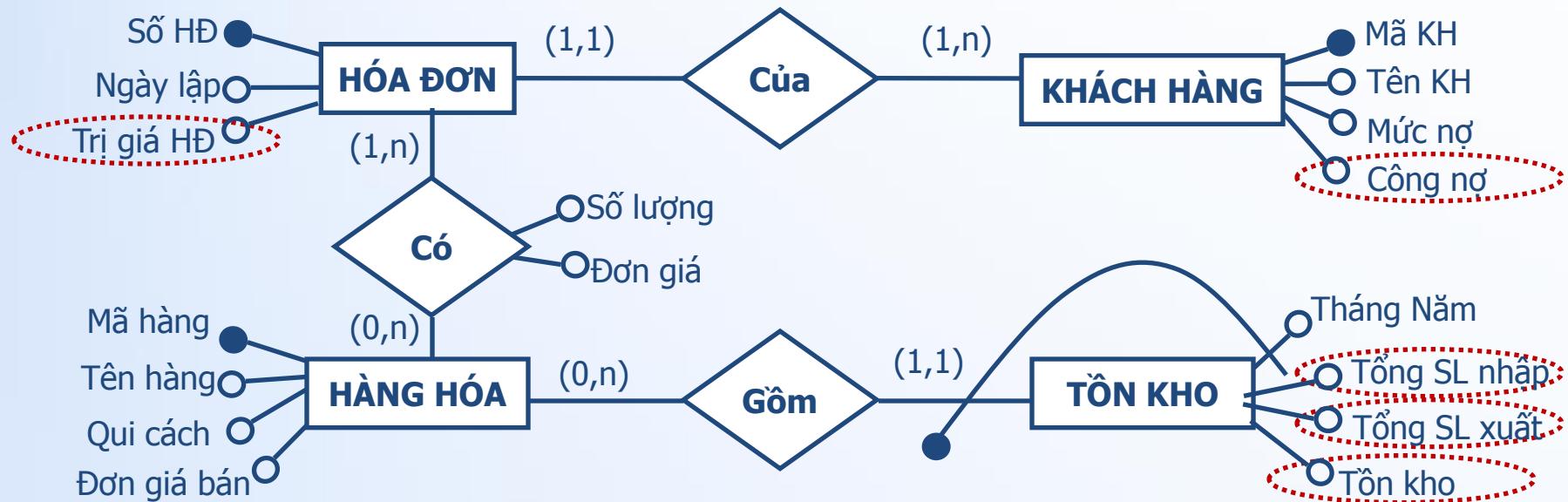
# Thiết kế luận lý dữ liệu cấp cao

- Quyết định dữ liệu suy diễn
- Chuyển đổi tổng quát hóa và tập con
- Chọn lựa khóa chính

# Quyết định dữ liệu suy diễn

## Dữ liệu suy diễn

- ◆ Là những thuộc tính mà giá trị của nó có thể tính toán số học từ những thuộc tính khác



# Quyết định dữ liệu suy diễn (tt)

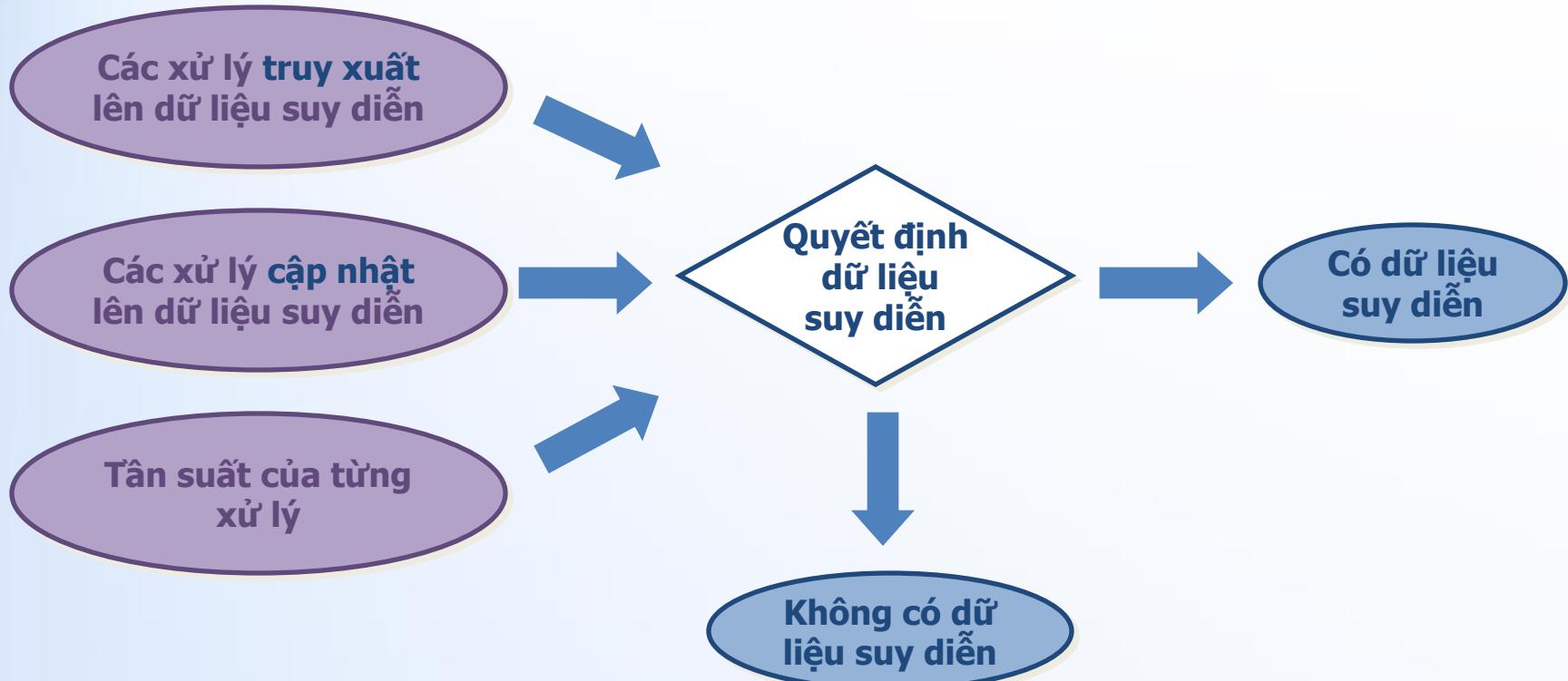
## ■ Ưu điểm

- ◆ Tăng tốc độ truy vấn
  - Không tính toán lại giá trị tại thời điểm truy vấn

## ■ Khuyết điểm

- ◆ Khi cập nhật phải kiểm tra tính nhất quán với các dữ liệu liên quan → giảm tốc độ cập nhật
- ◆ Tăng dung lượng lưu trữ
- ◆ Phải lập trình để bảo đảm ràng buộc

# Quyết định dữ liệu suy diễn (tt)



# Ví dụ



Khái niệm	Loại	Khối lượng
Khách hàng	Thực thể	15.000
Tài khoản	Thực thể	30.000
Giao dịch	Thực thể	600.000
Cửa	Mỗi kết hợp	30.000
Liên quan	Mỗi kết hợp	800.000

# Ví dụ (tt)

## ■ Có dữ liệu suy diễn (A)

Tên tác vụ	Khái niệm	Loại	Đọc/Ghi	Tần suất (Ngày)
Mở tài khoản	Tài khoản	Thực thể	Ghi	100
	Khách hàng	Thực thể	Ghi	100
	Của	Mối kết hợp	Ghi	100
Đọc tổng số dư khách hàng	Khách hàng	Thực thể	Đọc	3000
Gởi tiền	Tài khoản	Thực thể	Đọc	2000
	Khách hàng	Thực thể	Ghi	2000
	Tài khoản	Thực thể	Đọc	2000
	Khách hàng	Thực thể	Ghi	2000
Rút tiền	Tài khoản	Thực thể	Đọc	1000
	Khách hàng	Thực thể	Ghi	1000
	Tài khoản	Thực thể	Đọc	1000
	Khách hàng	Thực thể	Ghi	1000
<b>Khách hàng</b>			<b>Đọc</b> <b>Ghi</b>	<b>3000</b> <b>3000</b>

## ■ Không có dữ liệu suy diễn (B)

Tên tác vụ	Khái niệm	Loại	Đọc/Ghi	Tần suất (Ngày)
Mở tài khoản	Tài khoản Khách hàng Của	Thực thể Thực thể Mỗi kết hợp	Ghi Ghi Ghi	100 100 100
Đọc tổng số dư khách hàng	Khách hàng <b>Tài khoản</b> Của	Thực thể Thực thể Mỗi kết hợp	Đọc Đọc Đọc	3000 3000x2 3000x2
Gởi tiền	Tài khoản	Thực thể	Đọc Ghi	2000 2000
Rút tiền	Tài khoản	Thực thể	Đọc Ghi	1000 1000

<b>Tài khoản</b> Của	<b>Đọc</b> <b>Đọc</b>	<b>6000</b> <b>6000</b>
-------------------------	--------------------------	----------------------------

- Nếu  $A \gg B$

- ◆ Không sử dụng thuộc tính suy diễn

- Nếu  $A \ll B$

- ◆ Sử dụng thuộc tính suy diễn

- Chú ý

- ◆ Có những nhu cầu phát sinh 1 cách định tính  
→ không thể quyết định bằng phương pháp  
định lượng

# Chuyển đổi tổng quát hóa & tập con

## ■ Cân thiết khi

- ◆ Mô hình cài đặt không hỗ trợ
  - Mô hình quan hệ, mạng, phân cấp...

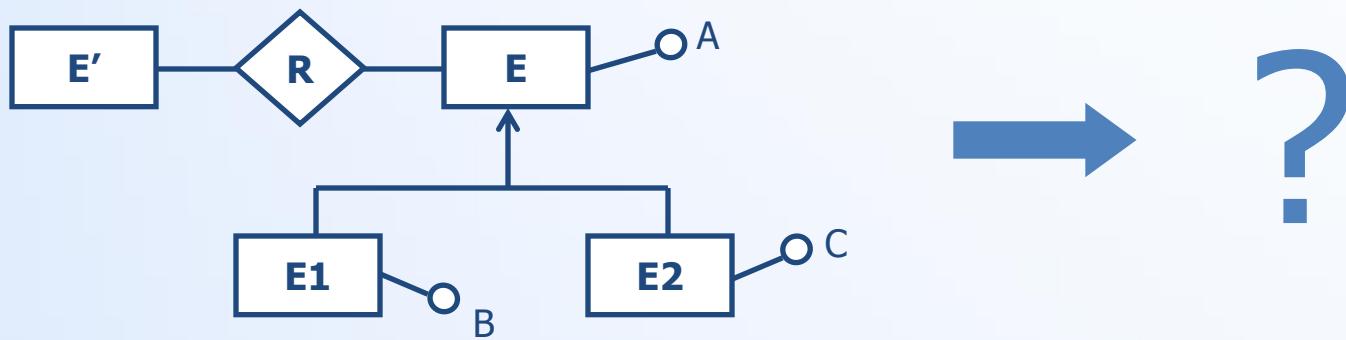
## ■ Xem xét

- ◆ Các đặc trưng kế thừa của thực-thể-chuyên-biệt từ thực-thể-tổng-quát
  - Thuộc tính, định danh, mối kết hợp
- ◆ Mối kết hợp tổng quát hóa

# Chuyển đổi tổng quát hóa & tập con

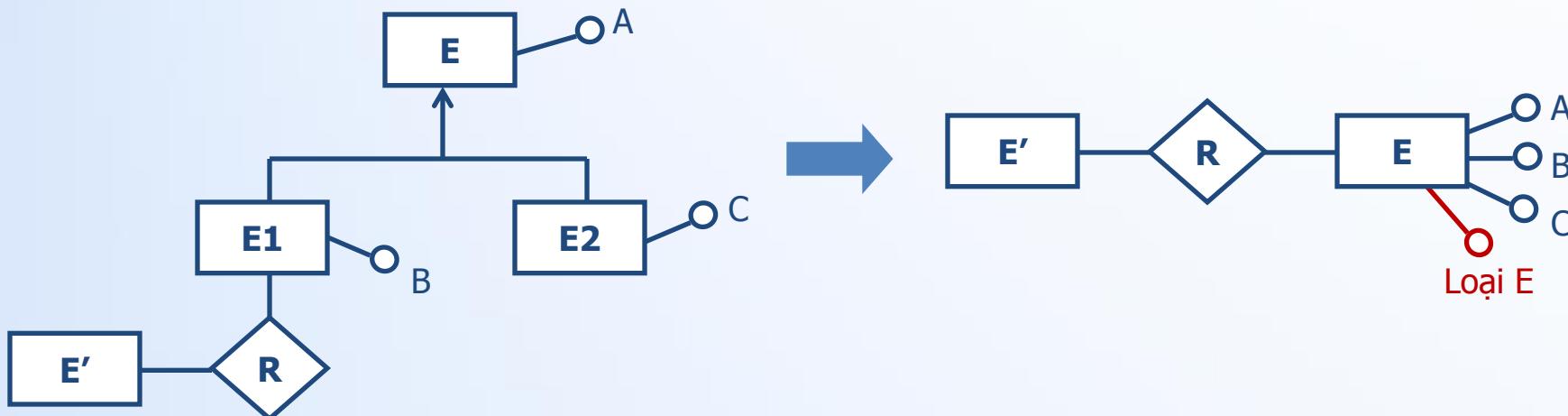
## ■ Phương án chọn lựa

- ◆ Dùng thực thể tổng quát
- ◆ Dùng thực thể chuyên biệt
- ◆ Dùng mối kết hợp



# Dùng thực thể tổng quát

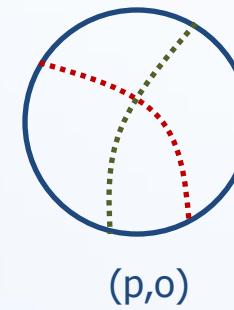
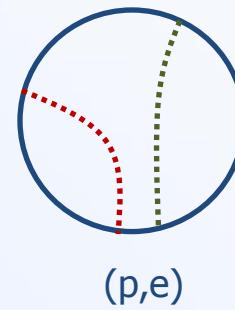
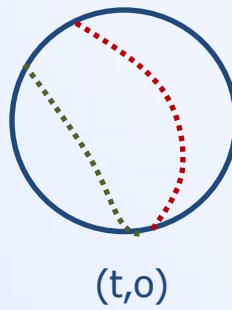
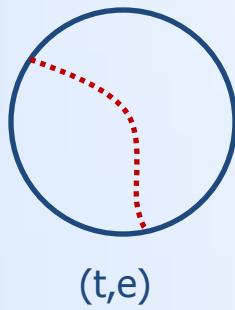
- Các đặc trưng của thực-thể-chuyên-biệt sẽ được chuyển sang thực-thể-tổng-quát
- Thêm thuộc tính LoạiE
  - ◆ MGT(LoạiE) = {E, E1, E2, E1E2}



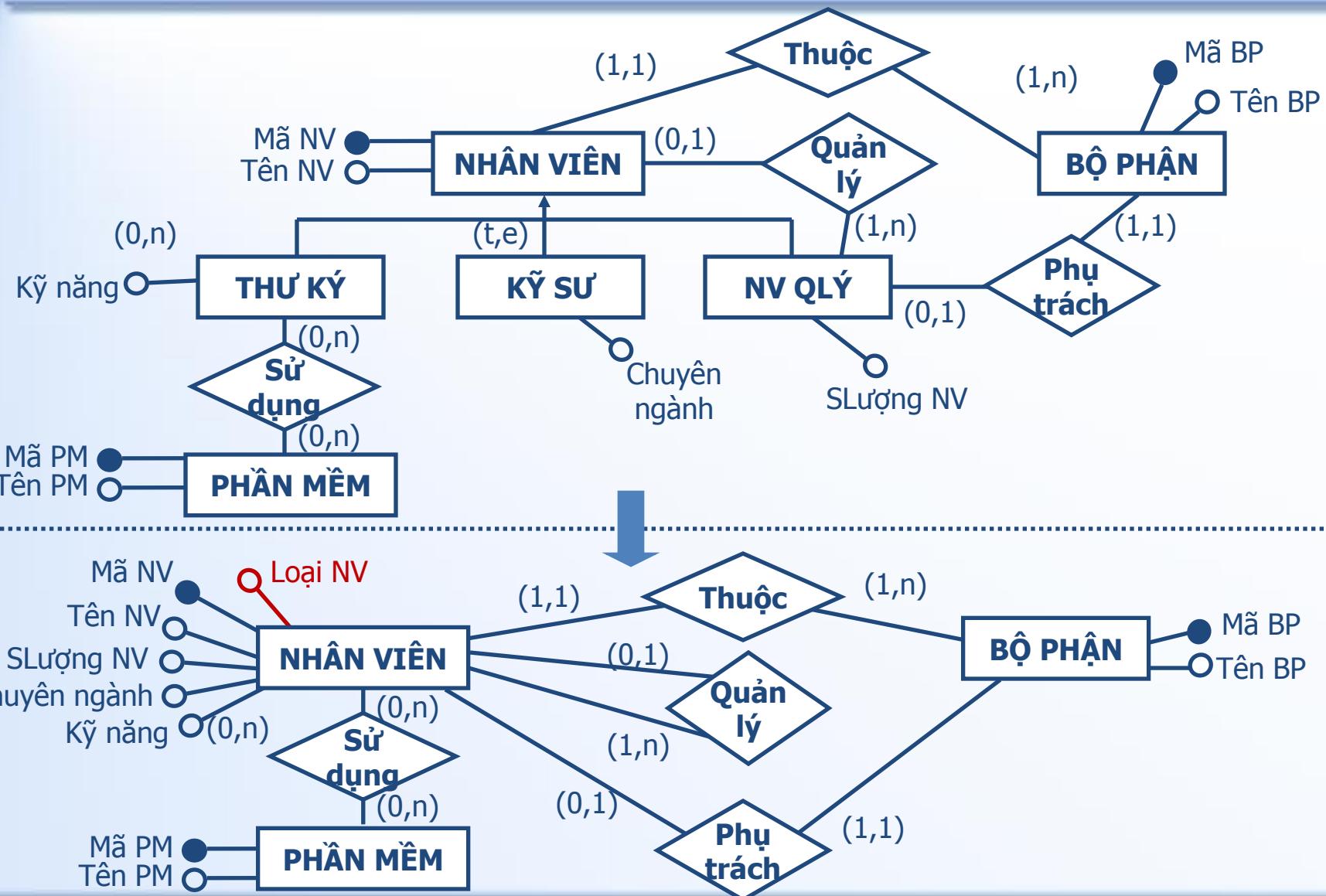
# Dùng thực thể tổng quát (tt)

## ■ Miền giá trị LoạiE

- ◆  $(t,e) : MGT(\text{LoạiE}) = \{E_1, E_2\}$
- ◆  $(t,o) : MGT(\text{LoạiE}) = \{E_1, E_2, E_1E_2\}$
- ◆  $(p,e) : MGT(\text{LoạiE}) = \{E, E_1, E_2\}$
- ◆  $(p,o) : MGT(\text{LoạiE}) = \{E, E_1, E_2, E_1E_2\}$



# Ví dụ

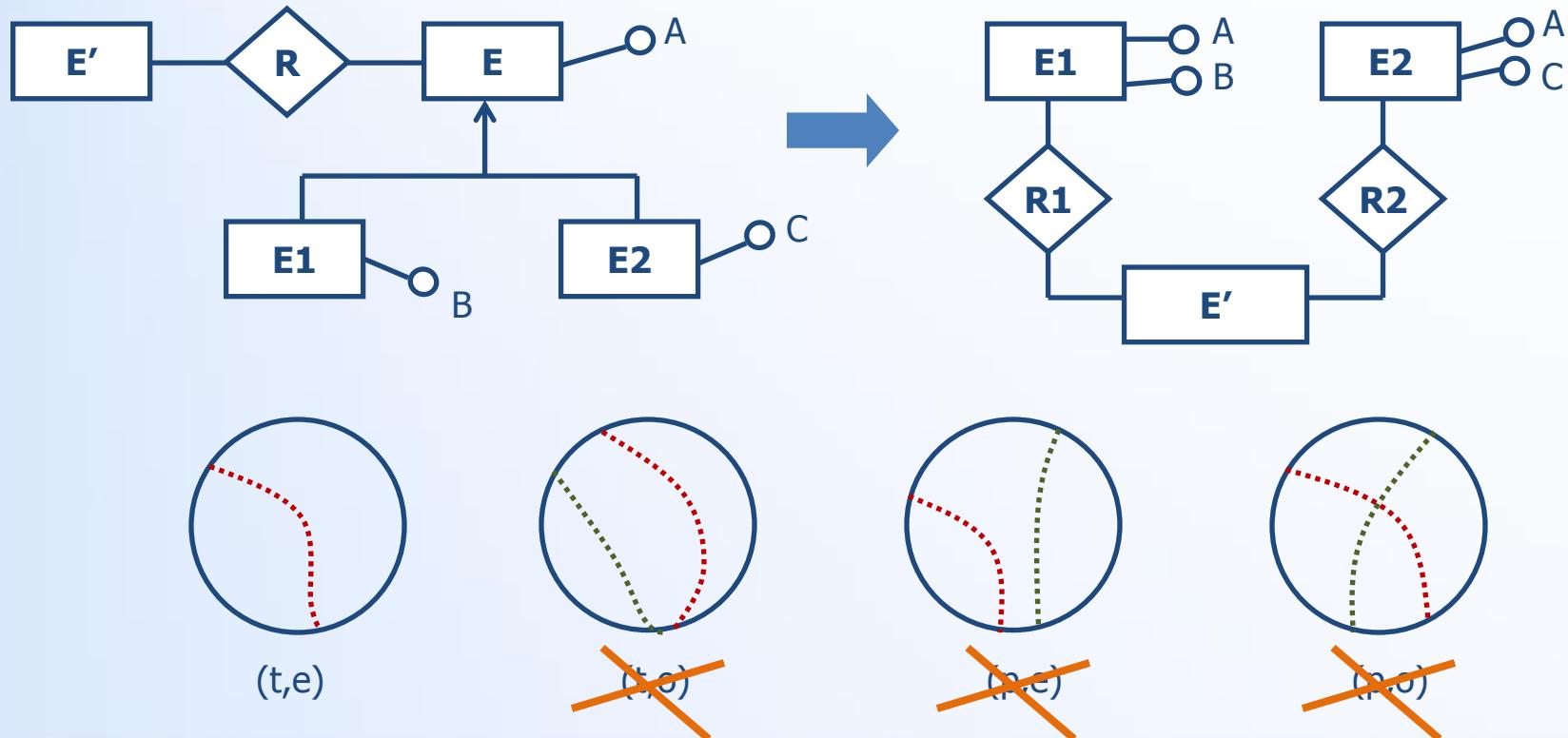


# Dùng thực thể tổng quát (tt)

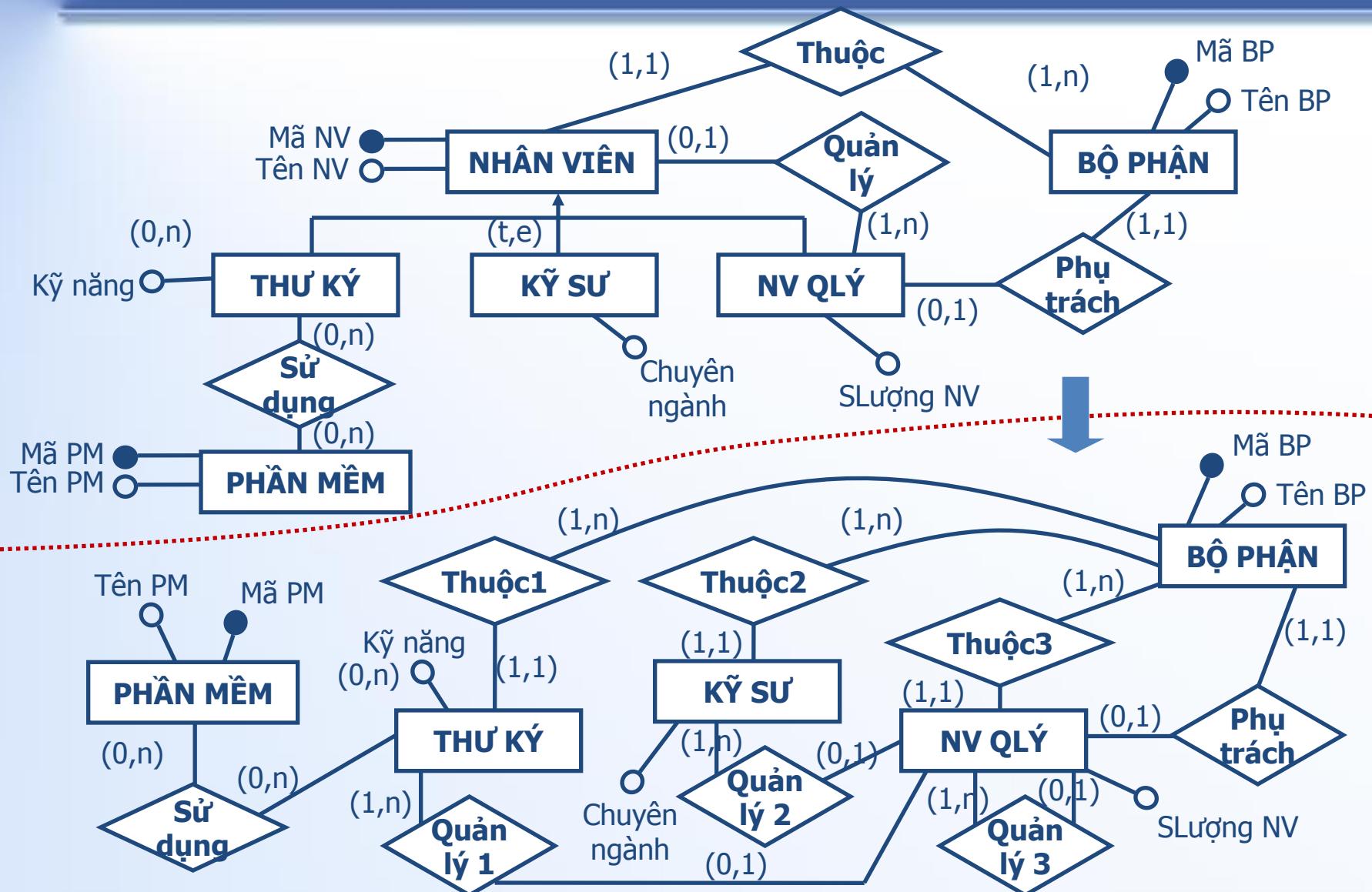
Ưu điểm	Khuyết điểm
p	thôi
t (e)	<p>p con</p> <p>Phát sinh thêm một số RBTV cần phải kiểm tra</p>

# Dùng thực thể chuyên biệt

- Chuyển đổi tất cả đặc trưng của thực-thể-tổng-quát xuống các thực-thể-chuyên-biệt



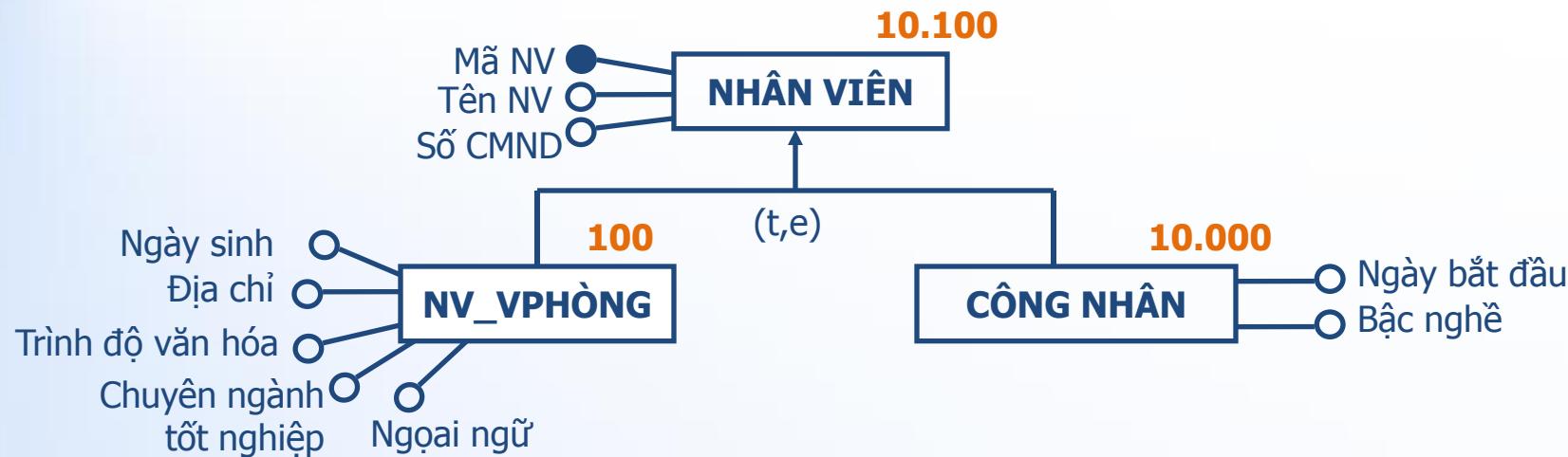
# Ví dụ



# Dùng thực thể chuyên biệt (tt)

Ưu điểm	Khuyết điểm
	t (e)
	i p con

# Lựa chọn chuyên biệt | tổng quát



Xử lý	Tần suất	Chọn lựa tối ưu
(o1) Tính lương sản phẩm cho công nhân phân xưởng	2/ tháng	Tách (gộp)
(o2) Tính lương cho nhân viên văn phòng	1/tháng	Tách
(o3) Tìm kiếm thông tin về công nhân	1000/ngày	Tách (gộp)
(o4) Tổng hợp danh sách chung của toàn bộ nhân viên	5/tháng	Gộp
(o5) Truy xuất thông tin nhân viên văn phòng	20/tháng	Tách

# Lựa chọn chuyên biệt | tổng quát

## ■ Phụ thuộc

### ◆ Dung lượng

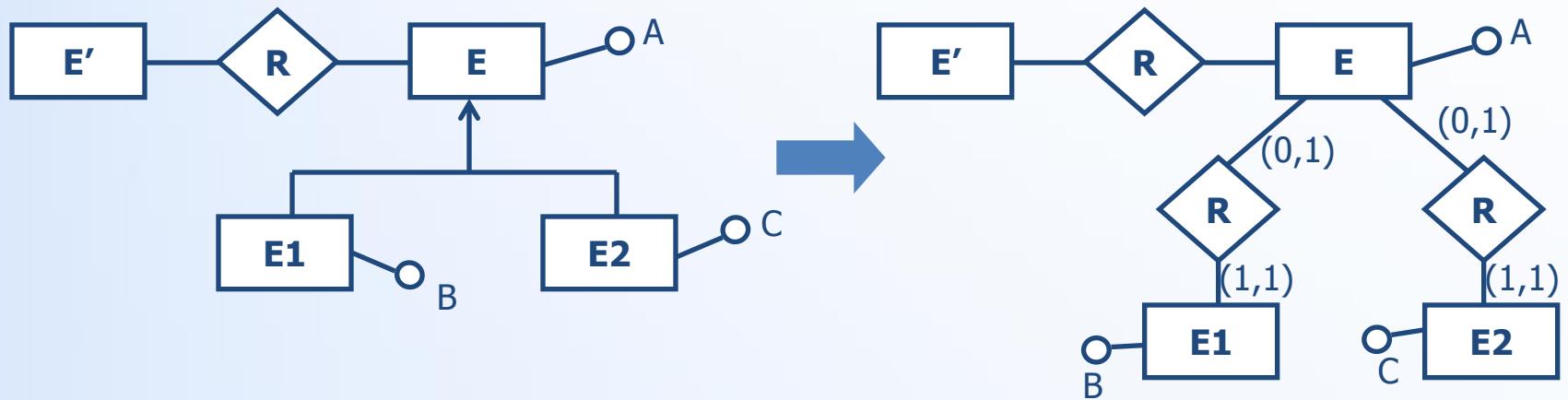
- Gộp: dung lượng lớn
- Tách: dung lượng tối ưu

### ◆ Xử lý

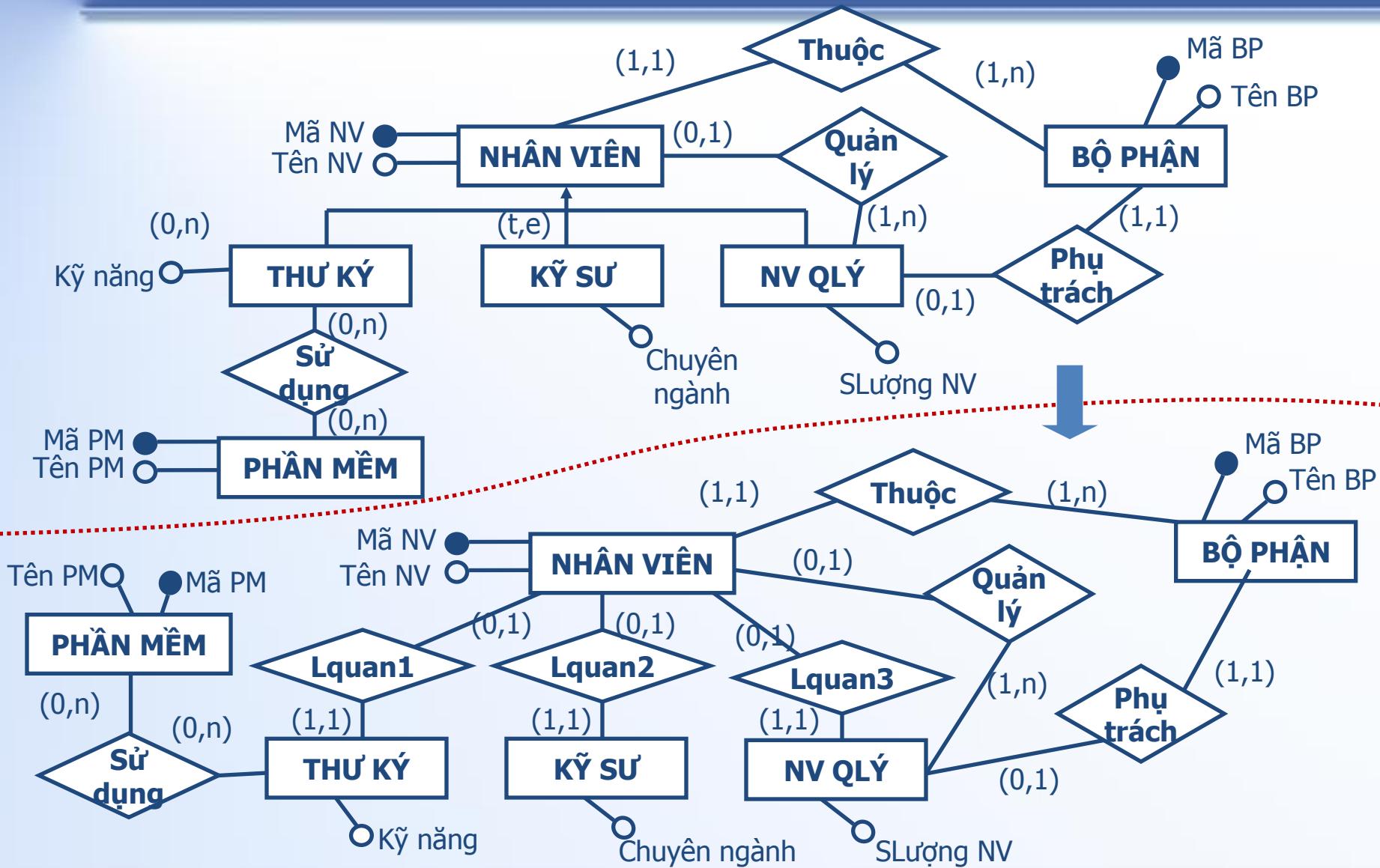
- Ưu tiên các xử lý có tần suất cao
- Ví dụ
  - Nếu ưu tiên o4 : gộp
  - Nếu ưu tiên o2, o3: tách

# Dùng mối kết hợp

- Bỏ tổng quát, chuyên biệt
- Tạo mối kết hợp từ thực-thể-tổng-quát đến thực-thể-chuyên-biệt



# Ví dụ



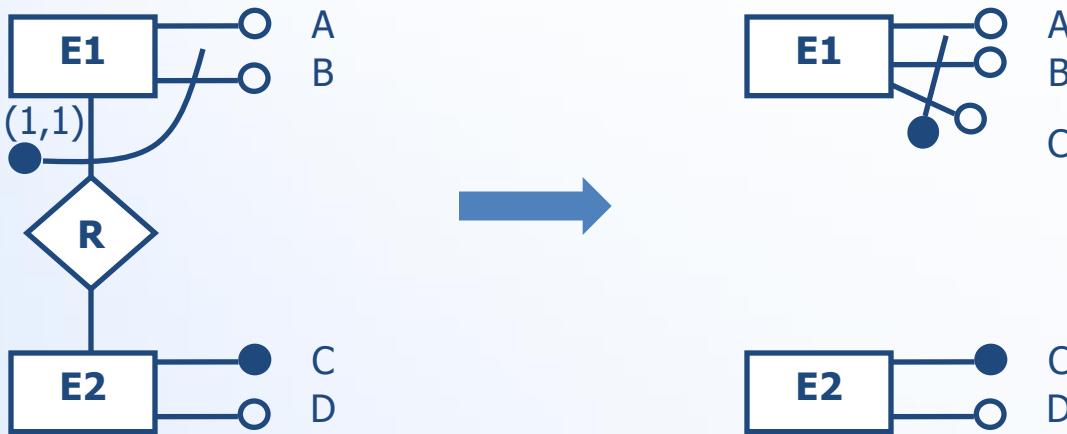
# Dùng mồi kết hợp (tt)

Ưu điểm	Khuyết điểm
t	a
ng	p -

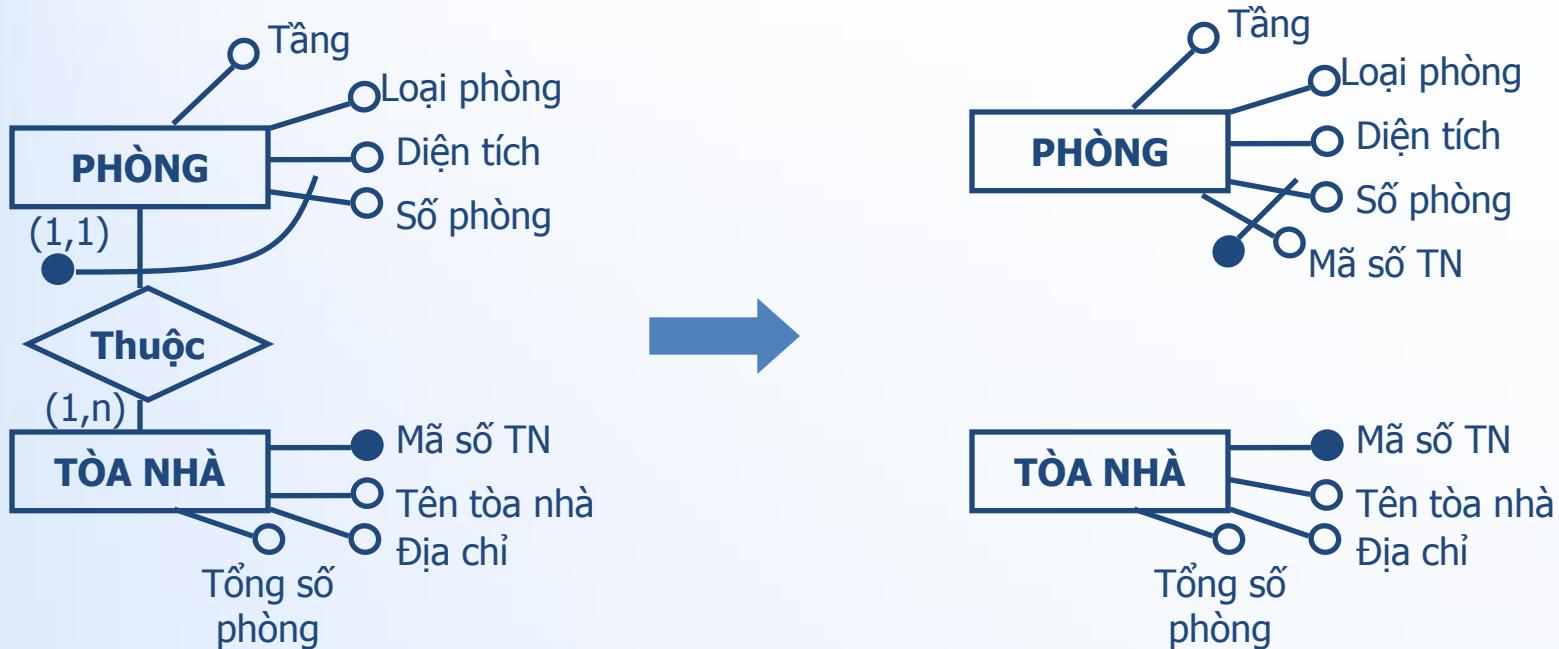
# Thiết kế luận lý dữ liệu cấp thấp

- Chuẩn bị
  - ◆ Loại bỏ định danh ngoài
  - ◆ Loại bỏ thuộc tính đa trị và thuộc tính kết hợp
- Chuyển đổi mô hình thực thể kết hợp sang mô hình quan hệ

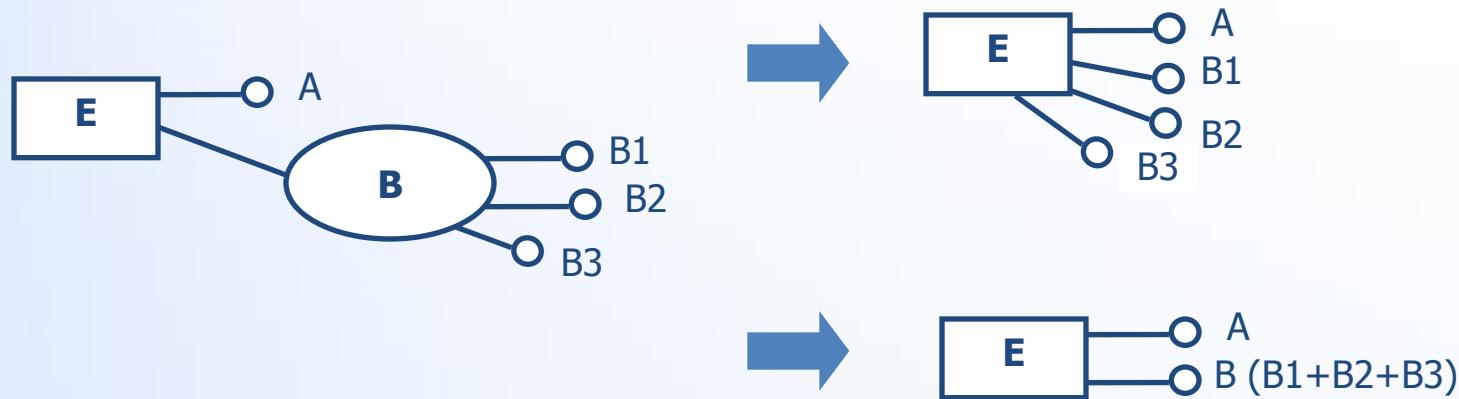
# Loại bỏ định danh bên ngoài



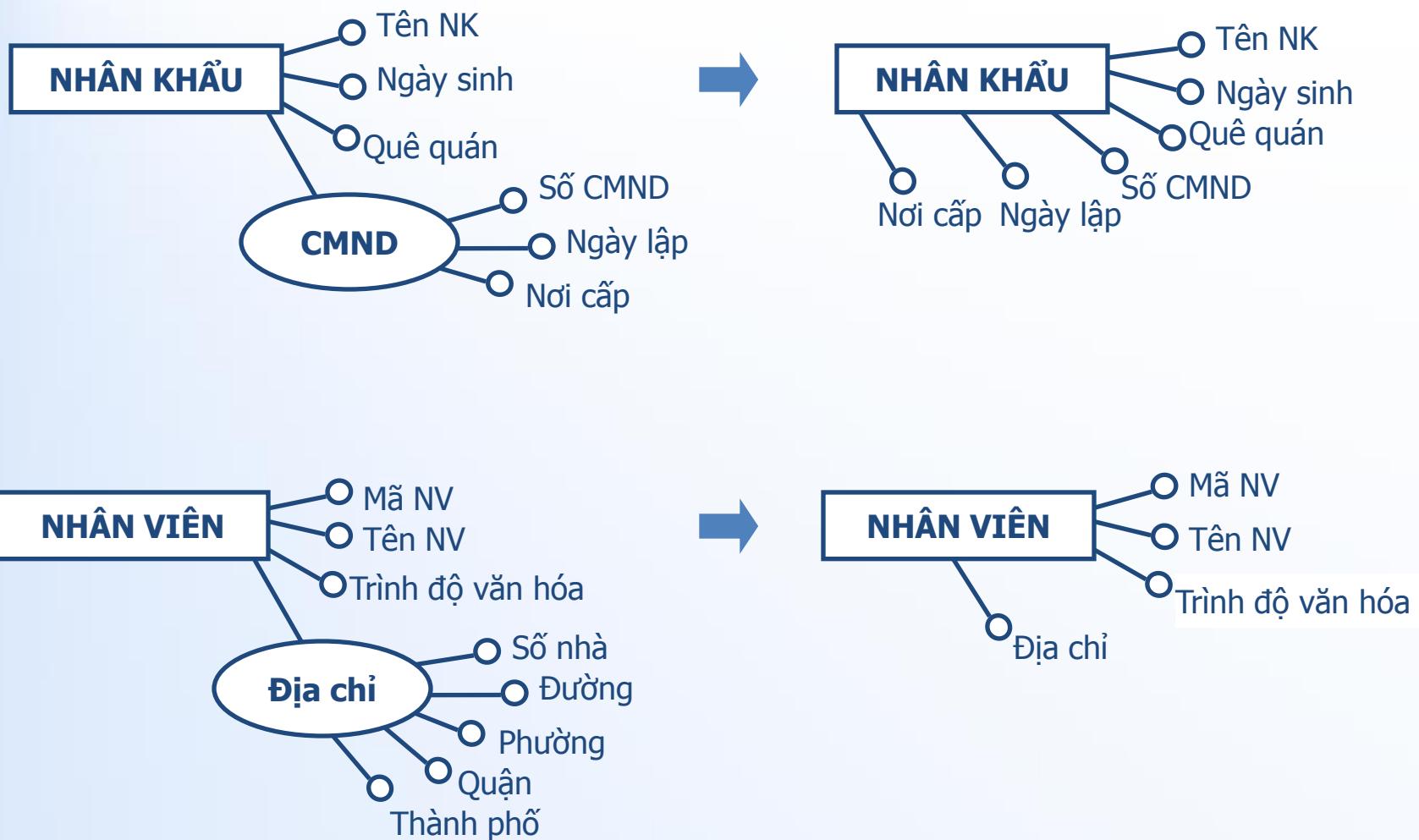
# Ví dụ



# Loại bỏ thuộc tính kết hợp



# Ví dụ

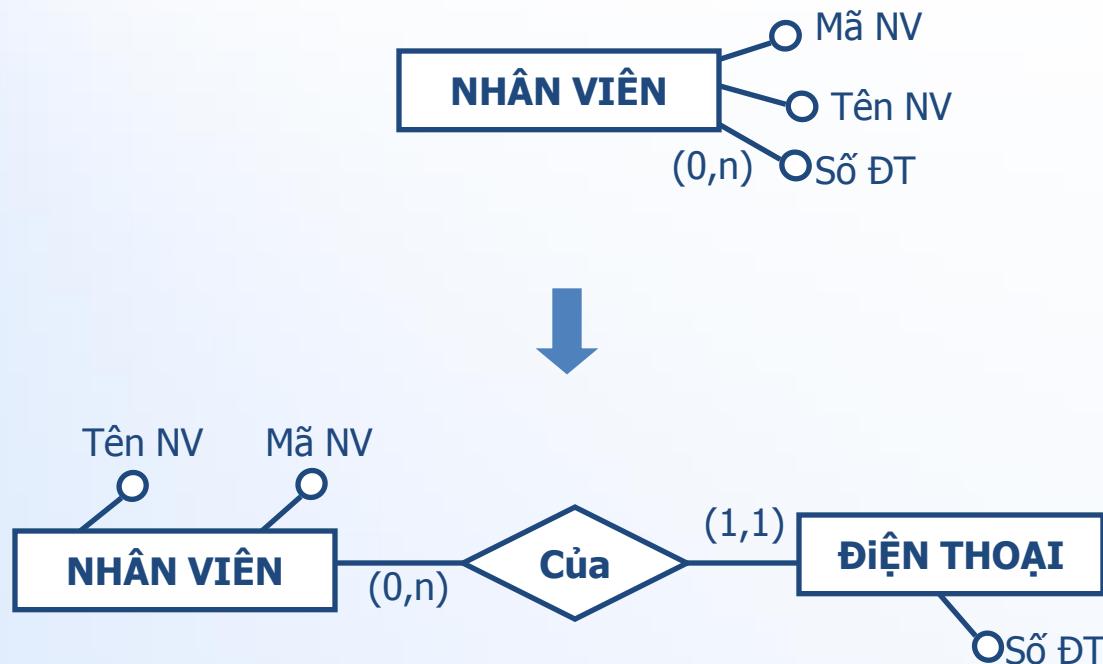


# Loại bỏ thuộc tính đa trị

## ■ Thuộc tính đa trị ở thực thể



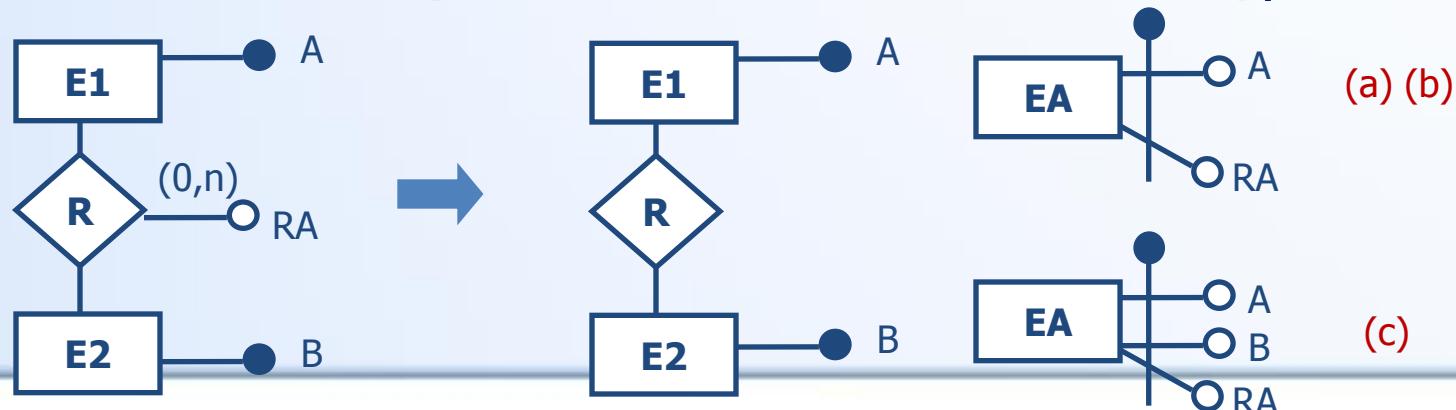
# Ví dụ



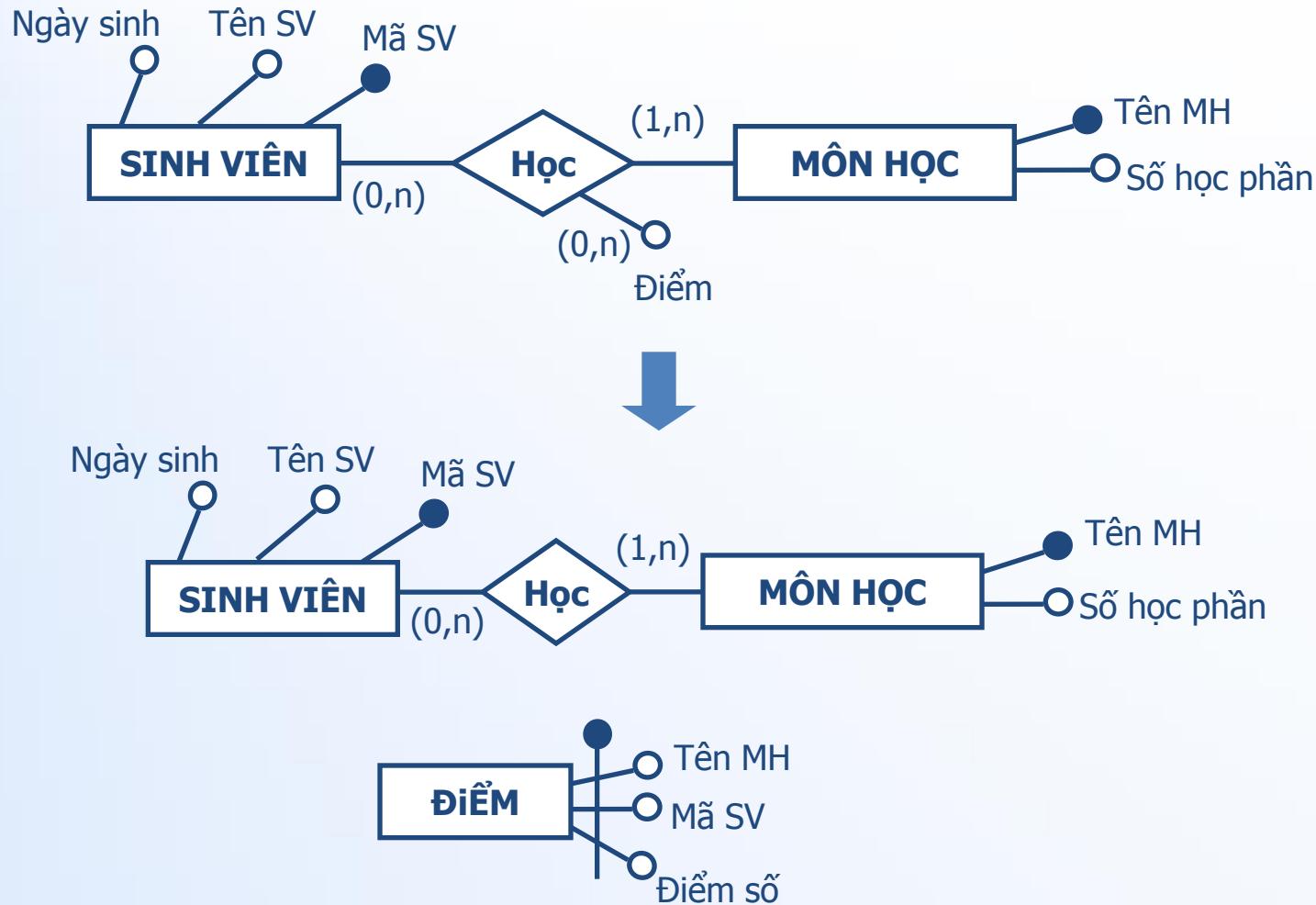
# Loại bỏ thuộc tính đa trị (tt)

## ■ Thuộc tính đa trị ở mỗi kết hợp

- ◆ (a) R là mỗi kết hợp 1-1
  - EA sẽ có định danh của E1 hoặc E2 kết hợp với RA
- ◆ (b) R là mỗi kết hợp 1-N
  - EA sẽ có định danh của E1 kết hợp với RA
- ◆ (c) R là mỗi kết hợp N-N
  - EA sẽ có định danh của E1 và E2 kết hợp với RA



# Ví dụ

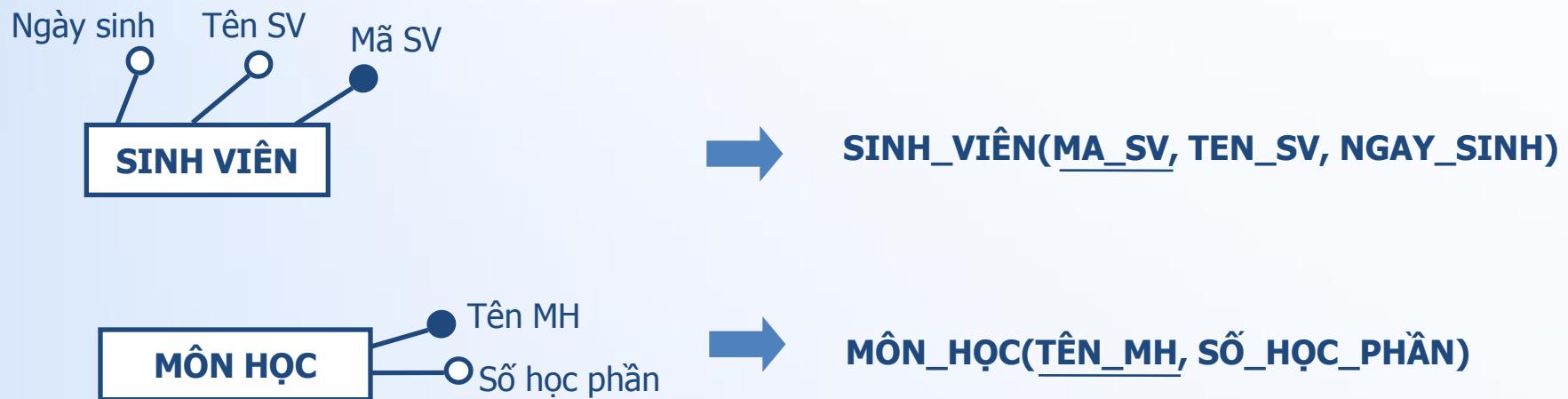


# Chuyển mô hình ER → Quan hệ

- Chuyển đổi thực thể
- Chuyển đổi mối kết hợp

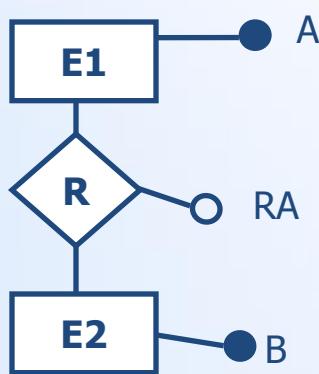
# Chuyển đổi thực thể

- 1 thực thể → 1 quan hệ
- Thuộc tính → thuộc tính
- Định danh → khóa chính

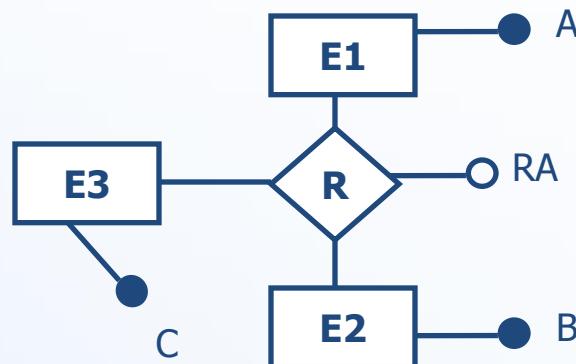


# Chuyển đổi môi kết hợp

## ■ Qui tắc chung



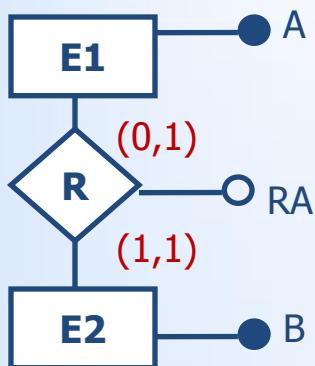
R(A, B, RA)



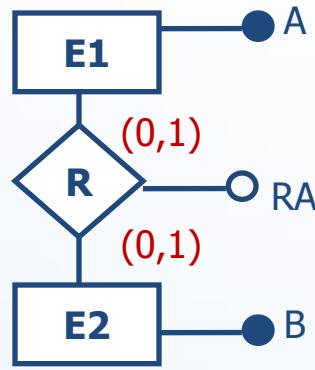
R(A, B, C, RA)

# Chuyển đổi môi kết hợp (tt)

## Môi kết hợp 1-1



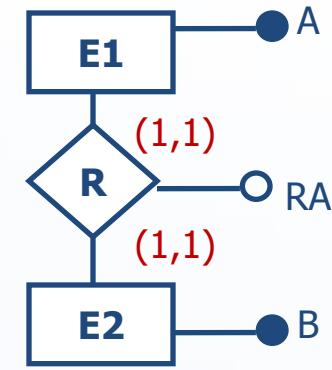
$E1(\underline{A}, \dots)$   
 $E2(\underline{B}, \dots, A)$ : A là khóa



$E1(\underline{A}, \dots)$   
 $E2(\underline{B}, \dots, A)$ : A là khóa

Hoặc

$E1(\underline{A}, \dots, B)$ : B là khóa  
 $E2(\underline{B}, \dots)$

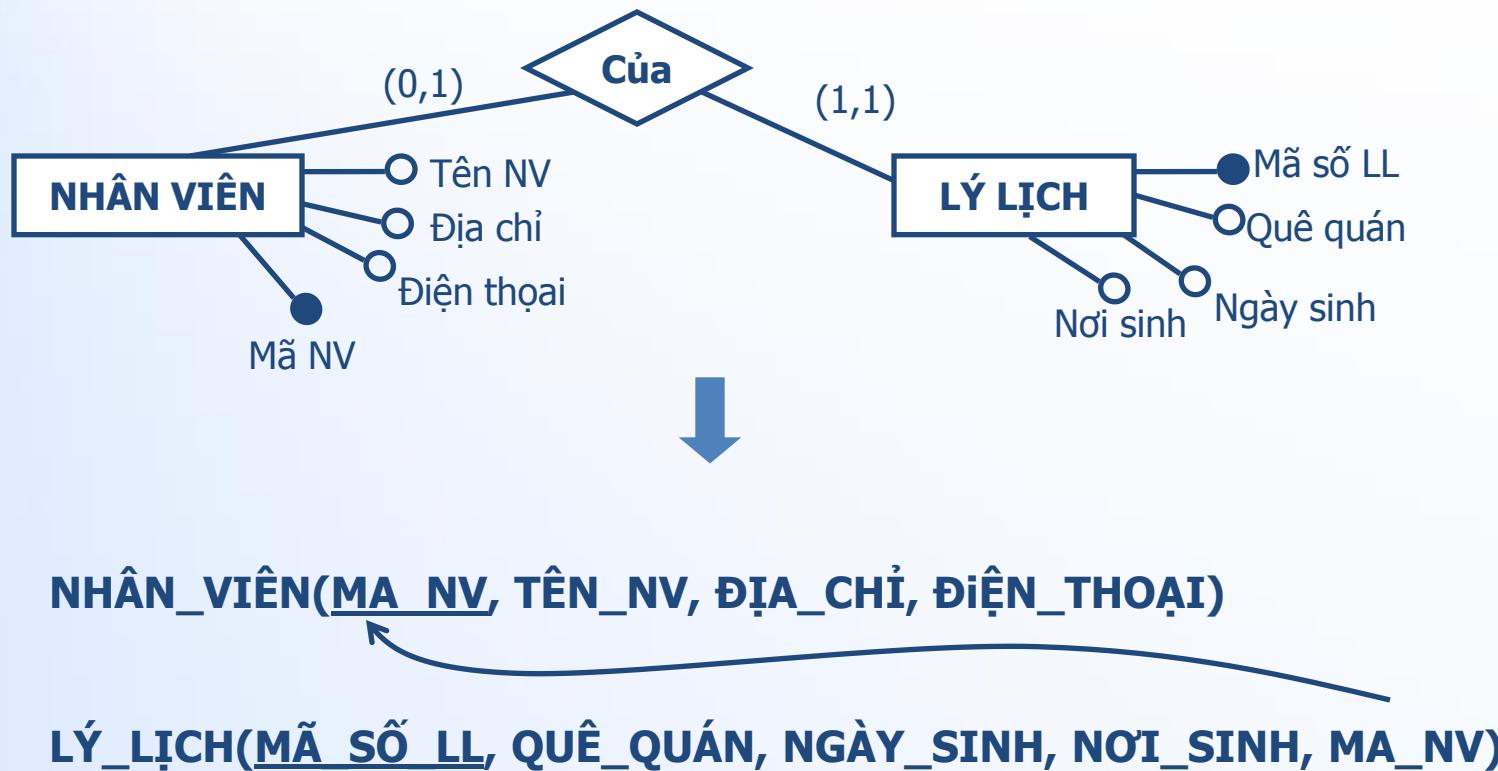


$E1E2(\underline{A}, \underline{B}, \dots)$ : B là khóa

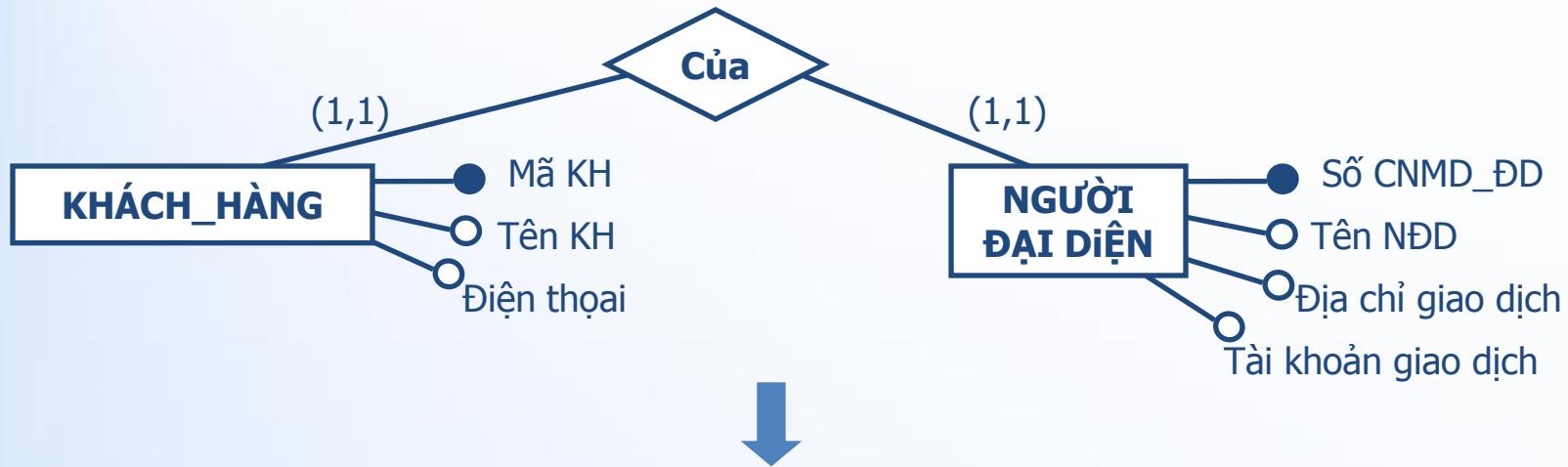
Hoặc

$E1E2(\underline{A}, \underline{B}, \dots)$ : A là khóa

# Ví dụ



# Ví dụ (tt)



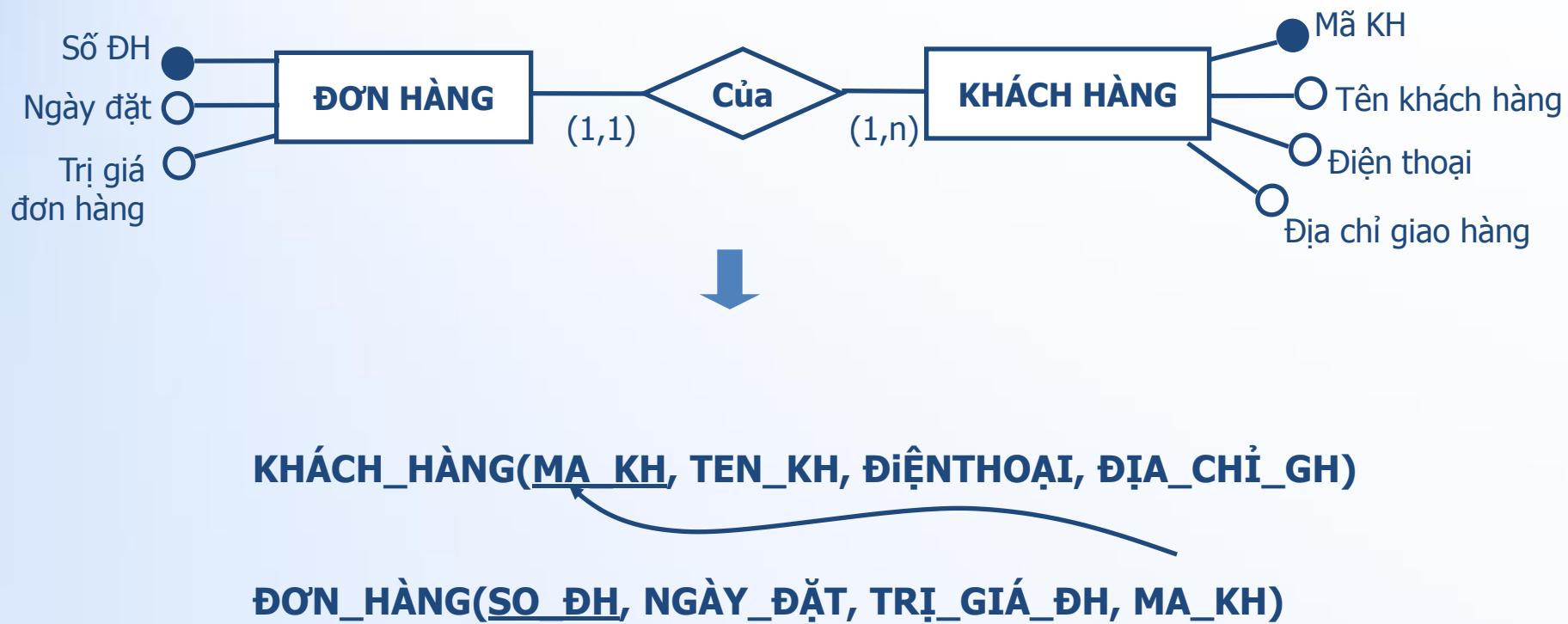
**KH\_NĐDIỆN (MÃ\_KH, TÊN\_KH, ĐIỆN\_THOẠI, SỐ\_CMND\_DD, TÊN\_NĐD,  
ĐỊA\_CHỈ\_GIAO\_DỊCH, TÀI\_KHOẢN\_GIAO\_DỊCH)**

# Chuyển đổi môi kết hợp (tt)

## ■ Mối kết hợp 1-N

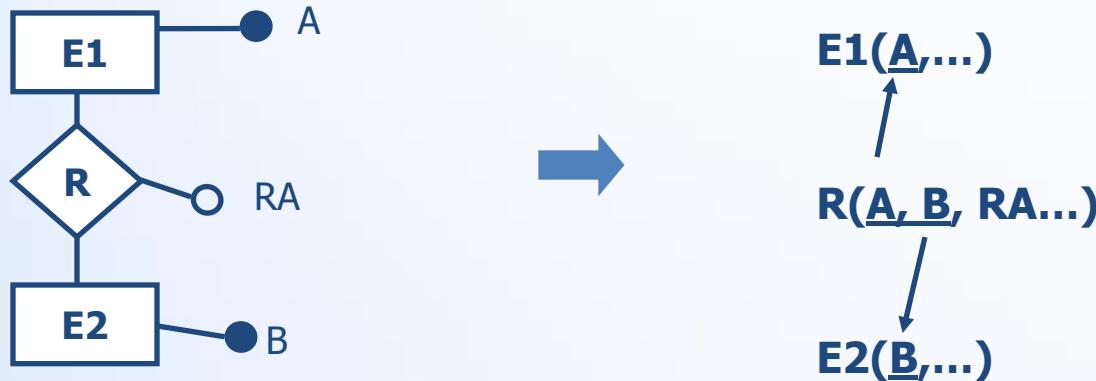


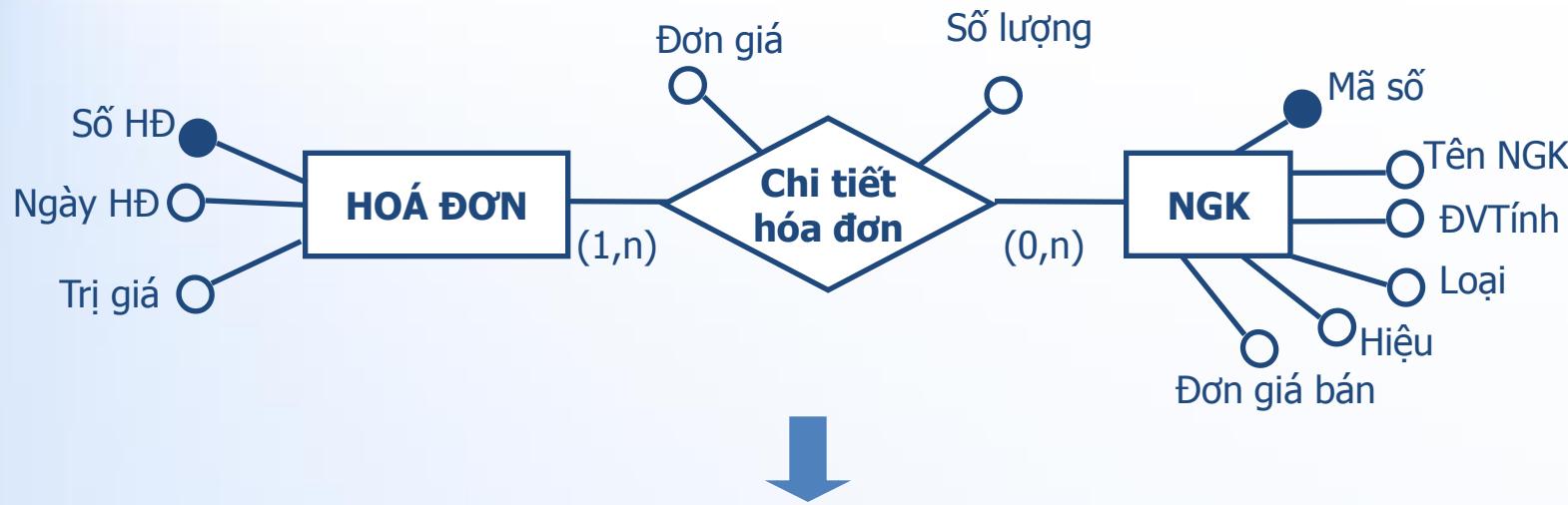
# Ví dụ



# Chuyển đổi môi kết hợp (tt)

## ■ Môi kết hợp N-N





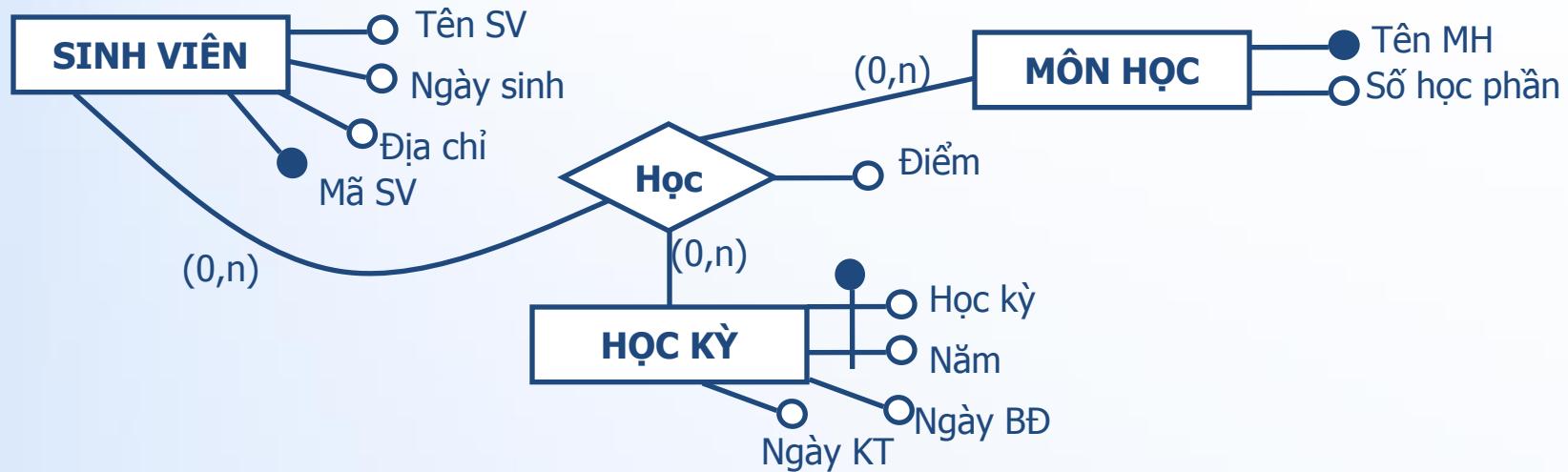
**HÓA ĐƠN(SỐ HD, NGÀY HD, TRỊ GIÁ)**

**CHITIET\_HD(SO HD, MÃ SỐ, SỐ LƯỢNG, ĐƠN GIÁ)**

**NGK(MÃ SỐ, TÊN NGK, ĐVTÍNH, LoAI, HiỆU, ĐƠNGIÁ\_BÁN)**

# Chuyển đổi mối kết hợp (tt)

## Mối kết hợp đa phân



**SINH\_VIÊN(MÃ\_SV, TÊN\_SV, NGÀY\_SINH, ĐỊA\_CHỈ)**

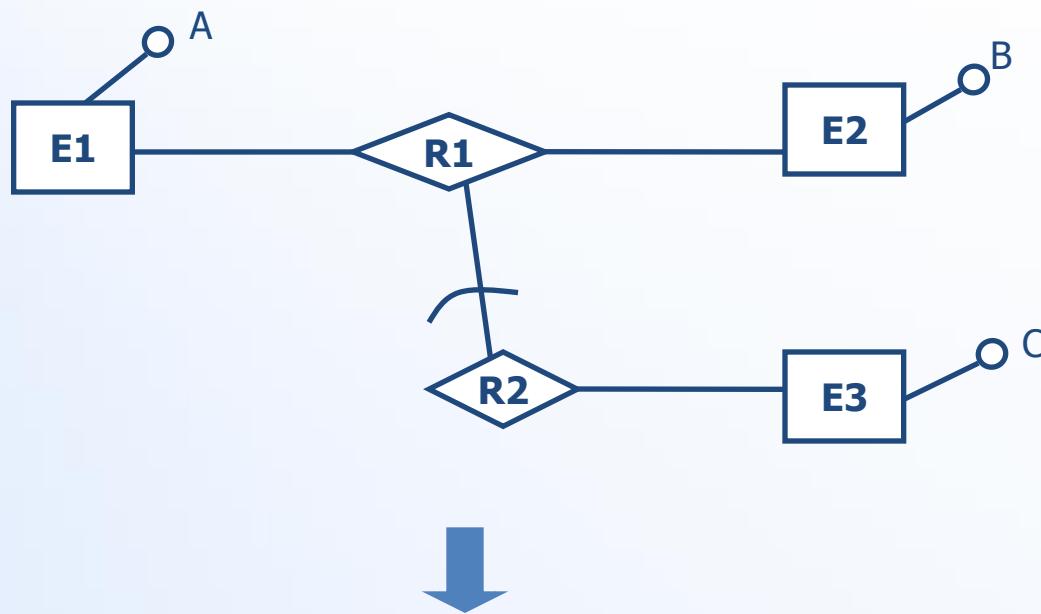
**MÔN\_HỌC(TÊN\_MH, SỐ\_HỌC\_PHẦN)**

**HỌC\_KỲ(HỌC\_KỲ, NĂM, NGÀY\_BĐ, NGÀY\_KT)**

**HỌC(MÃ\_SV, TÊN\_MH, HỌC\_KỲ, NĂM, ĐIỂM)**

# Chuyển đổi môi kết hợp (tt)

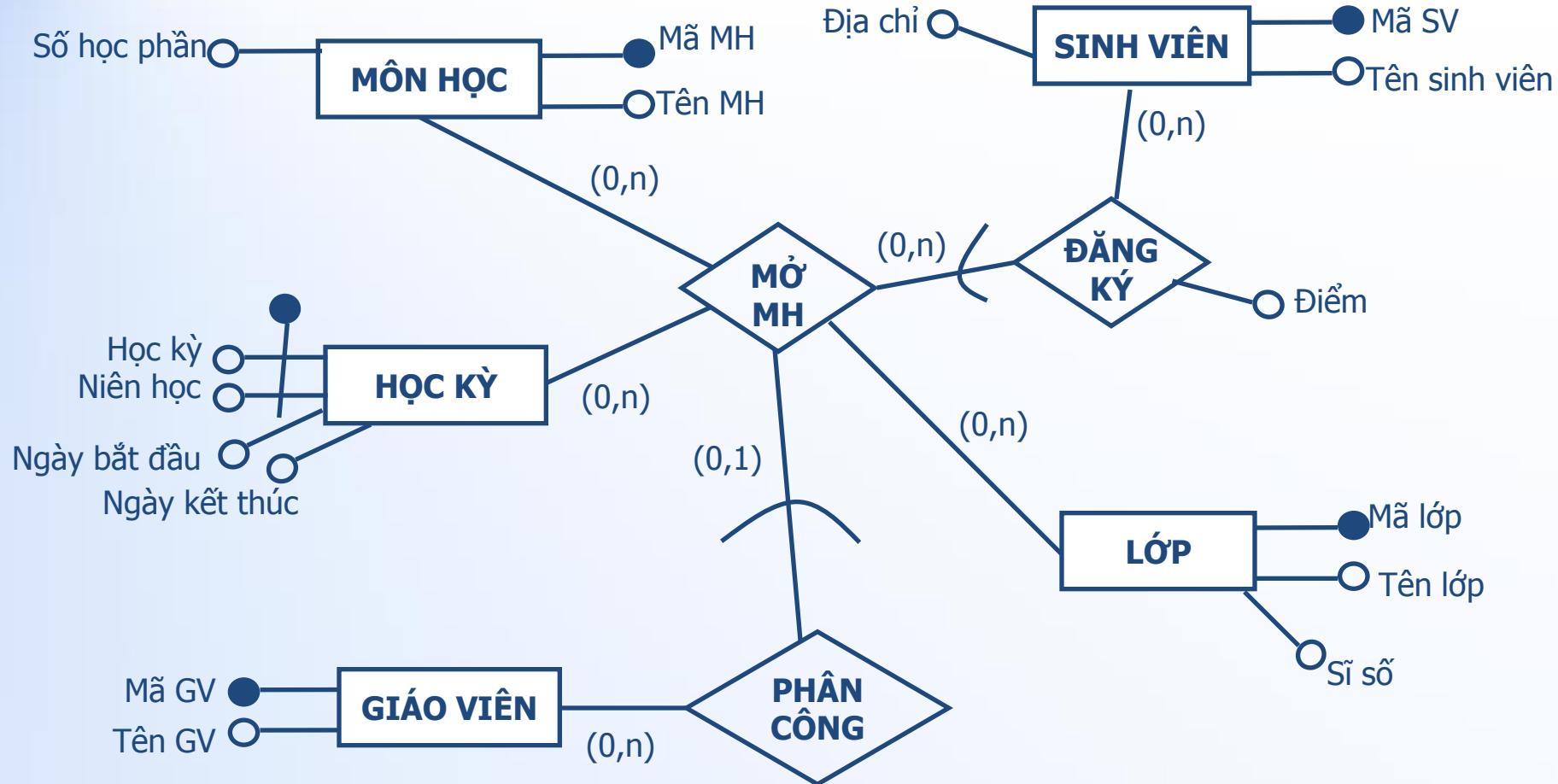
## ■ Mối kết hợp mở rộng



**R1(A,B, ...)**

**R2(A,B,C, ...)**

# Ví dụ



# Ví dụ (tt)

**MÔN\_HỌC**(MA\_MH, TÊN\_MH, SỐ\_HP)

**HỌC\_KỲ**(HOC\_KỲ, NIÊN\_HOC, NGÀY\_BĐ, NGÀY\_KT)

**GIÁO\_VIÊN**(MÃ\_GV, TÊN\_GV)

**SINH\_VIÊN**(MÃ\_SV, TÊN\_SV)

**LỚP**(MÃ\_LỚP, TÊN\_LỚP, SĨ\_SỐ)

**MỞ\_MH**(MÃ\_MH, MÃ\_LỚP, HOC\_KỲ, NIÊN\_HOC)

**ĐĂNG\_KÝ**(MÃ\_MH, MÃ\_LỚP, HOC\_KỲ, NIÊN\_HOC, MÃ\_SV, ĐIỂM)

**PHÂN\_CÔNG**(MÃ\_MH, MÃ\_LỚP, HOC\_KỲ, NIÊN\_HOC, MÃ\_GV)

Khóa chính  
quá phức tạp

# Nội dung chi tiết

- Thiết kế luận lý dữ liệu
- Thiết kế mã

- Tìm các hình thức mã hóa
  - ◆ Mã quận, mã bưu điện
  - ◆ Mã xe
  - ◆ Mã sinh viên
  - ◆ ...
- Dễ dàng cho việc
  - ◆ Quản lý và xử lý thông tin trong máy tính
  - ◆ Nhận dạng và phân loại dữ liệu
- Phải được thiết kế trên cơ sở uyển chuyển và có khả năng mở rộng

## ■ Ngữ nghĩa của mã hóa

- ◆ Duy nhất
  - Mã phải duy nhất để nhận dạng đối tượng dữ liệu
- ◆ Phân loại
  - Phân loại dữ liệu thành các nhóm khác nhau, dựa trên mã phân biệt được nhóm
- ◆ Sắp xếp
  - Mã hóa thể hiện thứ tự của dữ liệu
- ◆ Kiểm tra
  - Kiểm soát dữ liệu có được nhập đúng hay không

## ■ Một số đặc điểm

- ◆ Vùng mã hóa: phạm vi mà mã được sử dụng
  - Có liên kết với các hệ thống bên ngoài, mã nên là chuẩn công nghiệp chung
  - Trong cùng 1 đơn vị, mã có phạm vi toàn công ty
- ◆ Chu kỳ sử dụng
  - Ước lượng 1 lượng mã cần đủ trong chu kỳ
- ◆ Ngữ nghĩa
  - Có thể hiểu mã bởi các thành phần liên quan

# Ví dụ vùng mã hóa

- Mã môn học được thiết kế khác nhau trong cùng 1 trường

**TH201 - Thiết kế Cơ Sở Dữ Liệu**



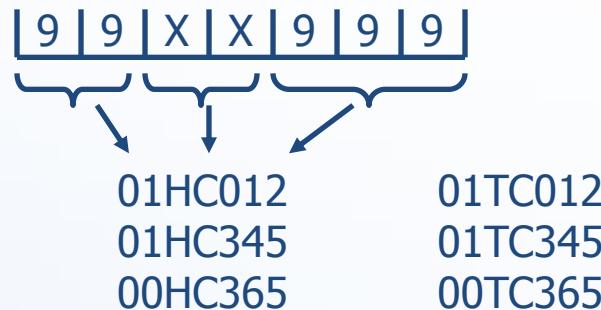
**CNTT201 - Thiết kế Cơ Sở Dữ Liệu**

Gây ra 1 số hoạt động không cần thiết như : chuyển đổi mã liên quan đến kết quả học tập trên môn học, ...

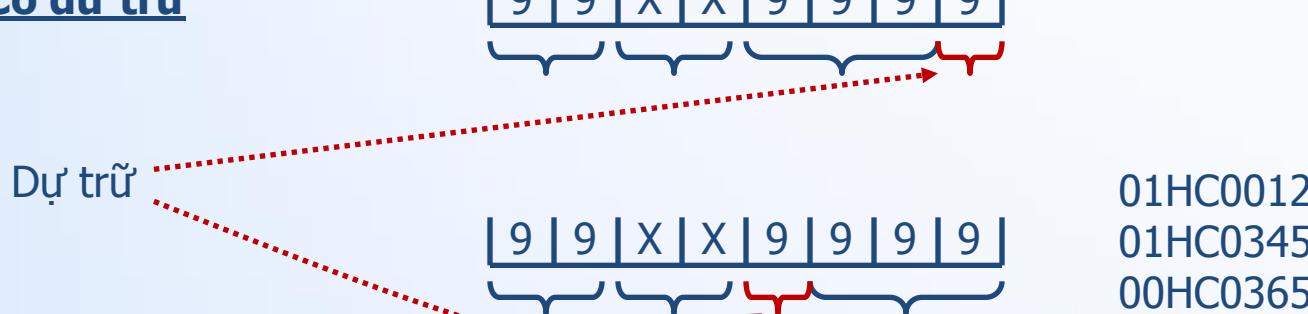
# Ví dụ chu kỳ sử dụng

## ■ Mã số sinh viên

### Không dữ trữ

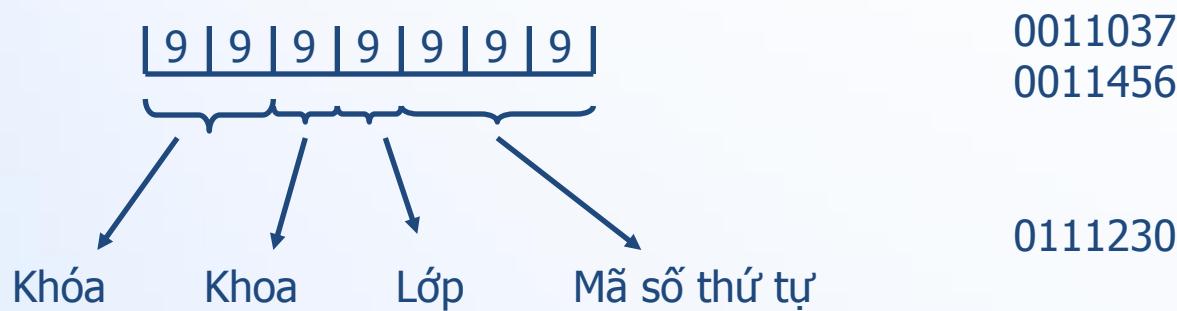


### Có dữ trữ



# Ví dụ ngữ nghĩa

## ■ Mã số sinh viên



# Một số loại mã

Tên mã	Mô tả	Ví dụ
Mã tuần tự	Một số được gán một cách tuần tự	<u>Mã số xe của tỉnh:</u> 50 Thành phố HCM 60 Đồng Nai 62 Long An 63 Tiền Giang
Mã khối	Một số được gán cho mỗi khối. Rồi một số sẽ được gán tuần tự trong khối đó → thuận tiện cho phân loại dữ liệu	<u>Mã khách hàng của từng chi nhánh được qui định như sau:</u> 0001 khách hàng tại trung tâm 1000 khách hàng chi nhánh A 2000 khách hàng chi nhánh B 3000 khách hàng chi nhánh C

# Một số loại mã (tt)

Tên mã	Mô tả	Ví dụ												
Mã thập phân	Các đối tượng sẽ được mã hóa từ 0 đến 9, rồi đến lượt thành viên của mỗi đối tượng này cũng sẽ được mã hóa từ 0 đến 9, ....	<u>Mã hóa phòng ban trong một công ty:</u> 00 Ban giám đốc 1 Bộ phận kinh doanh 1 Nhóm 1 2 Nhóm 2 2 Bộ phận phát triển 10 Chi nhánh A 20 Chi nhánh B 30 Chi nhánh C												
Mã theo ký số	Mỗi ký số của mã sẽ gán cho một ngữ nghĩa	<u>Mã SV của trường ĐHKHTN TPHCM</u> <table border="1"><tr><td>①</td><td>②</td><td>③</td><td>④</td></tr><tr><td>01</td><td>1</td><td>1</td><td>100</td></tr><tr><td>01</td><td>2</td><td>1</td><td>120</td></tr></table> ① Năm ② Khoa ③ Lớp ④ số thứ tự	①	②	③	④	01	1	1	100	01	2	1	120
①	②	③	④											
01	1	1	100											
01	2	1	120											

# Một số loại mã (tt)

Tên mã	Mô tả	Ví dụ
Mã gợi nhớ	Chữ viết tắt hoặc biểu tượng của đối tượng được dùng để mã hóa	<u>Mã hóa lớp học của trường đại học như sau:</u> 00TC Lớp tại chức khóa 2000 01TC Lớp tại chức khóa 2001 00HC Lớp hoàn chỉnh khóa 2000 01HC Lớp hoàn chỉnh khóa 2001
Mã kiểm tra	Dùng một ký số kiểm tra thêm vào sau mỗi mã số	

# Thiết kế dữ liệu

- Thiết kế luận lý dữ liệu
  - ◆ Thiết kế luận lý cấp cao
  - ◆ Thiết kế luận lý cấp thấp
- Thiết kế mã
- Thiết kế vật lý dữ liệu

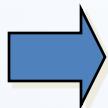
# Thiết kế dữ liệu vật lý

## ■ Mục đích:

- ◆ Chuyển các mô tả dữ liệu luận lý sang các đặc tả kỹ thuật nhằm lưu trữ và truy xuất dữ liệu
- ◆ Tạo một thiết kế cho việc lưu trữ dữ liệu nhằm cung cấp một hiệu năng phù hợp và đảm bảo tính toàn vẹn, an toàn và khả năng phục hồi của CSDL

# Thiết kế dữ liệu vật lý

- Mô hình quan hệ
- Khối lượng dữ liệu
- Định nghĩa của thuộc tính
- Yêu cầu về thời gian trả lời
- Nhu cầu về an toàn dữ liệu
- Nhu cầu về backup/resotore
- Hệ quản trị CSDL sử dụng



- Kiểu dữ liệu thuộc tính
- Mô tả mẫu tin vật lý
  - Tổ chức file
- Kiến trúc chỉ mục và CSDL
  - Tối ưu hóa truy vấn

# Thiết kế dữ liệu vật lý

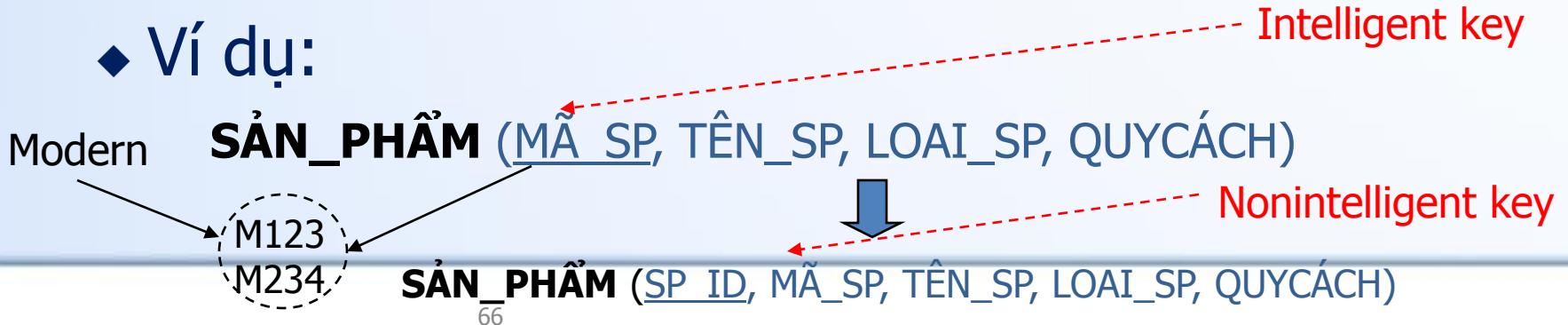
- Thiết kế field
- Phân chia dữ liệu (partition)
- Gộp dữ liệu (demormalization)
- Thiết kế file dữ liệu vật lý
- Tổ chức file chỉ mục
- Clustering file

# Thiết kế dữ liệu vật lý

- Thiết kế field: đơn vị dữ liệu nhỏ nhất
- Thiết kế field bao gồm:
  - ◆ Chọn kiểu dữ liệu: phải thỏa các tiêu chuẩn sau:
    - Tối thiểu không gian lưu trữ
    - Hiển thị tất cả tình huống giá trị
    - Cải tiến việc toàn vẹn dữ liệu
    - Hỗ trợ cho tất cả thao tác dữ liệu

# Thiết kế dữ liệu vật lý

- Thiết kế field – Chọn lựa khóa chính:
  - ◆ Việc chọn lựa khóa chính trong mô hình luân lý đôi khi không thuận tiện trong cập nhật
  - ◆ Thêm một thuộc tính khóa không có ý nghĩa thực tế (nonintelligent key) → hỗ trợ cho các phép toán cập nhật tốt hơn thuộc tính khóa trước đó
  - ◆ Ví dụ:



# Thiết kế dữ liệu vật lý

## ■ Thiết kế field bao gồm:

### ◆ Ví dụ: **SANPHAM**

SP_ID	Mã_SP	Tên_SP	Loại_SP	Quycách
1	M123	Sản phẩm X	M	12x4
2	M234	Sản phẩm Y	M	15x3
3	S012	Sản phẩm M	S	12x2
4	L121	Sản phẩm N	L	18x6

MÃ\_SP của table SANPHAM có thể bị thay đổi mà không ảnh hưởng đến table HÓAĐƠN có khóa ngoại tham chiếu đến SANPHAM

### **HÓAĐƠN**

HD_ID	Số_HD	Ngày_HD	SP_ID	SLượng	Đơn giá
1	001/HD	1/1/2004	1	10	200
2	002/HD	1/1/2004	1	5	200
3	003/HD	2/1/2004	2	120	120
4	004/HD	3/1/2004	4	200	700

# Thiết kế dữ liệu vật lý

- Thiết kế field – Chọn lựa khoá chính:
  - ◆ Khoá chính phức tạp sẽ làm giảm tốc độ truy cập CSDL trong máy tính → chọn lựa khoá chính thay thế (nonintelligent key) đơn giản và hiệu quả hơn
  - ◆ Ví dụ:

Khoá chính luận lý (intelligent key)

ĐĂNG\_KÝ(MÃ\_MH, MÃ\_LỚP, HỌC\_KỲ, NIÊN\_HỌC, MÃ\_SV, ĐIỂM)

Khoá chính mới (nonintelligent key)

ĐĂNG\_KÝ(DKY\_ID, MÃ\_MH, MÃ\_LỚP, HỌC\_KỲ, NIÊN\_HỌC, MÃ\_SV, ĐIỂM)

Tổ hợp các thuộc tính này được cài đặt là một khoá

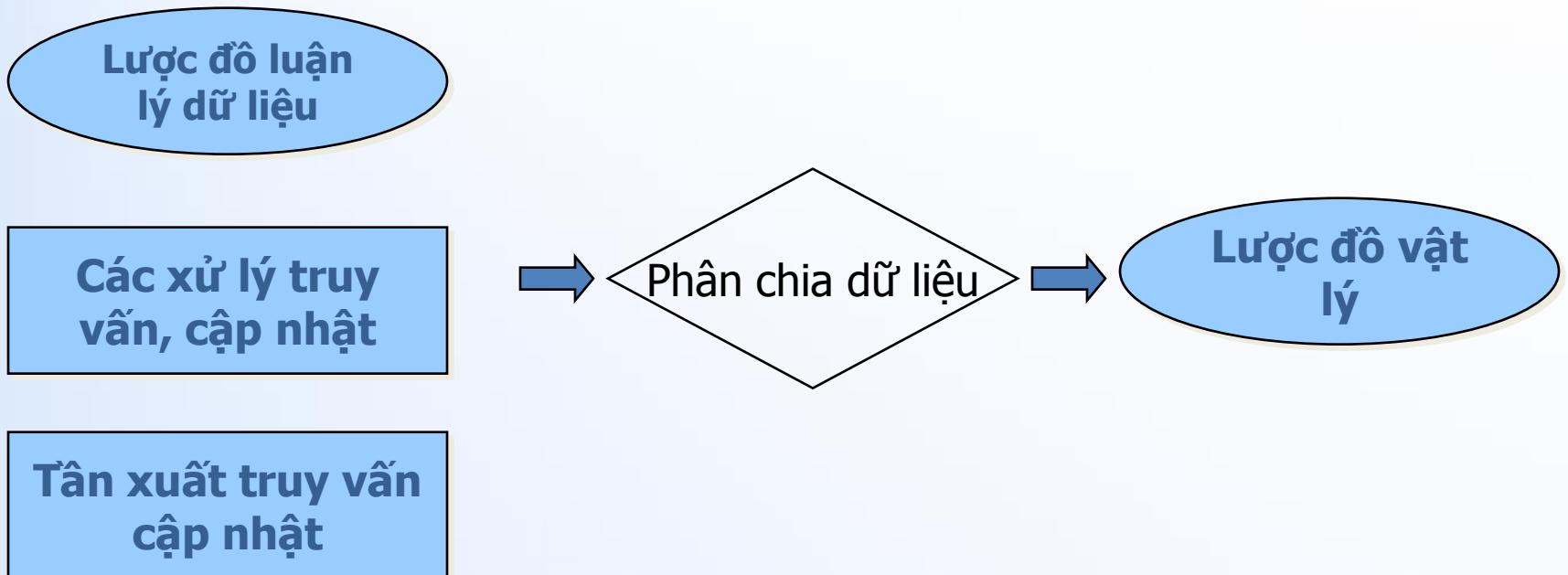
# Thiết kế dữ liệu vật lý

## ■ Phân chia dữ liệu (partition)

- ◆ Phân chia theo chiều ngang (horizontal partition): phân chia các dòng trong một table thành nhiều table khác nhau
- ◆ Tình huống áp dụng: khi nhiều người dùng khác nhau cần truy cập các dòng dữ liệu khác nhau
- ◆ Ưu điểm:
  - Tối ưu hóa tốc độ truy cập dữ liệu
- ◆ Nhược điểm
  - Phức tạp khi phải truy cập toàn bộ dữ liệu

# Thiết kế dữ liệu vật lý

## ■ Phân chia dữ liệu (partition)



# Thiết kế dữ liệu vật lý

## ■ Phân chia dữ liệu (partition)

- ◆ Ví dụ:

**HOA\_DON** **KL: ~10.000.000/năm**

Số_HD	Ngày_HD	Diễn_giải	Trị giá
Hd00001	1/1/04	Xxxxxx	1.000.000
Hd00002	2/1/04	Yyyyyyy	2.000.000
....			
Hd15000	1/1/05	Zxxxxxxxxx	1.400.000
Hd15001	2/1/05	Qqqqqqqqqq	2.100.000
...			
Hd30000	2/1/06	Asasasas	12.000.000
Hd30001	2/1/06	Dsdsdsds	1.000.000

## Các xử lý truy cập dữ liệu

Mã số	Tên xử lý	Tần suất
O1	Tìm hóa đơn	100/ngày
O2	Tính doanh thu tháng	1/tháng
O3	Tính doanh thu theo khách hàng	100/tháng
O4	Tổng hợp doanh số năm	1/năm
O5	Lập biểu đồ so sánh doanh số theo các năm	1/năm

# Thiết kế dữ liệu vật lý

- Phân chia dữ liệu  
(partition)
  - Ví dụ:

**HOA\_DON**      **KL: ~10.000.000/năm**

Số_HD	Ngày_HD	Diễn_giải	Trị giá
Hd00001	1/1/04	Xxxxxxx	1.000.000
Hd00002	2/1/04	Yyyyyyy	2.000.000
....			
Hd15000	1/1/05	Zxzxzzxzx	1.400.000
Hd15001	2/1/05	Qqqqqqqqq	2.100.000
...			
Hd30000	2/1/06	Asasasas	12.000.000
Hd30001	2/1/06	Dsdssdsds	1.000.000

Tách table HOA\_DON theo năm  
Các phép toán o1, o2, o4 hiệu quả hơn

**HD004**

Số_HD	Ngày_HD	Diễn_giải	Trị giá
Hd00001	1/1/04	Xxxxxxx	1.000.000
Hd00002	2/1/04	Yyyyyyy	2.000.000

**HD005**

Số_HD	Ngày_HD	Diễn_giải	Trị giá
Hd15000	1/1/05	Zxzxzzxzx	1.400.000
Hd15001	2/1/05	Qqqqqqqqq	2.100.000

**HD006**

Số_HD	Ngày_HD	Diễn_giải	Trị giá
Hd30000	2/1/06	Asasasas	12.000.000
Hd30001	2/1/06	Dsdssdsds	1.000.000

# Thiết kế dữ liệu vật lý

## ■ Phân chia dữ liệu (partition)

### ◆ Phân chia theo chiều dọc (vertical partition):

- Phân chia một cấu trúc luận lý thành những cấu trúc lưu trữ vật lý khác nhau
- Ví dụ:

**KHÁCH\_HANG**(MÃ\_KH, TÊN\_KH, DCHI, DT, TAIKHOAN, MASOTHUE,  
PTTTOAN, MUC\_NO, CONG\_NO)

**KH1**(MA\_KH, TEN\_KH, DCHI)

**KH2**(MA\_KH, DT, TAIKHOAN, MASOTHUE  
PTTTOAN, MUC\_NO, CONG\_NO)

Cấu trúc truy cập thường xuyên

Cấu trúc truy cập không thường xuyên

# Thiết kế dữ liệu vật lý

## ■ Gộp dữ liệu (denormalization)

- ◆ Mục tiêu:

- Tối ưu hóa truy vấn dữ liệu

- ◆ Hạn chế:

- Phát sinh trùng lắp dữ liệu

- Kiểm soát tính nhất quán dữ liệu

# Thiết kế dữ liệu vật lý

## ■ Gộp dữ liệu (denormalization)

### ◆ Gộp 2 quan hệ liên kết 1-1

**SINH\_VIÊN(MÃ\_SV, TEN\_SV, CHUYEN\_NGANH)**

**HỒSƠ\_HBỐNG(MÃ\_HS, NGÀY\_HS, KHẢ\_NĂNG,MA\_SV)**

Xử lý	Dữ liệu liên quan
O1	MÃ_SV, TEN_SV, CHUYEN_NGANH, NGÀY_SINH, KHẢ_NĂNG
O2	TEN_SV, NGÀY_HS, KHẢ_NĂNG



**SINH\_VIÊN(MÃ\_SV, TEN\_SV, CHUYEN\_NGANH, NGÀY\_HS, KHẢ\_NĂNG)**

# Thiết kế dữ liệu vật lý

- Gộp dữ liệu (denormalization)
  - ◆ Gộp 2 quan hệ liên kết 1-N

Mã_SV	Tên_SV	...	MA_CN
991100	N.V.A	...	HTTT
991101	N.T.B	...	CNPM
991112	P.V.C	...	CNPM
991120	T.T.N	...	HTTT
991200	T.V.M	...	CNPM

Mã_CN	Tên_CN
HTTT	Hệ thống thông tin
CNPM	Công nghệ phần mềm
MMT	Mạng máy tính
CNTT	Công nghệ tri thức

Truy vấn thường xuyên:  
- Q1 (Mã\_SV, TÊN\_SV, TÊN\_CN)

# Thiết kế dữ liệu vật lý

- Gộp dữ liệu (denormalization)
  - ◆ Gộp 2 quan hệ liên kết 1-N

Mã_SV	Tên_SV	...	MA_CN	Tên_CN
991100	N.V. A	...	HTTT	Hệ thống thông tin
991101	N.T.B	...	CNPM	Công nghệ phần mềm
991112	P.V.C	...	CNPM	Công nghệ phần mềm
991120	T.T.N	...	HTTT	Hệ thống thông tin
991200	T.V.M	...	CNPM	Công nghệ phần mềm

Truy vấn thường xuyên:

- Q1 (Mã\_SV, TÊN\_SV, TÊN\_CN)

Trùng lắp thông tin

**Cấu trúc gộp trên sẽ tối ưu hơn cho truy vấn Q1, nhưng sẽ dẫn đến trùng lắp thông tin**

