
Chương 3:

Sơ Đồ Lớp

Mục tiêu

- Tìm hiểu về sơ đồ lớp
- Cách phân tích bài toán để xây dựng sơ đồ lớp

Sơ đồ lớp là gì?

Sơ đồ lớp là một mô hình tĩnh để biểu diễn các lớp đối tượng và mối quan hệ giữa chúng, nó không thay đổi trong hệ thống theo thời gian. Sơ đồ lớp miêu tả các lớp đối tượng, bao gồm cả hành vi và trạng thái, cùng với mối quan hệ giữa chúng.

Các thành phần chủ yếu

- Lớp đối tượng (class)
- Thuộc tính (attributes)
- phương thức (operations)
- Mỗi liên hệ (relationships) giữa các lớp
- Bản số (cardinality)
- Các ràng buộc (constraint) và ghi chú (note)

Lớp (class)

- Là một tập hợp các đối tượng chia sẻ chung một cấu trúc và hành vi (cùng thuộc tính, hoạt động, mối quan hệ và ngữ nghĩa).
- Cấu trúc được mô tả bởi các thuộc tính và các mối quan hệ, còn hành vi được mô tả bởi các hoạt động
- Một lớp là sự trừu tượng của các đối tượng trong thế giới thực, và các đối tượng tồn tại trong thế giới thực là các thể hiện của lớp.

Lớp (class)

- Ký hiệu: Một lớp được mô tả gồm 3 thành phần: Tên, các thuộc tính và các phương thức.
- Ví dụ: Lớp Hóa_đơn
- Lưu ý: Phần thuộc tính và phương thức có thể giấu đi ở mức độ trình bày tổng quan.

Tên
Các thuộc tính
Các phương thức

Hóa_đơn
Số
Ngày lập
Tính_trị_giá()

Thuộc tính

- Là các giá trị dữ liệu có trong các đối tượng của cùng một lớp.
- Ví dụ:
 - *Số, ngày* là các thuộc tính của lớp *Hóa đơn*.
 - *Tên, tuổi, năm sinh, quê quán* là các thuộc tính của lớp *Người*.
 - *Nhãn hiệu, màu sắc, năm sản xuất* là các thuộc tính của lớp *Xe*.

Thuộc tính

- Thuộc tính thường có kiểu dữ liệu nguyên thủy: String, Integer, Real, Double, date/time,...
- Các thuộc tính dẫn xuất được đặt sau /
- Cú pháp: *[visibility] name [:type] [=defaultValue]*
- Ví dụ:

SINH VIÊN
<ul style="list-style-type: none">- MSSV: String- Ten: String- Nsinh: Int = 1984- /tuoi

Thao tác/Hành vi/Phương thức (method)

- Là chức năng hoặc hành vi của một lớp
- phương thức có thể là một hàm hay một phương thức
- Ví dụ:
 - *Mở, đóng, phóng to, thu nhỏ* là các phương thức của lớp *Cửa sổ*
 - *Tô màu, tính diện tích, tính chu vi* là các phương thức của lớp *Hình vẽ*
- Tất cả các đối tượng thuộc cùng một lớp đều có các phương thức giống nhau
- Cùng một phương thức có thể áp dụng trên nhiều lớp dữ liệu khác nhau (Polymorphism)

Thao tác/Hành vi/Phương thức (method)

- Các phương thức được mô tả trong ký hiệu lớp

- Cú pháp:

[visibility] name ([parameter-list]) [:return-type]

- Trong đó *parameter-list* là một danh sách các tham số hình thức cách nhau bởi dấu phẩy, mỗi tham số có cú pháp là: *name:type [= defaultValue]*

- Ví dụ:

- Tính chu vi: + Chu_vi(): Float = 0
- Vẽ hình: -Ve()
- Cộng 2 số: # Cong()

Tầm vực

- Phạm vi mà một thuộc tính/phương thức có thể thấy và tham chiếu được từ các lớp bên ngoài lớp đã định nghĩa chúng
- Tầm vực được định nghĩa khác nhau ở các NNLT khác nhau
- Ký hiệu theo UML:
 - + : public
 - : private
 - # : protected

DiaChi
- SoNha: String
- Duong: String
- ThanhPho: String
+ getSoNha(): String
+ getDuong(): String
+ getThanhPho():String

Tầm vực (tt)

- **Public (+):** Thuộc tính/phương thức có thể tham chiếu bởi bất kỳ phương thức nào trong bất kỳ đối tượng hoặc lớp khác
- **Protected (#):** Thuộc tính/phương thức có thể tham chiếu bởi bất kỳ phương thức nào trong cùng lớp hoặc trong lớp con của lớp đó
- **Private (-):** Thuộc tính/phương thức có thể tham chiếu bởi bất kỳ phương thức nào trong cùng lớp

Quan hệ giữa các lớp

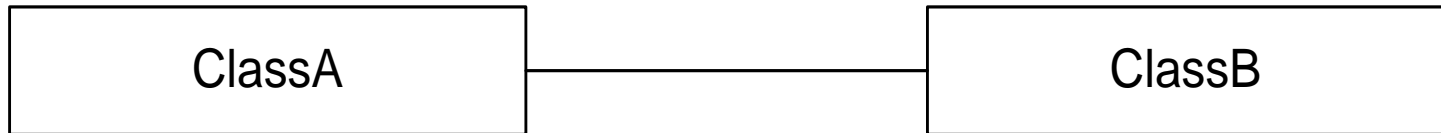
- Là quan hệ ngữ nghĩa được thiết lập giữa 1 lớp (phản thân), **2 lớp (nhị phân)** hay nhiều lớp (đa phân).
- Quan hệ được biểu diễn bởi các thành phần:
 - Tên: Thường là **cụm động từ** phản ánh mục đích của quan hệ.
 - Vai trò: Mô tả ngữ nghĩa tham gia vào một lớp của quan hệ đó
 - Bản số: Số các thể hiện trong một lớp có thể gắn với số thể hiện của lớp kết hợp với nó.

Quan hệ giữa các lớp

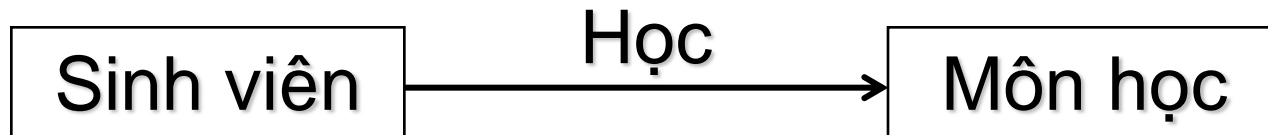
- Là kết nối ngữ nghĩa giữa các lớp.
- Thông qua các quan hệ, một lớp có thể truy xuất đến các thuộc tính và phương thức của một lớp khác
- Các loại quan hệ
 - Kết hợp - Associations
 - Kế thừa - Inheritance
 - Kết tập - Aggregation
 - Hợp thành - Composition
 - Phụ thuộc - Dependency

Quan hệ kết hợp

- Hoặc trong ClassA có thuộc tính có kiểu là ClassB
- Hoặc trong ClassB có thuộc tính có kiểu là ClassA

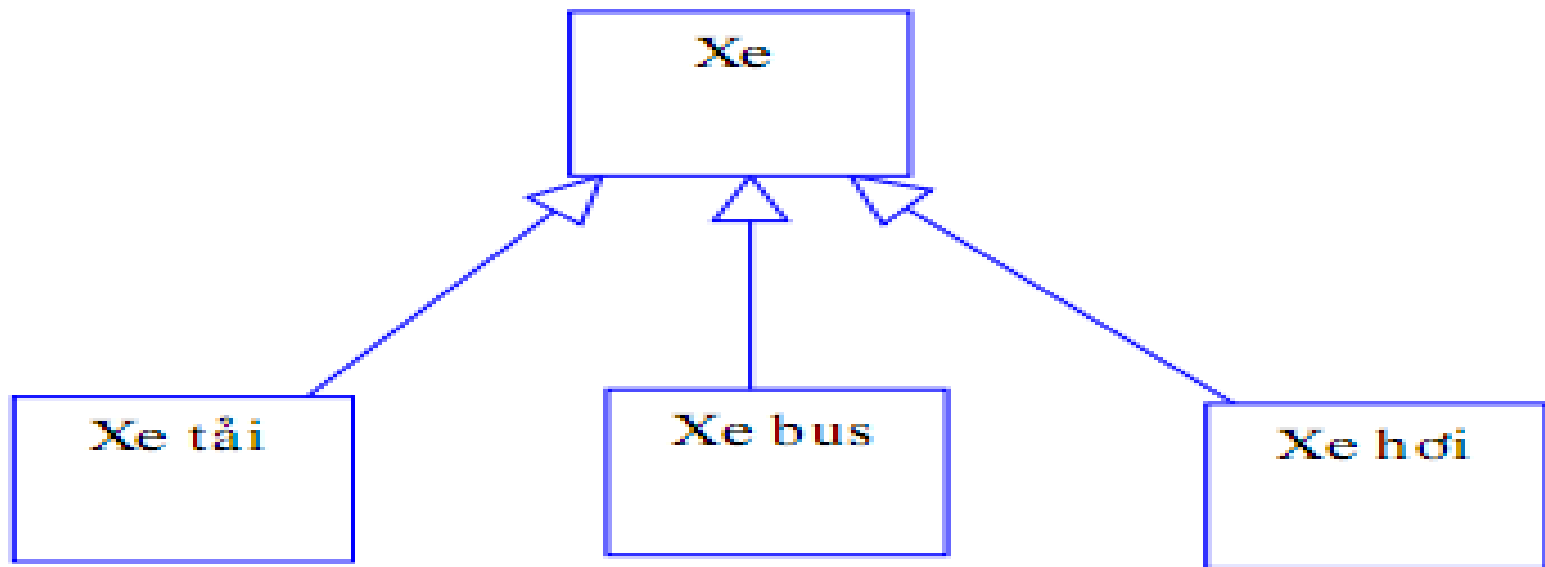


- Ví dụ:



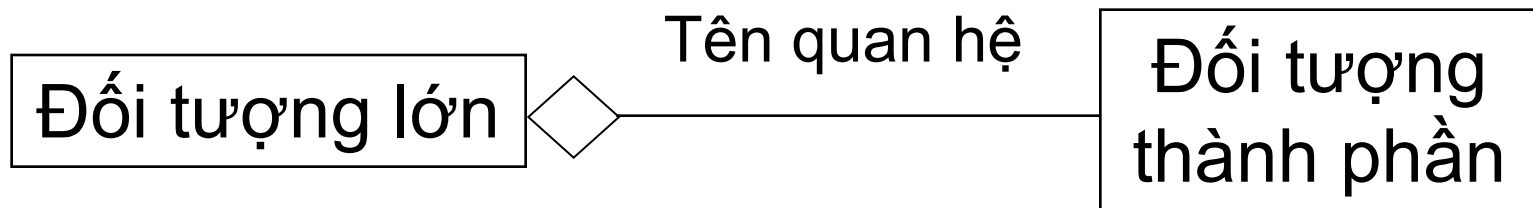
Quan hệ kế thừa

- Được thiết lập giữa một lớp tổng quát hơn đến một lớp chuyên biệt
- Thường dùng để phân loại một tập hợp đối tượng thành các đối tượng cụ thể hơn mà hệ thống cần làm rõ ngữ nghĩa.
- Ví dụ:



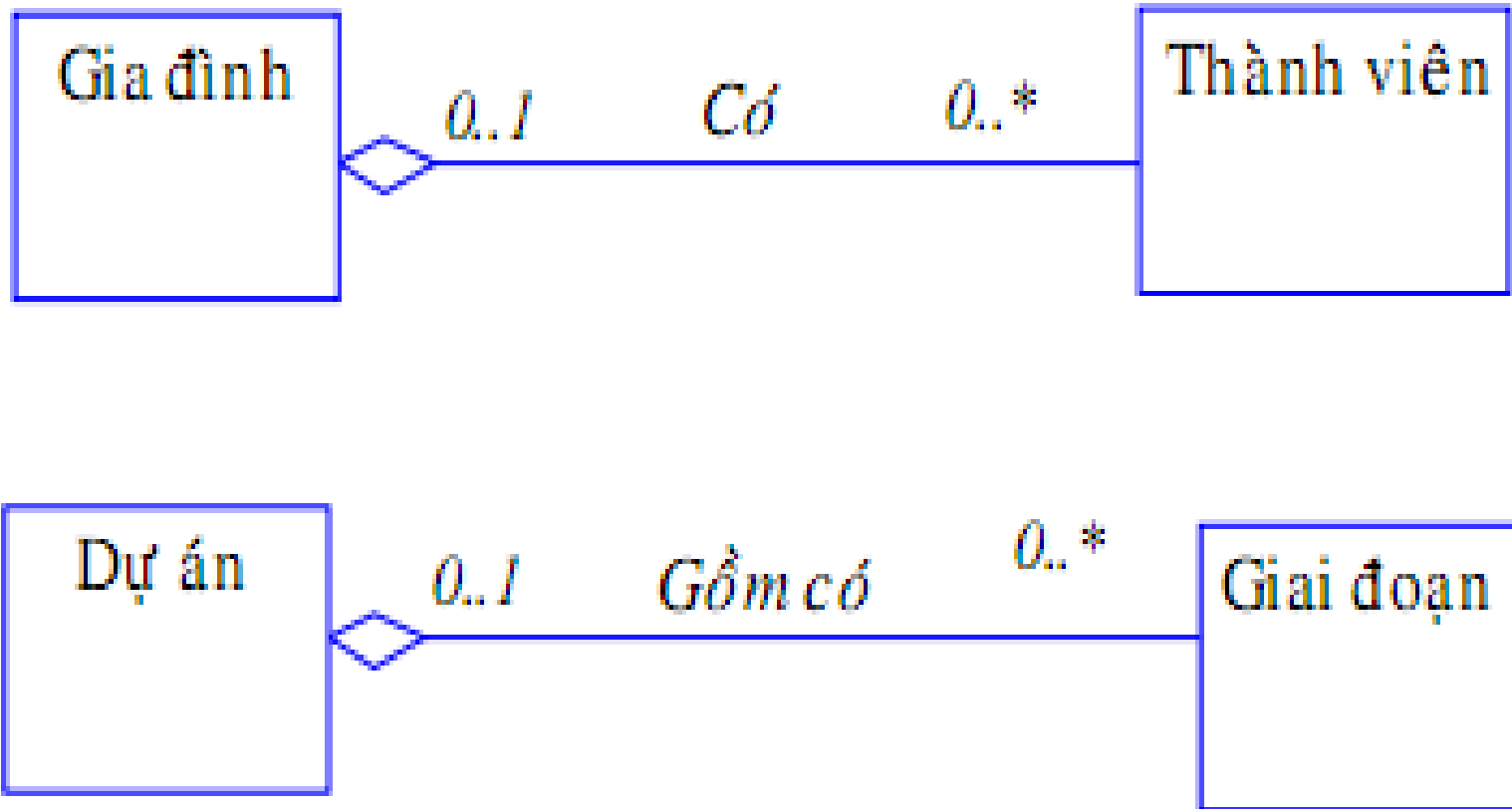
Quan hệ kết tập (Aggregation)

- Mô tả mối quan hệ giữa một đối tượng lớn hơn được tạo ra từ những đối tượng nhỏ hơn.
- Một quan hệ đặc biệt của loại quan hệ này là quan hệ có, tức là một đối tượng tổng thể có những đối tượng thành phần.
- Ký hiệu:



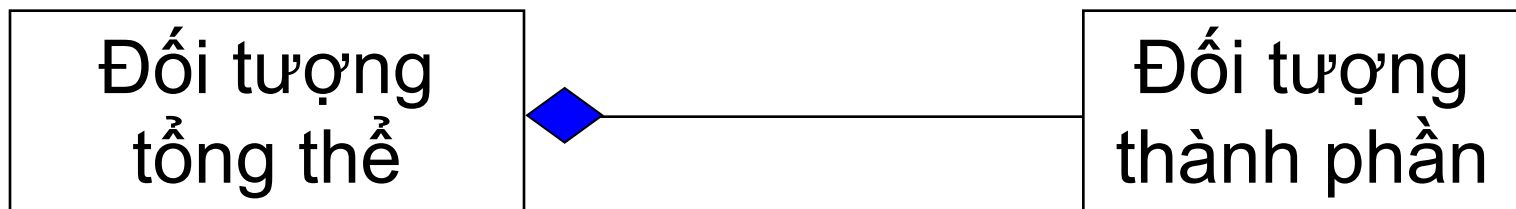
Quan hệ kết tập (Aggregation)

■ Ví dụ:



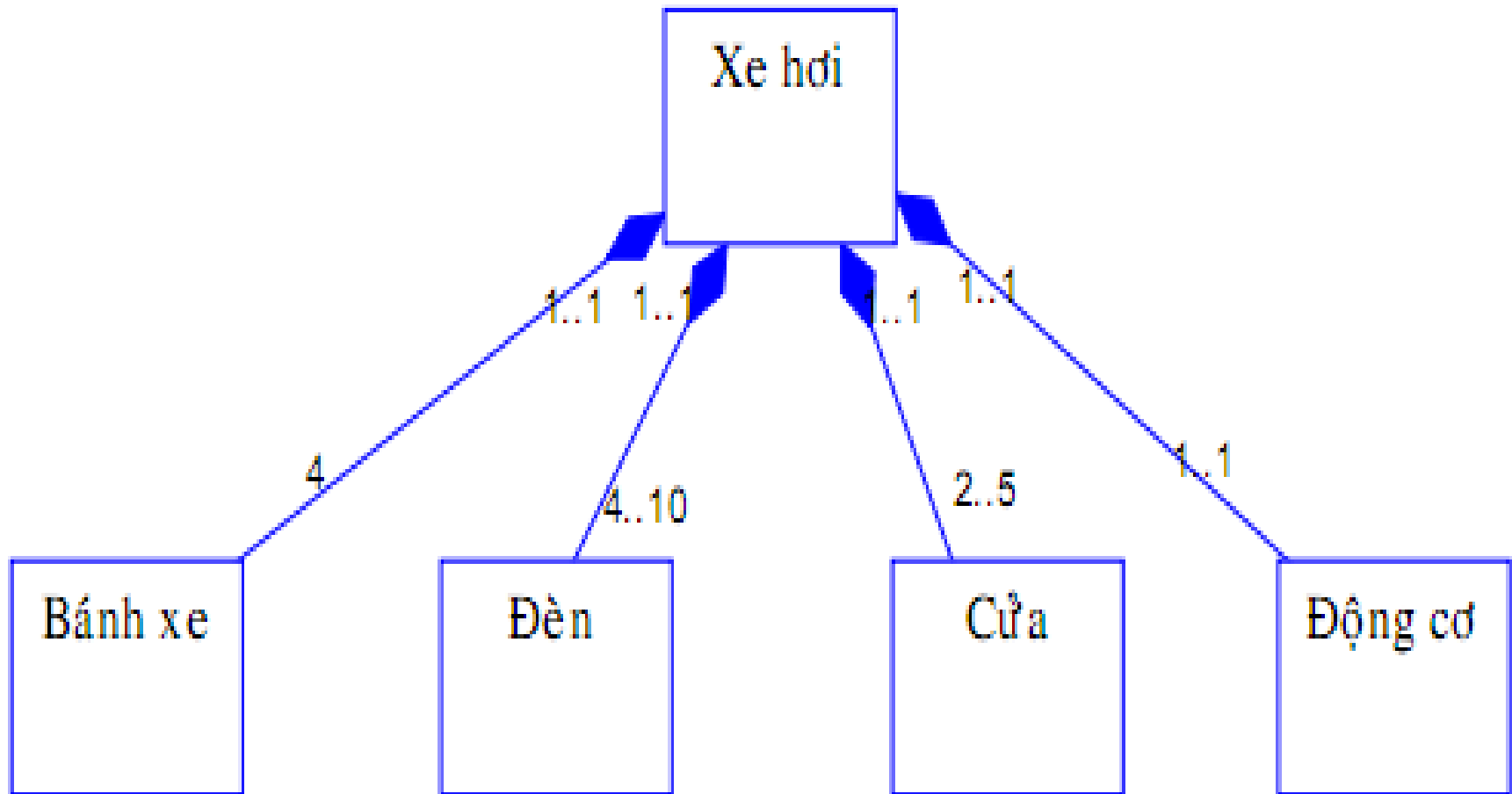
Quan hệ hợp thành (composition)

- Là một dạng đặc biệt của quan hệ kết tập, nó có liên hệ mạnh hơn để trình bày thành phần của một đối tượng phức hợp.
- Còn được xem là quan hệ thành phần- tổng thể, đối tượng tổng hợp sẽ quản lý việc tạo và hủy các đối tượng thành phần của nó.
- Ký hiệu:



Quan hệ hợp thành (composition)

- Ví dụ:

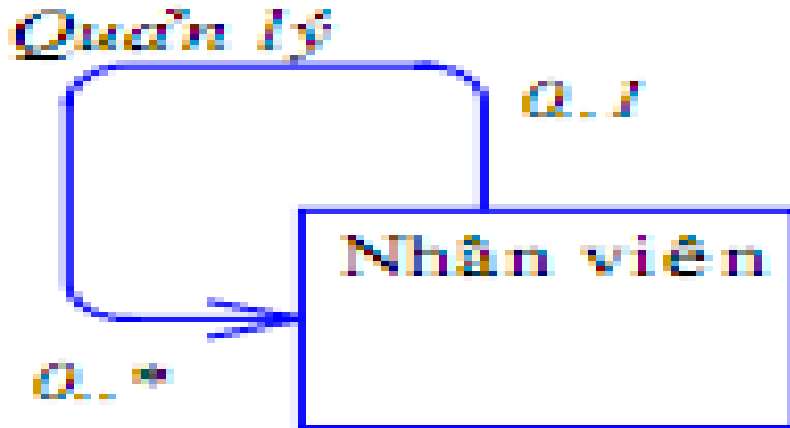


Bản số (cardinality)

- Biểu diễn số thể hiện tối thiểu và tối đa ở hai đầu của quan hệ.
- Bản số xuất hiện trong các quan hệ kết hợp, kết tập và hợp thành.
- Bản số nên được xác định sau khi đã xác định các đối tượng, các lớp và các mối quan hệ giữa các lớp
- Các loại bản số: 1 (chỉ một); * (không hoặc nhiều); 1..* (một hoặc nhiều); 0..1 (không hoặc một); 2..4 (thuộc phạm vi); 2,4..6,8 (thuộc phạm vi không liên tục).

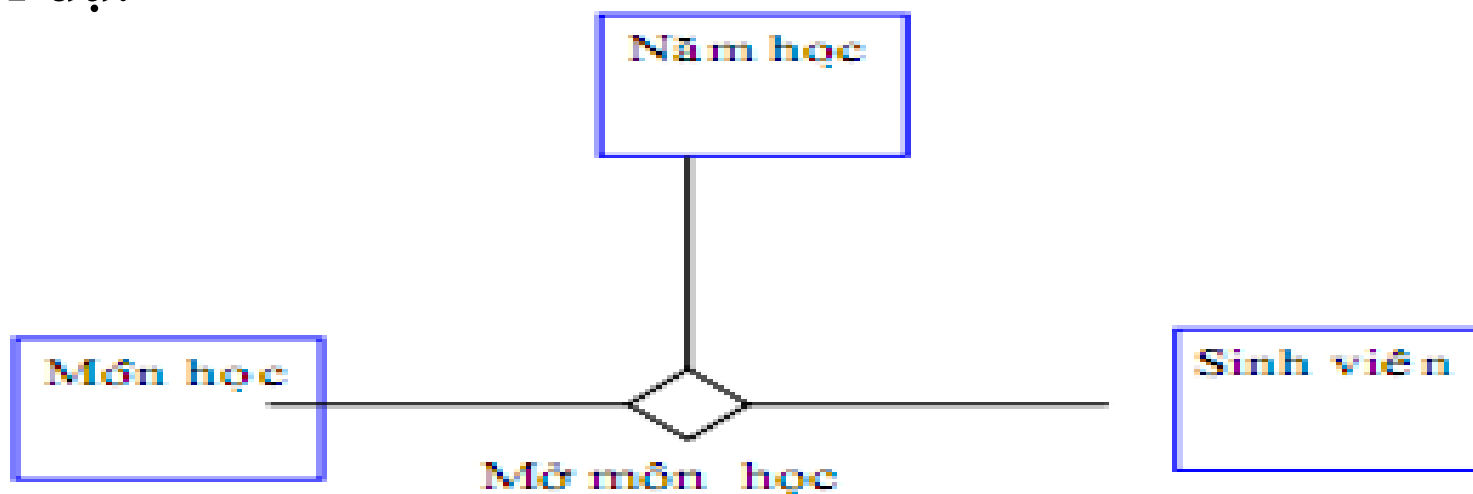
Quan hệ phản thân

- Là quan hệ giữa một lớp với chính nó.
- Ví dụ: Quan hệ **Quản lý** được thiết lập giữa lớp **Nhân viên** tới chính nó có ý nghĩa là một nhân viên có thể quản lý những nhân viên khác.



Quan hệ đa phân

- Là quan hệ được thiết lập từ 3 lớp trở lên.
- Ký hiệu: Dùng một hình thoi với ít nhất 3 kết nối đến các lớp tham gia vào quan hệ.
- Ví dụ:



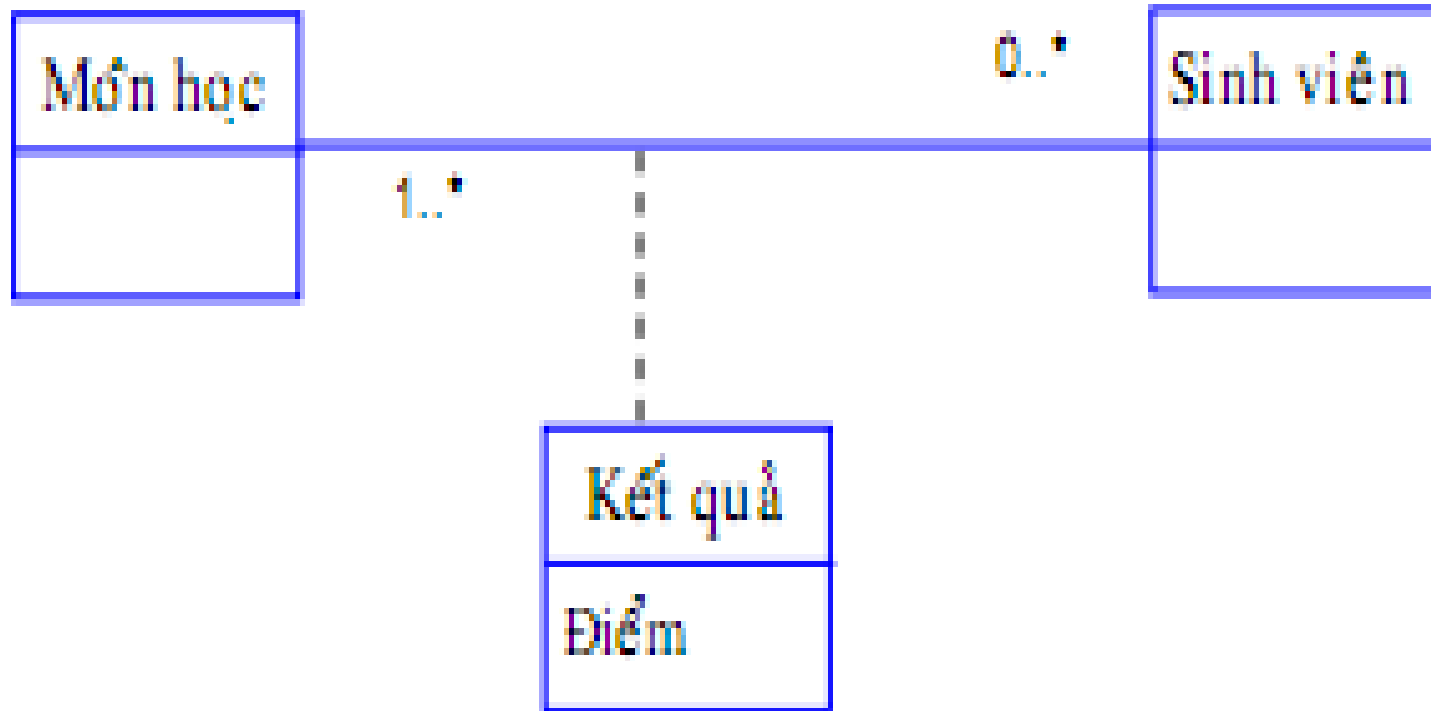
- Ít được sử dụng và thường được gom lại thành một lớp.

Lớp quan hệ

- Khi một quan hệ có các đặc trưng riêng (thuộc tính và phương thức) thì quan hệ đó được tách thành một lớp để chứa các thuộc tính và phương thức đó.
- Lớp này được gọi là lớp quan hệ và được gắn vào quan hệ đang có.
- Tên của lớp này chính là tên của quan hệ.
- Trường hợp phổ biến nhất của lớp quan hệ là quan hệ giữa 2 lớp có quan hệ nhiều-nhiều.

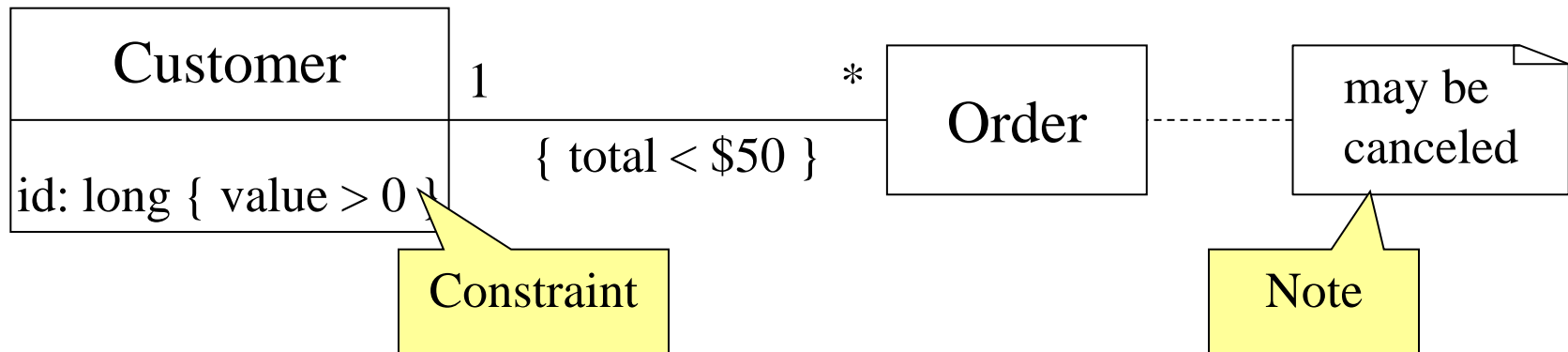
Lớp quan hệ

- Ví dụ:

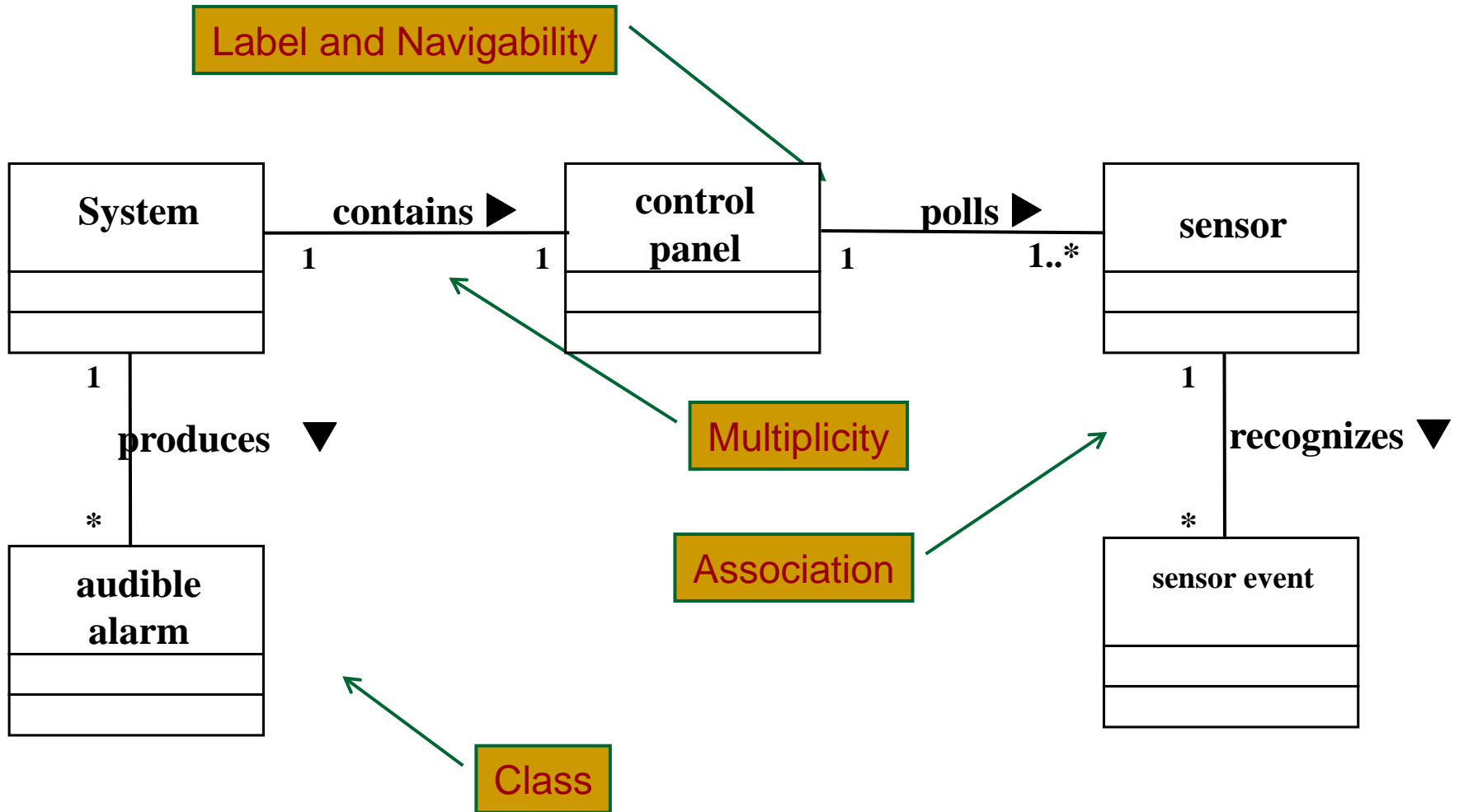


Các ràng buộc (constraint) và ghi chú (note)

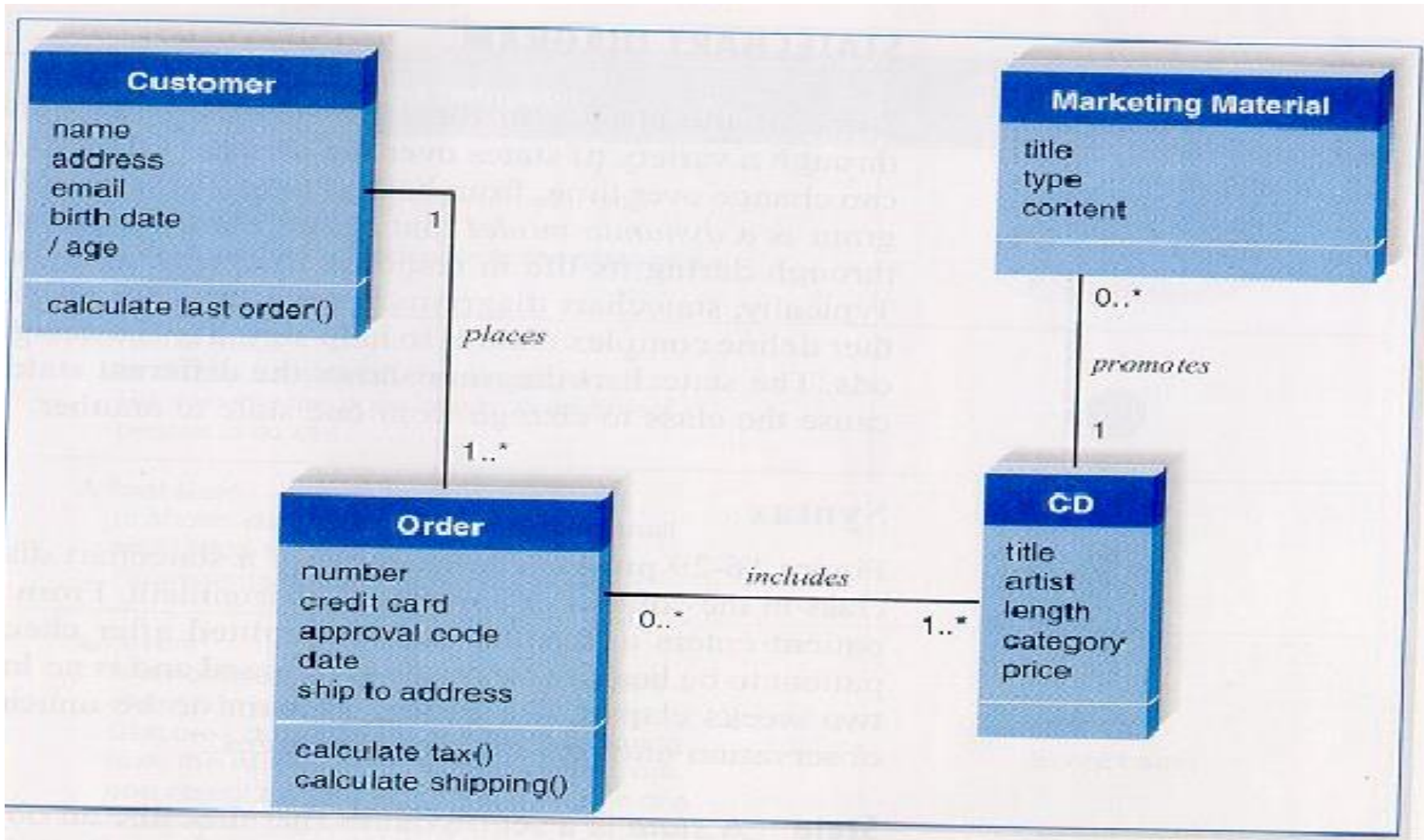
- Các ràng buộc và ghi chú dùng để chú thích các quan hệ, các thuộc tính, các phương thức và các lớp.
- Ràng buộc là các giới hạn ngữ nghĩa được ghi nhận lại.
 - UML hỗ trợ nhiều ràng buộc được định nghĩa sẵn.
- Ví dụ:



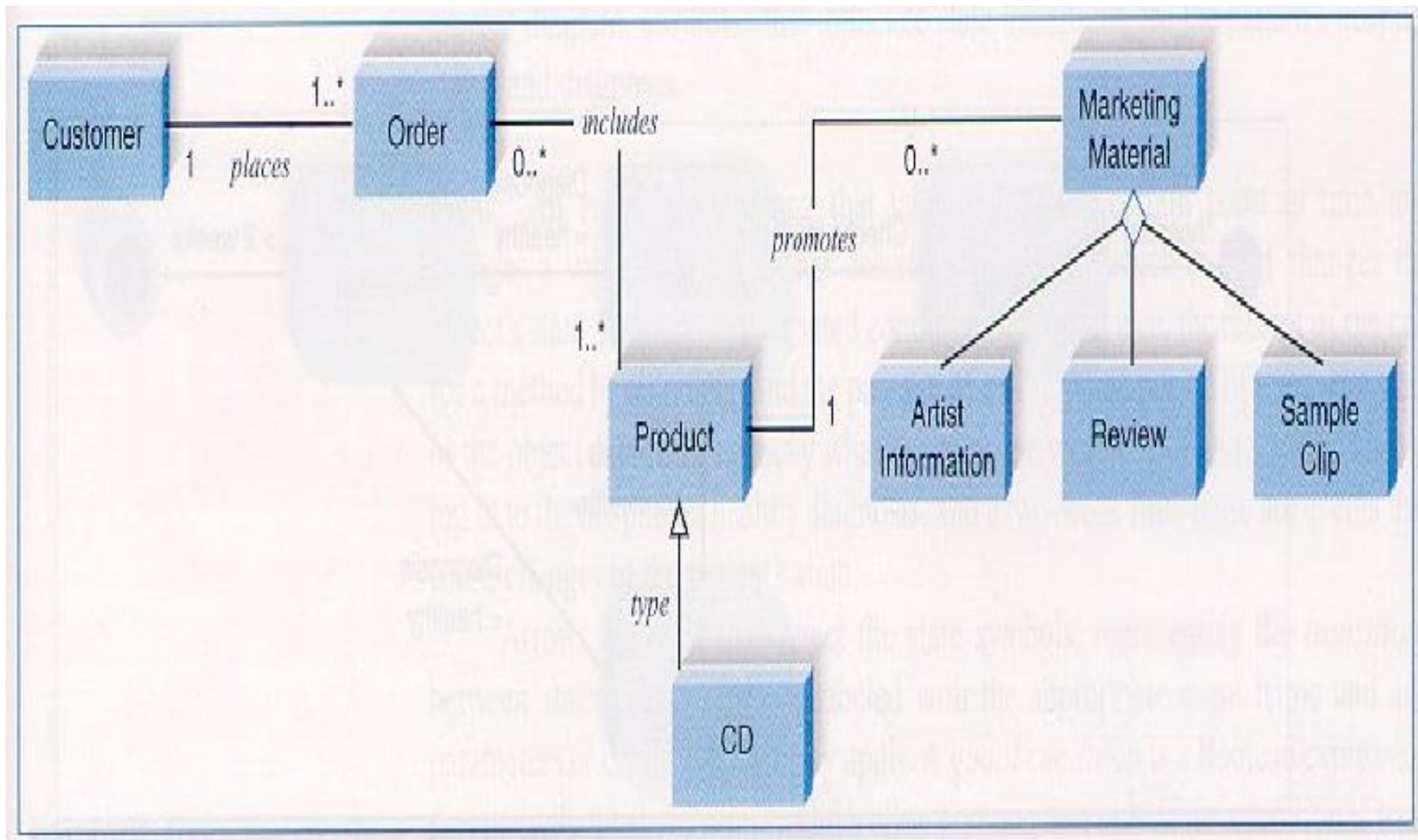
Ví dụ về sơ đồ lớp



Ví dụ về sơ đồ lớp



Ví dụ về sơ đồ lớp



XÂY DỰNG SƠ ĐỒ LỚP

Các bước xây dựng sơ đồ lớp

- **Xác định các lớp đối tượng**
- **Xác định quan hệ giữa các lớp đối tượng**
- **Vẽ sơ đồ lớp**
- **Xác định các thuộc tính**
- **Xác định các phương thức**

Xác định lớp đối tượng

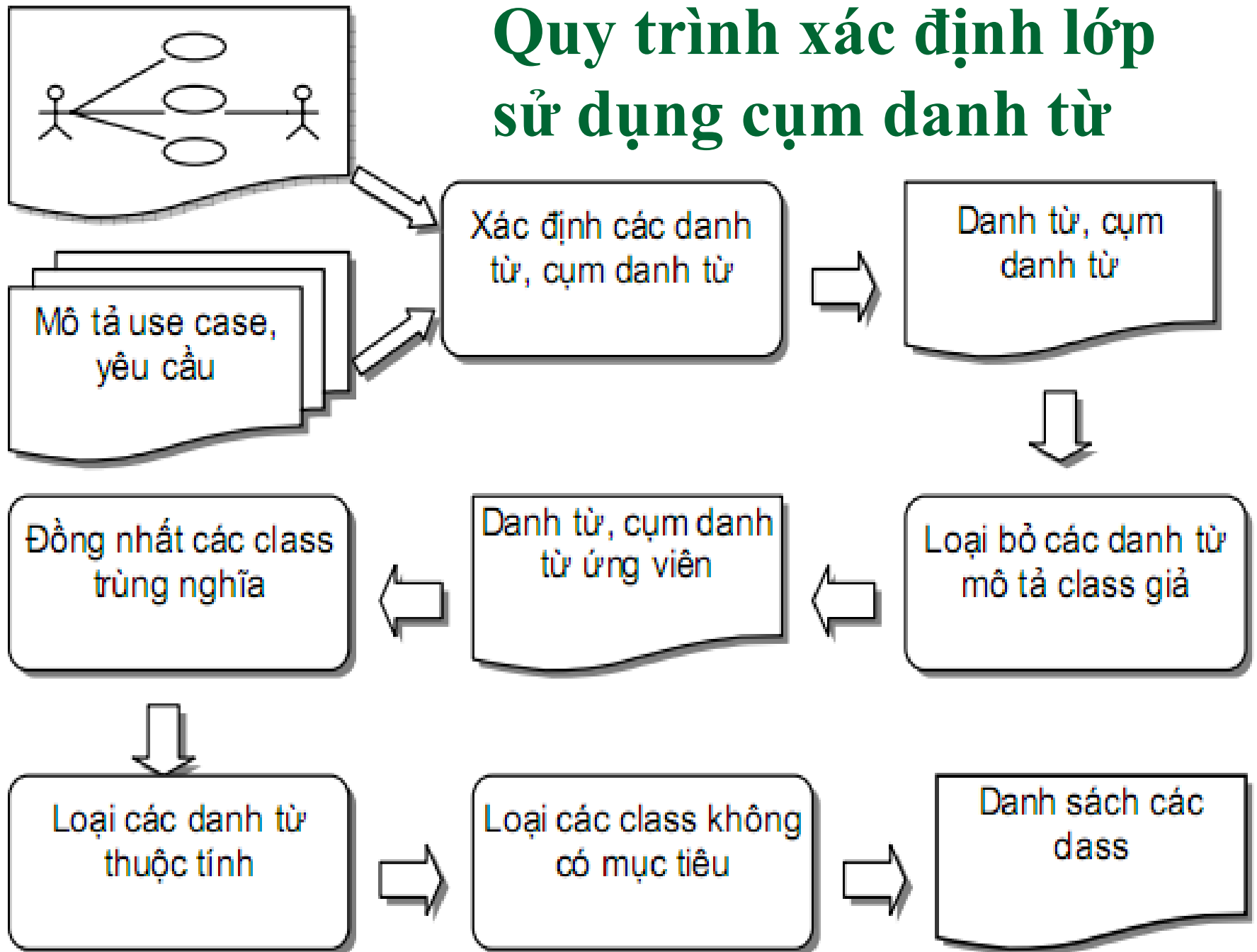
- Không có một phương thức chung để xác định các lớp đối tượng.
- Là quá trình sáng tạo và lặp đi lặp lại.
- Cần phải có sự thống nhất với các chuyên gia trong lĩnh vực ứng dụng nghiệp vụ.
- Các phương pháp xác định lớp:
 - Sử dụng cụm danh từ
 - Tiếp cận theo mẫu chung
 - Sử dụng use case

XÁC ĐỊNH LỚP SỬ DỤNG CỤM DANH TỪ

Giới thiệu

- Được đề xuất bởi Rebecca Wirfs-Brock, Brian Wilkerson và Lauren Wieber.
- Các lớp được xác định thông qua việc đọc các văn bản mô tả use case hoặc các mô tả yêu cầu để tìm kiếm và lấy ra các cụm danh từ.
- Các cụm danh từ lấy ra được gọi là các lớp ứng viên. Các cụm danh từ này được chia làm 3 loại:
 - ❑ Các lớp hiển nhiên
 - ❑ Các lớp mờ
 - ❑ Các lớp giả

Quy trình xác định lớp sử dụng cụm danh từ



Quy trình xác định lớp sử dụng cụm danh từ

B1: Khởi tạo danh sách các lớp ứng viên

- ❑ Tìm các danh từ hoặc cụm danh từ trong các mô tả use case, yêu cầu.
- ❑ Tất cả các lớp phải có ý nghĩa trong ứng dụng.
- ❑ Đặt tên cho lớp ứng viên.

B2: Loại bỏ các lớp giả

B3: Đồng nhất các lớp ứng viên trùng lặp

Xem xét lại danh sách các lớp ứng viên để loại bỏ bớt các danh từ, cụm danh từ có cùng ý nghĩa mặc dù có cách dùng từ khác nhau.

Quy trình xác định lớp sử dụng cụm danh từ

B4: Loại bỏ các danh từ, cụm danh từ có thể là các thuộc tính. Đó là các danh từ, cụm danh từ:

- ❑ Chỉ được sử dụng như là giá trị.
- ❑ Không có nhiều hơn một đặc trưng riêng hoặc chỉ mô tả một đặc trưng của đối tượng khác.

B5: Loại bỏ các lớp ứng viên không có mục tiêu hoặc không thuộc phạm vi hệ thống.

XÁC ĐỊNH LỚP SỬ DỤNG MẪU LỚP CHUNG

Giới thiệu

- Phương pháp này dựa trên một tập các mẫu lớp chung
- Khó áp dụng vì cần phải sẵn một tập các mẫu lớp chung
- Tập các mẫu lớp chung có thể được xây dựng dần dần
- Các mẫu chung bao gồm:
 - Lớp khái niệm (concept)
 - Lớp sự kiện (event)
 - Lớp tổ chức (organization)
 - Lớp con người (people)
 - Lớp vị trí (place)
 - Sự vật hữu hình và thiết bị

Các mẫu lớp chung

- Lớp khái niệm (concept)
 - Khái niệm là một quan niệm hoặc một sự hiểu biết riêng biệt về thế giới, là các ý tưởng, sự hiểu biết được chia sẻ trong cộng đồng và dùng để trao đổi.
 - Lớp khái niệm bao gồm các nguyên lý được dùng để tổ chức hoặc để lưu trữ các hoạt động và các trao đổi về mặt quản lý
 - Ví dụ: Phương pháp, phương pháp luận, mô hình,...

Các mẫu lớp chung

- Lớp sự kiện (event)
 - Các điểm thời gian cần lưu trữ
 - Các sự việc xảy ra tại một thời điểm hoặc một bước trong một dãy tuần tự các bước.
 - Liên quan đến các sự việc được lưu trữ là các thuộc tính (và các đối tượng chứa thuộc tính) như: ai, cái gì, khi nào, ở đâu, như thế nào, hoặc tại sao,...
 - Ví dụ: đăng ký, kết quả, hóa đơn, đơn hàng, phiếu nhập,....

Các mẫu lớp chung

- Lớp tổ chức (organization)
 - Là một tập hợp con người, tài nguyên, phương tiện, hoặc những nhóm xác định chức năng người dùng,...
 - Ví dụ: Đơn vị, phòng ban, chức danh,...
- Lớp vị trí (place)
 - Các vị trí vật lý mà hệ thống cần mô tả thông tin về nó.
 - Ví dụ: tòa nhà, kho, văn phòng, chi nhánh, đại lý,...

Các mẫu lớp chung

- Lớp con người (people)
 - Thể hiện vai trò khác nhau của người dùng trong việc tương tác với hệ thống.
 - Những đối tượng này thường là người dùng hệ thống hoặc những người không sử dụng hệ thống nhưng thông tin về họ được hệ thống lưu trữ
 - Ví dụ: Khách hàng, sinh viên, giáo viên, nhân viên,...

Các mẫu lớp chung

- Sự vật hữu hình và thiết bị
 - Các đối tượng vật lý hoặc các nhóm của đối tượng hữu hình có thể được cảm nhận trực quan và các thiết bị mà hệ thống tương tác.
 - Ví dụ:
 - Xe hơi, xe máy,... là các sự vật hữu hình.
 - Dụng cụ cảm ứng nhiệt, đầu đọc mã vạch,... là các thiết bị.

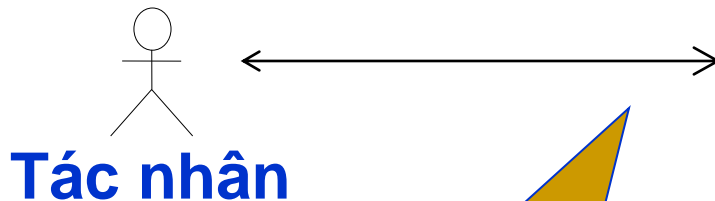
XÁC ĐỊNH LỚP BẰNG CÁCH PHÂN TÍCH HOẠT ĐỘNG USECASE

Xác định lớp bằng cách phân tích hoạt động usecase

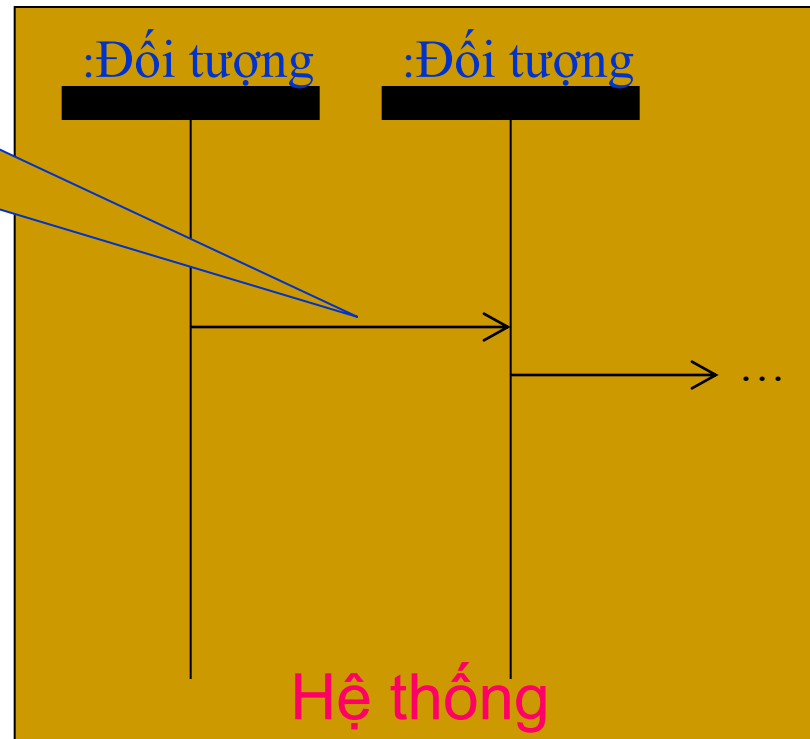
- Xác định các dòng tương tác của tác nhân với hệ thống trong một use case sử dụng sơ đồ cộng tác hoặc tuần tự.
 - Xác định các đối tượng bằng cách trả lời các câu hỏi:
 - Đối tượng nào của hệ thống sẽ trực tiếp tương tác với tác nhân?
 - Đối tượng nào tiếp theo được chuyển giao trách nhiệm?
 - Lặp lại các câu hỏi trên cho đến khi xét hết các dòng tương tác.
- Lần lượt xét các UC có trong hệ thống để xác định tất cả các đối tượng của hệ thống.

Xác định lớp bằng cách phân tích hoạt động usecase

Đối tượng nào tiếp theo được chuyển giao trách nhiệm ?



Đối tượng nào của hệ thống sẽ trực tiếp tương tác với tác nhân?



Kết thúc mô tả kịch bản của use case cho phép xác định tất cả các lớp liên tham gia.

Các bước xây dựng sơ đồ lớp

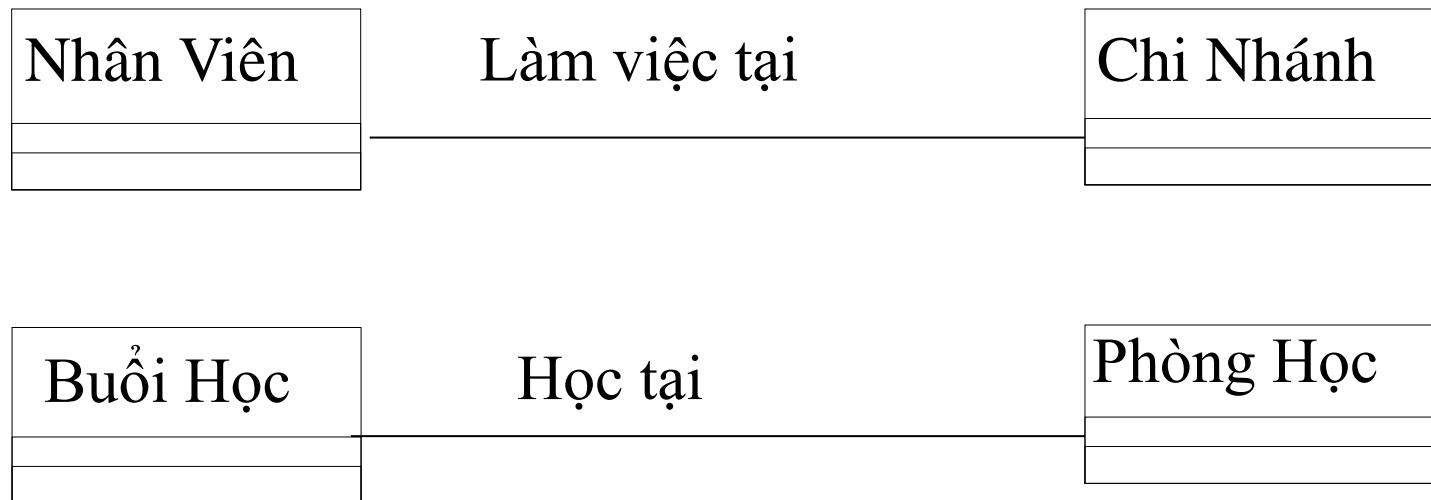
- Xác định các lớp đối tượng
- **Xác định quan hệ giữa các lớp đối tượng**
- **Vẽ sơ đồ lớp**
- Xác định các thuộc tính
- Xác định các hoạt động (phương thức)

Xác định quan hệ kết hợp (association)

- Một sự phụ thuộc giữa hai hay nhiều lớp có thể thiết lập thành mỗi kết hợp. Mỗi kết hợp thường tương ứng với một động từ hoặc cụm giới từ như là *thành phần của*, *làm việc cho*, *chứa trong*, ...
- Một tham chiếu từ một lớp đến một lớp khác là một quan hệ kết hợp.

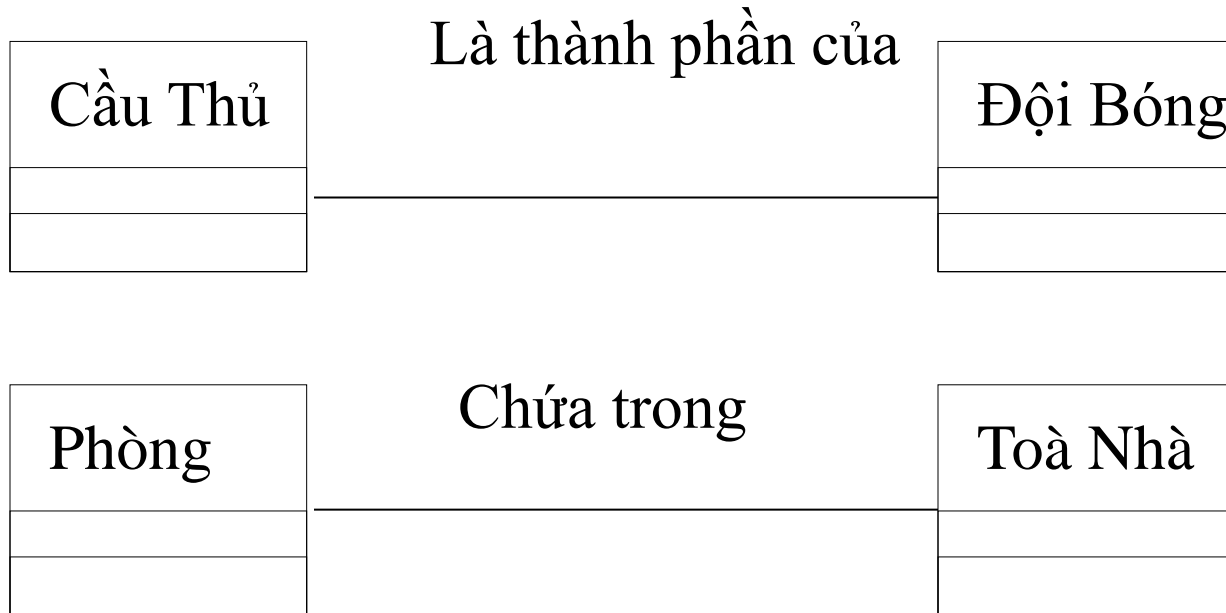
Xác định quan hệ kết hợp (association)

- Các mẫu xác định mối kết hợp:
 - Mối kết hợp vị trí (location): *liên kết tới, thành phần của, làm việc tại,*
 - Ví dụ:



Xác định quan hệ kết hợp (association)

- Các mẫu xác định mối kết hợp:
 - Mối kết hợp sở hữu: *của, có, thuộc, ...*
 - Ví dụ:



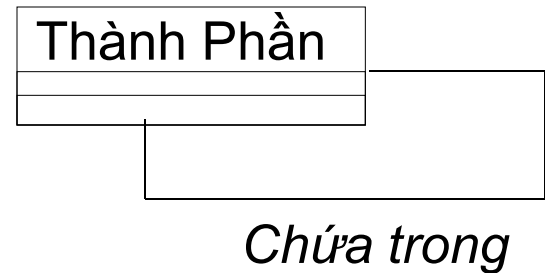
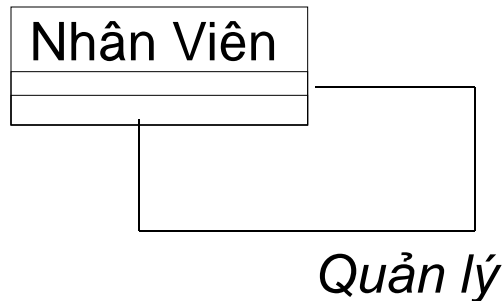
Xác định quan hệ kết hợp (association)

- Các mẫu xác định mối kết hợp:
 - Mối kết hợp truyền thông, liên lạc (communication):
đặt tới, trao đổi với, gửi cho, tiếp nhận từ, ...
 - Ví dụ:



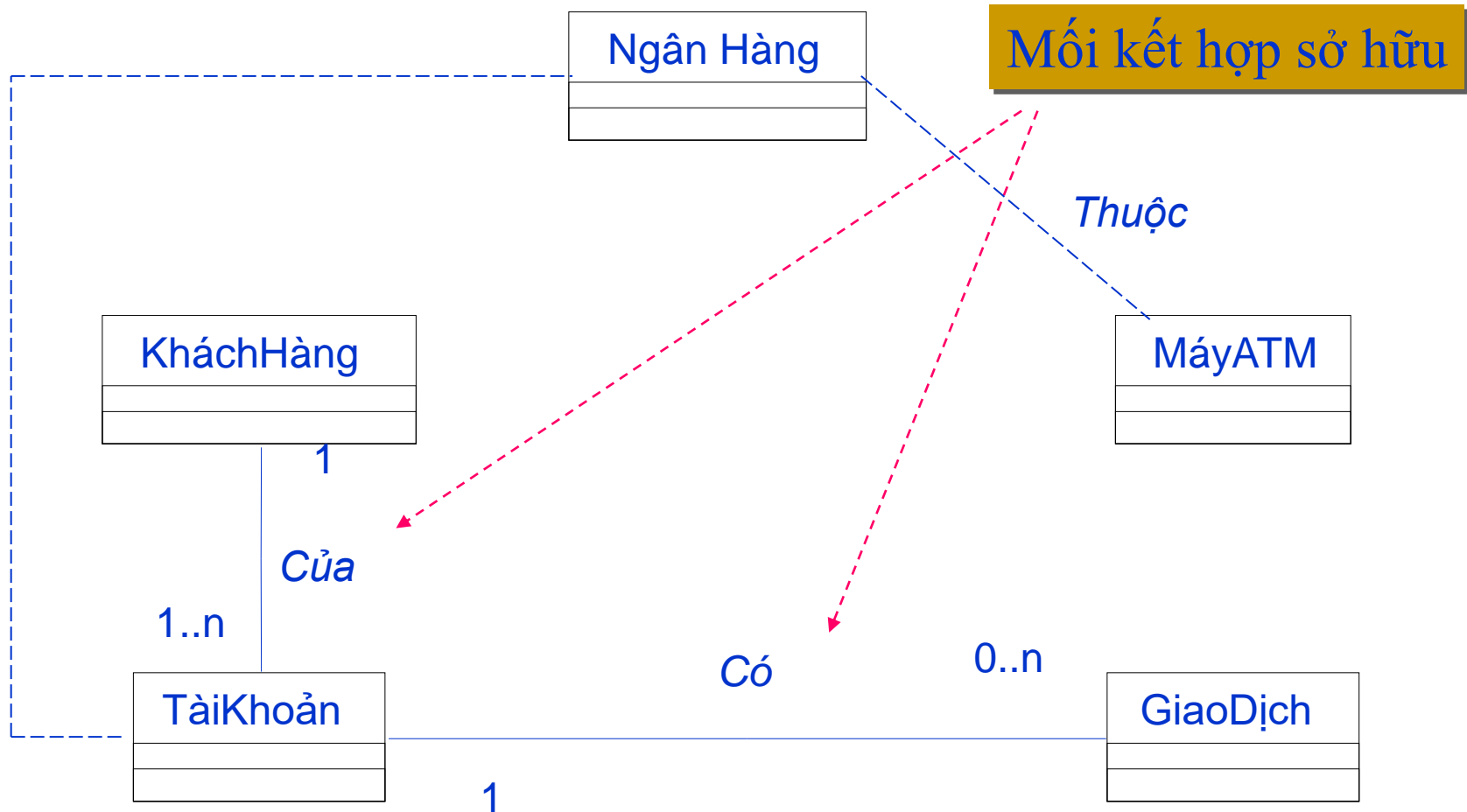
Xác định quan hệ kết hợp (association)

- Mỗi kết hợp phản thân: là mối quan hệ được thiết lập giữa một đối tượng của một lớp với một đối tượng khác cũng thuộc lớp đó.
- Ví dụ:



Xác định quan hệ kết hợp (association)

■ Ví dụ: hệ thống ATM

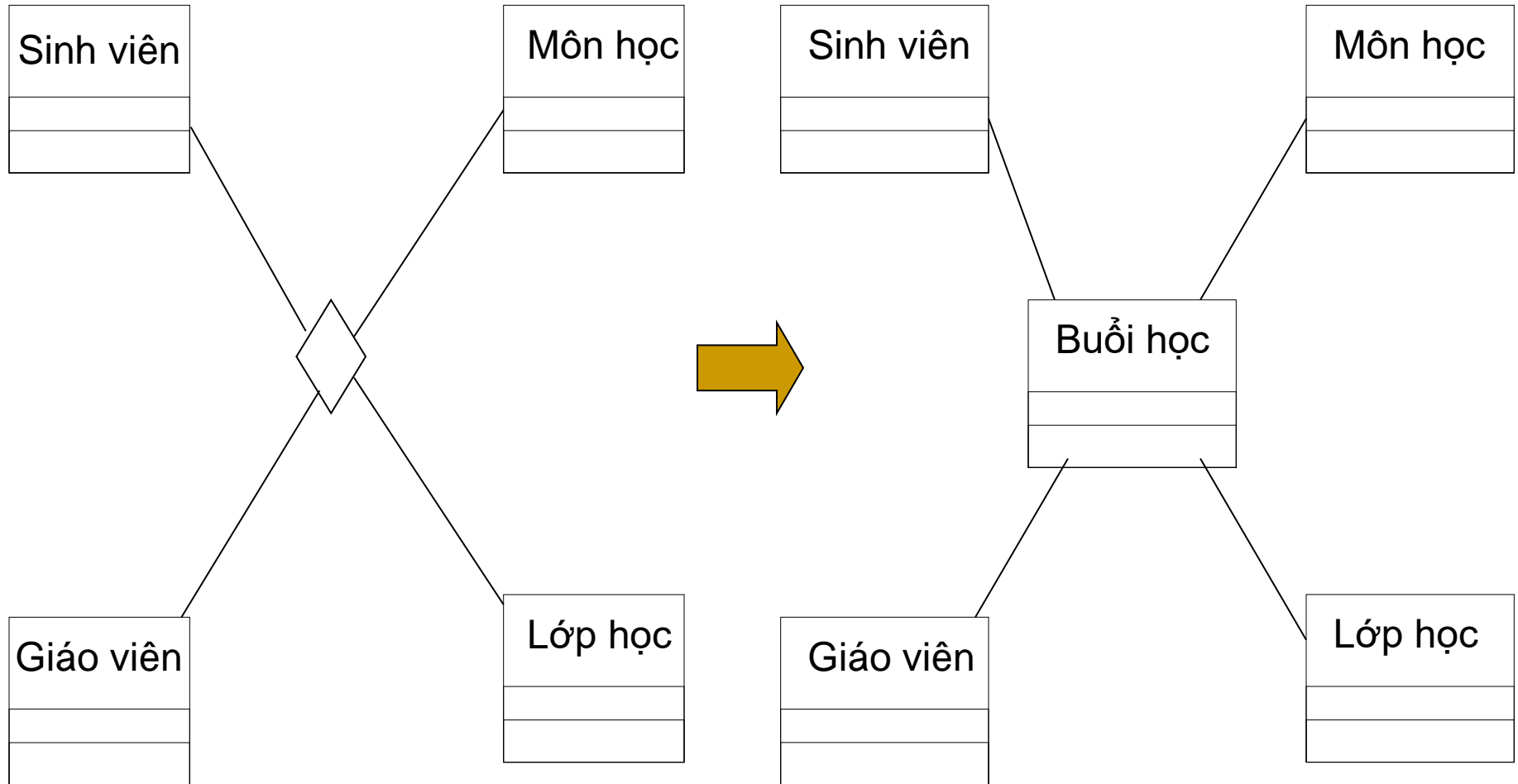


Xác định quan hệ kết hợp (association)

- Loại bỏ các mối kết hợp không cần thiết:
 - **Mối kết hợp cài đặt:** là mối kết hợp mô tả sự liên quan giữa các lớp trong giai đoạn thiết kế cài đặt hệ thống bên trong môi trường phát triển hoặc ngôn ngữ lập trình cụ thể và không phải là mối liên kết giữa các đối tượng mô tả nghiệp vụ
 - **Mối kết hợp đa phân:** là mối kết hợp giữa ba lớp trở lên, mối kết hợp này phức tạp trong cách thể hiện → Nếu có thể, phát biểu lại nó dùng mối kết hợp nhị phân

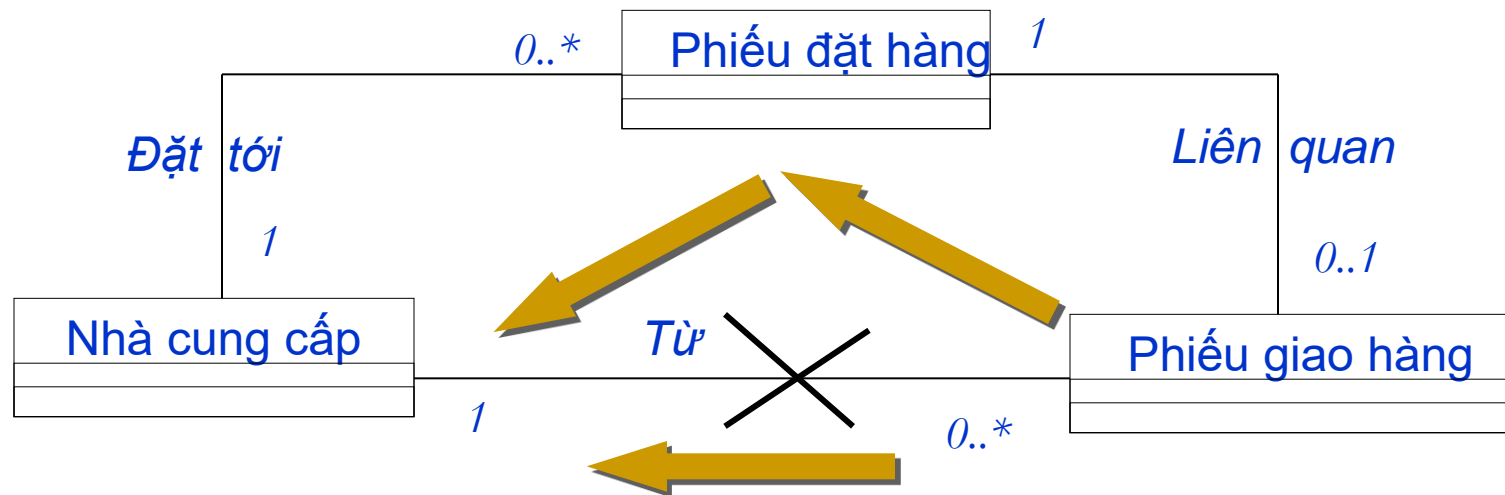
Xác định quan hệ kết hợp (association)

- Ví dụ: Loại bỏ kết hợp không cần thiết (đa phân)



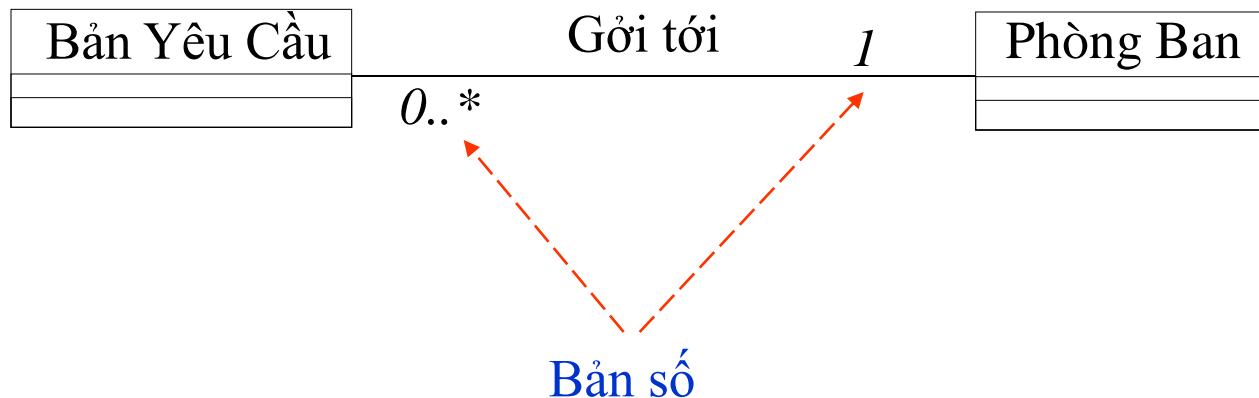
Xác định quan hệ kết hợp (association)

- Loại bỏ các mối kết hợp không cần thiết:
 - ❑ **Mối kết hợp trực tiếp dư thừa:** là các mối kết hợp được định nghĩa trong ngữ nghĩa của những mối kết hợp khác (còn gọi là mối kết hợp suy diễn hoặc bắc cầu)



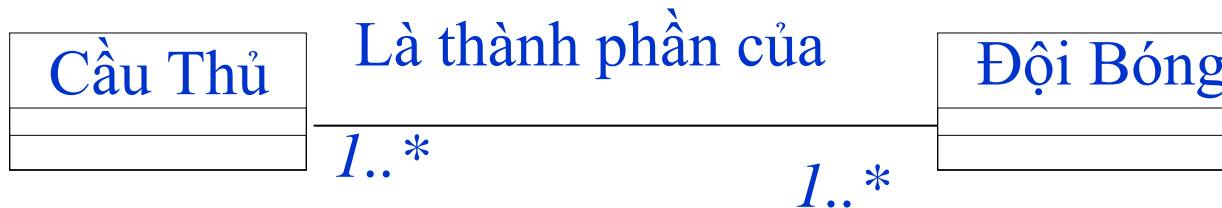
Xác định quan hệ kết hợp (association)

- Xác định bản số cho mỗi kết hợp: (min, max)
 - 1; 0..1;
 - 1..*;
 - 0..*;
 - a..* : a là hằng



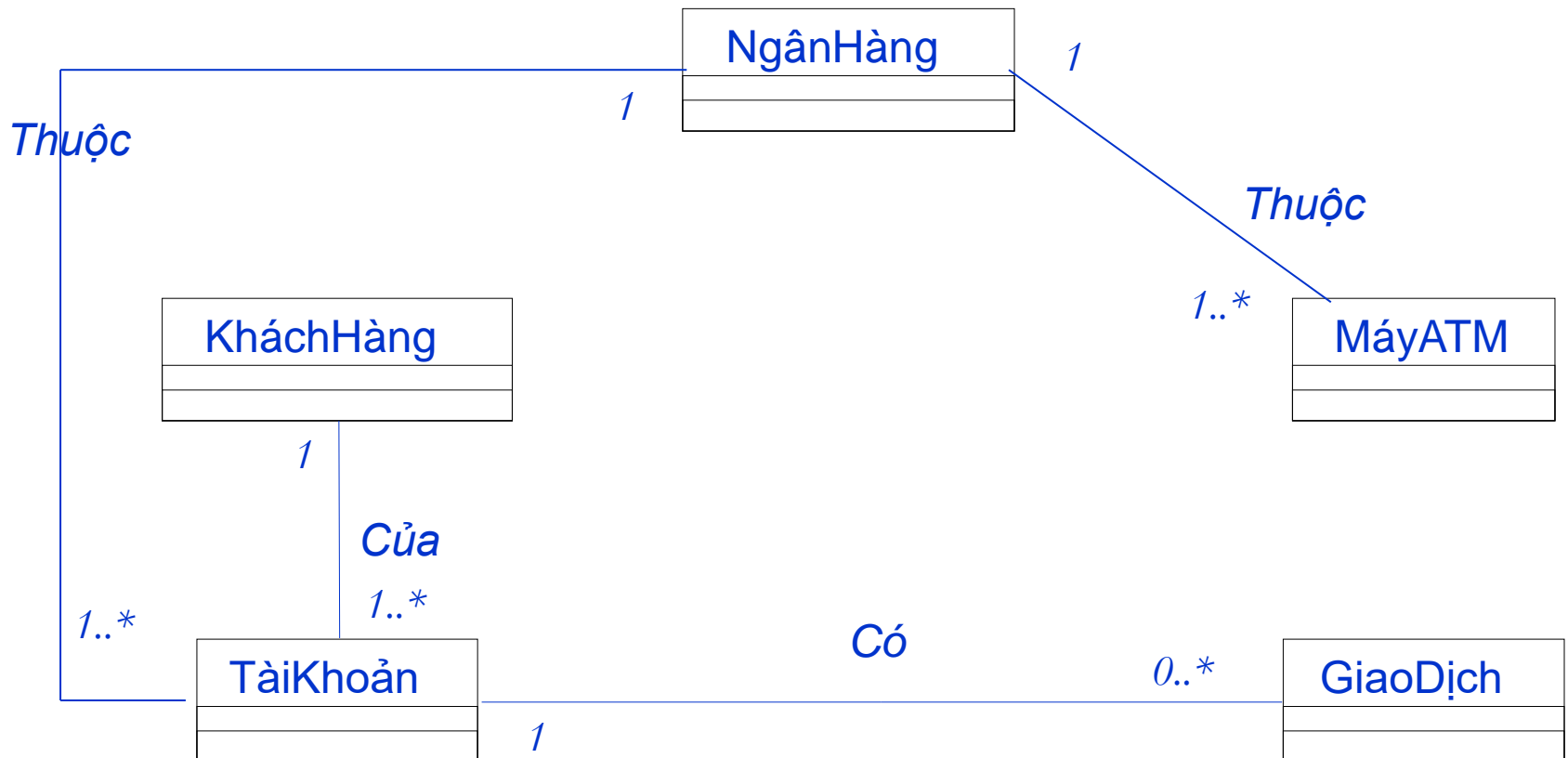
Xác định quan hệ kết hợp (association)

- Xác định bản số cho mỗi kết hợp
 - Ví dụ:

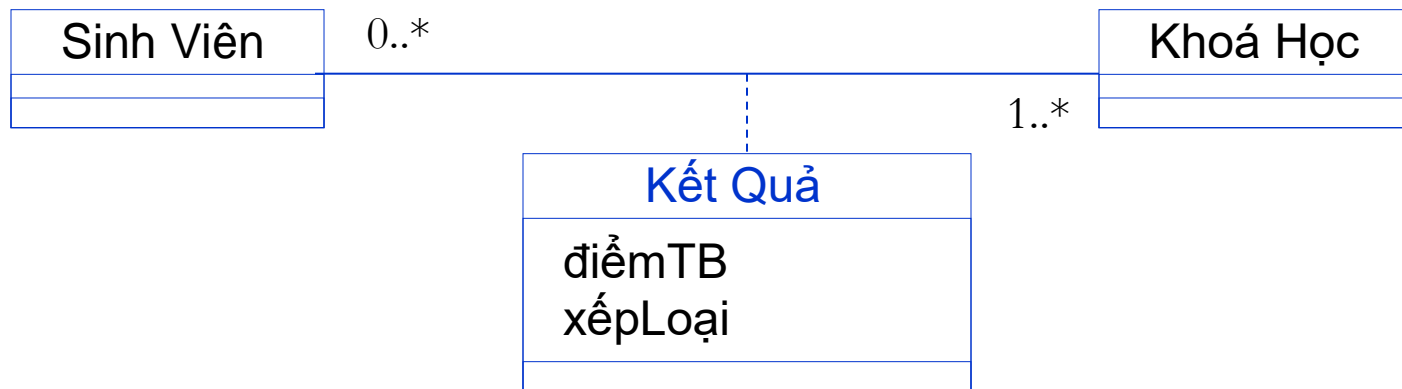


Xác định quan hệ kết hợp (association)

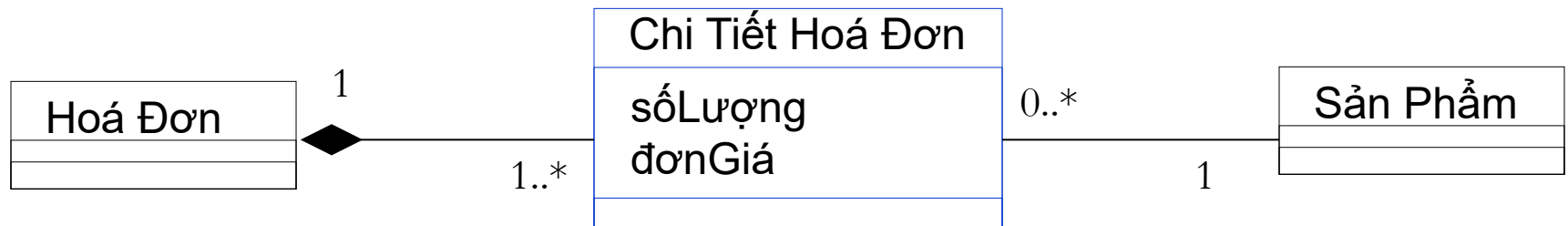
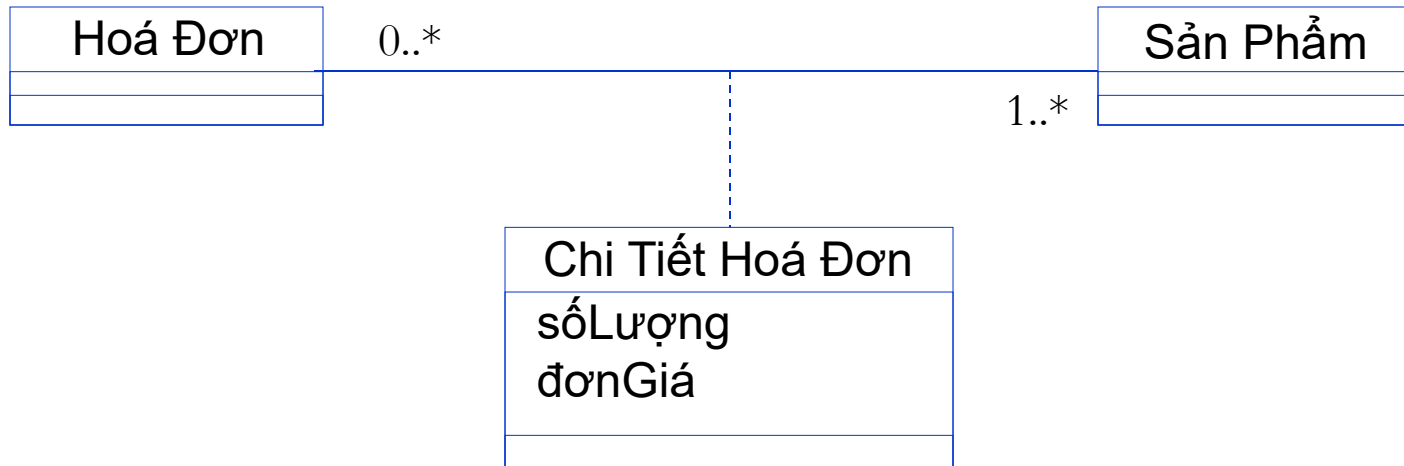
- Xác định bản số cho mỗi kết hợp:
 - Ví dụ: hệ thống ATM



Xác định lớp kết hợp (association class)



Xác định lớp kết hợp (association class)



Xác định quan hệ kế thừa (generalization)

- Thể hiện quan hệ kế thừa giữa các lớp và một cấu trúc phân cấp xác định những dòng kế thừa này.
- Tiếp cận top-down:
 - Từ một lớp chúng ta tìm kiếm cụm danh từ chứa tên lớp và tính từ (hoặc danh từ). Đánh giá xem cụm danh từ này có thể là một trường hợp đặc biệt cần được quản lý trong hệ thống không?
 - Tìm kiếm các đặc trưng riêng của lớp.
 - Xây dựng mối kết hợp chuyên biệt từ lớp này đến lớp ban đầu.

Xác định quan hệ kế thừa (generalization)

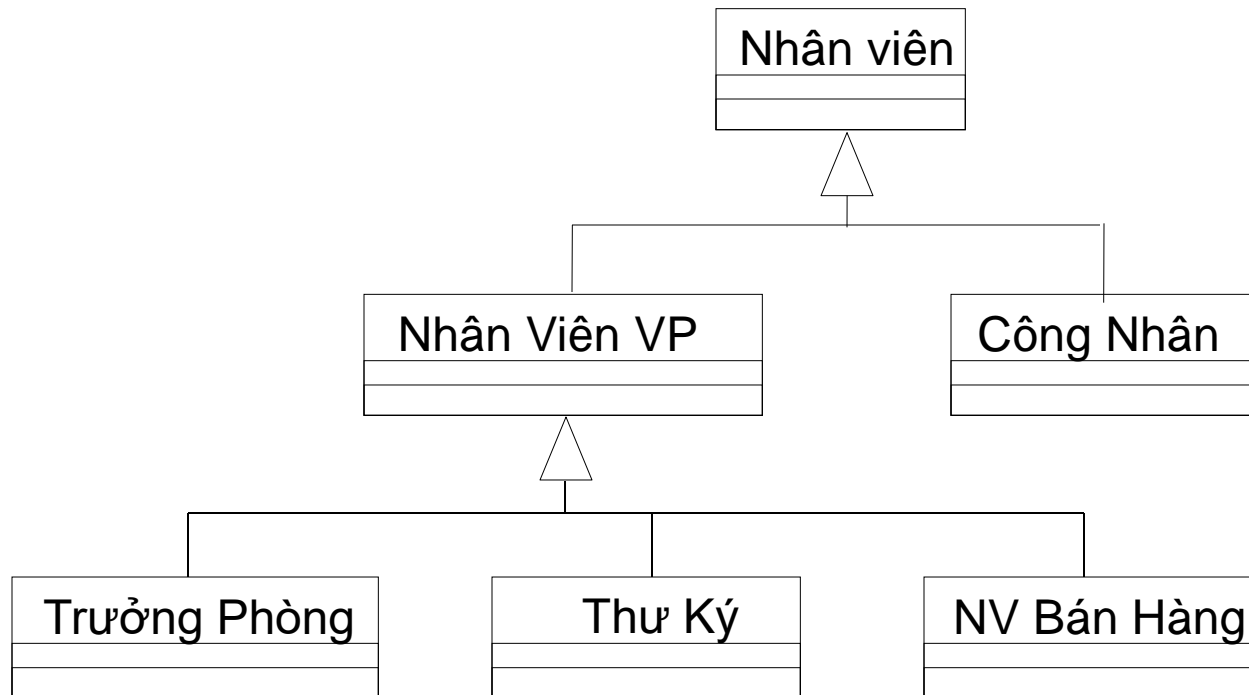
■ Ví dụ 1: Tiếp cận top-down



Ghi chú: chỉ cần đưa vào các lớp chuyên biệt mà chúng ta xác định được các đặc trưng riêng (thuộc tính, phương thức, quan hệ) của nó trong hệ thống.

Xác định quan hệ kế thừa (generalization)

■ Ví dụ 2:

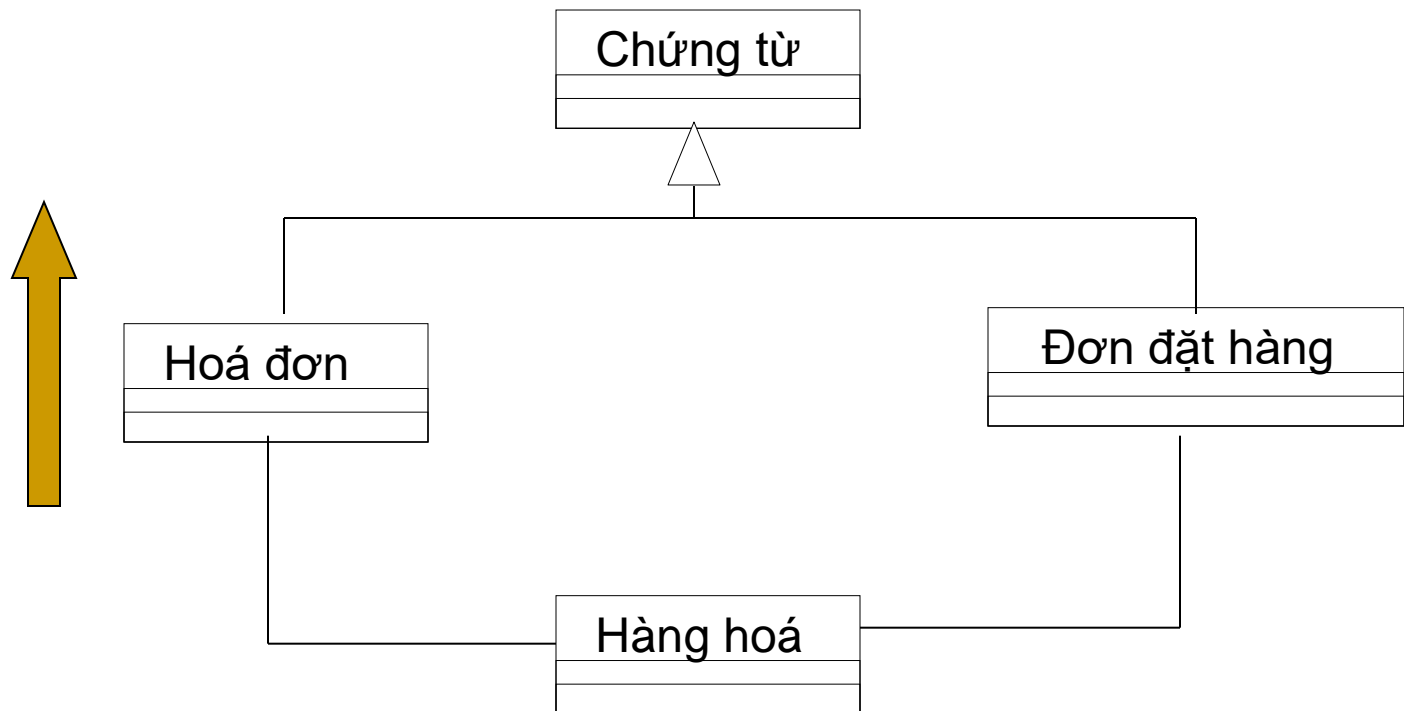


Xác định quan hệ kế thừa (generalization)

- Tiếp cận bottom-up:
 - Tìm kiếm trong các lớp để xác định xem có các thuộc tính và phương thức giống nhau không. Sau đó gom nhóm và đưa các thuộc tính và phương thức chung này lên một lớp tổng quát (trừu tượng).
 - Tạo mối kết hợp tổng quát hoá từ các lớp này đến lớp tổng quát mới xác định.

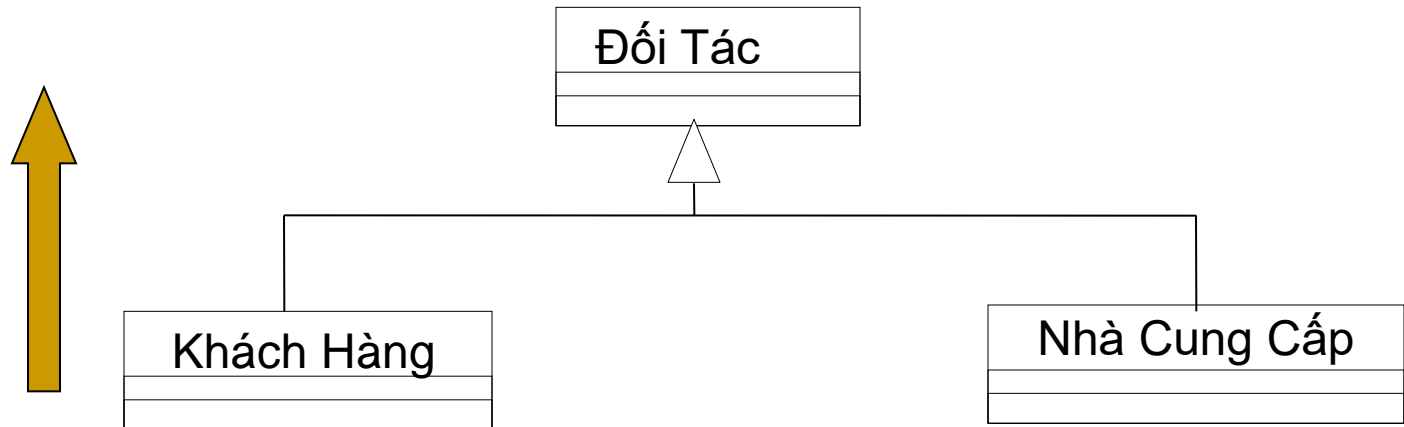
Xác định quan hệ kế thừa (generalization)

- Tiếp cận bottom-up – ví dụ:



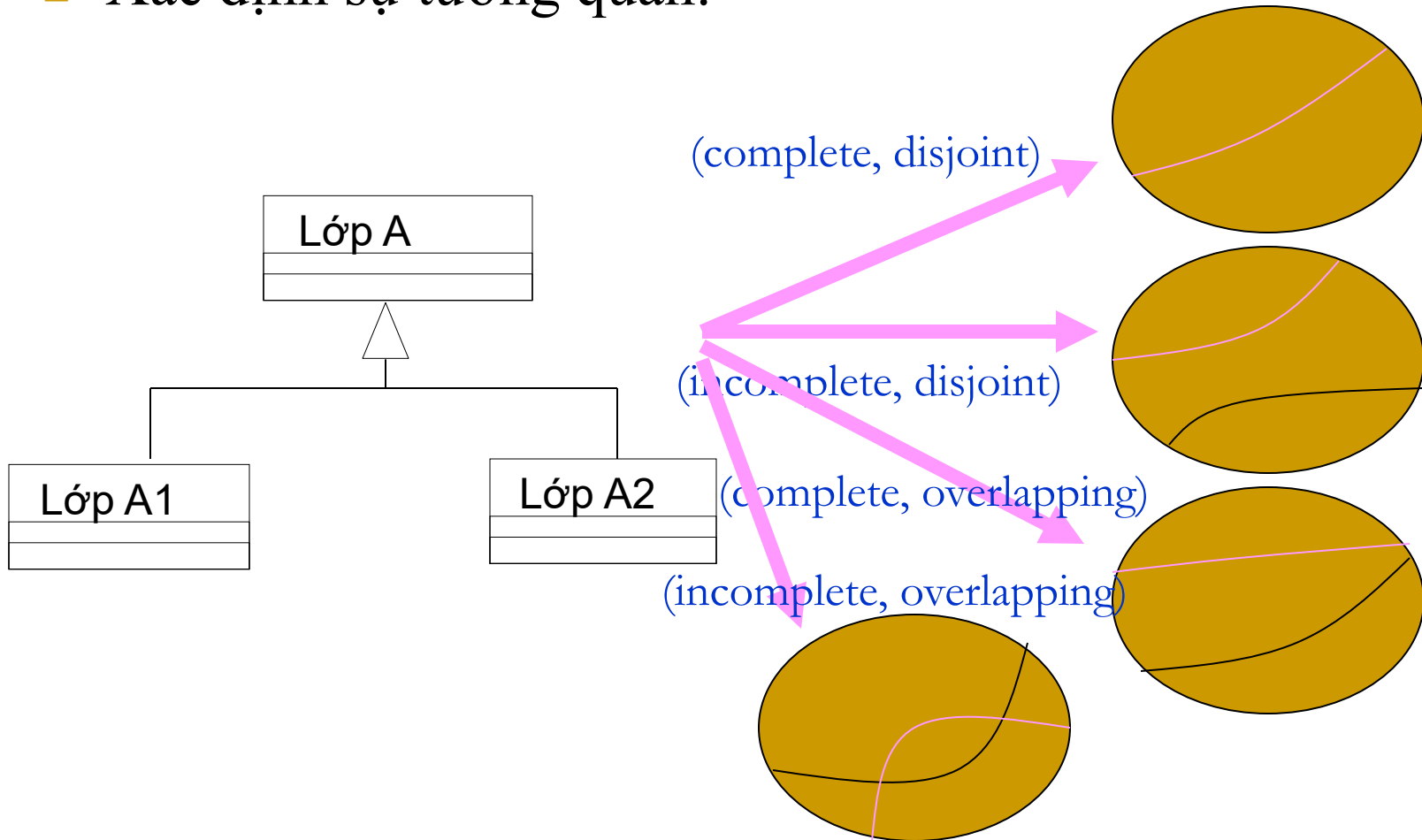
Xác định quan hệ kế thừa (generalization)

- Tiếp cận bottom-up – ví dụ 2:



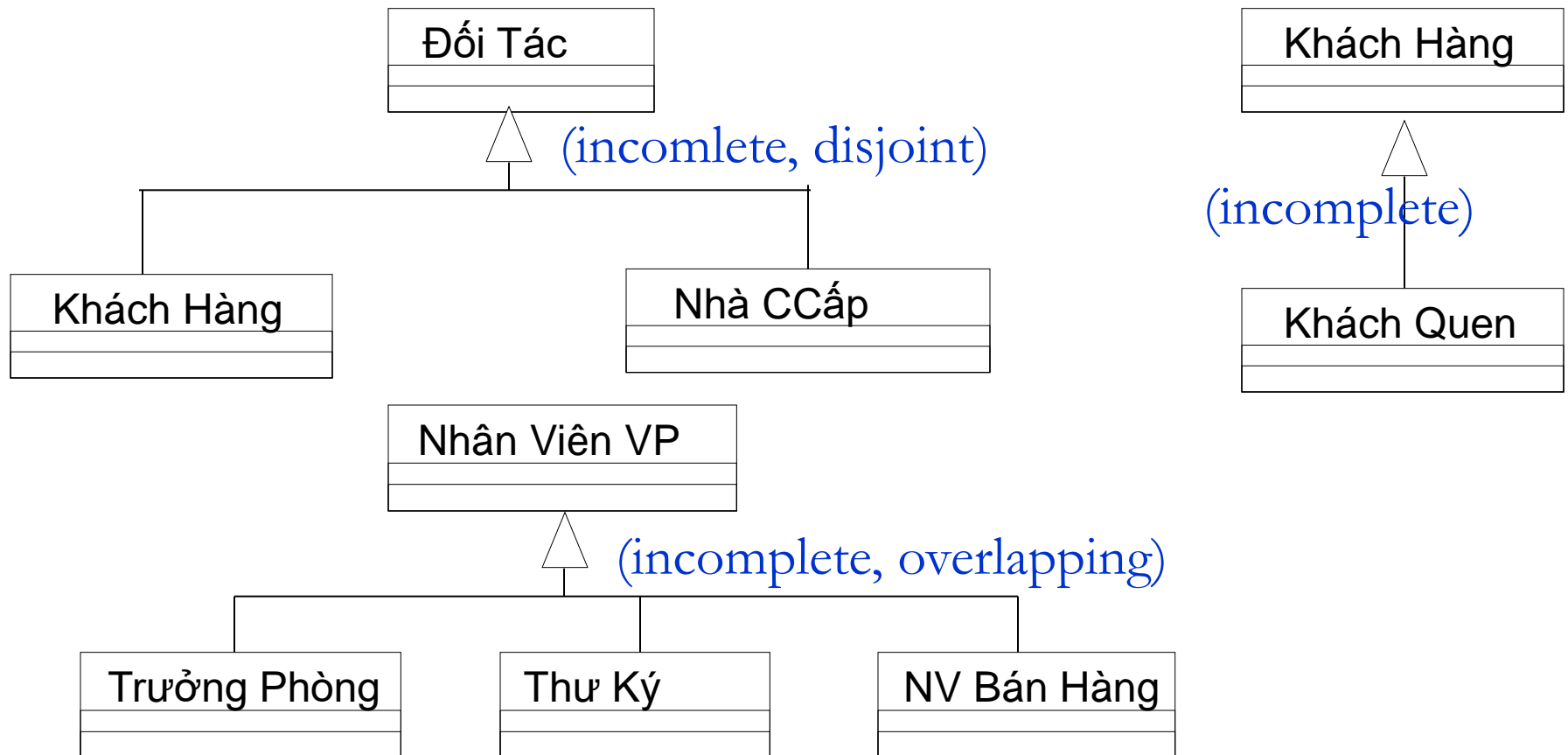
Xác định quan hệ kế thừa (generalization)

■ Xác định sự tương quan:



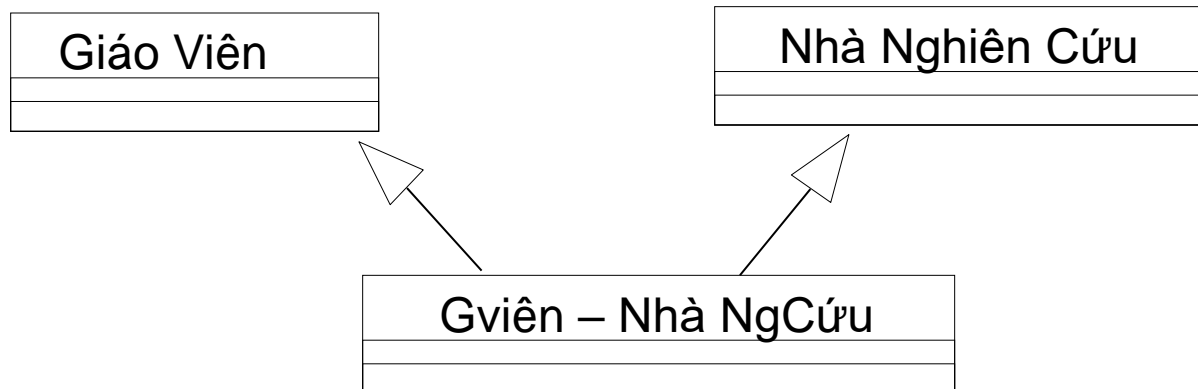
Xác định quan hệ kế thừa (generalization)

- Xác định sự tương quan: - ví dụ



Xác định quan hệ kế thừa (generalization)

- Vấn đề đa thừa kế:
 - Phức tạp trong vấn đề kế thừa
 - → Không nên sử dụng (phiên bản gốc UML không đưa vào)

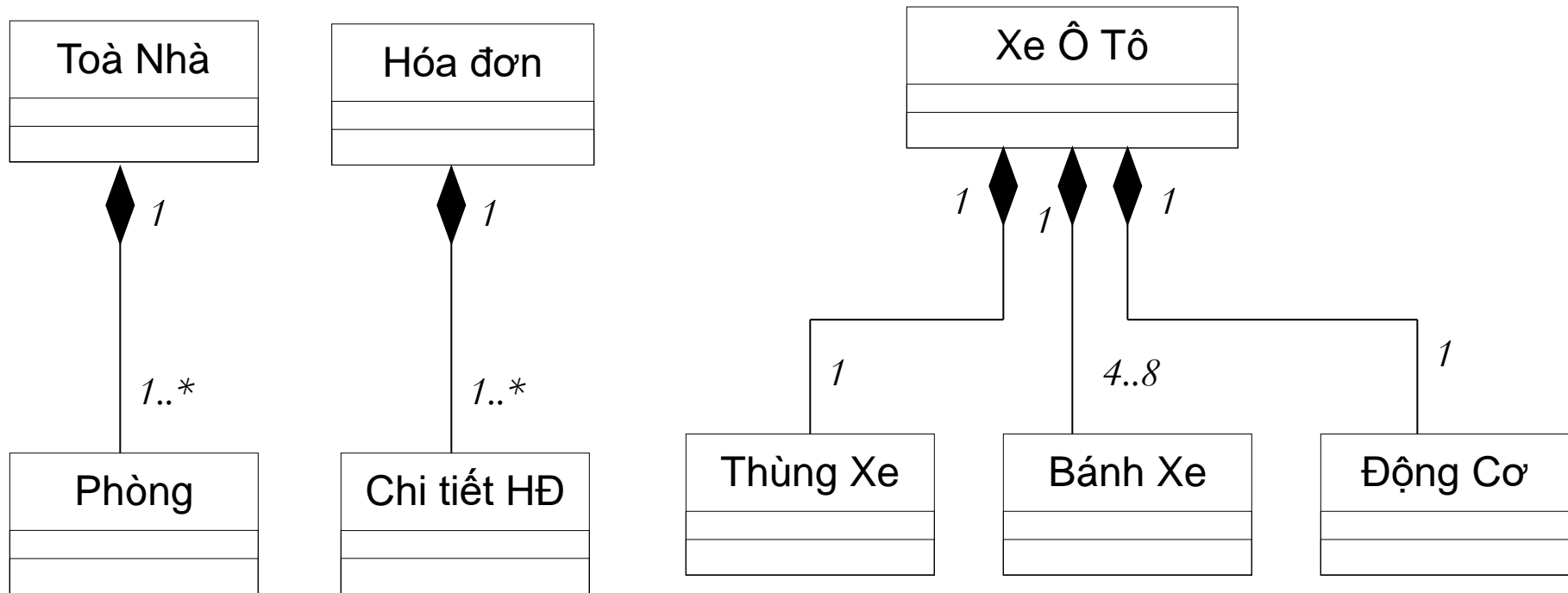


Xác định quan hệ kết tập (aggregation)

- Quan hệ kết tập được sử dụng trong tình huống một lớp hình thành bao gồm những lớp thành phần.
- Đặc trưng cơ bản:
 - **Tính bắc cầu:** Nếu **lớp A** là một thành phần của **lớp B** và **lớp B** là thành phần của **lớp C** \rightarrow **lớp A** là thành phần của **lớp C**.
 - **Tính phản đối xứng:** nếu **lớp A** là thành phần của **lớp B** thì **lớp B** không phải là thành phần của **lớp A**.
- Việc xác định quan hệ thành phần có thể dựa trên các kiểu quan hệ: Tập hợp, vật chứa, tập hợp-thành viên.

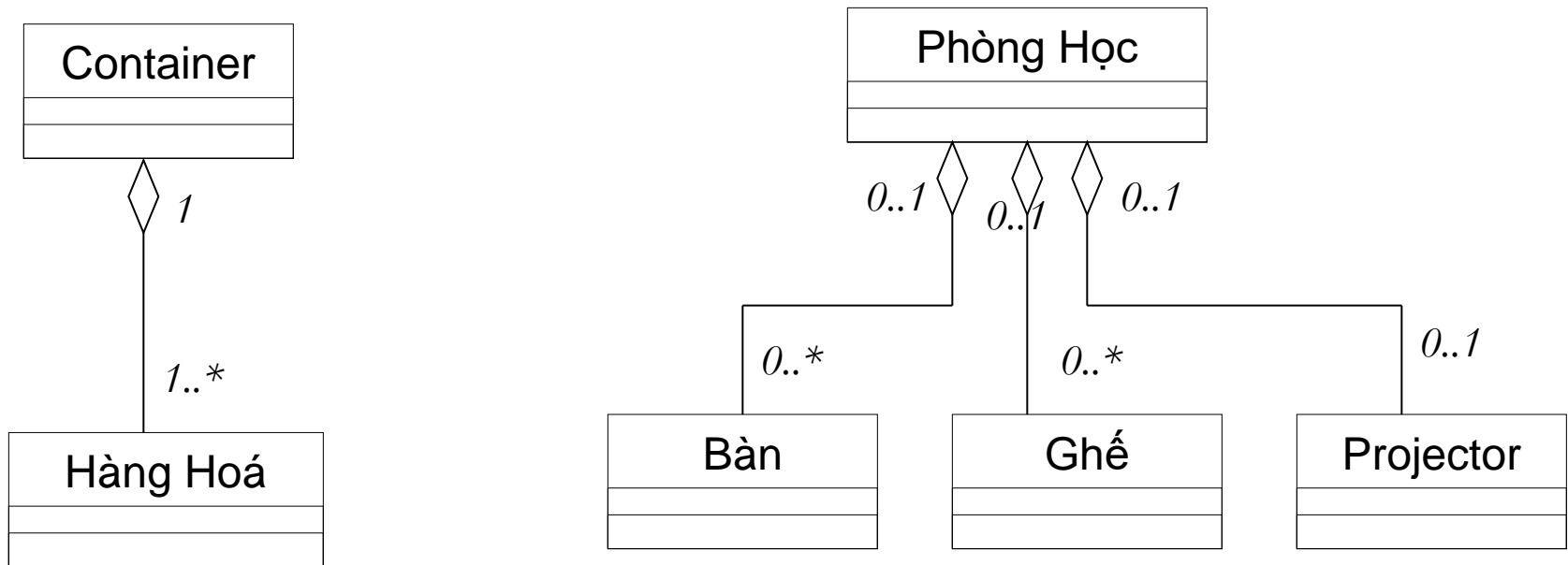
Xác định quan hệ kết tập (aggregation)

- Tập hợp: một đối tượng vật lý được hình thành từ các đối tượng vật lý thành phần khác



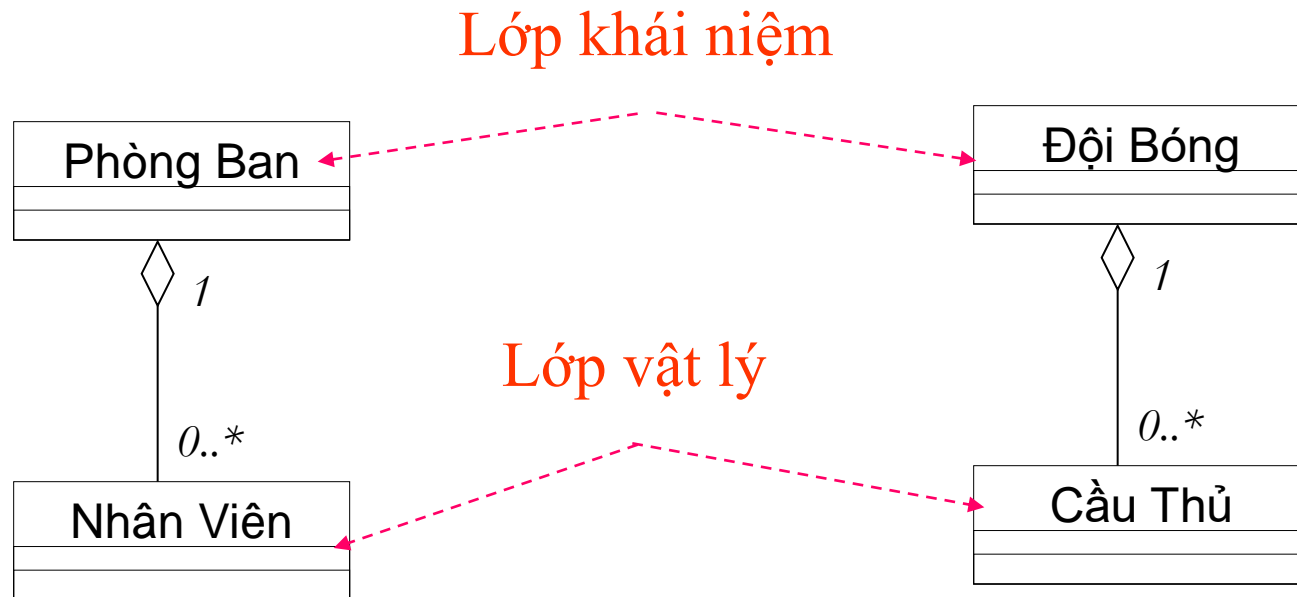
Xác định quan hệ kết tập (aggregation)

- Vật chứa: một đối tượng vật lý chứa đựng các thành phần nhưng không được cấu tạo bởi các thành phần



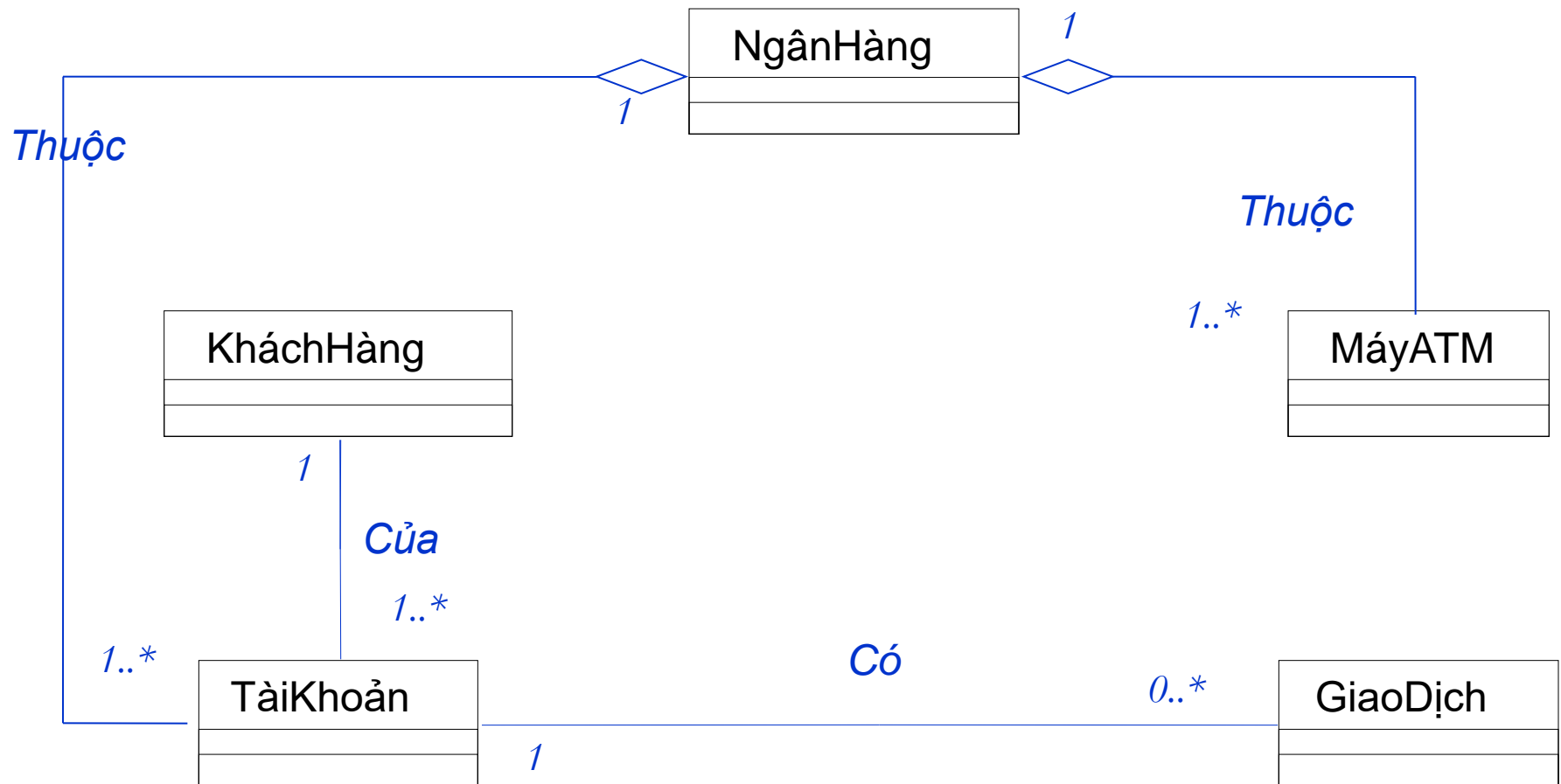
Xác định quan hệ kết tập (aggregation)

- Tập hợp – thành viên: một đối tượng khái niệm chứa các thành phần có thể vật lý hoặc khái niệm



Xác định quan hệ kết tập (aggregation)

- Ví dụ: hệ thống ATM



Các bước xây dựng sơ đồ lớp

- Xác định các lớp đối tượng
- Xác định quan hệ giữa các lớp đối tượng
- Vẽ sơ đồ lớp
- **Xác định các thuộc tính**
- Xác định các hoạt động

Xác định thuộc tính

- Trả lời câu hỏi sau đối với mỗi đối tượng đã được xác định:
 - ❑ Thông tin gì về đối tượng sẽ được quản lý?
- Nguyên tắc:
 - ❑ Tên: danh từ hoặc cụm danh từ
 - ❑ Đơn giản: chỉ dùng đủ thuộc tính để diễn đạt trạng thái đối tượng ở giai đoạn phân tích (thuộc tính sẽ được bổ sung chi tiết hơn ở các giai đoạn tiếp theo)
 - ❑ Không quá quan tâm về việc phải tìm ra đầy đủ các thuộc tính
 - ❑ Không quan tâm đến các thuộc tính mô tả cài đặt của đối tượng

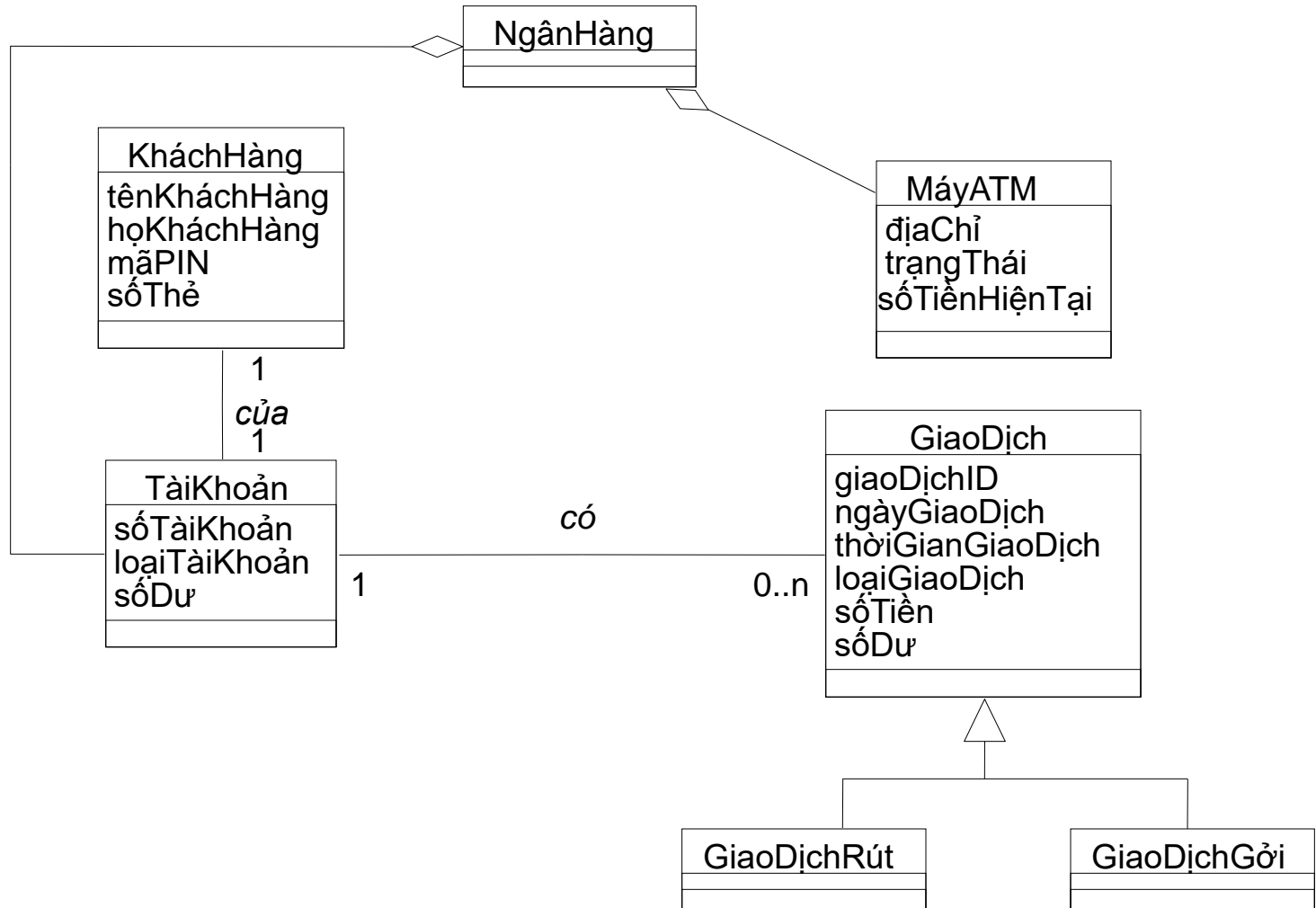
Xác định thuộc tính

- Ví dụ: hệ thống ATM
 - Lớp Khách Hàng: Phân tích lần lượt tất cả các use case có liên quan đến lớp Khách Hàng như là: “Đăng nhập”, “Xử lý PIN không hợp lệ”. Các thuộc tính của lớp khách hàng được xác định gồm:

KháchHàng
tênKháchHàng họKháchHàng mãPIN sốThẻ

Xác định thuộc tính

■ Ví dụ: hệ thống ATM



Các bước xây dựng sơ đồ lớp

- Xác định các lớp đối tượng
- Xác định quan hệ giữa các lớp đối tượng
- Vẽ sơ đồ lớp
- Xác định các thuộc tính
- **Xác định các phương thức**

Xác định phương thức

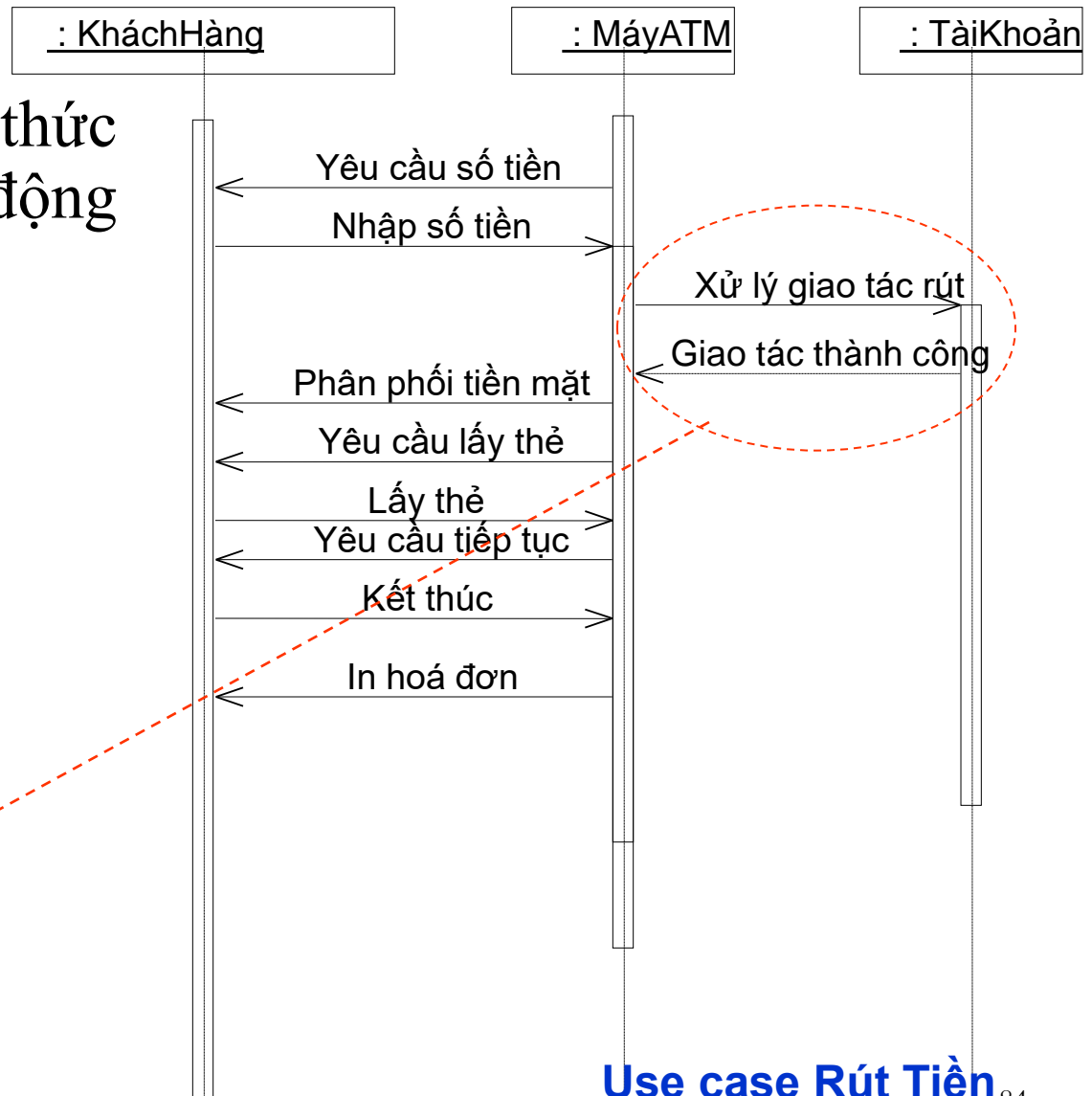
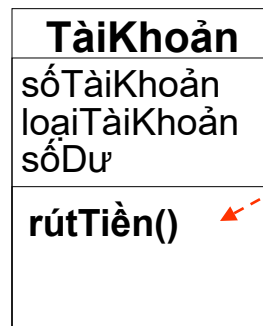
- Trả lời câu hỏi sau đối với mỗi đối tượng đã được xác định: **Các đối tượng chịu trách nhiệm xử lý gì về thông tin của nó để cung cấp dịch vụ cho hệ thống?**
- Nguyên tắc:
 - Tên: động từ + bổ ngữ
 - Chỉ quan tâm đến các phương thức có phạm vi toàn cục (public), các phương thức có phạm vi cục bộ sẽ được phát hiện và bổ sung trong giai đoạn thiết kế cài đặt (vd: constructor,...)
 - Các phương thức chịu trách nhiệm về các thao tác lên các thuộc tính của đối tượng: truy vấn, cập nhật, đọc và ghi.

Xác định phương thức thông qua phân tích hoạt động use case

- Phân tích các dòng thông điệp trong sơ đồ tuần tự để xem có thể chuyển một hoạt động thành một phương thức không?
- Nếu có, đặt tên cho phương thức ứng với hoạt động đó

Xác định phương thức thông qua phân tích hoạt động use case

- Xác định phương thức qua phân tích hoạt động use case:
- Ví dụ: lớp TàiKhoản
 - ❑ Rút Tiền
 - ❑ Gửi Tiền
 - ❑ Xem thông Tin Tài Khoản



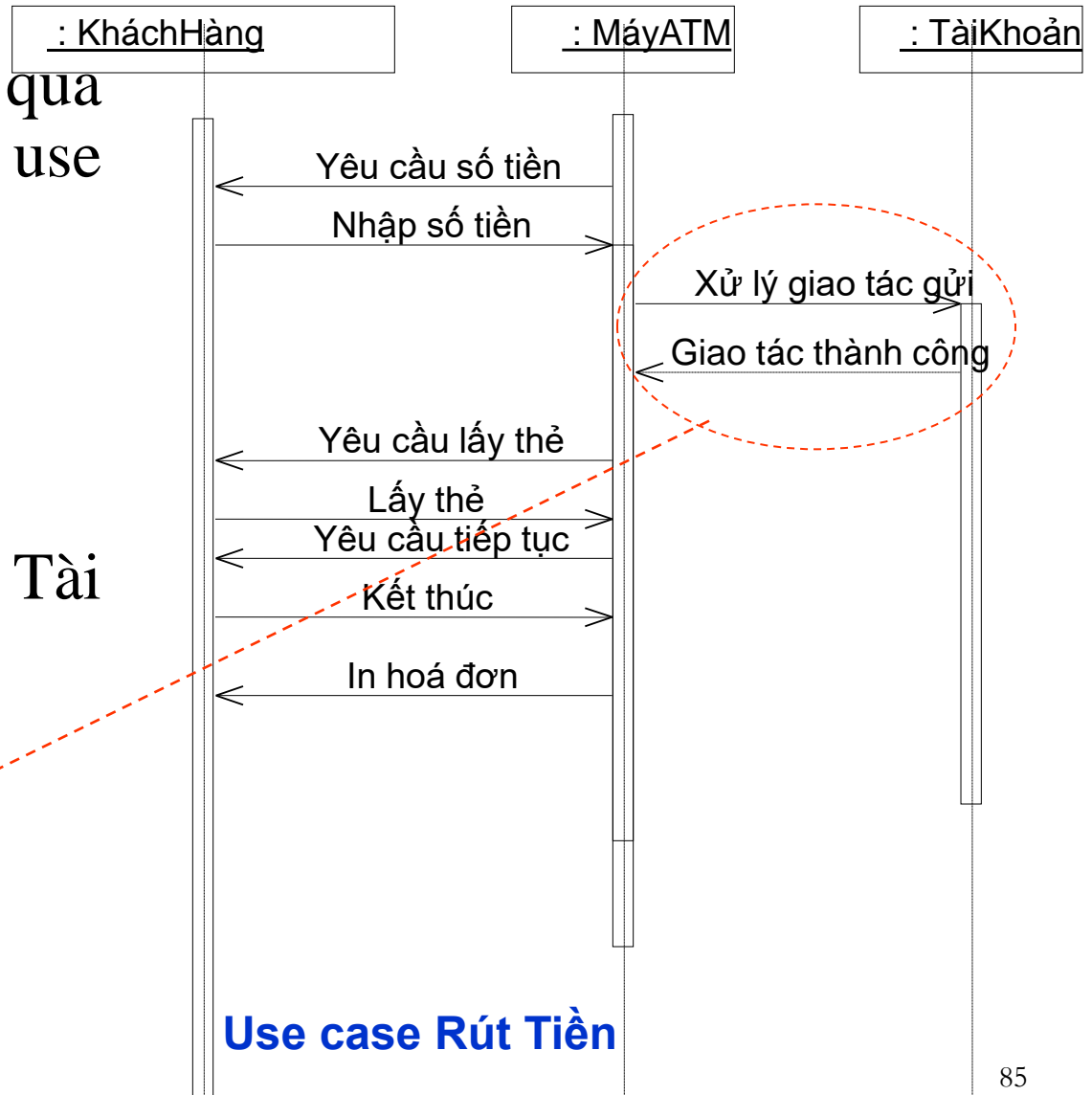
Xác định phương thức thông qua phân tích hoạt động use case

- Xác định method qua phân tích hoạt động use case:

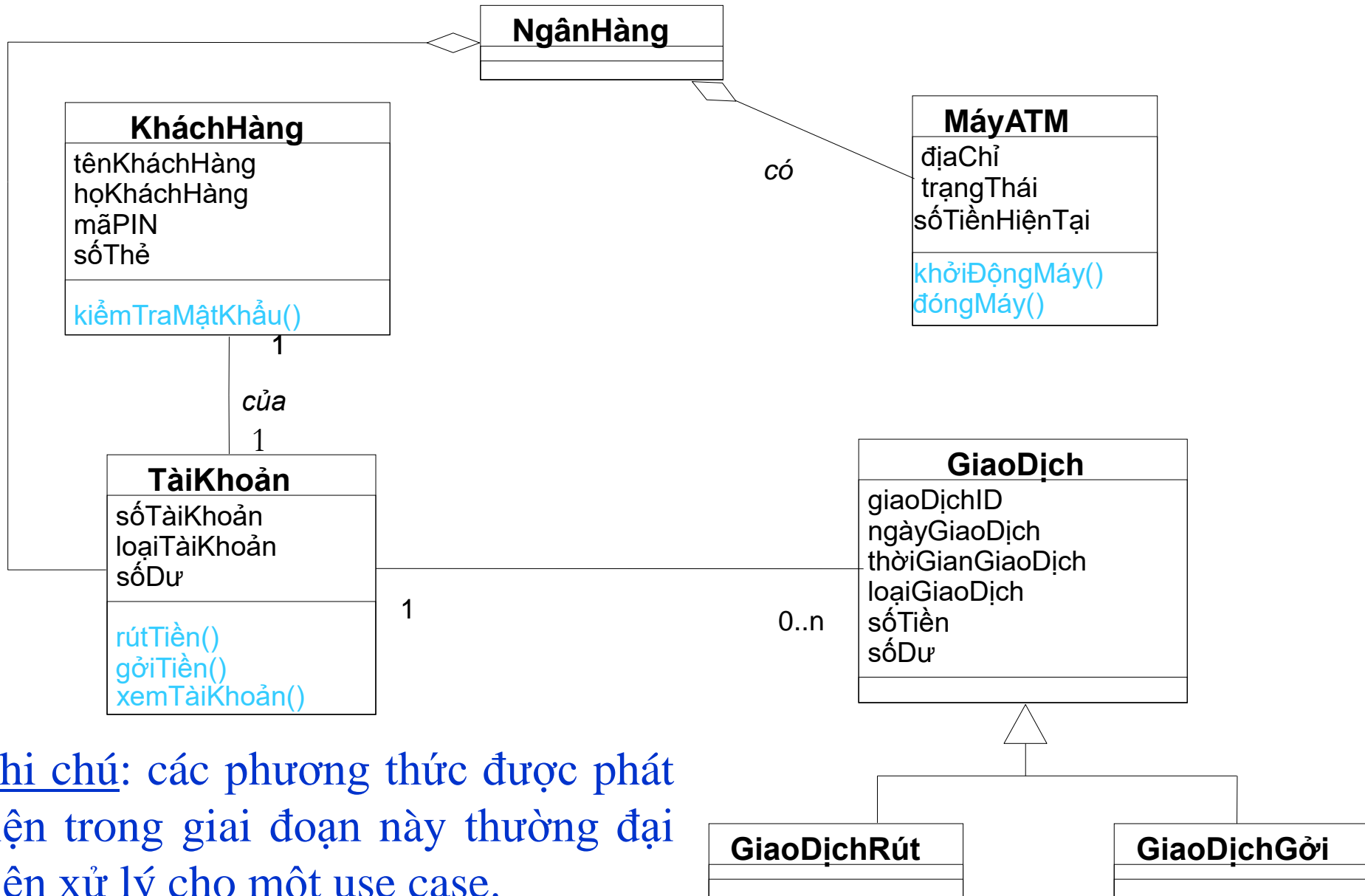
- Ví dụ: lớp TàiKhoản

- Rút Tiền
- Gửi Tiền
- Xem thông Tin Tài Khoản

TàiKhoản
sốTàiKhoản loạiTàiKhoản sốDư
rútTiền() gửiTiền() xemTàiKhoản()



Xác định phương thức: hệ thống ATM



Ghi chú: các phương thức được phát hiện trong giai đoạn này thường đại diện xử lý cho một use case.

Bài tập

- Xây dựng các sơ đồ lớp cho các bài toán trong phần bài tập