

PHÂN TÍCH THIẾT KẾ HỆ THỐNG THÔNG TIN

Information System Analysis and Design

Số tín chỉ: 4(3,1)

Số tiết: 75 tiết (45 LT- 30 TH)

Giảng viên: TS. Đinh Thị Thu Hương.

(Mobile: 0903087599 – e-mail: huongdtt2011@gmail.com)

Bộ môn: Khoa học về máy tính – Khoa Công nghệ Thông tin, SGU.

Chương 2:

MÔ HÌNH VÀ CÁC PHƯƠNG PHÁP MÔ HÌNH HÓA

- **Các phương tiện dùng trong quá trình xây dựng HTTT**
- Quy trình/Tiến trình phát triển hệ thống
- Lịch sử các phương pháp mô hình
- Một số mô hình tiêu biểu

2.1 Các phương tiện dùng trong quá trình xây dựng HTTT

1. Mô hình: Mô hình là một tập hợp các phần tử thường được dùng trong phép tương ứng với những lớp các đối tượng, các quan hệ nào đó trong lĩnh vực cần mô tả để có một sự biểu diễn cô đọng, tổng quát, có ý nghĩa, đơn giản và dễ hiểu. Mô hình được dùng biểu diễn dữ liệu và qui trình xử lý ở mức quan niệm.

2. Phương pháp: Phương pháp là cách thức tiếp cận để tìm hiểu và phát triển HTTT.

3. Công cụ:

Công cụ thủ công: văn bản, lưu đồ ngữ cảnh để mô tả, tổng hợp các kết quả điều tra → Dùng các giai đoạn đầu.

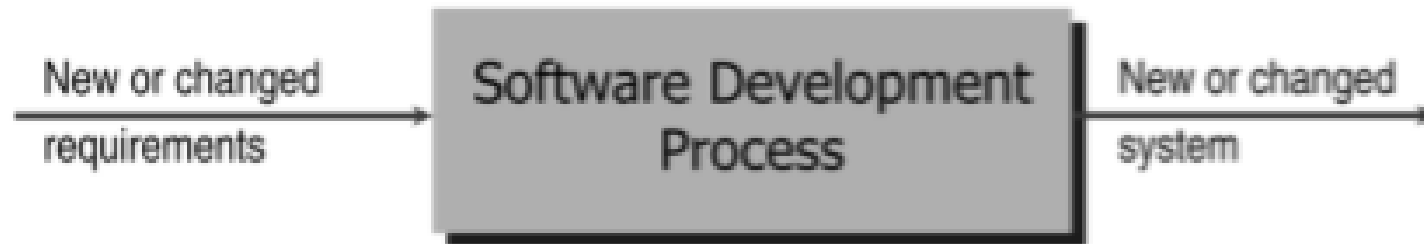
Công cụ tin học:

- **Phần mềm lập kế hoạch:** giai đoạn lập kế hoạch (Microsoft Project).
- **Phần mềm thiết kế:** giai đoạn thiết kế (Power Designer, Erwin, Designer 2000 ORACLE). Trong đó có các chức năng trợ giúp.
 - Thiết kế dữ liệu.
 - Thiết kế xử lý.
 - Thiết kế giao diện.
- **Các hệ quản trị CSDL, các ngôn ngữ lập trình:** giai đoạn lập trình, thử nghiệm và bảo trì.

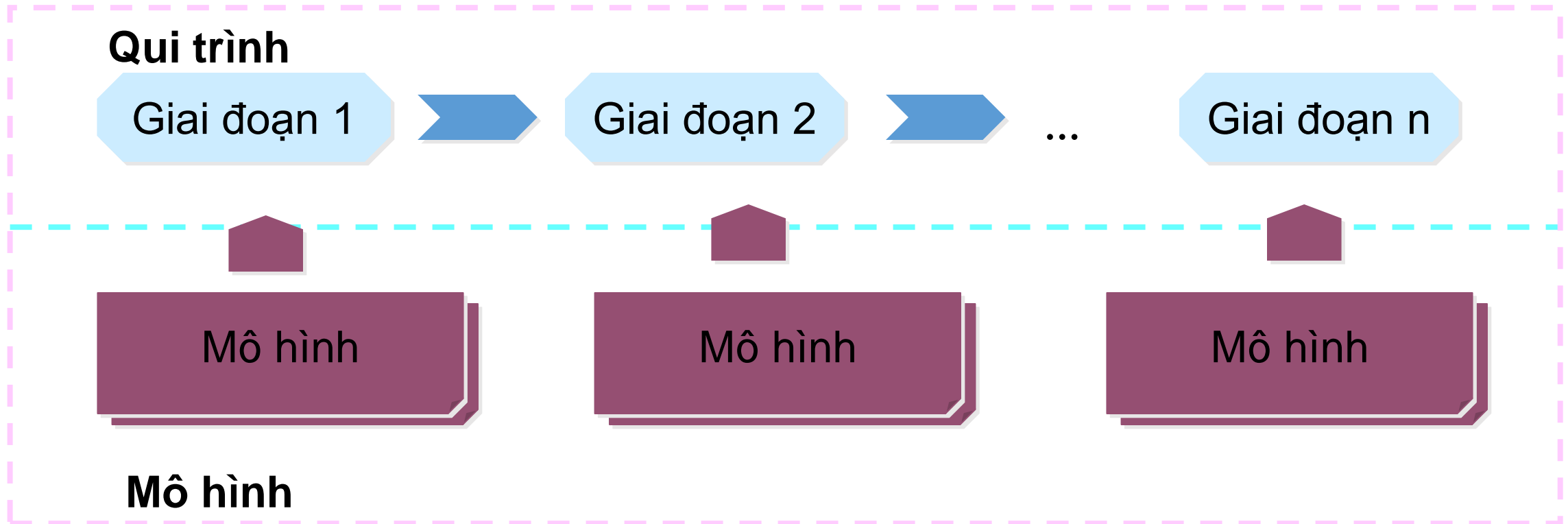
2.2 Qui trình/Tiến trình phát triển hệ thống

Tiến trình (process) là xác định ai (who) làm gì (what) và làm khi nào (when) và làm thế nào (how) để đạt tới mục đích mong muốn.

Tiến trình phát triển hệ thống phần mềm (Software Development Process – SDP) là tiến trình xây dựng sản phẩm phần mềm hay nâng cấp phần mềm đang có. Trong đó, mô tả tập các hoạt động cần thiết để chuyển đổi từ yêu cầu của người sử dụng sang hệ thống phần mềm.



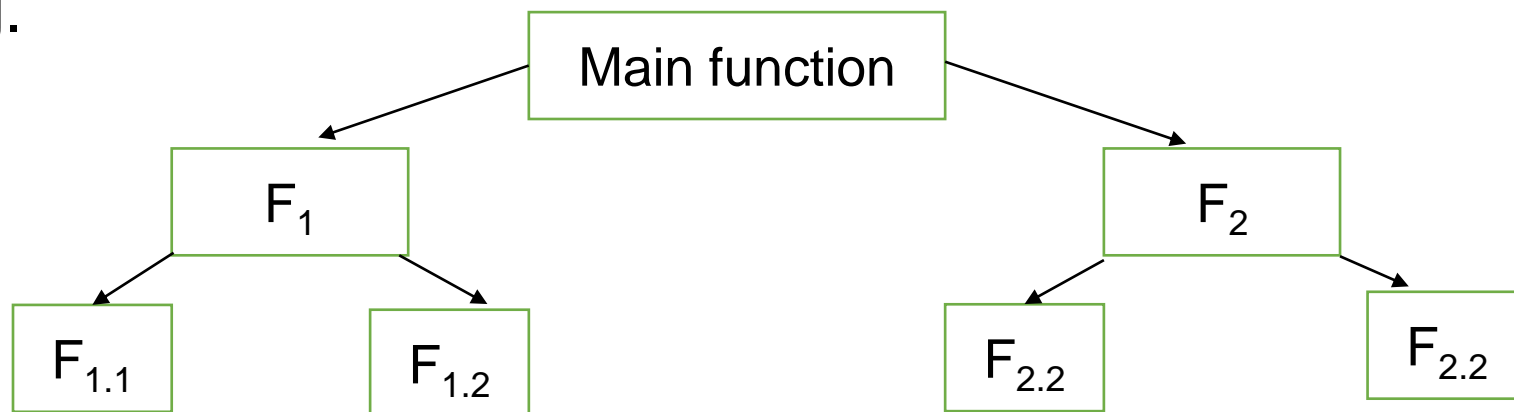
Quy trình/Tiến trình và mô hình



2.3 Một số phương pháp phát triển hệ thống

➤ Phương pháp hướng chức năng (Functional Oriented)

- ✓ Dựa vào chức năng, nhiệm vụ là chính
- ✓ Phân rã chức năng và làm mịn dần theo cách từ trên xuống (top – down)
 - Bị ảnh hưởng bởi các ngôn ngữ lập trình ALGOL, Pascal, C
 - Các hàm của hệ thống phần mềm được xem như tiêu chí cơ sở khi phân rã
- ✓ Tách chức năng khỏi dữ liệu.
- ✓ Các đơn thể chức năng trao đổi với nhau bằng cách truyền tham số hay sử dụng dữ liệu chung.



- ✓ Tiến trình phát triển tập trung vào thông tin mà hệ thống quản lý
 - Người phát triển hệ thống hỏi người sử dụng cần thông tin gì
 - Thiết kế CSDL để lưu trữ thông tin
 - Xây dựng màn hình nhập liệu
 - Hiển thị báo cáo
- ✓ Chỉ tập trung vào thông tin, ít quan tâm đến cái gì thực hiện với thông tin hay hành vi hệ thống
- ✓ Tiệm cận này gọi là tiệm cận hướng dữ liệu
 - Đã được áp dụng nhiều năm và tạo ra hàng ngàn hệ thống.
 - Thuận tiện cho thiết kế CSDL.
 - Bất tiện cho xây dựng các hệ thống tác nghiệp: yêu cầu hệ thống thay đổi theo thời gian.

Các hạn chế của tiếp cận hướng chức năng

- ✓ Sản phẩm hình thành từ giải pháp này khó bảo trì.

Mọi chức năng đều chia sẻ khối dữ liệu lớn.

Các chức năng phải hiểu rõ dữ liệu được lưu trữ thế nào.

Khi thay đổi cấu trúc dữ liệu kéo theo thay đổi mọi hàm liên quan.

- ✓ Tiến trình phát triển không ổn định

Thay đổi yêu cầu kéo theo thay đổi các chức năng.

Rất khó bảo toàn kiến trúc thiết kế ban đầu khi hệ thống tiến hóa.

- ✓ Tiếp cận này không hỗ trợ lập trình bằng ngôn ngữ hướng đối tượng như C++, Java, Smalltalk,...

➤ Phương pháp hướng đối tượng

✓ Mục tiêu: đưa ra những sản phẩm phần mềm tin cậy, dễ mở rộng, dễ thích nghi, phù hợp với các yêu cầu của khách hàng.

✓ Các đặc trưng

- Đặt trọng tâm vào dữ liệu (thực thể).
- Xem hệ thống như là một tập các thực thể, các đối tượng.
- Các lớp đối tượng trao đổi với nhau bằng các thông điệp.
- Đảm bảo tính mở và thích nghi.
- Hỗ trợ sử dụng lại và cơ chế kế thừa.

✓ Các nguyên tắc hướng đối tượng

- Tính kế thừa.
- Tính đa hình.
- Tính đóng gói.
- Tính trừu tượng.

2.4 Một số kiểu phát triển hệ thống

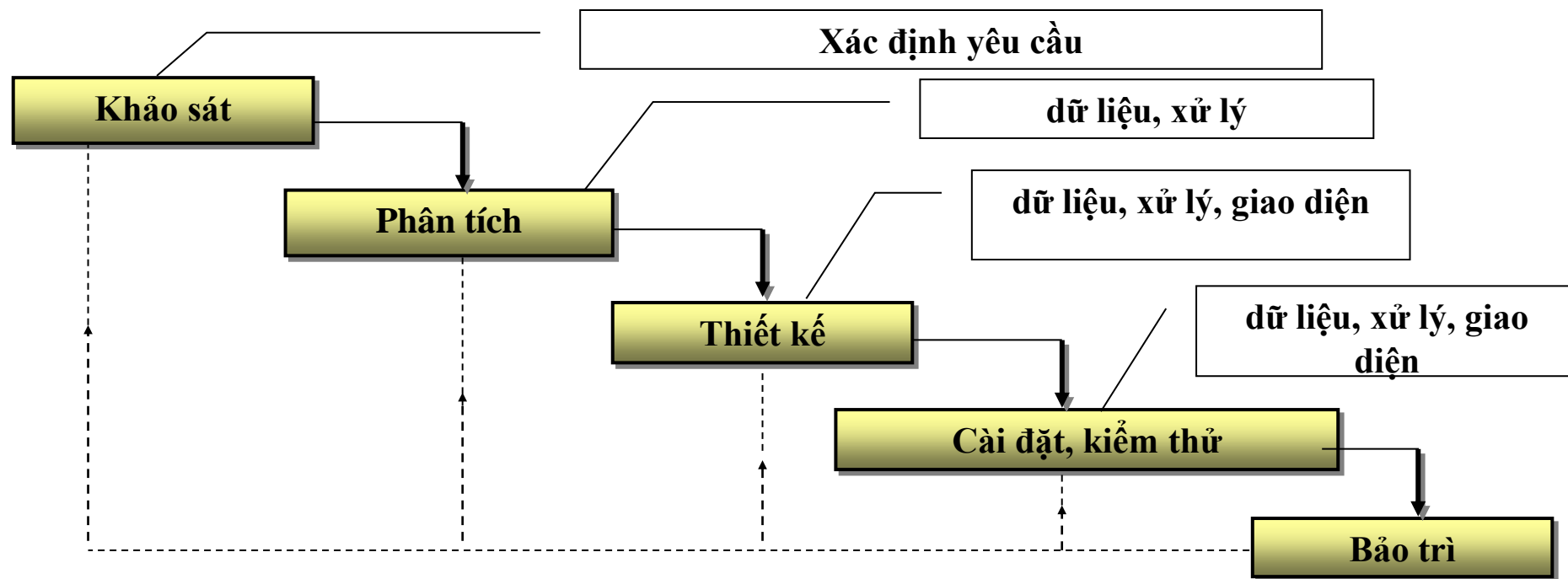
- Phân tích thiết kế có cấu trúc (*Structured Design*)
- Quy trình phát triển nhanh (Rapid Application Development - RAD)
- Hướng lập trình ứng dụng (Agile Development)

2.4 Một số kiểu phát triển hệ thống

➤ Phân tích thiết kế có cấu trúc (*Structured Design*)

- Thực hiện tuần tự các bước trong quy trình phát triển hệ thống.
- Sử dụng các mô hình và biểu đồ kỹ thuật để mô hình hóa quy trình nghiệp vụ của hệ thống.
- Điển hình là Quy trình thác nước và tăng trưởng/song song.

- **Qui trình thác nước (Waterfall- Royce, 1970)**



➤ Các tính chất

- **Tính tuần tự:** Thứ tự giai đoạn được thực hiện từ trên xuống, kết quả của giai đoạn trước sẽ là đầu vào cho giai đoạn sau.
- **Tính lặp:** Mỗi giai đoạn có thể quay trở lại tới các giai đoạn trước đó nếu cần thiết cho đến khi kết quả của nó được chấp nhận.
- **Tính song song:** Nhiều hoạt động trong một giai đoạn có thể được thực hiện song song với các hoạt động của giai đoạn khác.

Các giai đoạn

➤ **Giai đoạn khảo sát**

- Tìm hiểu thực tế.
- Nắm bắt những yêu cầu của người sử dụng.
- Lập kế hoạch triển khai.
- Đối tượng tham gia:
 - Những người chịu trách nhiệm triển khai HTTT (phía khách hàng).
 - Nhóm quản lý dự án (phía công ty phát triển).
 - Nhân viên nghiệp vụ (người sử dụng).
 - Chuyên viên tin học (người khảo sát).

Các giai đoạn

➤ **Giai đoạn phân tích**

- Mô tả lại thực tế thuộc phạm vi ứng dụng HTTT ở mức quan niệm, cấu trúc hóa yêu cầu.
 - Thành phần dữ liệu
 - Thành phần xử lý
- Phát sinh các phương án và lựa chọn phương án khả thi nhất.
- Giai đoạn phân tích độc lập với môi trường cài đặt ứng dụng HTTT.
- Đối tượng tham gia
 - Nhân viên nghiệp vụ (người sử dụng)
 - Chuyên viên tin học (chuyên viên phân tích, thiết kế)
 - Nhóm quản lý dự án (tổ chức, kế hoạch hóa, ...)

Các giai đoạn

➤ **Giai đoạn thiết kế**

- ✓ Mô hình hóa thành phần dữ liệu và xử lý ở mức.
 - Thiết kế luận lý (tổ chức logic)
 - Thiết kế dữ liệu
 - Thiết kế kiến trúc
 - Thiết kế giao diện
 - Thiết kế vật lý: chuyển đổi thiết kế luận lý sang các đặc tả phần cứng, phần mềm, kỹ thuật được chọn để cài đặt hệ thống
- ✓ Liên quan đến việc sử dụng một số công cụ tin học hỗ trợ cho chuyên viên tin học trong quá trình thiết kế.
- ✓ Đối tượng tham gia:
 - Nhóm quản lý dự án
 - Chuyên viên tin học (chuyên viên phân tích, thiết kế)

➤ **Giai đoạn cài đặt, thử nghiệm**

- Lập trình hệ thống.
- Kiểm tra những chức năng, phân hệ, sự kết hợp của những phân hệ khác nhau, tổng thể cả hệ thống thông tin.
- Thử nghiệm.
- Xây dựng tài liệu hệ thống: tài liệu đặc tả hệ thống, tài liệu sử dụng, tài liệu kỹ thuật cài đặt.
- Huấn luyện sử dụng.
- Đối tượng tham gia:
 - Nhóm quản lý dự án
 - Chuyên viên tin học (Lập trình viên, nhân viên kiểm tra chương trình, ..)

➤ **Giai đoạn khai thác, bảo trì**

- Đảm bảo duy trì hoạt động ổn định của hệ thống thông tin tin học hóa.
- Sửa các lỗi phát sinh trong quá trình sử dụng.
- Điều chỉnh những thay đổi sao cho phù hợp với các thay đổi hệ thống.
- Bổ sung, nâng cấp hệ thống mới.
- Đối tượng tham gia:
 - Nhóm quản lý dự án.
 - Những người khai thác.
 - Chuyên viên tin học (Lập trình viên, chuyên viên phân tích, thiết kế, nhân viên kiểm tra, ..).

2.4 Một số kiểu phát triển hệ thống

➤ Quy trình phát triển nhanh (Rapid Application Development - RAD)

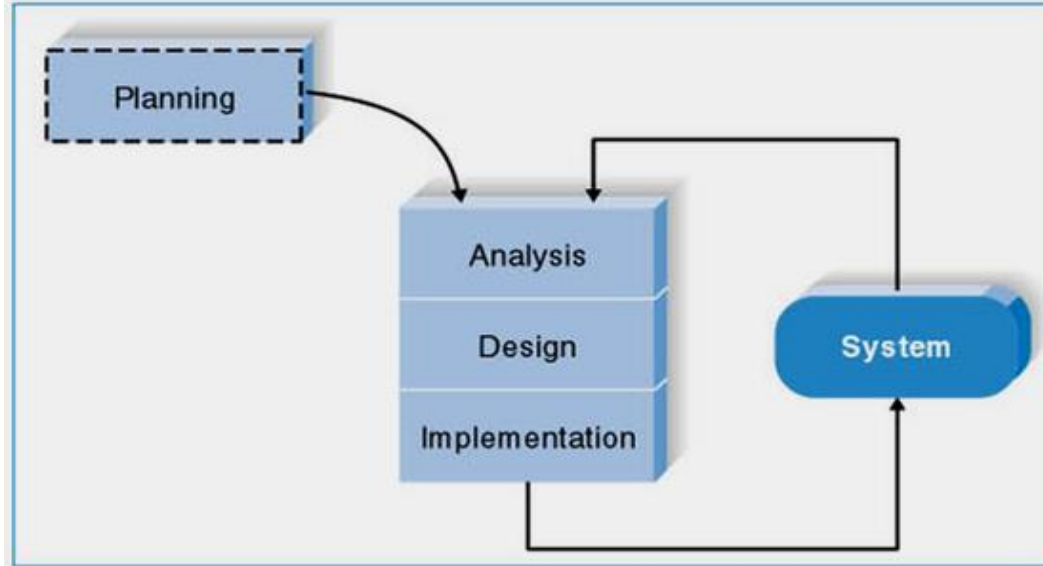
- Thực hiện phát triển từng phần hệ thống với mục đích chuyển giao cho người dùng sớm
- Cần sử dụng các kỹ thuật và công cụ để tăng tốc quá trình phân tích, thiết kế và cài đặt (vd: CASE – computer-aided software engineering).
- Điển hình là Quy trình xoắn ốc và làm bản mẫu.

2.4 Một số kiểu phát triển hệ thống

➤ Hướng lập trình ứng dụng (Agile Development)

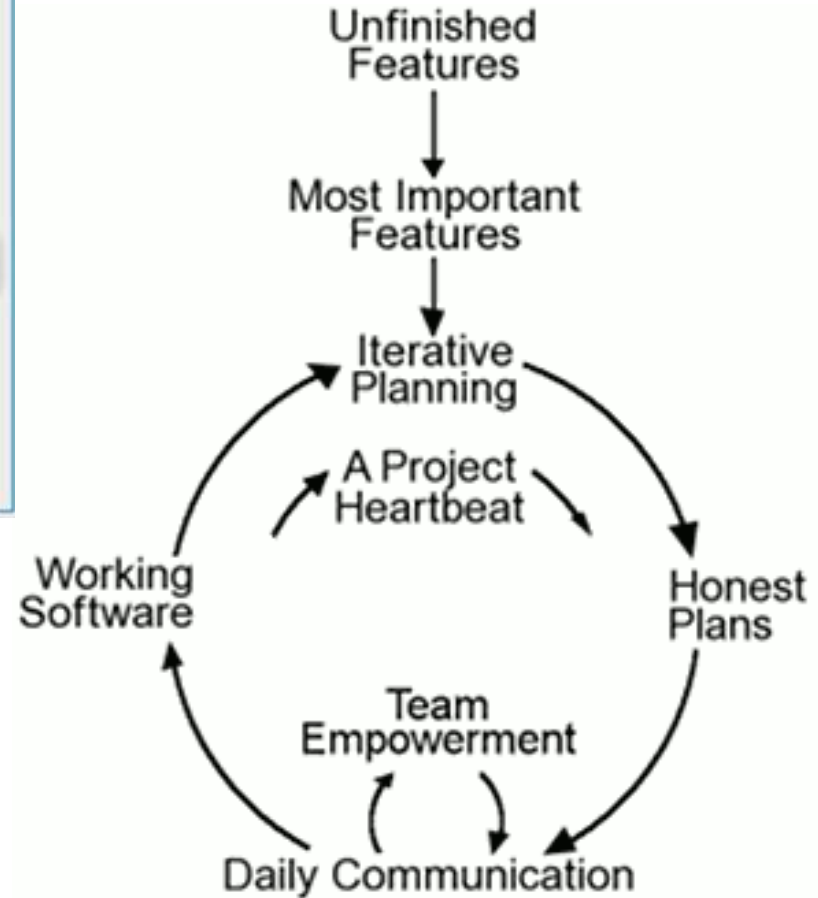
- Tập trung vào bước của chu trình phát triển và hạn chế việc sử dụng mô hình hóa, xây dựng tài liệu của các bước trung gian
- Phát triển ứng dụng đơn giản, lặp đi lặp lại.
- Điển hình: eXtreme-Programming (XP)

eXtreme-Programming



Các giá trị của XP

- Trao đổi thông tin (communication)
- Đơn giản (simplicity)
- Phản hồi (feedback)
- Thể mạnh (courage)



Đặc điểm

- Tương tác liên tục.
- Thiết kế đơn giản, sử dụng các nguyên lý và dạng thức thiết kế chung.
- Nhóm làm việc: người lập trình, khách hàng, người quản trị - khách hàng trực diện

2.5 Mô hình hóa hệ thống

Khái niệm

- Mô hình là một dạng trừu tượng hóa của một hệ thống thực
- Mô hình là một hình ảnh (một biểu diễn) của một hệ thống thực, được diễn tả:
 - + Ở một mức trừu tượng nào đó.
 - + Theo một quan điểm (hay một góc nhìn) nào đó.
 - + Bởi một hình thức diễn tả hiểu được (văn bản, phương trình, bảng, đồ thị,...) nào đó.
- Mô hình hóa là việc dung mô hình để nhận thức và diễn tả một hệ thống.

2.5 Mô hình hóa hệ thống

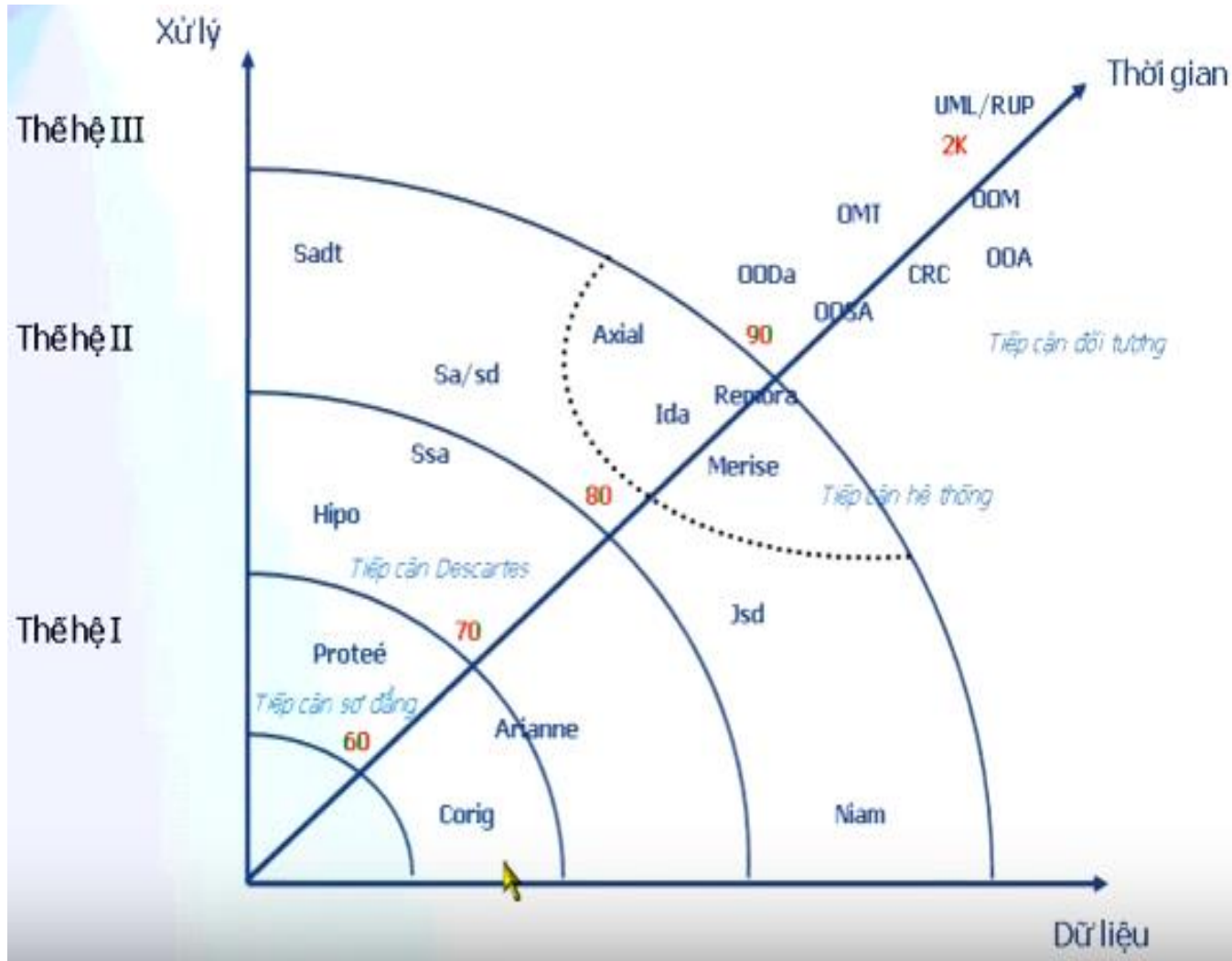
Mục đích

- Giúp hiểu và thực hiện được sự trừu tượng hóa, tổng quát hóa các khái niệm cơ sở nhằm giảm thiểu độ phức tạp của hệ thống.
- Giúp quan sát được hệ thống như nó vốn có và nó phải có.
- Giúp đặc tả được cấu trúc và hành vi của hệ thống.
- Giúp tạo khuôn mẫu và hướng dẫn cách xây dựng, thử nghiệm, mô phỏng, thực hiện, hoàn thiện theo mô hình.
- Là cơ sở để trao đổi

Mô hình hóa hệ thống phần mềm

- Mô hình hóa hướng chức năng lấy chức năng làm đơn vị phân rã hệ thống.
- Mô hình hóa hướng đối tượng lấy đối tượng làm đơn vị phân rã hệ thống..

2.6 Lịch sử các phương pháp mô hình hóa



Thập niên 60: Chuẩn hóa các kỹ thuật của các nhà phát triển ứng dụng.

Thập niên 70:

+ Phân rã các chức năng của HTTT theo mô hình phân cấp.

- Chia nhỏ công việc để xử lý.

+ Áp dụng phương pháp lập trình cấu trúc và đơn thể.

Thập niên 80:

+ Tiếp cận toàn diện

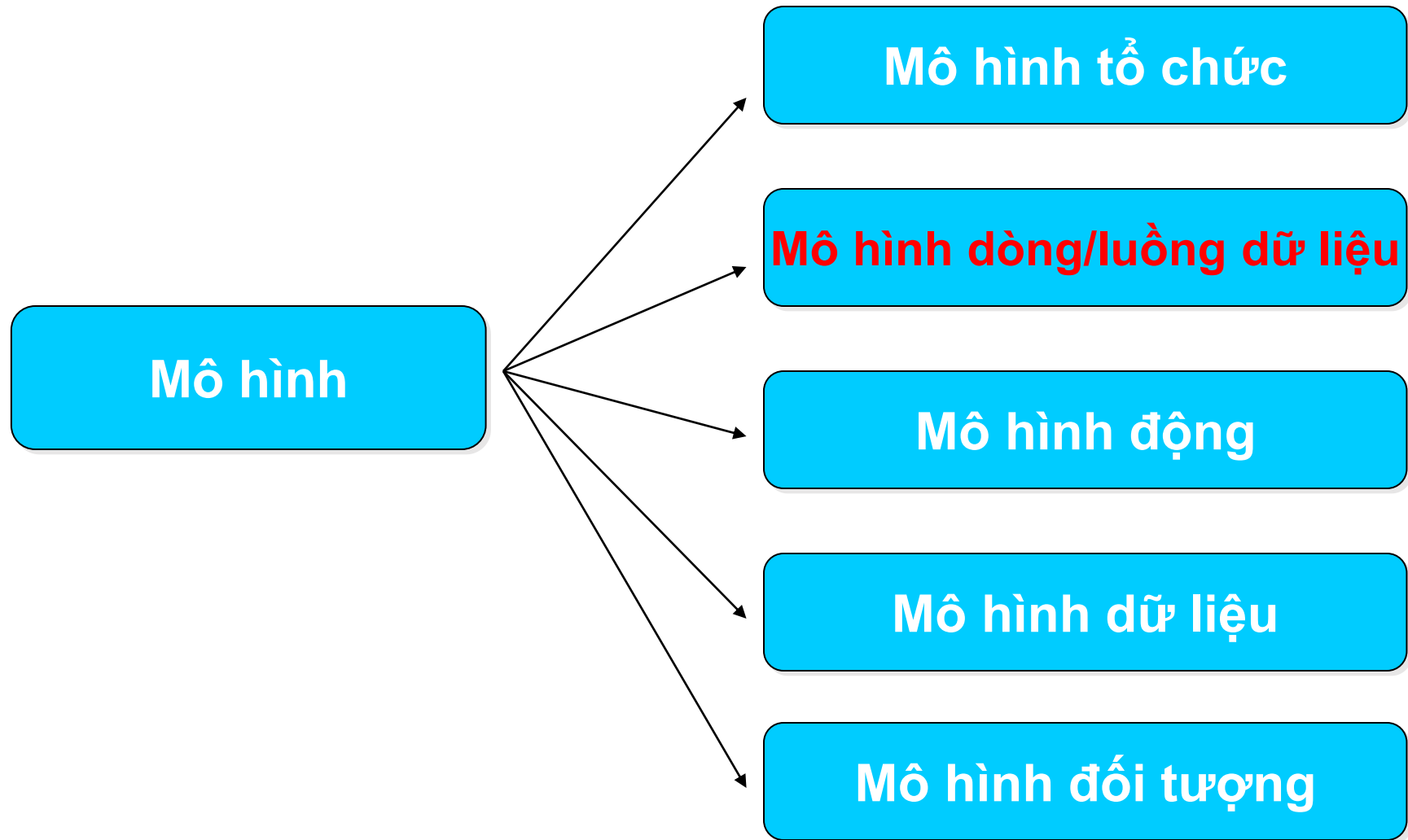
- Dữ liệu, xử lý, biến cố của HTTT.
- Phân rã có kết hợp các thành phần con

Thập niên 90:

+ Tổng hợp phương pháp ở thập niên 70 và 80.

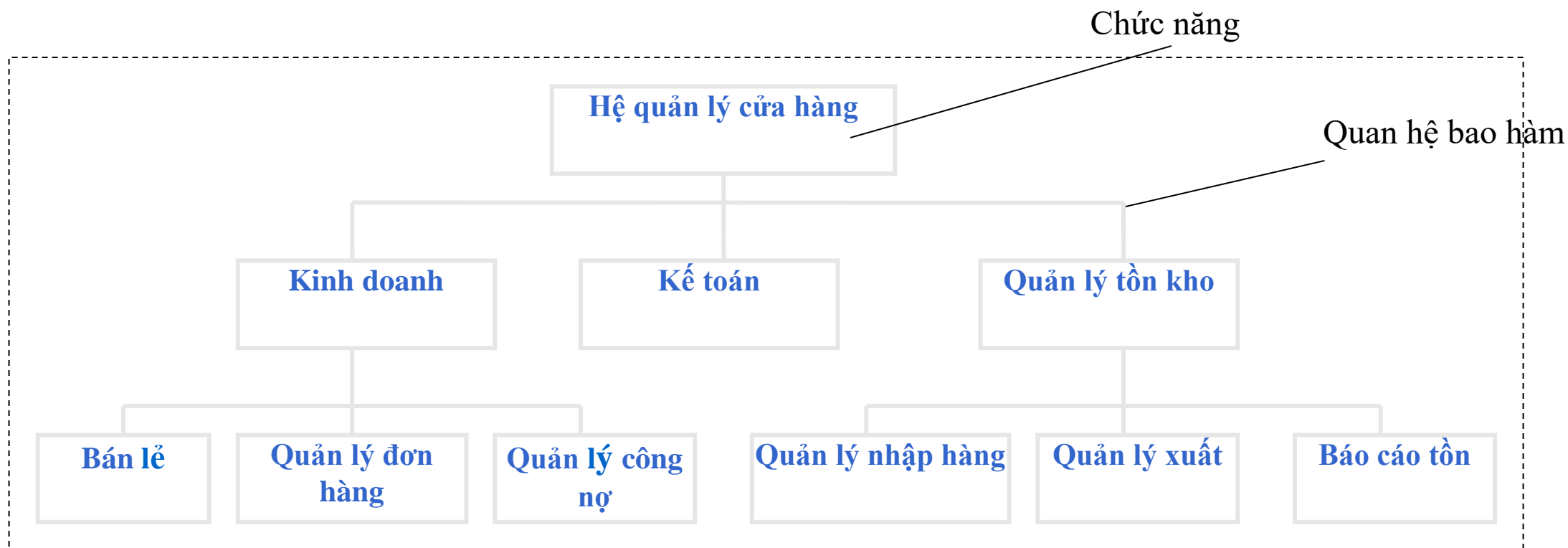
+ Đối tượng: gom nhóm cấu trúc dữ liệu và xử lý.

2.7 Một số mô hình tiêu biểu



2.7.1 Mô hình tổ chức

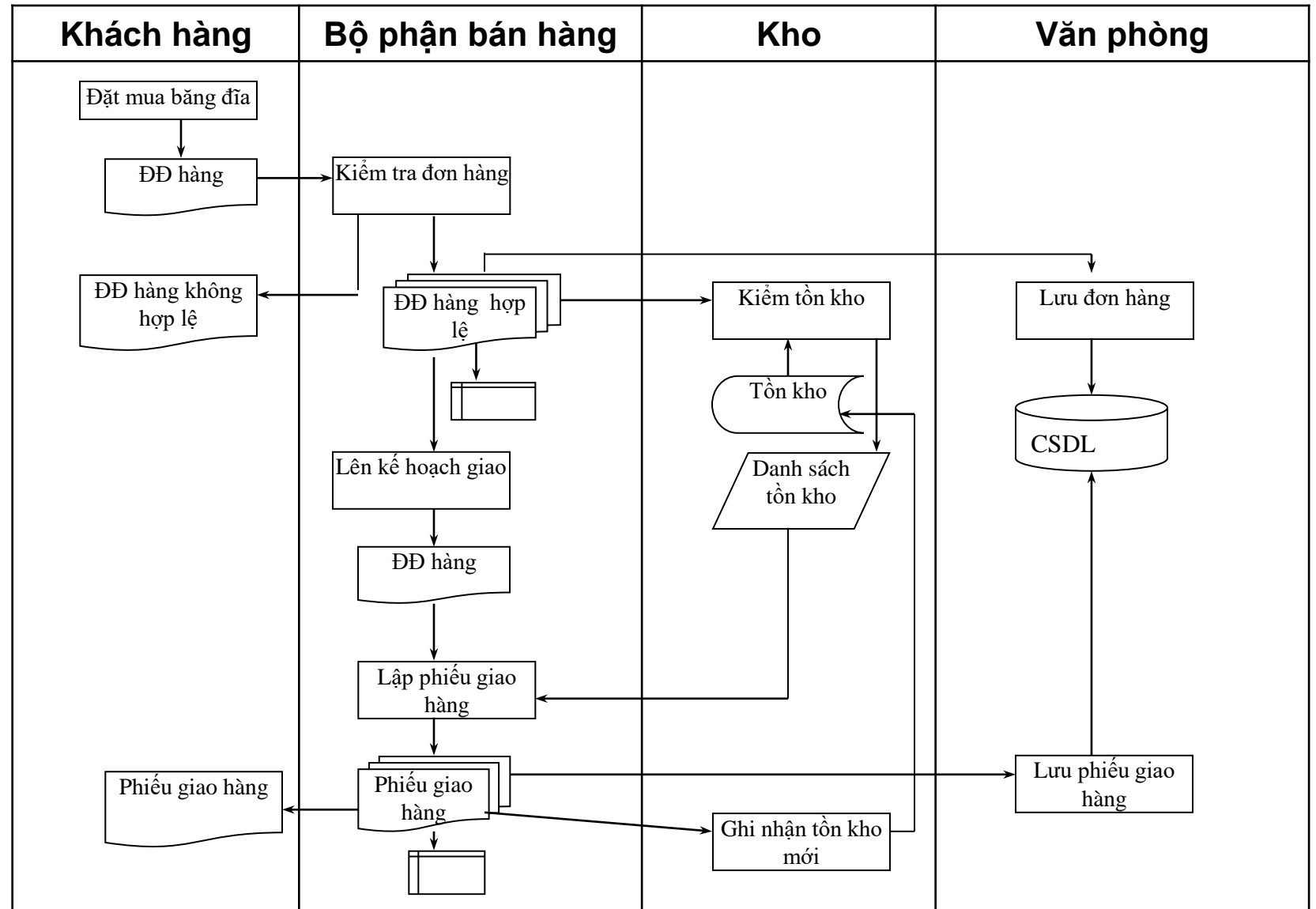
- **Mô hình phân cấp chức năng:** phân rã một chức năng tổng hợp thành những chức năng chi tiết hơn.



Ví dụ: biểu diễn các chức năng của hệ thống Đại lý băng đĩa ABC

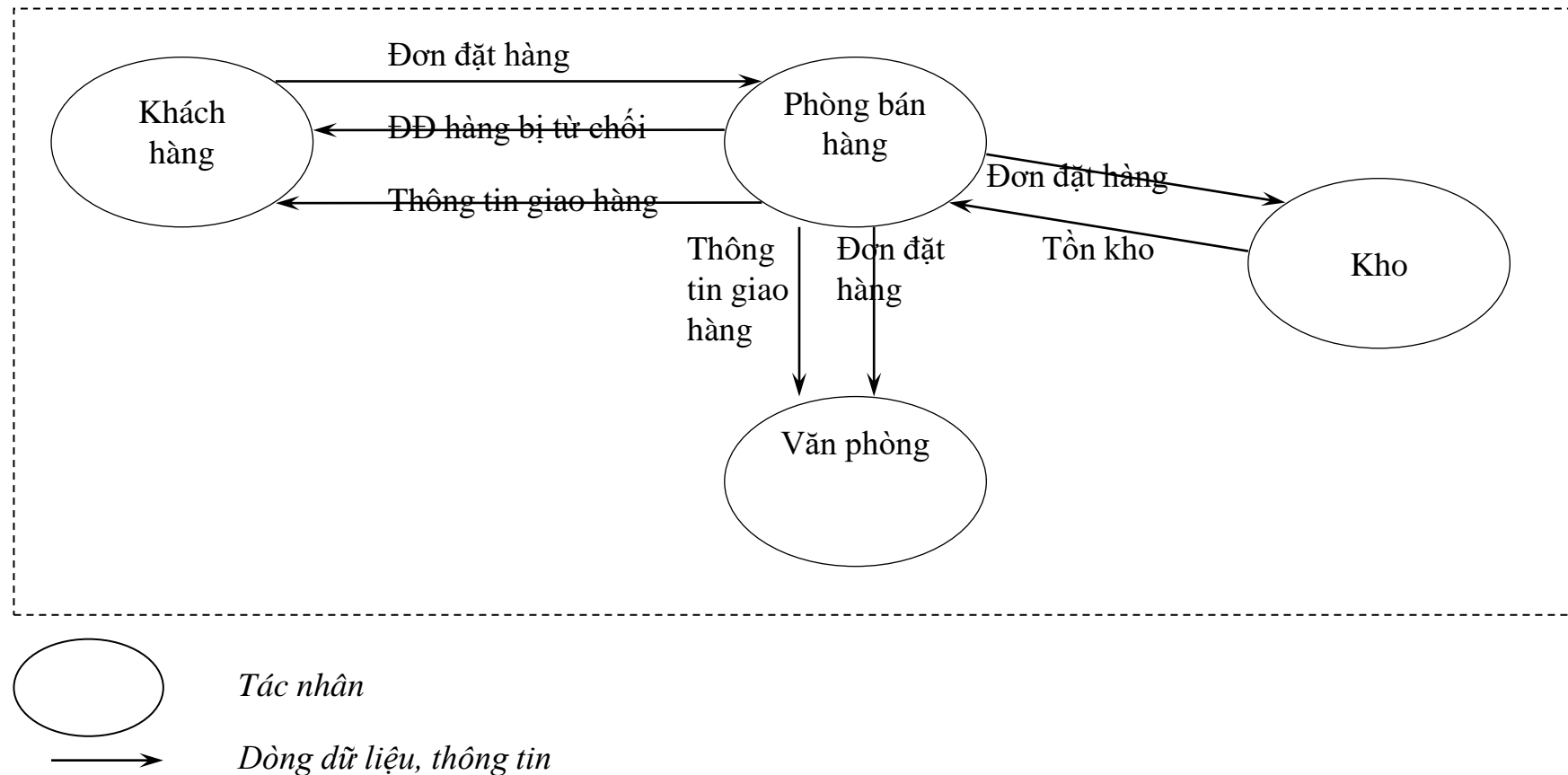
- **Mô hình luân chuyển:** diễn tả quá trình luân chuyển thông tin qua các không gian

Ví dụ: biểu diễn quá trình xử lý đặt hàng của hệ thống Đại lý băng đĩa ABC



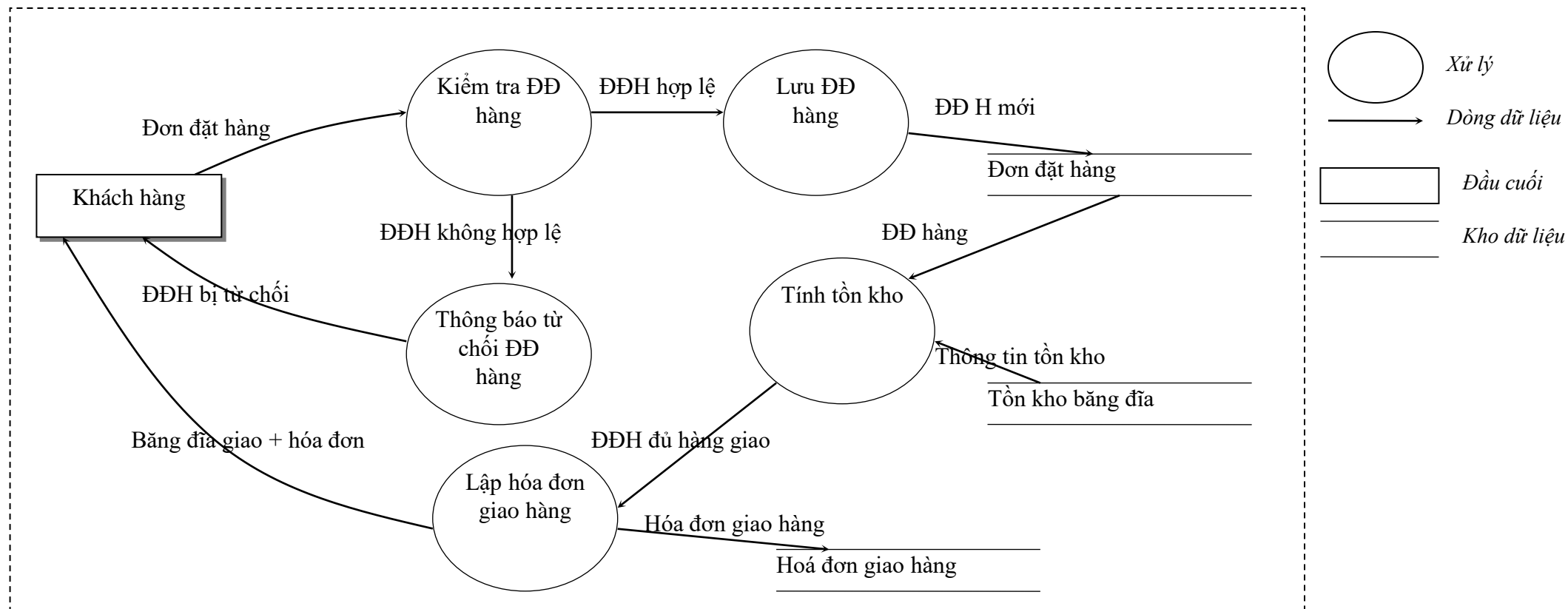
2.7.2 Mô hình dòng dữ liệu

- **Mô hình tương tác thông tin:** diễn tả dòng dữ liệu giữa các tác nhân



Ví dụ: Mô hình tương tác đặt hàng Công ty máy tính.

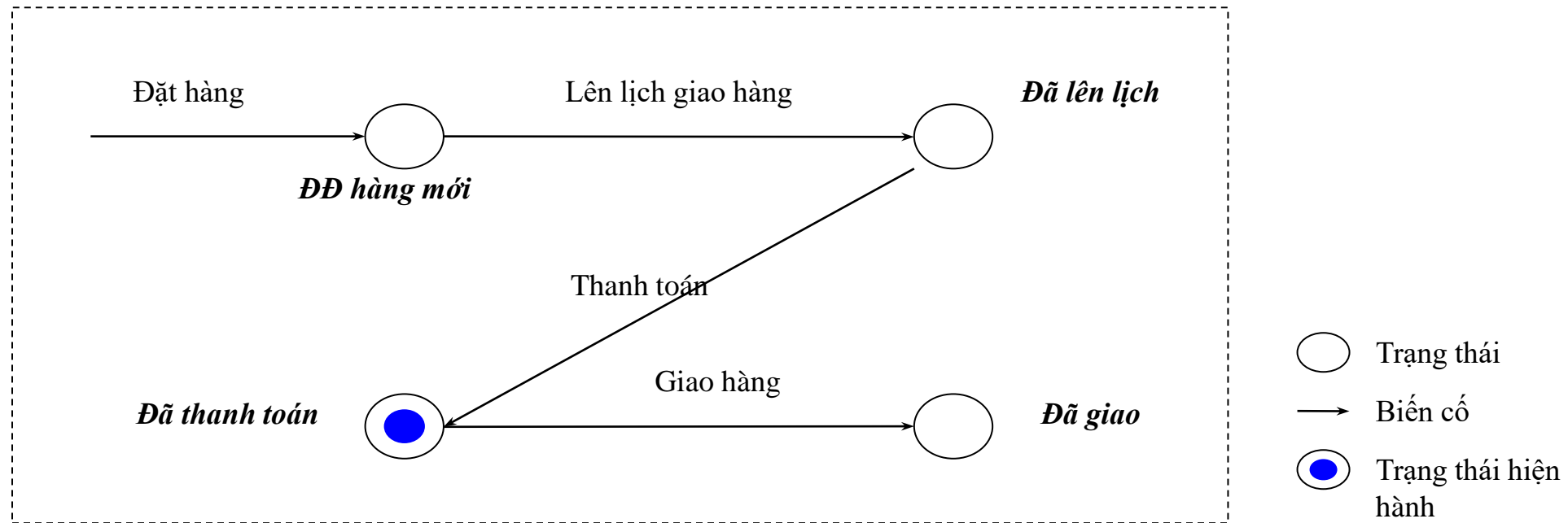
- **Mô hình dòng dữ liệu (Data flow diagram - DFD):** diễn tả dòng dữ liệu giữa các xử lý



Ví dụ: Biểu diễn mô hình DFD của xử lý đặt hàng Công ty máy tính.

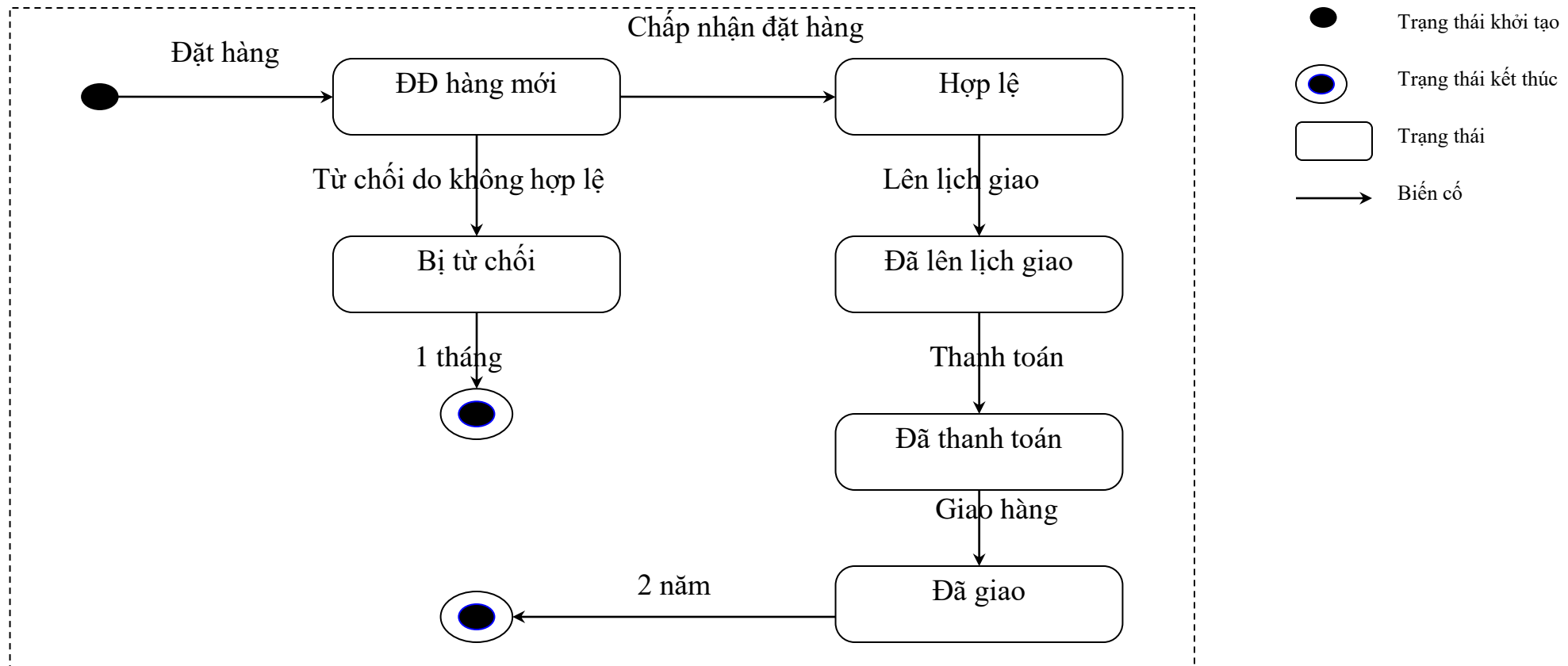
2.7.3 Mô hình động

- **Mô hình mạng Petri-net:** diễn tả biến cố và sự đồng bộ của biến cố



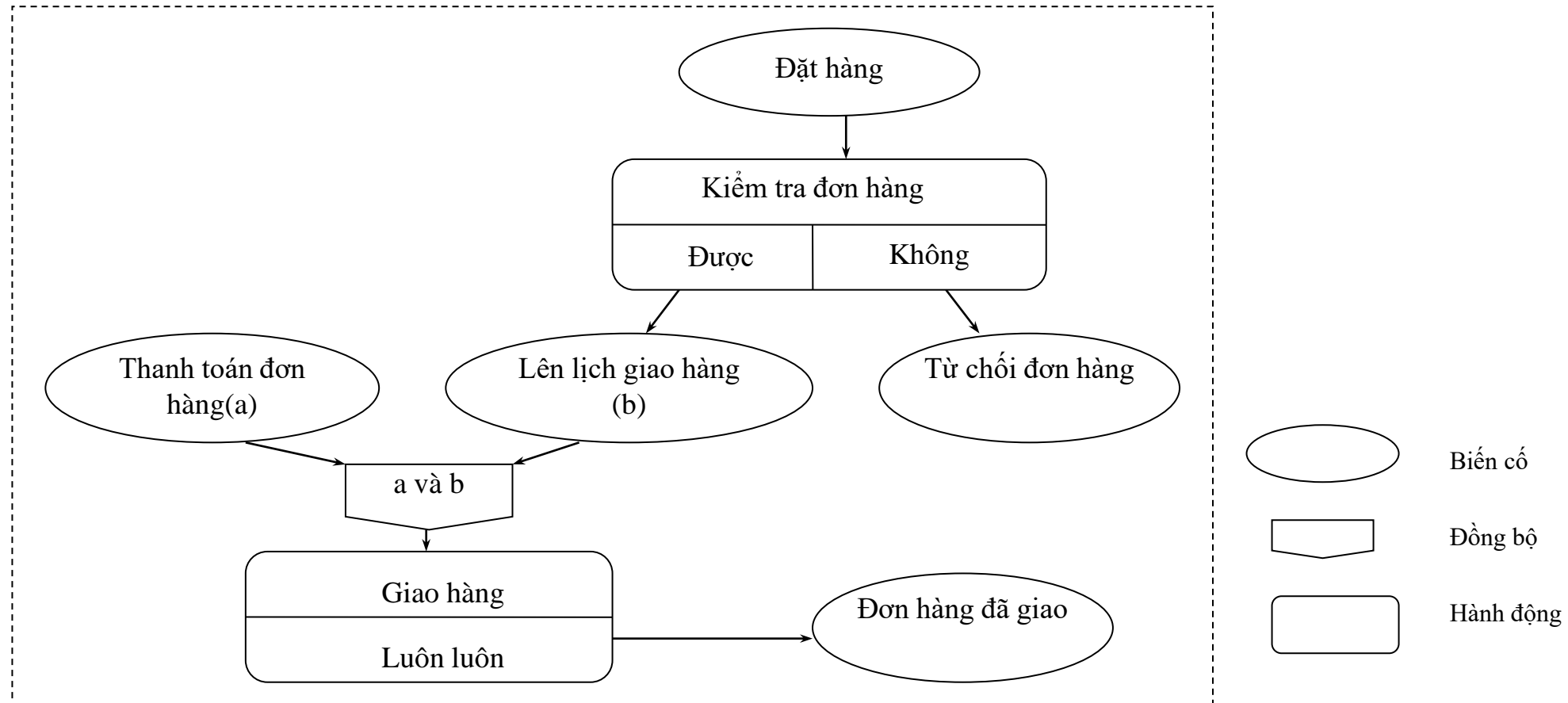
Ví dụ: các trạng thái của một đơn đặt hàng

■ Mô hình trạng thái: diễn tả biến cố và sự đồng bộ của biến cố



Ví dụ: trạng thái một đơn đặt hàng

- **Mô hình xử lý Merise:** diễn tả các quan niệm xử lý



Ví dụ: biểu diễn xử lý của một đơn đặt hàng

2.7.4 Mô hình dữ liệu

■ Mô hình dữ liệu quan hệ:

- Dựa trên lý thuyết tập hợp.
- Dễ dàng ánh xạ đến cấu trúc lưu trữ vật lý.
- Các khái niệm cơ bản:

- + Toán học: quan hệ, bộ, thuộc tính,...

- + Hướng dữ liệu: bảng, bản ghi, trường

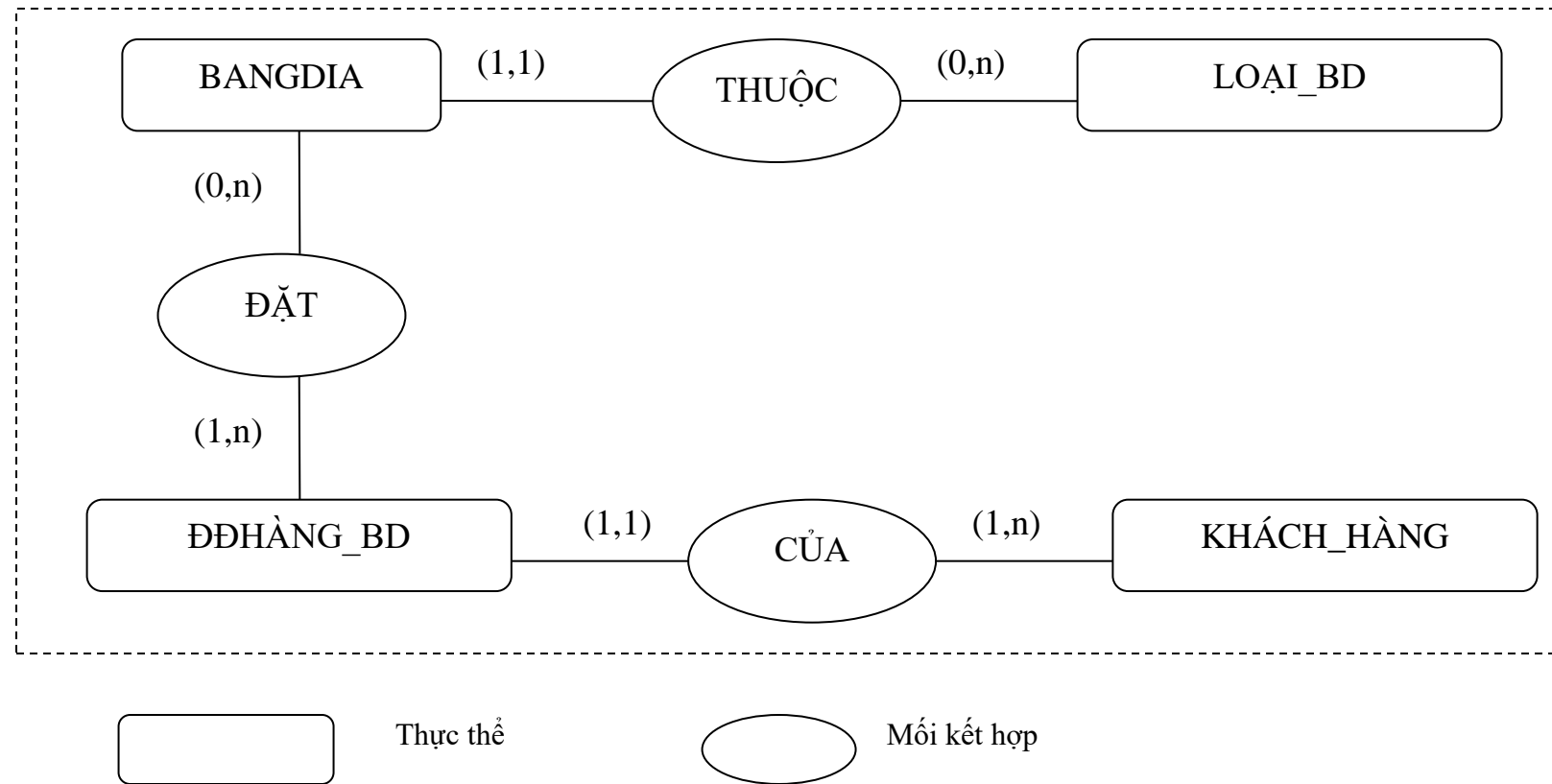
- Biểu diễn bởi lược đồ quan hệ như sau:

a/ BANGDIA(MA_BD, TEN_BD, LOAI, DVTINH, DON_GIA)

b/ ĐĐHANG_NGK(SO_DDH, NGÀY_DAT, KHACH_HANG, NGÀYGIAO, TRANG THAI)

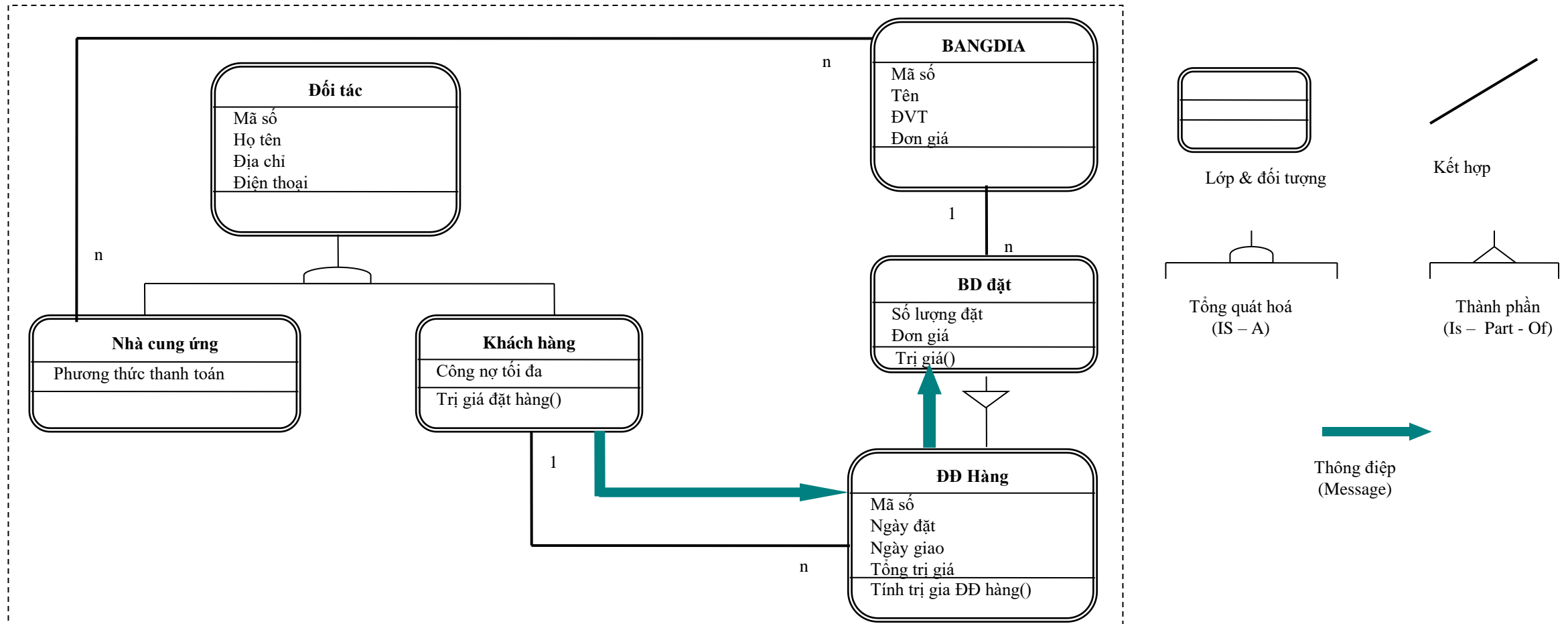
c/ CHITIET_DD(MA_BD, SO_DDH, SL_DAT, DONGIA_DAT)

- **Mô hình thực thể - kết hợp/liên kết:** diễn tả dữ liệu ở mức trừu tượng

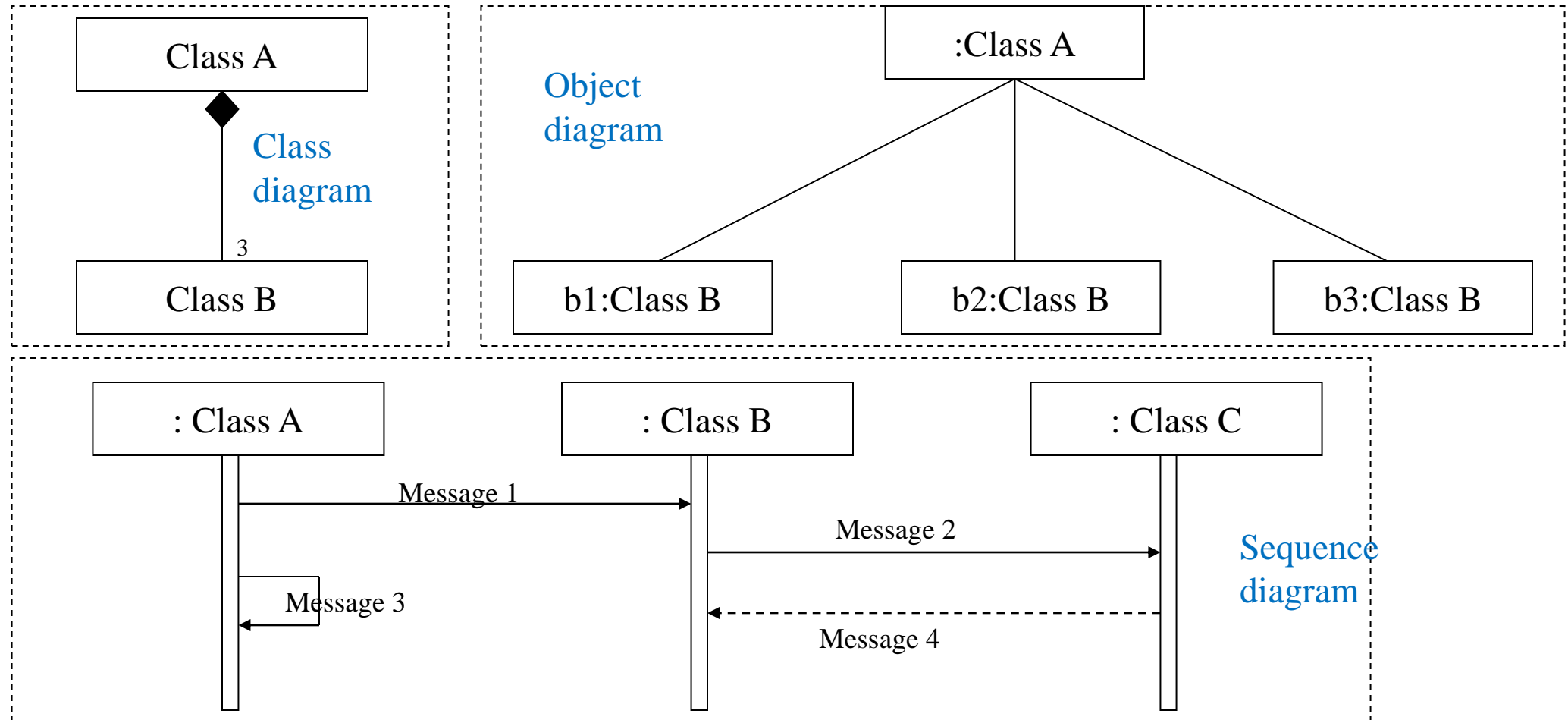


2.7.5 Mô hình đối tượng

■ Mô hình đối tượng theo OOA (Object Oriented Analysis)



UML (Unified Modeling Language)



■ Các khái niệm cơ bản của UML:

- Các góc nhìn
- Các biểu đồ

Câu 1. Các giai đoạn trung tâm trong quá trình phát triển 1 HTTT là:

- A. Khảo sát và phân tích.
- B. Phân tích và thiết kế.
- C. Thiết kế và lập trình.
- D. Lập trình và kiểm thử.

Câu 2. Mục đích của giai đoạn phân tích

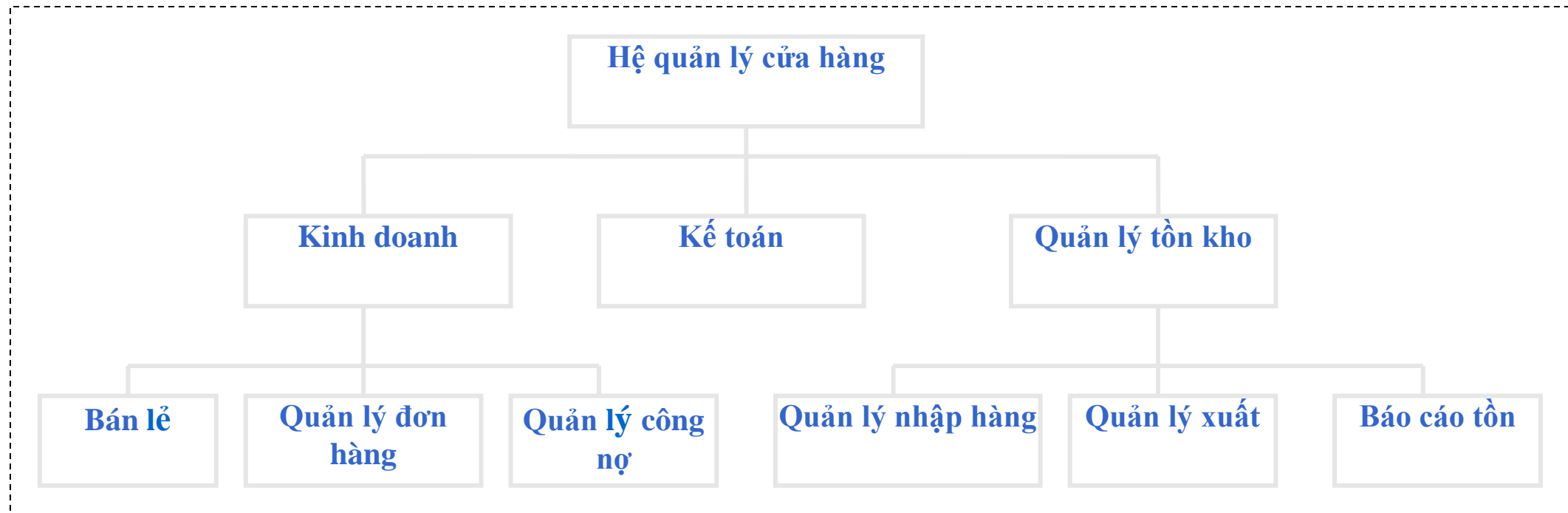
- a. Lên kế hoạch và tìm hiểu thông tin chức năng để thực hiện dự án.
- b. Lên mức thiết kế tổng thể cho dự án.
- c. Câu a và b.
- d. Không có câu nào đúng.

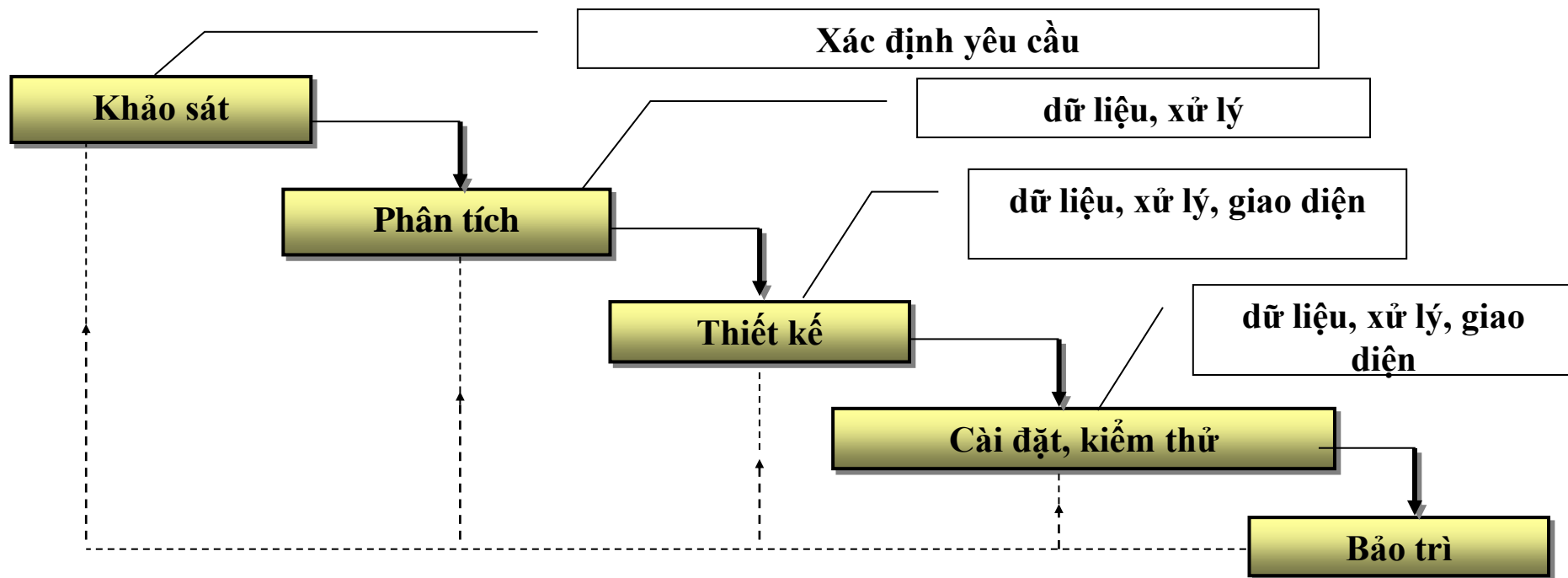
Câu 3. Mục đích của quá trình khảo sát hiện trạng nhằm để

- a. Tiếp cận với nghiệp vụ chuyên môn, môi trường hoạt động của hệ thống.
- b. Thương lượng với khách hàng để ký kết hợp đồng.
- c. Thiết kế hệ thống.
- d. Tất cả hệ thống.

BÀI THỰC HÀNH SỐ 2

Sử dụng phần mềm Power Designer/Microsoft visio vẽ mô hình phân cấp chức năng như sau:



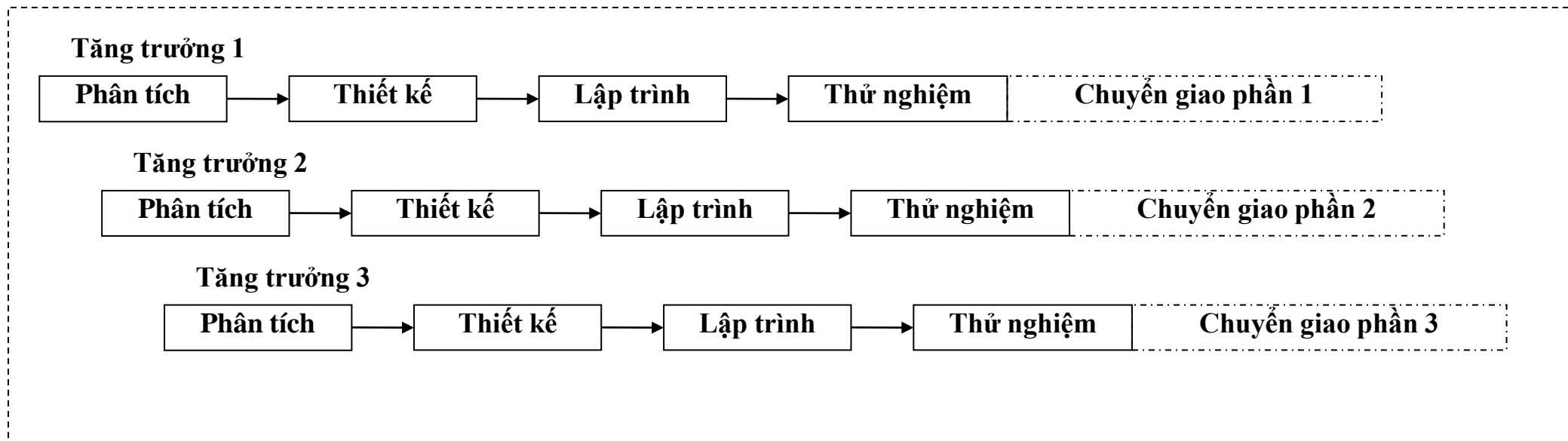


Tham khảo các qui trình phát triển hệ thống

Các qui trình phát triển hệ thống

▪ Qui trình tăng trưởng (D. R. Grahma, 1989)

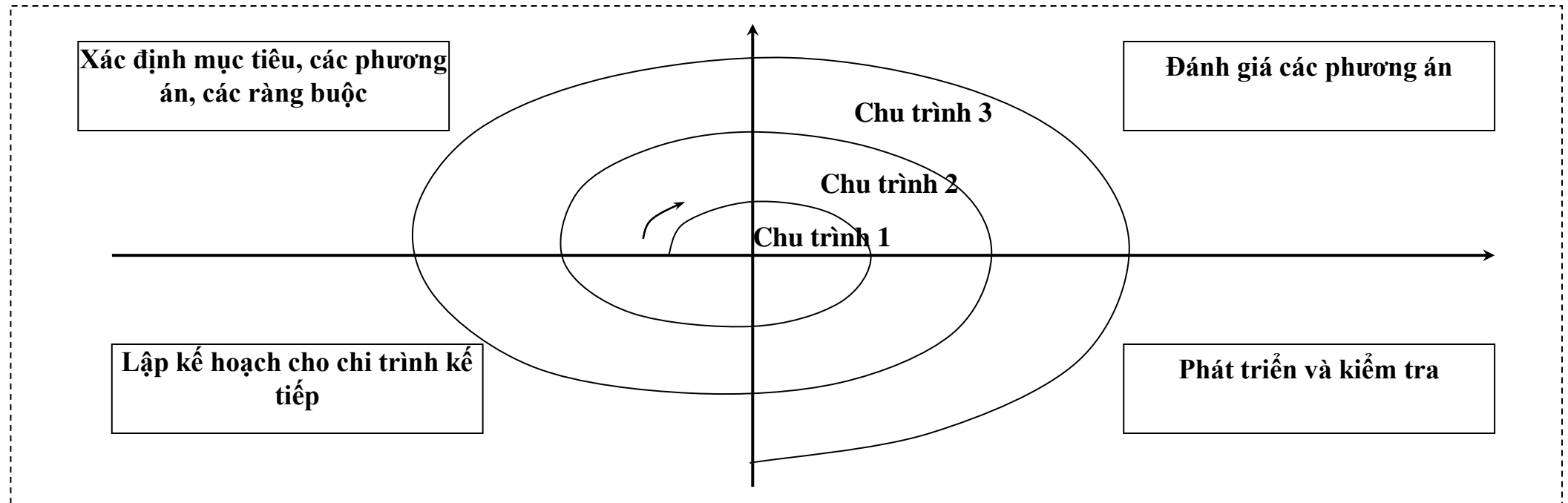
- Hoàn thành từng thành phần của hệ thống
- Mỗi bước tăng trưởng áp dụng qui trình tuyến tính xây dựng hoàn thành một phần của hệ thống
- Nhược điểm: chỉ phù hợp cho những hệ thống có sự phân chia rõ ràng và chuyển giao theo từng phần



Các qui trình phát triển hệ thống

▪ Qui trình xoắn ốc (Boehm, 1988)

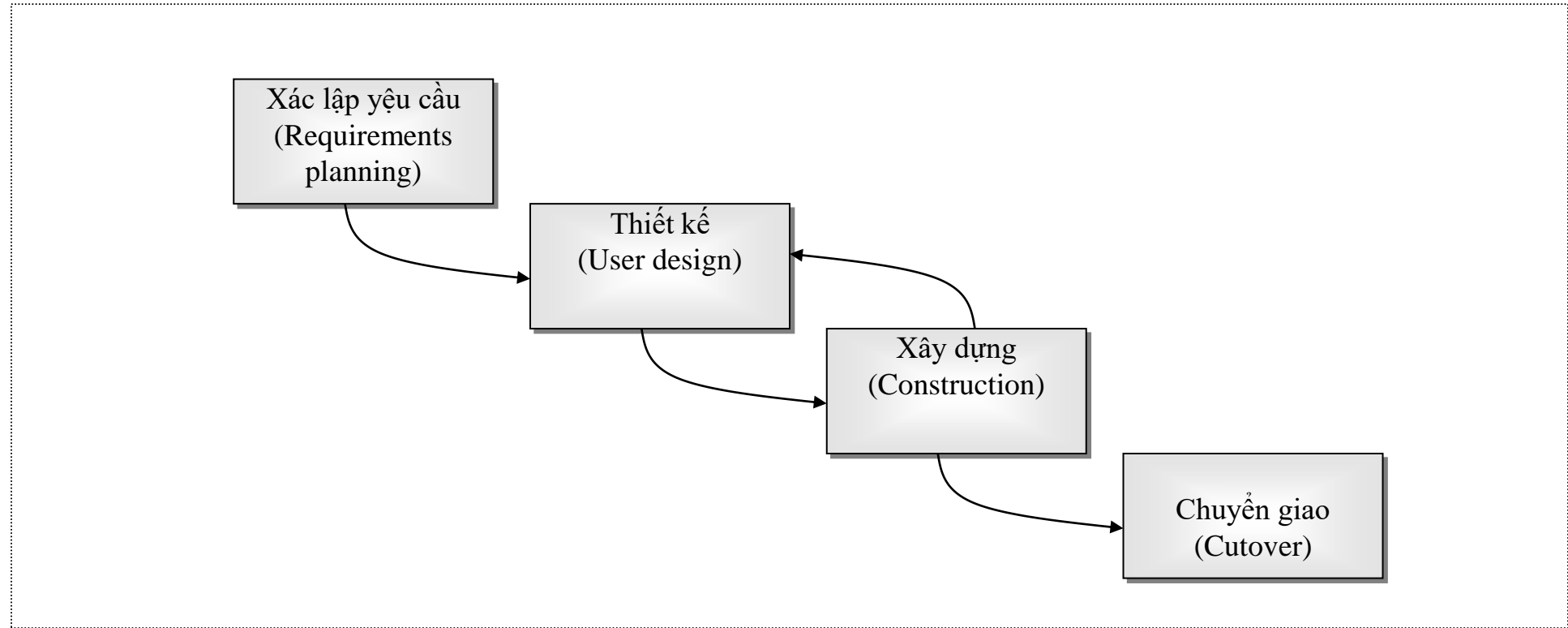
- Một đặc điểm quan trọng của qui trình này là nhấn mạnh việc quản lý rủi ro.
- Dựa trên khái niệm chu trình phát triển, qui trình này là các chu trình lặp.



Các qui trình phát triển hệ thống

- **Qui trình phát triển nhanh** (RAD – Rapid Development Application - James Martin, 1991)
 - Người phát triển hệ thống và các người sử dụng hệ thống sẽ làm việc kết hợp chặt chẽ với nhau
 - Không phải là một phương pháp luận riêng lẻ mà là một chiến lược chung để phát triển HTTT, có những đặc điểm sau:
 - Sử dụng các công cụ phần mềm và các môi trường phát triển trực quan để thể hiện tối đa các kết quả đạt được
 - Đẩy nhanh việc phân tích vấn đề, thiết kế một giải pháp hệ thống
 - Là một quá trình lặp thay đổi và điều chỉnh

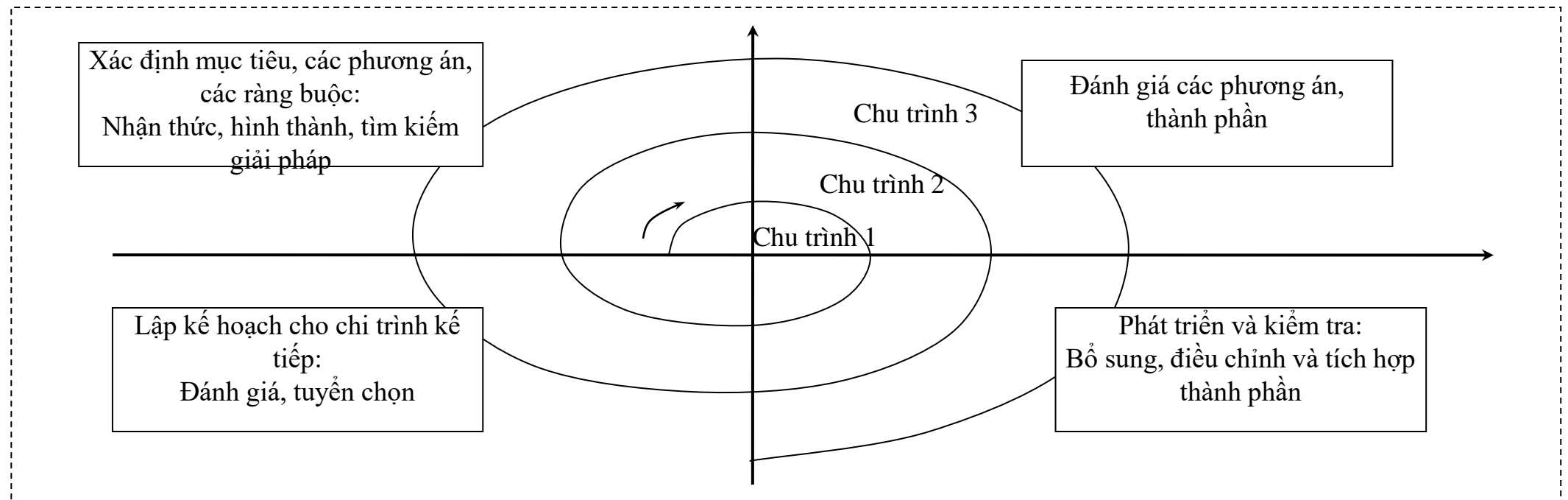
Các qui trình phát triển hệ thống



Các qui trình phát triển hệ thống

■ Qui trình lắp ráp thành phần (component)

- Dựa trên việc tái sử dụng thành phần phần mềm có sẵn
- Phát triển một phần mềm được thực hiện bằng cách tập hợp lại các thành phần có sẵn
- Là một quy trình lặp, mỗi chu trình gồm 4 giai đoạn



Các qui trình phát triển hệ thống

- Qui trình đồng nhất RUP (RUP – Rational Unified Process)

