PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG THÔNG TIN

Chương 5: THIẾT KẾ HỆ THỐNG THÔNG TIN

VIETHANIT

BỘ THÔNG TIN & TRUYỀN THÔNG

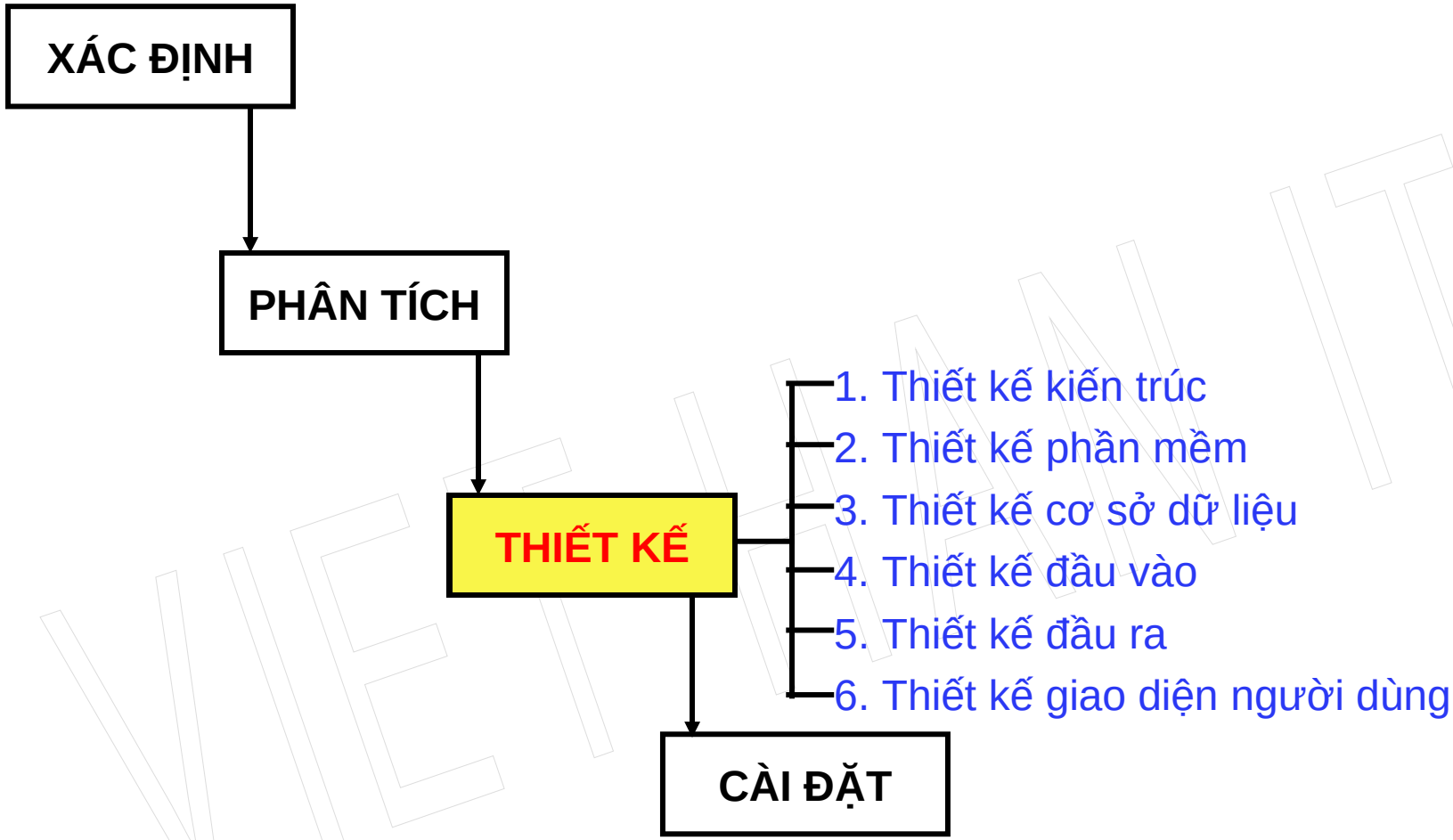
TRƯỜNG CAO ĐẲNG CNTT HỮU NGHỊ VIỆT - HÀN

THIẾT KẾ KIẾN TRÚC

ORACLE

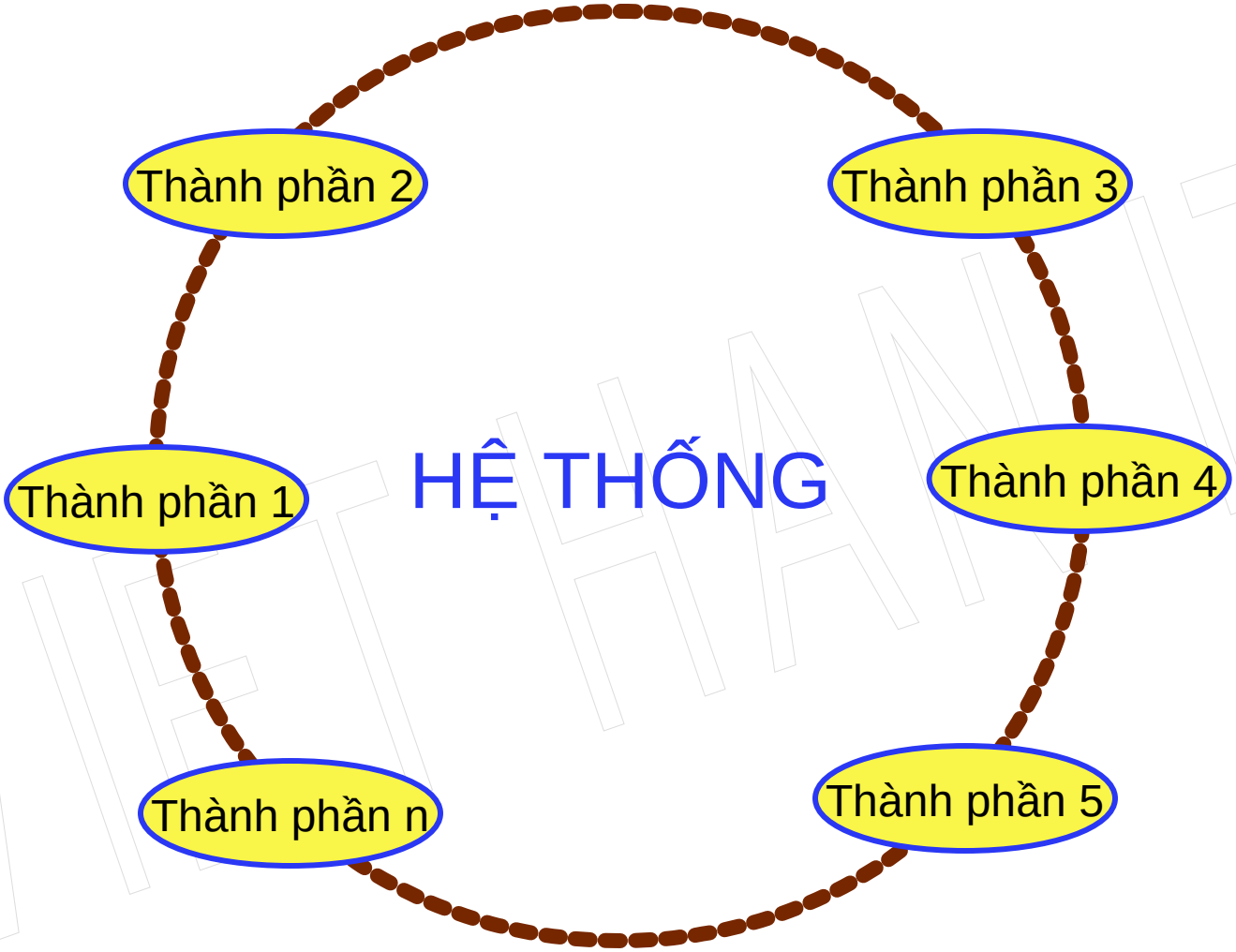
Copyright © 2008, Le Viet Truong

5.1 THIẾT KẾ KIẾN TRÚC



VIETHANIT

5.1 THIẾT KẾ KIẾN TRÚC



5.1 THIẾT KẾ KIẾN TRÚC

*** Ưu điểm:**

- Thỏa mãn các yêu cầu phi chức năng

- Tái sử dụng

*** Nhược điểm:**

- Sử dụng nhiều thành phần lớn:

. Tăng hiệu năng

. Giảm khả năng bảo trì

- Dữ liệu dư thừa:

. Cải thiện tính sẵn dùng

. Khó khăn trong việc bảo mật

- Nhiều giao tiếp:

. Giảm hiệu năng (tính an toàn)

VIETHANIT

5.1 THIẾT KẾ KIẾN TRÚC

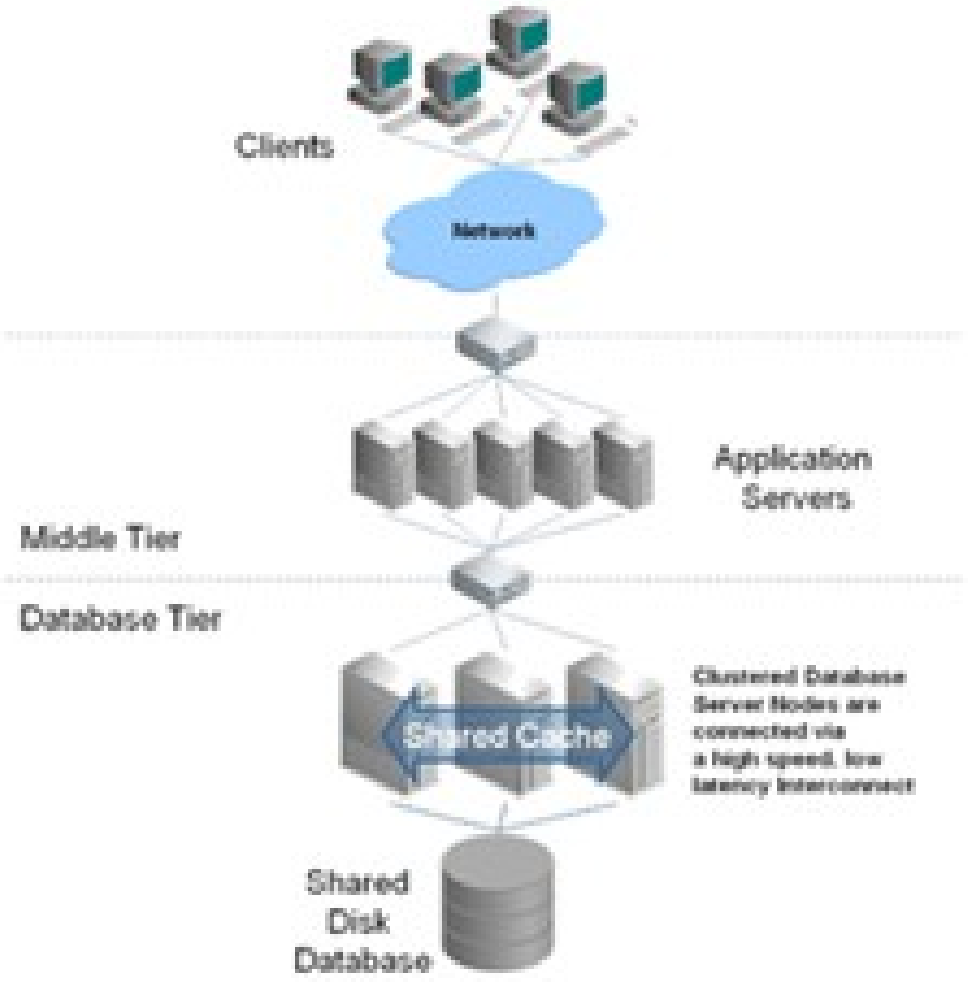
*** Các đặc điểm của kiến trúc HTTT:**

- Hiệu năng
  - . Hạn chế thao tác phức tạp
  - . Tối thiểu hóa giao tiếp
- Bảo mật
  - . Sử dụng kiến trúc phân lớp
  - . Kiểm soát chặt chẽ ở các lớp sâu hơn
- An toàn
- Sẵn dùng
- Có khả năng bảo trì

VIETHANIT

5.1 THIẾT KẾ KIẾN TRÚC

*** Ví dụ về kiến trúc 3 lớp**



5.1 THIẾT KẾ KIẾN TRÚC

*** Quy trình thiết kế kiến trúc dựa vào những quyết định:**

- Kiến trúc ứng dụng chung có được sử dụng lại hay không?
- Hệ thống thông tin sẽ được phân tán như thế nào?
- Những phong cách kiến trúc nào là thích hợp?
- Hệ thống thông tin sẽ được phân rã thành những mô-đun nào?
- Chiến lược điều khiển nào sẽ được sử dụng?
- Cách đánh giá thiết kế kiến trúc
- Kiến trúc sẽ được tư liệu hoá như thế nào?

5.1 THIẾT KẾ KIẾN TRÚC

*** Các mô hình kiến trúc cơ bản:**

- Mô hình cấu trúc tĩnh
- Mô hình quy trình động
- Mô hình giao diện
- Mô hình quan hệ
- Mô hình phân tán

VIETHANIT

5.1.2 Tổ chức hệ thống thông tin

*** Thiết kế kiến trúc: 3 bước**

- Xây dựng mô hình tổ chức HTTT

- Phân rã HTTT

- Các chiến lược điều khiển

VIETHANIT

5.1.2 Tổ chức hệ thống thông tin

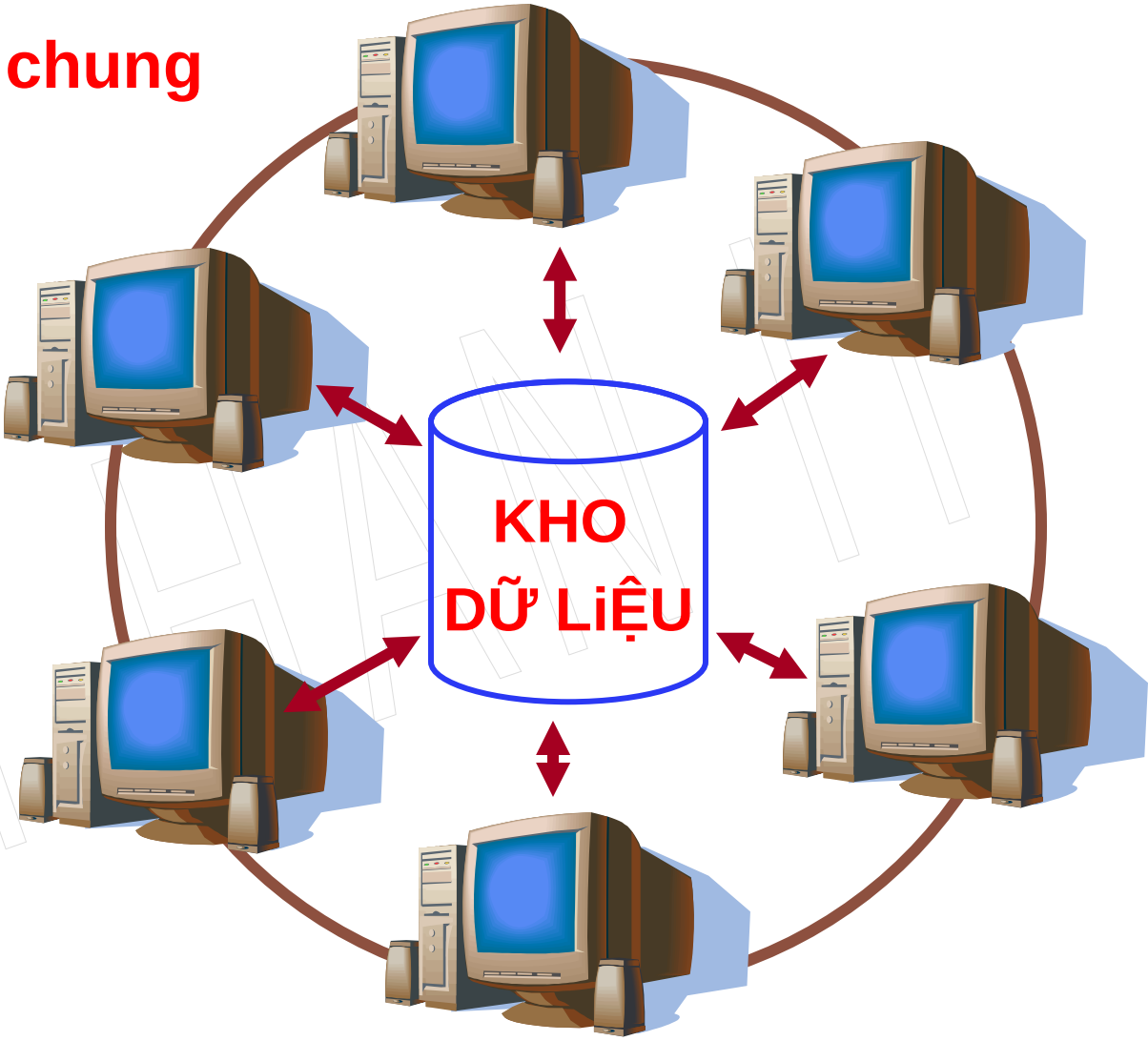
*** Tổ chức hệ thống thông tin**

- Kho dữ liệu dùng chung
- Client-Server
- Phân lớp hoặc máy trừu tượng

VIETHANIT

5.1.2 Tổ chức hệ thống thông tin

* Kho dữ liệu dùng chung



5.1.2 Tổ chức hệ thống thông tin

*** Ưu điểm của kho dữ liệu dùng chung**

- Hiệu quả để chia sẽ dữ liệu
- Quản lý dữ liệu tập trung

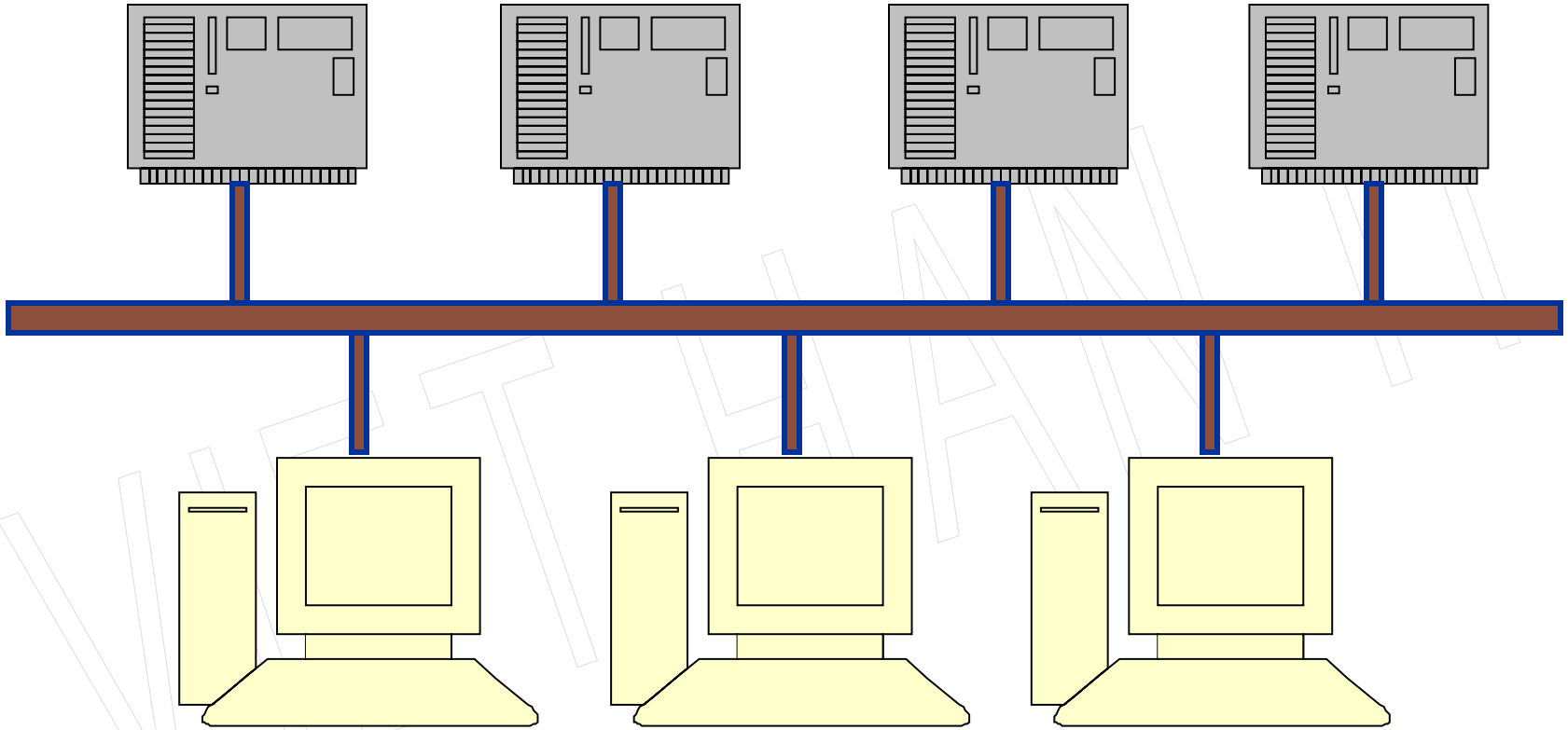
*** Nhược điểm của kho dữ liệu dùng chung**

- Các hệ thống con phải "chấp nhận" mô hình kho dữ liệu
- Cải tiến dữ liệu phức tạp và tốn kém
- Khó phân tán một cách có hiệu quả
- Giới hạn cho các chính sách quản lý ?

VIETHANIT

5.1.2 Tổ chức hệ thống thông tin

* Mô hình client/server



5.1.2 Tổ chức hệ thống thông tin

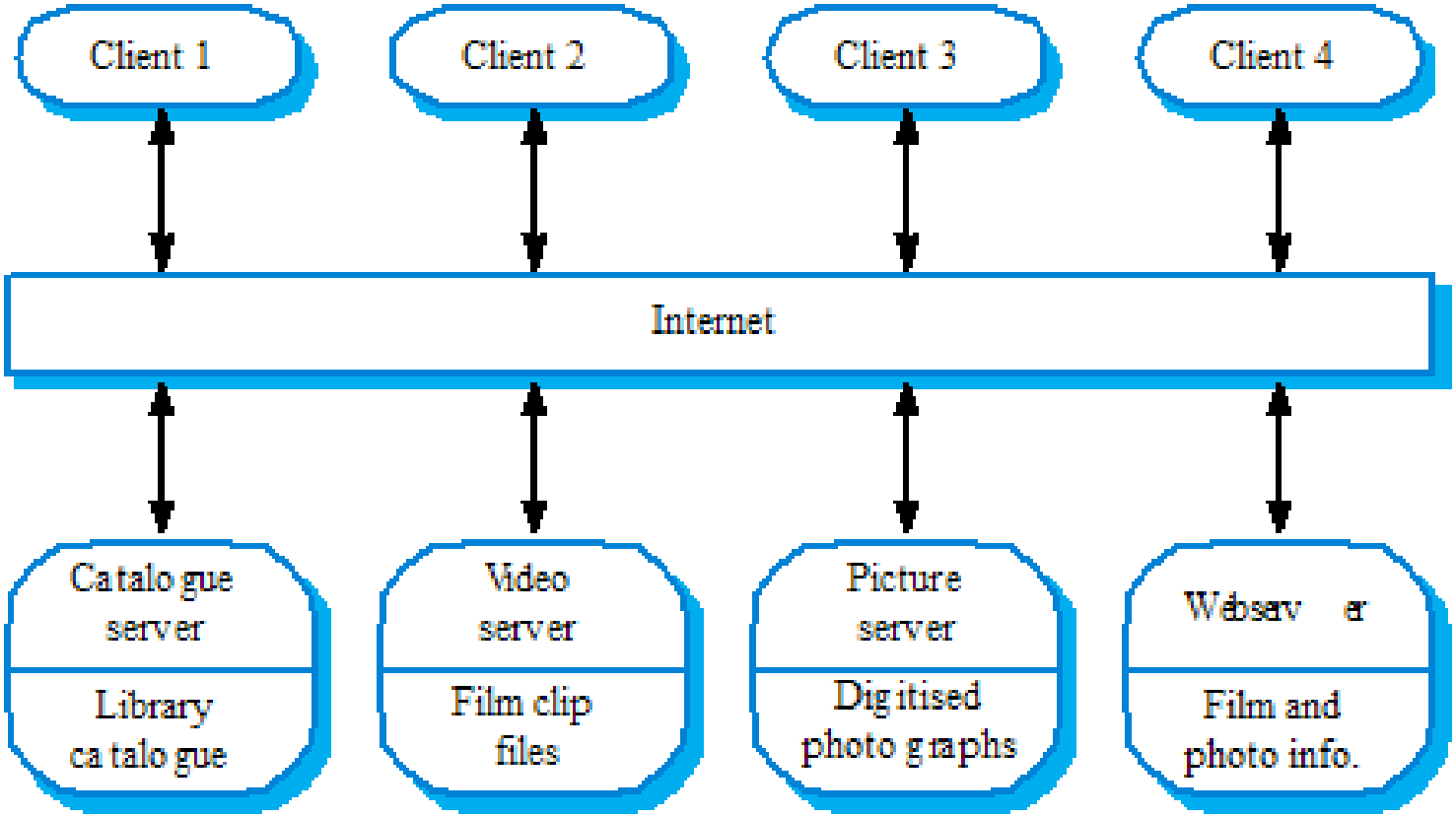
*** Ưu điểm của mô hình client/server**

- Phân tán dữ liệu rõ ràng
- Kết nối rõ ràng và có thể giảm chi phí phần cứng
- Dễ dàng bổ sung/nâng cấp server

*** Nhược điểm của mô hình client/server**

- Hiệu quả của việc trao đổi dữ liệu ?
- Quản lý các server không thống nhất, dư thừa.
- Không đăng ký tên và dịch vụ tập trung.

5.1.2 Tổ chức hệ thống thông tin



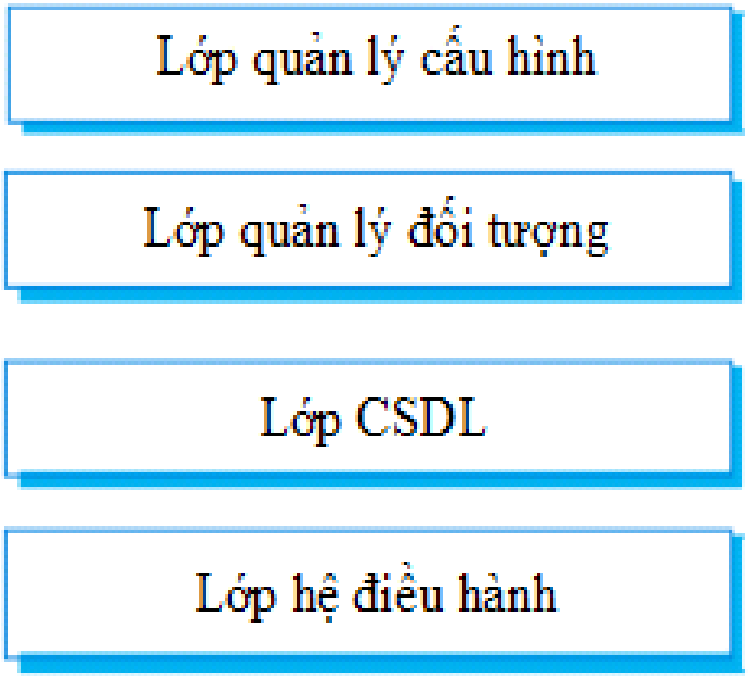
5.1.2 Tổ chức hệ thống thông tin

*** Mô hình phân lớp**

- Tổ chức HTTT thành nhiều lớp (máy trừu tượng)
- Mỗi lớp cung cấp mỗi tập dịch vụ
- Được dùng để mô hình hóa giao diện của các hệ thống con
- Hỗ trợ phát triển các hệ thống con theo kiểu tăng vòng ở nhiều lớp khác nhau
- Giao diện của một lớp thay đổi chỉ ảnh hưởng đến các lớp liền kề

VIETHANIT

5.1.2 Tổ chức hệ thống thông tin



Hình 5.3 Mô hình phân lớp của hệ thống quản lý phiên bản

5.1.3 Phân rã hệ thống thông tin

- Phân rã hệ thống con thành các mô-đun
- Có hai cách phân rã
  - Phân rã hướng đối tượng: thành các đối tượng tương tác với nhau
  - Pipeline hướng chức năng hoặc dòng dữ liệu: thành các mô-đun chức năng

VIETHANIT

5.1.3 Phân rã hệ thống thông tin

*** Phân rã hướng đối tượng**

- Gắn kết lỏng
- Dựa trên các giao diện đã được định nghĩa
- Liên quan đến xác định lớp đối tượng, thuộc tính và phương thức
- Khi cài đặt lớp thì sinh ra các đối tượng

VIETHANIT

5.1.3 Phân rã hệ thống thông tin

*** Ưu điểm của phân rã hướng đối tượng**

- Gắn kết lỏng
- Đối tượng phản ánh thực thể trong thế giới thực
- Ngôn ngữ lập trình hướng đối tượng

*** Nhược điểm của phân rã hướng đối tượng**

- Thay đổi giao diện của một đối tượng sẽ gây ra một số khó khăn
- Khó biểu diễn các thực thể phức tạp trong thế giới thực

5.1.3 Phân rã hệ thống thông tin

*** Pipeline hướng chức năng**

- Quy trình chuyển đổi thông tin đầu vào thành thông tin đầu ra
- Tuần tự hoặc song song
- Dữ liệu được xử lý riêng lẻ hoặc theo lô

VIETHANIT

5.1.3 Phân rã hệ thống thông tin

*** Ưu điểm của pipeline hướng chức năng**

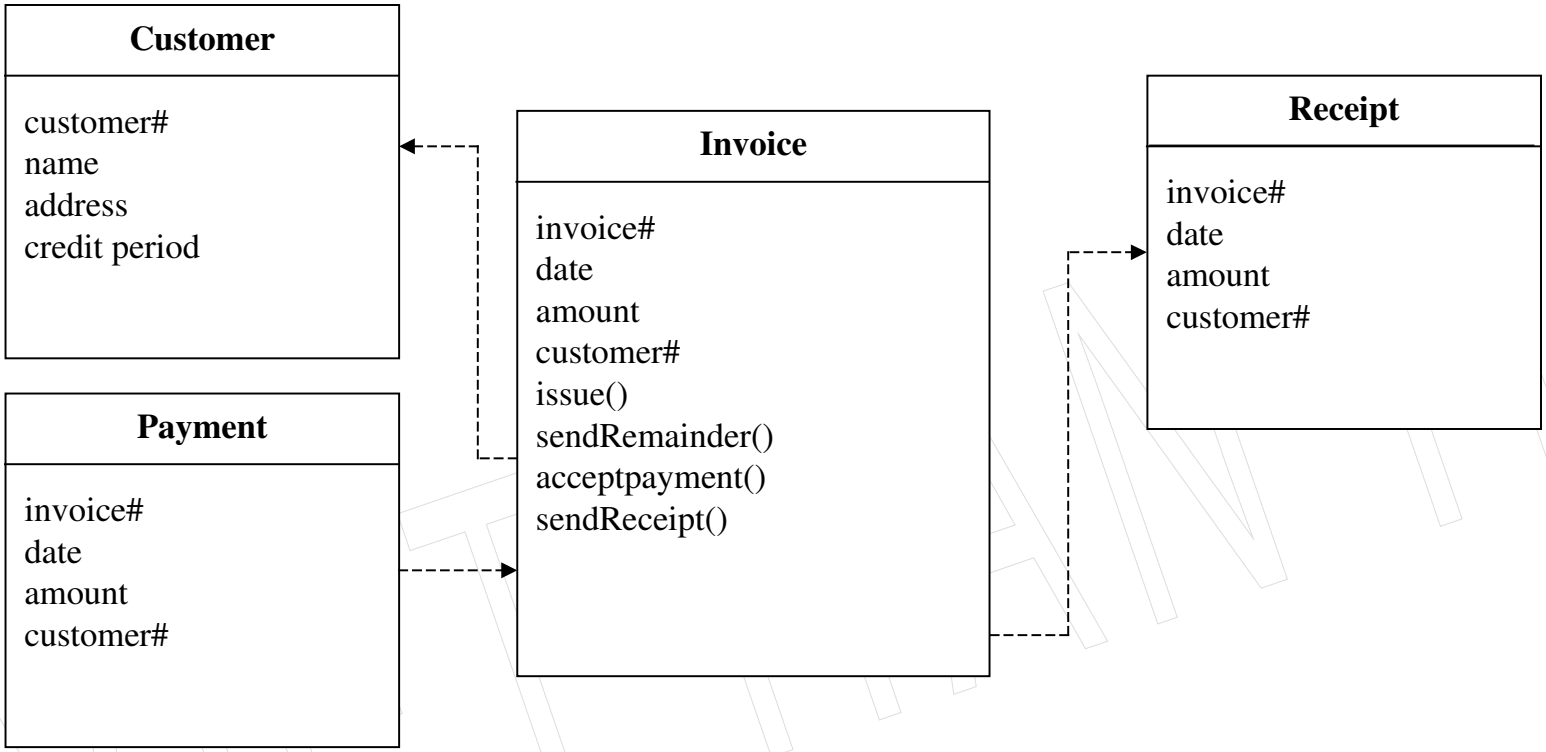
- Hỗ trợ tái sử dụng
- Cung cấp tài liệu để giao tiếp với các bên có liên quan
- Dễ dàng bổ sung
- Dễ dàng thực hiện

*** Nhược điểm của phân rã hướng đối tượng**

- Yêu cầu có định dạng dữ liệu dùng chung
- Khó hỗ trợ cho tương tác hướng sự kiện

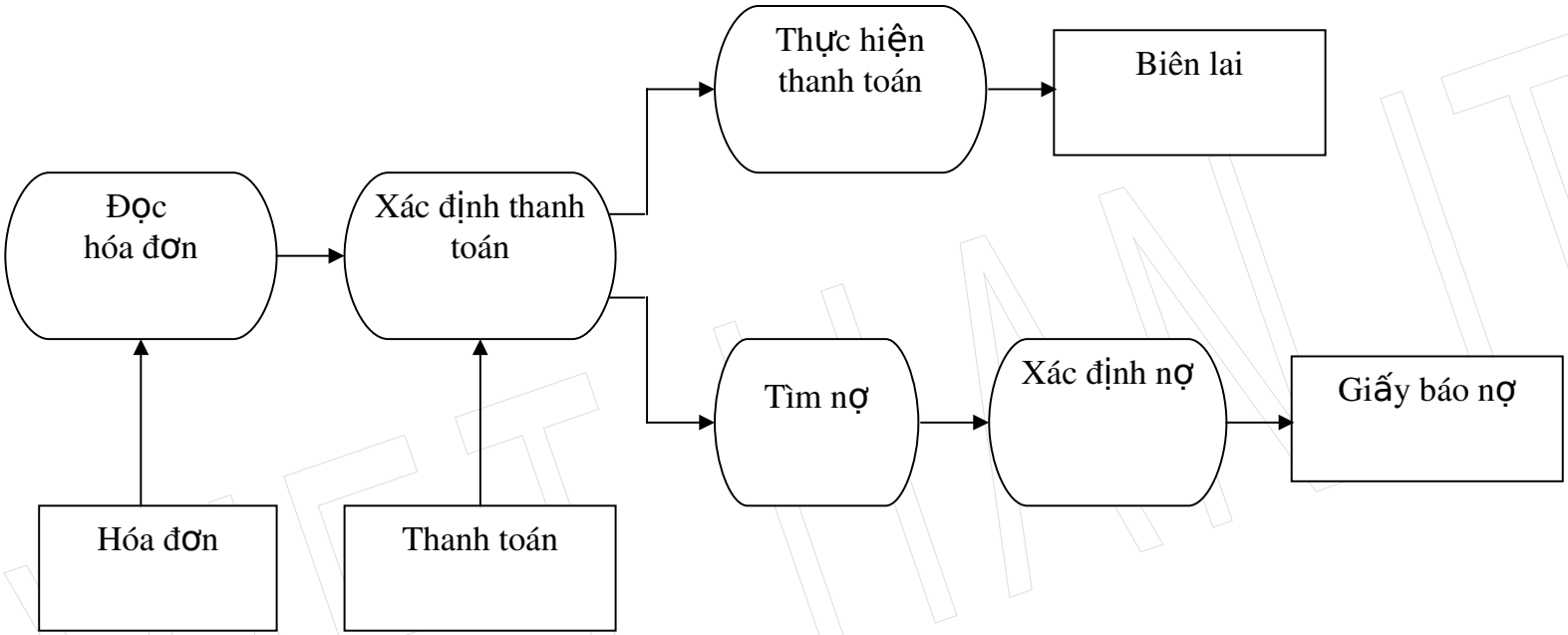
VIETHANIT

5.1.3 Phân rã hệ thống thông tin



Hình 5.4 Các đối tượng trong hệ thống xử lý hóa đơn

5.1.3 Phân rã hệ thống thông tin



Hình 5.5 Mô hình dòng dữ liệu hệ thống xử lý hóa đơn

5.1.4 Các chiến lược điều khiển

- Các hệ thống hoạt động đúng chỗ, đúng thời điểm
- Hai chiến lược điều khiển:
  . Điều khiển tập trung
  . Điều khiển hướng sự kiện

VIETHANIT

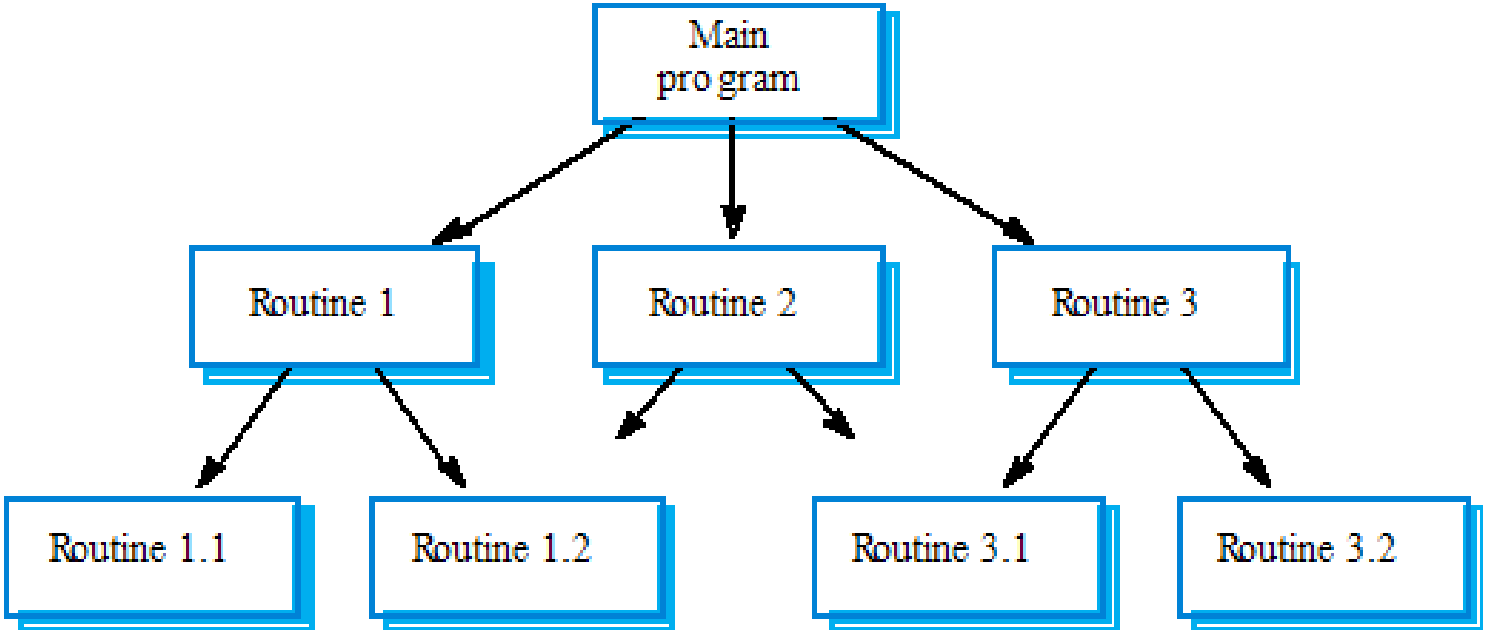
5.1.4 Các chiến lược điều khiển

* Điều khiển tập trung có hai mô hình

. Mô hình gọi - trả lời (call-return)

+ Phân cấp

+ Áp dụng cho các hệ thống tuần tự

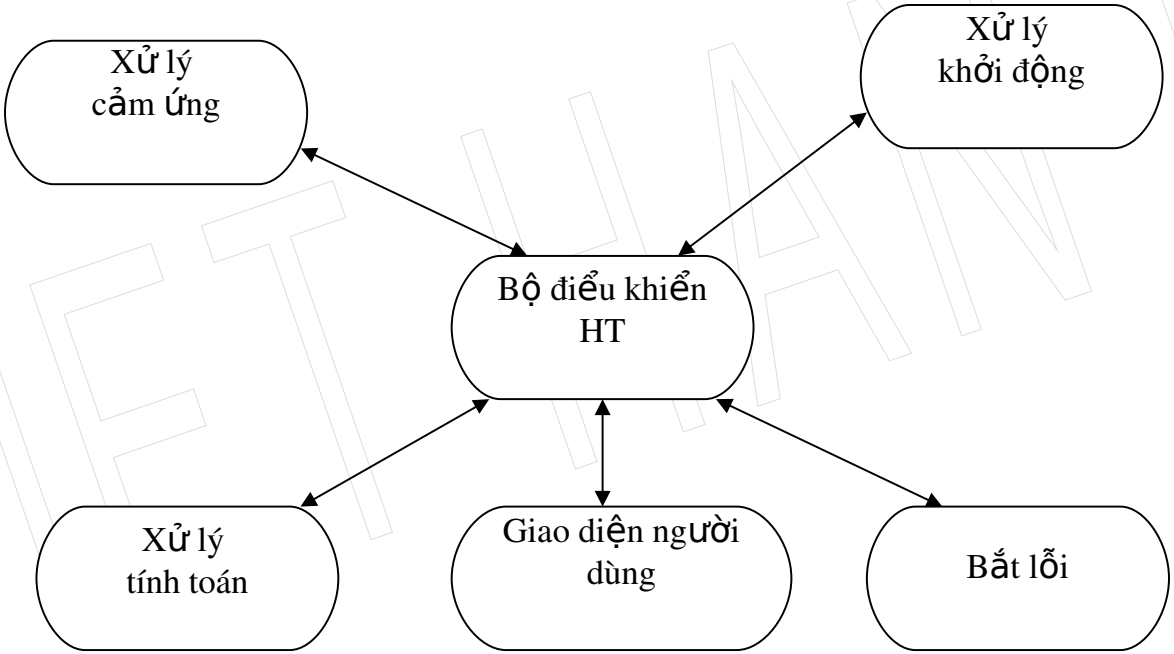


5.1.4 Các chiến lược điều khiển

* Điều khiển tập trung có hai mô hình

. Mô hình quản lý

+ Áp dụng cho các hệ thống song song

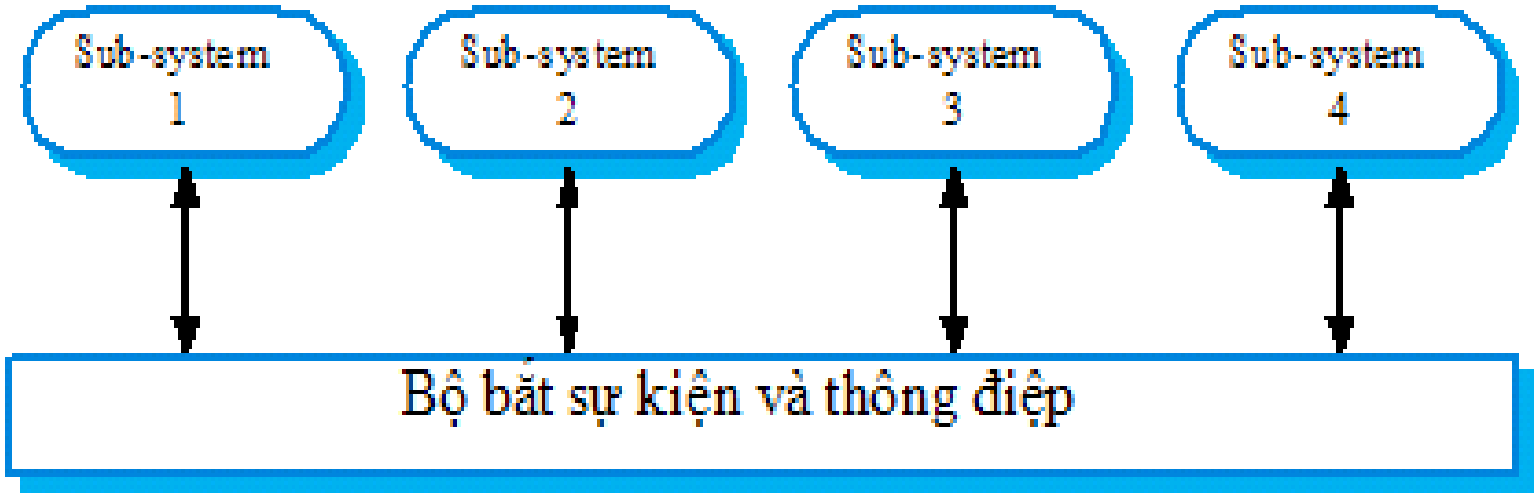


5.1.4 Các chiến lược điều khiển

*** Điều khiển hướng sự kiện có hai mô hình**

. Mô hình lan truyền

+ Sự kiện được lan truyền đến tất cả các hệ thống con



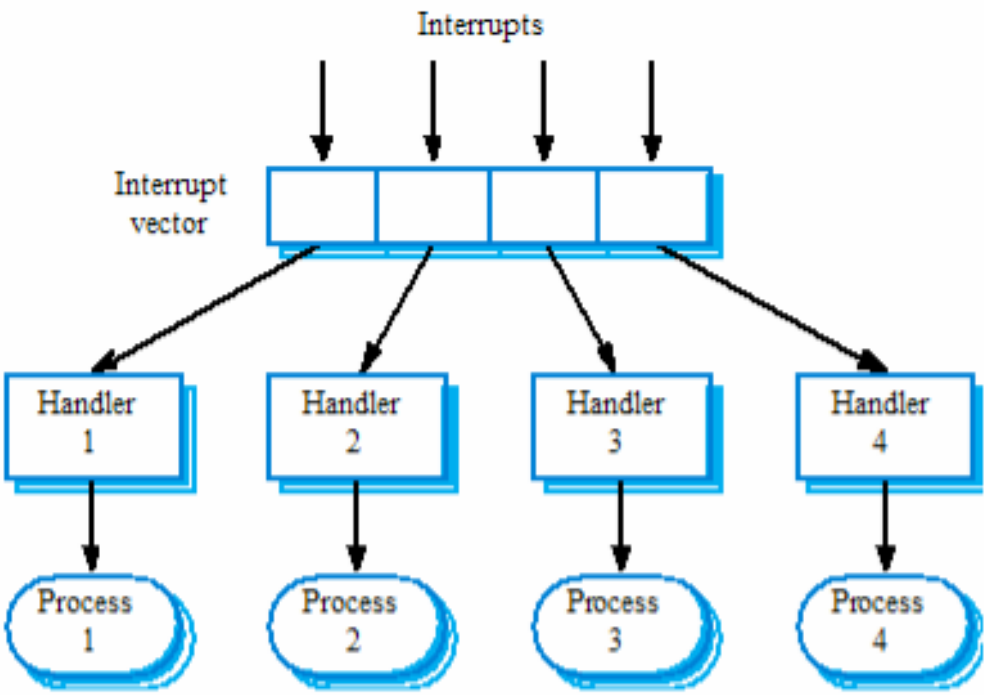
5.1.4 Các chiến lược điều khiển

* Điều khiển hướng sự kiện có hai mô hình

. Mô hình hướng ngắt

+ Sử dụng trong các HTTT thời gian thực

+ Có các ngắt và các bộ bắt ngắt



5.1.5 Các kiến trúc tham chiếu

- Các HTTT cùng miền ứng dụng: có kiến trúc chung giống nhau, có thể khác về chi tiết
- Có thể tái sử dụng kiến trúc của HTTT tương tự
- Có hai mô hình cho một miền ứng dụng cụ thể:
  . Mô hình chung: Bottom-Up
  . Mô hình tham chiếu: Top-Down

VIETHANIT

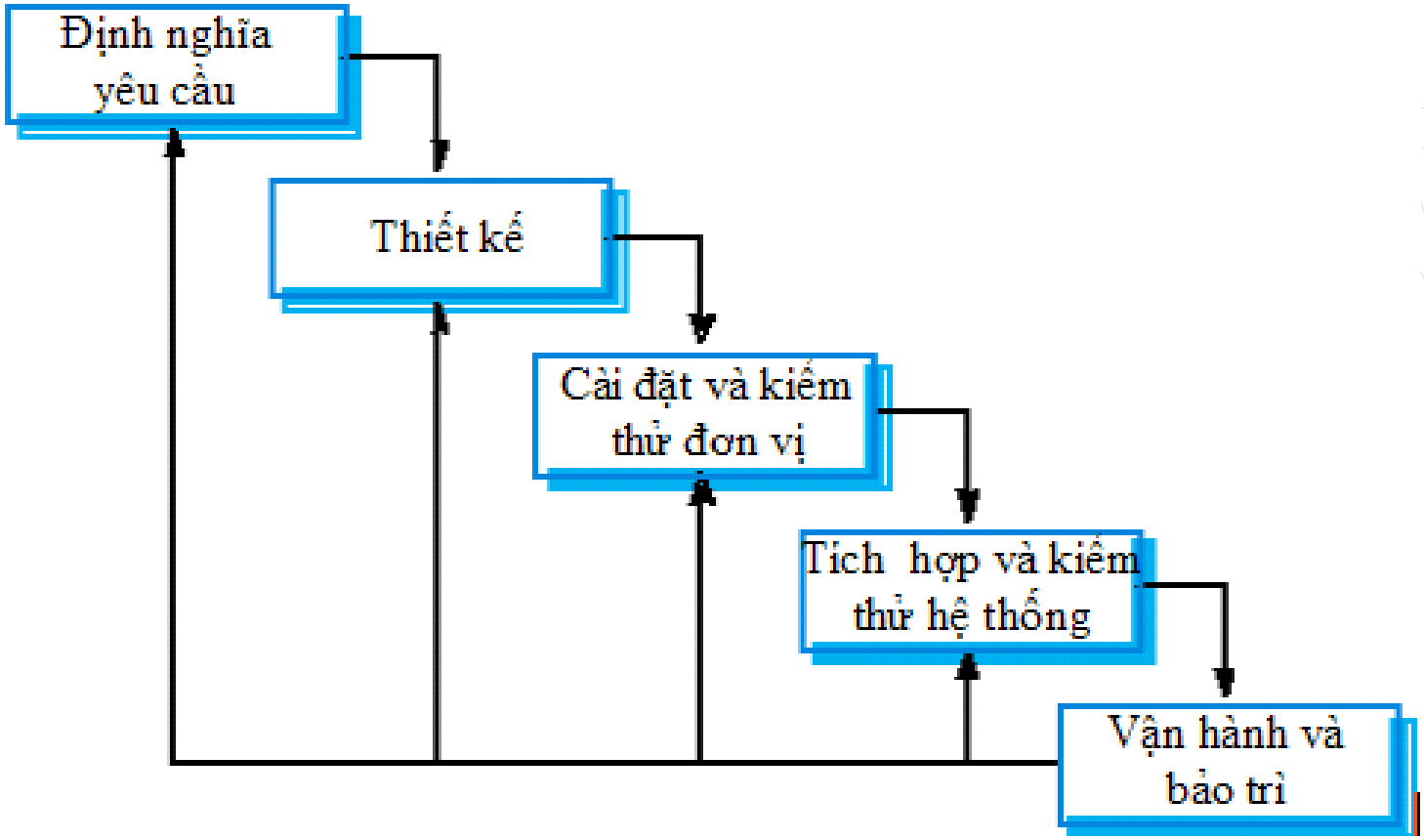
5.2 Thiết kế phần mềm

5.2.1 Các mô hình phát triển phần mềm

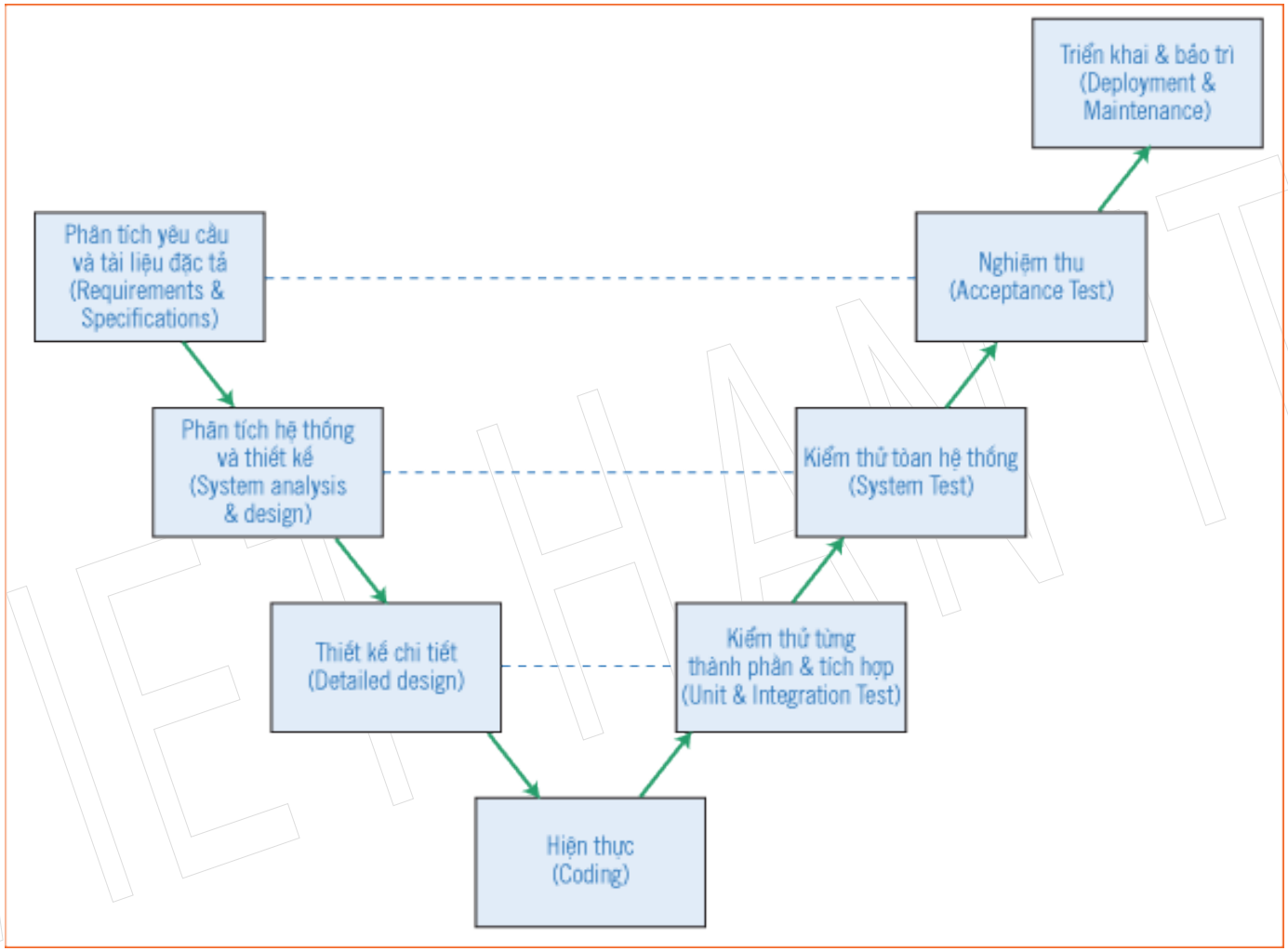
- Mô hình thác nước,
- Mô hình xây dựng tiến triển,
- Công nghệ phần mềm dựa thành phần,
- Mô hình phát triển lặp lại, tăng thêm,
- Mô hình xoắn ốc

VIETHANIT

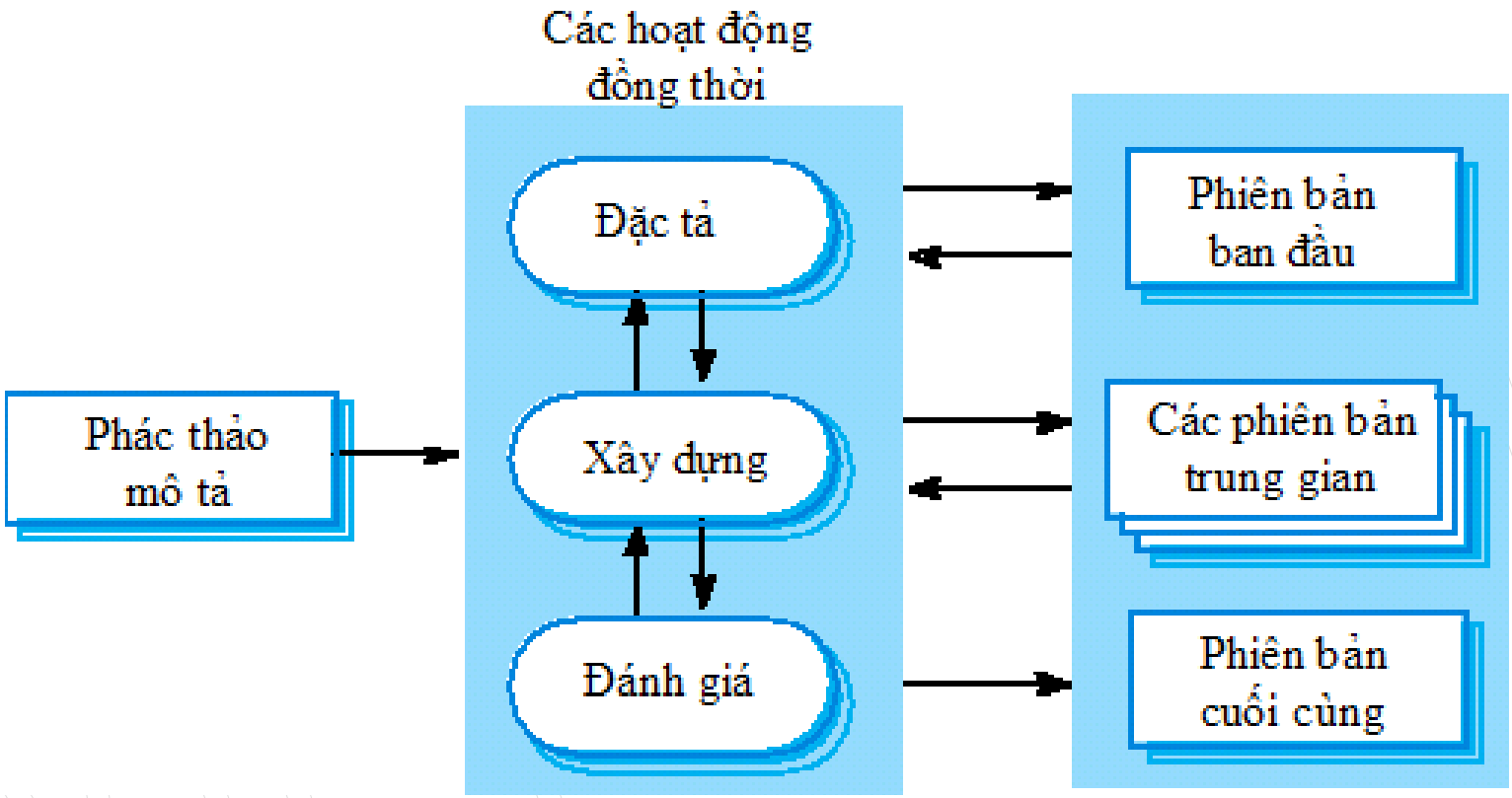
Mô hình thác nước



Mô hình chữ V


Triển khai & bảo trì
(Deployment &
Maintenance)
Phân tích yêu cầu
và tài liệu đặc tả
(Requirements &
Specifications)
Nghiệm thu
(Acceptance Test)
Phân tích hệ thống
và thiết kế
(System analysis
& design)
Kiểm thử toàn hệ thống
(System Test)
Thiết kế chi tiết
(Detailed design)
Kiểm thử từng
thành phần & tích hợp
(Unit & Integration Test)
Hiện thực
(Coding)

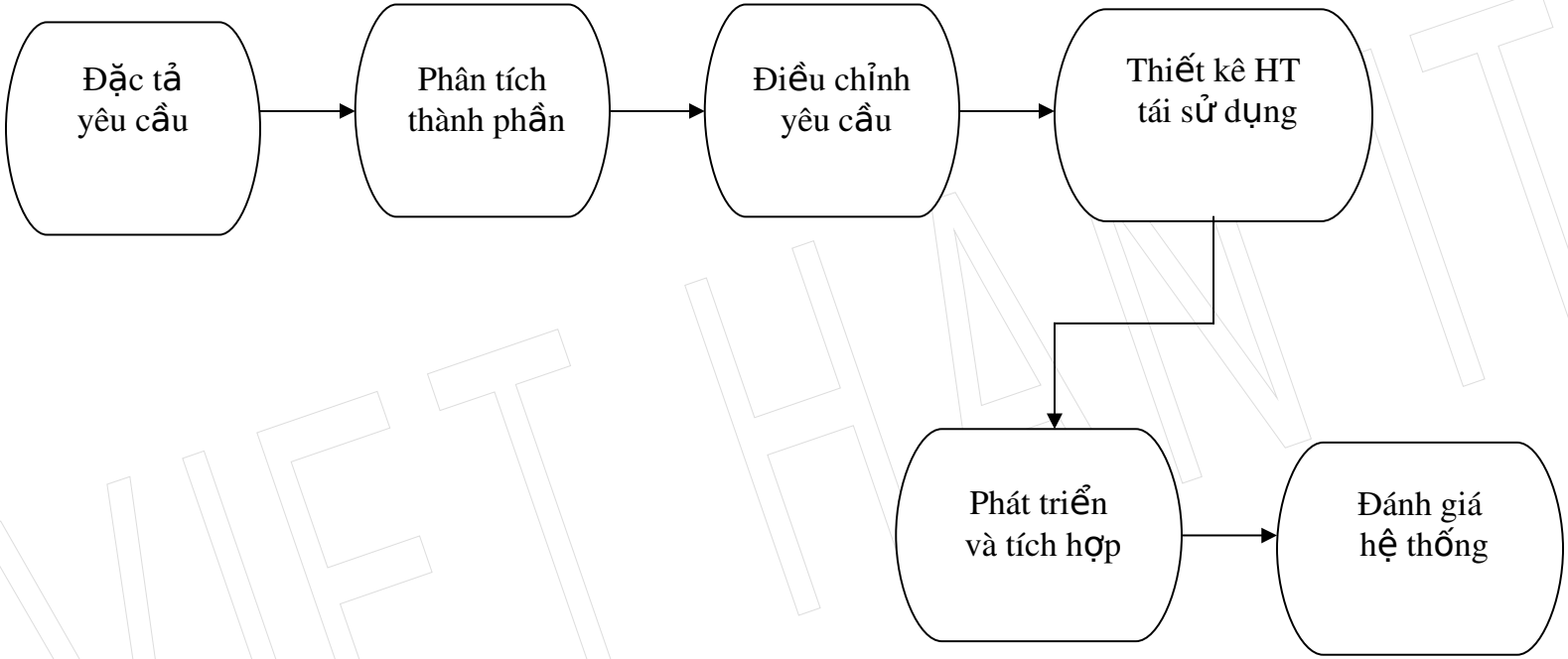
Mô hình xây dựng tiến triển



Có 02 phương pháp: Phát triển thăm dò và Loại bỏ mẫu thử

CNPM dựa thành phần

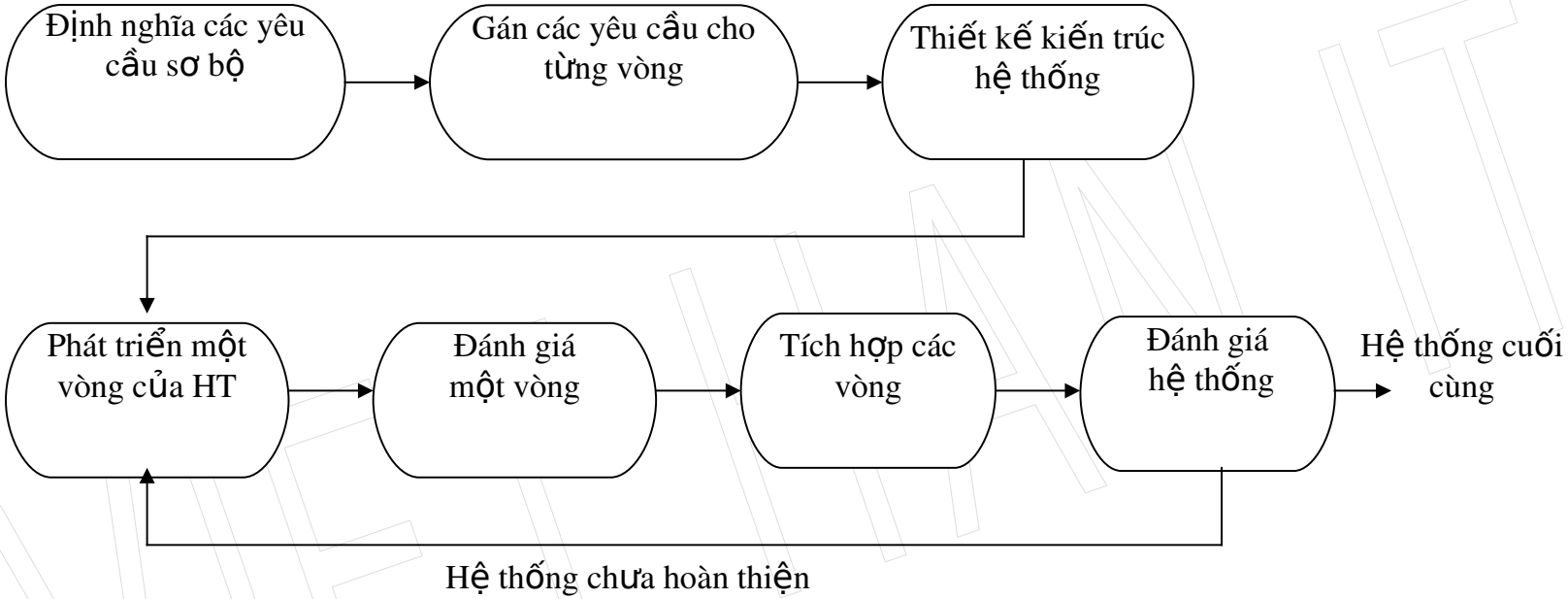
Dựa vào kỹ thuật tái sử dụng



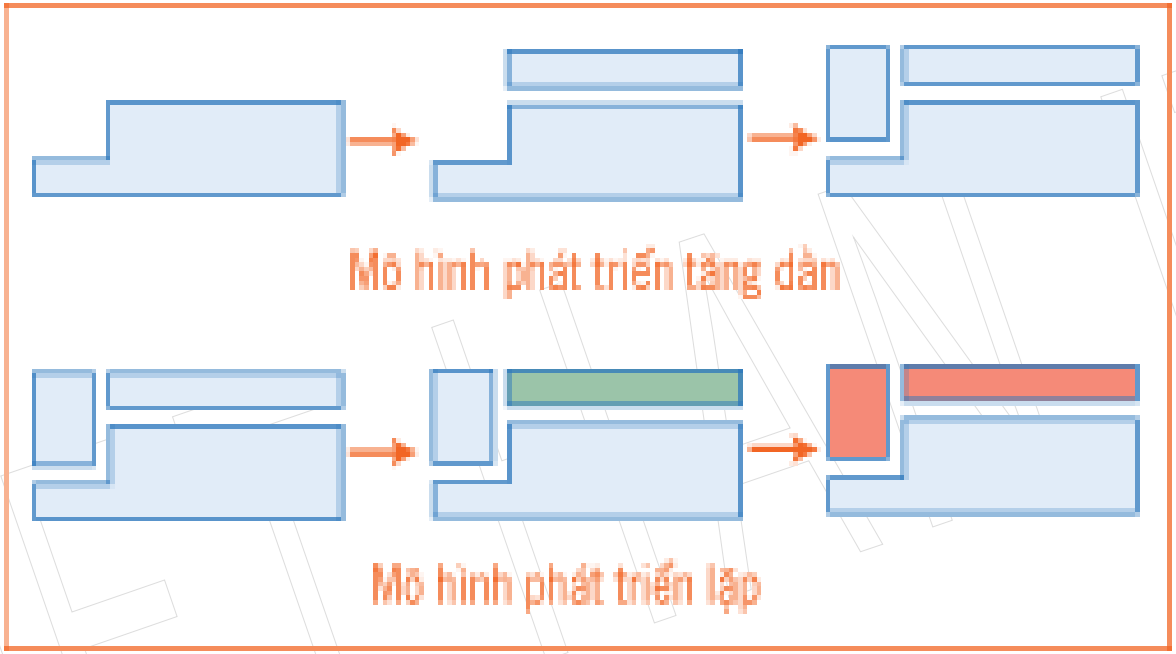
VIETHANIT

Mô hình phát triển lặp lại, tăng thêm

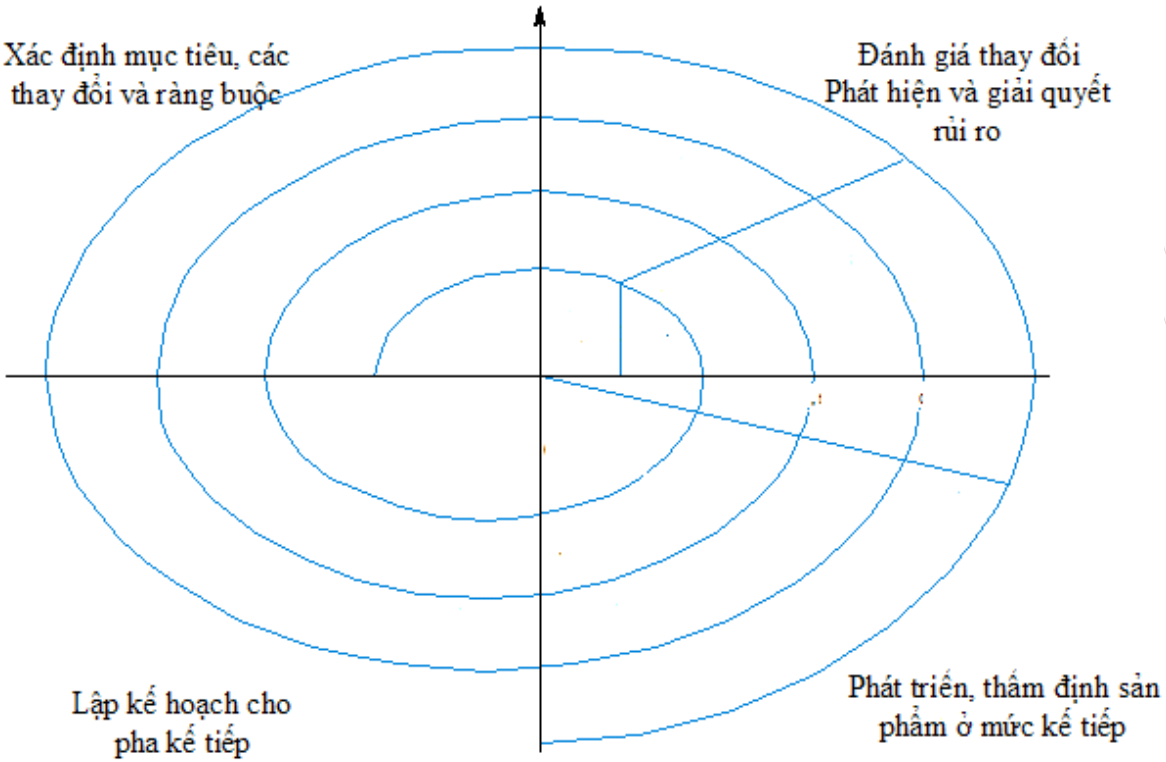
Dựa vào thứ tự ưu tiên các yêu cầu



Mô hình phát triển lặp lại, tăng thêm



Mô hình xoắn ốc



5.2 Thiết kế phần mềm

5.2.1 Các mô hình phát triển phần mềm

- Mô hình thác nước,
- Mô hình xây dựng tiến triển,
- Công nghệ phần mềm dựa thành phần,
- Mô hình phát triển lặp lại, tăng thêm,
- Mô hình xoắn ốc

VIETHANIT

5.2.2 Các hoạt động trong quy trình phần mềm

ĐẶC TẢ

THIẾT KẾ VÀ CÀI ĐẶT

ĐÁNH GIÁ

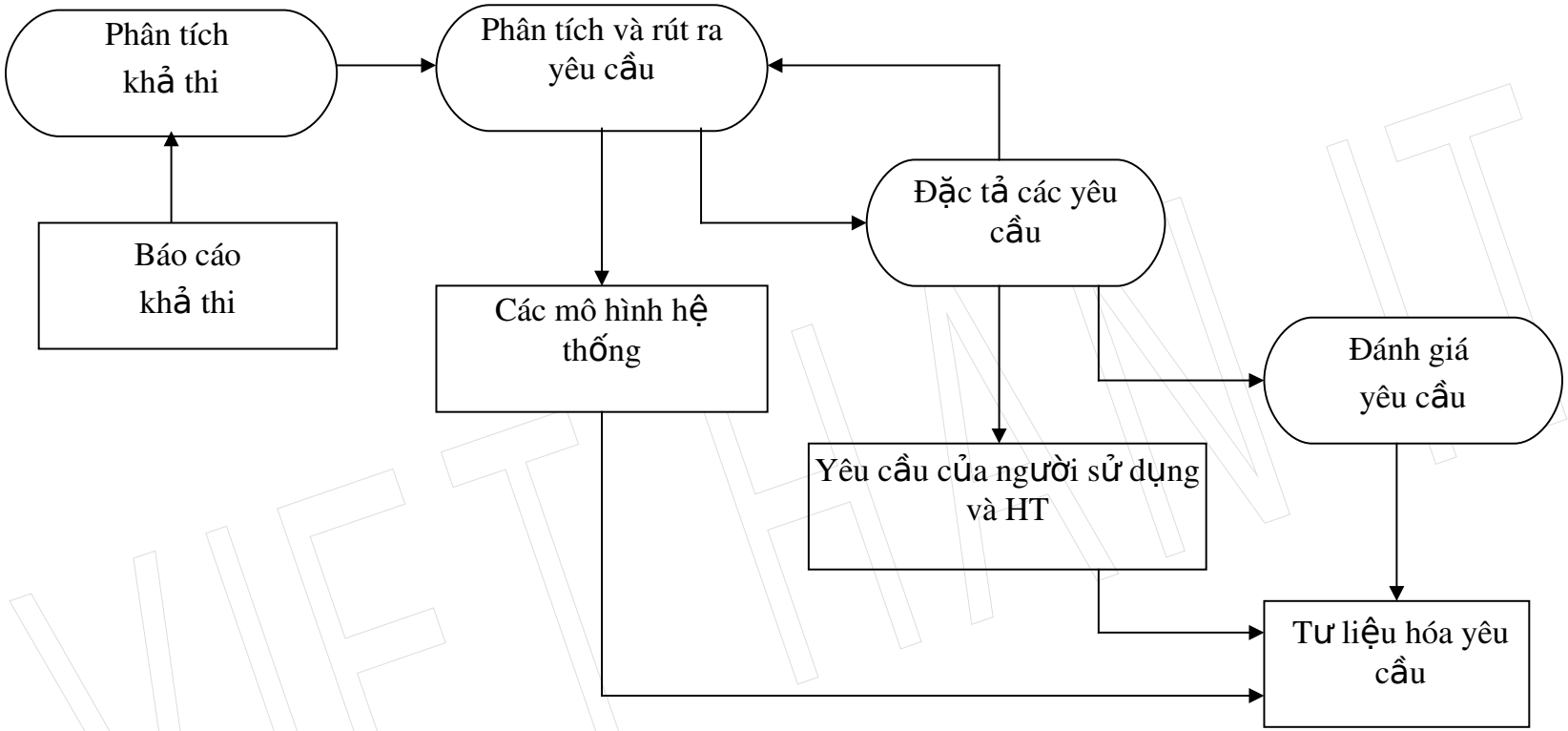
CẢI TiẾN

Giai đoạn đặc tả phần mềm

- Còn gọi là kỹ thuật xác định yêu cầu
- Gồm 4 giai đoạn chính:
  . Nghiên cứu khả thi
  . Phân tích và rút ra các yêu cầu
  . Đặc tả yêu cầu: Yêu cầu NSD và yêu cầu hệ thống
  . Đánh giá yêu cầu

VIETHANIT

Giai đoạn đặc tả phần mềm



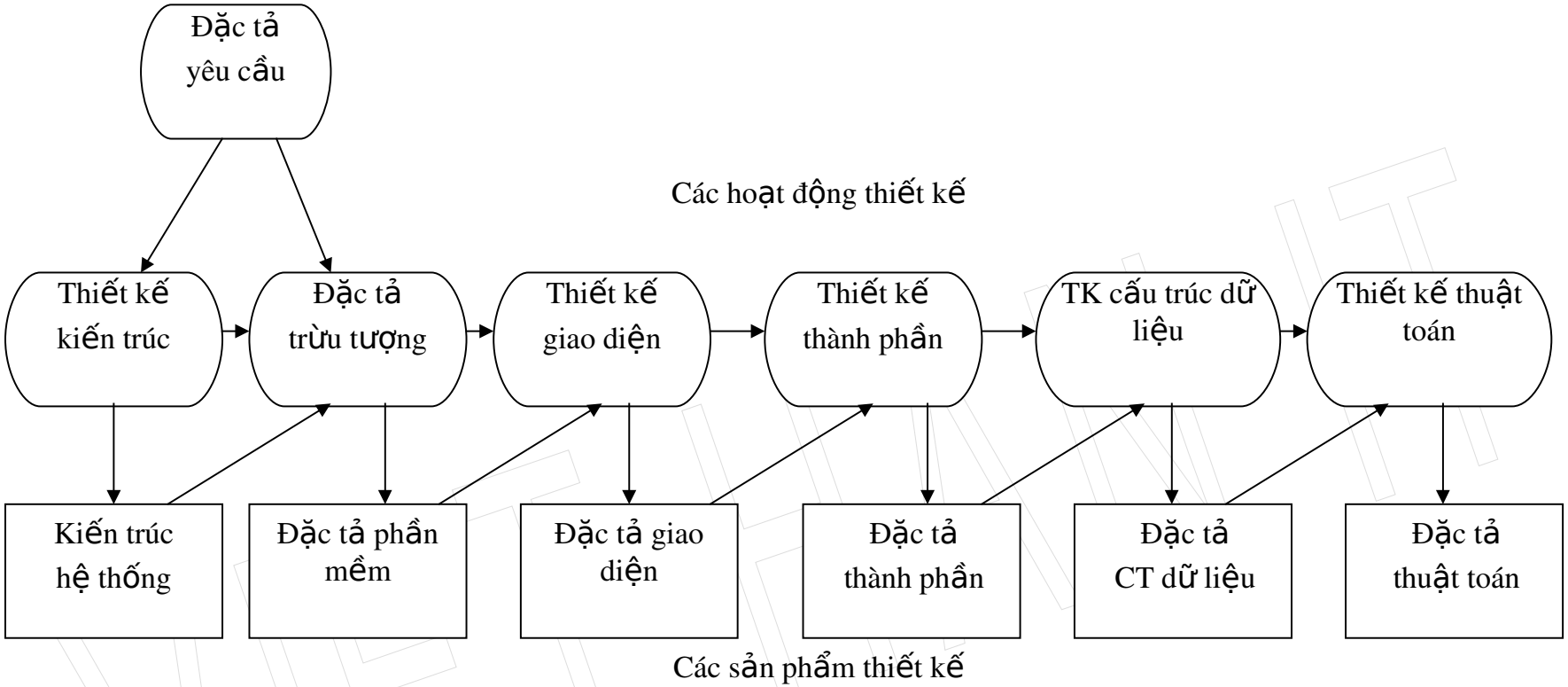
Hình 5.15 Kết quả tăng dần

Giai đoạn thiết kế

- Thiết kế kiến trúc
- Đặc tả trừu tượng
- Thiết kế giao diện
- Thiết kế thành phần
- Thiết kế cấu trúc dữ liệu
- Thiết kế thuật toán

VIETHANIT

Giai đoạn thiết kế



Hình 5.16 Mô hình chung của quy trình thiết kế

Giai đoạn cài đặt

- Là quy trình chuyển đổi từ tài liệu đặc tả hệ thống thông tin thành một hệ thống thông tin thực.
- Không có quy trình chung cho lập trình

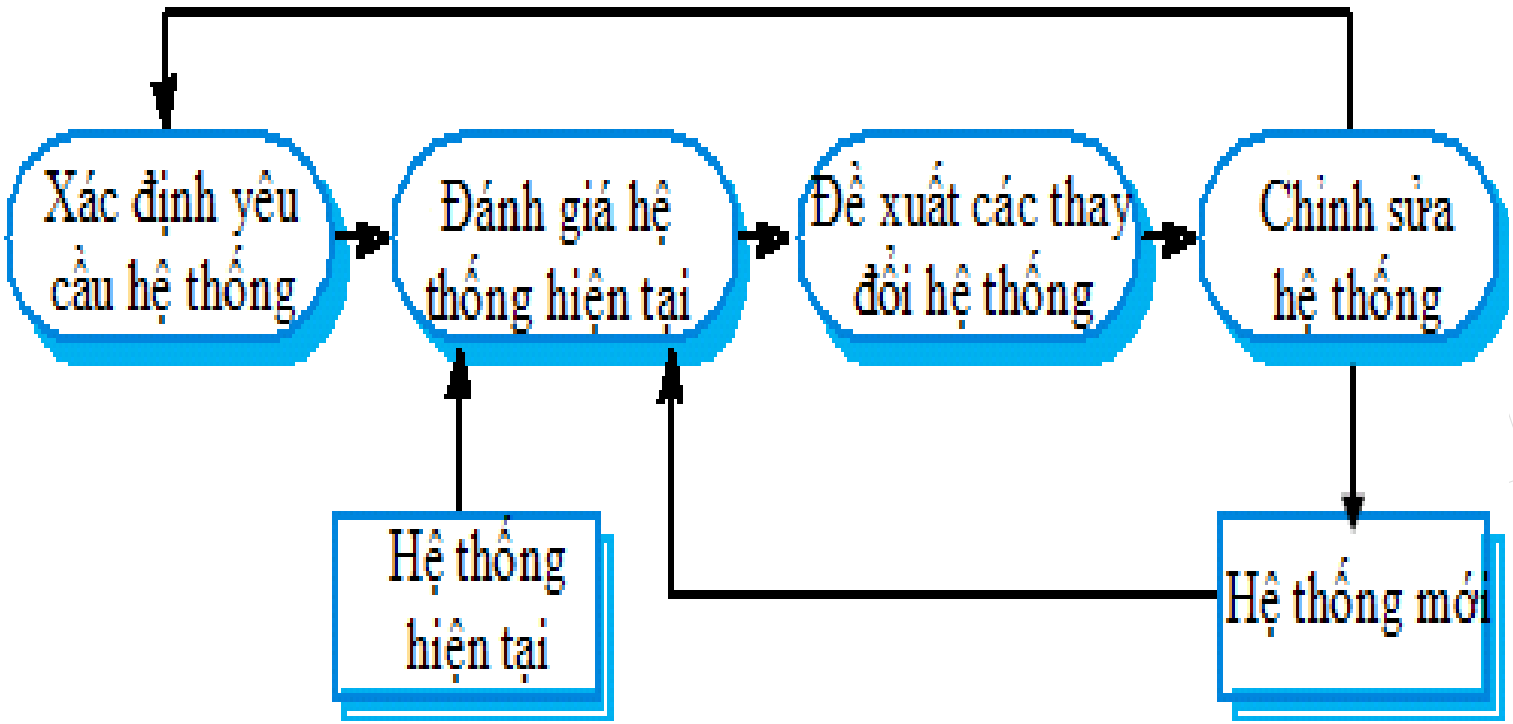
VIETHANIT

Giai đoạn đánh giá phần mềm

- V&V: Thẩm tra và đánh giá; Xác minh và thẩm định
- Kiểm thử cũng là một phần của V & V

VIETHANIT

Giai đoạn cải tiến phần mềm



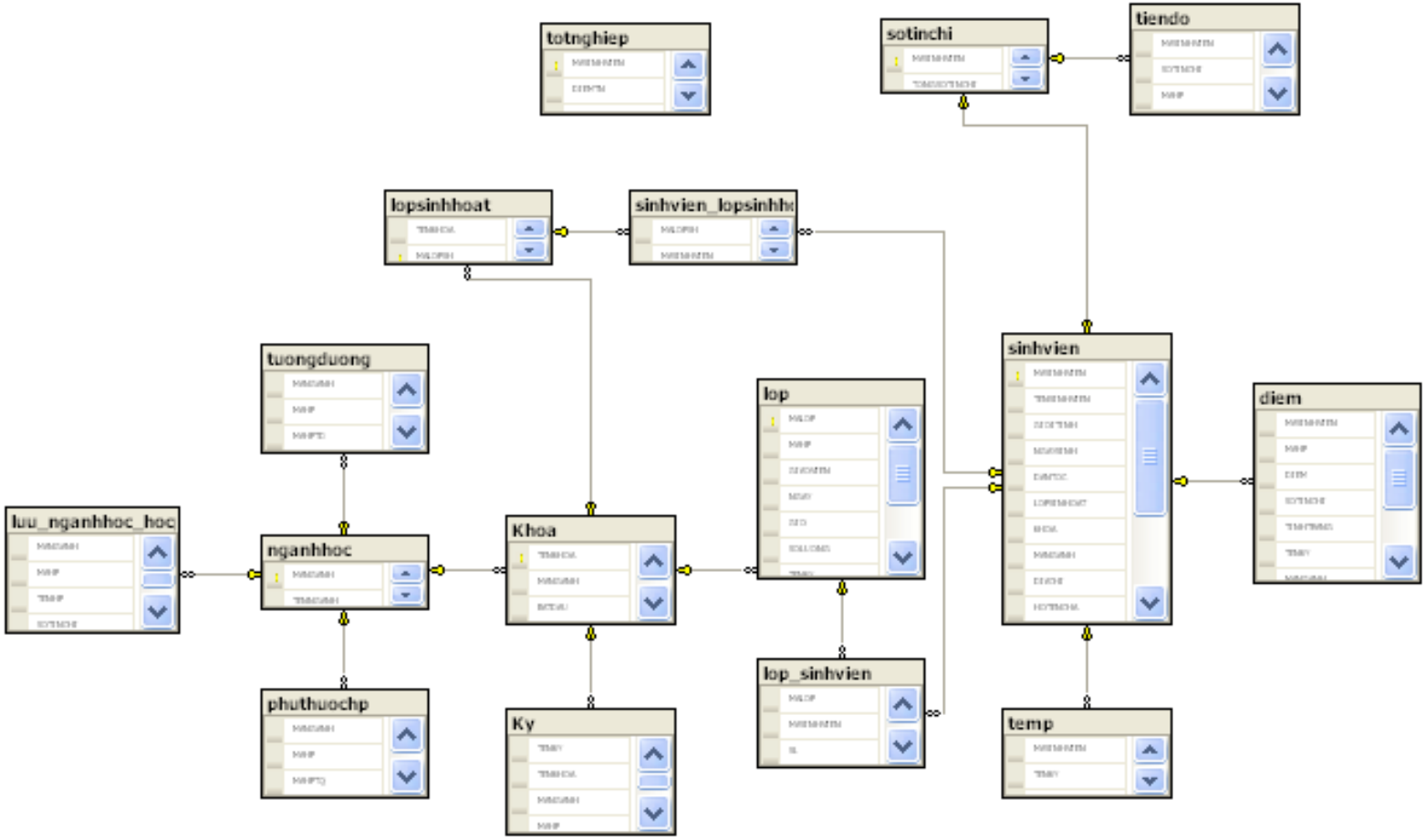
Hình 5.18 Cải tiến phần mềm

Phương pháp luận thiết kế cơ sở dữ liệu

- Cần có quy trình hợp lý để thiết kế CSDL
- Quá trình thông thường có các giai đoạn chính:
  . Thiết kế CSDL logic (khái niệm): định nghĩa dữ liệu và các quan hệ
  . Thiết kế CSDL vật lý: thi hành thiết kế và đảm bảo tính toàn vẹn
  . Hiệu chỉnh thiết kế logic
- Lược đồ và lược đồ con

VIETHANIT

Ví dụ một lược đồ CSDL



5.3.3 Triển khai mô hình dữ liệu logic dựa trên một CSDL quan hệ

5.3.3.1 CSDL quan hệ:

- Là CSDL lưu trữ và quản lý dữ liệu trong các bảng 2 chiều
- Các bảng quan hệ với nhau thông qua KHÓA
- Các đặc thù:
  . Schema
  . SQL
  . Trigger
  . Stored procedure

VIETHANIT

5.3.3 Triển khai mô hình dữ liệu logic dựa trên một CSDL quan hệ

5.3.3.2 Mô hình hóa dữ liệu

- Mô hình dữ liệu tốt là mô hình:

  . Một thuộc tính mô tả một và chỉ một thực thể

  . Một thuộc tính chỉ tồn tại duy nhất ở một thực thể, trừ thuộc tính KHÓA NGOẠI

- Để có mô hình dữ liệu tốt phải tiến hành chuẩn hóa

VIETHANIT

5.3.3.2 Mô hình hóa dữ liệu

Các bước tạo mô hình dữ liệu vật lý:

- Xem lại mô hình dữ liệu logic.
- Tạo bảng cho mỗi thực thể.
- Tạo trường cho mỗi thuộc tính.
- Tạo chỉ mục (index) cho mỗi khoá.
- Thiết kế khoá ngoại cho các quan hệ.
- Định nghĩa kích thước/kiểu dữ liệu, thuộc tính null, giá trị mặc định.
- Đánh giá và thiết lập các ràng buộc.
- Chú ý các công nghệ khác nhau cho các kiểu dữ liệu khác nhau.

* Tuỳ vào từng hệ quản trị cơ sở dữ liệu, có thể tạo cơ sở dữ liệu tương ứng trên ngôn ngữ định nghĩa dữ liệu.

Ví dụ tạo CSDL bằng SQL

create table tblSinhVien (MaSV char(6)not null primary key, HoTen varchar(30), DiaChi varchar(50))

create table tblMonHoc (MaMon char(6) not null primary key , TenMon varchar(50))

create table tblDiem (MaMon char(6) not null, MaSV char(6) not null,
constraint pk_Diem primary key (MaMon,MaSV),
constraint fk_Diem foreign key (MaMon) references tblMonHoc(MaMon),
constraint fk_tblDiem foreign key (MaSV) references tblSinhVien(MaSV), Diem float)

Các phần SV tự nghiên cứu

1. Thiết kế đầu vào
2. Thiết kế đầu ra
3. Thiết kế giao diện người dùng

VIETHANIT

Những việc cần làm trong bài tập lớn

1. Phát biểu bài toán (nói rõ các yêu cầu bài toán)
2. Xây dựng các quy trình nghiệp vụ
3. Xây dựng biểu đồ phân rã chức năng BFD
4. Xây dựng biểu đồ dòng dữ dữ liệu DFD
5. Xác định các thực thể và biểu đồ thực thể quan hệ ERD
6. Xây dựng mô hình dữ liệu quan hệ RDM
7. Thiết kế CSDL
8. Thiết kế đầu vào, đầu ra và giao diện người dùng

(Chỉ thiết kế, không lập trình)

Những việc cần làm trong bài tập lớn

- Hàng tuần, sẽ báo cáo từng phần
- Khi một SV báo cáo, thành viên còn lại trong nhóm và trong lớp phải PHẢN BiỆN.
- Thang điểm:

  . Nội dung BTL: <= 4 điểm

  . Trả lời câu hỏi: <= 2 điểm

  . Báo cáo : <= 2 điểm

  . Phản biện : <= 2 điểm

VIETHANIT