

Chương 5 MỨC VẬT LÝ CỦA HTTT

Ở mức tổ chức, khi xem xét mô hình cơ sở dữ liệu thực chất chúng ta chỉ quan tâm đến cấu trúc lô gic của dữ liệu. Cấu trúc đó được thể hiện một cách độc lập với máy tính và các phần mềm quản trị dữ liệu cụ thể. Mức vật lý sẽ là thể hiện cụ thể trên máy tính cho giải pháp dữ liệu đã được lựa chọn. Nó được thể hiện ở hai khía cạnh: *cấu trúc dữ liệu cụ thể* và *phương thức truy nhập*. Cũng như hai mức đã khảo sát ở trước, mức vật lý được mô tả qua hai mô hình: mô hình vật lý về dữ liệu và mô hình vật lý về xử lý.

5.1 Mô hình vật lý về dữ liệu

5.1.1 Thiết kế cơ sở dữ liệu vật lý

Thiết kế cơ sở dữ liệu vật lý là bước cuối cùng của giai đoạn thiết kế dữ liệu. Quá trình thiết kế cơ sở dữ liệu vật lý là quá trình ánh xạ cấu trúc dữ liệu logic được xây dựng ở mô hình tổ chức dữ liệu vào mô hình bên trong hệ thống. Đa số các hệ thống thông tin hiện nay đều sử dụng một hệ quản trị cơ sở dữ liệu nào đó để tạo ra cơ sở dữ liệu cho hệ thống. Thiết kế cơ sở dữ liệu vật lý bao gồm các bước sau:

- *Thiết kế cơ sở dữ liệu*: mô tả các file dữ liệu, file chỉ mục,... sẽ được truy cập trong bộ nhớ máy tính như thế nào.
- *Thiết kế hệ thống và cấu trúc chương trình*: mô tả các chương trình và các mô đun chương trình khác nhau tương ứng với sơ đồ luồng dữ liệu và những yêu cầu đặt ra trong các bước phân tích trước.
- *Thiết kế chiến lược xử lý phân tán*: mô tả hệ thống xử lý dữ liệu như thế nào và các xử lý cho người sử dụng trên mạng máy tính.

Thông thường, người ta sử dụng các thông tin dưới đây để tạo cơ sở dữ liệu vật lý:

- Các quan hệ đã chuẩn hoá
- Định nghĩa các thuộc tính

- Các mô tả cho biết ở đâu và khi nào dữ liệu được sử dụng (đọc, sửa chữa, xoá,...)
- Các công nghệ được sử dụng để triển khai việc xây dựng cơ sở dữ liệu.

Mặc dù trong giai đoạn thiết kế dữ liệu chúng ta đã cố gắng chuẩn hóa các lược đồ quan hệ với hy vọng là các lược đồ đều ở dạng chuẩn 3, nhưng khi cài đặt cụ thể các file dữ liệu để tiện lợi chúng ta có thể bổ sung vào một số trường tính toán, hình thành một số trường phức hợp, đưa vào một số trường được phân rã từ một trường khác. Thậm chí, có thể ghép hai lược đồ ở dạng chuẩn 3, phá vỡ ý nghĩa của chuẩn hóa, để tiện việc xử lý.

Hiện nay nhiều công cụ CASE độc lập hoặc được tích hợp trong một số hệ quản trị cơ sở dữ liệu để hỗ trợ cho quá trình thiết kế cơ sở dữ liệu. Trên cơ sở cấu trúc dữ liệu vật lý được cung cấp, các CASE sẽ tạo ra các hệ thống file cần thiết phục vụ cho hoạt động của hệ thống. Chẳng hạn, Designer của Oracle, SQL,...

Tập hợp tất cả các quan hệ được hình thành từ mô hình tổ chức dữ liệu, các file phục vụ cho hoạt động của hệ thống được gọi là mô hình vật lý về dữ liệu của hệ thống thông tin.

5.1.2 Thiết kế các trường

Ở mức vật lý, một trường được đồng nhất với một thuộc tính trong mô hình tổ chức dữ liệu. Trường là đơn vị dữ liệu nhỏ nhất một phần mềm hệ thống nhận ra.

- *Các yêu cầu về việc thiết kế các trường*
 - Tiết kiệm không gian nhớ
 - Biểu diễn được mọi giá trị có thể
 - Cài đặt các ràng buộc toàn vẹn của dữ liệu
 - Đặt giá trị mặc định (Default) để giảm thiểu thời gian nhập dữ liệu

- *Chọn kiểu dữ liệu và độ rộng của trường*

Hầu hết các hệ quản trị cơ sở dữ liệu thương mại đều cung cấp cho người dùng các kiểu dữ liệu thông dụng như: text, number, logical, date, time,... Khi chọn kiểu dữ liệu và độ rộng trường nên chọn đúng kiểu và khai báo độ rộng vừa đủ, không nên làm phức tạp cấu trúc dữ liệu của hệ thống.

5.1.2 Thiết kế các file

Một hệ thống thông tin hoạt động thường sử dụng sáu loại file dưới đây:

- ***File dữ liệu (data file)***: file chứa các dữ liệu nghiệp vụ liên quan đến mô hình logic dữ liệu và mô hình vật lý dữ liệu. Loại file này luôn tồn tại và có nội dung thay đổi. Ví dụ, file chứa các thông tin về khách hàng, file chứa các thông tin về sách trong thư viện,...
- ***File tham chiếu từ bảng (lookup table file)***: file chứa các dữ liệu được lấy từ các bảng dữ liệu. Những file này thường sử dụng trong các trường hợp lấy dữ liệu nhanh để kết xuất thông tin.
- ***File giao dịch (transaction file)***: là file dữ liệu tạm thời phục vụ cho các hoạt động hằng ngày của tổ chức. File này thường được thiết kế để phục vụ việc xử lý nhanh các tình huống có thể xảy ra.
- ***File làm việc (work file)***: file tạm thời để lưu kết quả trung gian, file này tự động xoá đi khi không cần thiết.
- ***File bảo vệ (protection file)***: file được thiết kế để lưu trữ các file khác nhau có nguy cơ bị sai hỏng trong quá trình làm việc.
- ***File lịch sử (history file)***: file chứa những dữ liệu cũ hiện không sử dụng, nhưng có thể sử dụng để làm một việc gì đó khi cần thiết.

5.1.3 Các hệ quản lý file

File là đơn vị lưu trữ của bộ nhớ ngoài dưới một hệ điều hành nào đó. Mọi thông tin lưu trên bộ nhớ ngoài đều được tổ chức thành từng file. Về bản chất

thông tin, file có thể là văn bản, chương trình, dữ liệu,... nhưng dù thế nào chúng chỉ là dãy các bit dữ liệu.

Quản lý file là thực hiện các thao tác như lưu trữ, tìm kiếm, di chuyển, xóa, thiết lập thuộc tính cho file. Mặc dù các thao tác này được thực hiện thông qua hệ điều hành nhưng trên thực tế có nhiều phần mềm được sử dụng để quản lý các file dễ dàng và tiện lợi hơn. Chú ý rằng không có sự tương hợp giữa kích thước file và bộ nhớ trong nên khi đọc/ghi một file hệ điều hành sử dụng chiến lược bộ nhớ đệm để lưu hình ảnh của file hoặc một đoạn của file đó trong bộ nhớ này.

5.1.4 Các cấu trúc dữ liệu và phương thức truy nhập

a. Tổ chức file tuần tự và truy nhập tuần tự:

Các bản ghi trong file được sắp xếp liên tiếp nhau. Việc truy nhập đến một nơi nào đó trong file được thực hiện theo thể thức duyệt lần lượt cho đến khi gặp bản ghi cần tìm. Cách này thường mất thời gian nhưng trong một số trường hợp là cách duy nhất để tìm kiếm thông tin.

b. Truy nhập ngẫu nhiên theo hàm băm

Trong trường hợp này các bản ghi được chia thành nhiều khối có độ dài như nhau và người ta xây dựng một hàm băm cho phép tính địa chỉ của khối dữ liệu chứa bản ghi theo khóa của bản ghi đó.

c. Truy nhập theo file chỉ mục

Các bản ghi các file có thể sắp xếp tùy ý. Một file chỉ mục được tạo ra cho phép xác định được vị trí của mỗi bản ghi cụ thể trong file gốc. Nhược điểm của phương pháp này là phải tốn không gian để lưu file chỉ mục

5.1.5 Thiết kế kiểm soát các file

Nhằm bảo vệ dữ liệu và chống lại sự phá hủy của người khác thông thường người ta sử dụng hai biện pháp kỹ thuật là sao lưu và mã hóa file dữ liệu.

a. Thủ tục sao lưu file

Các file quan trọng cần được lưu trữ vào một thiết bị riêng theo một chu kỳ được xác định, khi cần sẽ lấy ra để sử dụng. Từ việc nghiên cứu hệ thống, hoặc từ kinh nghiệm chúng ta có thể quyết định các file nào cần sao lưu. Việc tổ chức sao lưu cũng có thể thực hiện bởi phần mềm trợ giúp, phần mềm này có nhiệm vụ nhắc nhở người sử dụng công việc sao lưu.

b. Đặt mật khẩu cho chương trình và mã hoá nội dung file

Nhằm bảo đảm an toàn nội dung các file, nhất là một số ứng dụng về quân sự, tài chính,... thông thường người ta đặt mật khẩu (*password*) hoặc mã hóa nội dung file. File chỉ có thể được mở ra để làm việc nếu người sử dụng đưa dung mật khẩu. Mã hóa nội dung file là chuyển cách biểu diễn dữ liệu của file sang một dạng khác. Nhiều hệ điều hành và phần mềm quản lý dữ liệu đã cung cấp công cụ mã hóa và giải mã dữ liệu.

5.1.6 Xác định quy mô file và không gian lưu trữ cần thiết

Một ví dụ về thiết kế file dữ liệu

Trong hệ thống thông tin “***Quản lý kho hàng***” chúng ta đã có mô hình tổ chức dữ liệu của hệ thống là các quan hệ sau:

Nhà CC	(<u>Mã NCC</u> , Tên NCC, Đchỉ NCC)
Kho	(<u>Tên kho</u> , Đchỉ kho)
Khàng	(<u>Mã khách</u> , Tên khách, Đchỉ khách)
Phiếu nhập	(<u>Số phiếu N</u> , Ngày nhập, <u>Mã.NCC</u>)
Phiếu xuất	(<u>Sốphiếu X</u> , Ngày xuất, <u>Mãkhách</u>)
Hàng	(<u>Mãhàng</u> , Tênhàng, Đơnvị, Đơngiá, <u>Tên.kho</u>)
Gồm hàng_N	(<u>Sốphiếu N</u> , <u>Mãhàng</u> , SL_nhập)
Gồm hàng_X	(<u>Sốphiếu X</u> , <u>Mãhàng</u> , SL_xuất)
Chứa	(Tồn kho, <u>Tên.kho</u> , <u>Mã hàng</u>)

Dựa vào các khảo sát trước đây và các quan hệ trên hãy mô tả chúng dưới dạng các khai báo sau:

NHA_CC

Fieldname	Data type	Field size	Format	Validation Rule
MA_NCC (K)	Text	2	Chữ hoa	Len()=2
TEN_NCC	Text	30	Chữ đầu viết hoa	Not null
ĐCHI_NCC	Text	50	Chữ đầu viết hoa	

KHO

Fieldname	Data type	Field size	Format	Validation Rule
TENKHO (K)	Text	8	Chữ hoa	
DCHI_KHO	Text	25	Chữ đầu viết hoa	

KHHANG

Fieldname	Data type	Field size	Format	Validation Rule
MAKHACH (K)	Text	3	Chữ hoa	Len()=3
TENKHACH	Text	30	Chữ đầu viết hoa	Not null
ĐCHI_KHACH	Text	50	Chữ đầu viết hoa	

PHIEUXUAT

Fieldname	Data type	Field size	Format	Validation Rule
SOPHIEU_X (K)	Text	8	Chữ số	Len()=8
MAKHACH (FK)	Text	3	Chữ hoa	Len()=3
NGAYXUAT	Date	8	dd-mm-yy	

PHIEUNHAP

Fieldname	Data type	Field size	Format	Validation Rule
SOPHIEU_N (K)	Text	8	Chữ số	Len()=8
MA_NCC (FK)	Text	2	Chữ hoa	Len()=2
NGAYNHAP	Date	8	dd-mm-yy	

HANG

Fieldname	Data type	Field size	Format	Validation Rule
MAHANG (K)	Text	4	Chữ hoa+Chữ số	Len()=6
TENHANG	Text	30	Chữ đầu viết hoa	Not null
DONVI	Text	6	Chữ đầu viết hoa	
DONGIA	Num	7	Số nguyên	
TENKHO (FK)	Text	8	Chữ hoa	

HANGNHAP

Fieldname	Data type	Field size	Format	Validation Rule
SOPHIEU_N (K)	Text	8	Chữ số	Len()=8
MAHANG (K)	Text	4	Chữ hoa+Chữ số	Len()=6
SL_NHAP	Num	4	Số nguyên	

HANGXUAT

Fieldname	Data type	Field size	Format	Validation Rule
SOPHIEU_X (K)	Text	8	Chữ số	Len()=8
MAHANG (K)	Text	4	Chữ hoa+Chữ số	Len()=6
SL_XUAT	Num	4	Số nguyên	

CHUA

Fieldname	Data type	Field size	Format	Validation Rule
TENKHO (K)	Text	8	Chữ hoa	
TONKHO	Num	6	Số nguyên	
MAHANG (K)	Text	4	Chữ hoa+Chữ số	Len()=6

5.2 Mô hình vật lý về xử lý (mức tác nghiệp)

5.2.1 Mục đích:

Mô hình này trả lời cho câu hỏi cuối cùng là: các công việc hoạt động như thế nào? Từ mô hình tổ chức xử lý đã có, người phân tích sẽ tiến hành xem xét, biến các chức năng, công việc thành các đơn vị chương trình. Ứng với mỗi đơn vị chương trình này người phân tích phải viết một đặc tả chi tiết để chuẩn bị cho việc lập trình.

5.2.2 Mô đun xử lý

Mô đun xử lý là thể hiện các công việc có liên quan với nhau và được thực hiện liên mạch nhằm thực hiện một chức năng nào đó. Nói chung tiêu chuẩn để xác định một mô đun xử lý khá mờ, nó chỉ nêu lên phương hướng phân rã chức năng mà không xác định chính xác quy mô của các mô đun. Thông thường một mô đun xử lý thể hiện một công đoạn có bản chất là cập

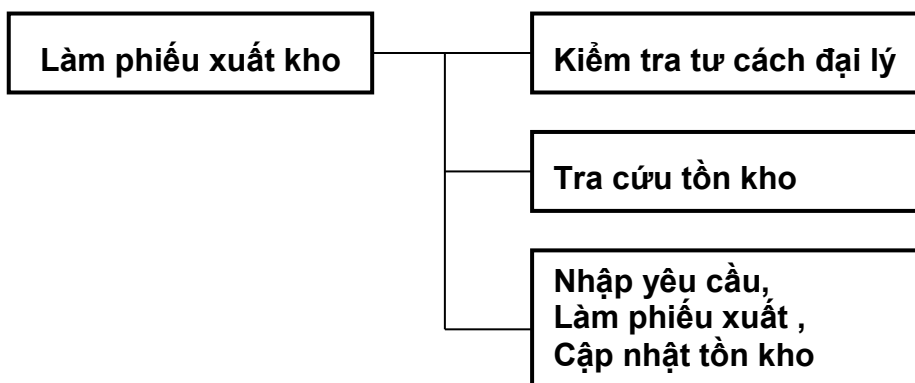
nhật hoặc tra cứu dữ liệu và thao tác trên một nhóm dữ liệu nhỏ.

Ví dụ, Chức năng *làm phiếu xuất kho* sẽ bao gồm các mô đun sau:

- Tra cứu danh sách các đại lý để kiểm tra khách hàng
- Kiểm tra hàng tồn kho
- Lấy yêu cầu để lập phiếu xuất và cập nhật tồn kho

5.2.3 Phân rã mô đun

Để dễ dàng trong việc mã hoá, cài đặt chương trình và sửa chữa chương trình, người ta phân rã một mô đun thành nhiều mô đun con. Một mô đun con phân rã đến lúc không thể tách thêm được nữa được gọi là mô đun sơ cấp. Tuy nhiên, việc phân rã này phải bảo đảm mối liên hệ giữa mô đun lớn với các mô đun con. Trong thực tế thường xảy ra trường hợp phân rã mô đun nhỏ đến một mức nào đó có thể xuất hiện các mô đun chung, điều này sẽ giảm nhẹ công sức lập trình sau này. Phân rã mô đun cũng gợi ra giao diện chọn chức năng theo kiểu thực đơn trong chương trình tổng thể sau này. Để mô tả việc phân rã mô đun thành nhiều mô đun con, người ta dùng sơ đồ phân rã chức năng như sau:



Các yếu tố để phân rã mô đun

a. *Phân rã mô đun theo điểm công tác*: điều này thể hiện ở chỗ nhiều người hoặc nhiều bộ phận có những công việc như nhau, như thế các chức năng có cùng một nơi làm việc được gom thành một mô đun. Ví dụ, các thông tin về nâng bậc lương, chuyển ngạch,... không những cần cho bộ phận tổ chức mà

còn cho bộ phận kế toán. do đó phải có một mô đun chung để cập nhật, tra cứu các thông tin này.

b. Phân rã mô đun theo hướng chức năng: theo cách này các chức năng có cùng chung một công việc được tổ chức riêng.

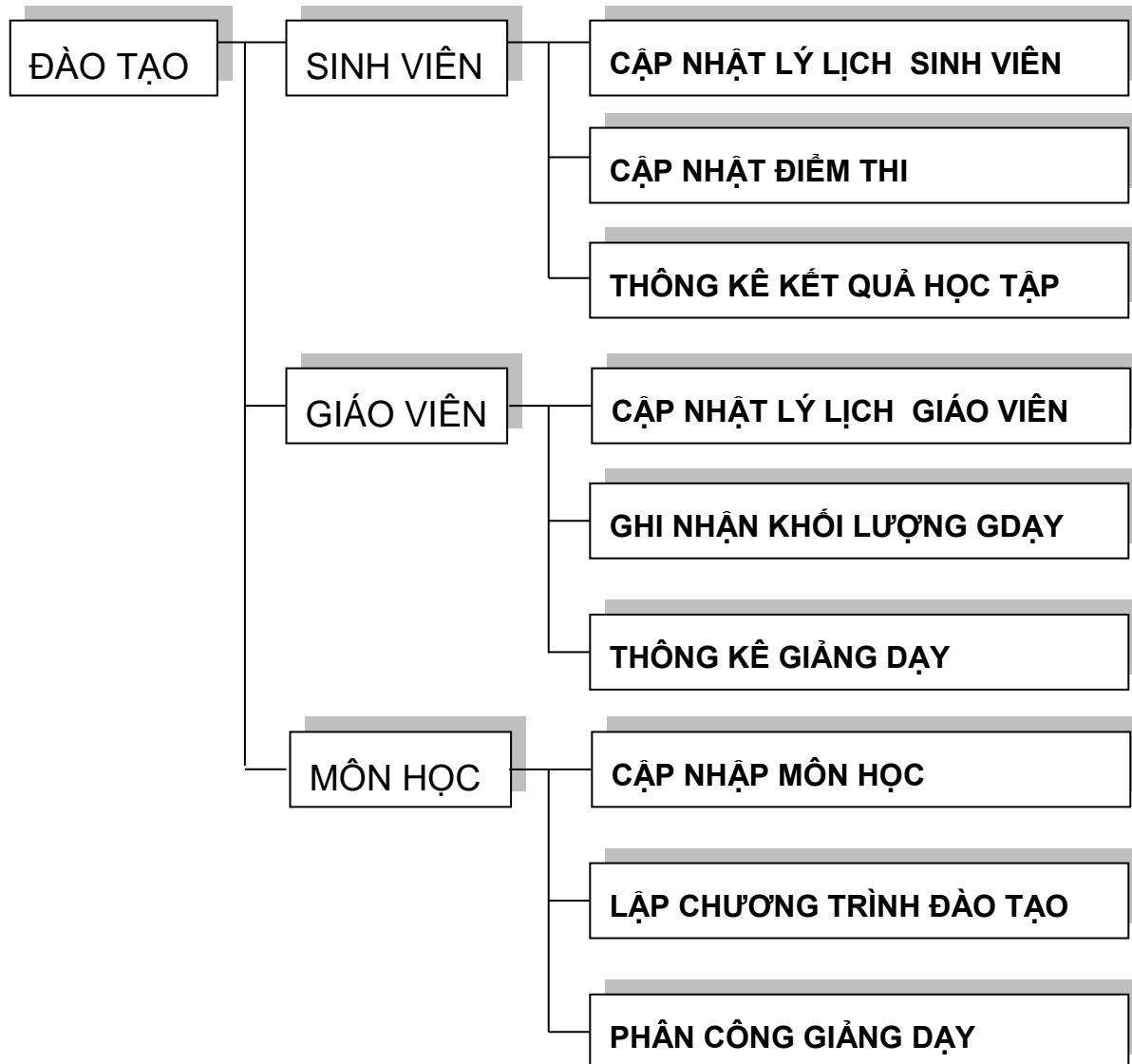
c. Phân rã mô đun theo thời gian: thời gian cũng có thể một yếu tố để phân rã mô đun. Ví dụ, việc in báo cáo kết quả học tập của sinh viên được thực hiện vào cuối năm học với hàng loạt các báo cáo khác như báo cáo khối lượng công tác của giáo viên,...

5.2.4 Sơ đồ tổng thể phân rã chức năng:

Dựa trên kết quả phân rã mô đun, người phân tích phải lên một sơ đồ tổng thể các chức năng để hướng đến cấu trúc hoá chương trình. Hiện nay có một vài quan điểm về việc gộp các mô đun thành từng nhóm chức năng trong chương trình.

a. Gộp các mô đun theo hướng đối tượng:

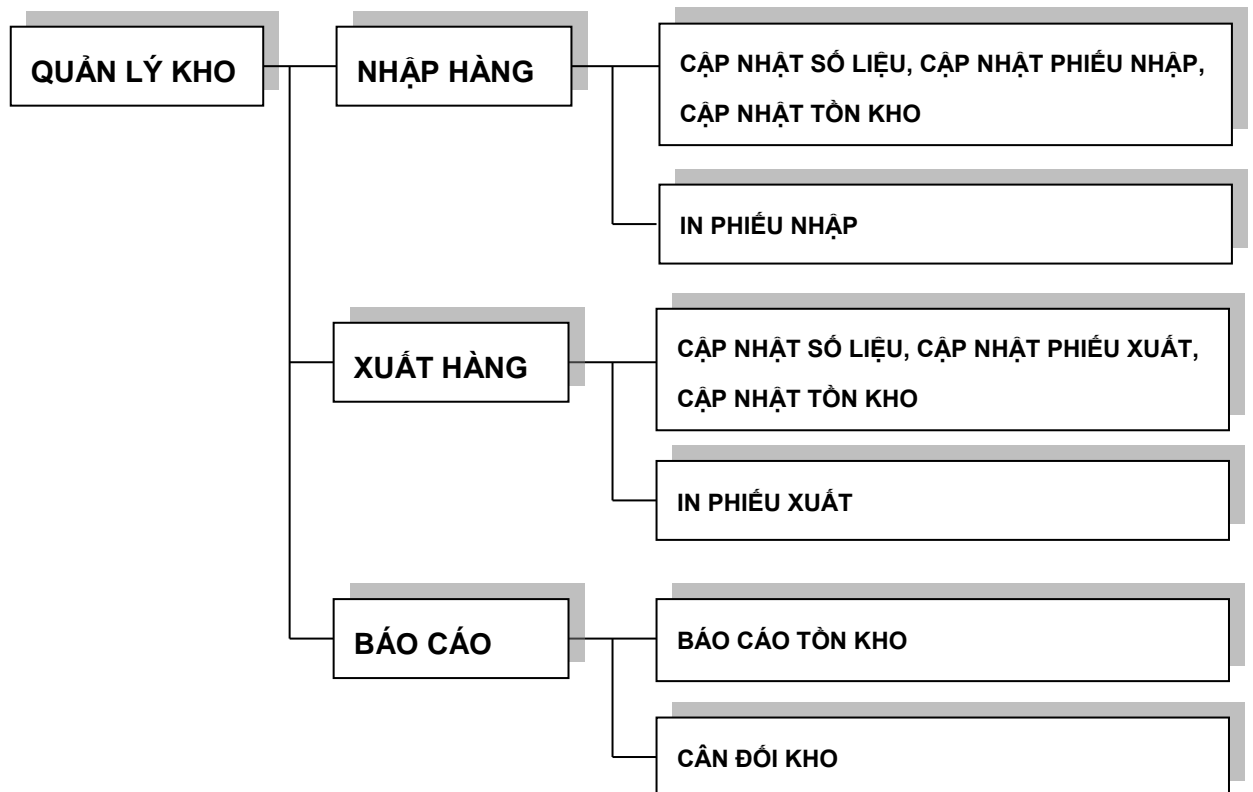
Gộp theo đối tượng là nhóm các chức năng theo dữ liệu hoặc theo tập thực thể. Ví dụ, ba tập thực thể chính trong hệ thống thông tin “Quản lý đào tạo” là sinh viên, giáo viên và môn học. Chúng ta có thể gộp các mô đun theo các tập thực thể này theo sơ đồ sau:



Hình 5.2a. Gộp các chức năng theo đối tượng

b. Gộp các mô đun theo sự kiện:

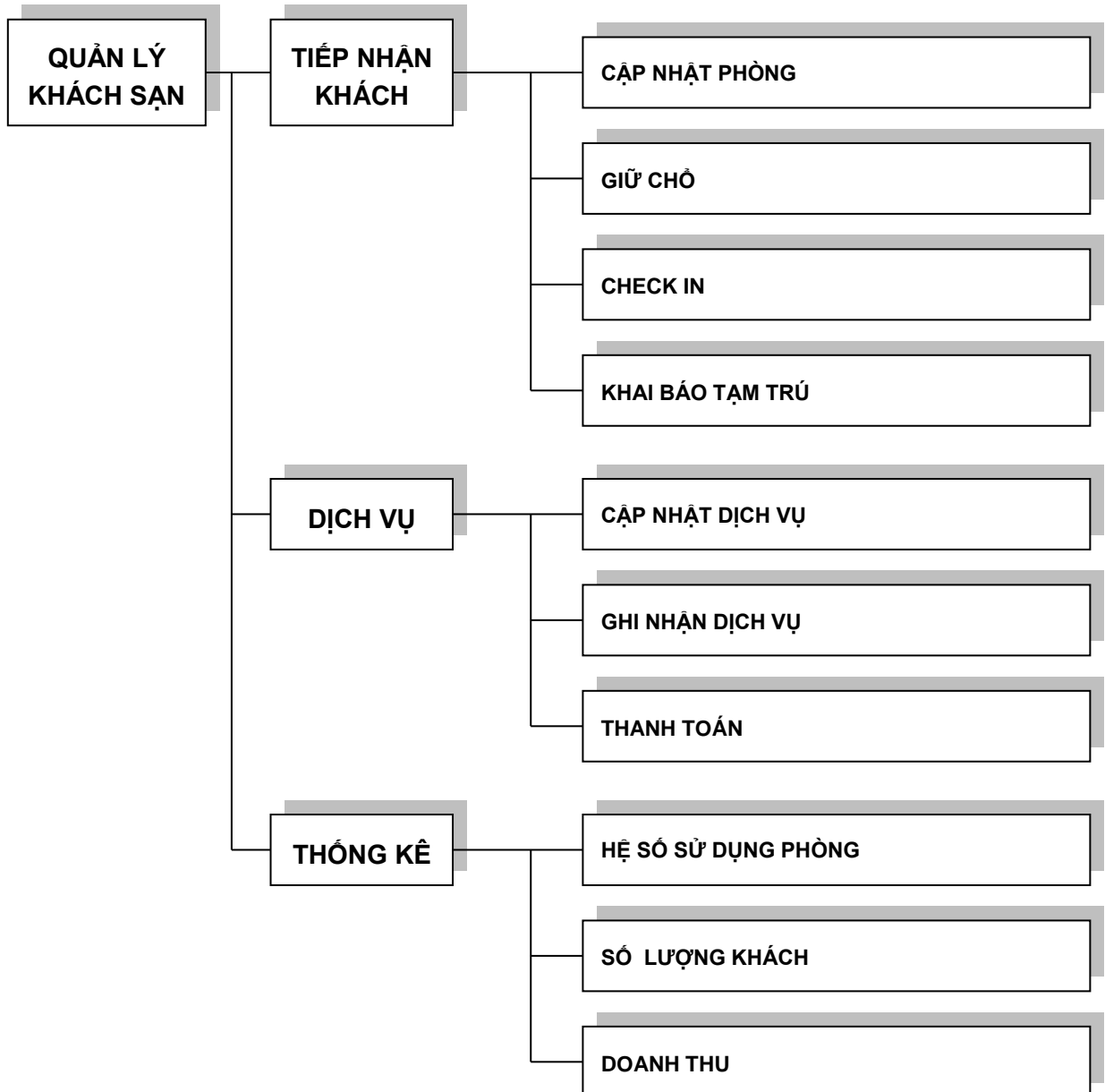
Gộp theo sự kiện là gộp theo hoạt động của hệ thống. Một sự kiện có thể gây ra một loạt các chức năng của hệ thống. Ví dụ, trong hệ thống thông tin “Quản lý kho hàng” có ba sự kiện chính là “Nhập hàng”, “Xuất hàng” và “Báo cáo”. Chúng ta có thể gộp theo sự kiện các mô đun này theo sơ đồ dưới đây.



Hình 5.2.b Gộp các chức năng theo sự kiện

c. Gộp các mô đun theo sự tiện lợi:

Gộp theo sự tiện lợi là gộp các mô đun theo tiêu chuẩn tiện dụng hoặc theo người sử dụng cụ thể hoặc theo mạch công việc. Ví dụ, trong hệ thống thông tin “Quản lý khách sạn” thường có các mạch công việc như sau: *Tiếp nhận khách* bao gồm các công việc: Cập nhật phòng, Giữ chỗ, Check in. *Dịch vụ* bao gồm các công việc: cập nhật dịch vụ, ghi nhận dịch vụ, thanh toán. *Thống kê* bao gồm các công việc: Hệ số sử dụng phòng, Số lượng khách, Doanh thu. Chúng ta có thể tổ chức các mô đun theo mạch công việc như sau:



Hình 5.2.c. Gộp các chức năng theo mạch

5.2.5 Mô tả các mô đun

Sau khi phân rã các mô đun, người phân tích phải chuyển giao các kết quả phân tích thiết kế cho người lập trình để chuẩn bị cài đặt. Các mô đun này phải được mô tả một cách chi tiết thông qua các biểu đồ được gọi là IPO Chart như sau:

IPO CHART		Số: _____
Name of modun: <tên mô đun>	Date: <ngày tạo môđun>	
System: <tên hệ thống thông tin>	Designer: <người thiết kế>	
Objective: <mô tả mục đích của mô đun>		
Call by: < danh sách các mô đun gọi mô đun này>	Call: < danh sách các mô đun mà mô đun này sẽ gọi>	
Input: < danh sách các tham biến và dữ liệu vào>	Output: < danh sách các tham biến và dữ liệu ra>	
Processing: < mô tả chi tiết quá trình xử lý bằng thuật giải rõ ràng>		

Ví dụ: Mô đun Nhập dữ liệu cho bảng Huyện trong hệ thống thông tin “Quản lý công chức”

IPO CHART		Số: _____
Name of modun: Nhập Huyện	Date: 01/01/2005	
System: Quản lý công chức	Designer: Nguyễn Mậu Hân	
Objective: Nhập dữ liệu cho bảng Huyện		
Call by: Main Menu	Call: None	
Input: Bảng Tỉnh, Huyện	Output: Bảng Huyện	
Processing: Tạo một Form nhập dữ liệu cho bảng Huyện. Trong Form tạo một Combo box để chọn Mã tỉnh, Mã tỉnh là khoá của bảng Tỉnh và là FK của bảng Huyện.		