

BÀI GIẢNG MÔN

THIẾT KẾ CƠ SỞ DỮ LIỆU

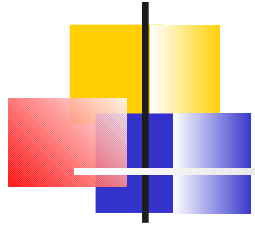


BÀI 1: PHÂN TÍCH HỆ THỐNG CÓ CẤU TRÚC

I. Khái niệm

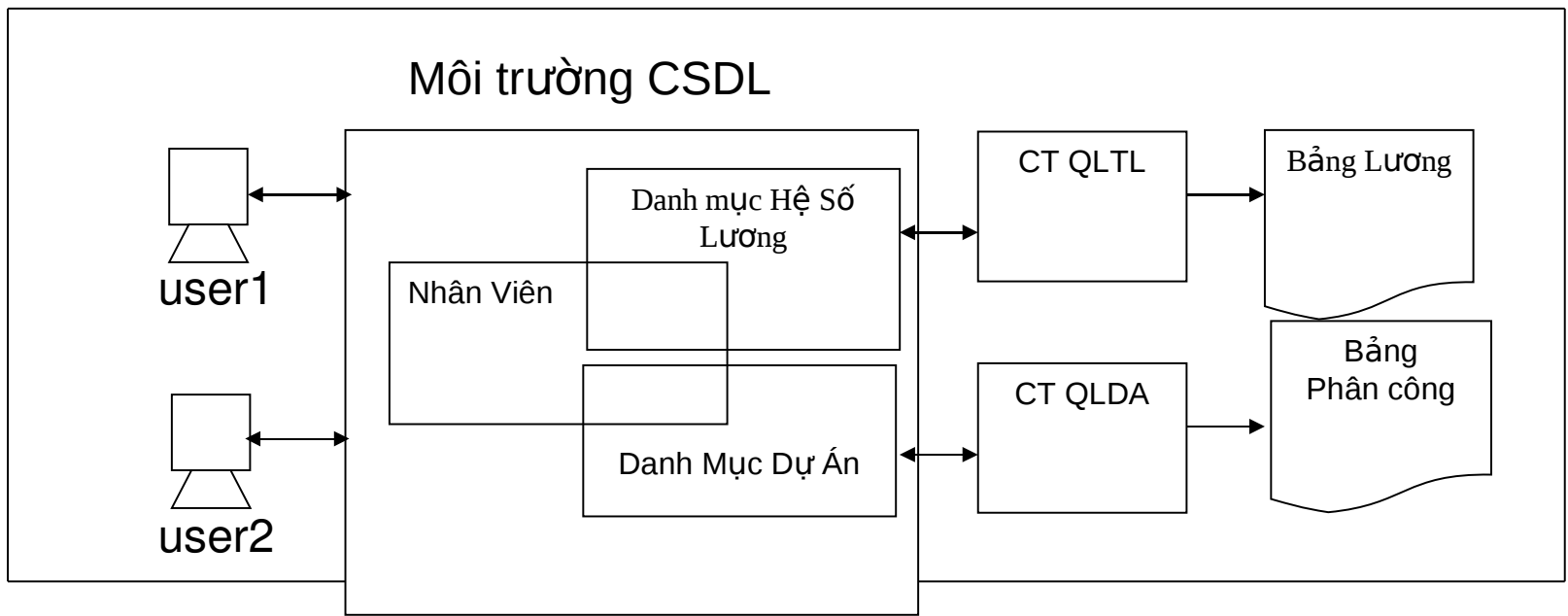
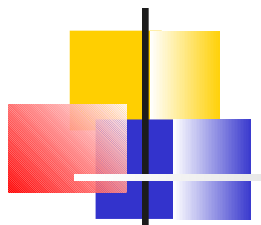
1. Khái niệm về hệ thống CSDL

Hệ thống CSDL của một hệ thống tin học là một tập hợp dữ liệu được tổ chức một cách chọn lọc lưu trên các thiết bị trữ tin, nhằm phục vụ đồng thời cho nhiều người, với nhiều mục đích xử lý và khai thác khác nhau.



Ví dụ: **Trong một công ty phần mềm:**

- Bộ phận quản lý tiền lương có nhu cầu lập bảng lương cho đơn vị với các thông tin ghi trên bảng lương như sau: STT, họ tên, hệ số lương, tiền lương, Chữ ký
- Trong đó, Tiền lương = hệ số lương x 500000; hệ số lương được phân chia dựa trên học vị.
- Bộ phận quản lý dự án có nhu cầu lập danh sách phân công nhân viên cho các dự án, với các thông tin: STT, họ tên, chuyên môn, dự án.
- Trong đó, nhân viên được phân công phải có chuyên môn phù hợp với yêu cầu chuyên môn của từng dự án



Hình ảnh về môi trường CSDL

Hệ thống CSDL được xây dựng sao cho có thể phục vụ cho các mục tiêu trên của các phòng ban



2. Mục tiêu chính công việc thiết kế CSDL

- Làm thế nào chuyển đổi các nhu cầu lên trở về khai thác dữ liệu của người sử dụng thành một hệ thống CSDL hiệu quả. Tính hiệu quả đã được thể hiện cụ thể bởi các tính chất : “Tính không trùng lặp”; “Tính nhất quán dữ liệu”; “Tính dễ khai thác”; “Dễ kiểm tra các qui tắc quản lý bởi các ràng buộc toán học”; “Dễ cập nhật và nâng cấp hệ thống”.
- Vôùi cương các nhu cầu lên trở về



Ví dụ:

CT1: NhanVien(MaNV, HoTen, ChuyenMon, HSLg,
TienLuong, ChuKy)

DanhMucDuAn(MaDA, TenDuAn,...)

CT2: NhanVien(MaNV, HoTen, ChuyenMon, Hocvi)

DMHSLuong(Hocvi, HeSoLuong)

DanhMucDuAn(MaDA, TenDuAn,...)

PhanCong(MaDA, MaNV)



3. Cauc thông tin vaø / ra quy trình thieát keá.

Thoàng tin vaø:

(1) Yeâu caàu veà thông tin: Duøng CSDL cho vaán ñeà gì? Xuaát phaùt töø ngôøi söù duïng coù nhu caàu vaø quan ñieäm nhö theá naø. Ta caàn phaûi ghi nhaän laïi heát.

Ôu ñaây chæ giôùi haïn ôu möùc döõ lieäu.

(2) Yeâu caàu veà xöù lyù: Moãi nhóm ngôøi söù duïng seõ neâu ra cauc yeâu caàu xöù lyù cuûa rieâng mình; Taàn suaát xöù lyù vaø khoái löõing döõ lieäu.

(3) Ñaéc tröng kyõ thuaät cuûa heä quaûn trò CSDL caàn söù duïng ñeã caøi ñaët CSDL

(4) Caáu hình thieát bò tin hoïc gì ñeã ñaùp öùng vöù (1), (2) vaø (3)

Thoàng tin ra:

(1) Caáu truùc quan nieäm CSDL

(2) Caáu truùc Logic CSDL

(3) Caáu truùc Vaät lyù CSDL



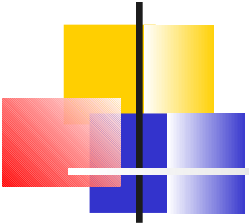
4. Chu kyø soáng cuûa moät CSDL

Moät öùng düng tin hoïc ñöôïc trieån khai thöïc hieän traûi qua caùc giai ñoaïn:

(i) **Giai ñoaïn xaây döïng CSDL**

- (a) Phaân tích caùc nhu caàu cuûa ngôøi söu düng
- (b) Thieát keá CSDL ôû möùc quan nieäm: nghóa laø xaùc ñònh noäi dung CSDL (chöùa nhöõng thông tin gì?). Chæ quan tâm ôû möùc döõ lieäu
- (c) Thieát keá CSDL ôû möùc Logic: Chia vaán ñeà caàn xöû lyù ra thaønh nhieàu böôùc. Ôû ñây chæ chuù yù ñeán caùc xöû lyù ñaët ra, nhöng chöa chuù yù ñeán phaàn meàm vaø phaàn cöùng.
- (d) Thieát keá CSDL ôû möùc vaät lyù: Caøi ñaët CSDL nhö theá naøo? Giaûi quyeát nhöõng vaán ñeà mang tính kyõ thuaät.

Ví duï: Söu düng phaàn meàm naøo? Vôùi caáu hình maùy

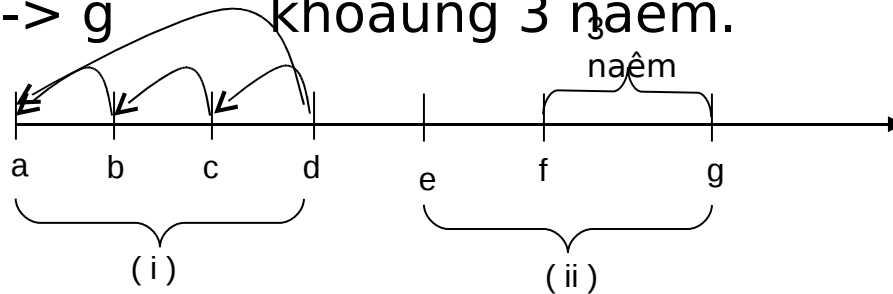


(ii) Giai ñoain thöu nghiäm vaø khai thaùc:

(e) Caøi ñaët vaø chaïy thöu nghiäm: Neáu coù sai soùt thì phaûi hieäu chænh laïi caáu truùc CSDL ôû caùc möùc quan nieäm; logic; vaät lyù.

(f) Ñöa cho ngôøi söu duïng khai thaùc.

(g) Thích öùng CSDL theo nhöõng nhu caàu möù: baét ñaàu töø f --> g khoaûng 3 ñaêm.



Quaù trình thieát keá laø chu trình soáng, neáu nhu caàu möù quaù nhieàu thì caàn phaûi chuaån bò CSDL möù ñeã thay theá CSDL cuõ.



5. Qui Trì nh Thi éát Keá CSDL

a. Giai ñoăin phaân tích nhu caàu:

i. Noăi dung:

- Thu thaăp thoâng tin veà döõ lieäu vaøxöû lýù töø ngöôøi söû duïng, töø caùc taøi lieäu, chöùng töø, bieău maãu thoáng keá lieân quan ñeán CSDL vaø caù nhöõng taøi lieäu cuûa CSDL cuõ (Neáu coù).
- Sau khi thu thaăp phaûi toảng hôïp vaø phaân tích nhöõng nhu caàu ñoù. Kieám tra xem coù nhöõng maâu thuaãn giöõa caùc nhu caàu khoâng



Ví dụ:

- - Tình trạng bán vé trong các chuyến bay, chuyến tàu nội địa phải xử lý tức thời, riêng rẽ tổng trợợng hợp.
- - Tình trạng möin, trâu saùch cuôa ñoãc giaù thõ vieãn ñoøi hoùì phaùì xử lý riêng rẽ nhõng thõì gian xử lý còu theỏ treỏ.
- - Tớnh löõng cho công nhân ñoøi hoùì xử lý chung toaøn boỏ vaø thõì gian xử lý theo ñònh kyø giõõa thàùng hay cuoái thàùng.



ii. Cách thực hiện:

Dùng kỹ thuật phỏng vấn:

- Tiếp cận
 - Giàn tiếp: tôi lập ra các câu hỏi trên giấy để User trả lời.
- Nội dung phỏng vấn: có liên quan
- Ban giám đốc
 - Các phòng ban có liên quan



b. Giai ñoãïn thieát keá quan
nieãm:

i. Muïc ñích:

Xaùc ñònh noãi dung döõ lieäu, moái quan heã
giöõa caùc döõ lieäu beãn trong CSDL.

Chöa caàn quan taâm caùch caøi ñaët. Phaûi xaùc
ñònh ñuùng vaø ñaày ñuû döõ lieäu, loaïi boû
caùc döõ lieäu thöøa.



ii. Cách thức hiện:

Do nhu cầu khai thác, mỗi nhóm người sẽ có những yêu cầu khác nhau về CSDL.

Ví dụ:

- Nói với người quản trò kinh doanh cần quan tâm đến các thành phẩm: Mã thành phẩm, tên, số lượng tồn, tồn giá bán.
- Nói với người quản lý kho: ngoại thông tin của các thành phẩm, người quản lý kho cần quan tâm đến các chứng tờ liên quan đến các thành phẩm: Số nội, giá thành, số lượng



c. Giai ñoạn thiết kế logic.

i. Mục đích:

Đây là bước chuyển tiếp. Cần biết cần nhắc đến trên nhu cầu xử lý, nghiên cứu cách sử dụng dữ liệu thông qua xử lý

Các thông tin cần: Tài suất, khối lượng ...

Trong giai đoạn thiết kế quan niệm, dữ liệu cần loại bỏ những thông tin trùng lặp. Những ưu giai đoạn thiết kế logic, cần phải cần nhắc, đưa trên hiệu quả xử lý, nên quyết định có hay không có các thiết kế thông tin trung lập



ii. Cách thức hiện:

- Chọn cấu trúc logic gần với phần mềm sẽ sử dụng các ãt CSDL.
- Ở giai ão này , người ta thường thể ã hiện thông tin theo mô hình Quan hệ.



d. Giai ñoạiñ thieát keá vaät lyù:

i. Muïc ñích:

Xây döïng möät caáu truùc vaät lyù phuï thuoäc vaø phaàn meàm vaø caáu hình phaàn cöùng maø ta ñaõ löïa choïn ñeå caøi ñaët CSDL.

Giai ñoạiñ naøy, ñôn giaûn hay phöùc taïp tuøy thuoäc vaøo ñaëc tröng kyõ thuaät cuûa phaàn meàm vaø phaàn cöùng.

ii. Caùch thöïc hieän:

- Choïn löïa phaàn meàm phuø hôïp vöùi ñoä phöùc taïp cuûa döï aùn
- Choïn löïa caáu hình phaàn cöùng
- Quyeát ñònh nhöõng vaán ñeà lieân quan ñeán An toaøn döõ lieäu vaø phuïc hoài döõ lieäu.



I. Các yêu cầu đối với một hệ CSDL

Một hệ cơ sở dữ liệu phải đảm bảo các yêu cầu sau:

1. Giảm thiểu dữ liệu thừa
2. Bảo đảm tính toàn vẹn của dữ liệu
3. Bảo đảm tính độc lập dữ liệu
4. Bảo vệ an toàn dữ liệu
5. Duy trì tính nhất quán của dữ liệu
6. Tạo dễ dàng khi sử dụng dữ liệu
7. Tiết kiệm không gian lưu trữ



1. **Giaûm thieäu dö thøa dö lieäu:**

Dö lieäu thøa seõ ñöôic giaûm thieäu nhø nhöõng tham chieäu cheù trong heã thoáng CSDL.

Thí du: Teân vaø ñò chæ nhaân vieân chæ caàn xuaát hieän möt laàn trong CSDL

2. **Ñâm baùu tí nh an toaøn dö lieäu:**

Bôui vì dö lieäu chæ xuaát hieän möt laàn trong CSDL, chùng ta coù theã yeân tâm raèng möi thay ñoãi trên dö lieäu ñeàu coù taùc düng veø sau khi söü düng dö lieäu. Cháu ng haïn chùng ta caàn thay ñoãi ñò chæ cuõa möt nhaân vieân tại möt nô duy nhaát trong CSDL vaø nhöõng thay ñoãi naøy seõ hieän höõu khi caàn truy vaán ñeän ñò chæ sau



3. Nũm baũu tí nh ñoăc l aăp döũ l i eău:

* Caáu truùc cuũa CSDL ñoøi hoũi döũ lieău phaũi ñoăc laăp vòuì nhuõng döũ lieău khaùc vaø ñoăc laăp vòuì phaàn meàm duøng ñeă truy xuaát döũ lieău. Neáu chuùng ta söũa ñoăi hoaëc xoaù döũ lieău, caùc döũ lieău khaùc seõ khoâng bò aũnh hööuũg.

Ví duĩ: Chaúng haĩn khi xoaù moät nhaân vieân thoai vieăc khoũi CSDL seõ khoâng laøm aũnh hööuũg ñeán döũ lieău khaùc. Khi xoaù moät nhaân vieân ra khoũi taăp tin nhaân vieân, chuùng ta khoâng ñöôic laøm maát caùc thoũng tin veà löông ñaõ traũ, tieàn thueá thu nhaăp. Hôn nõõa, döũ lieău veà phoøng laøm vieăc cuũa nhaân vieân naøy, soá ñieãn thoăi vaø vò trí coũng taùc . . .cuõng khoâng bò aũnh hööuũg.



4. Bảo vệ an toàn dữ liệu:

* Hệ thống CSDL còn có những hệ thống bảo mật nữa nhằm bảo vệ an toàn dữ liệu. Phần lớn những hệ thống CSDL đều duy trì hệ thống bảo mật của riêng chúng. Khi dữ liệu nội bộ lưu trữ tập trung, việc quản lý dữ liệu khá dễ dàng.

* Những hệ thống CSDL ẩn hình thông tin còn những thủ tục truy xuất riêng. Trước khi một người nội bộ truy xuất vào CSDL, anh ta phải có một tài khoản người dùng (user account) trên hệ thống đó. Sau khi người đó trao quyền truy xuất chung trong CSDL, anh ta người quản trị viên CSDL (database administrator, viết tắt là DBA) cấp quyền người xem ẩn hình (View) của CSDL. Người sử dụng chỉ người nội bộ sử dụng một số phần hình, một số lệnh để truy xuất dữ liệu trong một số tập tin, số mẫu tin hoặc thông tin. Ở đây DBA là một cá nhân hay một nhóm chịu trách nhiệm chính trong công việc quản lý dữ liệu.

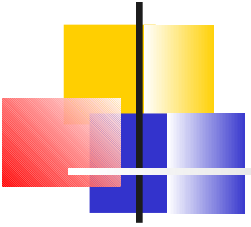


5. Duy trì tính nhất quán của dữ liệu:

Liệu:

Tính nhất quán của dữ liệu được duy trì trong môi trường CSDL. Tên và kích thước dữ liệu nên được thống nhất trong mọi ứng dụng.

Ví dụ: Chẳng hạn, kích thước của trường `FIRST_NAME` nên nhớ nhau trong mọi ứng dụng cơ sở dữ liệu nên CSDL này. Nếu `FIRST_NAME` được khai báo dài 20 ký tự, CSDL sẽ luôn truy xuất nó như một trường dài 20 ký tự.



6. Tại sao dễ dàng khi sử dụng dữ liệu I i eäu:

Trong môi trường CSDL sẽ dễ dàng sử dụng dữ liệu hơn. Phần lớn CSDL quan hệ đều chứa một ngôn ngữ vấn tin thân thiện và kèm với phần mềm quản trị CSDL. Trong những loại CSDL này, cần một ít thời gian huấn luyện, người sử dụng dễ dàng tạo các câu vấn tin trên CSDL mà không cần sự can thiệp của nhóm lập trình

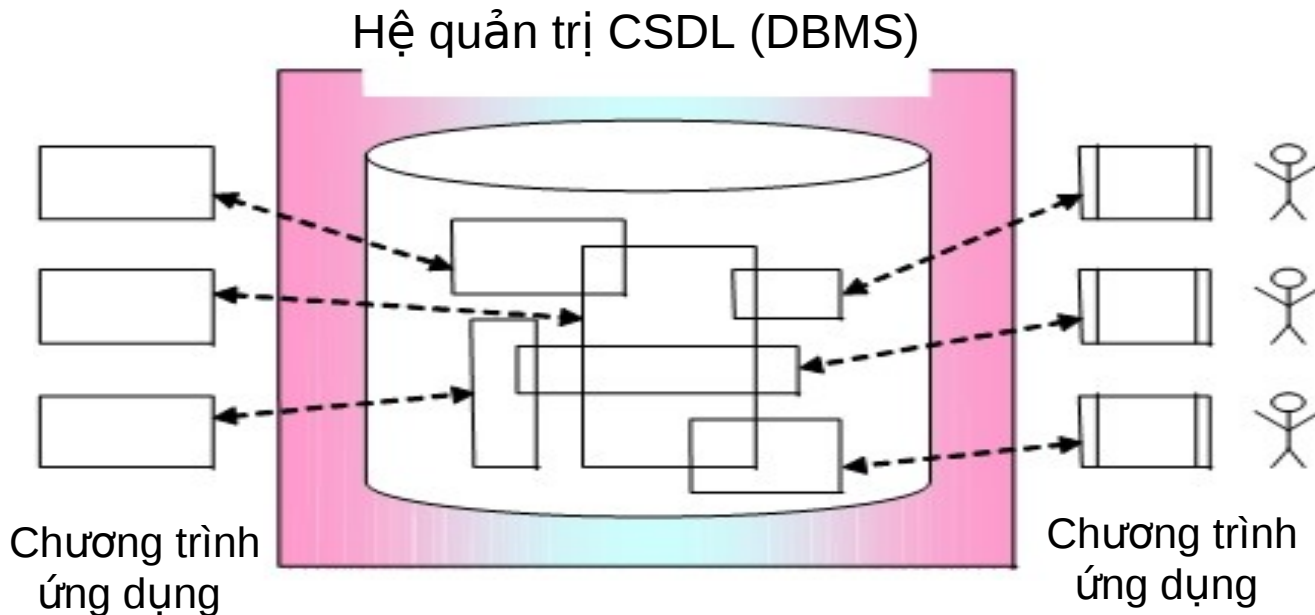


7. Tỉ eát ki eäm không gi an l ous tröõ:

Roõ raøng chuøng ta phaui loaui boũ dö thuøa, döõ lieäu seõ söõ düng ít choả löu tröõ hôn. Thay vì teän cuõa möät nhuøm nhaän vieän phaui hieän dieän trong 20 hoæc 25 thì nay chæ càn trong möät taäp tin. Tuy nhieän, böui vì ngöøoi söõ düng baét ñâu quen thuøc vöui vieäc söõ düng vaø cöu nhieäu quyèän häin hôn, hoĩ baét ñâu söõ düng chuøng thöøng xuyeän hôn. Vì vaãy naêng xuaát lao ñöäng seõ taêng vaø không gian löu tröõ seõ giaüm nhö hieäu suaát söõ düng CSDL.

III. Các thành viên trong hệ cơ sở dữ liệu

Một hệ CSDL gồm 4 hợp phần: dữ liệu, phần cứng, phần mềm và người dùng.



Hình : Sơ đồ lược giản về hệ thống CSDL



Dữ liệu

Trong CSDL, dữ liệu có hai đặc trưng chính:

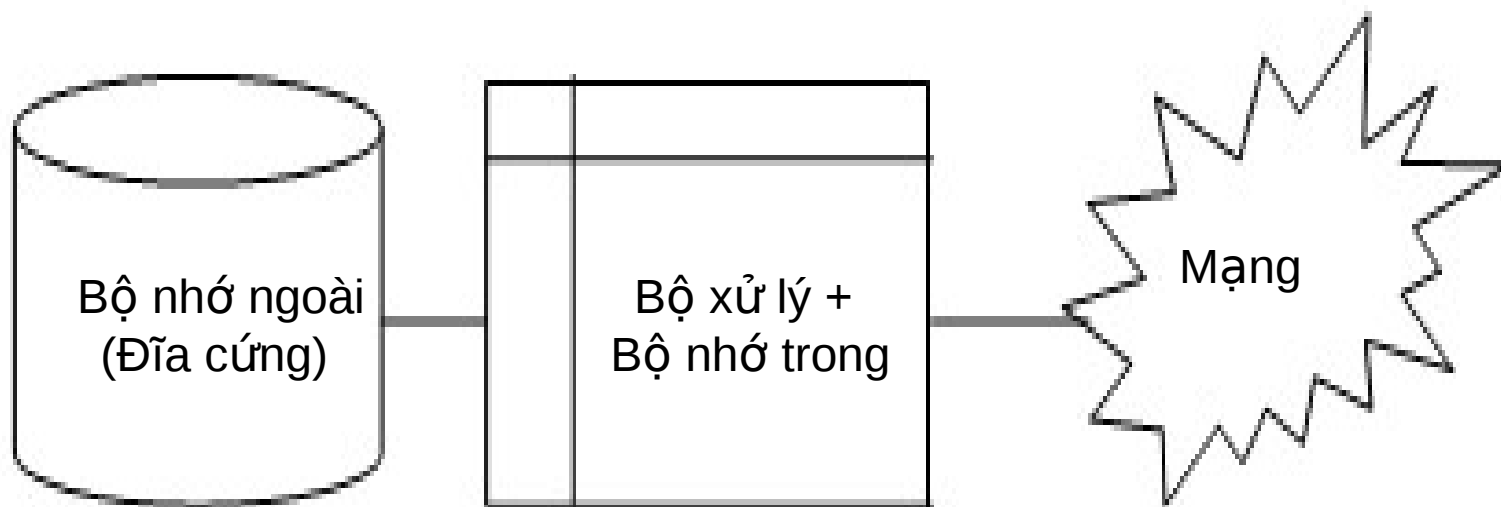
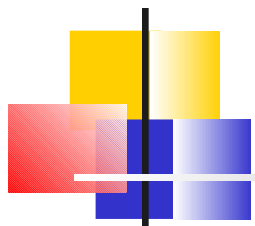
- Tính tích hợp: CSDL là nơi tập hợp nhiều hồ sơ, và người ta cố gắng loại bỏ đến mức tối đa các dữ liệu dư thừa. Lấy ví dụ, trong một cơ quan có danh sách nhân sự và bảng lương hàng tháng. Trong danh sách nhân sự, chúng ta có toàn bộ họ và tên cán bộ nhân viên. Trong bảng lương hàng tháng, chúng ta cũng thấy, bên cạnh các cột thông tin khác, là cột họ và tên cán bộ nhân viên. Rõ ràng, trong CSDL, không cần thiết phải lưu danh sách họ và tên cán bộ nhân viên một lần nữa. Họ và tên sẽ được lấy ra nhờ sử dụng tham chiếu.
- Tính chia sẻ: CSDL là nơi cho phép nhiều người dùng truy cập - tùy theo mục đích sử dụng của từng người. Hơn nữa, nhiều người dùng có thể truy cập đồng thời (chứ không phải người này phải đợi người kia truy cập xong mới đến lượt)



Phần cứng

Phần cứng của hệ thống CSDL gồm:

- Bộ nhớ ngoài: chủ yếu là đĩa từ, cụ thể hơn là đĩa cứng cùng với các bộ phận điều khiển khác như khối vào/ra, ổ đĩa, khối điều khiển, ... được sử dụng để lưu dữ liệu.
- Bộ xử lý và bộ nhớ trong dùng để chạy các phần mềm của hệ thống CSDL. Khối kết nối gồm card mạng/modem được sử dụng để kết nối hệ thống CSDL với thế giới bên ngoài.

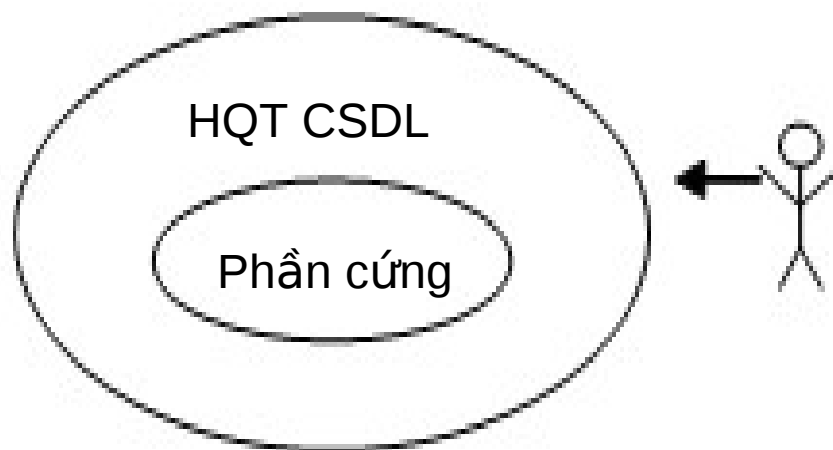
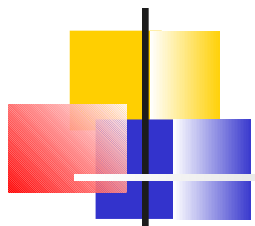


Hình 3: Các khối phần cứng cơ bản phục vụ hệ thống CSDL



Phần mềm

Đứng trung gian giữa phân vật lý (là nơi thực sự lưu dữ liệu) và người dùng của hệ thống CSDL là khối phần mềm quản trị CSDL, được gọi chung là Hệ quản trị (HQT) CSDL, trong tiếng Anh là Database Management System - DBMS. Tất cả các phép toán thực hiện trên CSDL đều phải thông qua HQT CSDL, như thêm mới hoặc hủy bỏ hồ sơ (bảng), trích xuất dữ liệu, cập nhật các hồ sơ (bảng) ,... Chức năng cơ bản của HQT CSDL là tạo lớp vỏ bọc phần cứng đối với người dùng - người dùng chỉ nhìn thấy CSDL thông qua các công cụ mà HQT CSDL cung cấp, chứ không cần quan tâm đến phân vật lý của CSDL.



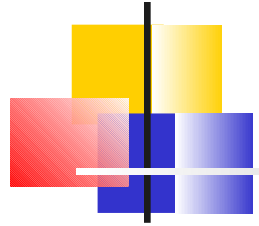
Hình 4: Hình ảnh về tạo lớp vỏ bọc của HQT CSDL



Người dùng

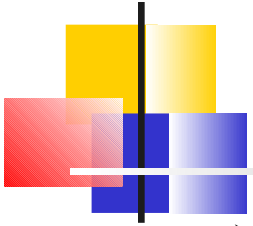
Người dùng CSDL có thể chia thành các lớp như sau:

- Lớp thứ nhất: lập trình viên CSDL: là người viết chương trình ứng dụng sử dụng CSDL thông qua một ngôn ngữ nào đó, như COBOL, C++, hay PHP. Các chương trình này sử dụng các phép toán lên CSDL theo cách thông thường: thêm bớt bảng, cập nhật dữ liệu, chủ yếu sử dụng các câu lệnh SQL. Các chương trình có thể được viết theo lô các lệnh hoặc cũng có thể hoạt động trực tuyến - nghĩa là hội thoại trực tiếp với HQT CSDL. Chức năng hoạt động trực tuyến thường được sử dụng để quản trị CSDL.
- Lớp thứ hai: người dùng cuối sử dụng các chương trình đã lập sẵn để giao tiếp với CSDL. Các chương trình đã lập sẵn gồm các chương trình được lập bởi các lập trình viên hoặc là một phần của HQT CSDL. Phần lớn các HQT CSDL đều cung cấp nhiều tiện ích lập sẵn. Một trong các tiện ích cơ bản đó là giao diện truy vấn. Trong giao diện này, người dùng có thể đưa ra các câu lệnh SQL và phần mềm sẽ cho kết quả của các câu lệnh đó.
- Lớp thứ ba: quản trị CSDL (database administrator - DBA) là người làm công tác quản trị CSDL



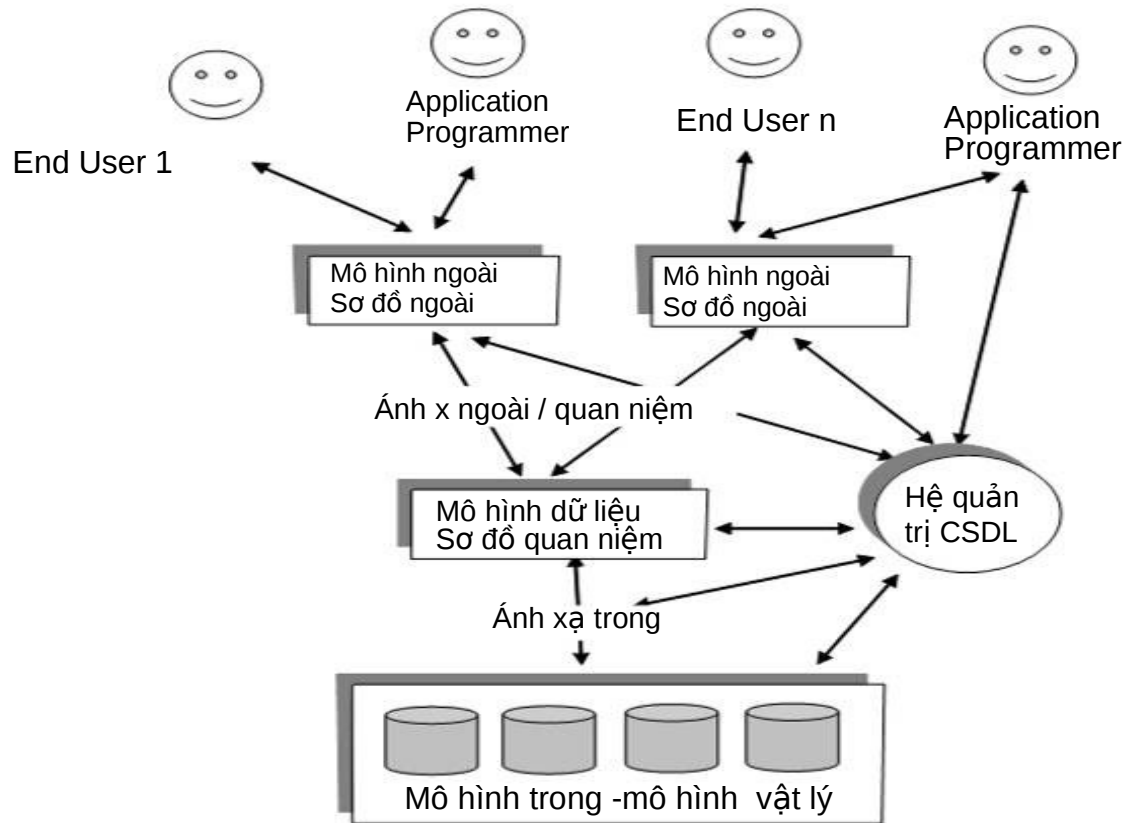
1. Mô hình ba mức của cơ sở dữ liệu

- Mô hình kiến trúc ba mức của hệ CSDL gồm: *Mức trong*, *Mức mô hình dữ liệu* (Mức quan niệm) và *Mức ngoài*. Giữa các mức tồn tại các ánh xạ quan niệm trong và ánh xạ quan niệm ngoài. Trung tâm của hệ thống là mức quan niệm, tức là mức mô hình dữ liệu. Ngoài ra còn có khái niệm người sử dụng, người quản trị CSDL, hệ quản trị CSDL



-
- **Mô hình ngoài:** Mô hình ngoài là nội dung thông tin của CSDL dưới cách nhìn của người sử dụng. Là nội dung thông tin của một phần dữ liệu tác nghiệp được một người hoặc một nhóm người quan tâm.
 - **Mô hình dữ liệu (Mô hình quan niệm):** Mô hình quan niệm là cách nhìn dữ liệu một cách tổng quát của người sử dụng
 - **Mô hình trong:** Mô hình trong là mô hình lưu trữ vật lý dữ liệu. Chỉ có duy nhất một và chỉ một cách biểu diễn CSDL dưới dạng cơ sở vật lý

- Như vậy mô hình dữ liệu là cách nhìn toàn bộ nội dung thông tin của CSDL, sơ đồ quan niệm là định nghĩa của cách nhìn ấy. là bước đi đầu tiên, quan trọng trong việc thiết kế và cài đặt các hệ CSDL



Hình ảnh về kiến trúc hệ cơ sở dữ liệu



Baøi taäp

Khảo sát, phân tích về hệ thống quản lý thư viện.

■ Đặt vấn đề:

- Khoa CNTT muốn xây dựng một hệ thống quản lý thư viện.
- Hệ thống này chỉ phục vụ cho cán bộ và giảng viên trong khoa.
- Mục đích chính của hệ thống quản lý thư viện là
 - Phục vụ bạn đọc
 - Nhận mã sách hoặc mã tạp chí, lưu trữ và quản lý chúng, tạo ra chỉ mục cho tài liệu
 - Theo dõi quá trình mượn trả của bạn đọc.



Khảo sát hệ thống quản lý thư viện

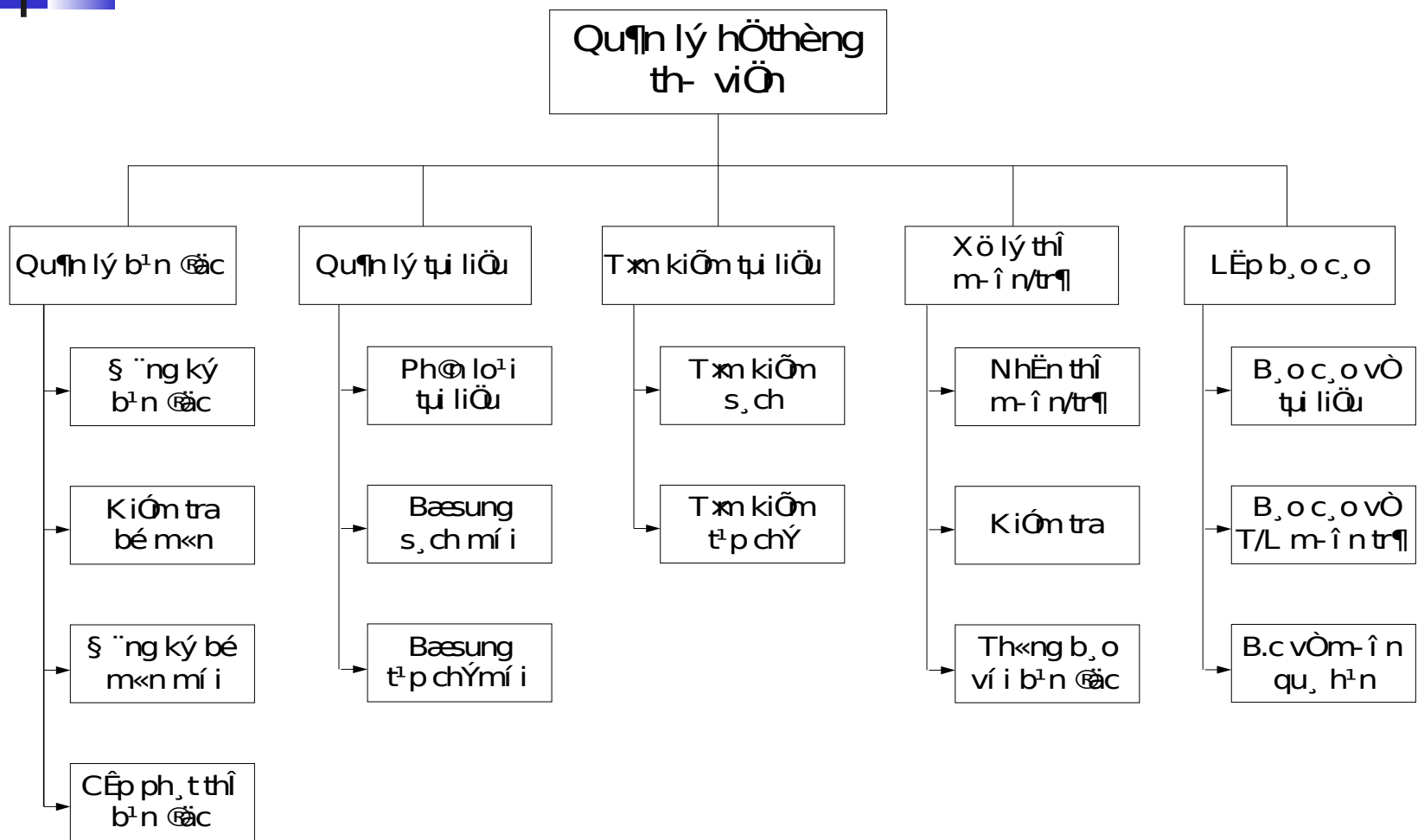
- Các hoạt động chính trong quá trình khảo sát hệ thống quản lý thư viện nhằm xác định:
 - Mối liên hệ giữa hệ thống thư viện hiện tại và các hệ thống quản lý thư viện khác.
 - Những chức năng chính của hệ thống quản lý thư viện
 - Những chức năng cần được nâng cấp trong hệ thống mới.



Phỏng vấn

- Những câu hỏi phỏng vấn gồm nội dung sau:
 - Công việc chính của hệ thống quản lý thư viện là gì?
 - Chức năng của từng công việc
 - Các công việc đang được thực hiện như thế nào?
 - Ai là người chịu trách nhiệm thực hiện?
 - Hạn chế và những khó khăn gặp phải của mỗi công việc
 - Thư viện phục vụ ai?
 - ...
- Báo cáo khảo sát

Biểu đồ phân cấp chức năng





CHƯƠNG II: CÁC CÔNG CỤ PHÂN TÍCH VÀ THIẾT KẾ

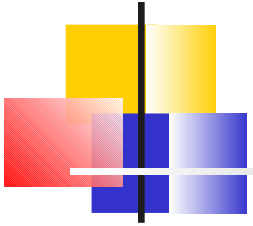
I. Các công cụ thiết kế

1. Sơ đồ dòng dữ liệu (Data Flow Diagram)

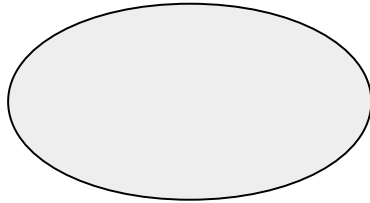
Có nhiều loại sơ đồ dòng dữ liệu:

- Context DFD: Sơ đồ phạm vi dòng dữ liệu.
- Analysis DFD: Sơ đồ phân tích dòng dữ liệu.
- Model DFD: Sơ đồ mô hình dòng dữ liệu.

CÁC SƠ ĐỒ CÓ THỂ TẠO MỘT TRONG BỐN KÝ HIỆU SAU



**Ký hiệu kết thúc
(Terminator)**



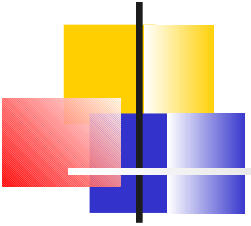
**Quá trình xử lý
(Process)**



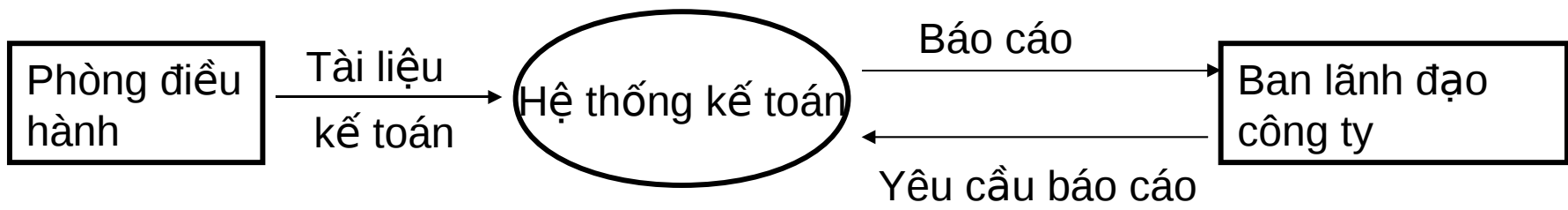
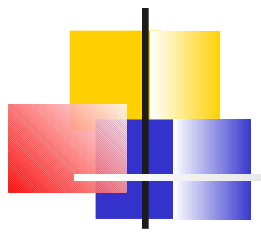
**Dòng dữ liệu
(data flow)**



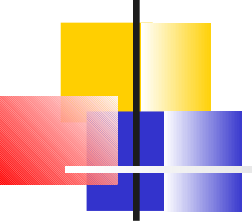
**Kho dữ liệu
(data store)**



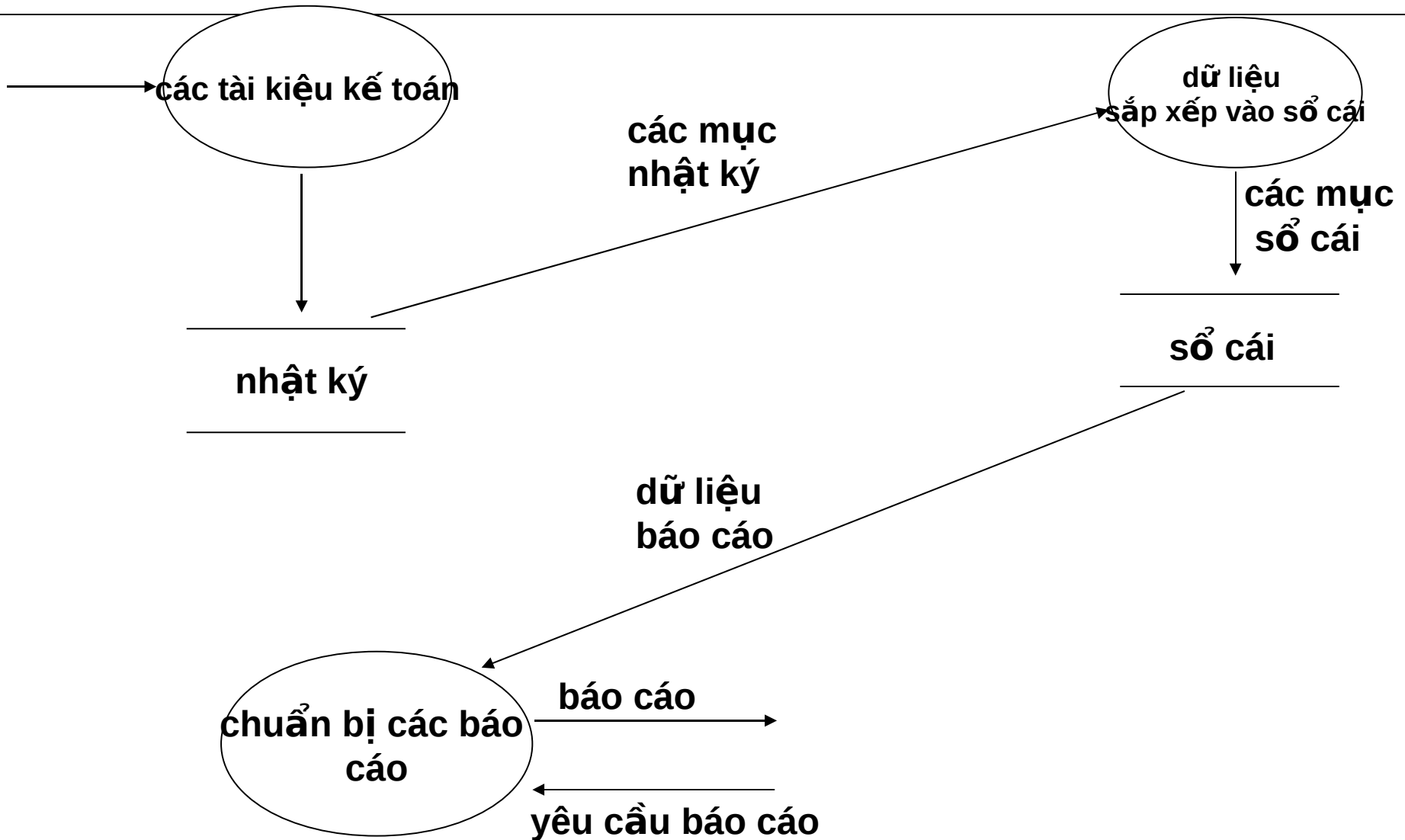
- Sơ đồ context được dùng để trình bày các *giới hạn* (phạm vi) của hệ thống mới hay hệ thống hiện có.
- Ký hiệu kết thúc dùng để chỉ ra các nguồn dữ liệu nhập và dữ liệu kết xuất.
 - Dòng dữ liệu mô tả đường di chuyển của dữ liệu đến và đi khỏi ký hiệu kết thúc.
 - Tất cả dữ liệu nhập và dữ liệu kết xuất đều đi vào và đi ra khỏi quá trình xử lý, đó chính là hệ thống chúng ta đang phân tích.



sơ đồ phạm vi dòng dữ liệu (context DFD) cho hệ thống kế toán



Sơ đồ Analysis DFD và model DFD trình bày chi tiết của hệ thống mới và hệ thống hiện có. Chúng ta không dùng ký hiệu kết thúc ở đây bởi vì chúng ta đang ở trong hệ thống. Kho dữ liệu là một vị trí lưu trữ dữ liệu giữa các quá trình xử lý, chẳng hạn như: thiết bị lưu trữ trên máy tính, các học tủ đựng tài liệu...



Sơ đồ phân tích dòng dữ liệu (Analysis DFD) cho hệ thống kế toán



2. Từ điển dữ liệu

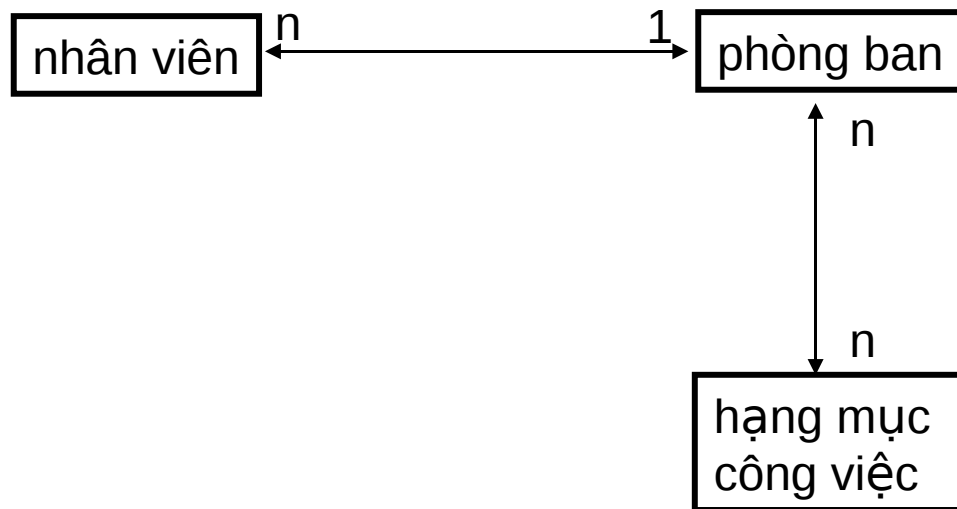
Từ điển dữ liệu dùng để ghi chép các thực thể trong hệ thống. Hay từ điển dữ liệu tập hợp các tài liệu về các thực thể dữ liệu vào *kho dữ liệu* để cho *các quá trình xử lý* sử dụng. Một từ điển dữ liệu tốt sẽ giúp chúng ta trả lời các câu hỏi đại loại như:

- sử dụng thực thể nào để tạo ra báo cáo này?
- việc tính tiền thưởng cuối năm sẽ sử dụng đến những thực thể nào?
- khi muốn rút ngắn tiến công việc độ ta phải thay đổi gì?...



3. sơ đồ thực thể:

Sơ đồ thực thể trình bày các mối liên hệ thực thể dữ liệu.



Dùng các hình hộp chữ nhật để biểu thị cho thực thể dữ liệu và các mũi tên biểu thị cho mối liên hệ

II. Khảo sát và nắm bắt các yêu cầu.

➔ Khảo sát hiện trạng là quá trình khám phá cách mà hệ thống đã được thiết kế và vận hành trong thực tế, làm bộc lộ các quan hệ nội tại giữa các thành phần trong hệ thống và mối liên hệ giữa hệ thống với yêu cầu.

1. Để tìm ra ưu điểm/khuyết điểm => xác định yêu cầu mới
2. Để tìm ra giải pháp khả thi cho các yêu cầu

➔ Xác định yêu cầu cho hệ thống là một quá trình tổng hợp thông tin mang tính hệ thống và khách quan, không thể chỉ dựa vào mô tả của một vài cá nhân, vì

- Mỗi cá nhân chỉ nhìn hệ thống theo một lĩnh vực chuyên môn đang phụ trách; do đó các phát biểu thường không bộc lộ được các ràng buộc tổng thể của hệ thống.
- Các phát biểu của nhiều người thường có mâu thuẫn nhau do mỗi người có quan điểm khác nhau về hệ thống hiện tại.

Nội dung khảo sát

1. Tìm hiểu mục đích, yêu cầu đối với hệ thống

- Xác định vai trò (lợi ích) của hệ thống đối với tổ chức.

2. Tìm hiểu các quy trình giữa các thành phần trong hệ thống

- Để làm bộc lộ mối quan hệ nội tại giữa các thành phần

a. Công việc: quy tắc quản lý, cách làm, kết quả, chuyển giao

b. Nguồn lực: mức độ, phương tiện, nhân lực, cách tổ chức bố trí

1. Tìm hiểu thông tin – dữ liệu của quy trình

- Các quy định, hướng dẫn, tiêu chuẩn
- Forms/Reports : có thông tin gì, khi nào cần, dùng để làm gì,...

2. Hệ thống thông tin trên máy tính (CBIS) hiện có

- Phạm vi, mức độ và cách nó trợ giúp users thực hiện công việc
- Vai trò (roles) của các users trong hệ thống.
- Phần mềm, mạng máy tính, thiết bị,...

Các phương pháp khảo sát truyền thống

1. Phỏng vấn cá nhân (interviews)
2. Phỏng vấn nhóm (group interviews)
3. Phiếu thăm dò (questionnaires)
4. Quan sát người sử dụng (viewing)
5. Phân tích tài liệu (document analysis)

Các phương pháp này thuần túy là chỉ dùng để biết về hệ thống hiện tại.

1. Phỏng vấn cá nhân (interviews)

- Phỏng vấn: hẹn gặp, tiếp xúc, hỏi và ghi nhận câu trả lời.
 - Phỏng vấn người nhân viên: Công việc của họ, thông tin mà họ cần để làm việc, cách xử lý thông tin,...
 - Phỏng vấn người quản lý: Xu hướng của tổ chức, các chính sách đang và sẽ áp dụng, mong muốn thay đổi, những ý kiến nhận xét về hệ thống hiện tại,...
- Ưu điểm
 - Có cơ hội hỏi thêm về những gì vừa mới biết
- Nhược điểm
 - Có thể có mâu thuẫn ý kiến riêng giữa các cá nhân
 - Tốn nhiều thời gian nếu cần phỏng vấn nhiều người

2. Phỏng vấn nhóm (group interviews)

- Phỏng vấn nhóm: Đặt câu hỏi chung cho nhiều người chủ chốt cùng một lúc trong cuộc họp, hội thảo.
- Ưu điểm
 - Ít tốn thời gian hơn phỏng vấn cá nhân
 - Gia tăng sự trao đổi về các “phát hiện mới” giữa những người tham gia => có cơ hội hiểu biết sâu hơn.
 - Hạn chế bớt sự mâu thuẫn ý kiến cá nhân
- Khuyết điểm: khó thu xếp cho cuộc phỏng vấn
 - Do có khoảng cách kiến thức chuyên môn giữa nhiều người tham gia phỏng vấn
 - Khó sắp xếp thời gian và địa điểm họp cho nhiều người cùng một lúc

3. Phiếu thăm dò (questionnaires)

- Gửi phiếu có ghi câu hỏi phỏng vấn đến nhiều người, sau đó phân tích/thống kê kết quả trả lời từ các phiếu đã quay về. Câu hỏi phải hết sức rõ ràng, dễ hiểu và dễ trả lời để người được phỏng vấn không bị nhầm lẫn.
- Ưu điểm
 - Rẻ hơn các loại phỏng vấn
 - Thống kê trên số lượng lớn phiếu quay về có thể nhận được thông tin tương đối khách quan.
- Khuyết điểm
 - Không có cơ hội để hỏi thêm !
 - Không chắc chắn ai là tác giả !!
 - Số phiếu quay về có thể quá ít.

So sánh Interviews và Questionnaires

Tính chất	Interviews	Questionnaires
Giàu thông tin	Cao	T.bình - Thấp
Thời gian	Có thể rất lâu	Thấp – T.bình
Chi phí	Có thể cao	vừa phải
Tìm hiểu sâu thêm	Tốt	Giới hạn
Độ tin cậy	Cao. Đã biết rõ người được phỏng vấn.	Không cao. Không xác định được tác giả.
Mức độ cộng tác	Người được phỏng vấn cùng tham gia giải quyết vấn đề và cam kết thực hiện	Không rõ các cam kết
Người tham dự	Số lượng giới hạn, đáp ứng tốt	Số lượng lớn, đáp ứng không tốt.

4. Quan sát người nhân viên

- Để biết họ thường làm gì, và xử lý công việc ra sao; đồng thời để đánh giá mức độ hiệu quả của các quy trình và các phương tiện hỗ trợ cho người nhân viên làm việc.
- Ưu điểm
 - Kiểm chứng được những gì đã biết.
 - Biết được cường độ của từng công việc trong thực tế
- Khuyết điểm
 - Sự quan sát có thể không khách quan, do người nhân viên sẽ thay đổi thói quen nếu biết mình đang bị quan sát.
 - Tốn nhiều thời gian ngồi quan sát.

5. Thu thập tài liệu

- Thu thập, đọc và tìm hiểu các tài liệu văn bản mô tả về hệ thống như hồ sơ thiết kế, các biểu mẫu nhập liệu, các báo cáo, các quy trình vận hành khai thác,...
- Ưu điểm:
 - Có nhiều thông tin chi tiết, chính xác.
 - Dễ dàng khái quát/hệ thống hóa được toàn bộ hệ thống
- Nhược điểm:
 - Tài liệu có thể bị lạc hậu so với thực tế

Các phương pháp khảo sát hiện đại

1. Làm mẫu thử (Prototyping)
 2. Joint Application Design (JAD)
 3. Tái cấu trúc tiến trình (Business Process Reengineering)
- Khác với phương pháp truyền thống, phương pháp hiện đại sử dụng nhiều loại nguồn lực trợ giúp (ngoài người được phỏng vấn) để định nghĩa hệ thống mới ngay trong khi đang khảo sát.

1. Làm mẫu thử (prototyping)

- Tìm hiểu sơ lược yêu cầu ban đầu, chuyển yêu cầu này thành 'demo' cho người sử dụng kiểm tra để hiệu chỉnh lại. Qua nhiều chu kỳ kiểm tra – hiệu chỉnh, bản demo (thể hiện yêu cầu+giải pháp của hệ thống mới) được hoàn chỉnh dần từ tổng quát đến chi tiết.
- Ưu điểm
 - Phân tích viên hiểu được cặn kẽ yêu cầu chi tiết từ User
 - User biết được hệ thống mới sẽ hoạt động như thế nào
- Khuyết điểm
 - Không thể diễn tả xử lý hệ thống (là những gì người sử dụng không nhìn thấy được, do tính tổng quát của nó)
 - Khó thống nhất quan điểm sử dụng từ nhiều users

2. Joint Application Design (JAD)

- Là một chuỗi các cuộc họp chuyên sâu có sử dụng các phương tiện “demo” như CASE tools, và có ❶ người sử dụng, ❷ người quản lý và ❸ người phát triển hệ thống cùng tham gia để đặc tả hoặc xem xét tường tận các yêu cầu cho hệ thống.
 1. Người sử dụng là người đưa ra yêu cầu về các chuyển giao sau khi thiết kế.
 2. Người quản lý là người nêu ra bài toán và quyết định có chấp nhận phương án không.
 3. Người phát triển hệ thống là người đưa ra phương án giải quyết bài toán.
- Ưu điểm: prototyping + phỏng vấn nhóm
- Khuyết điểm: khá tốn kém vì nhân lực+thời gian dài

3. Tái cấu trúc các tiến trình kinh doanh

- BPR: Thay vì “cải tiến” hệ thống cho phù hợp với mô hình tổ chức và các quy tắc quản lý hiện tại, phương pháp này hướng việc khảo sát vào việc tận dụng **ưu thế của các loại nguồn lực bên trong và bên ngoài** để tái thiết lại hệ thống.
 - Thay đổi mô hình và nghiệp vụ để ứng dụng CNTT
 - Phá bỏ các nguyên tắc lạc hậu trì trệ đang tồn tại
 - Quan điểm: "Nếu một tổ chức được xây dựng lại từ đầu, thì nó cần phải hoạt động như thế nào?"
- Ưu điểm: tạo ra thành tựu lớn nhưng có nhiều rủi ro.
- Ví dụ: Nhà sách “Amazon.com” bán sách điện tử thay cho các quyển sách giấy ⇒ không có chi phí lưu kho, không có quầy giao dịch và trưng bày, mở rộng kinh doanh, ... nhưng phải đối mặt với vấn đề “copy rights” của sách điện tử.

Đánh giá sơ lược sau khảo sát

1. Kết luận sơ lược sau khi khảo sát

- Tần suất, cường độ công việc cao ở đâu, khi nào
- Có bị nghẽn cổ chai ?
- Có xung khắc thông tin ?
- Hiệu quả xử lý các báo cáo

2. Nhận định sơ lược về cơ hội và thách thức để khắc phục, cải tiến hoặc cải cách để định hướng tập trung phân tích

1. Từ nội bộ của tổ chức: nguồn lực, kế hoạch, chiến lược
2. Môi trường bên ngoài: quy định của chính phủ và phương pháp, công nghệ mới trên thế giới



III. Sơ đồ dữ liệu (DFD- Data Flow Diagram)

1. Mục đích

Cho ta biết các phần tử cấu thành của hệ thống xét theo hoạt động chức năng, nhìn từ trên xuống theo mức độ chi tiết khác nhau. Các chức năng liên kết với nhau ntn và hoạt động thực tiễn ra làm sao.

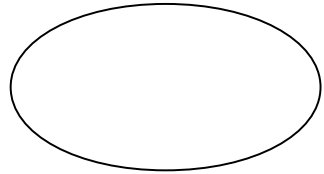
.Định nghĩa

Sơ đồ luồng dữ liệu là một công cụ mô tả dòng thông tin nghiệp vụ nối kết giữa các chức năng trong một phạm vi được xét.

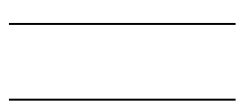
Lược đồ dòng dữ liệu - Data Flow

Diagram

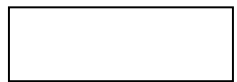
⇒ DFD là lược đồ sử dụng 4 ký hiệu cùng với các quy tắc vẽ để diễn tả các dòng dữ liệu di chuyển trong hệ thống.



Process : Là một hành động hoặc một hệ thống con xử lý trên dữ liệu (biến đổi, lưu trữ hoặc phân phối dữ liệu).



Data store : Là bộ phận dùng để lưu trữ dữ liệu, như tập tin, hồ sơ, CSDL,...



Source/Sink : Là thành phần phát sinh dữ liệu (source) cho hệ thống, hoặc tiêu thụ dữ liệu (sink) từ hệ thống.



Data flow : dòng dữ liệu cơ bản của lược đồ, chỉ ra 1 nội dung dữ liệu (không đổi) được gửi từ đâu và đi đến đâu.

⇒ Quy ước

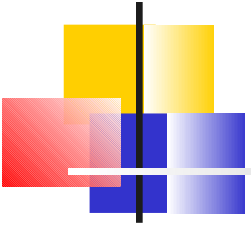
- Dùng **Động từ** để đặt tên cho Process
- Dùng **Danh từ** để đặt tên Data store, Source, Sink và Data flow



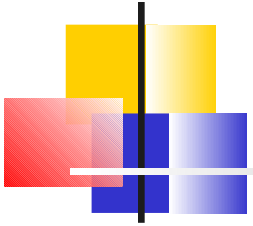
2. Các thành phần của DFD

- **Kho dữ liệu:** mô tả các dữ liệu được lưu giữ trên một vật mang tin nào đó trong một thời gian nhất định để cho một hoặc nhiều tiến trình hoặc tác nhân có thể truy cập đến nó, ký hiệu là hình chữ nhật tròn góc hoặc 2 gạch song song.

Ví dụ: Kho dữ liệu “Hồ sơ cán bộ”, “vật tư”



-
- **Quá trình (Process):** ký hiệu bằng vòng trong, tượng trưng cho chức năng hệ thống phải thực hiện thay đổi thông tin đầu vào để tạo thông tin mới
 - **Dòng dữ liệu (Flow):** ký hiệu bằng đường mũi tên chỉ hướng ra của thông tin, liên kết các quá trình, tượng trưng cho thông tin yêu cầu vào và thông tin biến đổi ra



- **Tác nhân trong, tác nhân ngoài:** là một người, một nhóm người hay một tổ chức ở bên trong hay ngoài lĩnh vực nghiên cứu, nhưng có quan hệ thông tin với phạm vi nghiên cứu.

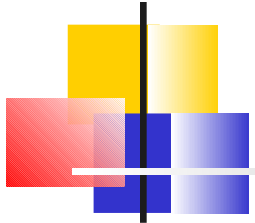
Biểu diễn:

- . Tác nhân ngoài biểu diễn bằng hình chữ nhật có dán nhãn
- . Tác nhân trong biểu diễn bằng hình chữ nhật hở một phía và trên có ghi nhãn



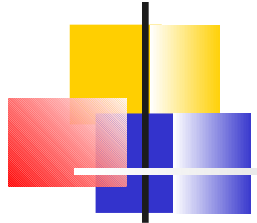
3. Một số lưu ý khi vẽ DFD

- **Về ô xử lý:**
 - ❖ Phải chuyển ô xử lý chỉ có một dòng dữ liệu đi ra thành tác nhân cung cấp dữ liệu.
 - ❖ Phải chuyển ô xử lý chỉ có một dòng dữ liệu đi vào thành tác nhân nhận dữ liệu.
 - ❖ Dùng động từ để đặt tên cho ô xử lý.



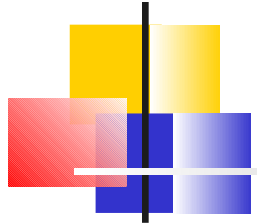
■ **Về kho dữ liệu:**

- ❖ Dòng dữ liệu không thể là cầu nối giữa hai kho dữ liệu
- ❖ Dòng dữ liệu không thể là cầu nối giữ tác nhân và kho dữ liệu.
- ❖ Dùng danh từ để đặt tên cho kho dữ liệu.



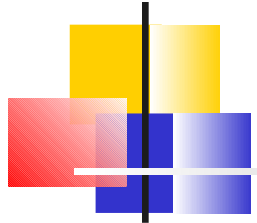
■ **VỀ tác nhân:**

- ❖ Dòng dữ liệu không thể là cầu nối giữa hai tác nhân
- ❖ Dùng danh từ để đặt tên cho tác nhân.



■ **Về dòng dữ liệu:**

- ❖ Dòng dữ liệu không thể là cầu nối của một xử lý.
- ❖ Dòng dữ liệu vào kho dữ liệu mang ý nghĩa cập nhật (thêm, sửa, xóa)
- ❖ Dòng dữ liệu ra khỏi kho dữ liệu mang ý nghĩa tham khảo.
- ❖ Dùng danh từ để đặt tên cho dòng dữ liệu.



- **Các ký hiệu lặp lại:**

- ❖ Để tránh các dòng giao nhau.
- ❖ Trên sơ đồ DFD ta có thể lặp lại các tác nhân, các kho dữ liệu...
- ❖ Các phần tử lặp lại này được đánh dấu bằng cách vẽ thêm đường kẻ đôi dọc cho kho dữ liệu và đường kẻ chéo cho tác nhân .



IV. TỪ ĐIỂN DỮ LIỆU (DATA DICTIONARY)

Nhằm làm rõ hơn các khái niệm của vấn đề, người ta phải xây dựng từ điển dữ liệu để giải thích thêm ý nghĩa của các tập thực thể và thuộc tính.



Ví dụ: Tập thực thể MẶT HÀNG

mã hàng

MẶT HÀNG

mô tả

đơn giá

thuế xuất

Sẽ có từ điển dữ liệu như sau:

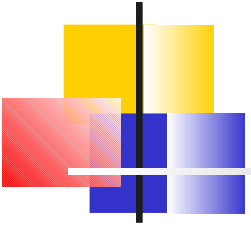
Thực thể	MẶT HÀNG
Tên khác	Hàng, sản phẩm, hàng hóa.
Mô tả	Hàng là những thứ được mua với số lượng khác nhau từ các nhà cung ứng, được lưu trữ trong kho và bán cho khách hàng.

Thuộc tính	Mã hàng:	Là một số dùng để phân biệt mặt hàng này với mặt hàng kia. Giá trị có dạng 0001>9999
	Mô tả :	Mô tả mặt hàng gồm qui tắc và hình dáng. Loại ký tự chuỗi gồm 100 ký tự. Có thể có giá trị rỗng.
	Đơn giá:	Đơn giá hiện tại của mặt hàng. Có loại dữ liệu số với 2 số thập phân, có giá trị từ 10 đến 50, mặt nhiên là 0.
	Thuế xuất:	Tỷ suất thuế bán của mặt hàng được ghi dưới dạng phần trăm. Có loại dữ liệu số, có giá trị từ 0 đến 99. Giá trị mặt nhiên 0.

- Thuoăc tđnh caàn ñeà caăp caùc noăi dung sau:
 - - Giaûi thđch ngöõ nghóa.
 - - Xaùc ñònh kieäu döõ lieäu.
 - - Xaùc ñònh mieàn giaù trò
 - - Xaùc ñònh giaù trò maëc nhieân
 - - Cho pheùp ñeỏ roăng khoâng
 - - Xaùc ñònh caùc raøng buoăc khaùc

V. Đặc tả tiến trình (process Specification)

- Phương pháp mô tả chi tiết của một chức năng, hay tiến trình được gọi là phương pháp đặc tả tiến trình (process Specification).
- Đặc tả tiến trình phải ngắn gọn, chính xác và chỉ ra được tất cả các hoạt động logic diễn ra trong tiến trình, cùng với tiến trình thực hiện chúng. Một bản đặc tả tiến trình gồm 2 phần: phần đầu đề và phần thân mô tả.



-
- Phần đầu đề gồm tên chức năng, các dữ liệu vào và dữ liệu ra.
 - Phần thân của bản đặc tả tiến trình mô tả chi tiết của tiến trình để xử lý dữ liệu vào thành dữ liệu ra. Nếu có nhiều hoạt động, phần mô tả này phải thể hiện được thứ tự thực hiện, hoặc các trường hợp thực hiện các hoạt động tương ứng.

Một số công cụ thường dùng để đặc tả tiến trình là phương trình toán học, ngôn ngữ có cấu trúc, sơ đồ khối và bảng quyết định.

*Phương pháp dùng bảng

Ví dụ: xét tiến trình xử lý phiếu yêu cầu của bạn đọc

1. Tên tiến trình: Xử lý phiếu yêu cầu của bạn đọc

+Dữ liệu vào :

- Số thẻ bạn đọc,
- Phiếu yêu cầu
- Bản ghi bạn đọc
- Sổ theo dõi mượn trả
- tài liệu

+Dữ liệu ra:

- Phiếu yêu cầu đã được xử lý
- Thông báo cho bạn đọc

2. Mô tả tiến trình

	Điều kiện và hành động	Các t/h có thể xảy ra							
Điều kiện	Có thể mượn hợp lệ								
	Có tài liệu								
	Không nợ tài liệu								
Hành động	Yêu cầu làm thẻ								
	Làm thủ tục mượn								
	Trả phiếu yêu cầu								

*Phương pháp dùng ngôn ngữ có cấu trúc(giả mã)

- Trong các phương pháp phân tích và thiết kế hệ thống, người ta thường dùng ngôn ngữ có cấu trúc để mô tả chi tiết của tiến trình. Ngôn ngữ có cấu trúc không phải là ngôn ngữ tự nhiên thông thường mà thường là ngôn ngữ lập trình, như giả Pascal... Sử dụng các giả ngôn ngữ này có thể tránh được các ràng buộc chặt chẽ của các ngôn ngữ khác mà không gây ra sự hiểu lầm, vẫn diễn đạt được đầy đủ các hoạt động mà không bị rườm rà. Thông thường trong ngôn ngữ có cấu trúc được sử dụng để mô tả tiến trình, phải loại bỏ các lượng từ, các câu phức hợp, các cách diễn đạt tình.

Trong ngôn ngữ chỉ còn lại các thành phần:

- Động từ được dùng ở dạng mệnh lệnh
- Câu đơn giản thuộc loại điều khiển hay khẳng định
- Thuật ngữ được định nghĩa trong từ điển dữ liệu
- Một số lượng hạn chế các từ logic
- Và các cấu trúc cơ bản của ngôn ngữ là lặp và rẽ nhánh

Như vậy một ngôn ngữ có cấu trúc dùng để mô tả tiến trình có nhiều đặc điểm quan trọng của ngôn ngữ lập trình, nhưng không chịu sự ràng buộc chặt chẽ của ngôn ngữ lập trình nên dễ hiểu, gần với ngôn ngữ tự nhiên và có cấu trúc tương đối tự do.

Ví dụ:

- 1. Tiến trình:** In phiếu đòi sách đối với bạn đọc có sách quá hạn
 - + Dữ liệu vào:
 - Số thẻ bạn đọc
 - Dữ liệu mượn trả
 - Ngày hiện thời
 - +Dữ liệu ra:
 - Phiếu đòi và danh sách tài liệu mượn quá hạn với từng bạn đọc.

2. Mô tả:

Với mỗi bạn đọc trong danh sách

a) K = số thẻ bạn đọc

b) Danh sách quá hạn (K): = 0

c) Với mỗi tài liệu = Số mượn (K)

+ M : = Số hiệu tài liệu

+ Nếu M . Ngày trả < Ngày hiện thời Thì

Thêm M vào Danh sách quá hạn (K);

d) Nếu Danh sách quá hạn (K) = 0 Thì

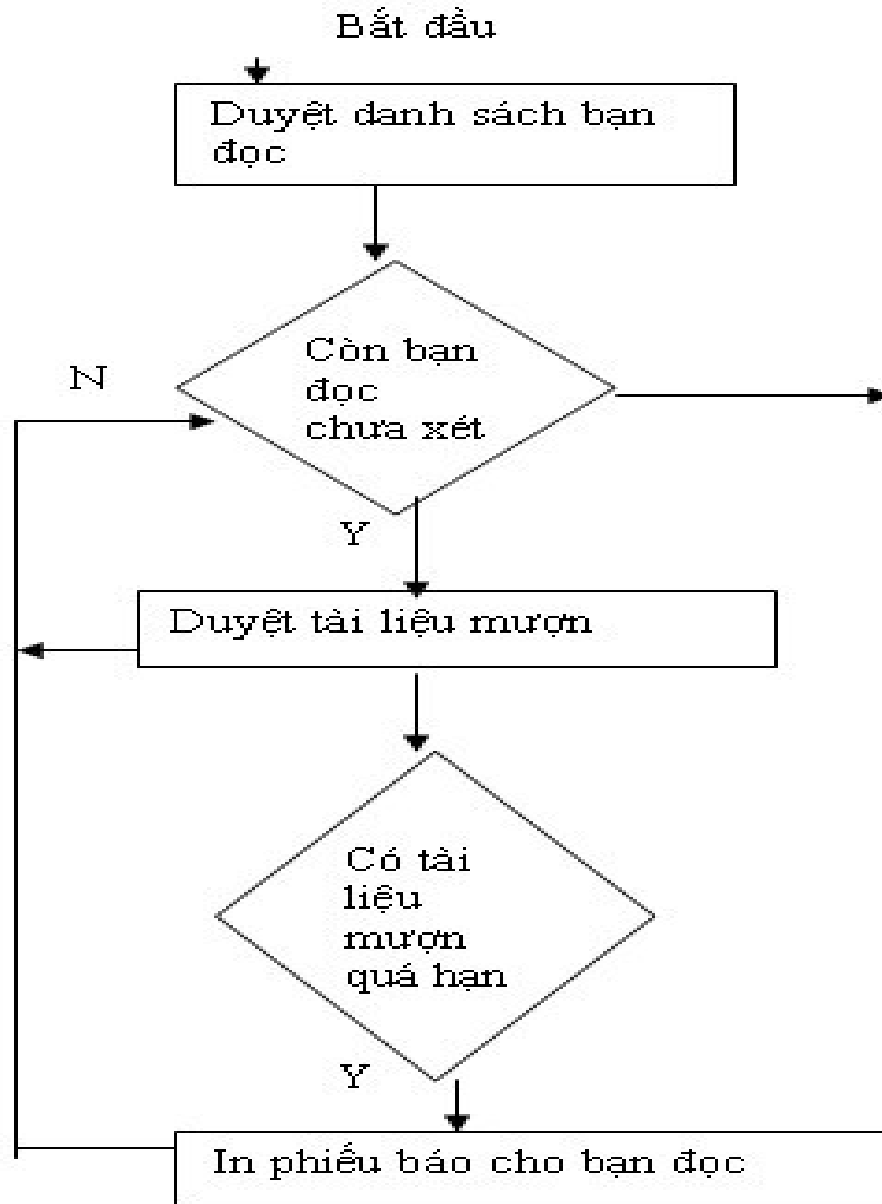
- In phiếu đòi

- In Danh sách quá hạn (K)

*Phương pháp dùng sơ đồ khối

Sơ đồ khối là loại biểu đồ diễn tả giải thuật quen thuộc và ưa dùng khi mới làm quen với việc viết chương trình. Sơ đồ khối là một công cụ dễ sử dụng và đơn giản, tuy nhiên khi lập trình nâng cao hoặc giải quyết những bài toán phức tạp và công kênh thì sơ đồ khối lại bộc lộ khá nhiều nhược điểm. Trong sơ đồ, bên cạnh các đường có hướng chỉ hướng điều khiển, có hai loại nút: nút hành động, tức là thực hiện một thao tác nào đó, và nút kiểm tra điều kiện.

Ví dụ: Sơ đồ khối đặc tả tiến trình “In phiếu đòi sách quá hạn”





VI. Sơ đồ liên kết các thực thể (ERD – Entity Relationship Diagram)

1. Mục đích:

- DFD chỉ ra *làm thế nào, ở đâu và khi nào* dữ liệu được xử lý, nhưng không chỉ ra *định nghĩa, cấu trúc và các quan hệ* của dữ liệu.
- Entity Relationship Diagram (ERD) là lược đồ thể hiện cấu trúc trừu tượng hóa của dữ liệu trong tổ chức, dựa trên khái niệm thực thể (entity) và quan hệ (relationship) giữa các thực thể, để nhằm thể hiện nội dung, ý nghĩa của dữ liệu trong hệ thống.



2. Các thành phần của ERD

Ba phần chính của ERD là: *thực thể*, *thuộc tính*, và *các mối quan hệ* giữa các thành phần đó

Thực thể (Entity)

➔ Nhân viên, Sinh viên, Môn học,... là các thực thể, là một khái niệm tổng quát hóa cho một nhóm các đối tượng (thể hiện, entity instance) trong thế giới thực có chung một số đặc điểm (thuộc tính). Vd: môn “PTTK”, môn “CSDL” là các thể hiện của thực thể MônHọc.

1. *Thực thể xác thực* mô tả các đối tượng tồn tại thực sự trong thế giới thực: Xe đạp, xe hơi, nhà, quyển sách,...
2. *Thực thể chức năng* mô tả mục đích, chức năng, hoặc nhiệm vụ của con người, thiết bị hoặc tổ chức: Sinh viên, nhân viên, khách hàng, nhà kho,...
3. *Thực thể sự kiện* mô tả các sự kiện hoặc biến cố: biên nhận, biên bản họp, kỳ thi,...
4. *Thực thể quan hệ* mô tả các quan hệ giữa các đối tượng: Quản lý, đăng ký, hợp đồng,...

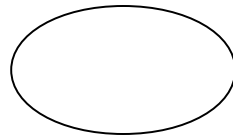
Thực thể (Entity)

- ➔ *Attribute* (thuộc tính) là đặc điểm chung của các đối tượng trong thực thể. Vd: MãNV, Tên, Địa chỉ, Kỹ Năng là các thuộc tính được quan tâm khi nghĩ về thực thể Nhân Viên.
- ➔ Khóa là 1 hoặc kết hợp nhiều thuộc tính để phân biệt các đối tượng trong thực thể với nhau. Vd: MãNV là 1 thuộc tính dùng để phân biệt các nhân viên trong tập thực thể Nhân Viên. Nếu biết MãNV của 1 nhân viên, ta sẽ tìm được tên của nhân viên, địa chỉ và kỹ năng của nhân viên đó, dựa trên dữ liệu của thực thể Nhân Viên.
- ➔ Yêu cầu đối với khóa của thực thể:
 1. Không thay đổi giá trị khi thể hiện tương ứng còn tồn tại
 2. Là thuộc tính không rỗng (Not Null)
 3. Không chứa thuộc tính có cấu trúc. Mã vật tư dạng "kho,loại" là thuộc tính có cấu trúc (không nguyên tố).

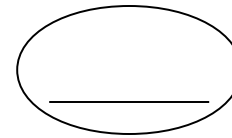
Các ký hiệu dùng trong ERD



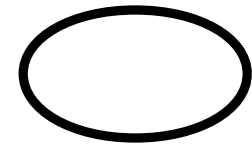
Entity



Attribute

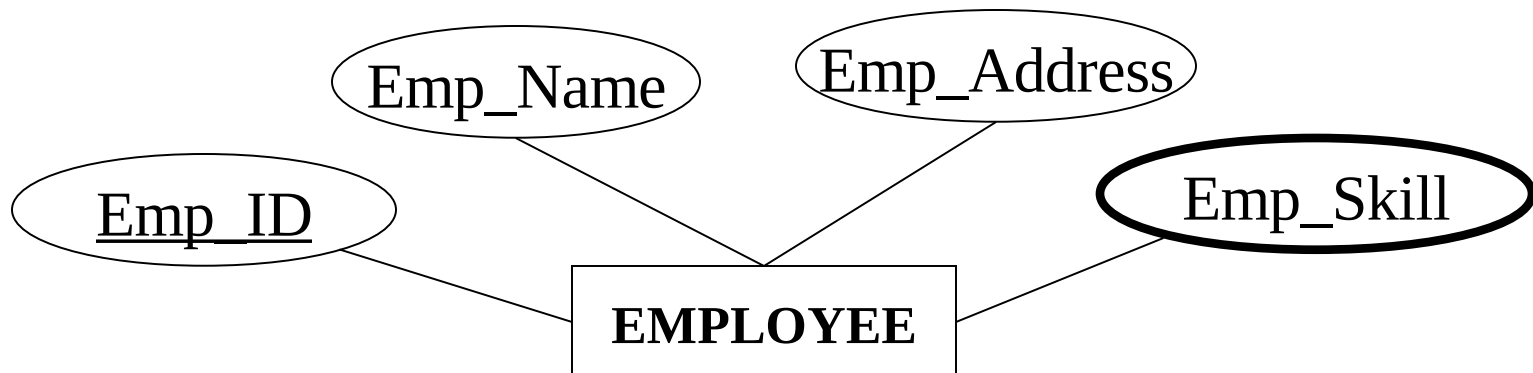


Key



Multivalued
attribute

Ví dụ: một nhân viên có 1 mã nhân viên dùng để phân biệt. Cơ quan chỉ quan tâm quản lý tên nhân viên, địa chỉ nhà riêng, và các kỹ năng của từng nhân viên. Thực thể nhân viên được diễn tả như sau:

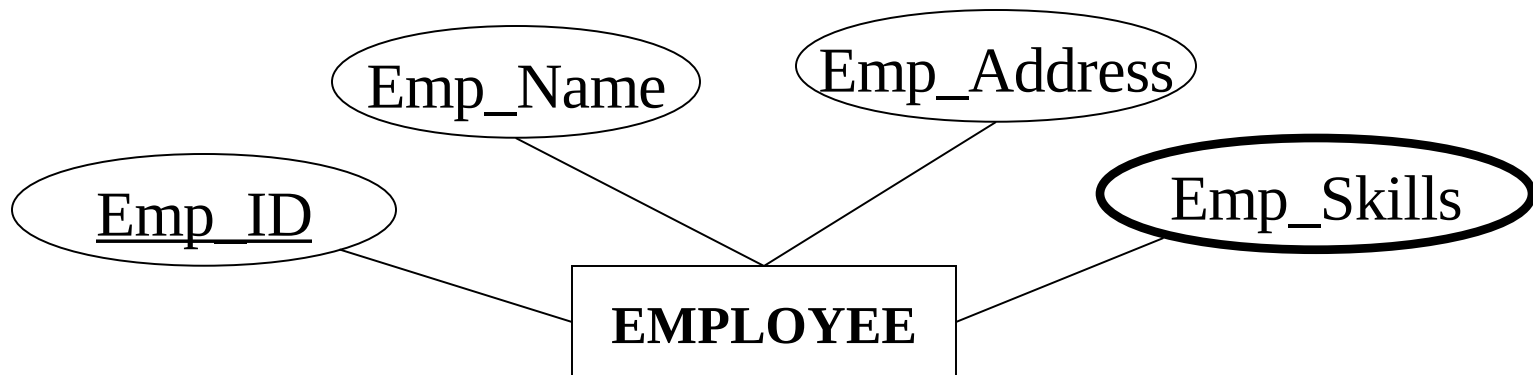


Thuộc tính đa trị (Multivalued Attribute)

⇒ là một thuộc tính có nhiều giá trị được sử dụng đồng thời để mô tả cho một thể hiện của thực thể

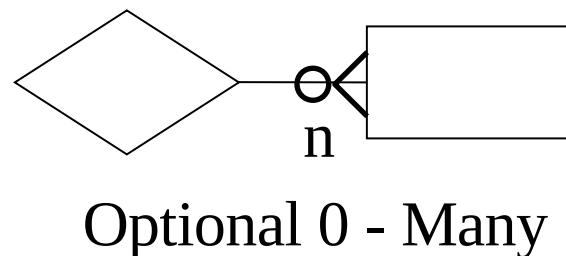
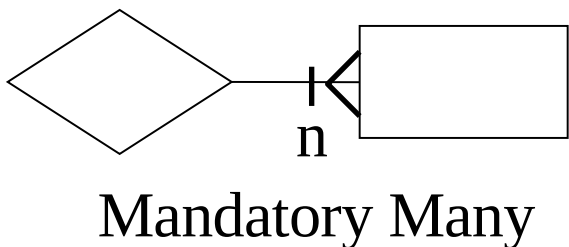
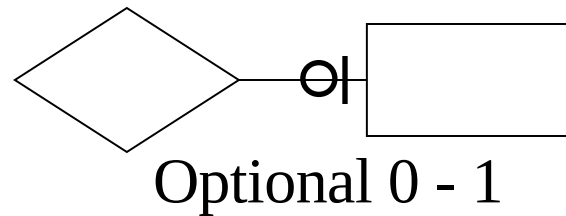
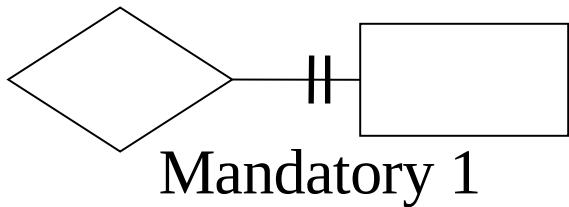
- Vd: thuộc tính “Skill” của 1 nhân viên. Một nhân viên thường có nhiều kỹ năng, skill sẽ có nhiều giá trị khác nhau

EMP_ID	EMP_Name	EMP_Skill
0210-67	Susan	Language
0210-67	Susan	Interpersonal

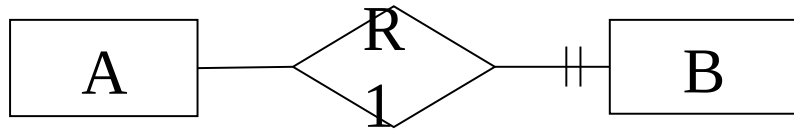


Quan hệ

- ➔ Là mối liên kết giữa một hoặc nhiều thực thể để chỉ ra sự liên kết về nội dung (và ý nghĩa) giữa các thực thể trong mối liên kết. Ví dụ: “Mỗi SINH VIÊN đăng ký nhiều MÔN HỌC”. Sự liên kết nội dung giữa thực thể Sinh viên và thực thể Môn học là việc đăng ký môn để học của mỗi sinh viên.
- ➔ **Cardinality**: là số thể hiện của thực thể B có thể (hoặc phải) liên kết với mỗi thể hiện của thực thể A. Vd: một sinh viên phải đăng ký học ít nhất là 1 môn, và nhiều nhất là 6 môn trong một học kỳ.



Số quan hệ (Cardinality)



Mỗi thể hiện của A có đúng 1 thể hiện tương ứng ở B theo quan hệ R1 (cardinality = [1,1]).



Mỗi thể hiện của A chỉ có 1 thể hiện tương ứng ở B, hoặc không có thể hiện tương ứng theo quan hệ R2 (cardinality = [0,1]).



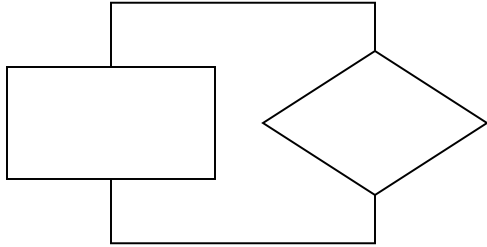
Mỗi thể hiện của A có ít nhất là 1 và tối đa là N thể hiện tương ứng ở B theo quan hệ R3 (cardinality = [1,N]).



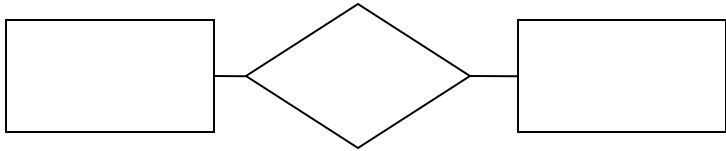
Mỗi thể hiện của A có tối đa là N thể hiện tương ứng ở B, hoặc không có thể hiện tương ứng theo quan hệ R4 (cardinality = [0,N]).

Relationship

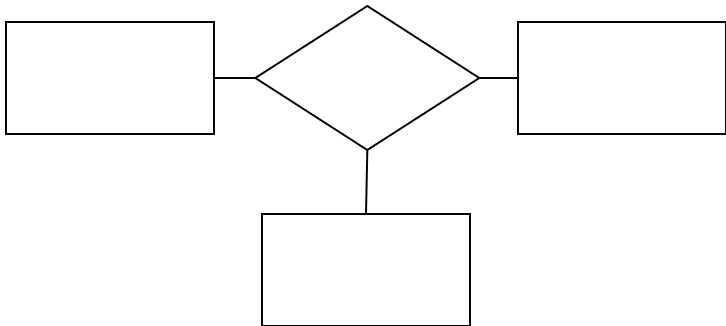
Có 3 loại quan hệ cơ bản giữa các thực thể:



Unary relationship: liên kết trên 1 thực thể

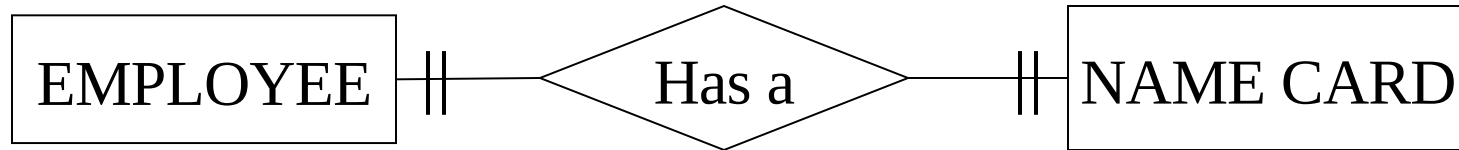


Binary relationship: liên kết 2 thực thể



Ternary relationship: liên kết 3 thực thể

Binary Relationship



One to One (1:1)

“Một nhân viên phải có (duy nhất) 1 bảng tên.”



One to Many (1:N)

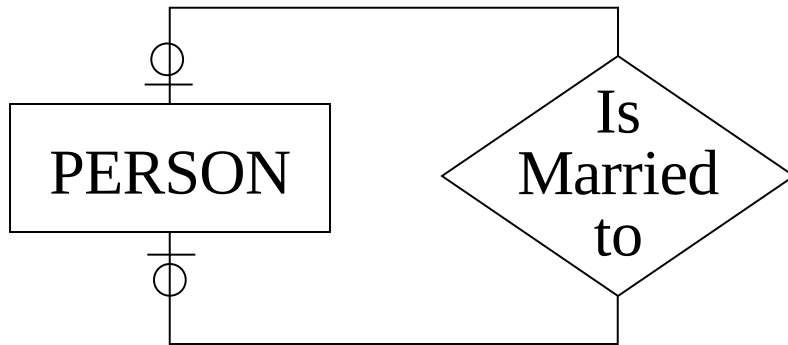
“Một dây chuyền sản phẩm phải chứa 1 hoặc nhiều sản phẩm. Một sản phẩm phải thuộc 1 dây chuyền sản xuất.”



Many to Many (M:N)

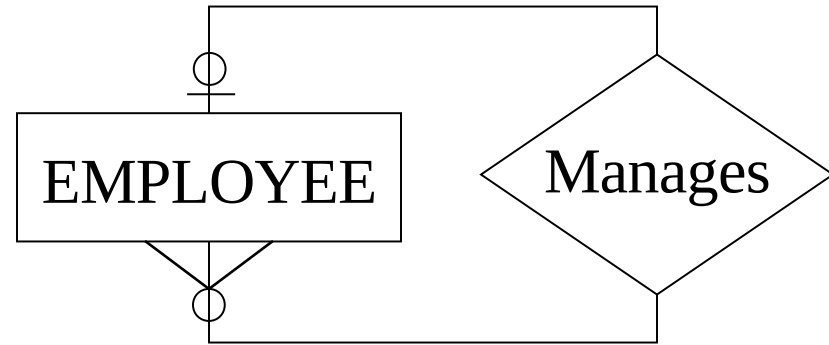
“Một sinh viên phải đăng ký 1 hoặc nhiều môn học. Một môn học có thể có nhiều sinh viên đăng ký, hoặc không có sinh viên đăng ký”

Unary Relationship



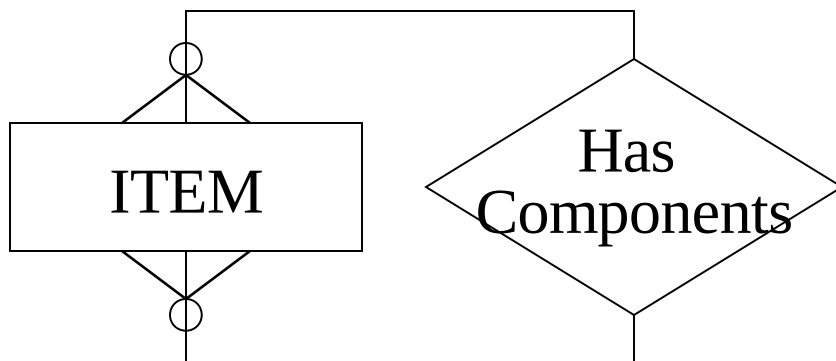
One to One

“Một người chỉ được kết hôn với một người khác”



One to Many

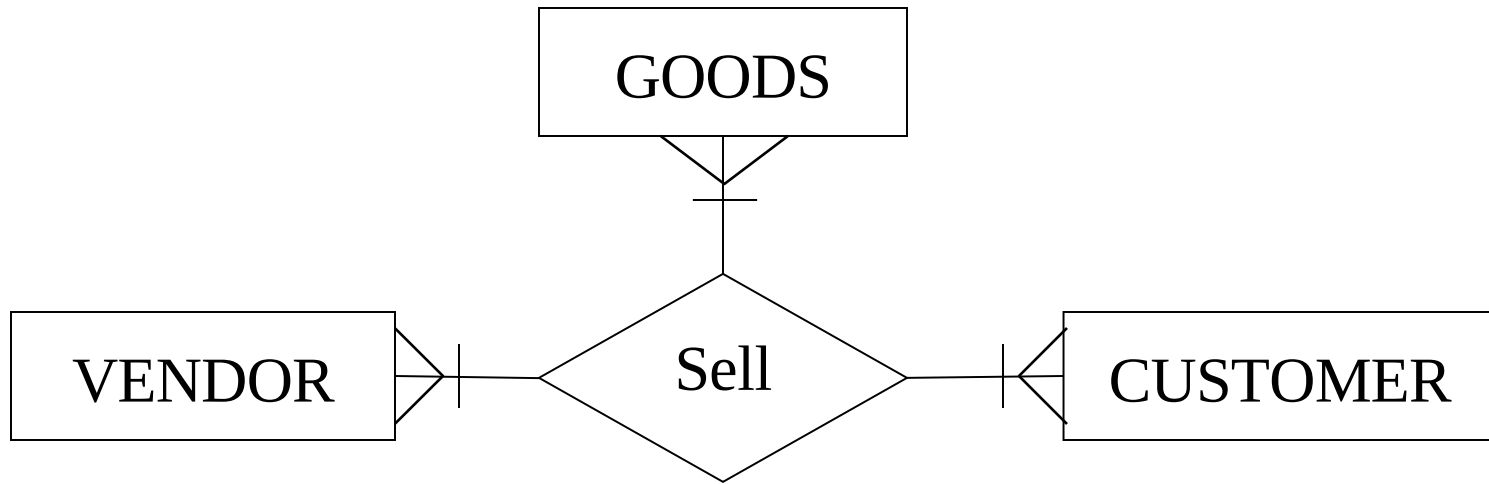
“Một nhân viên có thể quản lý nhiều nhân viên”



Many to Many

“1 Item có thể có nhiều thành phần, mỗi thành phần cũng là 1 Item có thể được sử dụng trong nhiều component khác”

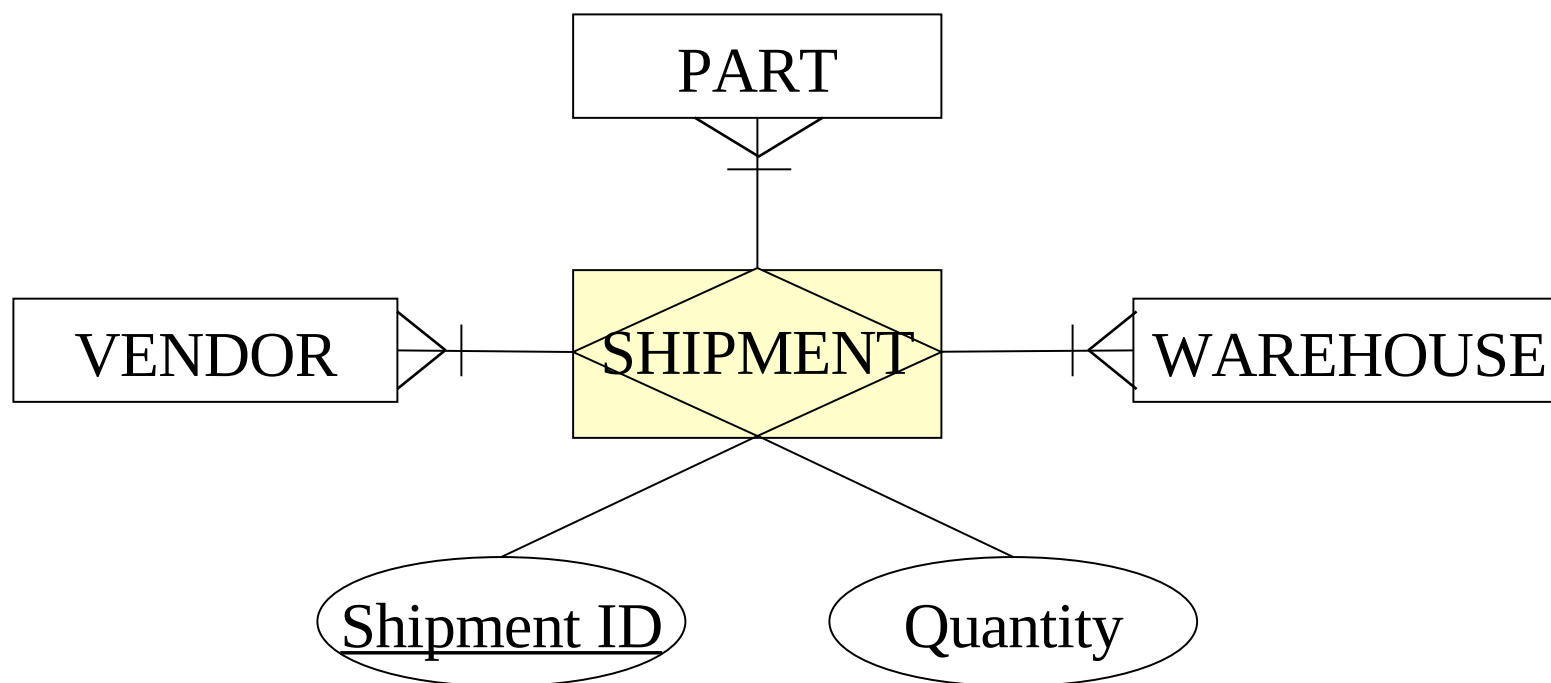
Ternary Relationship



“Một nhà cung cấp có thể bán nhiều mặt hàng cho nhiều khách hàng; khách hàng có thể mua hàng từ nhiều nhà cung cấp khác nhau ”

Associative Entity (thực thể liên kết)

➔ Là thực thể liên kết các thể hiện của một hoặc nhiều thực thể khác và có thêm các thuộc tính riêng trên liên kết giữa các thực thể



Thiết lập lược đồ ERD

1. *Định nghĩa các thực thể*, dựa trên vai trò, ý nghĩa của thực thể đối với hệ thống. Nên chọn danh từ để dùng làm tên cho thực thể, vd: MONHOC, SINHVIEN, KHOA,..
2. *Định nghĩa các quan hệ giữa các thực thể*. Tên của các quan hệ thường được diễn tả bằng động từ để chỉ các hành động, sự kiện liên kết các thể hiện trong các thực thể có quan hệ nhau.
3. *Xác định các thuộc tính của thực thể và quan hệ*. Thuộc tính của thực thể (hoặc quan hệ) là những đặc tính mà tất cả các thể hiện của thực thể (hoặc quan hệ) đều có. Thêm thuộc tính để tăng tính mô tả, hoặc để có thể dữ liệu phân biệt các thể hiện. Bỏ bớt thuộc tính nếu chúng dư thừa hoặc không liên quan đến vai trò, ý nghĩa của thực thể trong hệ thống.
4. *Xác định cardinality cho mỗi quan hệ*.

Biểu đồ thực thể liên kết

