

TRƯỜNG CAO ĐẲNG NGHỀ CÔNG NGHIỆP HÀ NỘI

Chủ biên: Vũ Thị Kim Phượng
Đồng tác giả: Nguyễn Thị Nhung



GIÁO TRÌNH PHÂN TÍCH THIẾT KẾ HỆ THỐNG THÔNG TIN (Lưu hành nội bộ)

Hà Nội năm 2011

Tuyên bố bản quyền

Giáo trình này sử dụng làm tài liệu giảng dạy nội bộ trong trường cao đẳng nghề Công nghiệp Hà Nội

Trường Cao đẳng nghề Công nghiệp Hà Nội không sử dụng và không cho phép bất kỳ cá nhân hay tổ chức nào sử dụng giáo trình này với mục đích kinh doanh.

Mọi trích dẫn, sử dụng giáo trình này với mục đích khác hay ở nơi khác đều phải được sự đồng ý bằng văn bản của trường Cao đẳng nghề Công nghiệp Hà Nội

CHƯƠNG 1: MỞ ĐẦU

Đặt vấn đề

Việc ứng dụng máy tính trong công tác quản lý thông tin ở nước ta hiện nay đang phát triển mạnh mẽ. Nhu cầu nắm bắt, trao đổi thông tin nhanh quyết định không nhỏ đến sự thành công trong kinh doanh của doanh nghiệp. Song song với việc phát triển phần cứng và sử dụng các phần mềm xử lý dữ liệu có sẵn, nhiều công ty đã tiến hành tự xây dựng hoặc tìm mua các chương trình quản lý dữ liệu phù hợp với đặc thù công việc riêng của mình.

Công việc xây dựng một chương trình quản lý, lâu nay vẫn do các lập trình viên đảm trách. Nhiều người trong số họ thường có thói quen bắt đầu công việc bằng cách phác thảo ra cơ sở dữ liệu, xong bắt tay vào xây dựng chương trình ngay. Trong quá trình lập trình, thấy thiếu thông tin cần quản lý nào thì điều chỉnh cơ sở dữ liệu và sửa lại chương trình. Cách làm này thoạt trông thì có vẻ tiện lợi, nhưng thực tế lại khiến lập trình viên mất rất nhiều thời gian và công sức, công ty phải chịu phí tổn cao để hoàn chỉnh một hệ thống thật sự hiệu quả, đáp ứng đúng nhu cầu của người sử dụng. Và như thế thì chỉ có thể tạm chấp nhận được với các chương trình nhỏ và đơn giản.

Đối với những hệ thống quản lý tương đối lớn, việc tổ chức chính xác cơ sở dữ liệu ngay từ đầu là chuyện không đơn giản, nhưng lại rất cần thiết, vì nó sẽ giảm phí tổn, thời gian lập trình, cũng như đảm bảo quá trình xử lý thông tin của hệ thống đạt hiệu quả cao nhất. Để làm được việc đó, cần phải thực hiện việc khảo sát và thiết kế hệ thống. Công việc này thường được đảm trách bởi những nhà phân tích thiết kế hệ thống nhiều kinh nghiệm.

Phân tích và thiết kế có cần phương pháp ?

Có việc gì làm một cách nghiêm túc mà không cần áp dụng một phương pháp ? Hướng chỉ việc phân tích và thiết kế một hệ thống vốn là một việc phức tạp và trường kỳ, đương nhiên là rất cần được triển khai theo một phương pháp hợp lý.

Vậy phương pháp phân tích và thiết kế là gì ? và nó giúp gì cho người xây dựng hệ thống? Một phương pháp phân tích và thiết kế là sự hợp thành của ba yếu tố:

Một tập hợp các khái niệm và mô hình, bao gồm các khái niệm cơ bản sử dụng trong phương pháp cùng với các cách biểu diễn chúng (thường dưới dạng đồ thị)

Một tiến độ triển khai, bao gồm các bước đi lần lượt, các hoạt động cần làm

Một công cụ trợ giúp, là một phần mềm giúp cho việc triển khai hệ thống thực hiện theo phương pháp được chặt chẽ và nhanh chóng.

Trước những năm 60, chưa định hình những phương pháp rõ rệt cho phân tích và thiết kế hệ thống. Người ta xây dựng hệ thống một cách tùy tiện, theo sở thích và kinh nghiệm cá nhân. Kết quả phân tích là những tập tài liệu dày cộm, trình bày dưới dạng văn tự dài dòng, khó đọc và khó trao đổi. Từ những năm 70 đến nay, nhiều phương pháp phân tích và thiết kế lần lượt ra đời. Mỗi phương pháp đều có những ưu và nhược điểm riêng, có thể được ưa chuộng ở nơi này, nhưng lại ít được ưa chuộng ở nơi khác. Sự phong phú và đa dạng trong phương pháp như vậy cũng có nghĩa là sự không thống nhất, không chuẩn hóa.

Tuy nhiên trải qua thời gian, một số phương pháp đã tỏ ra là có sức sống dẻo dai, bám trụ được cho đến ngày hôm nay. Trong số này phải kể trước hết đó là các phương pháp được gọi dưới một cái tên chung là các phương pháp có cấu trúc (hay các phương pháp trên xuống). Cũng không thể không kể đến một trào lưu mới, mãnh liệt: Đó là sự ra đời ồ ạt, từ năm 1990, của các phương pháp phân tích và thiết kế hướng đối tượng, để rồi qui vào một cái chuẩn, xuất hiện năm 1997, là UML (Unified Modeling Language).

Các phương pháp hướng chức năng

- **Phương pháp SADT (*Structured Analysis and Design Technie*)** xuất phát từ Mỹ, ý tưởng cơ bản của nó là phân rã một hệ thống lớn thành các hệ thống con đơn giản hơn.

SADT được xây dựng dựa trên 7 nguyên lý sau:

- o Sử dụng một mô hình.
- o Phân tích trên xuống
- o Dùng một mô hình chức năng và một mô hình quan niệm (còn được gọi là “mô hình thiết kế”)
- o Thể hiện tính đối ngẫu của hệ thống
- o Sử dụng các biểu diễn dưới dạng đồ họa
- o Phối hợp hoạt động của nhóm
- o Ưu tiên tuyệt đối cho hồ sơ viết

SADT được định nghĩa là phương pháp sử dụng các kỹ thuật:

- o Dòng dữ liệu hay còn gọi là lưu đồ dữ liệu (Data flow diagrams)
- o Từ điển dữ liệu (data dictionary)
- o Anh ngữ có cấu trúc
- o Cây quyết định

Phương pháp SADT có nhiều ưu điểm như dựa vào nguyên lý phân tích cấu trúc, thiết kế theo lối phân cấp, dựa trên các lưu đồ chức năng, tạo được các liên kết một nhiều, bảo đảm từ một dữ liệu vào sản xuất nhiều dữ liệu ra. Nhưng nhược điểm của nó là không bao gồm tiến trình phân tích và nếu không thận trọng sử dụng SADT có thể đưa tiến trình trùng lặp thông tin.

- **Phương pháp MERISE (MEthod pour Rassembler les Idees Sans Effort)** của Pháp dựa trên các mức bất biến (còn gọi là mức trừu tượng hoá) của hệ thống thông tin như mức quan niệm, mức tổ chức, mức vật lý và có sự kết hợp với mô hình.

Phương pháp MERISE là phương pháp phân tích có nguồn gốc từ Pháp, ra đời từ những năm cuối thập niên 70. Nó là kết quả nghiên cứu của nhiều tập thể nghiên cứu tin học nhằm đáp ứng các chờ đợi của người sử dụng. ý thức được về sự lạc hậu của các phương pháp phân tích cổ điển thế hệ thứ nhất.

Ý tưởng cơ bản của phương pháp MERISE là xuất phát từ ba mặt cơ bản sau:

- **Mặt thứ nhất:** Quan tâm đến chu kỳ sống của hệ thống thông tin, trải qua nhiều giai đoạn: “thai nghén” – quan niệm/ý niệm - quản trị - chết. Chu kỳ sống này đối với hệ thống tổ chức lớn có thể kéo dài từ 10-15 năm.
- **Mặt thứ hai:** Đề cập đến chu kỳ đặc tả của hệ thống thông tin còn được gọi là chu kỳ trừu tượng.

Hệ thống thông tin tựu trung lại như một toàn thể được miêu tả bởi nhiều tầng: “Bộ nhớ” của hệ thống thông tin được mô tả trên bình diện quan niệm, kế đó trên bình diện logic và cuối cùng trên bình diện vật lý.

“Quá trình xử lý” được mô tả trên bình diện quan niệm, kế tiếp là trên bình diện tổ chức và cuối cùng là trên bình diện tác nghiệp.

Mỗi tầng được mô tả dưới dạng mô hình tập trung tập hợp các thông số chính xác. Theo đó kho những thông số của tầng dưới tầng trước, tầng đang mô tả không biến đổi và nó chỉ thay đổi khi các tham số của mình thay đổi.

Mỗi mô hình được mô tả thông qua một hình thức dựa trên các nguyên tắc, nguyên ký ngữ vựng và cú pháp xác định. Có những qui tắc chuyển cho phép chuyển từ mô hình này sang mô hình khác một cách tự động nhiều hay ít.

- **Mặt thứ ba:** Mặt này có liên quan đến chu kỳ các quyết định (Lycle des Decisions) cần phải ra trong suốt chu kỳ sống của sản phẩm. Những quyết định có liên quan đến nội dung của những mô hình khác nhau của chu kỳ trừu tượng, đến các hình thái của quan niệm và liên quan đến sự phát triển của hệ thống

Đặc trưng cơ bản của phương pháp MERSE là:

- o Nhìn toàn cục
- o Tách rời các dữ liệu và xử lý
- o Tiếp cận theo mức

Có thể tóm tắt nội dung thứ hai và thứ ba thể hiện qua việc nhận thức và xây dựng các loại mô hình trong qua trình phân tích và thiết kế bằng bảng sau:

Mức	Dữ liệu	Xử lý
Quan niệm	Mô hình quan niệm dữ liệu	Mô hình quan niệm xử lý
Tổ chức	Mô hình logic dữ liệu	Mô hình tổ chức xử lý
Kỹ thuật	Mô hình vật lý dữ liệu	Mô hình tác vụ xử lý

Ưu điểm của phương pháp MERSE là cơ sở khoa học vững chắc. Hiện tại nó là một trong những phương pháp phân tích sử dụng dùng nhiều ở Pháp và các nước Châu Âu khi phải phân tích và thiết kế các hệ thống lớn. **Nhược điểm** của phương pháp này là công kênh, do đó để giải quyết các áp dụng nhỏ việc sử dụng phương pháp này nhiều khi dẫn đến việc kéo dài thời gian, năng nề không đáng có.

- **CASE (Computer-Aided System Engineering)** - phương pháp phân tích và thiết kế tự động nhờ sự trợ giúp của máy tính.

Từ kinh nghiệm và nghiên cứu trong quá trình xây dựng hệ thống, hãng Oracle đã đưa ra một tiếp cận công nghệ mới - Phương pháp luận phân tích và thiết kế hệ thống CASE*Method. Đây là một cách tiếp cận theo hướng "TOPDOWN" và rất phù hợp với yêu cầu xây dựng một hệ thống thông tin trong các doanh nghiệp sản xuất kinh doanh thương mại.

Các phương pháp hướng đối tượng

- **Phương pháp HOOD (Hierarchical Object Oriented Design)** là một phương pháp được lựa chọn để thiết kế các hệ thống thời gian thực. Những phương pháp này lại yêu cầu các phần mềm phải được mã hoá bằng ngôn ngữ lập trình ADA. Do vậy phương pháp này chỉ hỗ trợ cho việc thiết kế các đối tượng mà không hỗ trợ cho các tính năng kế thừa và phân lớp.

- **Phương pháp RDD (Responsibility Driven Design)** dựa trên việc mô hình hoá hệ thống thành các lớp. Các công việc mà hệ thống phải thực hiện được phân tích và chia ra cho các lớp của hệ thống. Các đối tượng trong các lớp của hệ thống trao đổi các thông báo với nhau nhằm thực hiện công việc đặt ra. Phương pháp RDD hỗ trợ cho các khái niệm về lớp, đối tượng và kế thừa trong cách tiếp cận hướng đối tượng.

- **Phương pháp OMT (*Object Modelling Technique*)** là một phương pháp được xem là mới nhất trong cách tiếp cận hướng đối tượng. Phương pháp này đã khắc phục được một số nhược điểm của các phương pháp tiếp cận hướng đối tượng trước mắc phải. Trên mặt lý thuyết ta thấy cách tiếp cận hướng đối tượng có các bước phát triển hơn so với tiếp cận hướng chức năng. Nhưng trong thực tế việc phân tích và thiết kế hệ thống theo cách tiếp cận hướng đối tượng gặp rất nhiều khó khăn vì chưa có nhiều các công cụ phát triển hỗ trợ cho việc thiết kế hướng đối tượng. Chính vì vậy cách tiếp cận này vẫn chưa được phát triển rộng rãi.

Môn học phân tích thiết kế hệ thống sẽ trình bày theo phương pháp gì?

Môn học phân tích thiết kế hệ thống sẽ trình bày theo phương pháp SADT là phương pháp thông dụng trong khối Anh, Mỹ.

Ý tưởng cơ bản của SADT là ở chỗ: Phân rã một hệ thống lớn thành các phân hệ nhỏ hơn và đơn giản hơn, dựa trên các nguyên lý :

Sử dụng mô hình; phân tích từ trên xuống, dùng mô hình chức năng, mô hình dữ liệu; sử dụng các biểu diễn dưới dạng đồ họa, phối hợp hoạt động nhóm

CHƯƠNG 2: ĐẠI CƯƠNG VỀ HỆ THỐNG THÔNG TIN TRONG QUẢN LÝ

I. Khái niệm hệ thống:

Thuật ngữ hệ thống không phải là mới. Từ lâu người ta đã nói đến hệ thống mặt trời, hệ thống triết học, hệ thống giao thông, hệ thống pháp luật, hệ thống tuần hoàn, hệ thống thủy lực, hệ thống thông tin,...

1. Định nghĩa hệ thống:

Là một tập hợp gồm nhiều phần tử có các mối ràng buộc lẫn nhau và cùng hoạt động hướng tới một mục tiêu nhất định nào đó.

Mục tiêu của hệ thống thường thể hiện dưới cái vào/cái ra.

Ví dụ: Vật tư, tiền, sản phẩm, dịch vụ thông tin, ...

2. Các phần tử của hệ thống:

Các phần tử của hệ thống được hiểu là các thành phần hợp thành của nó và được hiểu theo nghĩa rất rộng.

Các phần tử đó có thể rất đa dạng, chẳng hạn trong hệ mặt trời thì các phần tử là mặt trời, trái đất, Hải vương tinh, Hỏa tinh, ...; trong hệ thần kinh thì các phần tử là bộ óc, dây thần kinh, tủy sống,...; có khi các phần tử lại là những đối tượng trừu tượng, như là một phương pháp, một lập luận, một qui tắc,... như trong hệ thống tư tưởng. Như vậy các phần tử có thể rất khác biệt về bản chất, không những giữa các hệ thống khác nhau, mà có thể ngay trong cùng một hệ thống.

Các phần tử của một hệ thống có thể rất đơn giản, những cũng có thể rất phức tạp, thậm chí có thể là một hệ thống con

3. Quan hệ giữa các phần tử

Các phần của một hệ thống không phải được tập hợp lại một cách ngẫu nhiên, rời rạc, mà giữa chúng luôn tồn tại những quan hệ (hay các mối ràng buộc lẫn nhau), tạo thành một cấu trúc (hay tổ chức). Chẳng hạn trong một hệ thống hành chính, gồm các cán bộ và nhân viên, thì giữa họ tồn tại các mối ràng buộc về phân cấp, phân quyền, các quan hệ về đoàn thể, các quan hệ về dân sự,...

Trong các quan hệ đang tồn tại ta cần đặc biệt quan tâm đến các quan hệ ổn định và lâu dài, chẳng hạn như A là nhân viên của B. Còn các quan hệ mang

tính tạm như A và C vừa được cử đi công tác với nhau thì sẽ không được đề cập đến.

Khi nói đến Ổn định, không nhất thiết phải hiểu là hoàn toàn bất biến, tĩnh tại. Trên thực tế hầu hết các hệ thống đáng quan tâm đều có tính biến động. Biến động song vẫn giữ sự Ổn định trong tổ chức, trong quan hệ giữa các phần tử nghĩa là vẫn giữ cái bản chất hay các đặc trưng cốt lõi của hệ thống.

4. Sự hoạt động của hệ thống

Sự biến đổi của hệ thống thể hiện trên hai mặt:

Sự tiến triển, tức là các thành phần của nó (các phần tử và các quan hệ) có thể có phát sinh, có tăng trưởng, có suy thoái, có đào thải.

Sự hoạt động, tức là các phần tử của hệ thống, trong các mối ràng buộc đã định, cùng cộng tác với nhau để thực hiện một mục đích chung của hệ thống

Quá trình hoạt động của hệ thống là quá trình biến đổi những cái vào thành những cái ra nhất định. Chẳng hạn, một hệ thống sản xuất thì nhận vào các nguyên vật liệu, tiền và dịch vụ để sản xuất ra hàng hoá, vật tư,...

5. Quá trình trừu tượng hoá trong phân tích và thiết kế hệ thống

Trong hệ thống có sự phân cấp từ trên xuống dưới, nên xem xét hệ thống ở mức độ trừu tượng hoá nhất định. Quy trình này tiến hành như sau: Khi bắt đầu khảo sát sơ bộ hệ thống ta phải ghi nhận một cách trung thực những gì đang xảy ra trong thực tế, tức là chúng ta phải trả lời cho các câu hỏi dạng:

Quá trình trừu tượng hoá phải dựa trên các câu hỏi:

Hệ thống làm gì? (What)
Hệ thống thực hiện như thế nào?(how)
Bằng dụng cụ hay phương tiện gì?
Ai thực hiện? Làm ở đâu? Làm lúc nào?

Trả lời cho các câu hỏi trên chúng ta sẽ thu được mô tả hệ thống ở mức vật lý.

Tuy nhiên, các yếu tố vật lý có thể che khuất bản chất của hệ thống, vì vậy chúng ta không thể thấy hết các khuyết điểm của hệ thống.

Để nhận thức đúng bản chất của hệ thống ta cần loại bỏ các yếu tố vật lý của hệ thống, nghĩa là phải loại bỏ các yếu tố phụ, không chính yếu, chỉ giữ

lại các yếu tố phản ánh bản chất hệ thống. Để thực hiện được công việc này cần trả lời cho các câu hỏi thuộc hai dạng sau:

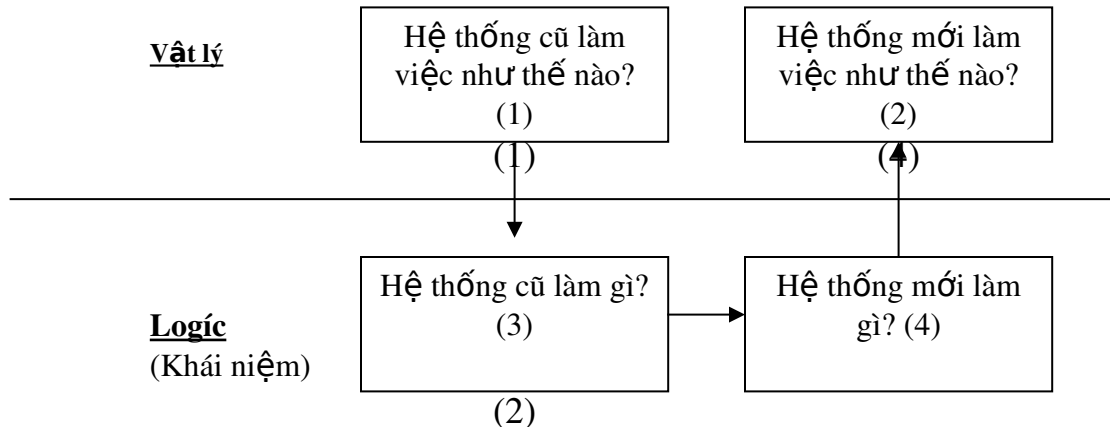
Hệ thống là gì? Và hệ thống làm gì ?

Sau khi trả lời các câu hỏi dạng trên chúng ta sẽ thu được mô tả hệ thống ở mức logic (khái niệm)

Đối với hệ thống cần xây dựng, việc mô tả đầy đủ nó ở mức logic là rất cần thiết trước khi tính đến các biện pháp cài đặt.

Giai đoạn thiết kế chính là lúc xem xét các biện pháp cài đặt cụ thể, nói như vậy có nghĩa là ở giai đoạn này ta phải diễn tả hoạt động của hệ thống ở mức độ vật lý, vpi đầy đủ các yếu tố về cài đặt và thực hiện.

Tóm tắt quá trình phân tích diễn ra như sau:



Cách mô tả này cũng cho ta hình dung được qui trình phân tích và thiết kế hệ thống:

Bắt đầu từ hệ thống cũ:

Xuất phát từ mức vật lý với mô tả các chức năng nghiệp vụ cùng các biện pháp, phương tiện cụ thể (1)

Chuyển sang mức logic bằng cách loại bỏ các yếu tố phụ (2)

Sang hệ thống mới:

Hình thành hệ thống mới ở mức logic (3)

Bổ xung thêm các biện pháp và phương tiện cụ thể,... để chuyển sang mức vật lý (4)

II. Hệ thống kinh doanh/ dịch vụ:

1. Khái niệm hệ thống kinh doanh:

Là hệ thống mà mục đích là kinh doanh hay dịch vụ .

- **Kinh doanh** là hoạt động của con người nhằm mang lại *lợi nhuận* (tức là thu giá trị thặng dư). Chẳng hạn sản xuất, phân phối hay lưu thông sản phẩm là các hoạt động kinh doanh.
- **Dịch vụ** là hoạt động của con người nhằm mang lại *lợi ích* (tức là cung cấp giá trị sử dụng). Chú ý rằng có những dịch vụ là phi lợi nhuận (bởi ở đó không thể có tăng năng suất, để từ đó tạo ra giá trị thặng dư), ví dụ các hoạt động giáo dục, y tế, từ thiện,...

Hệ thống kinh doanh / dịch vụ nói ở đây có thể có những qui mô khác nhau. *Qui mô nhỏ* như là một phân xưởng, một cửa hàng. *Qui mô vừa* như là một nhà máy, một công ty, một bệnh viện, một trường đại học. *Qui mô lớn* như là một tổng công ty, một ngành sản xuất, một tập đoàn kinh doanh đa quốc gia,... Để cho gọn, sau này ta gọi hệ thống kinh doanh/dịch vụ là *doanh nghiệp*

Đặc điểm chung của các hệ thống kinh doanh / dịch vụ so với các hệ thống khác (như các hệ thống vật lý, kỹ thuật hay sinh học,..) : Chúng là của con người và có sự tham gia của con người (nghĩa là mang những ưu, khuyết điểm của con người)

- o Cửa của con người, cho nên các mục tiêu của chúng là do con người định ra
- o Có con người tham gia, cho nên con người thường xuyên góp phần thúc đẩy hay kìm hãm sự phát triển của hệ thống, bởi vì con người *có trí thông minh, có khả năng sáng tạo, có tình cảm, có tham vọng*,...

Đặc điểm trên dẫn tới hai nét nổi bật của các hệ thống kinh doanh/dịch vụ:

- o Vai trò và *cơ chế điều khiển* (trong kinh doanh thường gọi là sự quản lý) là rất quan trọng, nhằm giữ cho hệ thống hướng đúng đích và đạt kết quả với chất lượng cao.
- o Vai trò của *thông tin* cũng rất quan trọng, nhằm phục vụ cho nhu cầu giao tiếp, trao đổi giữa con người với nhau

2. Các hệ thống con:

Hệ thống kinh doanh/ dịch vụ được chia thành ba hệ thống con

a) Hệ thống tác nghiệp: Bao gồm người, phương tiện, phương pháp, quy trình, tham gia trực tiếp vào việc thực hiện mục tiêu kinh doanh. Đây là hệ thống trực tiếp sản xuất.

b) Hệ thống quản lý (hệ quyết định): Bao gồm người, phương tiện, phương pháp, quy trình, tham gia vào việc đề xuất các quyết định trong kinh doanh.

VỀ mặt hình thức thì hoạt động quản lý luôn luôn là một dãy nối tiếp của hai việc:

- Đề xuất một quyết định kinh doanh
- Thực thi một quyết định kinh doanh

Ta hiểu *quyết định* chình là sự chọn lựa một trong những phương án hành động có thể để giải quyết một vấn đề nào đó

Tâm cỡ của các quyết định:

Quyết định chiến lược hay kế hoạch, có ảnh hưởng lâu dài và rộng khắp đối với hệ thống

Quyết định chiến thuật hay điều phối: Nhằm hiệu chỉnh các công việc cho hợp lý, nó có ảnh hưởng ngắn hạn hay cục bộ đối với hệ thống

Quyết định tác vụ: Nhằm thực thi các nhiệm vụ cụ thể trong hoạt động của hệ thống (là công việc cụ thể cho từng ngày, từng giờ)

Vậy *quyết định* Là sự lựa chọn một trong những phương án hành động có thể

Một quyết định trước khi được đề xuất phải trải qua hai bước:

- Thu thập thông tin (tìm hiểu vấn đề)
- Chọn phương án

Có hai cách chọn phương án

- Chọn theo giải thuật
- Chọn theo trực quan

Lưu ý:

Đứng trước một quyết định phải chú trọng tới tầm quan trọng của nó, quyết định mang tính chiến lược chiến thuật hay chỉ là những quyết định mang tính nhất thời.

Chức năng của hệ quyết định: Làm nhiệm vụ quản lý hệ thống thừa hành theo các mục đích đã xác định cho hệ thống tổng thể, nó hướng dẫn, hiệu chỉnh, kiểm soát quy trình kỹ thuật hay quy trình hành chính. Toàn bộ hệ quyết định biểu thị một tập hợp các đường lối các nguyên lý, quy tắc thủ tục và phương tiện để thực hiện các phép biến đổi trên biến điều khiển.

Ví dụ: Một hệ quyết định trong sản xuất áp dụng cho các xưởng đó là phương pháp bố trí và kiểm soát sản xuất.

Một hệ quản lý nhân viên sẽ xác định các phương pháp lựa chọn, đào tạo, đề bạt và đãi ngộ

c) Hệ thống thông tin:

Bao gồm người, phương tiện, phương pháp, quy trình, tham gia vào việc xử lý các thông tin kinh doanh, thông tin kinh doanh là những thông tin liên quan tới việc kinh doanh có 2 loại:

Thông tin tự nhiên: Là loại thông tin giữ nguyên hình trạng như khi phát sinh; chẳng hạn như lời nói, chữ viết, hình ảnh, biểu đồ,...

Thông tin có cấu trúc: Được lưu trữ trong sổ sách, trong các đĩa từ của máy tính bằng những FILE. Đây là những thông tin đã sơ xử lý và là loại thông tin mà ta quan tâm nhất

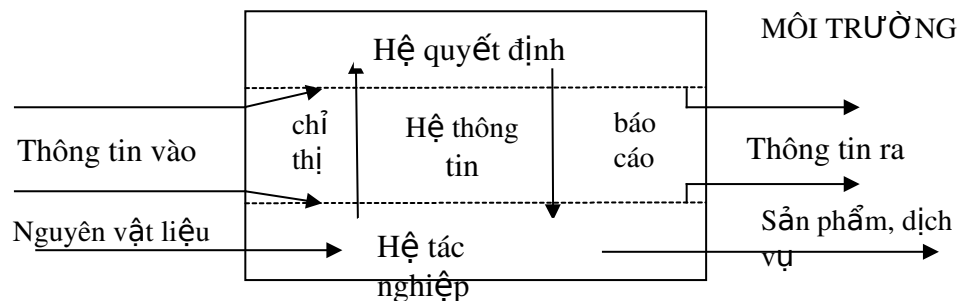
Hai loại thông tin này được xử lý một cách khác nhau, nói một cách rõ hơn thì thông tin tự nhiên là thông tin văn phòng, còn thông tin có cấu trúc là thông tin quản lý.

Nội dung:

- Nhằm phản ánh tình trạng nội bộ
- Nhằm phản ánh sự hoạt động kinh doanh
- Thông tin vào/ra

3. Vai trò và nhiệm vụ của hệ thống thông tin:

a. **Vai trò:** Đóng vai trò trung gian giữa hệ thống quyết định và hệ thống tác nghiệp thông qua mô hình sau:



Hệ thống kinh doanh/dịch vụ

(Hệ chia cắt chỉ mang tính phương pháp luận, nhằm cho ta một cách nhìn, một cách nghiên cứu đối với hệ thống, chứ không phải chia cắt vật lý)

b. Nhiệm vụ chung của hệ thống: Thu thập, lưu trữ, chế biến, truyền đạt các thông tin kinh doanh để đảm bảo hai nhiệm vụ sau:

- *Nhiệm vụ đối ngoại:*

- o Thu thập thông tin từ ngoài đến
- o Đưa các thông báo ra ngoài

- *Nhiệm vụ đối nội:*

- o Đóng vai trò liên lạc giữa các bộ phận
- o Cung cấp các thông tin kinh doanh cho hệ tác nghiệp và hệ quyết định gồm:
 - Thông tin phản ánh tình trạng nội bộ cơ quan
 - Thông tin phản ánh tình hình hoạt động kinh doanh

4. Các bộ phận hợp thành hệ thống thông tin:

Không kể đến con người, phương tiện, quy trình... thì hệ thống thông tin chỉ còn lại hai thành phần cơ bản là Các dữ liệu ghi nhận thực trạng của doanh nghiệp và các xử lý cho phép biến đổi các dữ liệu.

a) **Các dữ liệu:** (Chủ yếu là các thông tin có cấu trúc)

Đó là các thông tin được lưu trữ và duy trì nhằm phản ánh thực trạng hiện thời và quá khứ của doanh nghiệp, các dữ liệu này bao gồm hai loại sau:

- Các dữ liệu phản ánh cấu trúc nội bộ của cơ quan, chẳng hạn như dữ liệu về nhân sự, nhà xưởng, thiết bị,... Cấu trúc cơ quan không phải là cố định, mà có thể co biến động, vì khi có một *sự kiện tiến hoá* xảy ra (chẳng hạn khi có một nhân viên bị chết, một thiết bị mới được bổ sung, ...). Sự kiện tiến hoá thường xảy ra bất chợt, ngoài ý muốn của con người. Sự điều chỉnh lại các dữ liệu cho thích hợp khi có sự kiện tiến hoá xảy ra gọi là sự **cập nhập**

- Các dữ liệu phản ánh các hoạt động kinh doanh của cơ quan như là dữ liệu về sản xuất, mua bán, giao dịch,... Các hoạt động kinh doanh của doanh nghiệp, chẳng hạn như việc hoàn thành một mẻ sản phẩm, một đơn đặt hàng vừa nhận được, ... được gọi là các *sự kiện hoạt động*, khi có một sự kiện hoạt động xảy ra thì phải ghi nhận lại, điều này có nghĩa là làm thay đổi các dữ liệu phản ánh các hoạt động kinh doanh/dịch vụ của doanh nghiệp.

Có hai hình thức thể hiện của các dữ liệu:

- Sổ sách trong văn phòng
- Các tệp, các CSDL trong máy tính

b) **Các xử lý:** Bao gồm các chức năng, nhiệm vụ, các quy trình, qui tắc làm biến đổi thông tin nhằm hai mục đích chính:

- Sản sinh các thông tin theo thể thức qui định, chẳng hạn như các hoá đơn, các báo cáo, các đơn hàng,...

- Trợ giúp các quyết định, thông thường là cung cấp những thông tin cần thiết cho việc chọn lựa một quyết định của lãnh đạo

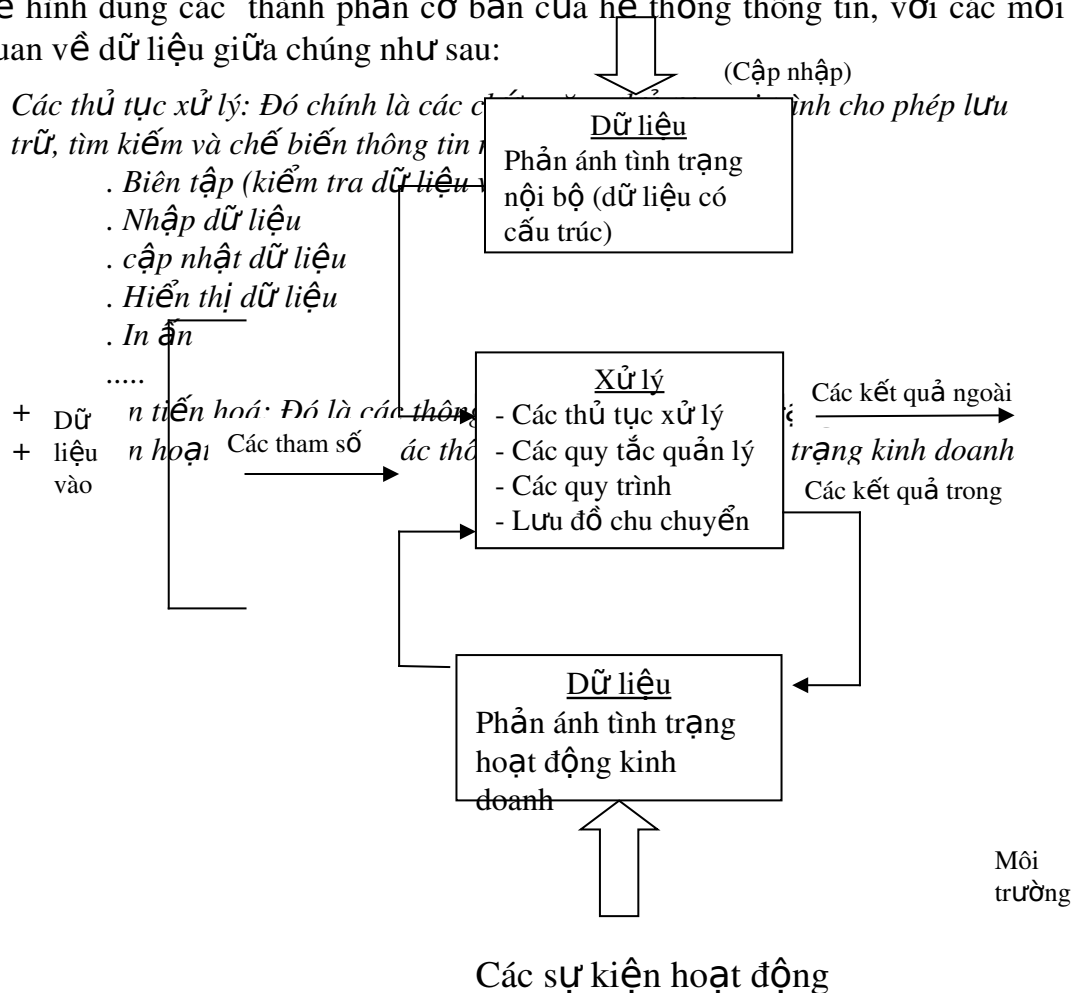
Mỗi xử lý thường là một sự áp dụng một *qui tắc quản lý* định sẵn và diễn ra theo một trật tự định sẵn (gọi là thủ tục). Các qui tắc quản lý và các thủ tục có thể được ấn định bởi hệ thống quyết định của doanh nghiệp, và như vậy chúng có thể bị điều chỉnh theo ý muốn (chẳng hạn các qui tắc tiêu thụ sản phẩm, phương pháp phân phối các trợ cấp, các qui định về khuyến mãi,...), nhưng chúng có thể được ấn định từ bên ngoài doanh nghiệp, đặc biệt là bởi Nhà nước (ví dụ các qui tắc thuế VAT, các tính lương và các bảo hiểm xã hội,...) và như vậy doanh nghiệp không được tùy tiện thay đổi.

Đầu vào của một xử lý có thể là các thông tin phản ánh cấu trúc doanh nghiệp và/hoặc các thông tin phản ánh hoạt động của doanh nghiệp, còn thông tin đầu ra có thể là:

- Các kết quả chuyển trực tiếp cho các cá nhân hay tổ chức ngoài doanh nghiệp như đơn đặt hàng, hoá đơn, thống kê bán hàng, báo cáo tài chính ...được gọi là các *kết quả ngoài*.

- Các kết quả được lưu trữ trở lại vào trong hệ thống để sau này dùng làm đầu vào cho các xử lý khác (thường là các thông tin về tình trạng, về lịch sử hay lưu trữ), được gọi là *Các sự kiện tiến hoá*

Có thể hình dung các thành phần cơ bản của hệ thống thông tin, với các mối liên quan về dữ liệu giữa chúng như sau:



5. Các hệ thống thông tin:

a. Hệ thống thông tin thủ công:

Hệ thống thông tin thủ công là một hệ thống mà trong đó dữ liệu được xử lý bằng phương pháp cảm quan với sự giúp đỡ của các công cụ thô sơ như bàn tính gỗ, máy tính cơ học,... tuy nhiên một hệ thống như vậy sẽ không chính xác, tốn công sức, thời gian và tiền bạc.

b. Hệ thống thông tin tự động hóa

Các hệ thông tin tự động hoá: Là hệ thông tin có máy tính trợ giúp

❖ Mức độ tự động hoá: Có 2 mức độ (2 phương pháp)

- **Tự động bộ phận** : Có sự phân chia giữa người và máy tính, Máy xử lý một phạm vi hẹp, nhiệm vụ hẹp trong hệ thống xử lý thông tin (còn gọi là phương pháp giống: Sử dụng máy tính ở từng bộ phận)

Ưu điểm:

- Dễ dàng thực hiện, rẻ
- Thời gian hoàn thành nhanh

Nhược điểm:

- Dư thừa và đâm đạp lên nhau
- Mâu thuẫn (không toàn vẹn)
- Nối kết các hệ thống trong tương lai khó khăn

- **Tự động hóa toàn bộ**: Máy tính xử lý toàn bộ, liên hoàn các giai đoạn (còn gọi là phương pháp hồ: Sử dụng máy tính ở khắp nơi)

Ưu điểm:

- Cho ta một công cụ quản lý thống nhất, cơ sở dữ liệu chung cho toàn bộ cơ quan
- Không dư thừa dữ liệu

Nhược điểm:

- Đắt, xây dựng tốn thời gian, công sức
- Thay đổi sẽ khó khăn

Việc lựa chọn phương pháp nào để triển khai là tùy thuộc vào bối cảnh cụ thể mà chính phân tích viên hệ thống phải xác định

❖ Phương pháp xử lý thông tin bằng máy tính

- **Xử lý theo mẻ**:(Batch processing) Thông tin đưa đến không được xử lý ngay mà tích lũy lại thành từng mẻ rồi mới xử lý, có tính chất định kỳ (đầy mẻ mới xử lý). Thường dùng cho những công việc tính lương, báo

cáo, tổng kết, thống kê, kết xuất thông tin, In các giấy tờ giao dịch với số lượng lớn, các xử lý có tính chất định kỳ.

- **Xử lý trực tuyến:** Thông tin đến là xử lý ngay. Thường dùng vào cập nhật và xử lý 1 số lượng nhỏ các giao dịch, hiển thị, chỉnh đốn, sửa chữa nội dung các tệp, phục vụ trực tiếp khách hàng tại chỗ (vd: bán vé máy bay).

Đặc điểm: Máy tính phải thường trực (online), giao tiếp giữa người và máy theo phương thức đối thoại (interactive)

- ❖ **Đánh giá xử lý trực tuyến:** Ngày nay người ta có xu hướng dùng xử lý trực tuyến nhiều do máy có giá thành thấp, nhưng điều đó không hẳn đã hay.

Ưu điểm:

- Kiểm tra được sự đúng đắn của dữ liệu ngay khi mới nhập
- Giảm được công việc giấy tờ của các khâu trung gian
- Người dùng tham gia vào qui trình một cách trực tiếp do đó hiểu rõ hệ thống (hiểu rõ qui trình xử lý)
- Cho trả lời nhanh chóng

Nhược điểm:

- Chi phí đắt (cho cả phần cứng lẫn phần mềm)
 - Xây dựng tốn công và tốn thời gian
 - Sử dụng CPU lãng phí, không kinh tế
 - Xử lý chậm khi khối lượng thông tin lớn
 - Khó phục hồi dữ liệu khi có sự cố
 - Thao tác máy phức tạp
 - Khó bảo đảm tính tin cậy
 - Khó phục hồi sự kiện khi có sự cố
 - Khó bảo mật
- **Xử lý thời gian thực (Real-time processing):** Là hành vi của một hệ thống phải thoả mãn một số điều kiện ràng buộc rất ngặt nghèo về thời gian, chẳng hạn phải chịu hạn định đối với thời gian trả lời. Thông thường thì thời gian ở đây máy tính lệ thuộc vào một hệ thống ngoài (chẳng hạn một tên lửa, một lò nung,...), hệ thống này hoạt động theo một tiến độ riêng của nó và máy tính, với mục đích điều khiển sự hoạt động của hệ thống ngoài này, cần phải phản ứng một cách kịp thời đối với mọi biến động trạng thái của nó.

- **Xử lý phân tán (Distributed processing):** Việc xử lý thực hiện trên mạng mà các nút là các đầu cuối nặng (nghĩa là các máy tính thực sự). Thông tin xuất hiện, có thể được xử lý một phần ở đầu cuối, rồi được chuyển đi xử lý tiếp ở đầu cuối khác (hoặc ở bộ xử lý trung tâm). Ở đây, các cơ sở dữ liệu cũng có thể rải rác ở các nút của mạng.

6. Các giai đoạn phân tích và thiết kế một hệ thống thông tin

Đây là một quá trình nhận thức và mô tả đối với một hệ thống tin, các công việc cần hoàn thành là:

- o Xác định các vấn đề và yêu cầu của hệ thống
- o Xác định được mục tiêu và các vấn đề ưu tiên trong hệ thống
- o Thiết kế logic (trả lời câu hỏi What?)
- o Thiết kế vật lý (trả lời câu hỏi How? Đưa những biện pháp, phương tiện)
- o Cài đặt (lập trình)
- o Tiến hành khai thác và bảo trì

Để hoàn thành các công việc trên người ta chia thành các giai đoạn, việc phân chia các giai đoạn trong quá trình phát triển hệ thống là một đặc trưng của mỗi phương pháp. Trong phương pháp phân tích có cấu trúc người ta chia quá trình này thành 6 giai đoạn sau:

❖ **Giai đoạn 1: Khảo sát hiện trạng và xác lập dự án**

- Tìm hiểu và đánh giá hiện trạng
- Xác định mục tiêu, phạm vi và khả năng của dự án
- Phác họa giải pháp và cân nhắc tính khả thi
- Lập dự trù và kế hoạch

❖ **Giai đoạn 2: Phân tích hệ thống**

Phân tích sâu hơn các chức năng, các dữ liệu của hệ thống cũ. Phải mô tả được hệ thống cũ và hệ thống mới mà ta định thiết kế

Trong giai đoạn này cần sử dụng một số công cụ sau:

- Biểu đồ phân cấp chức năng (BPC): Để xây dựng các chức năng cơ bản nhất của hệ thống
- Biểu đồ luồng dữ liệu (BLD): Để xây dựng các biểu đồ luồng dữ liệu cho hệ thống
- Biểu đồ cấu trúc dữ liệu (BCD): Để xây dựng lược đồ dữ liệu cho hệ thống mới

❖ **Giai đoạn 3: Thiết kế tổng thể**

- Thiết kế hệ thống mới ở mức logic
- Phân chia hệ thống thành 2 phần là hệ thống máy tính và hệ thống thủ công (xác định vai trò của máy tính)
- Phân chia hệ thống máy tính thành các hệ thống máy tính con (xác định các MODUL)

❖ **Giai đoạn 4: Thiết kế chi tiết**

- Thiết kế các thủ tục thủ công và giao diện người máy
- Thiết kế các kiểm soát và phục hồi , bảo mật
- Thiết kế CSDL
- Thiết kế chương trình và các mẫu thử

❖ **Giai đoạn 5: Lập trình và cài đặt (chiếm một phần rất nhỏ)**

- Lập trình
- Chạy thử và cài đặt
- Lập các tài liệu hướng dẫn và huấn luyện

❖ **Giai đoạn 6: Khai thác và bảo trì**

* Theo thống kê tính về chi phí cho các giai đoạn này như sau:

- Phân tích thiết kế : Chiếm 17%
- Chương trình : Chiếm 8%
- Test : Chiếm 25%
- Khai thác và bảo trì : Chiếm 50%

Câu hỏi cuối chương

1. Tại sao khi xây dựng các phần mềm lại cần phải phân tích và thiết kế hệ thống?
2. Nêu vai trò hệ thống thông tin trong hệ thống kinh doanh/dịch vụ
3. Nêu các giai đoạn của quá trình phân tích và thiết kế hệ thống
4. Những lĩnh vực ứng dụng nào phù hợp với phương thức xử lý thông tin:
 - Theo mẻ
 - Trực tuyến

CHƯƠNG 3: CÁC CÔNG CỤ DIỄN TẢ XỬ LÝ

Mở đầu: Phân tích thiết kế nói chung là nhận thức và mô tả một hệ thống, bởi vậy người ta thường dùng các mô hình, các biểu đồ để trừu tượng hoá và giúp con người trao đổi lẫn nhau. Mỗi mô hình là một khuôn dạng để nhận thức (có ý thức chủ quan).

Mục tiêu của phân tích mô hình xử lý là đưa ra một cách xác định các yêu cầu của người dùng trong quá trình phát triển hệ thống; những yêu cầu này được bám sát từ một loạt các sự kiện mà người phân tích thu được qua phỏng vấn đặt câu hỏi, đọc tài liệu và qua các phép đo thử nghiệm.

Có một số công cụ chính để diễn tả chức năng của hệ thống:

- Biểu đồ phân cấp chức năng (BPC)

Functional Hierarchical Decomposition Diagram (FHD)

- Biểu đồ luồng dữ liệu (BLD)

Data Flow Diagram (DFD)

- Các kí hiệu mở rộng của hãng IBM

- Sơ đồ thuật toán (Algorithm)

- Ngôn ngữ giả trình (Pseudo Code)

- Đặc tả các qui tắc quản lý

- Từ điển định nghĩa chức năng xử lý

I/ Biểu đồ phân cấp chức năng (BPC):

Biểu đồ phân cấp chức năng (BPC) là công cụ khởi đầu để mô tả hệ thống qua chức năng do công ty IBM đề xuất từ giữa thế kỷ trước cho đến nay nó vẫn được sử dụng rộng rãi.

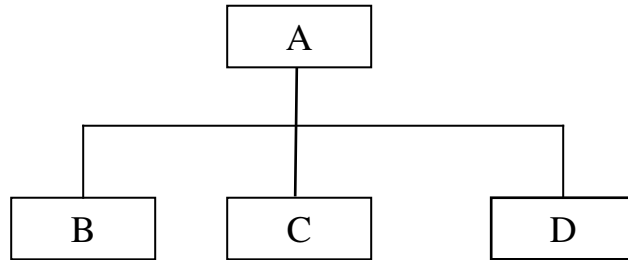
BPC cho phép phân rã dần các chức năng từ các chức năng mức cao thành các chức năng chi tiết nhỏ hơn, và kết quả cuối cùng ta thu được biểu đồ dạng cây chức năng, nó xác định một cách rõ ràng, dễ hiểu cái gì xảy ra trong hệ thống. .

Thành phần của biểu đồ bao gồm:

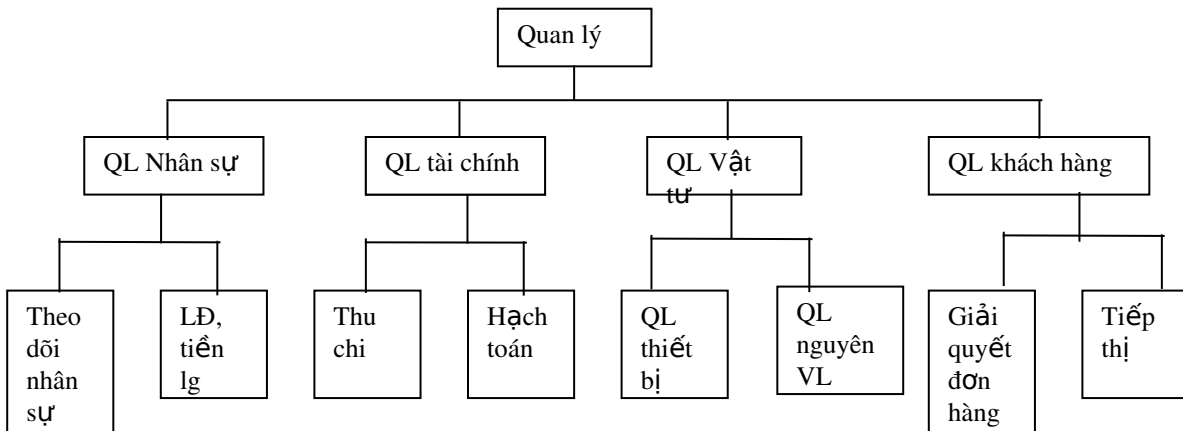
- ❖ **Các chức năng:** Được ký hiệu bằng hình chữ nhật, trong có ghi tên chức năng, tên chức năng được đặt theo cách: Động từ và có thêm bổ ngữ nếu cần

Tên

- ❖ **Các kết nối:** Kết nối giữa các chức năng có tính chất phân cấp được ký hiệu bằng các đoạn thẳng nối chức năng “cha” với các chức năng “con”



Ví dụ: Sơ đồ BPC của hệ thống quản lý xí nghiệp



Hình 3.1: Sơ đồ BPC

Đặc điểm của BPC:

- Cung cấp một cách nhìn khái về toàn bộ hệ thống, trong đó các chức năng được nhìn một cách khái quát nhất, trực quan dễ hiểu, thể hiện tính cấu trúc của phân rã chức năng (Functionally Decomposed)
- Dễ thành lập vì tính đơn giản: Nó trình bày hệ thống phải làm gì? hơn là hệ thống làm như thế nào?
- Mang tính chất tĩnh vì bỏ qua mối liên quan thông tin giữa các chức năng trong hệ thống. Các chức năng không bị lặp lại và không dư thừa
- Rất gần gũi với sơ đồ tổ chức nhưng ta không đồng nhất nó với sơ đồ tổ chức (phần lớn các tổ chức của doanh nghiệp nói chung thường gần liền với chức năng)

II/ Biểu đồ luồng dữ liệu (BLD):

1. **Mục đích:** Diễn tả tập hợp các chức năng xử lý thông tin của hệ thống trong các mối quan hệ sau:

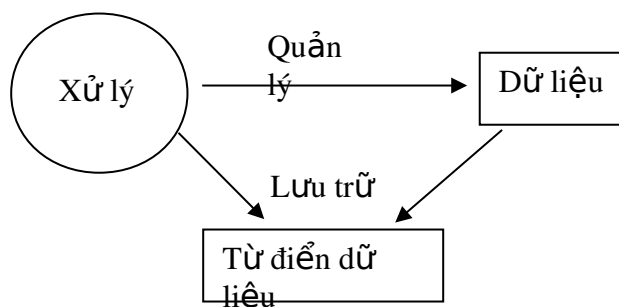
- Trước, sau và trong tiến trình xử lý
- Bàn giao thông tin cho nhau (Kế thừa thông tin của nhau)

Biểu đồ giúp chúng ta thấy được đằng sau những cái gì thực tế xảy ra trong hệ thống (cái bản chất), làm rõ những chức năng và thông tin nào cần thiết cho quản lý.

Biểu đồ này dựa vào phương pháp phát triển hệ thống có cấu trúc bao gồm 3 kỹ thuật phân tích chính:

- Sơ đồ dòng dữ liệu (BLD) mô tả quan hệ giữa quá trình xử lý và các dòng dữ liệu
- Từ điển định nghĩa dữ liệu (Data Dictionary Definitions) mô tả các phần tử dòng dữ liệu
- Xác định quá trình xử lý (Process Specifications) mô tả quá trình xử lý một cách chi tiết

Mối quan hệ giữa 3 thành phần là bức tranh sinh động của hệ thống được thể hiện qua sơ đồ sau:



2. **Tác dụng:** BLD là công cụ chính của quá trình phân tích, nhằm mục đích thiết kế trao đổi và tạo lập dữ liệu. Nó thể hiện rõ ràng và khá đầy đủ các nét đặc trưng của hệ thống trong các giai đoạn: Phân tích, thiết kế, trao đổi tư liệu.

3. **Các yếu tố hợp thành** (các thành phần):

Hình thức biểu diễn: Trong một số tài liệu khác nhau với các phương pháp tiếp cận khác nhau (MEN, SSADM) người ta thường dùng những kí hiệu không hoàn toàn giống nhau. Tuy vậy các thành phần cơ bản không thay đổi và nó được sử dụng nhất quán trong các quá trình phân tích, thiết kế.

Các thành phần của biểu đồ gồm 5 thành phần:

- Chức năng xử lý (Process)
- Luồng thông tin (Data Flows)

- o Kho dữ liệu (Data Store)
- o Tác nhân ngoài (External Entity)
- o Tác nhân trong

a) **Chức năng xử lý:**

Là chức năng biểu đạt các thao tác, nhiệm vụ hay tiến trình xử lý nào đó. Tính chất quan trọng của chức năng là *biến đổi* thông tin. Tức là nó phải làm thay đổi thông tin từ đầu vào theo một cách nào đó như tổ chức lại thông tin hoặc tạo ra thông tin mới.

Biểu diễn chức năng người ta thường sử dụng :

- Bằng 1 đường tròn bên trong có ghi tên chức năng
- Tên chức năng: Động từ + Bỏ ngữ

Ví dụ:



Lưu ý: Trong thực tế tên của các chức năng trong biểu đồ BLD phải trùng với tên của các chức năng đã đặt trong biểu đồ BPC.

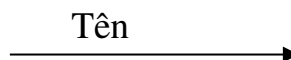
b) **Luồng dữ liệu:**

Chính là luồng thông tin vào hoặc ra của 1 chức năng xử lý. Bởi vậy luồng dữ liệu được coi như các giao diện giữa các thành phần của biểu đồ.

Khi nói đến luồng thông tin ta nên hiểu ở đây có một thông tin được chuyển đến một chức năng để xử lý hoặc chuyển ra khỏi một chức năng như một kết quả xử lý, mà không cần quan tâm đến hình thức truyền dẫn (bằng tay, qua máy Fax hay qua máy điện thoại,...)

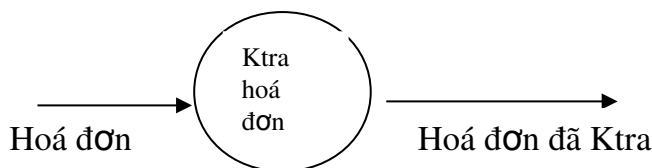
Biểu diễn: Một luồng dữ liệu được chỉ ra trên biểu đồ

- Bằng một đường kẻ có mũi tên (có một hoặc hai mũi tên ở hai đầu), chỉ hướng lan truyền của thông tin và trên đó có ghi tên nhãn là tên luồng thông tin mang theo



- Tên của luồng thông tin: Là danh từ + tính từ nếu cần

Ví dụ:



c) Kho dữ liệu:

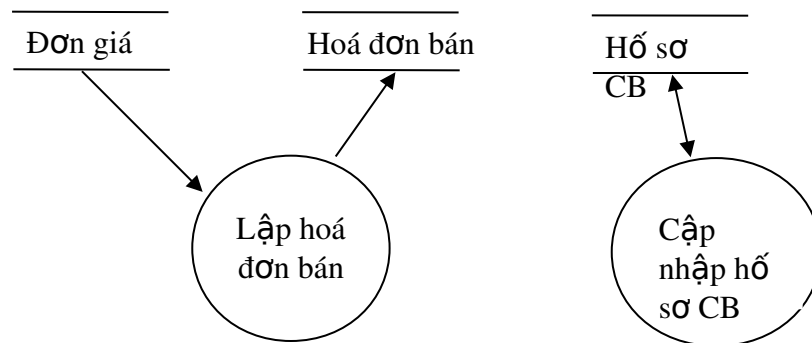
Là thông tin cần lưu giữ lại trong 1 khoảng thời gian để sau đó một hay nhiều chức năng xử lý sẽ sử dụng

Kho dữ liệu bao gồm một nghĩa rất rộng các dữ liệu lưu trữ: Dưới dạng vật lý chúng có thể là các tài liệu, hồ sơ trong văn phòng hoặc các file trên các thiết bị mang tin (đĩa từ, đĩa quang,...) của máy tính. Các phương tiện vật lý không phải là điều ta quan tâm, điều chủ yếu là thông tin chứa trong chúng, tức là dạng logic của chúng .

Biểu diễn:

- *Kho dữ liệu được biểu diễn bằng hình chữ nhật hở hai đầu hay cặp đoạn thẳng song song, trên đó có ghi tên của kho*
- *Tên của kho dữ liệu: Phải là một danh từ + tính từ nếu cần*

Ví dụ:



Ví dụ trên:

Chức năng *Lập hoá đơn bán hàng* cần tham khảo thông tin từ kho *Đơn giá* (mũi tên từ kho *Đơn giá* đi ra).

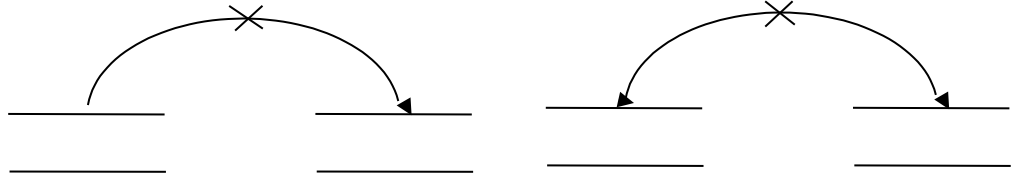
Sau khi lập xong hoá đơn cần lưu trữ lại những hoá đơn đã lập thì mũi tên có hướng đi vào kho (*hoá đơn bán*)

Chức năng *Cập nhập hồ sơ CB* có nhiệm vụ vừa ghi thông tin vào kho và vừa lấy thông tin ra để kiểm tra, chỉnh sửa, thì khi đó mũi tên có hai chiều (vào/ra từ kho)

Lưu ý:

- *Luồng dữ liệu vào hoặc ra không cần mang tên vì tên luồng trùng với tên kho. Ngoại trừ trường hợp trích từ kho 1 phần thông tin, khi đó phải có tên cho luồng*
- *Kho dữ liệu phải có ít nhất một luồng vào và ít nhất một luồng ra. Nếu kho chỉ có luồng vào mà không có luồng ra là kho “vô tích sự”, nếu kho chỉ có luồng ra mà không có luồng vào là kho “rỗng”*

- Kho dữ liệu chỉ là nơi lưu trữ thông tin, không phải là nơi xử lý thông tin. Do đó không bao giờ có trao đổi thông tin giữa hai kho



- Để tiện cho việc trình bày cùng một kho dữ liệu có thể xuất hiện ở nhiều nơi trong biểu đồ

d) Tác nhân ngoài: (đối tác)

Là một người, nhóm người hay một tổ chức ngoài hệ thống nhưng có trao đổi thông tin với hệ thống

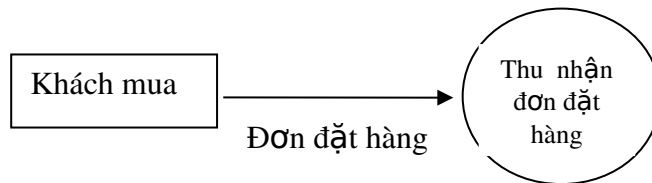
Sự có mặt các tác nhân ngoài trên sơ đồ chỉ ra giới hạn của hệ thống, và định rõ mối quan hệ của hệ thống với thế giới bên ngoài. Nhưng điều quan trọng cần hiểu là “ngoài hệ thống” không nhất thiết có nghĩa là bên ngoài tổ chức, chẳng hạn như đối với hệ thống *Xử lý đơn hàng* thì *bộ phận kế toán*, *bộ phận mua hàng*, và *bộ phận kho hàng* vẫn là tác nhân ngoài. Hay đối với hệ thống *Tuyển sinh đại học* thì tác nhân ngoài vẫn có thể là *thí sinh*, *giáo viên chấm thi* và *hội đồng tuyển sinh*.

Tác nhân ngoài là phần sống còn của hệ thống. Chúng là nguồn cung cấp thông tin cho hệ thống cũng như chúng nhận các sản phẩm thông tin từ hệ thống.

Biểu diễn:

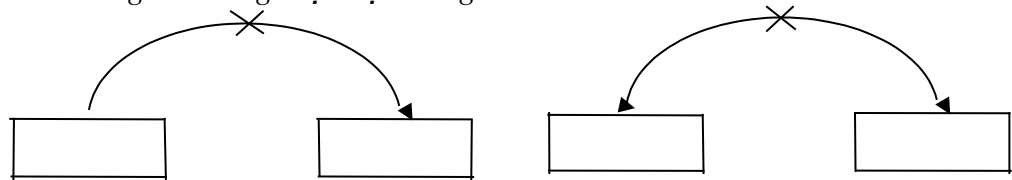
- Bằng một hình chữ nhật trong có tên của tác nhân ngoài
- Tên của tác nhân ngoài: Danh từ + Tính từ nếu cần

Ví dụ:



Lưu ý:

- Không được xảy ra trường hợp có sự trao đổi thông tin trực tiếp giữa hai tác nhân ngoài trong một hệ thống



- Để tiện cho việc trình bày cùng một tác nhân ngoài có thể xuất hiện ở nhiều nơi trong biểu đồ

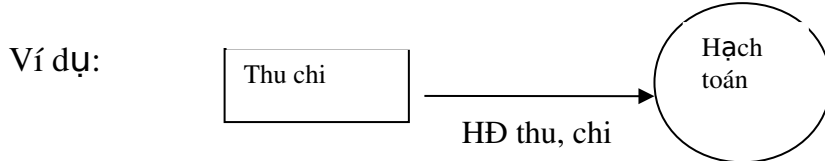
e) Tác nhân trong:

Là một chức năng hay một hệ thống con của hệ thống đang xét nhưng lại được mô tả ở trang khác của biểu đồ.

Thông thường mọi biểu đồ đều có thể bao gồm một số trang, nhất là trong các hệ thống phức tạp và với khuôn khổ giấy có hạn, thông tin được truyền giữa các quá trình trên các trang khác nhau được chỉ ra nhờ kí hiệu tác nhân trong này.

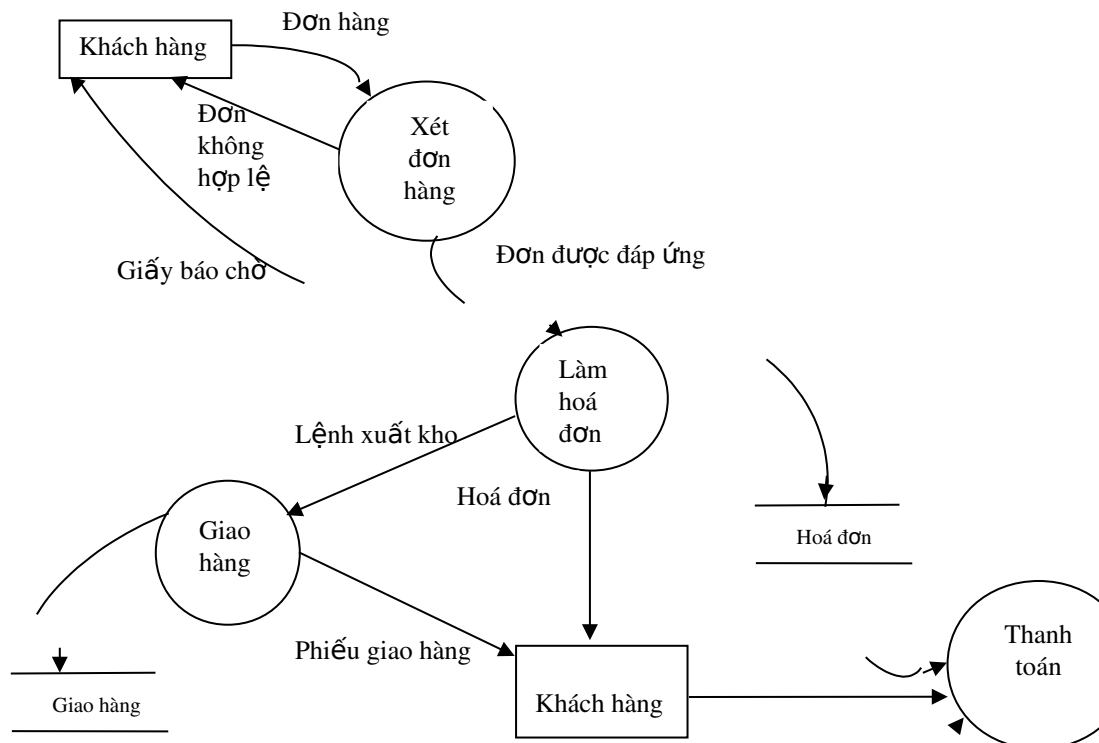
Biểu diễn:

- Bằng một hình chữ nhật khuyết một cạnh, trong có tên của tác nhân trong
- Tên của tác nhân trong : Động từ + Bổ ngữ nếu cần



Lưu ý: Để tiện cho việc trình bày cùng một tác nhân trong có thể xuất hiện ở nhiều nơi trong biểu đồ

4. Ví dụ: BLD biểu diễn hoạt động bán hàng của một công ty thương mại



Nhận xét:

Trong các BLD ở hình 2.1 ta không thấy có các thông tin điều khiển. Vậy các chức năng được khởi động và thực hiện ra sao? Ở đây có một nguyên tắc ngầm định giải quyết việc này, đó là nguyên tắc kích hoạt bằng dữ liệu (data triggered) mà nội dung là:

- Một chức năng trong BLD được khởi động khi nó hội đủ các dữ liệu cần thiết (ở đầu vào)
- Khi được khởi động, thì một chức năng được xem là được thực hiện vô cùng nhanh, nghĩa là dữ liệu ở đầu ra có ngay lập tức

Như vậy các yếu tố thời gian và điều khiển không được xét tới. Vậy nó có sự hạn chế đáng kể trong khả năng biểu diễn. Các yếu tố về điều khiển và biến đổi hành vi trong thời gian sẽ được đề cập trong một khuôn khổ giáo trình khác.

5. Các thể hiện khác của BLD:

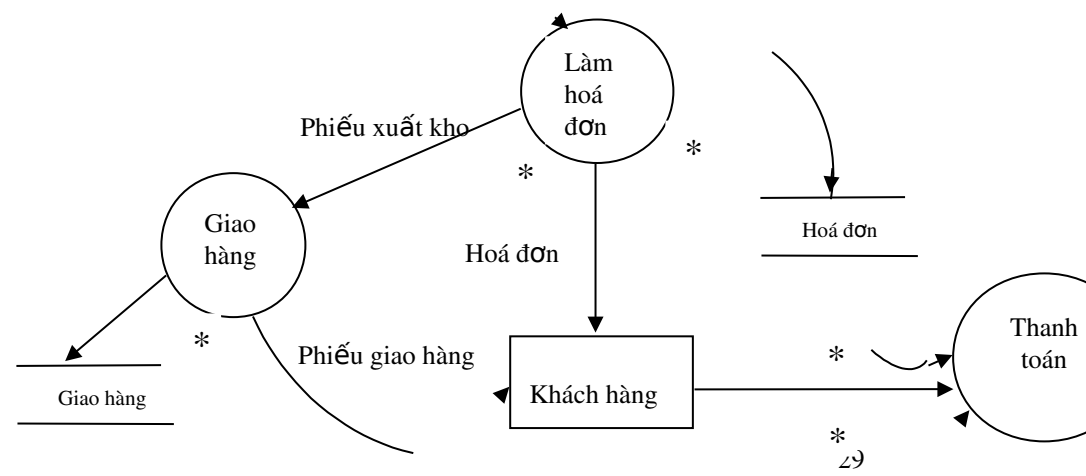
Các biểu đồ BPC, BLD trên đôi khi chưa sáng tỏ với thực tế vì thực chất mô hình còn giản lược. Bởi vậy khi cần thiết người ta đưa ra một số khái niệm để trừu tượng hoá, lấy được bản chất của vấn đề.

a) Sự đồng bộ hoá:

Bổ sung thêm một số ký hiệu bên cạnh luồng dữ liệu

Ký hiệu: * để biểu thị liên từ and (và) Đơn hàng đồng xảy ra cùng nhau
 + để biểu thị liên từ or (hoặc) Đơn hàng đồng xảy ra cùng nhau
 ⊕ để biểu thị hoặc loại trừ (XOR): Không sử dụng cùng nhau

Ví dụ: BLD biểu diễn hoạt động bán hàng của công ty thương mại được gắn thêm điều kiện đồng bộ hoá như sau:
 Khách hàng (đơn hàng) → Xét đơn hàng (đơn hàng) → Giấy báo chờ (đơn hàng) → Đơn được đáp ứng (đơn hàng)



Hình 3.3

b) Phương pháp MERISE, ở đây mỗi chức năng (biểu diễn bằng hình chữ nhật) có gắn thêm một điều kiện đồng bộ hóa ở đầu vào, chức năng chỉ thực hiện khi điều kiện này đúng (qui ước: \wedge là và; \vee là hoặc, \neg là không)

V: Chỉ luồng dữ liệu vào

r: chỉ luồng dữ liệu ra

Ví dụ: BLD biểu diễn hoạt động bán hàng của một công ty thương mại được gắn thêm điều kiện đồng bộ hoá theo phương pháp MERISE (**bổ xung sau**)

c) Phân bố các chức năng theo địa chỉ và thời gian: Các yếu tố về điều khiển và biến đổi hành vi trong thời gian sẽ được đề cập trong một khuôn khổ giáo trình khác

d) Đưa thêm các ký hiệu vật lý vào biểu đồ (**bổ xung sau**)

6. **Bài tập:** Vẽ biểu đồ BPC và BLD cho hệ thống tuyển sinh sau:

+ Thí sinh: Gửi đến hệ thống hồ sơ dự thi, bộ phận tiếp nhận hồ sơ nhận hồ sơ của thí sinh xử lý và giao cho thí sinh số báo danh

+ Thí sinh gửi bài thi cho bộ phận nhận bài thi, bộ phận này có nhiệm vụ:

- . Đánh phách bài thi
- . Giao bài thi đã đọc phách cho giáo viên chấm
- . Lưu cất mỗi liên hệ giữa phách và số báo danh

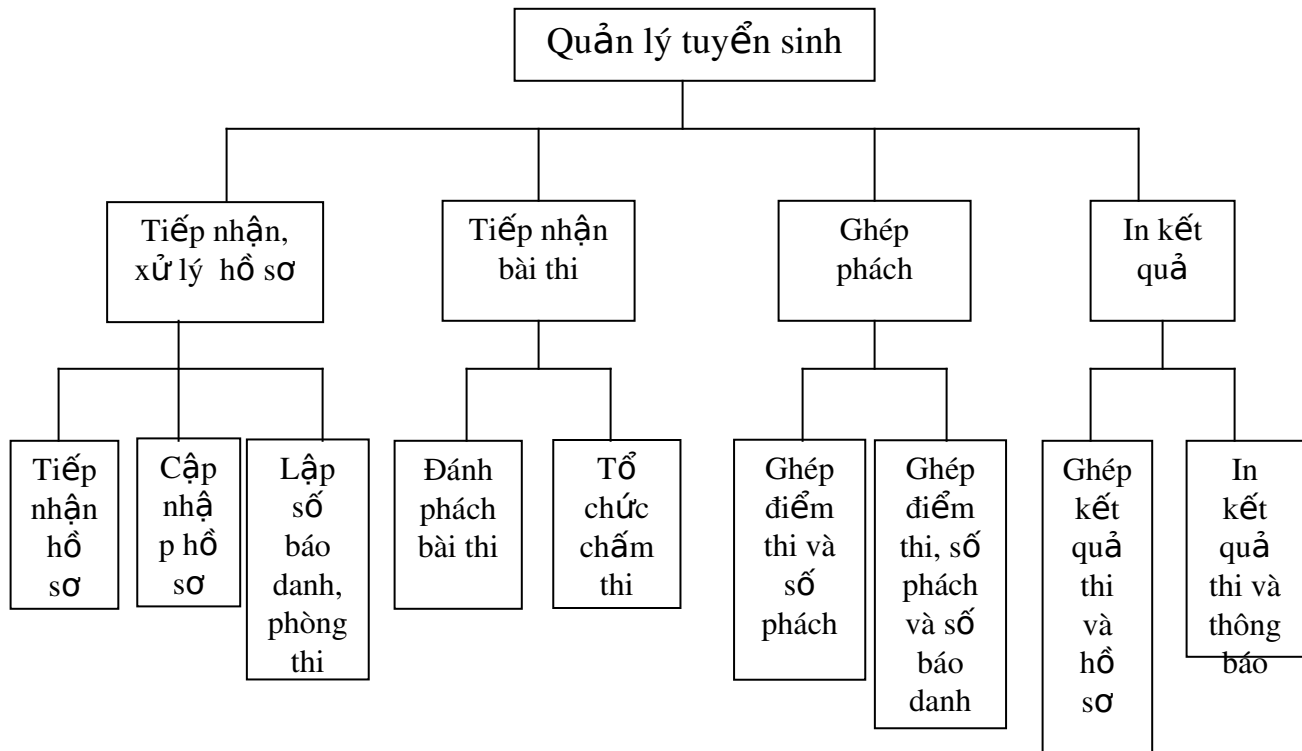
+ Thầy giáo trả bài thi đã chấm điểm cho bộ phận ghép phách, bộ phận này làm các công việc sau:

. Xem xét mối liên hệ giữa SBD và số phách và mối liên hệ giữa phách và điểm để đưa ra liên hệ giữa SBD và điểm

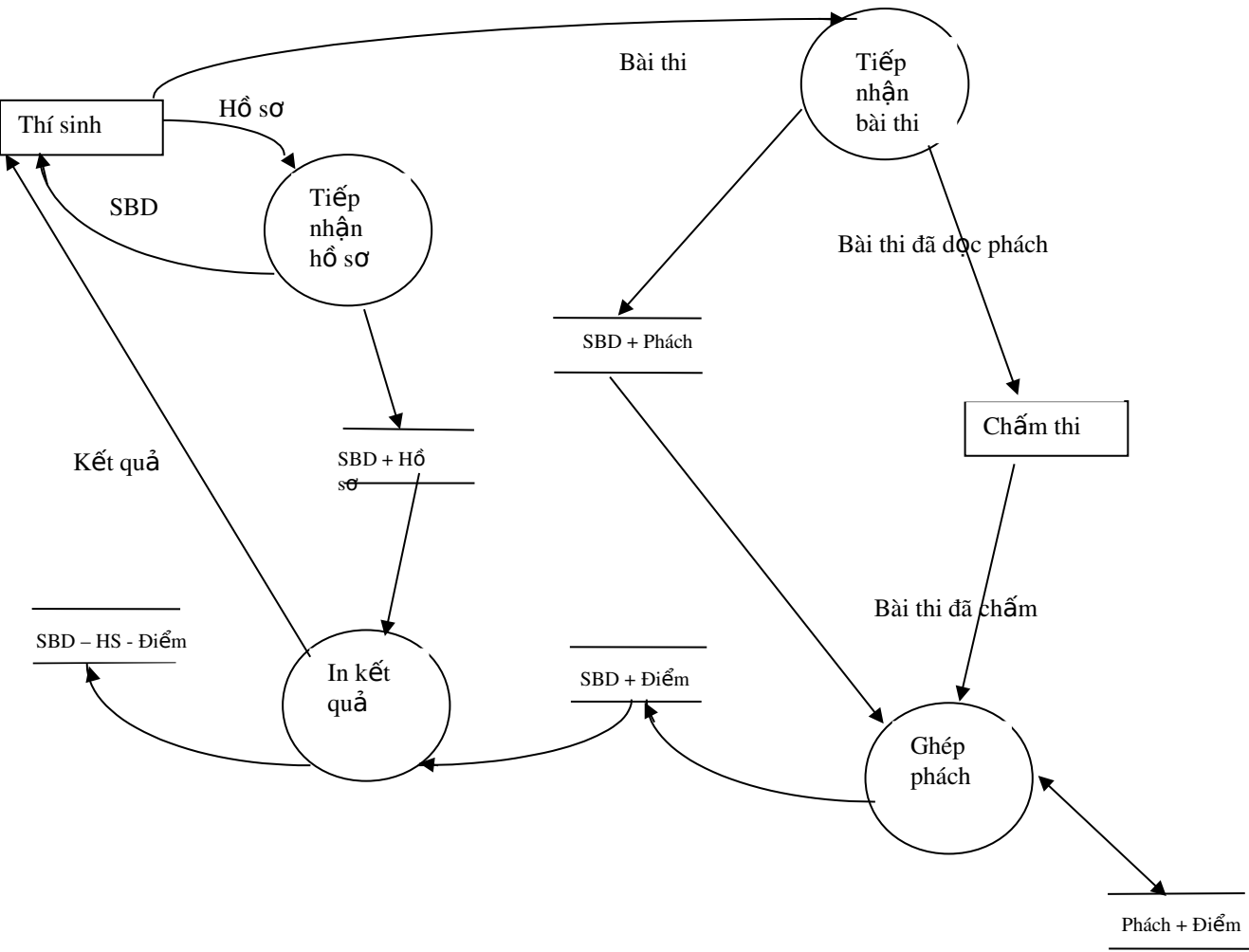
. Gửi kết quả cho bộ phận in kết quả

+ Bộ phận in kết quả sẽ trả kết quả cho thí sinh và lưu cất một bản sao

Biểu đồ Phân cấp chức năng (BPC):



Biểu đồ luồng dữ liệu (BLD):



7. Các phương tiện diễn tả chức năng:

a. Đặc tả chức năng:

Trong biểu đồ BLD, để diễn tả một chức năng phức tạp ta phân rã nó thành nhiều chức năng đơn giản hơn, quá trình phân rã dần từng bước sẽ ngừng ở một mức nào đó vì có phân tích sâu thêm sẽ vượt qua câu hỏi “Hệ thống làm gì?” để lấn sang giai đoạn thiết kế trả lời câu hỏi “Hệ thống làm việc như thế nào?” hoặc là chức năng thu được ở mức cuối cùng đã rất đơn giản cũng cần được giải thích (nếu không nó vẫn cứ là “hộp đen”). Bây giờ sự giải thích chức năng phải được thực hiện bởi những phương tiện diễn tả trực tiếp (khác với BPC và BLD). Gọi đó là *sự đặc tả* chức năng, thường gọi tắt là P-Spec (Process Specification)

Một đặc tả chức năng thường được trình bày một cách ngắn gọn, không vượt quá một trang A4 và gồm hai phần:

- Phần đầu đề gồm:
 - Tên chức năng
 - Các dữ liệu vào
 - Các dữ liệu ra
- Phần thân: Mô tả nội dung xử lý, ở đó thường sử dụng các phương tiện mô tả sau đây (liệt kê theo thứ tự ưu tiên giảm dần)
 - Các phương trình toán học
 - Các bảng quyết định hay cây quyết định
 - Các sơ đồ khối
 - Các ngôn ngữ tự nhiên cấu trúc hóa
 - (không nên sử dụng ngôn ngữ tự do)

Đầu đề	
Tên chức năng:	Tính kết quả bảo vệ đồ án
Đầu vào:	Điểm của người phản biện Điểm của người hướng dẫn Số các Ủy viên hội đồng Điểm của từng Ủy viên hội đồng
Thân	
Kết quả bảo vệ = $(\text{Điểm của người phản biện} + \text{Điểm của người hướng dẫn} + \text{Tổng điểm của các Ủy viên hội đồng}) / 3$ Số các Ủy viên hội đồng	

Hình 3.4: Một sự đặc tả chức năng

Hình 2.3 cho một ví dụ đặc tả chức năng, trong đó phần thân dùng một phương trình toán học.

b. Ngôn ngữ tự nhiên cấu trúc hóa: Được sử dụng để diễn tả các giải thuật hay còn gọi là *các quy trình xử lý*

Để miêu tả giải thuật người ta thường dùng:

- Sơ đồ khối
- Ngôn ngữ diễn tả giải thuật

Ngôn ngữ diễn tả giải thuật là một ngôn ngữ tự nhiên được cấu trúc hoá, bao gồm các thành phần sau :

- Các câu đơn giản, các danh từ, các dòng sự kiện
- Các cấu trúc đơn điều khiển:
 - *Tuần tự*: Là trật tự trước sau trong một quá trình thực hiện, câu đứng trước được thực hiện trước, câu đứng sau được thực hiện sau
 - *Chọn*: nếu thì ... không thì
 - *Lặp*: Với mỗi lặp

Ví dụ : Để diễn tả giải thuật **Lập hoá đơn** ta thể hiện như sau:

LẬP HÓA ĐƠN

Với *mỗi đơn hàng* thực hiện các công việc sau (lặp) :

- (1) Viết tên khách hàng và địa chỉ vào hoá đơn
- (2) Nếu khách hàng thuộc loại *Ưu tiên* thì:
 - (2.1) Tra cứu xuất miễn giảm từ tệp “miễn giảm”, sử dụng hệ số ưu tiên
 - (2.1) Nếu không, ghi hệ số miễn giảm là 0%
- (3) Với *mỗi mặt hàng trên đơn hàng* lặp:
 - (3.1) Chép mã mặt hàng và số lượng yêu cầu vào hoá đơn
 - (3.2) Tra cứu đơn giá từ tệp đơn giá (bằng cách dùng mã mặt hàng)
 - (3.3) Lập tổng bộ phận theo công thức định sẵn
- (4) Lấy tổng các bộ phận
- (5) Lập tiền phải trả = Tổng các tổng bộ phận - Tiền ứng trước

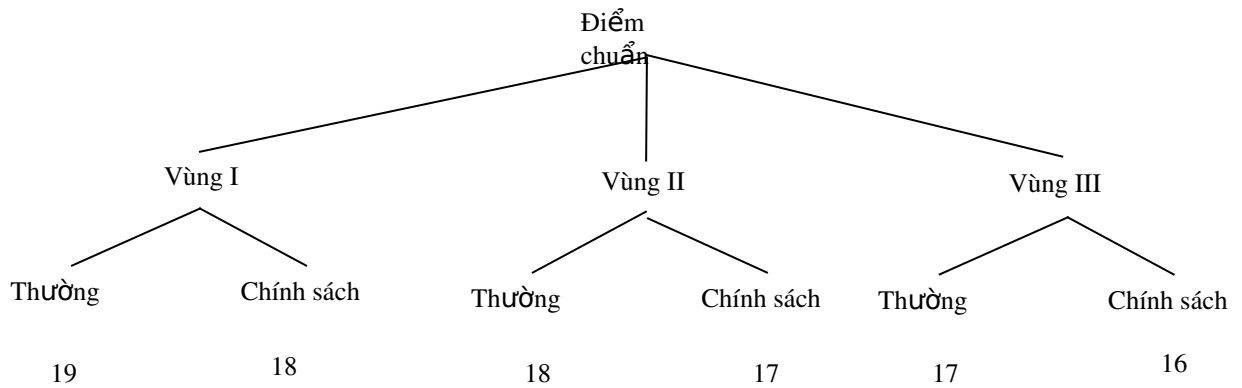
c. Các bảng quyết định hay cây quyết định: Được sử dụng để diễn tả các quy tắc quản lý.

Trong các qui trình xử lý thường có các qui tắc định sẵn, các qui tắc này là: Sự lựa chọn giữa nhiều tình huống và phụ thuộc vào nhiều biến, do đó đòi hỏi chúng ta phải tổng hợp. Hai công cụ sau đây sẽ hỗ trợ công việc tổng hợp này.

- Cây quyết định
- Bảng quyết định

❖ **Cây quyết định**

Cây quyết định chia các trường hợp nhờ cấu trúc cây. Chẳng hạn Qui tắc lập điểm chuẩn của một trường đại học như sau:



Hình 3.5: Sơ đồ cây quyết định

❖ **Bảng quyết định**

Là một bảng hai chiều, trong đó một chiều được tách làm hai phần; một phần cho các điều kiện vào và phần kia cho các hành động hay biến ra; chiều thứ hai là các trường hợp có thể xảy ra tùy vào giá trị của các điều kiện, ứng với mỗi trường hợp là một cột hay một dòng; Các hành động được lựa chọn sẽ được đánh dấu x hoặc nếu cái ra là biến thì cho các giá trị tương ứng của biến đó.

Điều kiện (các trường hợp)	→				
Hoạt động (các hành động)	→				
		Vùng			
		Diện	Vùng I	Vùng II	Vùng III
		Thường	19	18	17
		C sách	18	17	16

Hình 3.6: Sơ đồ bảng quyết định

Chú ý: Trong một bảng phức tạp đôi khi một phần tử trong bảng quyết định lại là một bảng quyết định khác

Bài tập chương 3

1. Tại sao luồng dữ liệu vào/ra từ kho dữ liệu thường không có tên ?
2. Trong biểu đồ BLD có khi nào không có tác nhân ngoài không ? Tại sao ?
3. Khi thể hiện một biểu đồ BPC, BLD chúng ta hay phạm phải những sai sót nào? Hãy giải thích ?
4. Biểu đồ BLD cho ta biết mối quan hệ gì giữa các thành phần trong hệ thống?
5. Hãy so sánh biểu đồ BPC với biểu đồ BLD

CHƯƠNG IV: CÁC MÔ HÌNH VÀ PHƯƠNG TIỆN DIỄN TẢ DỮ LIỆU

Một hệ thống trong trạng thái vận động bao gồm hai yếu tố là các chức năng xử lý và dữ liệu. Giữa xử lý và dữ liệu có mối quan hệ mật thiết, chặt chẽ, mặt khác bản thân dữ liệu có mối liên kết nội bộ không liên quan đến xử lý đó là tính độc lập dữ liệu. Mô tả dữ liệu được xem như việc xác định tên, dạng dữ liệu và tính chất của dữ liệu, nó không phụ thuộc vào người sử dụng, đồng thời không phụ thuộc vào yêu cầu tìm kiếm và thay đổi thông tin.

Trong chương này, để thuận tiện cho phương pháp nghiên cứu chúng ta chỉ tập trung đề cập đến các phương tiện và mô hình diễn tả dữ liệu. Đó là các thông tin được quan tâm đến trong quản lý, nó được lưu trữ lâu dài, được xử lý và sử dụng trong hệ thống kinh doanh/dịch vụ.

Có nhiều công cụ để mô tả dữ liệu, đó là các cách trừu tượng hoá dữ liệu đặc biệt là mối quan hệ của dữ liệu nhằm phổ biến những cái chung nhất mà con người có thể trao đổi lẫn nhau.

Ở đây chúng ta đề cập đến 4 công cụ chủ yếu là:

- ❖ Mã hoá dữ liệu (coding)
- ❖ Từ điển dữ liệu (Data Dictionary)
- ❖ Mô hình thực thể liên kết ER (Entity – Relationship)
- ❖ Mô hình quan hệ (Relational Data Base Modeling)

Trước khi đề cập đến các công cụ để mô tả dữ liệu trên chúng ta tìm hiểu đôi nét về khái niệm dữ liệu, các tệp và cơ sở dữ liệu.

I. Khái niệm dữ liệu:

1) Các cấp bậc của dữ liệu: 6 cấp bậc

Mỗi hệ thống thông tin đều có cấp bậc tổ chức dữ liệu và mỗi mức kế tiếp trong cấp bậc này đều là kết quả tổ hợp các phần tử của mức trước đó. Dữ liệu được tổ chức logic theo kiểu này cho tới khi đạt tới CSDL

- a) **Bit:** Là mức đầu tiên được xử lý một cách tự động không cần tác động của người lập trình hay người dùng cuối cùng.
- b) **Bit và ký tự (byte):** Ký tự được biểu diễn bởi một nhóm các bit tương ứng theo hệ thống mã hoá (Như ASCII; EBCDIC)
 - . Bit là đơn vị cơ bản cho bộ nhớ chính và phụ
 - . Ký tự: Là đơn vị cho nhận thức của con người, dưới dạng ghi nhớ dữ liệu thì ký tự thường chính là BYTE
- c) **Phần tử dữ liệu (FIELD – trường):** Là đơn vị logic cấp thấp nhất trong cấp bậc dữ liệu

(Ví dụ: Chữ H là một ký tự riêng biệt ít có ý nghĩa gì nếu nằm ngoài ngữ cảnh nhưng khi ký tự này được tổ hợp để tạo thành một cái tên như Hà nội, Hải phòng, Hồ chí Minh, ... thì chúng tạo nên một đơn vị logic).

Thí dụ: xét một người về các khía cạnh sau: Số chứng minh thư, họ tên, địa chỉ, quê quán,..... Thì tất cả được gọi là phần tử dữ liệu (trường)

Mỗi phần tử dữ liệu được ghi trong bộ nhớ với một số lượng ký tự xác định. Khi 1 phần tử dữ liệu được truy nhập đến thì nội dung xác định của nó được gọi là khoản mục dữ liệu

- d) **Bản ghi (RECORD):** Các phần tử dữ liệu có quan hệ với nhau trong việc mô tả một sự kiện (số bán hàng, việc đặt chỗ khách sạn) hay sự vật (một con người) được gộp nhóm một cách logic để tạo thành bản ghi . Bản ghi là đơn vị logic nhỏ nhất mà ta có thể thâm nhập được từ tệp (Vì muốn biết một thuộc tính nào đó thì phải tìm kiếm tới bản ghi chứa thuộc tính đó)
- e) **Tệp (FILE):** Là tập hợp các bản ghi có quan hệ với nhau (tệp nhân sự chứa các bản ghi cho từng nhân viên, tệp hàng hoá chứa các bản ghi của từng mặt hàng,...
- f) **CSDL:** Về bản chất CSDL là một tập hợp các tệp có quan hệ logic lẫn nhau. Là nguồn dữ liệu cho mọi hệ thông tin dựa trên máy tính.

Trong CSDL, dữ liệu được móc nối với nhau và có quan hệ với nhau sao cho giảm thiểu sự dư thừa (Ví dụ ta xây dựng 1 tệp phòng ban trong đó có thuộc tính mã phòng ban và tên phòng ban các tệp khác khi cần đến thuộc tính phòng ban thì dùng mã phòng ban chứ không dùng tên phòng ban, khi cần đổi tên phòng ban thì chỉ cần đổi tên phòng ban trong tệp phòng ban. Như vậy thì trong CSDL thì dữ liệu về phòng ban của một mã phòng ban nào đấy chỉ được ghi nhớ một lần và chỉ cần một lần cập nhật)

* Ngày nay các công ty lớn (các hệ thống thông tin lớn) có xu hướng xây dựng các hệ CSDL thống nhất phục vụ chung cho toàn bộ hệ thống. Việc phát triển phần mềm được tốt hơn và hiệu quả hơn do các dữ liệu đã có sẵn và được cung cấp đầy đủ. Dữ liệu trong các CSDL là độc lập trong các phần mềm nên có thể thay đổi nội dung (thêm, bớt, sửa đổi) dữ liệu mà không ảnh hưởng đến chương trình.

2) Các tệp:

a) Cấu trúc tệp:

- Tệp là tập hợp các bản ghi có quan hệ với nhau và mỗi bản ghi ứng với một đối tượng
- Đơn vị nhỏ nhất của bản ghi là trường, mỗi trường ghi 1 tính chất (1Thuộc tính của đối tượng)
- Trường khoá: Giá trị của nó dùng để phân biệt bản ghi này với bản ghi khác

b) Các thao tác trên tệp:

- ❖ Mức toàn tệp:
 - Tạo lập tệp
 - Loại bỏ tệp
 - Sắp xếp tệp
 - Hoà nhập/ phân rã, liên kết tệp
- ❖ Mức bản ghi:
 - Tra cứu bản ghi
 - Bổ sung/Loại bỏ
 - Cập nhật

c) Các đặc trưng “ phi vật lý “ của tệp:

❖ Phân loại tệp theo phương thức sử dụng tệp:

- *Tệp thường trú*: Là những tệp được lưu trữ thường xuyên và lâu dài trong hệ thống
- *Tệp vận động*: Chứa những thông tin dùng để tạo lập hay cập nhật tệp thường trú, thông thường tệp này được xây dựng từ bàn phím và được cất giữ thành một tệp(ví dụ?)
- *Tệp liên kết* (trung gian): Chứa thông tin cần chuyển qua một tệp khác(ví dụ?)
- *Tệp tạm thời*: Dùng để lưu cất kết quả trung gian trong quá trình làm việc
- *Tệp bảng*: Chứa các bảng giá trị để sử dụng trong chương trình

❖ Phân loại theo tần suất hoạt động:

- *Khái niệm tần suất*: Là khái niệm thước đo dùng để đo các mức độ làm việc của tệp, nghĩa là các số lần thao tác trên tệp trong một thời gian nào đó(đối với các thao tác trên bản ghi như: tra cứu, cập nhật, bổ sung, loại bỏ)
- *Định nghĩa*: Tần suất hoạt động là tỷ số giữa số thao tác thực hiện hành động đó trong một thời hạn (kỳ) cho trước trên tổng số các bản ghi lấy trung bình trong thời hạn đó

$$\text{Tần suất hoạt động} = \frac{\text{Số thao tác trong 1 thời kỳ cho trước}}{\text{Tổng số bản ghi lấy tb trong thời hạn đó}}$$

Chú ý: . Tần suất đo bằng %

- Tần suất có thể có giá trị >100%
- Tần suất tăng tiến có thể <0%

(Tần suất tăng tiến= Tần suất bổ sung - Tần suất loại bỏ. Nó phản ánh sự lớn lên hay nhỏ đi của tệp)

- Tần suất quay vòng= Tần suất bổ sung + tần suất loại bỏ

Ví dụ: Tập có 1200 bản ghi hàng ngày có 90 lần cập nhật

$$\text{Tần suất cập nhật/ngày} = 90/1200 = 7,5\%$$

$$\text{Tần suất cập nhật/tháng} = 90 * 30 / 1200 = 225\%$$

- ❖ **Số bản sao (phiên bản):** Để tránh mất mát người ta cần cất những tệp dưới những bản sao và phải cần theo dõi
 - Ngày tạo bản sao
 - Tần số cập nhật
 - Thời gian (khoảng cách thời gian giữa các đợt sao lưu)

- ❖ **Đặt tên tệp:** Nên đưa ra một qui luật đặt tên

Ví dụ: Chữ cái thứ nhất dùng để chỉ loại tệp

Chữ cái thứ hai dùng để chỉ giá mang tệp

4 ký tự tiếp theo là tên gọi diễn tả nội dung tệp

2 ký tự cuối cùng nói về số bản sao

(tệp thường trú là T; tệp vận động là V;...; dùng chữ F để chỉ đĩa mềm; dùng chữ H để chỉ đĩa cứng; dùng chữ B để chỉ băng từ)

d) Các đặc trưng vật lý

- Loại dùng: Băng từ, đĩa từ hay đĩa quang,..
- Chiều dài bản ghi: - Logic: độ dài thực sự của bản ghi
- Vật lý: Độ dài trên giá mang
- Thể tích tệp : Không gian bị chiếm trên giá mang

e) Tổ chức tệp và phương thức truy nhập tệp: Ta xét 4 trường hợp sau

(1) Tổ chức tuần tự và truy nhập tuần tự

(2) Tổ chức trực tiếp và truy nhập trực tiếp

(3) Tổ chức tuần tự có chỉ dẫn và truy nhập tuần tự hay trực tiếp

(4) Tổ chức móc nối và truy nhập theo móc nối

- ❖ Trường hợp (1): Về nguyên tắc thì khi tạo lập tệp, các bản ghi được đặt một cách kế tiếp nhau, và khi sử dụng thì phải truy nhập tuần tự qua các bản ghi trước

- Giá mang: Bìa đục lỗ, trống từ
- Các thao tác trên tệp này:
 - Tra cứu tuần tự
 - Bổ sung cuối tệp

- Loại bỏ logic
 - Cập nhật: Đọc viết riêng rẽ (còn trên đĩa thì có thể đọc viết tiến hành đồng thời)
- ❖ Trường hợp (2): Địa chỉ được tính toán theo khoá nhờ hàm trực truy
- Giá mang: Đĩa từ, trống từ
 - Thao tác:
 - Tra cứu trực tiếp
 - Bổ sung
 - Ưu điểm:
 - Tần suất cập nhật bé (dưới 20%)
 - Tần suất bổ sung lớn
 - Hàm trực truy thiết kế tốt thì nó có thể nhanh gấp từ 2-10 lần so với phương pháp tuần tự có chỉ dẫn
- ❖ Trường hợp (3):
- Nguyên tắc tổ chức:
 - Bản ghi xếp liên tiếp
 - Mỗi bản ghi có 1 khoá cho phép trực truy theo khoá

Mỗi khi có 1 khoá cho phép trực truy thì sẽ có các tệp chỉ dẫn để tạo ra các đường dẫn vào các bản ghi trên đĩa. Như vậy một tệp có thể được lưu trên nhiều bộ đĩa

Cụ thể, khi 1 tệp mà được chứa trên nhiều đĩa thì:

Bảng chỉ dẫn chủ: xác định đĩa

Bảng chỉ dẫn rãnh: xác định rãnh

Bảng chỉ dẫn bản ghi: Thường là tìm theo tuần tự

- Giá mang: Đĩa từ
 - Các thao tác:
 - Tra cứu tuần tự hoặc trực tiếp
 - Loại bỏ logic
 - Bổ sung: Dùng phương pháp móc nối
 - Ứng dụng (xem thêm)
- ❖ Trường hợp (4): Dùng con trỏ

- Nguyên tắc: móc nối, mỗi bản ghi có 1 con trỏ chỉ tới bản ghi tiếp theo. Phương pháp này do người lập trình tự quản lý lấy và các bản ghi có thể đặt bất kỳ chỗ nào trên giá mang
- Thao tác:
 - Tra cứu tuần tự
 - Bổ sung
 - Cập nhật
 - Loại bỏ: vật lý
- Ứng dụng: Quản lý vùng nhớ tự do cách chặt chẽ
Có ích khi: Nút cửa đồ thị

II. Sự mã hoá:

1. Khái niệm mã hoá

Mã là tên vẫn tất gán cho một đối tượng nào đó hay nói cách khác mỗi đối tượng cần có tên và vấn đề đặt ra là ta sẽ đặt tên cho đối tượng như thế nào. Trong mỗi đối tượng gồm nhiều thuộc tính khác nhau thì yêu cầu mã hoá cho các thuộc tính cũng là yêu cầu cần thiết. Ngoài ra mã còn là hình thức chuẩn hoá dữ liệu và bảo mật dữ liệu đặc biệt trong các hệ thống thông tin xử lý bằng máy tính.

Ví dụ:

Khi ta cần xác định một công dân thì *số chứng minh thư* hoặc *số hộ chiếu* là mã của công dân đó

Khi ta cần xác định xe ô tô hay xe máy thì *biển số xe* là mã của xe đó

2. Các đối tượng cần mã hoá

- Các ứng dụng tin học trong cơ quan
- Các nhiệm vụ
- Các modul xử lý
- Các chương trình
- Các tài liệu
- Các tệp
- Các thông tin trong các tài liệu và các tệp
- Các biến trong các chương trình
- ...

3. Chất lượng của mã hoá

Trong thực tế ta gặp rất nhiều đối tượng cần mã hóa như mã hoá ngành nghề đào tạo, mã hoá các bệnh, mã số điện thoại, mã thẻ sinh viên, thẻ bảo hiểm y tế,... chúng ta có nhiều phương pháp mã khác nhau. Do vậy cần xác định một số tiêu chí để đánh giá chất lượng của việc mã hóa.

- **Mã không được nhập nhằng:** Thể hiện ánh xạ 1-1 giữa mã hoá và giải mã. Nghĩa là mỗi đối tượng được xác định rõ ràng với một mã nhất định
- **Thích ứng với phương thức sử dụng:**
 - Để xử lý thủ công mã phải dễ hiểu, dễ giải mã
 - Để xử lý tự động mã phải được xây dựng chặt chẽ (có cú pháp chặt chẽ)
- **Có khả năng mở rộng và xen thêm**
 - Thêm phía cuối (sau) của các mã đã có
 - Xen mã mới vào giữa các mã đã có (thường mã xen phải dùng phương pháp cóc nhảy, *nhảy đều đặn dựa vào một kết quả thống kê hoặc nhảy cóc với bước nhảy có giá trị nhất định để tránh tình trạng “bùng nổ” mã*
- **Phải ngăn gọn** làm giảm kích cỡ của mã, đây cũng là mục tiêu của mã hoá. Tuy nhiên điều này đôi khi mâu thuẫn với khái niệm mở rộng mã sau này.
- **Có tính gợi ý** có nghĩa là khi đọc mã con người có thể đoán ra đối tượng được mã.

4. Các loại mã

❖ **Mã hoá liên tiếp:** Dùng các số nguyên liên tiếp để chỉ các đối tượng

Ví dụ: Các khách hàng của một công ty nào đó có thể được mã hoá theo thứ tự thời gian: 001, 002, 003,...

Ưu điểm:

- Không nhập nhằng
- Đơn giản, thông dụng
- Mở rộng phía sau được

Nhược điểm:

- Không xen được
- Thiếu tính gợi ý cần phải có bảng tương ứng
- Không phân theo nhóm

❖ **Mã hoá theo lát:** Dùng các số nguyên nhưng phân ra từng lát (lớp) cho từng loại đối tượng, trong mỗi lát dùng mã liên tiếp

Ví dụ:

Ưu điểm:

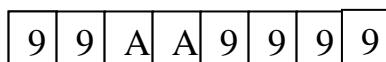
- Không nhập nhằng
- Đơn giản, dễ áp dụng
- Mở rộng và xen thêm được

Nhược điểm:

Không gợi ý

❖ **Mã hoá phân đoạn:** Mã được phân thành nhiều đoạn, mỗi đoạn mang một ý nghĩa riêng

Ví dụ: Số đăng ký xe máy



↓ ↓ ↓
Tỉnh (TP) Chỉ phân khối Số liên tiếp

Ưu điểm:

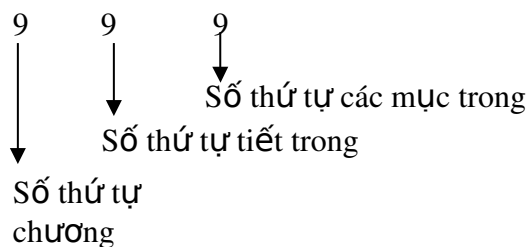
- Không nhập nhằng
- Mở rộng và xen thêm được
- Phổ dụng
- Cho phép thiết lập các phương thức kiểm tra gián tiếp đối với mã của các đối tượng

Nhược điểm:

- Thường dài → Thao tác nặng nề
- Thao tác công kênh vì mã có nhiều đoạn
- Vẫn có thể bị bão hoà

❖ **Mã hoá phân cấp:** Đây là một trường hợp riêng của mã phân đoạn, mỗi đoạn trở tới một tập các đối tượng và tập các đối tượng đó bao nhau từ trái qua phải

Ví dụ:



Ưu điểm:

- Giống như mã phân đoạn
- Tìm kiếm dễ dàng (sử dụng phương pháp duyệt cây)

Nhược điểm: Giống như mã phân đoạn

❖ **Mã hoá diễn nghĩa:** Gán một tên ngắn gọn, nhưng phản ánh nội dung của đối tượng

Ví dụ: Đội bóng các nước tham gia giải Tiger cup được mã hoá bằng cách lấy 3 ký ự đầu của tên nước

VIE: Vietnam, THA: Thailand, SIN: Singapore, IND: Indonesia, MAL: Malaysia

Ưu điểm: Tiện dùng cho xử lý thủ công

Nhược điểm: Không giải mã được bằng máy tính

5. Cách lựa chọn sự mã hoá:

Có nhiều phương pháp mã hoá khác nhau, có thể sử dụng kết hợp nhiều kiểu để đạt chất lượng mã tốt nhất. Việc lựa chọn mã hoá cần dựa vào các yếu tố sau:

- Nghiên cứu việc sử dụng mã sau này
- Nghiên cứu số lượng các đối tượng cần mã hoá, lường trước sự mở rộng của mã
- Nghiên cứu sự phân bố thống kê của các đối tượng cần mã (để phân bố theo lớp)
- Tìm xem đã có mã hoá nào đã được dùng cho các đối tượng này chưa (để kế thừa nếu thấy tốt)
- Thử nghiệm trước khi dùng chính thức

III. Từ điển dữ liệu

Từ điển dữ liệu (còn gọi là từ điển yêu cầu) là bộ phận của tư liệu trong phân tích thiết kế, nó là văn phạm giả hình thức mô tả nội dung của các sự vật, đối tượng theo định nghĩa có cấu trúc. Trong biểu đồ BLD các chức năng xử lý, kho dữ liệu, luồng dữ liệu chỉ mô tả ở mức khái quát thường là tập hợp các khoản mục riêng lẻ. Các khái quát này cần được mô tả chi tiết hoá hơn qua công cụ từ điển dữ liệu

a) Mục đích: Nhằm mô tả tốt hơn các thành phần xuất hiện trong BLD như: Luồng dữ liệu, kho dữ liệu, ... Từ điển dữ liệu là một tư liệu tập trung về mọi tên gọi của tất cả các đối tượng xuất hiện trong hệ thống ở các giai đoạn phân tích, thiết kế, cài đặt và bảo trì

Chẳng hạn:

- Ở mức logic, có:
 - Các luồng dữ liệu, các giao dịch, các sự kiện
 - Các chức năng xử lý
 - Các thực thể
 - Các thuộc tính,...
- Ở mức vật lý, có:
 - Các tệp
 - Các chương trình
 - Các chương trình con, môđun, thủ tục,...

Từ điển dữ liệu là cần thiết đặc biệt cho quá trình triển khai các hệ thống lớn, có nhiều người tham gia, nó cho phép:

- **Trong phân tích và thiết kế:** Quản lý tập trung và chính xác mọi thuật ngữ và các mã dùng trong hệ thống, kiểm soát được sự trùng lặp, đồng nghĩa hay âm dị nghĩa,...
- **Trong cài đặt:** Người cài đặt hiểu được chính xác các thuật ngữ từ kết quả phân tích và thiết kế
- **Trong bảo trì:** Khi cần thay đổi, thì phát hiện được các mối liên quan, các ảnh hưởng có thể nảy sinh. Ví dụ đổi một tên thì biết rõ ràng tên đó được dùng ở những nơi nào để thay đổi

b) Các hình thức thực hiện từ điển dữ liệu:

Từ điển dữ liệu có thể được thực hiện và duy trì theo hai cách:

- **Bằng tay:** Đó là một tập tài liệu (như một từ điển thông thường) thành lập bởi người thiết kế và sau đó được duy trì và cập nhật bởi người quản trị hệ thống.
- **Bằng máy tính:** Dùng một hệ mềm, cho phép dễ dàng thành lập, thay đổi. Trong trường hợp này cần có một ngôn ngữ đặc tả thích hợp, thuận tiện cả cho người và cho máy tính trong việc miêu tả cấu trúc của các dữ liệu phức hợp.

Cũng như từ điển thông thường, từ điển dữ liệu là một tập hợp các mục từ, mỗi mục từ tương ứng với một tên gọi kèm với các giải thích tương ứng. Thường thì mỗi mục từ được chép trên một tờ giấy rời cho dễ sắp xếp.

c) Nội dung các mục từ:

Trong mục từ, ngoài tên gọi và các tên đồng nghĩa thì phần giải thích thường đề cập đến bốn loại đặc điểm:

Đặc điểm về cấu trúc: Là nguyên thủy (đơn) hay phức hợp (nhóm)

Đặc điểm về bản chất: Là liên tục hay rời rạc

Đặc điểm về chi tiết: Miền giá trị, đơn vị đo, độ chính xác, độ phân giải, số lượng, tần số, mức ưu tiên,...

Đặc điểm về liên hệ: Từ đâu và đến đâu, đầu vào và đầu ra, dùng ở đâu,...

Tuy nhiên nội dung của các mục từ thường thay đổi theo loại của đối tượng mang tên gọi. Ta thường phân biệt các loại sau:

- Luồng dữ liệu
- Kho dữ liệu (hay tệp dữ liệu)
- dữ liệu sơ cấp (hay phần tử dữ liệu)
- Chức năng xử lý (hoặc chương trình, môđun)

	Định nghĩa luồng dữ liệu
Tên luồng dữ liệu	: Hoá đơn
Tên đồng nghĩa	: Hoá đơn kiêm phiếu thu
Vị trí (từ/đến)	
Từ	: Lập hoá đơn
Đến	: Giải quyết bán hàng theo tuần
Hợp thành	: Tên khách hàng Ngày hoá đơn Ngày Tháng Năm Các khoản bán hàng Tên mặt hàng Số lượng Thành tiền
Giải thích	: Giải trình tiền trả cho một đơn mua hàng
Lập ngày 10/10/2008	Bởi: N.V.K

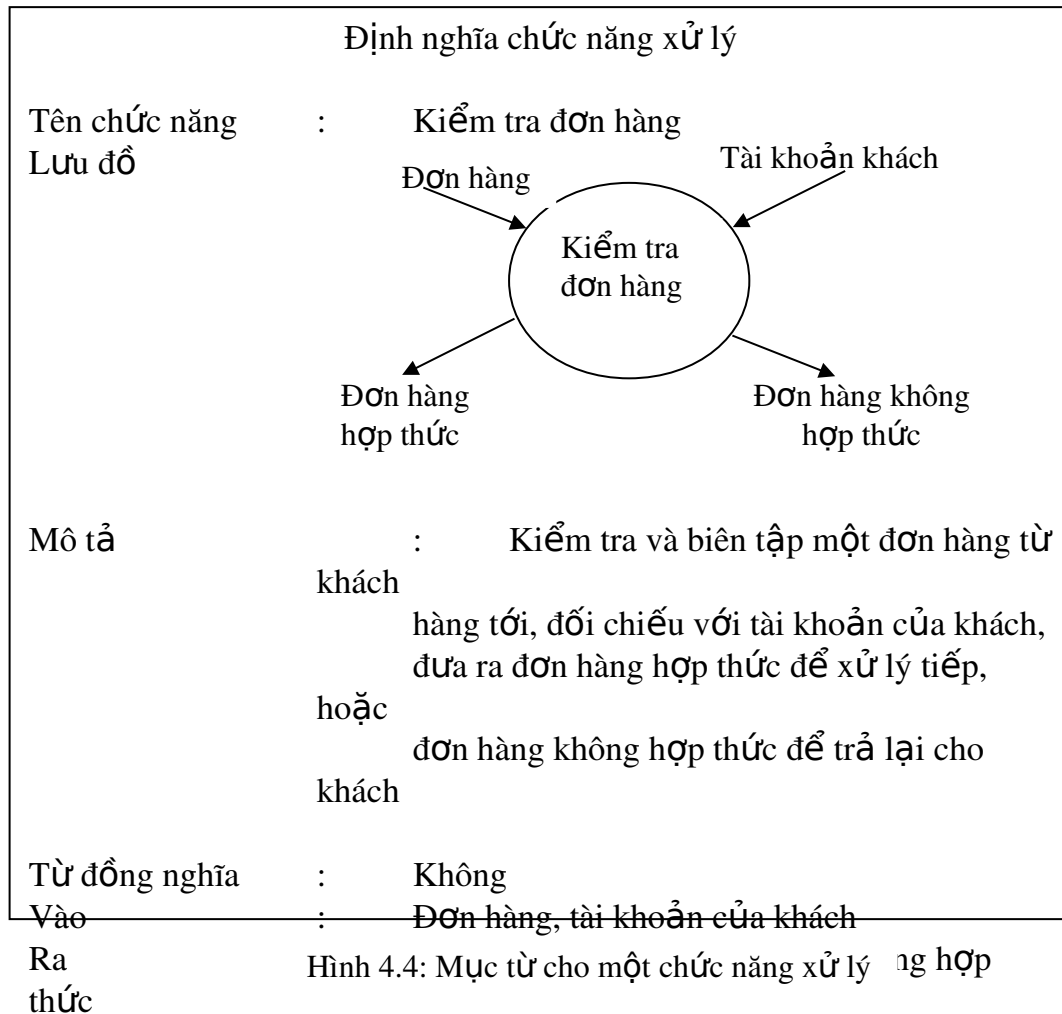
Hình 4.1 : Mục từ cho một luồng dữ liệu

Định nghĩa dữ liệu sơ cấp	
Tên dữ liệu sơ cấp :	Ngày mở tài khoản
Mô tả :	Là ngày mà một tài khoản của khách hàng bắt đầu hoạt động
Từ đồng nghĩa :	Ngày TK
Hợp thành :	Ngày + Tháng + Năm
Bản ghi, tệp liên quan:	Tệp khách hàng
Các xử lý có liên quan:	Biên tập đơn hàng Xây dựng tệp khách hàng
Đặc điểm dữ liệu :	Số ký tự 6, kiểu N
Các giá trị (miễn có thể):	Khuôn dạng DDMMYYYY năm không trước 2000, ngày phải trước ngày

Lập ngày 10/10/2000 Hình 4.2: Mục từ cho một dữ liệu sơ cấp. Ví: N.V.K

Định nghĩa tệp	
Tên tệp :	Nhân viên
Mô tả :	Chứa mọi thông tin về các nhân viên trong cơ quan
Từ đồng nghĩa :	Không
Hợp thành :	Mã số NV Tên NV Ngày bắt đầu công tác Lương Phòng Số hiệu đồ án + } từ 0 - 3 Người phụ trách đồ án
Tổ chức :	Tuần tự theo mã số NV
Các xử lý liên quan :	Cập nhập nhân viên Tìm kiếm nhân viên

Lập ngày 10/10/2000 Hình 4.3: Mục từ cho một tệp dữ liệu. Ví: N.V.K



Hình 4.4: Mục từ cho một chức năng xử lý đơn hàng hợp thức

Lập ngày 10/10/2008 Bởi: N.V.K
 Trong mỗi mục từ, phần định nghĩa dữ liệu (hợp thành, giá trị,...) thường được viết một cách ngắn gọn dựa vào một số quy tắc sau:

- Đối với các dữ liệu phức hợp: Sự hợp thành của nó được thực hiện bởi các cấu trúc tuần tự, chọn, lặp. Có nhiều cách thể hiện các cấu trúc đó:
 - Dùng các từ khoá (tiếng Anh hay tiếng Việt): IS (là), AND (và), OR (hoặc), INTERATION(LẶP), EITHER ()
 - Tuần tự: Dữ liệu A IS dữ liệu P
AND dữ liệu Q
AND dữ liệu R
 - Lặp: Dữ liệu B IS INTERATION OF dữ liệu S
 - Chọn: Dữ liệu C IS EITHER dữ liệu T
OR dữ liệu U
OR dữ liệu V

Thí dụ: Địa chỉ IS Tên
 AND Số nhà
 AND Đường phố
 AND Quận, huyện
 AND Tỉnh, thành phố

Tệp nhân viên IS ITERATION OF
 Bản ghi nhân viên

Giao dịch khách hàng IS EITHER Khách gửi tiền
 OR Khách rút tiền
 OR Khách xem số dư tài khoản

Dùng ký pháp logic EBNF (Extended Backus Naur Form)

:: Có nghĩa là: Hợp thành bởi

+ Có nghĩa là: Và

{...} Có nghĩa là: Lặp

[...] Có nghĩa là: Tùy chọn (không hay một lần)

<...> Có nghĩa là: Hoặc không loại trừ

(... | ...) Có nghĩa là: Hoặc có loại trừ

... Có nghĩa là: Chú thích

Ví dụ: Đơn hàng : : *các thông tin về khách hàng*

Tên khách hàng +
 Địa chỉ khách hàng +
 Số hiệu đơn hàng +
 $\left. \begin{array}{l} \text{Tên mặt hàng +} \\ \text{Mã hàng +} \\ \text{Đơn giá +} \\ \text{Số lượng} \end{array} \right\} +$
 Tổng tiền

Giao dịch khách hàng : : (khách gửi tiền | Khách rút tiền)

- Đối với các dữ liệu đơn (sơ cấp): Định nghĩa theo các giá trị mà nó có thể nhận được (giá trị và ý nghĩa)

<i>Thí dụ:</i>	Giá trị	Ý nghĩa	Giá trị	Ý nghĩa
	AA	lớn	C	Kém
	A	tốt	D	Không chấp nhận được
	B	chấp nhận được	DD	dứt khoát loại trừ

Nhược điểm chính của từ điển dữ liệu là nó thiếu tính hình thức (không chặt chẽ, kém trừu tượng) và nhất là nó không phản ánh được các mối liên quan vốn có giữa các dữ liệu. Bởi vậy, thay vì từ điển dữ liệu, ta sẽ lấy mô hình thực thể/liên kết (E/A) hay mô hình quan hệ làm mô hình dữ liệu của giai đoạn phân tích.

IV. **Mô hình thực thể liên kết:**

Mô hình TT/LK (E/A Entity/Association Model) là mô hình dữ liệu do P.P Chen đưa ra năm 1976 và sau đó được dùng khá phổ biến trên thế giới.

Nó có ưu điểm là khá đơn giản và gắn với tư duy trực quan. Khi xem xét các thông tin, người ta thường gom cụm chúng xung quanh các vật thể. Chẳng hạn các thông tin về tên, tuổi, địa chỉ, chiều cao, cân nặng,... được gom cụm với nhau xung quanh một người, trong khi các thông tin về số đăng ký, nhả máu, kiểu dáng, màu sơn, dung tích xy lanh,... lại được gom với nhau xung quanh một xe máy. Mô hình thực thể/liên kết mô tả tập hợp các dữ liệu dùng trong một hệ thống theo cách gom cụm nhóm như vậy.

1. **Mục đích, sử dụng và yêu cầu:**

- ❖ **Mục đích:** Cho ta một khuôn dạng giúp cho quá trình nhận thức và biểu diễn các dữ liệu sử dụng trong hệ thống thông tin đồng thời cho ta biết cấu trúc cụ thể của dữ liệu
- ❖ **Sử dụng:**
 - o Trong phân tích về dữ liệu của hệ thống cũ
 - o Trong thiết kế về dữ liệu của hệ thống mới
 - o Làm tư liệu, trao đổi
- ❖ **Các câu của bước phân tích dữ liệu:**
 - Không bỏ sót thông tin: điều này có nghĩa là phải phát hiện để đưa vào lược đồ cấu trúc dữ liệu mọi thông tin cần cho hệ thống đang được xây dựng
 - Không dư thừa thông tin: Nghĩa là các thông tin đưa vào lược đồ đó không được trùng lặp, nói một cách khác đi mỗi thông tin chỉ có một bản trong hệ thống. Sự dư thừa thông tin không những làm tốn bộ nhớ, mà quan trọng hơn là dễ gây ra sự mâu thuẫn của thông tin lưu trữ trong cơ sở dữ liệu

Một mô hình thực thể/liên kết được tạo thành từ ba yếu tố cơ bản: Thực thể, liên kết và thuộc tính

2. **Thực thể và kiểu thực thể:**

a. **Thực thể** (entity) : là đối tượng mà ta cần quan tâm trong công tác quản lý, đối tượng đó có thể là:

• Rất cụ thể như:

Nhân viên của một cơ quan cần quản lý

Tờ hoá đơn của doanh nghiệp cần quản lý

• Rất trừu tượng như:

Khoa tin học

Ngành toán ứng dụng,...

Tiêu chuẩn xác định thực thể:

Có ích cho quản lý

Và phân biệt được giữa các thực thể với nhau

Ví dụ: Các khách hàng đều có tài khoản của họ và các nhà cung cấp cung cấp các mặt hàng, ở đây các đối tượng được quan tâm:

Tài khoản là đối tượng cụ thể

Khách hàng

Nhà cung cấp

Mặt hàng

Đối tượng trừu tượng

b. **Kiểu thực thể:**

○ **Kiểu thực thể:** Là một tập hợp chứa tất cả các thực thể cùng loại được mô tả bằng những đặc trưng giống nhau

○ *Ví dụ:* Một nhân viên là một thực thể, tập hợp các nhân viên của cùng một hệ thống tạo thành một kiểu thực thể NHÂN VIÊN

○ *Mô tả kiểu thực thể:* Dùng một hình chữ nhật trong ghi tên của kiểu thực thể

Nhân viên

3. Thuộc tính:

Sau khi đã xác định được các thực thể, tiếp đến ta cần xác định các thông tin cần lưu trữ cho mỗi thực thể (Xác định các thuộc tính)

▪ **Định nghĩa:** Thuộc tính là giá trị thể hiện 1 đặc điểm nào đó của một thực thể hay một liên kết

▪ **Ví dụ:** 1 Cán bộ Nguyễn Văn A

tuổi: 30

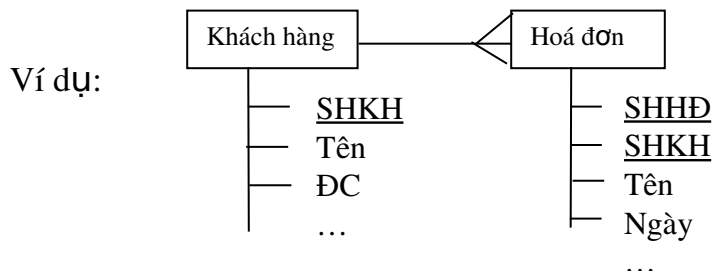
Địa chỉ: 3 Hùng Vương

Trình độ: Kỹ sư

1 thực thể Danh sách các thuộc tính

- **Kiểu thuộc tính:** Là tập hợp các giá trị của cùng loại thuộc tính
Ví dụ: Tuổi: 1,2,3, ...,150 đây chính là 1 kiểu thuộc tính có tên trong 1 tập hợp nào đó
Tuổi, tổng tiền,... Là các kiểu thuộc tính
- **Danh sách các thuộc tính:** Là tập hợp các thuộc tính của một thực thể
Ví dụ: Khách hàng: SHKH; TENKH; DCKH;...
- **Các loại thuộc tính: Có 3 loại**
 - *Thuộc tính khoá:* Để nhận diện và phân biệt các đối tượng với nhau trong cùng một kiểu thực thể. Còn có thể nhận biết một liên kết
Ví dụ: SHKH
 - *Thuộc tính mô tả:* Là những thuộc tính không phải là thuộc tính khoá nó dùng để miêu tả các thực thể và nó chỉ xuất hiện 1 lần trong thực thể mà thôi
Ví dụ: TENKH,DCKH,SOTK,...
 - *Thuộc tính kết nối:* Việc xuất hiện thuộc tính khoá của 1 thực thể trong 1 thực thể khác thể hiện sự kết nối giữa 2 thực thể

Nhận xét: trong quan hệ 1 nhiều thì thuộc tính kết nối thường xuất hiện ở đầu nhiều



4. Liên kết và kiểu liên kết:

- **Một liên kết:** Là sự ghép nối giữa 2 hay nhiều thực thể phản ánh một sự ràng buộc về quản lý
- **Kiểu liên kết:** Là tập hợp các liên kết cùng bản chất. Giữa các kiểu thực thể có thể tồn tại nhiều mối liên kết, mỗi mối liên kết xác định một tên duy nhất

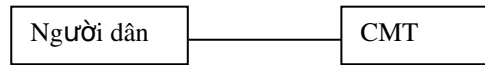
Ví dụ: một quan hệ giữa A và B là một tập hợp các cặp (a,b) trong đó a A; b B

- **Các loại liên kết:**

- *Liên kết 1-1 (đọc là một một):* Hai thực thể A và B có mối liên kết 1-1, nghĩa là ứng với một thực thể trong A có một thực thể trong B và ngược lại. Liên kết này còn gọi là liên kết tầm thường và ít xảy ra trong thực tế, thông thường mang đặc trưng bảo mật. (chẳng hạn một số báo danh ứng với một môn thi có một số phách hay người dân và số chứng minh)

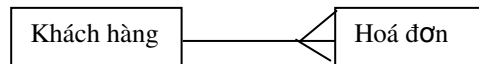
Ví dụ: Nguyễn Văn A có số chứng minh thư: 0904021

Biểu diễn: Biểu diễn kết nối bằng đoạn thẳng giữa hai kiểu thực thể



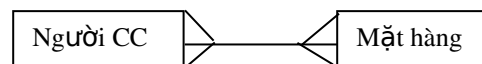
- *Liên kết 1-n (đọc là một nhiều):* Hai thực thể A và B có mối liên kết 1-n, nghĩa là ứng với một thực thể trong A có nhiều thực thể trong B, ngược lại ứng với một thực thể trong B chỉ có một thực thể trong A. Ví dụ mối quan hệ giữa khách hàng với hoá đơn, hay mối quan hệ giữa khách hàng và tài khoản

Biểu diễn: Biểu diễn kết nối bằng đoạn thẳng giữa hai kiểu thực thể và thêm trục 3 (hay chân gà) về phía nhiều



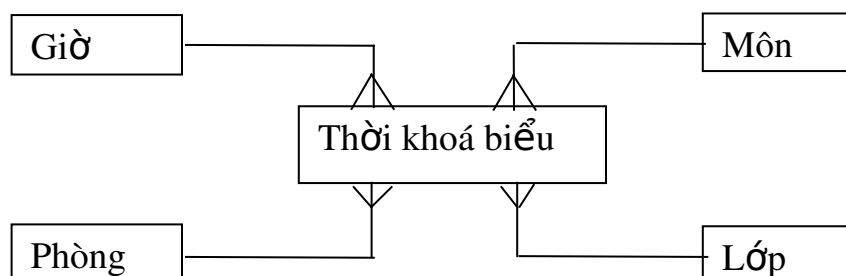
- *Liên kết n-n (đọc là nhiều nhiều):* Hai thực thể A và B có mối liên kết n-n, nghĩa là ứng với một thực thể trong A có nhiều thực thể trong B, ngược lại ứng với một thực thể trong B có nhiều thực thể trong A. Ví dụ mối quan hệ giữa người cung cấp với mặt hàng

Biểu diễn: Không biểu diễn kết nối bằng đoạn thẳng giữa hai kiểu thực thể và thêm hai trục 3 (hay chân gà) về hai phía



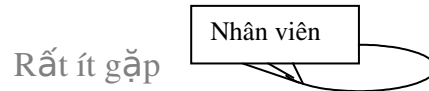
Nhưng không được viết như trên

Chú ý: Khi gặp liên kết nhiều nhiều thì sẽ được thực thể hoá bằng cách bổ sung thực thể trung gian để biến đổi một liên kết n-n thành hai liên kết 1-n.



- *Liên kết đệ qui*: Là mối liên kết giữa 2 thực thể thuộc cùng một kiểu

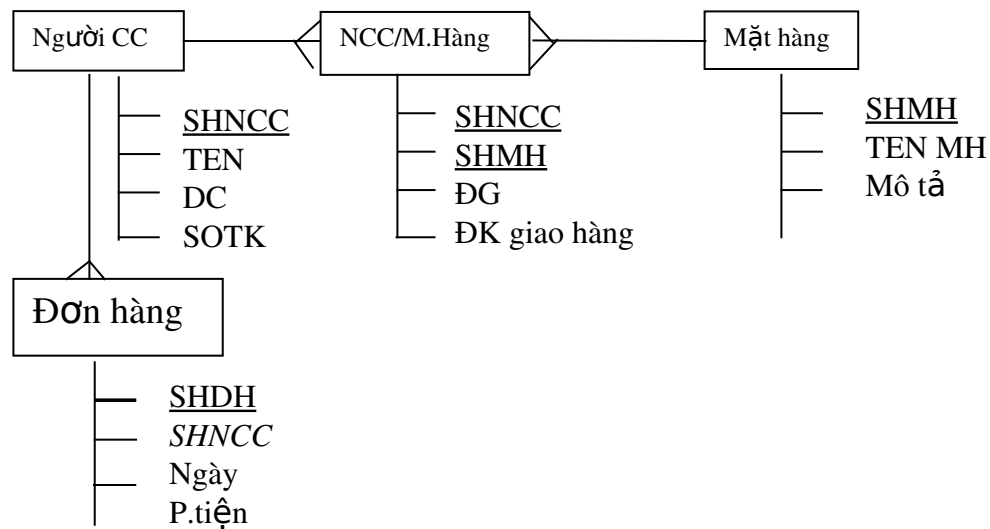
Ví dụ: Mối liên kết giữa nhân viên và thủ trưởng



5. Biểu diễn đồ họa của một thực thể:

Như đã biết, để mô tả một thực thể người ta dùng một hình chữ nhật trong đó có ghi tên thực thể đó, cách mô tả này chỉ là bước đầu, để mô tả chính xác một thực thể thì thực thể đó cần được mô tả kèm theo các thuộc tính của nó (tức là danh sách thuộc tính của nó).

Ví dụ:



Chú ý:

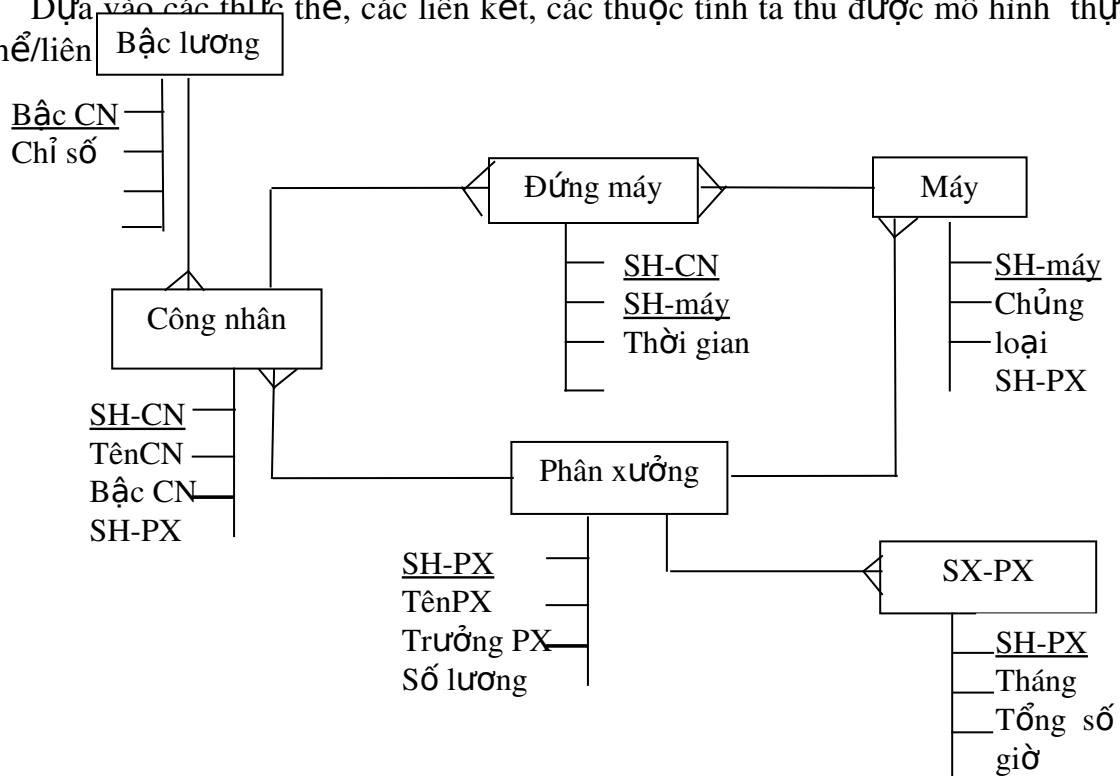
- Trong danh sách các thuộc tính thì thuộc tính khoá được gạch chân
- Thuộc tính kết nối còn gọi là khoá ngoài của kiểu thực thể
- Sự xuất hiện của thuộc tính kết nối là biểu hiện của mối liên kết 1-n

6. Ví dụ: Xét một phạm vi nhỏ trong công tác quản lý ở một nhà máy X. Một cách cụ thể hơn là phạm vi theo dõi lao động củ từng công nhân và phân xưởng (một hệ thống con trong một hệ thống lớn). Để tiện cho

việc biểu diễn, chúng ta sẽ mã hoá (viết tắt) các thuộc tính của các thực thể:

- Đối với công nhân:
 - SH-CN (*số hiệu công nhân*)
 - TênCN (*tên công nhân*)
 - DC-CN (*địa chỉ công nhân*)
 - BậcCN (*bậc thợ*)
 - Chỉ số (*chỉ số lương công nhân*)
- Đối với phân xưởng:
 - SH-PX (*số hiệu phân xưởng*)
 - Tên-PX (*tên phân xưởng*)
 - Trưởng PX (*trưởng phân xưởng*)
 - Số lượng (*số lượng công nhân trong phân xưởng*)
- Đối với máy:
 - SH-máy (*số hiệu máy*)
 - Chủng loại (*chủng loại máy*)
- Đối với hoạt động sản xuất của phân xưởng: (SX_PX)
 - Tháng, thời gian (*thời gian CN làm việc trong tháng trên một máy*)
 - Tổng số giờ (*Tổng số giờ các máy đã chạy (trong một tháng) của PX*)

Dựa vào các thực thể, các liên kết, các thuộc tính ta thu được mô hình thực thể/liên



Hình 4.5: Mô hình E/A

V. Mô hình quan hệ

Tiếp theo việc xét mối liên kết giữa các thực thể, ta cần xét đến mối quan hệ giữa các thuộc tính trong danh sách thuộc tính, bằng cách áp dụng một tập các qui tắc phân tích của *mô hình quan hệ* vào danh sách đó, chuyển chúng thành một dạng mà :

- Tối thiểu việc lặp lại
 - Tránh dư thừa
 - Xác định và giải quyết sự nhập nhằng
- o *Sự lặp*: Là một thuộc tính xuất hiện ở nhiều bảng thực thể mà không phải để liên kết các thuộc tính (Không phải là thuộc tính kết nối)
- o *Dư thừa*: Là thuộc tính có thể suy diễn được từ các thuộc tính khác (VD Tiền=DG*SL)
- o *Nhập nhằng*: Xuất hiện khi nhiều người cùng sử dụng thông tin và được hiểu theo nhiều nghĩa khác nhau

1. Định nghĩa quan hệ:

Gọi $U = \{A_1, A_2, \dots, A_n\}$ là tập hữu hạn các thuộc tính. Mỗi thuộc tính A_i ($i=1, \dots, n$) có miền trị tương ứng là $\text{dom}(A_i)$

Quan hệ trên tập thuộc tính U là tập con của tích Đề các:

$$r \subseteq \text{dom}(A_1) \times \text{dom}(A_2) \times \dots \times \text{dom}(A_n)$$

Ghi chú:

- Đương nhiên quan hệ r có thể bị thay đổi theo thời gian do việc thực hiện các phép cập nhật trên các bộ của quan hệ (bổ xung, loại bỏ, sửa đổi,...). Điều này có nghĩa là một quan hệ r còn là một hàm của thời gian
- Để chỉ một quan hệ r trên tập thuộc tính $\{A_1, A_2, \dots, A_n\}$, ta dùng ký hiệu $r(U)$ hay $r(A_1, A_2, \dots, A_n)$
- Để mô tả quan hệ r gồm p bộ trên lược đồ có n thuộc tính, người ta dùng một bảng gồm n cột và $p+1$ hàng, hàng thứ nhất là tên các thuộc tính, các hàng còn lại, mỗi hàng là một bộ của quan hệ.

A_1	A_2	...	A_n
a_{11}	a_{21}	...	a_{n1}
a_{12}	a_{22}	...	a_{n2}
...

a_{1n}	a_{2p}	...	a_{np}
----------	----------	-----	----------

Ví dụ: Để phục vụ việc chấm công trong một đơn vị sản xuất người ta xét quan hệ chấm-công sau:

SH-CN	SH-máy	Số giờ	SH-PX	Trưởng PX
C1	m_1	20	P_1	Giáp
C1	M_2	30	P_1	Giáp
C2	m_1	10	P_1	Giáp
C2	M_3	15	P_2	Át
C3	M_4	12	P_3	Bính
C4	M_2	22	P_1	Giáp

Hình 4.6: Quan hệ Chấm-công

Lúc đó một bộ $\langle c, m, g, p, x \rangle$ có nghĩa như sau:

Công nhân có mã c, làm việc trên máy m, với số giờ g, tại phân xưởng p do ông x là trưởng phân xưởng

Lưu ý rằng nếu chỉ quan tâm tới tên quan hệ và các thuộc tính người ta thu được lược đồ quan hệ, chẳng hạn:

Chấm-công (SH-CN, SH-máy, Số giờ, SH-PX, Trưởng PX) là một lược đồ quan hệ

Như vậy bảng trên được gọi là lược đồ quan hệ hay đồ thị quan hệ

2. Phụ thuộc hàm:

Trước khi trình bày khái niệm phụ thuộc hàm chúng ta dùng một số qui ước sau trong cách viết:

- Dùng các chữ cái lớn A, B, C, ... để chỉ các tập thuộc tính, A B sẽ viết là AB hoặc A,B
- Dùng các chữ cái thường (có thể có chỉ số) để chỉ giá trị của các thuộc tính, (chẳng hạn a_1 là giá trị của thuộc tính A: $a_1 \in A$)
- Dùng cặp móc nhọn để chỉ các bộ, chẳng hạn $\langle a_1, b_1, c_1 \rangle$ là một bộ của quan hệ R(A,B,C)

a) Định nghĩa phụ thuộc hàm:

Cho quan hệ R(A, B, C), trong đó C có thể rỗng. Ta nói tập các thuộc tính B là *phụ thuộc hàm* vào tập các thuộc tính A, nếu trong R bất cứ hai bộ $\langle a_1, b_1, c_1 \rangle$, $\langle a_2, b_2, c_2 \rangle$ nào mà có $a_1 = a_2$, thì cũng đều có $b_1 = b_2$. Nói cách khác, luôn luôn có

cùng một giá trị của B đi liền với một giá trị cho trước của A trong quan hệ R (cho nên cũng nói A xác định B)

Ký hiệu phụ thuộc hàm: $A \rightarrow B$

Ví dụ: Trong quan hệ chấm-công ở hình 4.6 ta thấy có một số các phụ thuộc hàm như sau:

SH-PX \rightarrow Trưởng PX

SH-máy \rightarrow SH-PX, Trưởng PX

SH-CN, SH-máy \rightarrow Số giờ, SH-PX, Trưởng PX

b) Phụ thuộc hàm sơ đẳng, phụ thuộc hàm trực tiếp

Xét một quan hệ $R(A, B, C)$:

Một phụ thuộc hàm $A \rightarrow B$ là phụ thuộc hàm sơ đẳng nếu không tồn tại tập thuộc tính A' mà $A' \rightarrow B$. Nói cách khác không có thuộc tính thừa trong vế trái của phụ thuộc hàm (tức là các thuộc tính ở vế phải phụ thuộc hoàn toàn vào toàn bộ các thuộc tính ở vế trái)

Ví dụ: Trong quan hệ chấm-công trên thì:

SH-PX \rightarrow Trưởng PX	}	Là PTH sơ đẳng
SH-CN, SH-máy \rightarrow Số giờ		

SH-CN, SH-máy \rightarrow SH-PX Không phải là PTH sơ đẳng

Vì chỉ cần SH-CN \rightarrow SH-PX hoặc SH-máy \rightarrow SH-PX

Một phụ thuộc hàm $A \rightarrow B$ là phụ thuộc hàm trực tiếp nếu không tồn tại tập thuộc tính C trong r khác với A và B, mà $A \rightarrow C$ và $C \rightarrow B$

Ví dụ: Trong quan hệ chấm-công trên thì:

SH-PX \rightarrow Trưởng PX là PTH trực tiếp

Còn SH-máy \rightarrow Trưởng PX Không phải là PTH trực tiếp

Vì SH-máy \rightarrow SH-PX Và SH-PX \rightarrow Trưởng PX

c) Khoá

- Định nghĩa

Xét một quan hệ $R(A, B, C)$:

A là khoá nếu $A \rightarrow B$ là một phụ thuộc hàm sơ đẳng. Nói cách khác. Nói cách khác:

- o Giá trị của A xác định duy nhất giá trị của các thuộc tính còn lại, tức là xác định duy nhất một bộ, và

- o Trong A không có thuộc tính thừa (A là tối thiểu)
- **Khoá đơn:** Mỗi khoá là một thuộc tính
SH-PX Trưỡng PX

Khoá kép: Mỗi khoá lớn hơn một thuộc tính
SH-CN, SH-máy Số giờ

3. Các dạng chuẩn của quan hệ:

Để ý rằng nếu trong một quan hệ nào tồn tại các phụ thuộc hàm không sơ đẳng, không trực tiếp thì dẫn đến việc dư thừa thông tin, cho nên khi thiết kế một hệ thống nào đó thì ta cố gắng loại trừ các phụ thuộc hàm không sơ đẳng, không trực tiếp giữa khoá và các thuộc tính không khoá. Bởi vì những yếu tố đó sẽ gây ra các dị thường trong CSDL khi thực hiện các thao tác Cập nhật, loại bỏ, bổ sung.

❖ Một quan hệ R ở dạng chuẩn 1 (1NF) nếu miền giá trị của mỗi thuộc tính đều là miền đơn (*Qui tắc loại bỏ nhóm lặp làm xuất hiện kiểu thực thể mới*)

❖ Một quan hệ R ở dạng chuẩn 2 (2NF) nếu nó đồng thời thoả mãn hai điều kiện sau:

- Đã ở dạng chuẩn 1
- Nếu mọi phụ thuộc hàm giữa khoá và các thuộc tính không khoá đều là sơ đẳng

(Qui tắc loại bỏ những phụ thuộc bộ phận vào khoá)

❖ Một quan hệ R ở dạng chuẩn 3 (3NF) nếu nó đồng thời thoả mãn hai điều kiện sau:

- Đã ở dạng chuẩn 2
- Nếu mọi phụ thuộc hàm giữa khoá và các thuộc tính không khoá đều là trực tiếp

(Qui tắc loại bỏ những phụ thuộc giữa các thuộc tính không phải là khoá)

4. Chuẩn hoá:

Thuật ngữ chuẩn hoá được hiểu là sự phân rã, nhưng không làm mất mát thông tin một quan hệ R thành một tập các quan hệ ở 3NF.

Có nhiều giải thuật chuẩn hoá, nhưng tập trung chủ yếu theo hai hướng phân tích và tổng hợp.

❖ Chuẩn hoá theo hướng phân tích:

Thực hiện chuẩn hoá dần theo ba bước: 1NF, 2NF và 3NF

- Đưa về dạng chuẩn 1 bằng cách các thuộc tính lặp (không đơn):
 - Nhóm các thuộc tính đơn còn lại tạo thành một quan hệ. Chọn khoá cho nó.
 - Nhóm các thuộc tính lặp tách ra, cùng với khoá của quan hệ trên tạo thành một quan hệ (hay một số quan hệ theo chủ đề). Chọn khoá cho (các) quan hệ này, thường là khoá kép, trong đó khoá của quan hệ trên là một thành phần của khoá.

Các quan hệ lập được đều là 1NF

- Đưa về dạng chuẩn 2, bằng cách tách các nhóm thuộc tính phụ thuộc hàm vào một bộ phận của khoá (*xét các quan hệ có khoá kép*)
 - Nhóm còn lại tạo thành một quan hệ với khoá cũ
 - Mỗi nhóm tách ra, gồm các thuộc tính cùng phụ thuộc vào một (hay một số) thuộc tính nào đó của khoá, cộng thêm (các) thuộc tính mà chúng phụ thuộc tạo thành một quan hệ, với khoá là các thuộc tính cộng thêm này

Các quan hệ lập được đều là 2NF

- Đưa về dạng chuẩn 3, bằng cách tách các nhóm thuộc tính phụ thuộc hàm vào một (hay một số) bộ phận của khoá.
 - Nhóm còn lại tạo thành một quan hệ với khoá cũ
 - Mỗi nhóm tách ra, gồm các thuộc tính cùng phụ thuộc hàm vào một (hay một số) thuộc tính không phải là khoá, cộng thêm (các) thuộc tính mà chúng phụ thuộc tạo thành một quan hệ, với khoá là các thuộc tính cộng thêm này

Các quan hệ lập được đều là 3NF

Ví dụ 1: Xuất phát từ một đơn hàng, trên đó ta gom được một số các thuộc tính sau (đây chính là các thông tin cần quản lý)

- Các thuộc tính đơn:

SH-ĐH

SH-NgCC

Tên-NgCC

ĐC-NgCC

Ngày-ĐH

Tổng cộng

- Các thuộc tính lặp:

Mã-MH
 Mô tả-MH
 Đơn vị tính
 Đơn giá
 Số lượng đặt
 Thành tiền

Các thuộc tính Thành tiền, tổng cộng là những thuộc tính tính toán bị loại khỏi danh sách.

Các bước chuẩn hoá 1NF, 2NF, 3NF thực hiện theo các bước của chuẩn hoá, trong đó phân rã dần dần các quan hệ, khi đi từ cột này qua cột khác, từ trái sang phải. Một mũi tên diễn tả rằng quan hệ cũ được chuyển y nguyên sang cột mới.

Danh sách thuộc tính	1NF	2NF	3NF
SH-ĐH	<u>SH-ĐH</u>	→	<u>SH-ĐH</u>
SH-NgCC	SH-NgCC		<u>SH-NgCC</u>
Tên-NgCC	Tên-NgCC		Ngày-ĐH
ĐC-NgCC	ĐC-NgCC		
Ngày-ĐH	Ngày-ĐH		<u>SH-NgCC</u>
Mã-MH			Tên-NgCC
Mô tả-MH			ĐC-NgCC
Đơn vị tính			
Đơn giá	<u>SH-ĐH</u>	<u>SH-ĐH</u>	→
Số lượng đặt	Mã-MH	<u>Mã-MH</u>	
	Mô tả-MH	Số lượng đặt	
	Đơn vị tính		
	Đơn giá	<u>Mã-MH</u>	→
	Số lượng đặt	Mô tả-MH	
		Đơn vị tính	
		Đơn giá	

Cuối cùng ta thu được 4 quan hệ ở 3NF, đặt tên cho các quan hệ đó (căn cứ vào khoá của chúng) ta có:

Đơn hàng (SH-ĐH, SH-NgCC, Ngày-ĐH)

Người cung cấp (SH-NgCC, Tên-NgCC, ĐC-NgCC)

Dòng đơn hàng (SH-ĐH, Mã-MH, Số lượng đặt)

Mặt hàng (Mã-MH, Mô tả-MH, Đơn vị tính, Đơn giá)

Ví dụ 2: Là quan hệ Chấm-công cho ở hình 4.6 như sau:

SH-CN	SH-máy	Số giờ	SH-PX	Trưởng PX
c ₁	m ₁	20	p ₁	Giáp
c ₁	m ₂	30	p ₁	Giáp
c ₂	m ₁	10	p ₁	Giáp
c ₂	m ₃	15	p ₂	Ất
c ₃	m ₄	12	p ₃	Bính
c ₄	m ₂	22	p ₁	Giáp

Hình 4.6: Quan hệ Chấm-công

Ở đây quan hệ đã ở 1NF, ta chỉ cần thực hiện hai bước 2NF và 3NF như sau:

Danh sách thuộc tính	1NF	2NF	3NF
SH-CN	<u>SH-CN</u>	<u>SH-CN</u>	→
SH-máy	<u>SH-máy</u>	<u>SH-máy</u>	
số giờ	số giờ	số giờ	
SH-PX	SH-PX		
Trưởng PX	Trưởng PX		
		<u>SH-máy</u>	<u>SH-máy</u>
		SH-PX	SH-PX
		Trưởng PX	
			<u>SH-PX</u>
			Trưởng PX

Đặt tên cho các quan hệ 3NF thu được, ta có kết quả chuẩn hoá sau:

Công (SH-CN, SH-máy, số giờ)

Máy (SH-máy, SH-PX)

Phân xưởng (SH-PX, Trưởng PX)

Nếu viết đầy đủ giá trị các thuộc tính ta có 3 bảng giá trị sau:

SH-CN	SH-máy	Số giờ
c_1	m_1	20
c_1	m_2	30
c_2	m_1	10
c_2	m_3	15
c_3	m_4	12
c_4	m_2	22

SH-máy	SH-PX
m_1	p_1
m_2	p_1
m_3	p_2
m_4	p_3

SH-PX	Trưởng PX
p_1	Giáp
p_2	Ất
p_3	Bính

Kết quả cho thấy ta đã loại bỏ được các khuyết tật như:

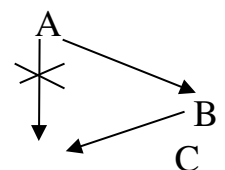
- Các sự lặp lại thông tin như (p_1 , Giáp) lặp lại 3 lần, thì nay không còn nữa
- Khi cập nhật (p_1 , Giáp) thành (p_1 , Mùi) thì chỉ cần điều chỉnh một chỗ (ở quan hệ phân xưởng) mà không phải là ở 3 chỗ như cũ
- Khi bổ sung thông tin (p_4 , Đinh) chỉ việc thêm một bộ mới cho quan hệ Phân xưởng (mà không cần chờ đợi một sự chấm công)
- Khi loại bỏ các thông tin về công nhân c_3 (đã chết) chỉ việc loại các dòng 4 và 5 trong quan hệ chấm công, mà không làm mất thông tin (p_3 , Bính) nằm ở quan hệ phân xưởng

Chuẩn hoá theo hướng tổng hợp:

Cách làm này cho ngay các lược đồ quan hệ 3NF mà không qua các giai đoạn 1NF, 2NF.

- Xuất phát từ một danh sách các thuộc tính (loại trừ những thuộc tính tính toán, các thuộc tính có giá trị không đổi)
- Lập danh sách các phụ thuộc hàm có thể tồn tại giữa các thuộc tính

Ví dụ: A B; A,B C; A,B F



- (iii) Xây dựng một đồ thị phụ thuộc hàm như sau:
 - o Mỗi thuộc tính trong danh sách là một nút
 - o Mỗi nhóm thuộc tính là vế trái của một PTH cũng là một nút
 - o Nếu A B thì ta vẽ một cung từ nút A tới nút B

Sau đó:

- (iv) Loại bỏ các phụ thuộc hàm không trực tiếp (xoá các cung khép kín)
- (v) Dùng các hình chữ nhật để khoanh vùng trên đồ thị để thu được các quan hệ như sau: Mỗi nút trong (nút có con) lấy làm khoá, gom nút đó lại cùng với các con của nó tạo thành một quan hệ. Mỗi quan hệ thu được đều là 3NF

Ví dụ: Với đơn hàng ở ví dụ trên, ta có các phụ thuộc hàm sau:

SH-ĐH SH-NgCC, Tên-NgCC, ĐC-NgCC, Ngày-ĐH

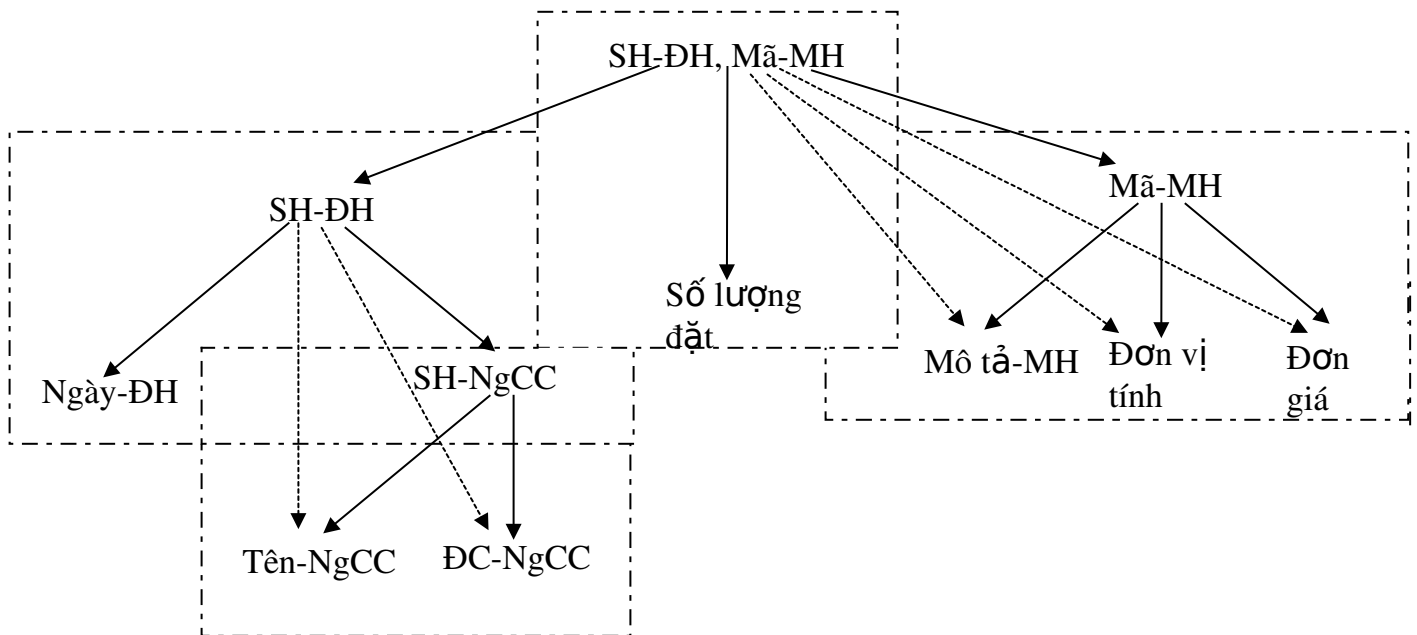
SH-NgCC Tên-NgCC, ĐC-NgCC

Mã-MH Mô tả-MH, Đơn vị tính, Đơn giá

SH-ĐH, Mã-MH Mô tả-MH, Đơn giá, Đơn vị tính, Số lượng đặt

Bổ sung thêm phụ thuộc hàm tầm thường: SH-ĐH, Mã-MH SH-ĐH, Mã-MH

Ta lập được đồ thị phụ thuộc hàm như sau:



Sau khi khoanh vùng (giải thích trên cách khoanh này), đặt tên cho các quan hệ, ta thu được:

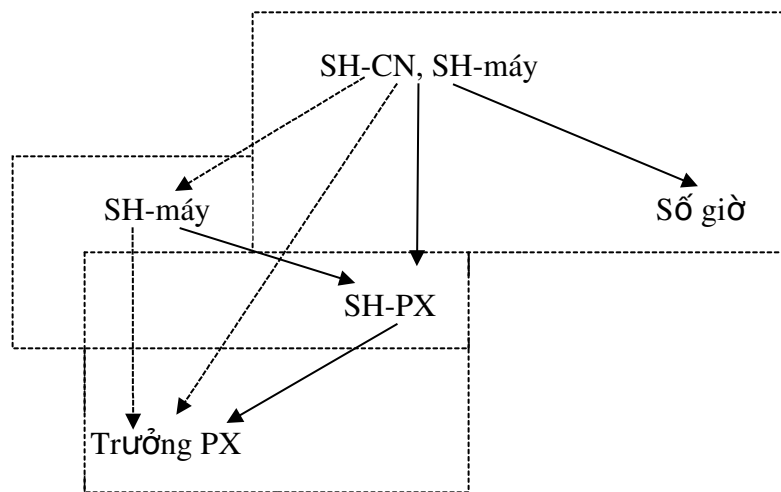
Đơn hàng (SH-ĐH, SH-NgCC, Ngày-ĐH)

Người cung cấp (SH-NgCC, Tên-NgCC, ĐC-NgCC)

Dòng đơn hàng (SH-ĐH, Mã-MH, Số lượng đặt)

Mặt hàng (Mã-MH, Mô tả-MH, Đơn vị tính, Đơn giá)

Chuyển sang ví dụ chấm công, tương tự như trên ta cũng có:



Sau khi khoanh vùng (giải thích trên cách khoanh này), đặt tên cho các quan hệ, ta thu được:

Đứng máy (SH-CN, SH-máy Số giờ)

Máy (SH-máy SH-PX)

Phân xưởng (SH-PX Trưởng PX)

CHƯƠNG V: KHẢO SÁT HIỆN TRẠNG VÀ XÁC LẬP DỰ ÁN

Mục tiêu của chương này là có được một sự hiểu biết đầy đủ về các vấn đề, các yêu cầu của người dùng để có thể hình dung được một cách toàn diện các vấn đề của dự án, ước lượng được giá thành và thời gian thực hiện.

Các hoạt động chính cần làm trong giai đoạn này là:

- o *Tìm hiểu thấu đáo về các vấn đề của người dùng và những gì cần thiết để giải quyết vấn đề đó*
- o *Cần phải quyết định có thực hiện hay không thực hiện dự án. Ta cần phải biết chắc rằng dự án là khả thi và có nhiều cơ hội để thành công*
- o *Nếu dự án có thể thực hiện được, cần phân tích đánh giá các rủi ro có thể xảy ra và chi tiết hoá tất cả các kết quả cần đạt được, khi nào và giá thành là bao nhiêu?*

Cũng cần nói thêm là ngay từ giai đoạn này, ta phải bắt đầu ngay các hoạt động về quản lý dự án, xem xét, báo cáo, tư liệu hoá và tiếp tục tiến hành các hoạt động đó cho đến khi kết thúc dự án.

I. Mục đích, yêu cầu (của giai đoạn I):

1) Mục đích:

Khi triển khai một hệ thống tin, thường gặp một số khó khăn như sau:

- Sự phức tạp, đa dạng và xa lạ của hệ thống cần phát triển như: Quản lý bay, kế toán, ngân hàng, thuế vụ, hải quan, ... Vậy tất cả các nhóm dự án đó đã từng tiếp cận khi nào chưa?
- Phải thiết lập được cầu nối giữa những người phát triển hệ thống với người dùng
- Đối đầu với sự thay đổi liên tục về đủ loại các phương diện: Môi trường, công nghệ, chất lượng nhân lực,...

Đây chính là bài toán, suy cho cùng do chính chúng ta đặt ra, cần giải quyết một cách đồng bộ

Tóm lại, giai đoạn khảo sát hệ thống nhằm bốn mục đích sau:

- ❖ *Tìm hiểu, đánh giá hiện trạng sự hoạt động của hệ thống cũ*
- ❖ *Xác định phạm vi, khả năng, mục tiêu dự án*
- ❖ *Phác hoạ giải pháp mới, cân nhắc tính khả thi của dự án*
- ❖ *Vạch kế hoạch cho dự án cùng với các dự trù tổng quát*

2) Yêu cầu:

Tất cả các kết quả thu được trong giai đoạn này cần phải viết ra thành tài liệu, nội dung của tài liệu này bao gồm:

- Giới thiệu về nền tảng cơ sở, về tổ chức
- Đặt vấn đề
- Mô tả các giải pháp kỹ thuật có thể sử dụng để giải quyết vấn đề như: Các thiết bị phần cứng, phần mềm khác nhau, mua hoặc tự xây dựng lấy các ứng dụng,...
- Đánh giá về tài chính cho mỗi giải pháp đó
- Phân tích đề xuất lựa chọn giải pháp tối ưu
- Chứng tỏ rằng tại thời điểm hiện tại đơn vị có thể triển khai thực hiện dự án khả thi này.
- Tiếp tục triển khai dự án như thế nào ?

II. Tìm hiểu, đánh giá hiện trạng:

Thường thì chúng ta phát triển một hệ thống mới nhằm thay thế cho một hệ thống cũ đã có sẵn, nhưng không còn đáp ứng được những yêu cầu mới nữa. Bởi vậy việc tìm hiểu những nhu cầu mà hệ thống mới cần đáp ứng phải bắt đầu từ việc khảo sát và đánh giá hệ thống cũ đó. Do hệ thống này đang tồn tại, đang hoạt động nên ta gọi là hiện trạng. Việc tiến hành khảo sát là nhằm để:

- Tiếp cận với nghiệp vụ chuyên môn, môi trường hoạt động của hệ thống
- Tìm hiểu các chức năng, nhiệm vụ và cung cấp cách hoạt động của hệ thống
- Chỉ ra những chỗ hợp lý của hệ thống cũ, đây chính là cái kế thừa và đồng thời cũng làm sáng tỏ những bất hợp lý của hệ thống cũ là cái cần sửa đổi, điều chỉnh.

1) Các phương pháp khảo sát hiện trạng:

❖ Khảo sát để:

- Xác định ai là những người dùng
- Xác định cách thu thập thông tin ra sao?
- Thu thập thông tin và diễn tả thông tin
- Mô tả hệ thống hiện thời

❖ Qui trình khảo sát được tiến hành qua 4 bước:

Quyết định, lãnh đạo: Bao gồm ban giám đốc, hội đồng quản trị,... Là những người ra quyết định, quan tâm đến việc ra quyết định dài hạn(Chiến lược), đây là những nguồn tốt đáp ứng nhu cầu của thông tin quản lý

Chuyên gia, cố vấn: Là những người có nghiệp vụ cao, lĩnh vực chuyên sâu của họ có thể bị ảnh hưởng đến hệ thống mới, không nên bỏ qua những vấn đề mà họ đề xuất

Điều phối, quản lý: Những người quản lý (Trưởng, phó phòng, ban, phân xưởng), là những người thường biết rõ về cơ quan mình nhưng thường không nhìn nhận vấn đề trong tương lai xa được, họ chỉ là những tác nhân quan trọng trong việc ra quyết định chứ không phải là những người ra quyết định

Thao tác, thực hành: Bao gồm thủ kho, kế toán, ... Là những người trực tiếp sử dụng hệ thống (họ là những người làm việc trực tiếp với các thao tác của hệ thống, do đó họ có thể nhận ra được những khó khăn, những vấn đề không ai biết đến ngoài họ. Sự cộng tác của họ có thể là xây dựng hoặc phá bỏ một hệ thống mới)

❖ Các hình thức tiến hành:

▪ Quan sát theo dõi: Chính thức và không chính thức

- *Chính thức*: Việc quan sát và theo dõi sẽ được thông báo trước, có trong lịch trình. Để thấy rằng với phương pháp này sẽ không gặp trở ngại nào về phương diện thời gian và khả thi, nhưng rất có thể sẽ không thu được những thông tin không chính xác.
- *Không chính thức*: Đặc trưng của phương pháp này là chúng ta sẽ đơn phương tiến hành quan sát mà không báo trước, do đó các thông tin thu thập được sẽ trung thực và chính xác hơn, nhưng rất có thể không thực hiện được buổi quan sát, theo dõi (Ví dụ đơn vị cần quan sát có buổi họp công đoàn vào đúng lúc ta đến quan sát)

Bởi vậy một biện pháp dung hòa nên tiến hành ở đây là kết hợp một cách linh hoạt cả 2 phương pháp

▪ Phỏng vấn:

Đó là cách thức làm việc tay đôi hay theo nhóm, trong đó điều tra viên đưa ra các câu hỏi và chốt lọc lấy thông tin qua câu trả lời của những người được điều tra, cũng cần nhấn mạnh rằng đây là phương pháp cơ bản thông dụng cho mọi cuộc điều tra, chắc chắn là chúng ta ít nhiều cũng đã được phỏng vấn một đôi lần.

Có hai loại câu hỏi thường được sử dụng:

- Câu hỏi mở (mang tính gợi ý), là câu hỏi mà khả năng trả lời là rất lớn, ngay cả người trả lời thực hiện phỏng vấn cũng không hình dung hết được. Thể loại câu hỏi này tỏ ra có ích khi chưa có ý định rõ ràng, mà chỉ có ý thăm dò, gợi mở vấn đề. (Câu hỏi mở nên dùng cho người có hiểu biết rộng như cán bộ lãnh đạo, chuyên gia, cố vấn,...)

- Câu hỏi đóng là câu hỏi mà các phương án trả lời đã được dự kiến sẵn, người được phỏng vấn chỉ cần khẳng định đó là phương án nào. Thể loại câu hỏi này rất có ích khi ta có sẵn chủ đề điều tra và cần biết rõ các chi tiết nào.

- Phiếu điều tra

Đây là một hình thức phỏng vấn từ xa, các câu hỏi được liệt kê trong một mẫu điều tra, và người được phỏng vấn sẽ ghi các câu trả lời vào mẫu

Việc sử dụng các loại câu hỏi cũng như trật tự các câu hỏi cũng giống như khi phỏng vấn trực tiếp, người được hỏi có khi chỉ cần trả lời một phần các câu hỏi, thậm chí không trả lời.

So với phỏng vấn trực tiếp, phương pháp này thiếu sự giao tiếp giữa người phỏng vấn và người trả lời, cho nên thông tin nhận được có thể kém chính xác, nhưng đổi lại điểm mạnh của nó là ở chỗ có thể mở rộng diện điều tra mà ít tốn kém.

- Nghiên cứu tài liệu viết

Phương pháp này được tiến hành thông qua việc tham khảo các chứng từ giao dịch như : Hoá đơn, đơn hàng, phiếu tồn kho, ... hoặc các loại sổ sách, tài liệu khác có trong đơn vị, bằng cách này ta sẽ thu nhận được nhiều loại thông tin đa dạng về thể loại, phong phú về nội dung.

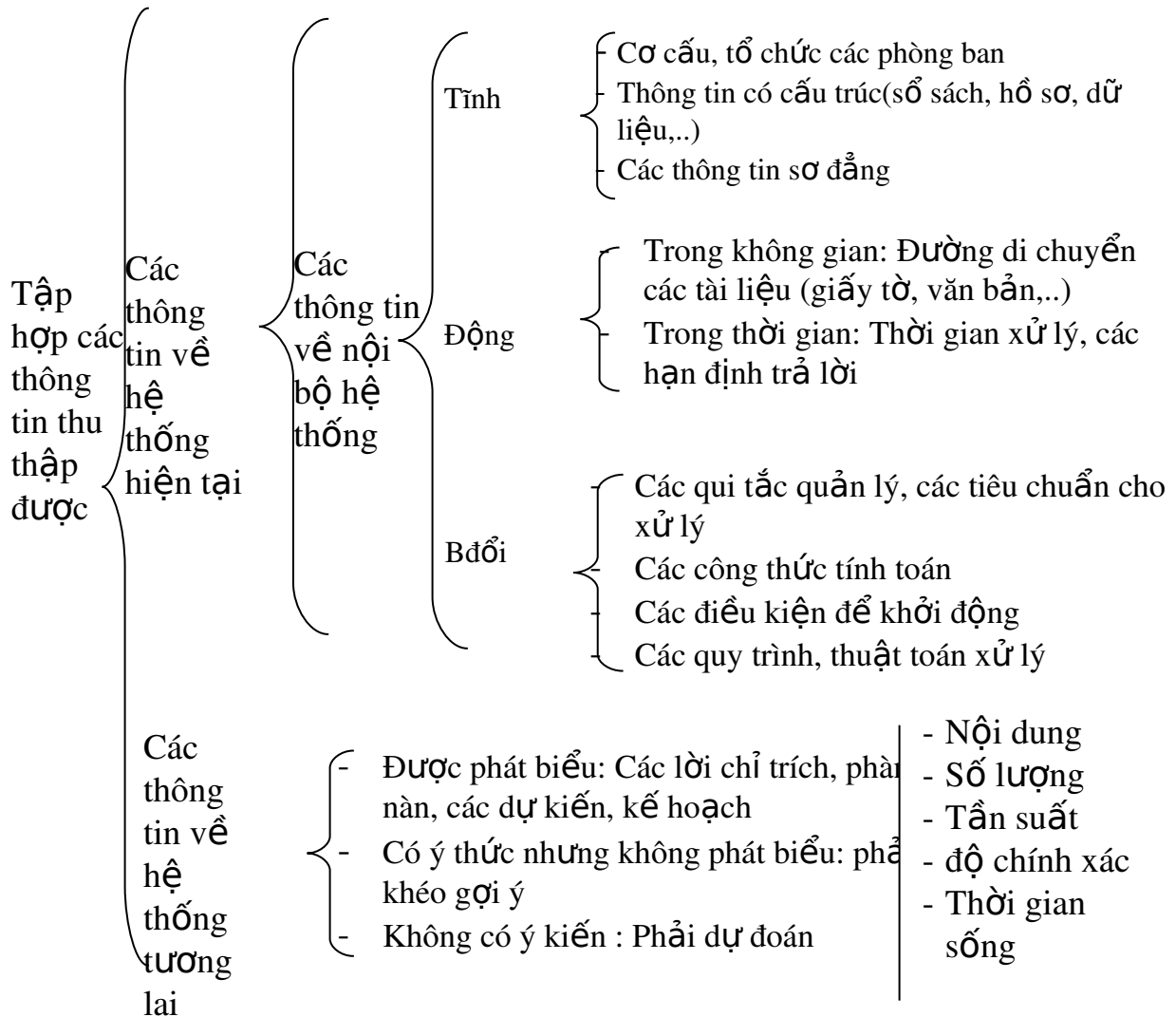
- Các buổi họp, hội thảo

- ❖ Một số lưu ý:

- Kinh nghiệm cho thấy tốt nhất là các chuyên gia phân tích và tin học hãy hỗ trợ cùng với người dùng viết lên được rõ yêu cầu của họ. Chuyên gia tin học không thể nghĩ thay cho người sử dụng những yêu cầu thực sự trong công việc nghiệp vụ của họ được. Nhưng mặt khác, vì những người dùng có thể chưa hiểu rõ được là CNTT có thể giúp họ được những gì một cách cụ thể nên đôi khi họ khó phát biểu được chính xác các mong muốn của mình, mặc dù chính họ là người hiểu rõ hơn ai cả hệ thống đang vận hành và tồn tại cần cải tiến.
- Phải tìm đến những người dùng tiêu biểu đầu cuối, những người có quyền cho ý kiến quyết định về hệ thống định xây dựng và đánh giá xem nó sẽ ảnh hưởng đến công việc của đơn vị. Nếu dự án là nội bộ thì việc tiếp cận đến người dùng không mấy khó khăn, nhưng đối với dự án có sử dụng lực lượng ở bên ngoài thì người quản lý dự án phải làm sao tạo điều kiện cho việc giao tiếp giữa hai bên
- Qui trình phỏng vấn thường đi từ qui trình thông tin trong đơn vị, đầu ra, tần suất, độ chính xác, thời gian và sau đó mới xác định để đáp ứng được thì cần đầu vào là gì, ở đâu, khi nào, ai có trách nhiệm và tại sao phải cung cấp.

- Đối với những yêu cầu không rõ ràng, có thể phải làm số dựng mẫu sẵn hỏi ý kiến người dùng, hay tìm đến hỏi người dùng khác và chính xác hoá các yêu cầu một cách dần dần, qua nhiều lần
- Tránh thay đổi yêu cầu khi đã bắt tay vào thực hiện dự án vì sẽ tốn kém thêm.
- Đối với các vấn đề chưa xác định rõ (ví dụ việc cần một hệ điều hành nào đó), có thể có các giả định trước và nhấn mạnh sẽ phải xử lý ra sao trong trường hợp sau này sẽ không theo các giả định đó nữa.
- Sử dụng hai bước ước lượng nếu dự án quá phức tạp. Tại thời điểm này có thể ước lượng với sai số $\pm 50\%$. Đó là ước lượng loại C
- Tài liệu này nhất thiết phải được người dùng nhất trí thông qua. Đây là một trong những điểm mốc rất quan trọng

2) Phân loại thông tin đã thu thập:



3) Phát hiện những yếu kém của hiện trạng và yêu cầu cho tương lai

Điểm kết thúc của giai đoạn khảo sát hệ thống là chỉ ra được những yếu kém của hiện trạng và các yêu cầu cho tương lai.

- ❖ Sự yếu kém của hệ thống thể hiện dưới ba phương diện sau:
 - Sự thiếu, vắng: Thiếu thông tin cho xử lý, thiếu nhân lực, thiếu phương tiện xử lý thông tin, bỏ sót công việc đáng làm.
 - Sự kém hiệu lực thể hiện ở nhiều mặt :
 - Hệ thống qua tải, ùn tắc về thông tin
 - Phương pháp xử lý không chặt chẽ
 - Cơ cấu tổ chức bất hợp lý
 - Vận chuyển thông tin không hợp lý, quá nhiều khâu,...
 - Giấy tờ, sổ sách trình bày kém, thiếu thông tin, cấu trúc dở
 - Sự ùn tắc, quá tải, người làm quá mệt mỏi
 - Sự tốn kém:
 - Chi phí quá cao
 - Lãng phí vô ích
- ❖ Yêu cầu cho tương lai: Đồng thời với việc chỉ ra các yếu kém của hệ thống đang tồn tại, ta cũng cần phải nêu ra các yêu cầu cho tương lai, đó là:
 - Những nhu cầu chưa được đáp ứng trong hiện tại
 - Những mong muốn, nguyện vọng của nhân viên
 - Các dự kiến, kế hoạch phát triển trong tương lai của lãnh đạo
- ❖ Xét ví dụ cung ứng vật tư:

Phê phán hiện trạng: Với hệ thống cung ứng vật tư như được mô tả trong ví dụ có thể nêu ra các yếu kém sau:

 - Thiếu vắng: Thiếu chức năng quản lý kho với một kho hàng lưu trữ các mặt hàng thông dụng, để mỗi khi một phân xưởng nào đó cần đến có thể cung ứng ngay mà không cần phải đi mua, mặt khác bộ phận này sẽ chủ động đề xuất việc đi mua các mặt hàng này để bổ sung vào kho khi thấy chúng sắp hết.
 - Kém hiệu lực: Thời gian từ khi một phân xưởng gửi dự trù đến khi phân xưởng này nhận được vật tư quá dài (do phải qua các khâu tập hợp các bản dự trù, tìm nhà cung cấp, gửi đơn hàng, đến khi hàng về thì việc xác định địa chỉ phát hàng lại tiến hành thủ công.

- Tốn kém: Do công tác đối chiếu thủ công nên phải sử dụng nhiều nhân công và việc sai sót, chậm trễ là đương nhiên

III. Xác định phạm vi, mục tiêu, ưu tiên của dự án

Sau khi đã thấy rõ các yếu kém của hiện trạng và các yêu cầu phát triển của hệ thống trong tương lai, thì việc xây dựng một hệ thống mới thay thế cho hệ thống cũ cần phải đưa ra bàn bạc để đi đến sự nhất trí giữa các bên.

Các công việc cần thực hiện là:

- Xác định phạm vi và các hạn chế của dự án
- Xác định các mục tiêu và ưu tiên cho dự án
- Đề xuất một số giải pháp và tính khả thi của chúng
- Lập kế hoạch triển khai dự án

1. Xác định phạm vi và các hạn chế :

Phạm vi dự án là vùng mà dự án tác động lên, phạm vi có thể là tổng thể, toàn cục hay chỉ cục bộ, việc xác định phạm vi là tùy thuộc vào ngữ cảnh cụ thể của bài toán cần giải quyết. Điều quan trọng ở đây là giữa cơ quan chủ quản và nhóm phát triển dự án phải có sự thỏa thuận minh bạch, rõ ràng.

Trong một cơ quan, dù lớn hay nhỏ, hệ thống quản lý thường bao gồm nhiều hệ con (quản lý nhân sự và tiền lương, quản lý khách hàng,...), như vậy cần thiết phải có một nghiên cứu tổng thể về chiến lược tin học hoá công tác quản lý của cơ quan, qua đó xác định các hệ thống con, cùng các loại giao diện giữa chúng, trên cơ sở ấy sẽ quyết định chọn một số hệ con đưa vào phạm vi giải quyết của dự án đang được đề cập, cũng cần lưu ý rằng phạm vi của dự án cũng có thể điều chỉnh đôi chút trong các bước của giai đoạn phân tích và thiết kế hệ thống.

Các hạn chế đối với dự án, do cơ quan chủ quản đề xuất tùy theo khả năng của mình, cũng cần được trao đổi kỹ. Thường có ba loại hạn chế:

- Nhân lực: Số người tham gia dự án, trình độ, năng lực của những người sử dụng hệ thống sau này
- Thiết bị, kỹ thuật: Các khả năng về thiết bị, vật tư và kỹ thuật có thể đáp ứng
- Tài chính: Mức độ đầu tư và chi phí cho dự án

2. Xác định các mục tiêu và ưu tiên cho dự án

Thông thường thì một hệ thống thông tin được xây dựng là nhằm vào các mục tiêu sau:

- Mang lại lợi ích nghiệp vụ: Tăng cường khả năng xử lý, đáp ứng yêu cầu nghiệp vụ một cách chính xác, tin cậy, an toàn, bí mật,...

- Mang lại lợi ích kinh tế: Giảm biên chế cán bộ, giảm chi phí hoạt động nhưng lại tăng thu nhập, hoàn vốn nhanh.
- Mang lại lợi ích sử dụng: Nhanh chóng, thuận tiện
- Khắc phục yếu kém của hệ thống cũ, hỗ trợ chiến lược phát triển lâu dài, đáp ứng các ưu tiên và ràng buộc.
- Đóng góp thiết thực vào công cuộc phát triển tiềm lực và cơ sở hạ tầng về CNTT

Bên cạnh các mục tiêu cần đạt được cũng cần phải ghi nhận một số ưu tiên mà sau này khi phát triển hệ thống, ta luôn phải để ý đến khi phải lựa chọn các quyết định. Các ưu tiên đó hoặc là do phía cơ quan chủ quản và người dùng đề xuất hoặc là do một hoàn cảnh khách quan nào đó tạo ra.

Quay lại ví dụ cung ứng vật tư, thì hệ cần xây dựng phải đáp ứng hai mục tiêu sau:

- Rút ngắn thời gian giữa dự trù và phát hàng
- Loại trừ các sai sót trong việc xử lý các đơn hàng

Mục tiêu là thế nhưng quan trọng hơn là phải đưa ra các biện pháp cụ thể để đạt được mục tiêu đó. Về nguyên tắc có thể có nhiều giải pháp, trong trường hợp cụ thể này các biện pháp sau đây là khả thi:

- Thêm chức năng quản lý kho hàng dự trữ
- Cải tiến phương pháp để xác định địa chỉ phát hàng
- Cải tiến lại phương pháp kiểm tra hàng về, hàng đặt, tiền trả để tránh sai sót
- Tự động hoá công tác đối chiếu

Bên cạnh đó cũng cần ghi nhận một ưu tiên do phía cơ quan chủ quản mong muốn là tận dụng phần cứng và phần mềm đã có trên hai máy tính đó.

3. Phác hoạ và nghiên cứu tính khả thi

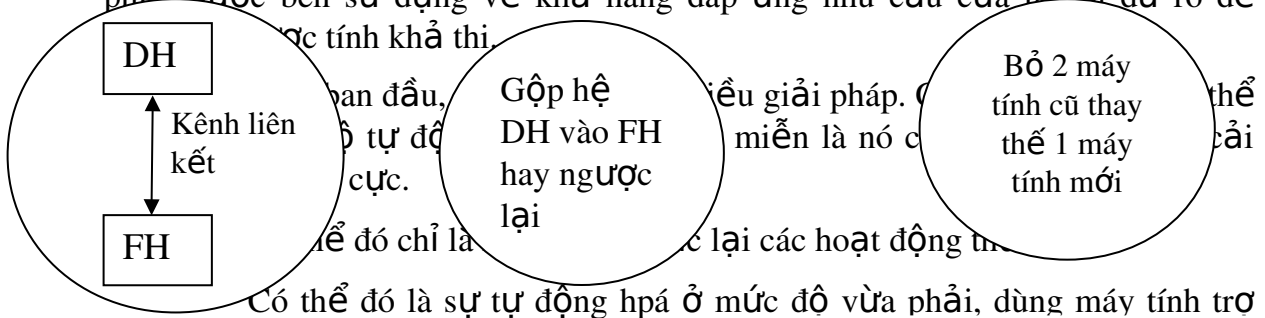
Việc đưa ra giải pháp vào giai đoạn này có phần hơi sớm vì ở giai đoạn này mới chỉ tìm hiểu sơ bộ, chưa có một sự phân tích sâu sắc và một sự cân nhắc kỹ lưỡng nhưng việc đó không thể thoái thác vì các lý do sau:

- Không thể để bên người dùng chờ đợi hơn, một khi họ sắp phải đầu tư một khoản tiền, mà nên để họ nhìn thấy một số giải pháp đầy triển vọng và khả thi.
- Mặt khác, đội ngũ phát triển dự án cũng cần có một sự xác định giải pháp như thế, bởi vì một giải pháp sẽ giúp ta dự đoán mức đầu tư, thấy được định hướng tốt cho cả quá trình xây dựng hệ thống tiếp theo.

Việc đưa ra một giải pháp tuy có hơi sớm, song có thể làm được vì đây chỉ là giải pháp thô, nó chỉ bao gồm các quyết định đại thể như:

- Chức năng chính của hệ thống, đầu vào, đầu ra, các biện pháp chính để đáp ứng nhu cầu của người dùng
- Kiến trúc tổng thể của hệ thống, bao gồm kiến trúc phần mềm (các hệ thống con, các mô đun chính,...) và kiến trúc phần cứng (mạng, các máy tính, các thiết bị,...).

Cái chủ yếu của giải pháp là, tuy chưa chi tiết, song nó phải đủ rõ để thuyết phục được bên sử dụng về khả năng đáp ứng nhu cầu của nó và đủ rõ để

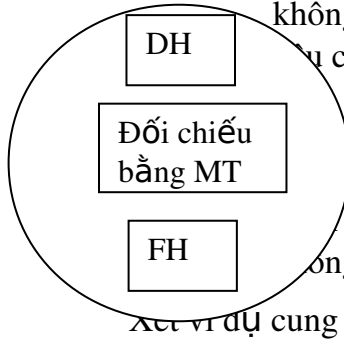


Giải pháp 1: Thiết lập một kênh liên kết giữa hai máy tính, loại bỏ tổ: Đối chiếu: Giải pháp này không khả thi về mặt kỹ thuật vì các MT hiện có tương thích không để nối ghép

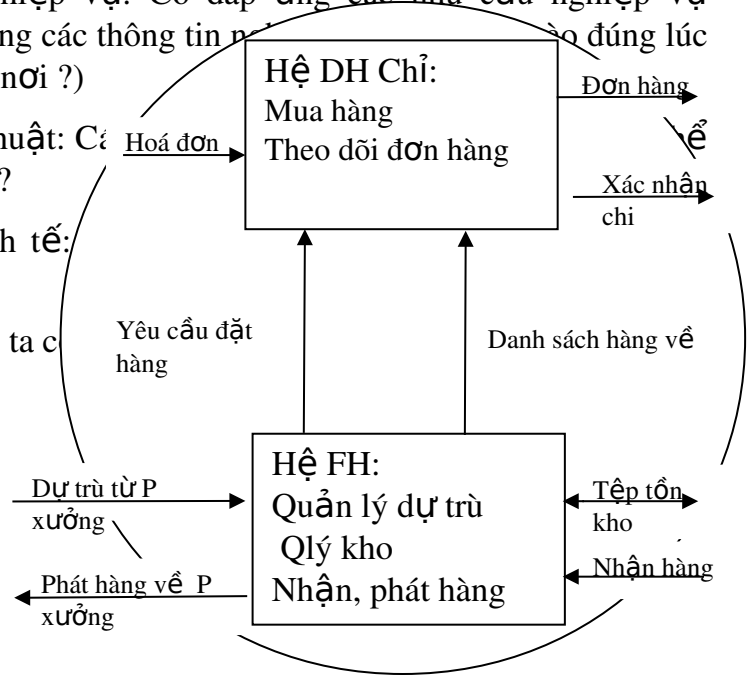
Giải pháp 2: Phải viết lại một nửa chương trình, vậy sẽ tốn kém một số giai đoạn cơ sở đó cùng với bên chọn ra một giải pháp phù hợp nhất, tính khả thi có thể được đánh giá về các mặt sau:

Giải pháp 3: Bỏ hai MT hiện có đưa bài toán vào TTMT chung cho cả cơ quan, giải pháp này sẽ khả thi nếu chúng ta có ý định dùng hệ CUVT như là một hệ con của một hệ thống lớn của cả cơ quan, tuy nhiên biện pháp này đòi hỏi phải viết lại các chương trình gây tốn kém ứng dụng đã có sẵn, dẫn đến gây lãng phí nhiều mặt

- Tính khả thi về nghiệp vụ: Có đáp ứng các nhu cầu nghiệp vụ không? (cung cấp đúng các thông tin nghiệp vụ đúng lúc đúng nơi và đến đúng nơi?)



- Tính khả thi về kỹ thuật: Có thực hiện được không?
- Tính khả thi về kinh tế: Có hiệu quả không?



Giải pháp 4: Gặp khó khăn như giải pháp 1

Giải pháp 5: Có tính khả thi cao nhất vì hai MT cũng như các ứng dụng sẵn có được sử dụng triệt để (thỏa mãn các ưu tiên mà cơ quan đề xuất). Việc đưa kho hàng dự trữ cũng như việc chuyển đổi các chức năng giữa hai hệ DH và FH thực sự đáp ứng các mục tiêu mà lại không gây tốn kém nhiều. Vậy giải pháp này sẽ là lựa chọn của chúng ta.

4. Lập dự trù và kế hoạch triển khai:

Sau khi có được một sự hiểu biết đầy đủ về các vấn đề, các yêu cầu của người dùng để có thể hình dung được một cách toàn diện về các vấn đề của dự án thì đây là thời điểm cần vạch ra một kế hoạch rõ ràng và chi tiết.

Lưu ý rằng ngay từ giai đoạn này phải bắt đầu ngay các hoạt động về quản lý dự án. Xem xét, báo cáo và tư liệu hoá cũng như tiếp tục tiến hành các hoạt động đó cho đến khi kết thúc dự án.

❖ *Hợp đồng triển khai dự án:*

Một khi dự án xây dựng hệ thống mới đã được khẳng định, cần thiết phải có một hợp đồng giữa hai bên A và B để chốt lại những nội dung chủ yếu của dự án đó. Hợp đồng nên viết thành văn bản, chứa đựng những nội dung chủ yếu sau:

- Bài toán đặt ra và các nhu cầu về thông tin
 - Phạm vi và hạn chế
 - Mục tiêu và ưu tiên
 - Giải pháp và tính khả thi
 - Dự trù thiết bị và kinh phí
 - Phân công trách nhiệm và nhân sự
 - Phương pháp và tiến trình triển khai
- Các nội dung này đã được đề cập ở những mục trước, nay được đưa vào hợp đồng như sự chấp nhận của cả hai bên
- Ba nội dung này cần được xem xét thêm sau đây:

❖ *Dự trù thiết bị và kinh phí:*

Mặc dù đang ở giai đoạn đầu của dự án, nhưng cũng cần phải có một quỹ thời gian nhất định cho công việc này, cho nên cũng không phải là sớm khi xem xét đến thiết bị và kinh phí. Việc dự trù thiết bị sẽ dựa trên các căn cứ sau đây:

- Khối lượng dữ liệu cần lưu trữ lâu dài
- Các dạng làm việc (xử lý theo mẻ, làm việc từ xa,...)
- Số lượng người dùng
- Khối lượng thông tin thu thập
- Khối lượng thông tin cần kết xuất, các tài liệu in ra

Trên các căn cứ này, ta có thể xác định được cấu hình của thiết bị, bao gồm:

- Máy mạng hay máy lẻ
- Các thể loại thiết bị ngoại vi, có lưu ý đến các thiết bị đặc chủng
- Các đường truyền
- Các phần mềm như: Hệ điều hành, hệ QTCSDL,...

Việc mua các thiết bị nên chú ý các mặt sau

- Hướng dẫn sử dụng
- Bảo hành

- Chất lượng và giá cả
- Điều kiện giao hàng, bảo hành

Bên cạnh kinh phí dành cho thiết bị, cần phải tính đến kinh phí cho quá trình triển khai dự án. Kinh phí này do hai bên thỏa thuận và dựa vào các yếu tố sau:

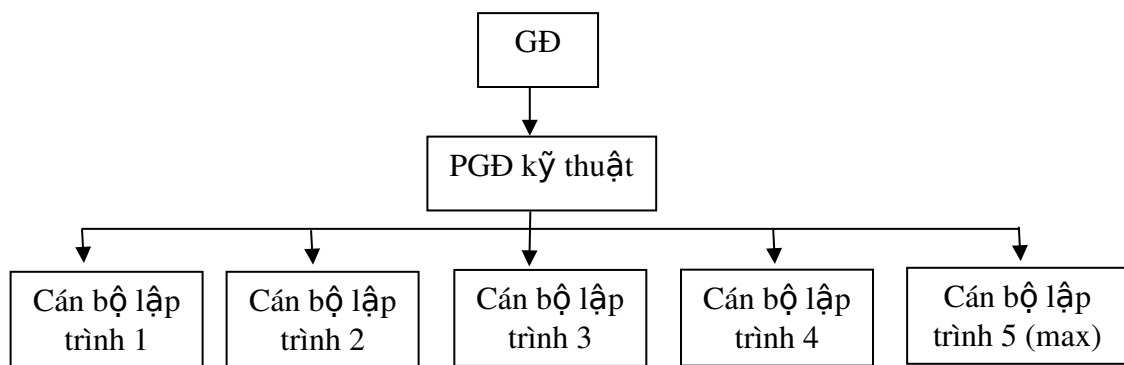
- Khối lượng công việc và nhân sự của các bước
- Chất lượng sản phẩm, thời hạn hoàn thành và mức độ bảo hành sản phẩm.

Kinh phí thường được phân bổ cho các hạng mục, các giai đoạn phát triển ứng dụng

❖ *Phân công trách nhiệm và nhân sự dự án:*

Nhân sự dự án trước hết là tổ chức dự án, cần phải biết chính xác ai làm việc gì, ở đâu và khi nào?

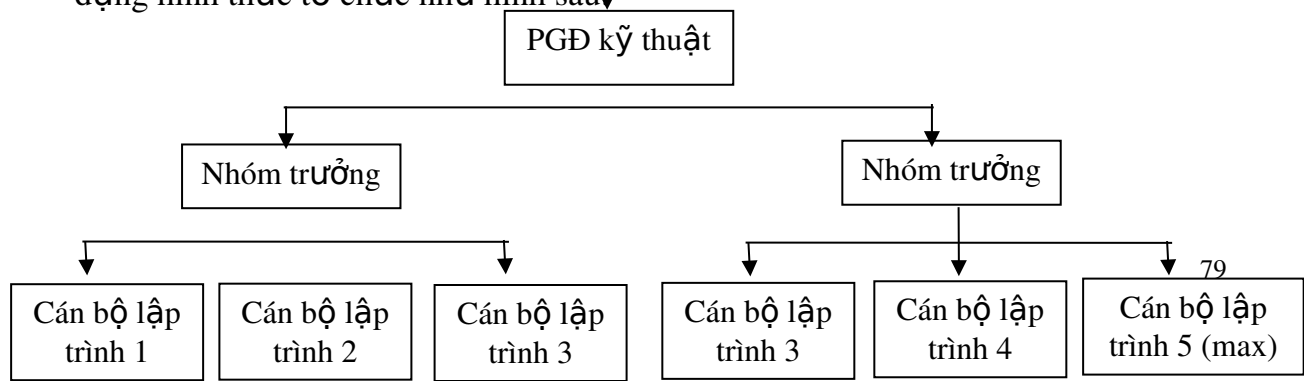
- **Đối với các dự án qui mô vừa và nhỏ:**



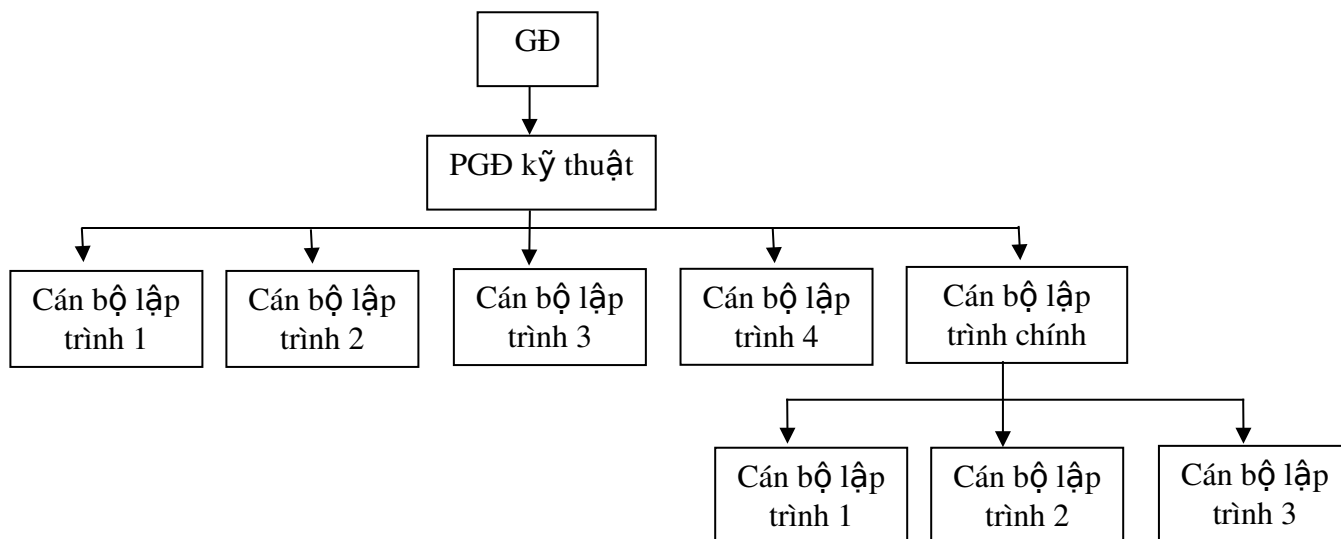
Mỗi người trong tổ có công việc riêng của mình. Công việc của các cán bộ lập trình, như tên đã gọi, là viết chương trình phần mềm. PGĐ kỹ thuật giám sát, chỉ đạo tất cả các công việc thuộc về kỹ thuật, giải quyết các vấn đề liên quan đến hệ thống, đồng thời chịu trách nhiệm về chất lượng sản phẩm. Trên cùng, GD ban quản lý dự án lãnh đạo về mặt quản lý, đảm bảo liên hệ giữa dự án với bên ngoài, đồng thời chịu trách nhiệm về kế hoạch kiểm soát

- **Đối với các dự án qui mô lớn:**

Người ta nhận thấy rằng, mỗi GD và nhóm kỹ thuật thường theo dõi sát được 5 cán bộ lập trình. Vì vậy, các dự án qui mô lớn hơn, có thể áp dụng hình thức tổ chức như hình sau



Hoặc:



Câu hỏi và bài tập:

1. Nêu mục đích và yêu cầu của giai đoạn khảo sát và xác lập dự án
2. Các hình thức tiến hành trong khi tìm hiểu và đánh giá hiện trạng ? Hãy so sánh các hình thức đó
3. Hãy thực tập việc khảo sát các hệ thống tin thường gặp như:
 - Hệ thống tin quản lý thư viện
 - Hệ thống tin quản lý nhân sự
 - Hệ thống tin quản lý vật tư
 - Hệ thống tin quản lý học tập
 - Hệ thống tin quản lý khách sạn
 - Hệ thống tin quản lý tuyển sinh

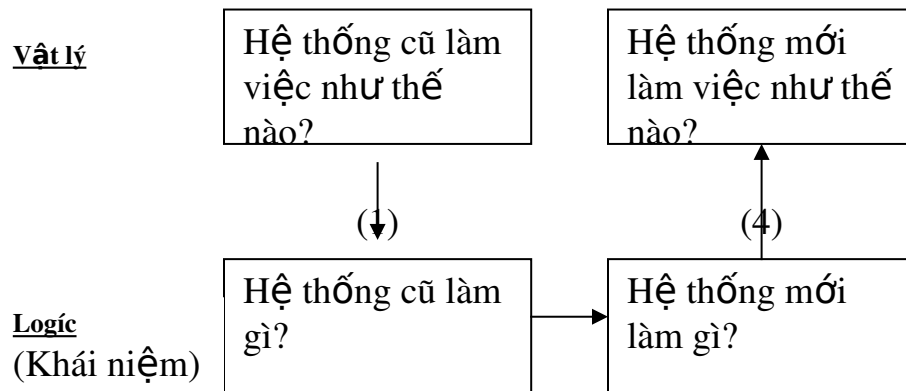
CHƯƠNG VI: PHÂN TÍCH HỆ THỐNG

I. Đại cương về phân tích hệ thống:

1. Mục đích:

Phân tích: Chính là khảo sát, nhận diện và phân lập các thành phần của một đối tượng phức hợp, đồng thời chỉ ra các mối liên quan của chúng. Công việc phân tích này nhằm là rõ cấu trúc hệ thống về hai phương diện: Xử lý và dữ liệu

Về nghĩa hẹp: Giai đoạn phân tích tức là nghiên cứu sâu hơn, chi tiết hơn về cả mặt chức năng cũng như dữ liệu. Tức là từ HT cũ ta hình thành hệ thống mới ở mức Logic. Do vậy giai đoạn này còn được gọi là giai đoạn thiết kế logic



2. Hai phương diện của phân tích hệ thống

Như đã biết một hệ thống thông tin nêu lược bỏ con người và phương tiện, phương pháp thì chỉ còn lại xử lý và dữ liệu. Vì vậy khi tiến hành phân tích hệ thống người ta luôn đề cập đến hai mặt sau:

- Phân tích hệ thống về xử lý
- Phân tích hệ thống về dữ liệu

Hai quá trình này có thể tách riêng vì dữ liệu thường tương đối độc lập với xử lý, vì xử lý có thể thay đổi theo yêu cầu, trong khi đó dữ liệu ít thay đổi về cấu trúc, nói cách khác dữ liệu có cấu trúc nội tại không liên quan đến xử lý.

II. Phân tích hệ thống về dữ liệu:

1. Đường lối thực hiện:

- Phân tích hệ thống từ trên xuống: Phân tích từ đại thể đến chi tiết, thể hiện ở cách phân mức BPC và BLD
- Chuyển dịch từ mức vật lý sang mức Logic
- Chuyển từ BLD logic của HT cũ sang BLD logic của HT mới

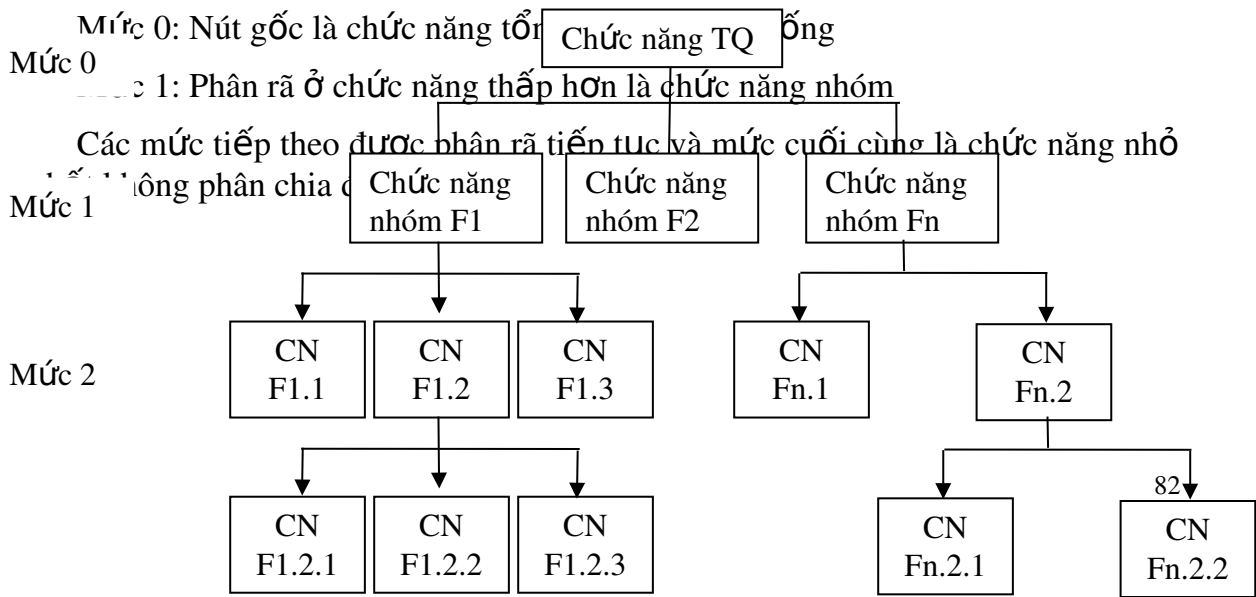
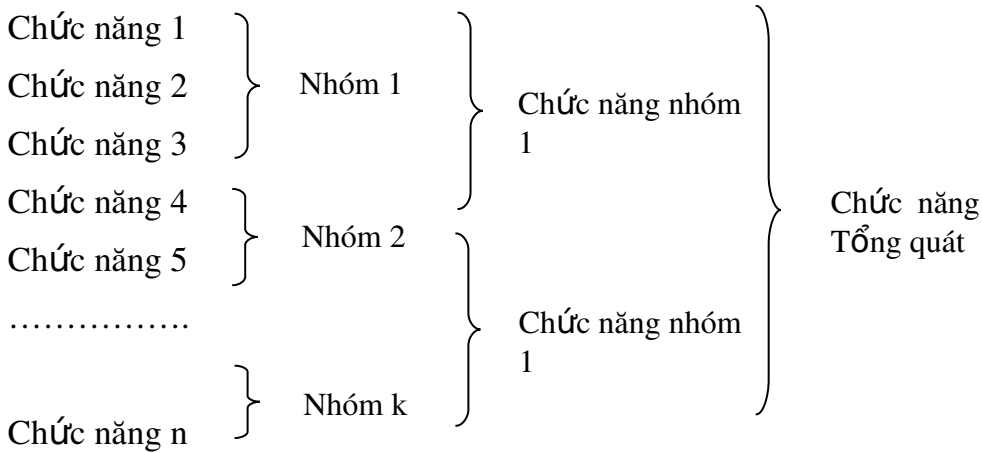
2. Phân tích hệ thống từ trên xuống :

Là sự phân tích từ tổng thể đi tới chi tiết bằng nhiều mức BPC, BLD

a. Biểu đồ phân cấp chức năng (BPC) :

Đây là biểu đồ mô tả tĩnh, bằng kỹ thuật phân mức ta xây dựng biểu đồ dưới dạng cây, trong đó mỗi nút tương ứng một chức năng

Tại giai đoạn khảo sát sơ bộ hệ thống ta liệt kê các chức năng của hệ thống. Các chức năng này phân theo nhóm chức năng và xếp gần nhau, chúng được đánh số theo thứ tự và theo nhóm dạng phân tích như sau:



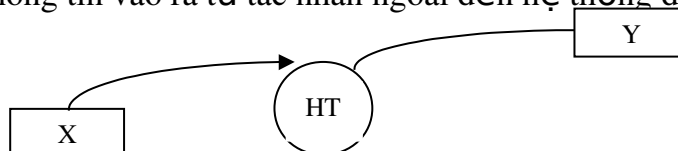
b. Biểu đồ luồng dữ liệu (BLD):

Với BLD ta có thể diễn tả hệ thống bằng cách theo dõi sự dịch chuyển, xử lý dữ liệu.

Trong phương pháp này người ta sử dụng kỹ thuật phân mức, hay còn gọi là “phân tích từ trên xuống”, qui trình phân mức tiến hành phân tích chức năng hệ thống bằng cách đi dần từ mô tả đại thể đến những mô tả chi tiết thông qua nhiều mức. Sự dịch chuyển tới một mức tiếp theo thực ra là một quá trình phân rã.

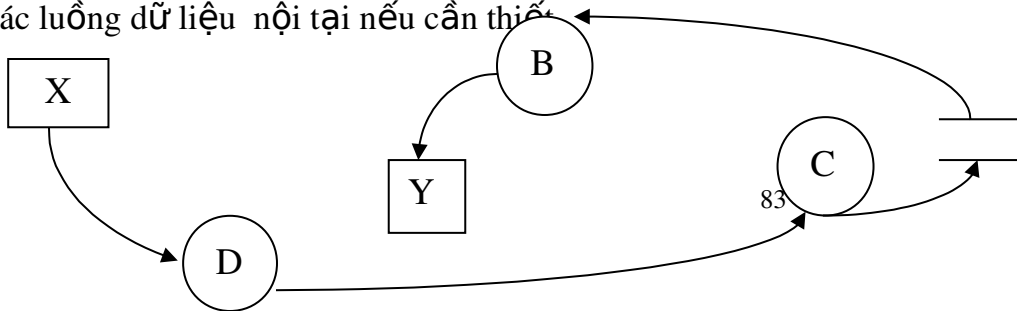
Với BLD thì quá trình phân tích trên xuống lại là quá trình thành lập dần dần các BLD diễn tả các chức năng hệ thống theo từng mức, mỗi mức là một tập hợp các BLD như thế còn gọi là phương pháp phân tích động.

- ❖ Mức đầu tiên (mức 0): Gọi là mức khung cảnh, đây là mô hình hệ thống ở mức tổng quát nhất, ta xem cả hệ thống như một chức năng. Tại mức này hệ thống chỉ có duy nhất một chức năng, các nhân ngoài và đồng thời các luồng thông tin vào ra từ tác nhân ngoài đến hệ thống được xác định

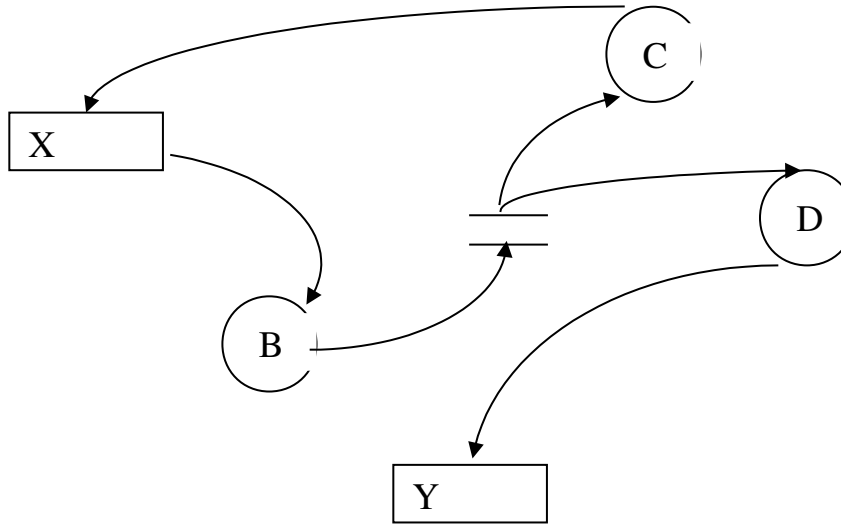


Biểu đồ mức khung cảnh

- ❖ Mức một (mức đỉnh): Được phân rã từ BLD mức ngữ cảnh với các chức năng tương ứng ở mức 2 của BPC. Các nguyên tắc phân rã:
 - Các luồng dữ liệu được bảo toàn
 - Các tác nhân ngoài được bảo toàn
 - Có thể xuất hiện các kho dữ liệu
 - Bổ sung thêm các luồng dữ liệu nội tại nếu cần thiết



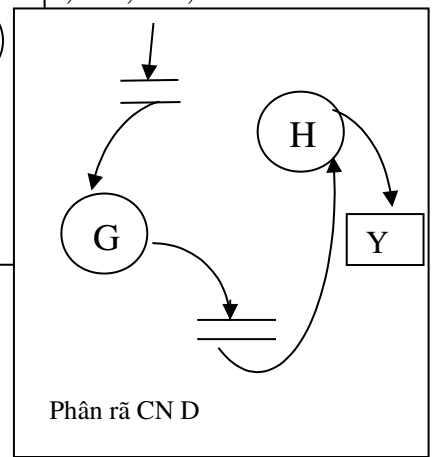
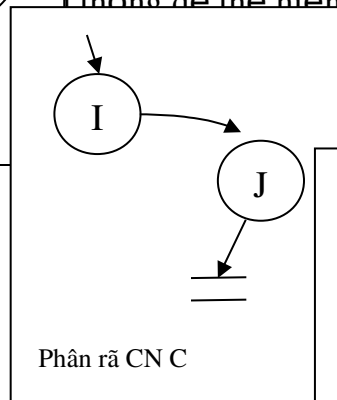
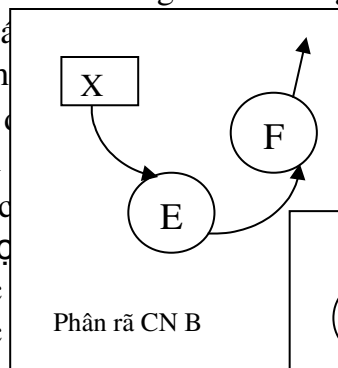
Hoặc



❖ Mức 2: Gọi là BLD mức dưới đỉnh và được phân rã từ BLD mức đỉnh tương ứng với mức 2,3,... của BPC, theo nguyên tắc sau:

- Mỗi chức năng ở mức đỉnh được phân rã thành một BLD ở mức dưới đỉnh (đối với HT đơn giản thì có thể ghép lại thành một biểu đồ BLD dưới đỉnh)
- Tác nhân ngoài xuất hiện đầy đủ ở mức khung cảnh, được bảo tồn ở các mức dưới không thể bổ sung thêm

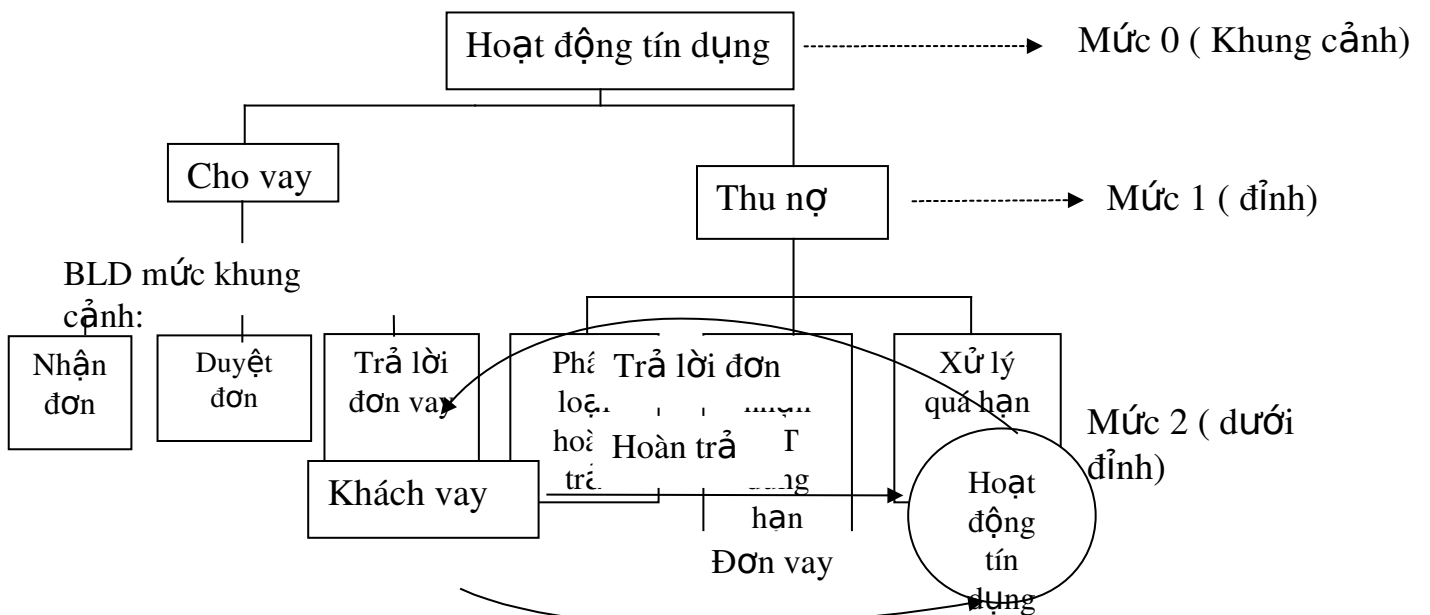
- Vẽ lại các con thích hợp, những phải vào hay ra ở chức năng các luồng thông dữ liệu nội bộ từ mức đỉnh và dẫn xuất hiện thêm
- Các kho theo nhu
- Các chức tương hợ
- . Ở mức
- . Ở mức



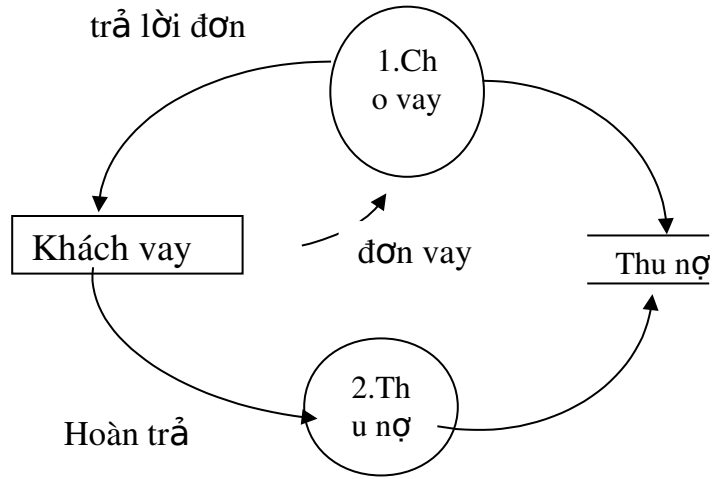
...; 2.1, 2.2,...

Thí dụ: Hoạt động của một cơ sở tín dụng

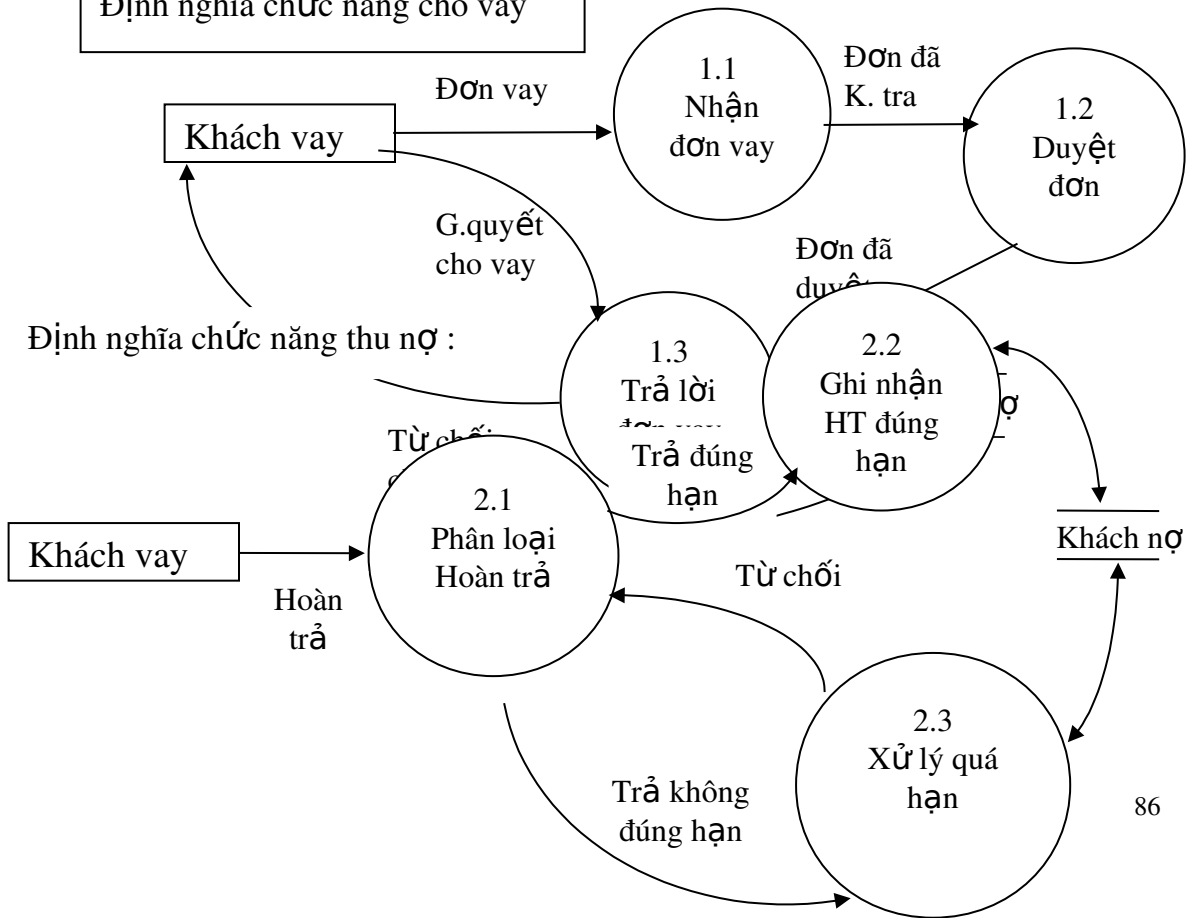
❖ BPC:



BLD mức đỉnh:



BLD(mức dưới đỉnh)
Định nghĩa chức năng cho vay



3. Chuyển từ BLD vật lý sang BLD logic:

Như chúng ta đã biết, mục đích của phân tích là để hình thành được một mô hình logic của hệ thống. Một trong những ưu điểm của phân tích có cấu trúc là có thể tiếp cận HT ở hai mức Vật lý và logic.

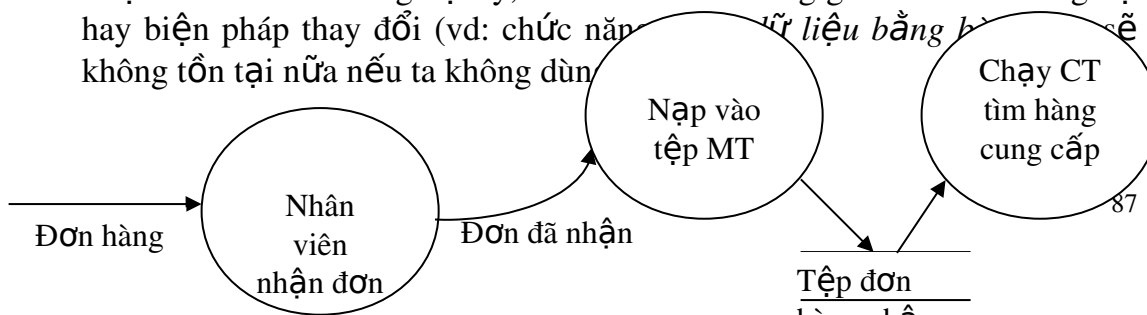
Quá trình phân rã BLD ở các mức chỉ có thể dẫn chúng ta đến những mô tả chi tiết, nhất là khi mô tả đó lại dựa vào những kết quả thu được thông qua khảo sát thực tế hiện trạng của một HT đang hoạt động. Vì vậy BLD này là sự trả lời cho câu hỏi: *Làm như thế nào? Ai làm? bằng phương tiện gì? ở đâu?...* Theo định nghĩa đây chính là mức vật lý, tuy phức tạp nhưng lại cần thiết bởi vì đó là sự khởi đầu của quá trình phát triển HT.

Với BLD ở mức vật lý, chúng ta khó mà hiểu được bản chất của hệ thống, điều này có nghĩa là cần phải chuyển sang mô tả HT ở mức logic.

Thông thường thì BLD mức vật lý dùng trong khảo sát HT (cũ) và trong thiết kế HT mới, còn BLD logic được dùng cho việc tìm hiểu (phân tích) các yêu cầu của HT cũ cũng như HT mới

BLD mức vật lý chứa đựng trong nó rất nhiều các yếu tố vật lý, thường là hai loại yếu tố vật lý sau:

- ❖ Loại 1: Các yếu tố vật lý xuất hiện tường minh thông qua ngôn ngữ hay hình vẽ trong biểu đồ như:
 - Các phương tiện, phương thức được dùng để thực hiện chức năng (máy tính, bàn phím, xử lý thủ công,...)
 - Các giá mang thông tin (đĩa từ, sổ sách,...)
 - Các cá nhân thực hiện chức năng (kế toán, giám đốc, thư ký,...)
- ❖ Loại 2: Các chức năng vật lý, đó là các chức năng gắn liền với công cụ hay biện pháp thay đổi (vd: chức năng *đổi dữ liệu bằng bàn phím* sẽ không tồn tại nữa nếu ta không dùng



Ví dụ:

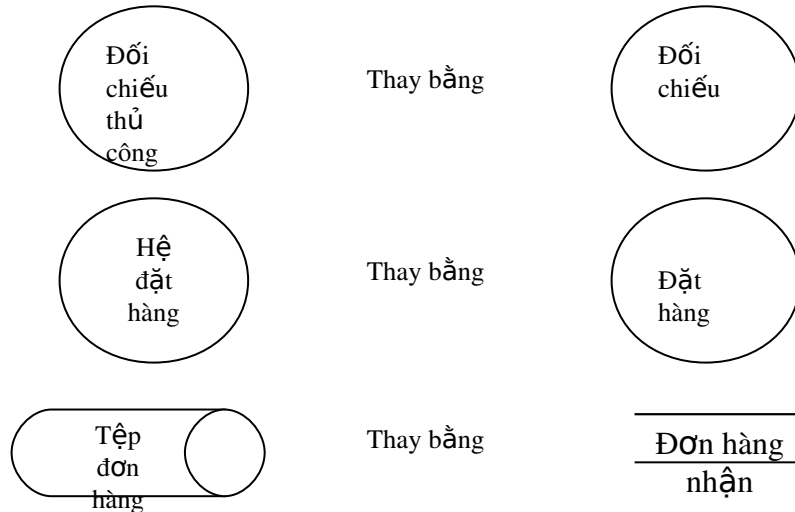
Để thấy các yếu tố xuất hiện tường minh là: Nhân viên, máy tính, tệp,...

Các yếu tố vật lý này xen lẫn vào BLD, có thể làm lu mờ bản chất của HT, cho nên loại chúng ra khỏi biểu đồ.

Xuất phát từ BLD ở mức đỉnh:

Để loại bỏ các yếu tố vật lý loại 1, hãy loại ra khỏi biểu đồ các ngôn từ, hình vẽ thể hiện phương tiện, giá mang thông tin hay các tác nhân, chỉ giữ lại sự diễn tả nội dung của chức năng hay thông tin mà thôi.

Ví dụ:



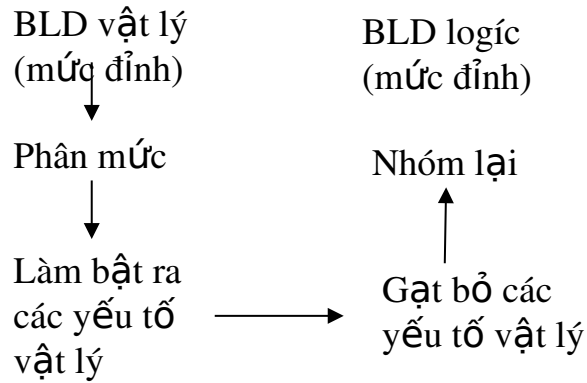
Để loại bỏ các yếu tố vật lý loại 1, thay bằng Thanh toán → lộ diện trong biểu đồ. Thường thì các chức năng vật lý là nhỏ, không xuất hiện ở các BLD mức trên, vì vậy ta phải triển khai xuống các mức dưới. Quá trình phân rã này sẽ tách các chức năng vật lý ra khỏi các chức năng logic. Từ đó sẽ loại được các chức năng vật lý khỏi biểu đồ.

Sau khi loại hết các chức năng vật lý, các chức năng còn lại là các chức năng logic ở một mức nào đó, tổ chức lại biểu đồ, từ dưới lên, xuất phát từ các chức năng logic này ta sẽ thu được BLD mức đỉnh theo các bước như sau:

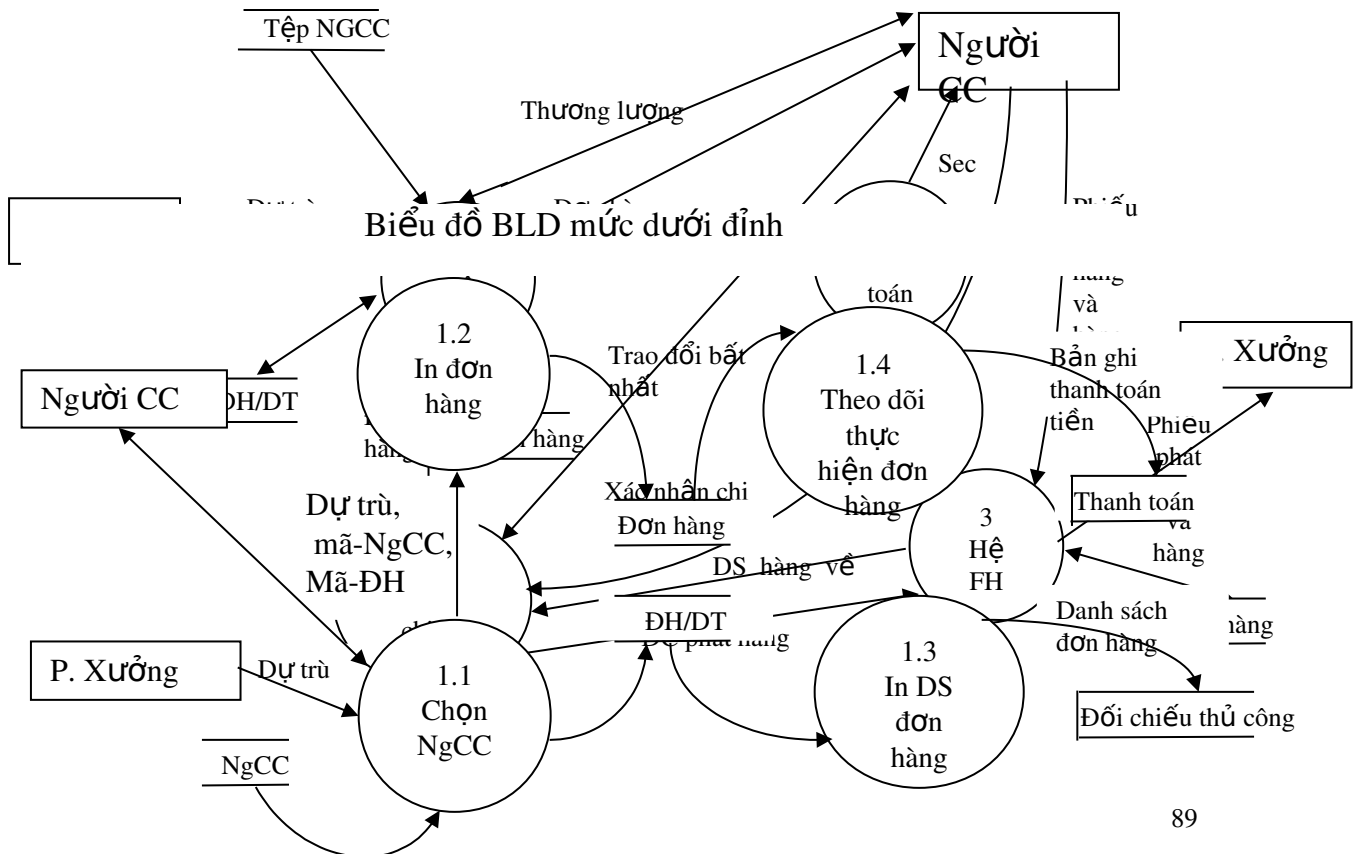
- Trước hết, do các chức năng vật lý bị loại bỏ, một số luồng dữ liệu bị đứt rời, hãy tìm cách nối chúng lại một cách hợp lý
- Tiếp đó, nhìn nhận lại nội dung các chức năng, tìm cách gom nhóm các chức năng gần gũi nhau trong một mục đích xử lý nào đó vào một chức năng lớn, quá trình này có thể lặp lại nhiều lần cho tới khi ta thu được BLD mức đỉnh

Nói tóm lại, qui trình chuyển đổi BLD mức vật lý sang BLD mức logic diễn ra qua sơ đồ sau:

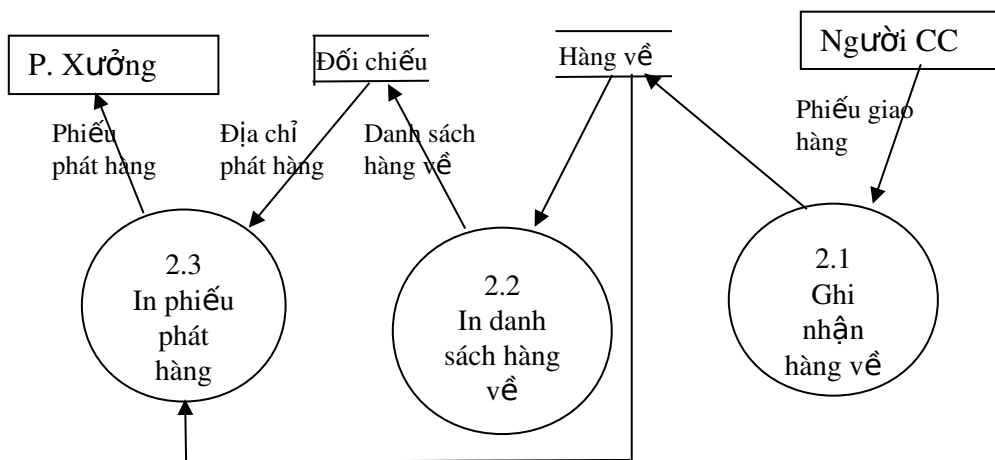
Sơ đồ chuyển BLD vật lý sang BLD logic



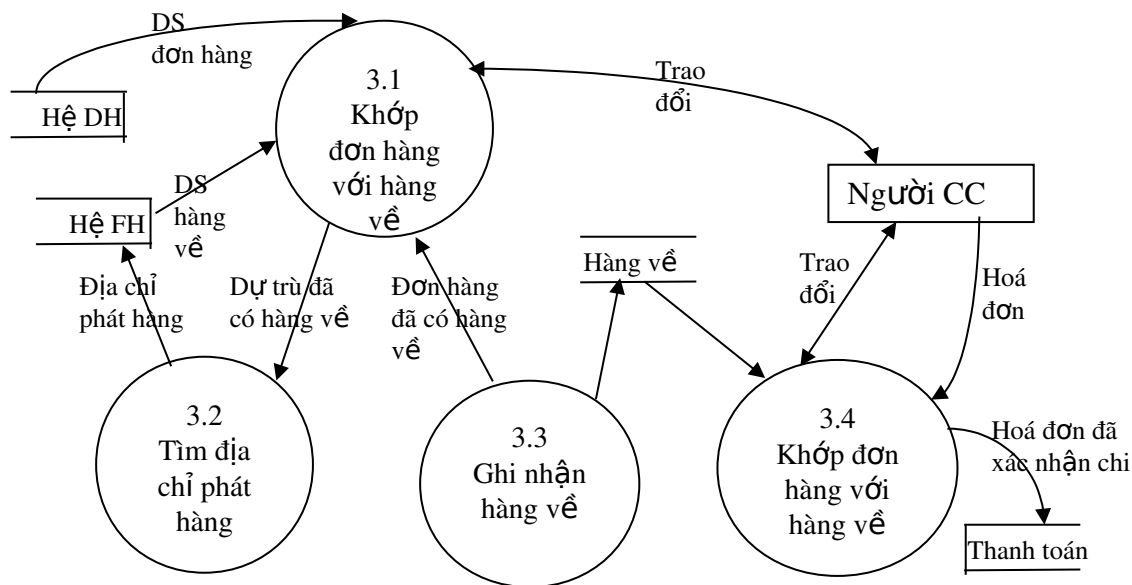
Thí dụ: Xuất phát từ BLD mức đỉnh của ví dụ tổng quát



Phân rã chức năng 1 (HỆ DH):



Phân rã chức năng 2 (HỆ FH):



Phân rã chức năng 3 (Đối chiếu thủ công):

Trước hết ta tiến hành loại bỏ các yếu tố vật lý loại 1, trong trường hợp này là loại bỏ từ, cụm từ: “hệ”, “thủ công”, thay bằng từ “In” bằng từ “làm”, “Séc” bằng từ “thanh toán”.

Tiếp đến là các yếu tố loại 2, các chức năng 1.3 và 2.2 là các chức năng vật lý, vì chúng chỉ làm nhiệm vụ đưa các thông tin từ các tệp ĐH/DT, đơn hàng, hàng về lên giấy

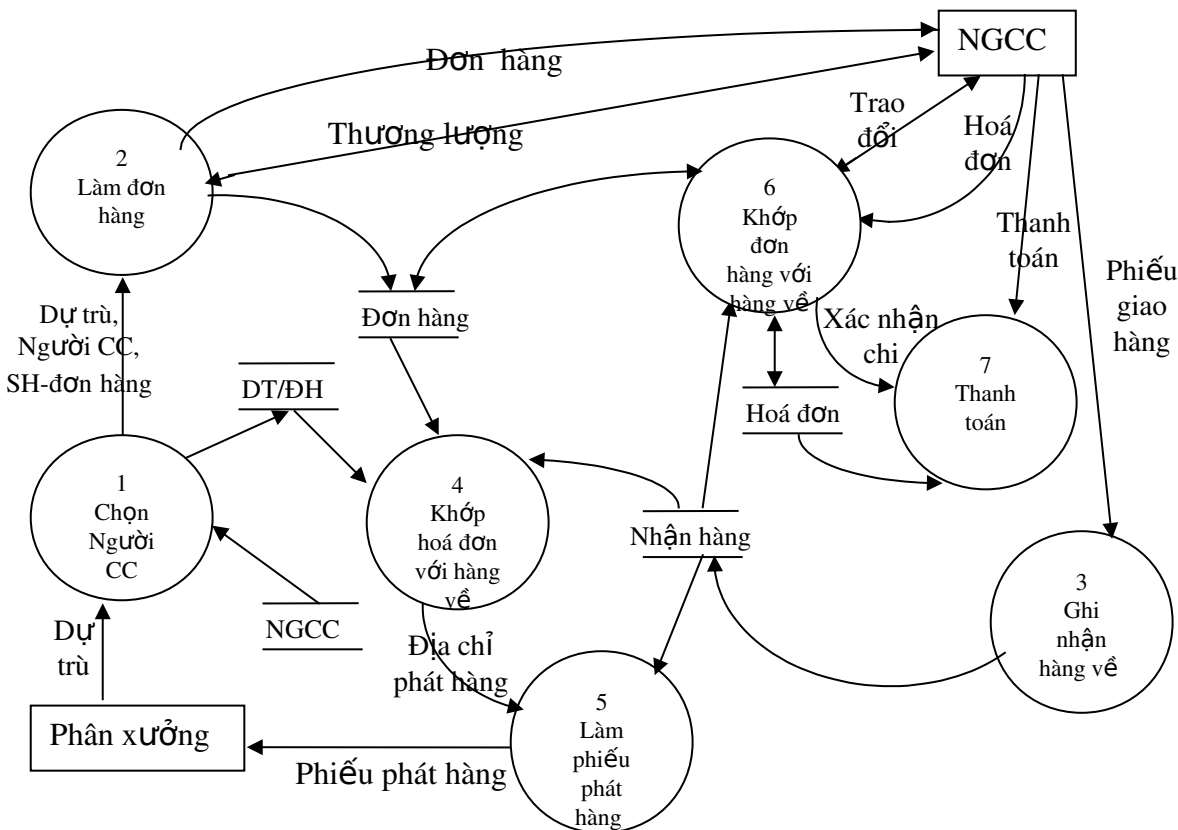
Với các chức năng còn lại ta tiến hành gom nhóm chúng theo sự gần gũi về mục đích xử lý:

- Chức năng 1.1 đánh số lại là 1 (chọn người cung cấp)
- Chức năng 1.2 trở thành 2 (Làm đơn hàng)
- Chức năng 2.1 trở thành 3 (Ghi nhận hàng về)
- Chức năng 3.1, 3.2, 3.3 gộp thành 4 (khớp đơn hàng với hàng về)
- Chức năng 2.3 trở thành 5 (Làm phiếu phát hàng)
- Chức năng 3.4 và 1.4 gộp thành 6 (khớp hoá đơn với hàng về)
- Chức năng 4 trở thành chức năng 7

Sau cùng chúng ta thu được BLD logic mức đỉnh như sau, trong đó một số luồng dữ liệu được vẽ lại cho phép các chức năng lấy trực tiếp thông tin từ các kho dữ liệu

Để ý rằng trong BLD mới thu được chỉ có các chức năng gắn với mục đích xử lý mới được giữ lại, vì vậy BLD này phản ánh rõ hơn bản chất của HT

BLD logic ta thu được:



4. Chuyển từ BLD logic của HT cũ sang BLD logic của HT mới :

Thông thường, chúng ta xây dựng BLD logic của HT mới dựa vào BLD logic của HT hiện hành (cũ), qua các bước thêm, bớt hay điều chỉnh mà thôi.

Muốn vậy, xem xét lại các thiếu sót về hiện trạng trước đây đối với HT cũ, cũng như rà soát lại các mục tiêu và ưu tiên đã đề xuất đối với HT mới. Nhờ những phân tích chi tiết vừa tiến hành, nay có thể khẳng định lại, bổ sung hay sửa đổi các phê phán, mục tiêu và các ưu tiên đó.

Dựa vào các căn cứ này, đối chiếu với BLD logic của HT cũ, phát hiện những chỗ thiếu sót, dư thừa hay cần điều chỉnh lại. Khoanh các vùng cần thay đổi lại bằng đường cong khép kín đứt nét. Đối với từng vùng cần thay đổi này, để sắp xếp lại, ta tiến hành như sau:

- Xoá bỏ phần BLD bên trong vùng, song phải bảo toàn các luồng dữ liệu vào/ra
- Xác định các chức năng tổng quát mới của vùng thay đổi
- Thiết lập một trung tâm biến đổi mới thực hiện chức năng tổng quát nói trên, gồm các chức năng, các kho dữ liệu và các luồng thông tin vào/ra của vùng, đồng thời cũn có thêm luồng vào/ra mới.

Qua lại ví dụ hệ thống CUVT, như ta đã xét trong các mục trước, HT này có 3 nhược điểm sau:

- Thiếu kho hàng dự trữ các vật tư thiết yếu, để giải quyết ngay các mặt hàng thông dụng, khi một phân xưởng có yêu cầu
- Quy trình cung ứng vật tư quá lâu, nhiều sai sót do khâu đối chiếu thủ công.
- Tổn nhân lực

Hai nhược điểm sau đều bắt nguồn từ sự chia cắt các bộ phận, thông tin dư thừa nhiều trong HT, buộc phải đối chiếu thủ công vừa chậm, vừa không chính xác lại tốn tiền của. Bởi vậy BLD logic của HT cũ không còn thể hiện các nhược điểm của nó nữa, vì các yếu tố vật lý đã xoá bỏ hết rồi.

Trong khi đó nhược điểm đầu tiên là một thiếu sót logic, vậy cần phải lập một vùng thay đổi trên BLD logic của HT cũ. Vùng thay đổi này cắt ngang luồng dữ liệu dự trữ (từ phân xưởng đến chức năng 1). Với chức năng tổng quát của vùng đó là quản lý kho hàng dự trữ để đáp ứng ngay các vật tư thông dụng, khi có các yêu cầu từ các phân xưởng.

Cụ thể hơn nữa, ở đây cần một kho dữ liệu về các vật tư dự trữ, cùng với các chức năng duyệt dự trữ để xem xét khả năng kho hàng dự trữ có đáp ứng được một dự trữ không hay phải chuyển sang đặt hàng, mặt khác chính chức năng này sẽ chủ động đưa ra yêu cầu đặt hàng mỗi khi có một loạt vật tư nào đó sụt dưới ngưỡng qui định của nó.

Tóm lại:

Bổ sung các kho dữ liệu cần thiết

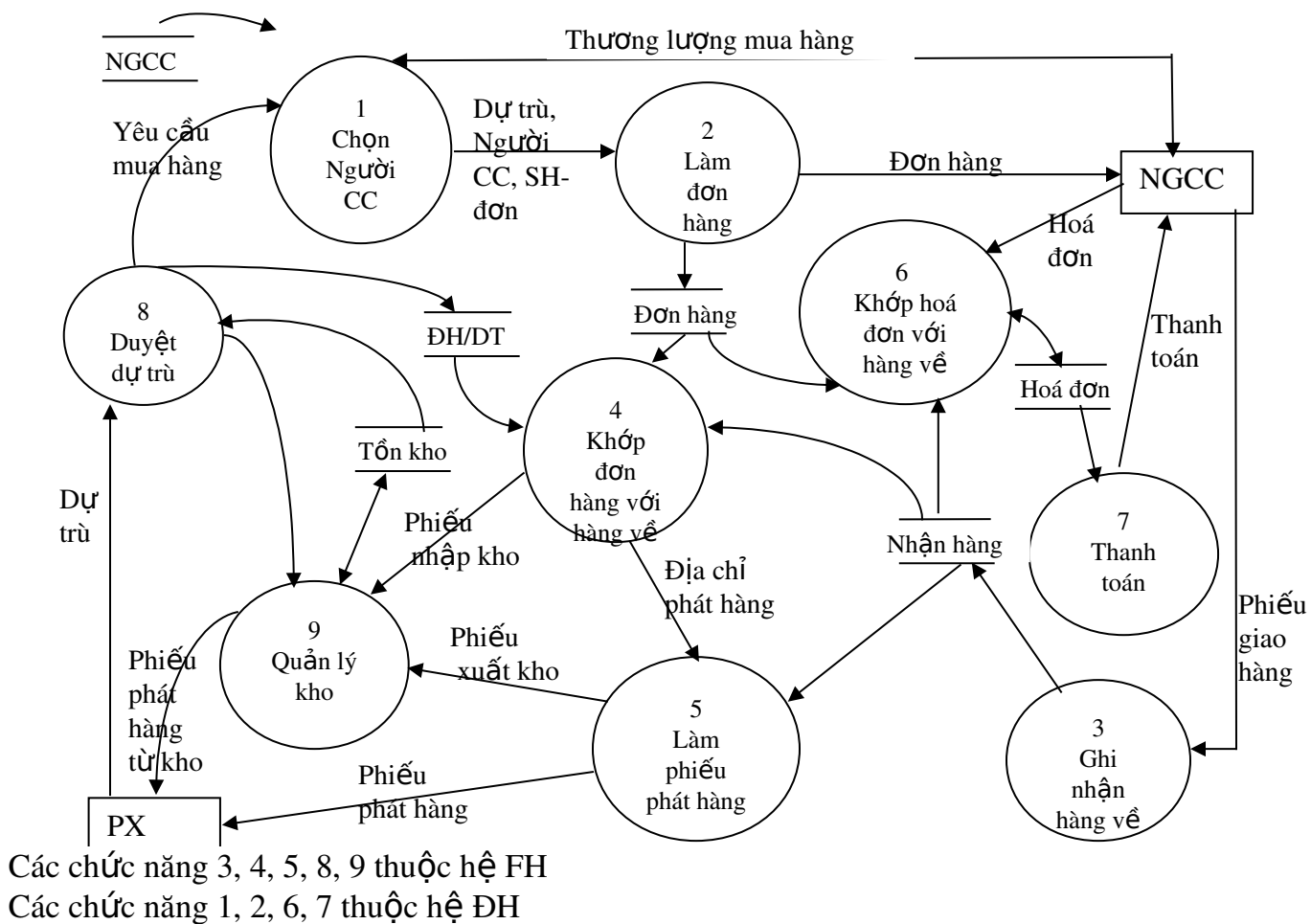
Theo dõi luồng TT vào và phải biệt lập chức năng xử lý sơ bộ luồng đó

Với mỗi luồng ra lập chức năng chuẩn bị TT ra

Khoanh vùng tập trung biến đổi (gồm một số chức năng)

Lập các TT dữ liệu nội bộ

Chức năng 8,9 là phần mới thêm vào



CHƯƠNG VII: PHÂN TÍCH HỆ THỐNG VỀ DỮ LIỆU

1. Đại cương:

Phân tích hệ thống bao gồm việc phân tích về xử lý và phân tích về dữ liệu. Chúng ta tách tạm thời việc phân tích dữ liệu vì dữ liệu có tính độc lập tương đối. Dữ liệu là đối tượng của xử lý.

a) Mục đích:

Nhằm phát hiện ra các dữ liệu cần có, mặt khác là tìm ra cấu trúc nội tại của các dữ liệu đó để rồi thiết lập được “ con đường “ truy nhập vào dữ liệu sau này.

Cụ thể: Lập được lược đồ dữ liệu hay còn gọi là biểu đồ cấu trúc dữ liệu(BCD). Trong biểu đồ này miêu tả dữ liệu của chúng ta gồm những thông tin gì và mối quan hệ giữa chúng.

Như đã biết dữ liệu ở đây được hiểu là những thông tin mà ta cần lưu trữ lâu dài để phục vụ các chức năng xử lý, có hai loại thông tin cần lưu trữ:

- Các thông tin phản ánh cấu trúc của cơ quan
- Các thông tin phản ánh các hoạt động kinh doanh của cơ quan

b) Yêu cầu:

- Tránh bỏ sót dữ liệu
- Tránh sự trùng lặp của dữ liệu.

Lưu ý rằng: Không phải ta bỏ qua mối liên quan tất yếu giữa các xử lý và dữ liệu, mà trong giai đoạn phân tích này ta chỉ tạm gác lại việc xem xét mối liên quan này cho tới giai đoạn thiết kế, khi mà ta cần thiết kế cơ sở dữ liệu

c) Phương pháp thực hiện: Dựa vào các mô hình sau:

- Mô hình TT/ LK : Phương pháp mang tính trực quan, đi từ trên xuống dưới, bằng cách xác định các thực thể, mối liên kết giữa chúng rồi đến các thuộc tính. Phương pháp này bao trùm được nhiều thông tin, tuy nhiên kết quả còn nhiều thông tin thừa.
- Mô hình quan hệ : Xuất phát từ danh sách các thuộc tính rồi đi đến các lược đồ quan hệ xuống dưới quan hệ, phương pháp này đi từ dưới lên. Kết quả là vừa đủ dữ liệu cho các xử lý.

2. Xây dựng BCD theo mô hình TT/ LK:

Trước tiên ta phải thu thập thông tin theo 3 yếu tố sau:

- Kiểu thực thể
- Kiểu liên kết
- Các thuộc tính

a) Phát hiện các kiểu thực thể:

Để phát hiện các kiểu thực thể ta thường tìm từ 3 nguồn sau:

- Các tài nguyên: Vật tư , nhân lực , nhà xưởng, môi trường.
- Các giao dịch: Đó là các thông tin đến từ môi trường bên ngoài nhằm kích động một chuỗi các hoạt động nào đó của hệ thống như: Hoá đơn, phiếu giao hàng, đơn hàng, ...
- Các thông tin đã được cấu trúc hoá: sổ sách, hồ sơ, các bảng biểu, thống kê, tổng hợp.

Chú ý: Cần dựa vào 2 điều kiện (có ích cho quản lý và phân biệt được) để chọn lựa

b) Phát hiện các kiểu liên kết: Trên thực tế có rất nhiều các liên kết giữa các thực thể nhưng ta chỉ ghi nhận những kiểu liên kết có ích cho quản lý đó chính là các liên kết giữa các thực thể mà ta đã phát hiện .

- Liên kết 1-1
- Liên kết 1- nhiều: Đó là các liên kết thường hay gặp nhất, thông qua các đường truy nhập, thường là được lần theo khoá có thể qua nhiều tệp khác nhau (khoá của thực thể 1 xuất hiện ở đầu nhiều của liên kết).
- Liên kết nhiều-nhiều: Có thể liên kết nhiều bên, nhiều phía (như thời khóa biểu).

c) Phát hiện thuộc tính:

- Khoá : Để nhận diện đối tượng (khoá đơn hoặc khoá kép).
- Mô tả: Chỉ xuất hiện ở mỗi kiểu thực thể, dùng để mô tả các đặc điểm của thực thể
- Kết nối : Thể hiện vai trò kết nối hai kiểu thực thể (là khoá của thực thể khác)

Thí dụ: xét ví dụ tổng hợp

❖ Các thực thể sơ bộ phát hiện là:

- Tài nguyên: Mặt hàng, người cung cấp, phân xưởng.
- Giao dịch :

Dự trữ và dòng dự trữ.

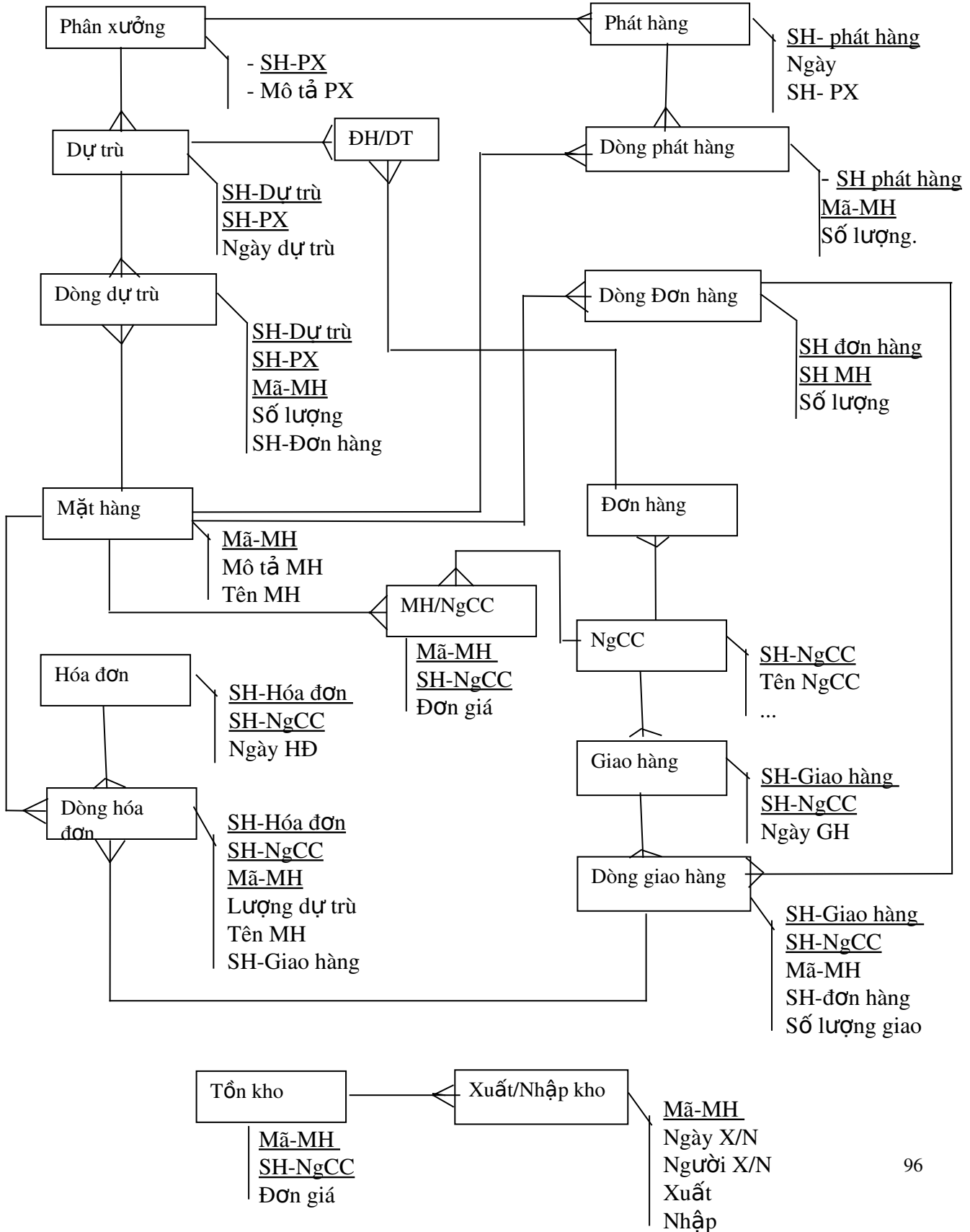
Đơn hàng và dòng đơn hàng.

Giao hàng và dòng giao hàng.

Phát hàng và dòng phát hàng.

Hóa đơn và dòng hóa đơn

- Thông tin có cấu trúc: ĐH/ DT, tồn kho, xuất nhập kho.



3. Xây dựng BCD theo mô hình quan hệ:

Những nội dung trước đã chỉ ra rằng: Khi thiết kế một cơ sở dữ liệu quan hệ thường đòi hỏi phải chọn các lược đồ quan hệ. Việc chọn tập các lược đồ này có thể tốt hơn hay xấu hơn tập các lược đồ khác dựa trên một số tiêu chuẩn nào đó. Trọng tâm của việc thiết kế các lược đồ cơ sở dữ liệu là ta tổ chức bao nhiêu lược đồ và mỗi lược đồ có những thuộc tính nào để đảm bảo hai tính chất sau:

- Không dư thừa và trùng lặp dữ liệu

Ví dụ: Xét lược đồ quan hệ chấm công

Ta nhận thấy rằng, các cặp giá trị dạng (p1, Giáp), với nghĩa là ông Giáp là quản đốc phân xưởng p1, đã có thể xuất hiện nhiều lần trong quan hệ này, nên đây là một lược đồ quan hệ không tốt, bị trùng lặp dữ liệu

- Nhất quán dữ liệu: Trong một lược đồ quan hệ xác định được nhiều phụ thuộc hàm, tất cả các quan hệ xác định trên lược đồ quan hệ phải thỏa mãn các phụ thuộc hàm trên lược đồ ấy

Ví dụ: Xét lược đồ quan hệ chấm công vừa nêu, mỗi công nhân khi thực hiện một công nhất định tại một phân xưởng nào đó, thì ta phải nhập số hiệu phân xưởng, tên vị quản đốc phân xưởng này một lần nữa. Các thông tin lần nhập sau có thể khác lần nhập trước như: Trùng tên quản đốc nhưng khác số hiệu phân xưởng, hay ngược lại, ... Dẫn đến hiện tượng tự mâu thuẫn trong CSDL. Hơn nữa khi một phân xưởng thay đổi quản đốc ta có thể không cập nhập thông tin này trong các bộ tương ứng của quan hệ. Như vậy đây là một lược đồ quan hệ không nhất quán dữ liệu.

4. Thành lập biểu đồ BCD

Để thành lập biểu đồ BCD ta chia thành 4 bước:

- **Bước 1:** Thành lập danh sách thuộc tính (danh sách xuất phát): Xuất phát từ những “điểm” khác nhau dẫn đến có nhiều nguồn danh sách xuất phát từ các nguồn:

Danh sách các thông tin cơ bản

Xuất phát từ một hay một số tài liệu xuất ra của hệ thống, nhặt các tiêu thức (cái ra Cái cần) “Phương pháp cái ra” Method Sortir

- **Bước 2:** Tu chỉnh lại danh sách:

Loại bỏ các tên đồng nghĩa

Loại bỏ các thuộc tính tính toán. Vì đây là giai đoạn logic chỉ tính đến đầy đủ và hợp lý chưa nói đến tiện lợi, sau này đến giai đoạn thiết kế có thể ta lại bổ sung thêm thuộc tính này.

Loại bỏ các thuộc tính tích lũy (Thực chất cũng từ thuộc tính tính toán): SL tồn = nhập - xuất

Thay thế các thuộc tính không đơn bằng các thuộc tính đơn

- **Bước 3:** Tìm các phụ thuộc hàm trong danh sách nói trên: Việc xác định các phụ thuộc hàm thực chất phải dựa vào ý nghĩa trên thực tế

Rà từng cặp thuộc tính trong những danh sách trên hoặc máy móc và đơn giản hơn là lập bảng 2 chiều.

Tìm các phụ thuộc hàm vế phải không đơn (gồm nhiều thuộc tính)

- **Bước 4:** Chuẩn hóa bằng phương pháp phân rã hoặc phương pháp tổng hợp

- **Bước 5:** Lập lại các bước 1- 4 trên các danh sách xuất phát khác,...

Nếu cùng một kiểu thực thể, từ các danh sách xuất phát khác nhau có các quan hệ danh sách thuộc tính khác nhau thì ta gộp lại. Khi gộp lại có thể xuất hiện phụ thuộc hàm bắc cầu (phụ thuộc hàm không trực tiếp), nên khi gộp xong phải tiếp tục chuẩn hóa quan hệ vừa gộp này.

Ví dụ:

Đơn hàng (SH-Đơn, SH-Khách hàng, Ngày ĐH)

Từ phiếu giao hàng: Đơn hàng (SH-Đơn, Tình trạng đơn, Đ/C giao hàng)

Gộp lại, nghiên cứu các phụ thuộc hàm nảy sinh bên trong.

Giả sử mỗi khách hàng có một địa chỉ giao hàng duy nhất, khi ấy có

SH-Khách hàng Đ/c giao hàng

Chuẩn hóa để được:

Đơn hàng (SH-Đơn, SH-Khách hàng, Ngày ĐH, Tình trạng đơn)

Khách hàng (SH-Khách hàng, Đ/C giao hàng)

KẾT LUẬN:

- ❖ Phân tích hệ thống về mặt chức năng:
 - BPC
 - BLD
- ❖ Phân tích hệ thống về mặt dữ liệu:
 - Mô hình thực thể liên kết
 - Mô hình quan hệ

Phân rã

Tổng hợp

Sau đó so sánh các phương pháp để xem có sai sót gì không?

Chỉ giữ lại những liên kết 1-nhiều cần để làm các đường truy nhập vì các liên kết nhiều-nhiều được tách ra, các liên kết 1-1 ít sử dụng

CHƯƠNG VIII: THIẾT KẾ TỔNG THỂ VÀ THIẾT KẾ GIAO DIỆN NGƯỜI MÁY

I. Đại cương:

Mục đích của việc thiết kế là nhằm thu được một mô tả hệ thống ở mức vật lý trên cơ sở mô tả hệ thống ở mức logic.

Xuất phát điểm của giai đoạn thiết kế chính là các BPC, BLD và các BCD (Theo mô hình TT/LK, mô hình quan hệ)

Các bước tiến hành:

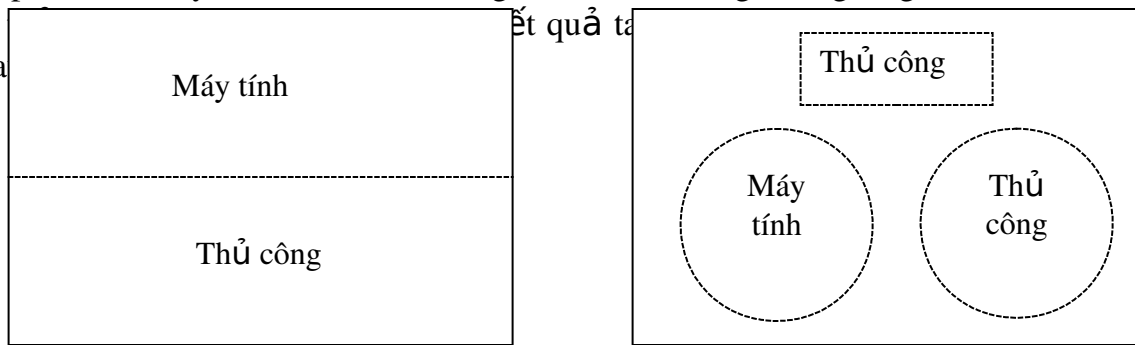
- Thiết kế tổng thể, bao gồm:
 - Phân định hệ thống con máy tính và hệ thống thủ công
 - Phân định các hệ thống con máy tính
- Thiết kế các giao diện: Giao diện người/máy, các thao tác thủ công
- Thiết kế các kiểm soát: Bảo mật, chống lại các sự cố kỹ thuật, chống ăn cắp, phá hoại,...
- Thiết kế CSDL
- Thiết kế chương trình

Tóm lại: Ở chương này chúng ta cần đưa ra một cấu trúc tổng thể của hệ thống. Cấu trúc này thể hiện sự phân chia hệ thống thành nhiều hệ thống con và sự phân tách giữa phần xử lý thủ công với phần xử lý tự động

II. Phân định hệ thống máy tính và hệ thống thủ công:

Đây chính là thời điểm cần chỉ rõ những chức năng nào sẽ do máy tính thực hiện, những chức năng nào vẫn do máy tính đảm nhiệm

Sự phân định này được thể hiện bằng cách vạch đường ranh giới giữa hai hệ thống như sau



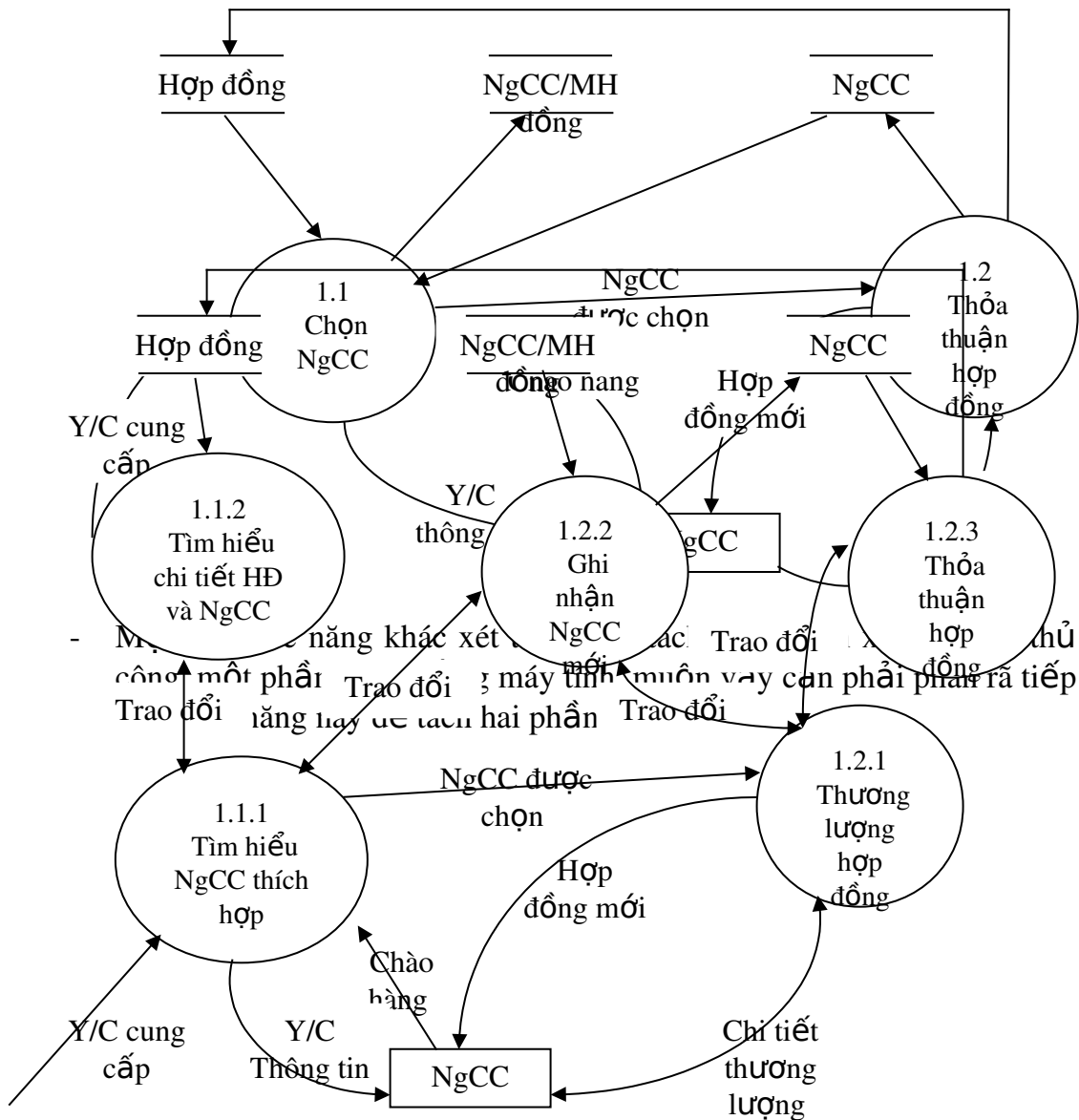
1. Đối với các chức năng xử lý:

Từng chức năng được đem ra thẩm định một cách kỹ càng để chia thành hai loại:

- Các chức năng tự động
- Các chức năng thủ công

Có thể xảy ra hai tình huống:

- Một chức năng sẽ được chuyển chọn vẹn sang nhóm xử lý tự động hay sang nhóm xử lý thủ công



2. Đối với các kho dữ liệu:

Cũng được phân làm hai loại:

- Kho dữ liệu được chuyển sang phần máy tính là các kiểu thực thể và sẽ tiếp tục có mặt trong BLD, về sau trở thành các tệp dữ liệu
- Kho dữ liệu chuyển sang phần thủ công là:

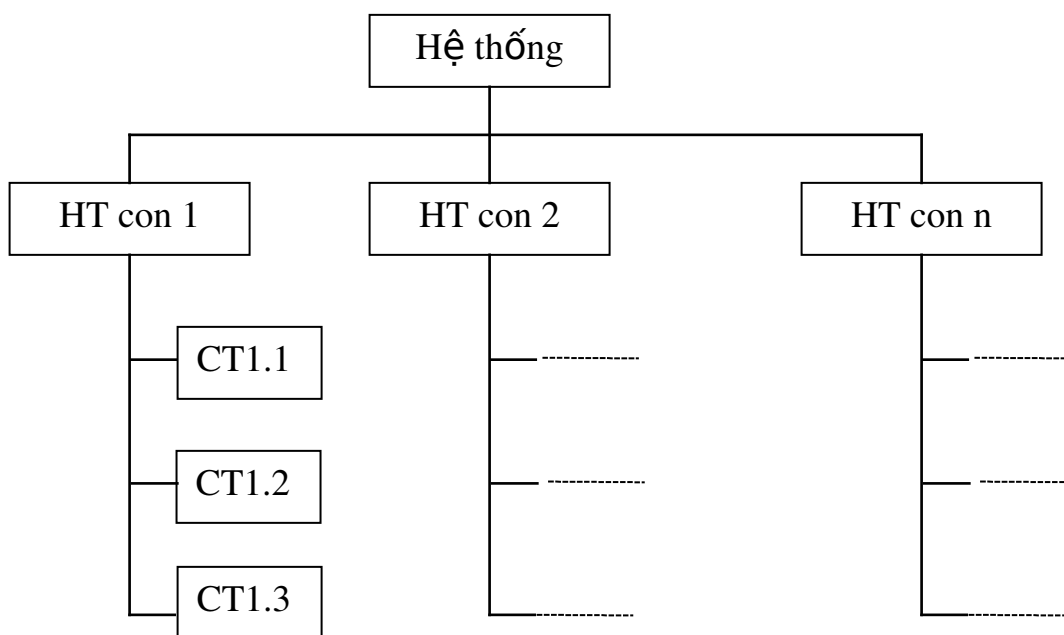
Các sổ sách, bảng biểu,...

Các hồ sơ trong văn phòng

Những kiểu thực thể tương ứng với các kho dữ liệu thủ công được loại ra khỏi BCD

III. Phân định hệ thống thành các hệ thống con máy tính

Một hệ thống con máy tính thực chất là một bộ riêng lẻ các chương trình:



Sự phân chia hệ thống thành các hệ thống con là nhằm tạo ra những thuận lợi trong cài đặt, bảo trì sau này. Trên thực tế, việc phân chia như vậy đã được thực hiện một cách tự nhiên ngay khi tiến hành phân tích hệ thống về mặt chức năng

Thông thường, các BLD ở mức cao cho ta những gợi ý tốt: Mỗi chức năng chính xuất hiện ở đó chính là một hệ thống con, gồm các chức năng sẽ được phân rã từ nó khi chúng ta đi xuống các mức dưới

Tuy vậy, việc phân chia hệ thống thành các hệ thống con thực ra không phải chỉ căn cứ vào các chức năng, mà còn phải xem xét đến nhiều khía cạnh khác, đặc biệt là các căn cứ nảy sinh từ thực tế như.

- ❖ *Theo thực thể*: Gom tụ vào một hệ thống con những chức năng xung quanh một thực thể, hay một nhóm kiểu thực thể: Hệ thống con khách hàng (bao gồm các chức năng về xử lý đơn hàng, làm hóa đơn, thanh toán,...); hệ thống con vật tư (bao gồm các chức năng xuất, nhập hàng, kiểm kê,...)
- ❖ *Theo giao dịch*: Gom các chức năng được khởi động khi có một giao dịch nào đó xảy ra, như: Một đơn hàng xuất hiện sẽ khởi động một loạt các chức năng: Ghi nhận đơn hàng, duyệt đơn hàng, kiểm tra khả năng đáp ứng của kho hàng,...
- ❖ *Theo trung tâm biến đổi*: Nếu nhận thấy trong BLD có một nhóm các chức năng cùng hợp tác để thực hiện một tính toán, một thao tác biến đổi thông tin,... thì sẽ gom chúng lại thành một hệ thống con

Ví dụ: Khi cần tính lương sẽ kéo theo những đầu vào: Cấp bậc, hệ số, ngày công,... Và đầu ra là bảng lương chi tiết, bảng tổng hợp lương... Khi đó ta sẽ gom các chức năng xử lý vào/ra cùng với các chức năng của trung tâm biến đổi thành một hệ thống con.

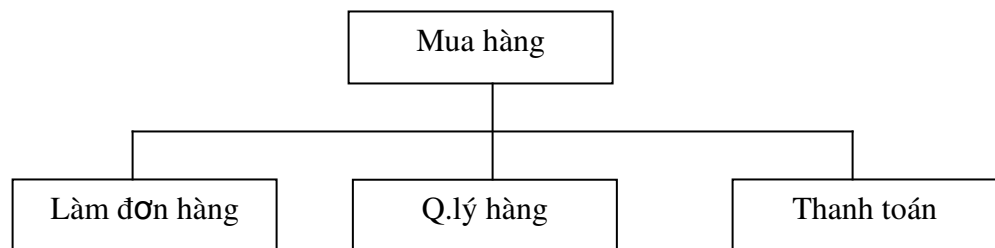
- ❖ *Theo tính thiết thực*: Đó chính là những căn cứ nhằm:
 - Tạo thuận lợi cho cấu trúc kinh doanh
 - Thích ứng với phần cứng sẵn có
 - Phù hợp với trình độ đội ngũ cán bộ
 - ...

IV. Cách biểu diễn:

Để mô tả các hệ thống con máy tính, chúng ta có thể sử dụng hai phương pháp:

1. Phương pháp mô tả tĩnh: Dùng biểu đồ BPC

Ví dụ:



2. Phương pháp mô tả động:

Ta mô tả phương pháp này thông qua ví dụ CƯVT: Xuất phát từ BLD mức đỉnh ta phân hệ thống thành các hệ thống con. Căn cứ của sự phân chia này và dựa trên cấu hình phần cứng vốn có, vì muốn giữ lại hai máy tính, chúng ta chọn giải pháp 5, ở đó chúng ta giữ lại hai hệ con ĐH và FH có điều chỉnh đôi chút

Tiếp theo chúng ta sẽ phân hoạch lại một lần nữa: Chức năng 4 đặt ở hệ ĐH, nhưng chuyển một phần việc là tìm địa chỉ phát hàng sang hệ FH (phù hợp với ý đồ điều chỉnh của giải pháp 5).

Đưa lại tên để phân ánh rõ nội dung hơn:

- Hệ thống ĐH được đặt lại tên là *Đặt hàng và theo dõi đơn hàng* (HT2), bao gồm bốn chức năng 1, 2, 4, 6.
- Hệ thống FH được đặt tên là *Quản lý kho, nhận và phát hàng* (HT2), gồm các chức năng 3, 5, 8, 9 và phần việc mới chuyển sang.
- Chức năng 7 chuyển sang cho phòng tài vụ, nên không ở trong hệ CWVT nữa.

Ta nhận thấy, hai hệ thống con thu được ở trên lại có thể phân chia tiếp:

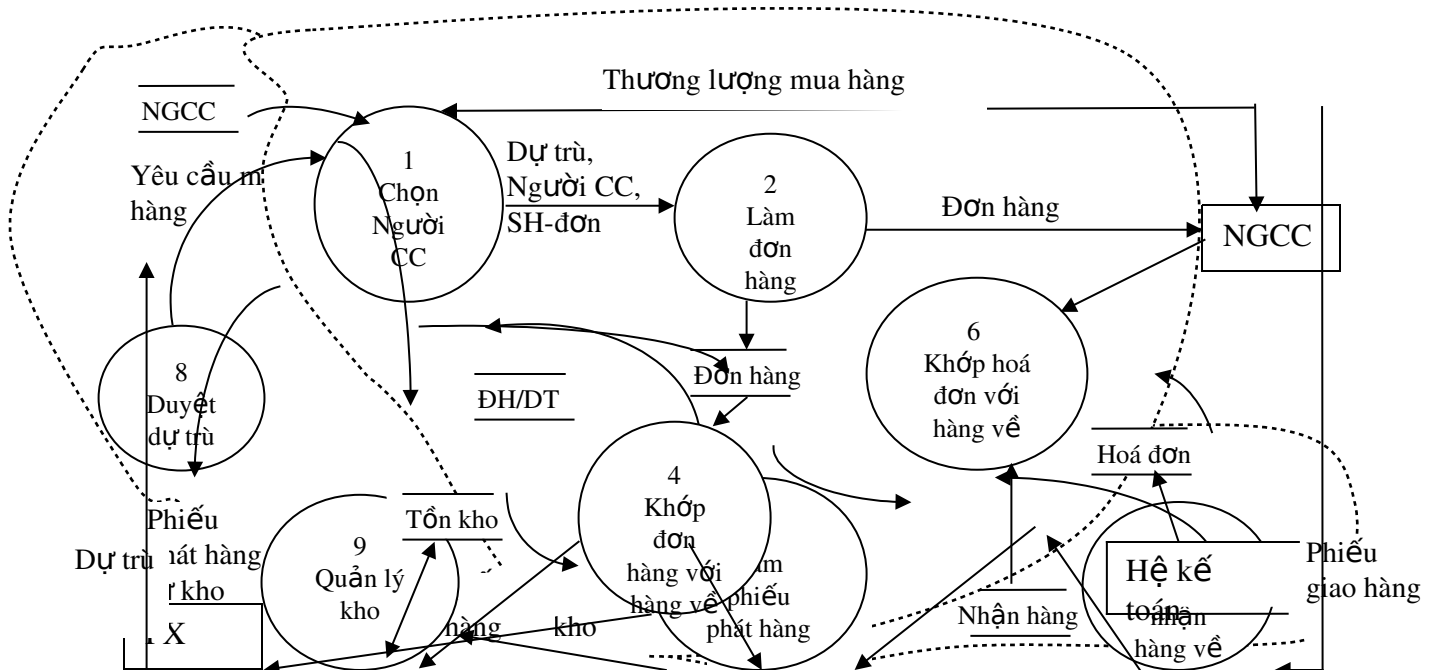
HT1: thành

- Hệ 1.1: Hệ đặt hàng
- Hệ 1.2: Hệ theo dõi đơn hàng

HT2: thành

- Hệ 2.1: Hệ quản lý kho
- Hệ 2.2: Hệ nhận và phát hàng

Sự phân tách trên thể hiện như sau:



V. Thiết kế chi tiết thủ tục thủ công và giao diện người/máy

Công việc tiếp theo sau khi tách hệ thống máy tính và hệ thống thủ công này là:

- Thiết kế các thủ tục thủ công (các chức năng do con người đảm nhận)
- Thiết kế các giao diện người/máy, tức là các mô thức chuyển giao dữ liệu giữa phần thủ công và phần tự động, nơi có các luồng dữ liệu xuyên qua ranh giới giữa chúng. Các giao diện này phục vụ việc đưa thông tin vào máy tính hoặc thu nhận các thông tin từ máy tính qua máy in, màn hình,...

1. Thiết kế thủ tục thủ công:

Các chức năng nằm trong phần thủ công cũng cần được đề cập đúng mức, vì sản phẩm đầu ra của chức năng này sẽ được đưa vào máy tính. Các chức năng này phải được mô tả thành các qui trình, công việc, ta nên gom chúng thành các nhiệm vụ lớn rồi giao cho một người hoặc một nhóm người thực hiện. Cũng như khi phân hệ thống thành các hệ thống con, việc gom nhóm các chức năng thủ công cũng dựa trên một số tiêu chí:

- Theo giao dịch
- Theo kho dữ liệu
- Theo địa lý
- Theo sự phân công trách nhiệm
- ...

Nội dung và hình thức của các nhiệm vụ lại tùy thuộc vào phương thức làm việc giữa người và máy tính, làm việc theo mẻ hay trực tuyến, mức độ khéo léo và độ chính xác, ... Đặc biệt là cách xử lý các sai sót gặp phải.

Các chức năng thủ công mới xuất hiện để phục vụ máy tính như:

- Mã hóa thông tin thu thập
- Kiểm soát và sửa chữa thông tin
- Nhập thông tin
- Kiểm tra tài liệu xuất
- Phân phối tài liệu xuất

2. Thiết kế các phương thức thu thập thông tin cho máy tính

- a. Xác định các thông tin cần đưa vào máy tính

Yêu cầu: Các thông tin đưa vào máy tính cần thỏa mãn các điều kiện:

- Đáp ứng các yêu cầu của ứng dụng
- Không sai sót (phải kiểm tra lại)
- Thông tin phải dễ hiểu, dễ dùng
- Gõ càng ít phím càng tốt
- ...

b. Nghiên cứu các phương thức thu thập

Có một số phương pháp thường được dùng để thu thập thông tin như sau:

- Trực tuyến: Thông tin đến thu thập ngay một cách đơn lẻ
- Trì hoãn (theo mẻ): Thông tin được gom thành mẻ rồi đưa vào
- Trực tiếp từ các cảm biến

Một trong những công cụ thường dùng khi tiến hành thu thập thông tin là mẫu điều tra và có cấu trúc như sau:

- *Phần đầu*: Thường là một tiêu đề phản ánh nội dung cần thực hiện và các thông tin chung
- *Phần thân*: Thể hiện nội dung, gồm các câu hỏi thăm dò, nhằm khai thác thông tin, Phần này thường được gom theo nhóm có liên kết chặt chẽ với nhau. Các thông tin có cấu trúc thường được trình bày thành bảng gồm nhiều cột, nhiều dòng
- *Phần cuối*: Chứa các thông tin kết thúc như: Ngày lập, chữ ký, các ước đoán.

Chú ý:

- Ở phần mặt trái phiếu thăm dò nên chứa các thông tin hướng dẫn sử dụng
- Thứ tự các dòng thông tin trình bày trong phần thân tài liệu có thể được quyết định tùy theo các căn cứ khác nhau, như:
 - Theo thứ tự ưu tiên
 - Theo thứ tự quen dùng để dễ hiểu
 - Các thông tin có cấu trúc nên để sau các thông tin đơn
- Các câu hỏi phải rõ ràng, dễ hiểu
- Về biểu mẫu thu thập thông tin, có ba cách lấy thông tin:
 - Có khung để điền
 - Các trường hợp để chọn
 - Các câu hỏi để trả lời (câu hỏi mở, câu hỏi đóng)

c. Thiết kế các phương thức xuất thông tin

❖ Xác định thông tin xuất:

Yêu cầu:

- Phải đầy đủ các thông tin cần xuất
- Phải chính xác (phải qua kiểm tra)
- Dễ đọc, dễ hiểu

❖ Xác định các phương thức xuất:

- In ấn:

In lập tức, đơn chiếc

In theo mẻ

- Xuất ra đĩa bộ nhớ ngoài), màn hình, xung điện

- Dạng tài liệu xuất ra:

Không có khung in sẵn

Có khung in sẵn

- Cấu trúc tài liệu xuất ra:

Đầu

Thân

Cuối

VI. Thiết kế màn hình và đơn chọn

- Mục đích: Sử dụng màn hình để đối thoại giữa người và máy, với đặc điểm:

Vào/ra phải gắn nhau

Thông tin trao đổi là tối thiểu

- Yêu cầu:

Phải sáng sủa, không lộn xộn

Phải chỉ rõ cái gì cần làm, thông tin nào được yêu cầu

- Hình thức đối thoại: Có 3 hình thức

Câu lệnh và câu nhắc: Máy tính hỏi và người nhắc lại

Tên bạn là gì?
Trần Thanh Bình
Địa chỉ bạn ở đâu?
131 Phố Thái Thịnh, Đống Đa, Hà Nội
....

Đơn chọn:

Thông tin về khách hàng:
(chọn một trong các việc sau)

1. Xem
2. Sửa
3. Nhập mới
4. kết thúc

Điền mẫu:

Vào các thông tin khách hàng:

Tên:
Địa chỉ:
Tài khoản:
Có ghi thông tin không ? (C/K)

Bài tập cuối chương:

1. Thiết kế tổng thể thực hiện các nhiệm vụ gì ?
2. Hệ thống con của HT là gì ? Vai trò của HT con trong HT? Cho VD minh họa
3. Hãy thiết kế giao diện cho chương trình cập nhập dữ liệu khi có độc giả yêu cầu mượn sách trong HT thư viện
4. Xét HT tuyển sinh vào một trường ĐH, hãy phân định HT máy tính và thủ công cho hợp lý và logic
5. Phân tích HT QL sản xuất của một nhà máy thành các HTC: Nhân sự, vật tư, lương, kế toán, kế hoạch, tiếp thị
6. Phân định HT kinh doanh tiền tệ tại ngân hàng với các chức năng: Tín dụng, tiết kiệm, đầu tư,...
7. Thiết kế dữ liệu đầu vào của HT:
 - QL nhân sự của trường ĐH
 - Hóa đơn thanh toán và các phiếu X/N của HT kinh doanh
 - Hồ sơ bệnh án trong các bệnh viện

CHƯƠNG IX: THIẾT KẾ CƠ SỞ DỮ LIỆU (xem lại chương này)

I. Đại cương:

Cơ sở dữ liệu là nơi lưu trữ lâu dài các thông tin của hệ thống, phục vụ các chức năng của hệ thống. Thiết kế CSDL hay nói đơn giản là thiết kế các tệp là một trong hai phương diện không thể thiếu được khi phát triển một hệ thống tin. CSDL phải được tổ chức tốt theo hai tiêu chuẩn:

- *Hợp lý*: Nghĩa là phải đủ, đúng và không dư thừa
- *Truy nhập thuận tiện*: Nghĩa là phải hỗ trợ tốt cho các thao tác tìm kiếm, cập nhật, bổ sung,...

II. Điểm xuất phát:

Sau giai đoạn phân tích hệ thống về mặt dữ liệu, chúng ta đã thu được BCD theo mô hình quan hệ và mô hình TT/LK, trong đó đặc biệt lưu tâm đến kho dữ liệu, đây chính là các căn cứ để chúng ta tạo lập CSDL cho hệ thống.

Mặc dù ở giai đoạn phân tích chúng ta đã cố gắng đạt được các tiêu chuẩn của lý thuyết về dữ liệu là đủ và không dư thừa thông qua các dạng chuẩn của mô hình quan hệ. Tuy nhiên khi cài đặt cụ thể để tiện lợi ta có thể bổ sung thêm một số thuộc tính tính toán, tích lũy, lặp lại một số thuộc tính, ghép một số kiểu thực thể thành một tệp (File)

1. Đưa thêm các thuộc tính tình huống:

Từ BCD để truy nhập nhanh và thuận tiện ta thực hiện các bước sau:

- Thêm thuộc tính tình huống, Thường là tính toán (Thành tiền=SL*ĐG), tích lũy (số dư tiết kiệm, lượng hàng tồn kho, số dư tài khoản)
- Lặp lại các thuộc tính từ những file khác.
- Gộp các kiểu thực thể, kiểu liên kết vào một file (có thể phải chấp nhận dư thừa) để bớt số lần truy nhập hoặc tách thành nhiều file vì không phải bao giờ cũng dùng hết các kiểu thực thể liên kết trong một lần truy nhập
- Lặp lại các file chỉ dẫn (index) để truy nhập được nhanh, căn cứ vào xử lý (nhu cầu sử dụng)

2. Nghiên cứu các đường truy nhập

Như đã phân tích, CSDL phải được thiết kế sao cho có thể đáp ứng các yêu cầu truy nhập một cách nhanh chóng và tiện lợi. Tuy nhiên sự đáp ứng này không phải hướng tới bất kỳ truy nhập nào, mà chủ yếu là hướng tới các yêu cầu truy nhập của các chức năng xử lý thông tin có mặt trong hệ thống

Mỗi đường truy cập gắn liền với chức năng xử lý, bởi vậy ta phải lần lượt xét từng chức năng trong BLD để tìm xem trong mỗi chức năng xử lý có những yêu cầu truy nhập nào.

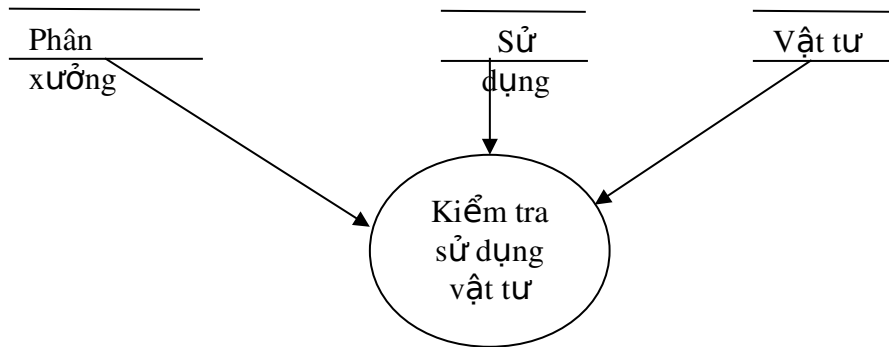
Ứng với mỗi xử lý ta cần chỉ ra các vấn đề sau:

- Truy nhập file nào?
- Khóa nào được sử dụng ?
- Tra cứu gì ?
- Tần suất truy nhập ?

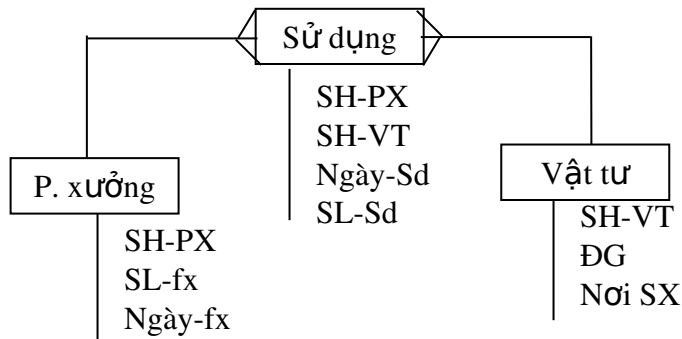
Nếu khóa và thông tin tra cứu trong cùng một file, lúc đó ta có một truy nhập trực tiếp. Các trường hợp còn lại gọi là truy nhập gián tiếp. Việc truy nhập gián tiếp thông qua đường truy cập bằng cách lần theo các mối liên kết 1-nhiều.

Vi dụ: Kiểm tra sử dụng vật tư của các phân xưởng trong nhà máy sản xuất:

- Về xử lý ta có một phân BLD sau đây:



- Về dữ liệu ta có BCD tương ứng như sau:



Giả sử khi nghiên cứu nội dung của chức năng “Điều tra việc sử dụng vật tư”, ta thấy nó chứa ba yêu cầu truy nhập là:

- A: Tìm ngày thành lập của phân xưởng
- B: Tìm đơn giá của một vật tư được sử dụng bởi một phân xưởng với SH-FX đã cho

C: Tìm ngày thành lập của các phân xưởng đã sử dụng một vật tư với SH-VT đã cho

Mỗi yêu cầu tạo ra một đường truy nhập gồm nhiều bước:

- Truy nhập file nào?
- Khóa nào được sử dụng ?
- Tra cứu gì ?
- Tần suất truy nhập ?

Yêu cầu A chỉ gồm một bước:

- Tập: Phân xưởng
- Khóa: SH-PX
- Tra cứu: Ngày thành lập phân xưởng
- Tần suất: 56 lần/tuần

Yêu cầu B cần hai bước:

BC1:

- Tập tra cứu: Sử dụng
- Khóa: SH-PX
- Tra cứu: SH-VT
- Tần suất: 34 lần/tuần

BC2:

- Tập tra cứu: Vật tư
- Khóa: SH-VT
- Tra cứu: Đơn giá
- Tần suất: $34 \times 10 = 40$ lần/tuần

Yêu cầu C cần hai bước:

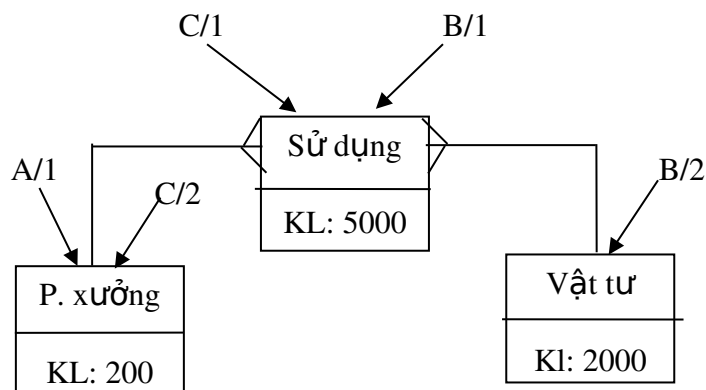
BC1:

- Tập tra cứu: Sử dụng
- Khóa: SH-VT
- Tra cứu: SH-FX
- Tần suất: 30 lần/tuần

BC2:

- Tập tra cứu: Phân xưởng
- Khóa: SH-FX
- Tra cứu: Ngày thành lập
- Tần suất: $30 \times 20 = 600$ lần/tuần

Sau cùng chúng ta tổng hợp các kết quả vào BCD



Qua sự tổng hợp này, ta có thể biết được:

- Bảng nào có nhu cầu truy nhập cao
- Khóa tìm kiếm nào được dùng nhiều
- Các cụm thông tin (trường) nào thường được tra cứu cùng nhau

Các nhận xét này giúp chúng ta xét lại cấu trúc bản ghi sau này.

III. Chuyển BCD thành các File dữ liệu:

Tập là một tập hợp các bản ghi cùng kiểu, mỗi bản ghi gồm nhiều trường và chiếm một số byte nhất định trong bộ nhớ.

Vì vậy, muốn tìm kiếm một bản ghi trong tập thì phải đọc lần lượt các bản ghi, bắt đầu từ bản ghi đầu tiên cho tới khi gặp được bản ghi mong muốn, một cơ chế truy nhập như vậy gọi là *truy nhập tuần tự*. Cơ chế này tốn nhiều thời gian, nhất là đối với các tập lớn (có hàng triệu, thậm chí hàng tỷ bản ghi)

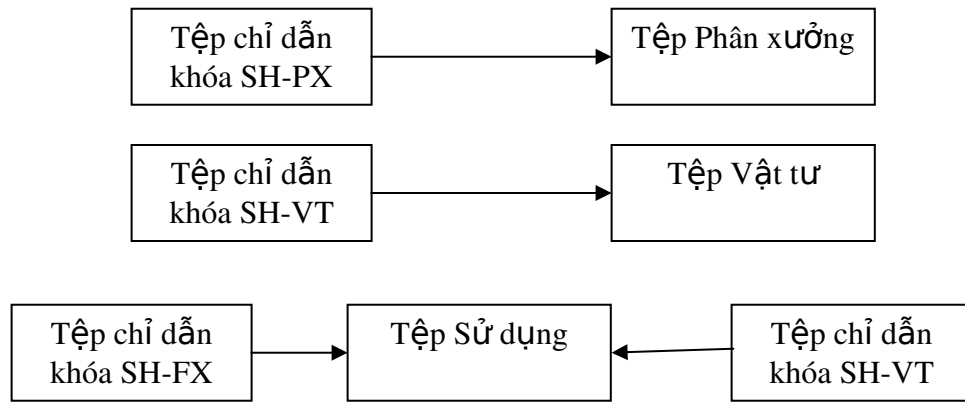
Nhằm khắc phục nhược điểm này, cần tạo ra khả năng *truy nhập trực tiếp*, cơ chế *truy nhập trực tiếp* được thực hiện nhờ tạo một tập chỉ dẫn. Tập chỉ dẫn tuy có cùng số lượng bản ghi như tập chính, nhưng “nhẹ” hơn tập chính, bởi vì mỗi bản ghi của nó chỉ có hai trường: Một trường là khóa chỉ dẫn, có giá trị trùng với một trường trong tập chính được dùng làm khóa tìm kiếm cho nhu cầu truy nhập, còn trường kia chứa con trỏ chỉ tới bản ghi tương ứng trong tập chính. Nhờ có con trỏ

Lập sẵn và khóa chỉ dẫn nên có thể dễ dàng truy nhập vào bản ghi cần tìm mà không cần phải tìm kiếm tuần tự

Nguyên tắc chuyển đổi:

- Nói chung, chuyển mỗi kiểu thực thể trong BCD chuyển thành một tập và có thêm thuộc tính tình huống (nếu cần)
- Thêm các tập chỉ dẫn đối với khóa tìm kiếm có tần suất sử dụng cao. Chúng ta lưu ý là không thể lập tập chỉ dẫn cho mọi khóa tìm kiếm, bởi có một hạn chế về hệ thống: Hệ nào cũng có ngưỡng tối đa cho các tập được mở tại cùng một thời điểm
- Khi cần có thể phân rã một thực thể (thường khi có quá nhiều thuộc tính) thành nhiều tập. Ngược lại có thể gộp nhiều thực thể thành một tập để hạn chế những đường truy nhập gián tiếp, tất nhiên nó sẽ phá vỡ mất tính chuẩn hóa

Xét ví dụ quản lý sử dụng vật tư ở trên ta có thể lập các file chỉ dẫn như hình sau:



Các Cơ sở dữ liệu

Cấu trúc lưu trữ vật lý phụ thuộc theo mô hình Cơ sở dữ liệu: Mô hình quan hệ, mô hình mạng hay mô hình phân cấp

- *Mô hình quan hệ : Mỗi bảng trong lược đồ logic chuyển đổi trực tiếp thành một quan hệ (mỗi trường thành một thuộc tính của quan hệ). Không có cài đặt gì đặc biệt đối với các kết nối giữa các bảng, ngoại trừ sự có mặt (vốn có) của các khoá ngoài. Một ngôn ngữ định nghĩa dữ liệu cho phép định nghĩa các quan hệ, và các lệnh trong một ngôn ngữ thao tác dữ liệu cho phép lưu trữ và tìm kiếm các dữ liệu dựa trên các phép toán của đại số quan hệ (thay cho sự lần dò theo các con trỏ)*
- *Mô hình mạng: Cơ sở dữ liệu theo mô hình mạng vẫn lưu trữ dữ liệu dưới dạng các bản ghi. Ngoài ra các liên hệ cha – con giữa các kiểu bản ghi (tức là các liên kết 1-nhiều) cũng được thiết lập dưới dạng các bộ con trỏ, gọi là các “set – Type”. Như vậy việc chuyển đổi từ lược đồ logic sang cấu trúc lưu trữ theo mô hình mạng cũng khá đơn giản: Mỗi bảng vẫn giữ nguyên là kiểu bản ghi. Mỗi kết nối được chuyển thành một “set – type” và được gán một cái tên (chẳng hạn set1, set2, ...). Các khoá ngoài, với vai trò kết nối các bản ghi của hai bảng, khi đã có các “set – type” trở nên vô dụng, cho nên được gỡ bỏ.*
- *Mô hình phân cấp: Cũng giống như mô hình mạng, song bắt buộc mỗi kiểu bản ghi chỉ có thể có nhiều nhất một cha. Vậy trước hết phải loại bỏ trường hợp nhiều cha trong lược đồ logic, bằng cách xét các liên kết 1- nhiều từ các cha, chỉ giữ lại một cha có cơ số trung bình N nhỏ nhất, còn thì cắt đứt với các cha khác. Sau đó chuyển đổi sang cấu trúc lưu trữ (gồm các kiểu bản ghi và các set-type) như ở mô hình mạng*

- A: Tìm ngày thành lập của phân xưởng nếu biết SHPX
- B: Tìm đơn giá của các vật tư được sử dụng bởi một phân xưởng với SHPX đã cho
- C: Tìm ngày thành lập của các phân xưởng đã sử dụng một mặt hàng với mã vật tư đã cho

Với mỗi A, B, C phải tìm đường truy nhập, mỗi đường truy nhập gồm nhiều bước, trong mỗi bước ấy ta ghi nhận với các nội dung sau:

- Tên tệp
- Tên đường truy nhập (A, B hay C)
- Bước thứ mấy
- Khoá truy nhập
- Mục đích truy nhập
- Tần số truy nhập

- Ví dụ:
- Đường A có một bước
 - Tệp sử dụng : Phân xưởng

- o Khoá : SHPX
 - o Tra cứu: Ngày thành lập
 - o Tần xuất: 150 lần/tuần
- Đường B có 2 bước
- Bước 1: B1
- o Tập sử dụng: Sử dụng
 - o Khoá: SHPX
 - o Tra cứu: Mã vật tư
 - o Tần xuất: 30 lần/tuần, mỗi lần tìm được $5000/100=50$ bản ghi
- Bước 2: B2
- o Tập sử dụng: Vật tư
 - o Khoá: Mã vật tư
 - o Tra cứu: Đơn giá
 - o Tần xuất: $30 * 50 = 150$ lần/tuần
- Đường C gồm 2 bước
- Bước 1: C1
- o Tập sử dụng: Sử dụng
 - o Khoá: Mã vật tư
 - o Tra cứu: SHPX
 - o Tần xuất: 20 lần/tuần, mỗi lần tìm được $5000/2000=2,5$ bản ghi
- Bước 2: C2
- o Tập sử dụng: Phân xưởng
 - o Khoá: SHPX
 - o Tra cứu: Ngày thành lập
 - o Tần xuất: $20 * 2,5=50$ lần/tuần

Cuối cùng ta tổng hợp các yêu cầu cần truy nhập:

CHƯƠNG X: THIẾT KẾ CÁC KIỂM SOÁT

I. Đại cương:

Thiết kế các kiểm soát là hạng mục không thể thiếu khi muốn phát triển một hệ thống tin. Mục đích của hạng mục này nhằm đề xuất các biện pháp giúp hệ thống đảm bảo được các yêu cầu sau:

- *Tính chính xác (Exactly)*: Tính chính xác trước hết thể hiện ở chỗ hệ thống làm việc đúng đắn, không đưa ra các kết quả tính toán sai lệch; tính chính xác còn có nghĩa là dữ liệu trong hệ thống phải xác thực, không mâu thuẫn và phải được kiểm tra
- *Tính an toàn (Safety)*: Thể hiện ở chỗ hệ thống không bị xâm hại khi có sự cố kỹ thuật
- *Tính an ninh (Security)*: Điều này có nghĩa là hệ thống phải có khả năng ngăn ngừa các xâm phạm vô tình hay cố ý từ phía con người
- *Tính riêng tư (Privacy)*: Hệ thống phải đảm bảo được các quyền truy nhập riêng tư đối với từng người sử dụng khác nhau.

II. Các nguy cơ đe dọa hệ thống nói chung

Để đảm bảo an toàn và sự hoạt động có hiệu quả của một hệ thống, rất nhiều vấn đề cần phải được quan tâm giải quyết. Các nguy cơ này bao gồm:

1. Các nguy cơ vật lý:

Các nguy cơ vật lý bao gồm từ những vấn đề kỹ thuật đến các thảm họa do thiên nhiên gây ra, ảnh hưởng đến độ an toàn của dữ liệu và hệ thống tin.

Ví dụ: Hỏa hoạn, điều kiện môi trường, sự cố nguồn điện,...

2. Các nguy cơ tin học:

- Các lỗi của hệ thống: Chủ yếu là các lỗi trong quá trình phát triển các ứng dụng, có thể do: Những người quản trị hệ thống, sự cố phần cứng gây ra,...
- Các lỗi do người sử dụng gây ra: Trong đó quan trọng nhất là các truy nhập kể cả hợp pháp và bất hợp pháp nhằm thay đổi, thêm, bớt dữ liệu phục vụ những mục đích xấu.
- Lỗi do virus tin học lây nhiễm vào hệ thống gây ra.

III. Tiếp cận hệ thống về phương diện kiểm soát

1. Phân tích và đánh giá các nguy cơ thất thoát:

Các điểm hở của hệ thống:

Trước tiên chúng ta quan tâm tới các điểm hở: Chính là nơi mà một tác nhân ngoài vì vô tình hay cố ý gây tác hại cho hệ thống, làm thiệt hại cho cơ quan

chủ quản. Người phân tích về kiểm soát trước hết phải phát hiện ra mọi điểm hở tiềm năng của hệ thống đang được xây dựng. Các điểm hở có thể là:

- Mọi luồng dữ liệu đi và đến một tác nhân ngoài
- Mọi luồng dữ liệu đi qua ranh giới giữa phần máy tính và thủ công
- Các kho dữ liệu.
- Các điểm truy nhập từ xa.

Với các điểm hở như vậy có thể gây ra các đe dọa sau:

- Mất thông tin: Do bị ăn cắp hoặc hỏng thông tin, điều đó có thể dẫn đến việc thất thoát tài sản, gây tổn kém, lãng phí
- Thông tin sai lệch còn dẫn đến những quyết định sai lầm trong kinh doanh

Đánh giá các nguy cơ:

Tất cả các xâm hại đến hệ thống cần được cân nhắc theo các khía cạnh sau:

- Các chi phí để tạo lại trạng thái bình thường của hệ thống
- Các tổn thất trong kinh doanh

2. Thiết kế các mức bảo mật:

Có rất nhiều biện pháp phòng ngừa nhằm ngăn chặn các âm mưu xâm hại đến hệ thống. Tuy nhiên mỗi biện pháp đều có những ưu, nhược điểm riêng, bởi vậy ta phải kết hợp đồng thời nhiều giải pháp để có thể đạt được hiệu quả cao nhất. Ta tham khảo một số giải pháp sau:

- Bảo mật vật lý:

Đó là các biện pháp sử dụng các công cụ vật lý, hoặc tác động lên phần cứng thay vì các phần mềm, cụ thể như: Dùng khóa hay lắp báo động để ngăn chặn và phát hiện hành động xâm nhập trái phép vào nơi bảo quản hệ thống.

Loại bỏ ổ đĩa ngoài khỏi các trạm làm việc trên mạng

- Nhận dạng nhân sự: Đó là một hình thức để nhận dạng trước khi bắt đầu một phiên làm việc
- Mật khẩu: Đây là một công cụ phổ dụng và hiệu quả
- Tạo mật mã: Dữ liệu được biến đổi dưới dạng không thể nhận thức được bằng cách thông thường
- Bảo mật bằng gọi lại: Sự truy nhập thực hiện một cách gián tiếp có kiểm soát
- Tường lửa (Firewall): Đó là một hệ thống, bao gồm cả phần cứng và phần mềm, được đặt giữa các mạng của một tổ chức, một đơn

vị, một quốc gia,... Chức năng của tường lửa là ngăn chặn các truy nhập trái phép. Loại bỏ các gói tin vào hoặc ra hệ thống khi cần thiết.

3. Phân biệt riêng tư:

Phân biệt riêng tư là việc phân loại các quyền truy nhập khác nhau, cụ thể là:

- Gán cho mỗi người dùng một số quyền truy nhập nhất định
- Cho phép một số người dùng được phép ủy quyền và rút quyền truy nhập cho người khác

IV. Các nguyên tắc đảm bảo an ninh cho hệ thống

1. Các nguyên tắc đảm bảo an ninh cho CSDL

- Đảm bảo tính toàn vẹn của CSDL
- Đảm bảo tính tin cậy của dữ liệu
- Đảm bảo khả năng sẵn sàng phục vụ của CSDL

2. Đảm bảo an ninh cho dữ liệu trong CSDL

Dữ liệu chỉ được nạp vào CSDL, sửa chữa, xóa đối với những người có quyền hạn hợp pháp. Các tác vụ này chỉ có thể thực hiện khi người thực hiện đã trả lời đúng tất cả các thông tin đã được đăng ký trong hệ thống khi hệ thống yêu cầu.

Dữ liệu được sao lưu làm hai bản, trong đó có một bản được lưu trữ tại máy chủ, một bản được lưu ra ngoài đĩa quang. Hệ thống cần được trang bị những công cụ phòng chống virus trên mạng để thường xuyên kiểm tra và loại trừ nguy cơ virus tin học thâm nhập ào hệ thống qua tác vụ truyền dẫn, dẫn đến phá hoại CSDL

3. Đảm bảo an ninh đối với người sử dụng CSDL

Với những người sử dụng hợp pháp (những người có đăng ký và được chấp nhận), hệ thống sẽ kiểm soát mức độ quyền truy nhập tương ứng với người sử dụng để khẳng định một tác vụ truy nhập hệ thống cụ thể có hợp lệ hay không ?. Nếu xuất hiện một tác vụ truy nhập không hợp lệ, hệ thống ngoài việc không phục vụ sẽ tiến hành ghi biên bản và thông báo ngay cho người có trách nhiệm quản lý hệ thống. Quyền truy nhập hệ thống ứng với từng người như sau:

- Quyền truy nhập để cập nhật dữ liệu vào CSDL của hệ thông
- Quyền truy nhập để sửa chữa các là thông tin
- Quyền truy nhập để sửa chữa, cập nhật, xóa, thống kê dữ liệu trong CSDL của hệ thống

V. Các biện pháp đảm bảo an ninh cho CSDL

1. Phòng chống các nguy cơ:

Trong khuôn khổ đơn vị, các vấn đề về điều kiện môi trường, kiểm soát các hoạt động không được phép của con người ở nơi làm việc,... Do đó, ta chỉ đề cập đến việc bảo vệ dữ liệu của CSDL tránh các thảm họa thiên nhiên gây ra. Vấn đề này được những người có trách nhiệm quản trị CSDL, sao lưu và phục hồi dữ liệu của hệ thống một cách thường xuyên.

2. Các giải pháp kỹ thuật cụ thể

Để thực hiện tất cả các biện pháp an ninh cho CSDL như trên, các giải pháp kỹ thuật được tổ chức như sau:

- o Những người có quyền hạn nạp thông tin vào hệ thống, thống kê
- o Những cán bộ kỹ thuật có trách nhiệm kiểm tra, theo dõi hệ thống
- o Những cán bộ kỹ thuật quyền sửa chữa các dữ liệu của hệ thống

Những người thuộc các nhóm trên nhất thiết phải sử dụng các số hiệu nhận dạng đăng ký trước với hệ thống, tuy nhiên số hiệu đăng ký tùy theo mức độ và quyền hạn của người được phép sử dụng

Những truy nhập “bất hợp pháp” được hiểu là không có đăng ký trước hoặc đăng ký nhưng không được phép truy nhập vào các mức độ khác của CSDL

Các biện pháp:

- Cài đặt hệ quản trị dữ liệu
- Tạo cơ sở dữ liệu và điều chỉnh cấu trúc CSDL
- Đăng ký người sử dụng
- Sao lưu và khôi phục CSDL
- Kiểm tra và theo dõi việc truy nhập vào CSDL
- Giám sát và tối ưu hiệu suất của CSDL

Các nhiệm vụ sau sẽ được thể hiện qua chức năng quản trị thường xuyên của hệ thống và được đề cập cụ thể trong chương này.

a. Sao lưu và khôi phục dữ liệu

Một trong những nhiệm vụ quan trọng của người quản trị hệ thống phải đảm bảo cho cơ sở dữ liệu luôn ở trạng thái làm việc sẵn sàng. Trạng thái làm việc của CSDL được thể hiện thông qua khả năng cho phép người sử dụng truy nhập vào CSDL bất kỳ lúc nào, nội dung của CSDL luôn được cập nhật và toàn bộ CSDL ở trạng thái ổn định và chính xác

Đối với tất cả các hệ thống CSDL, khả năng xảy ra các sự cố về hệ thống hay thiết bị luôn tồn tại ngoài ý muốn của con người. Ngoài ra các sự cố về thiên nhiên, môi trường và con người cũng là một tác nhân quan trọng dẫn đến sự phá hủy dữ liệu của hệ thống. Để đảm bảo an toàn cho dữ liệu của CSDL. Chức năng sao lưu và phục hồi là một trong những chức năng quan trọng của

hệ thống. Chức năng này phải đảm bảo khôi phục lại dữ liệu càng sớm càng tốt, đồng thời cách ly người sử dụng với các vấn đề do sự cố gây ra

Để có thể xây dựng được hoàn chỉnh chức năng sao lưu và khôi phục dữ liệu, trước hết cần xác định các nhân tố gây ra các sự cố. Có hai nhóm cơ bản:

- Các yếu tố thuộc hệ thống thông tin
- Các yếu tố bên ngoài hệ thống thông tin
- ❖ Các yếu tố thuộc hệ thống thông tin:

Các yếu tố này thường làm hoạt động của hệ thống thông tin bị treo, gây ảnh hưởng tới các hoạt động vào/ra của ổ đĩa, làm mất thông tin trong bộ nhớ của máy chủ.

Lỗi do người sử dụng: Người sử dụng là các nhân viên kỹ thuật do bất cẩn có thể làm hỏng dữ liệu của hệ thống, ví dụ loại bỏ bảng dữ liệu của hệ thống

Lỗi do các câu lệnh gây ra (statement failure)

Các tiến trình gây lỗi dẫn đến hệ thống bị treo

Lỗi mạng từ máy tính: Lỗi này gây ra khi người sử dụng mạng LAN hay truy nhập vào CSDL thông qua các trạm. Khi mạng máy tính bị sự cố có thể dẫn đến làm thay đổi sự hoạt động bình thường của hệ thống, đặc biệt với các hệ dữ liệu phân tán, vì vậy đảm bảo tính ổn định của CSDL trên toàn mạng. Hệ thống bị sập do các nguyên nhân về phần cứng và phần mềm dẫn đến làm mất thông tin trong vùng nhớ, lỗi này có thể xảy ra do khởi động một tiến trình hay sự cố về nguồn

- ❖ Các yếu tố bên ngoài hệ thống thông tin: Như sự phá hoại hay sự sửa đổi dữ liệu có chủ tâm của con người về mặt phần cứng và phần mềm.

b. Các quy trình sao lưu và khôi phục của người quản trị hệ thống:

Khi hệ thống được đưa vào hoạt động, một trong những trách nhiệm chính của người quản trị hệ thống là sao lưu và khôi phục dữ liệu của hệ thống khi có sự cố và sử dụng các biện pháp khác nhau để đưa hệ thống trở lại hoạt động bình thường. Những việc cần làm:

- Dự đoán trước được tất cả các khả năng sự cố có thể xảy ra và hiểu được bản chất của nó
- Chọn đúng phương pháp sao lưu dữ liệu thích hợp với CSDL
- Thực hiện đúng các công việc sao lưu dữ liệu
- Nắm bắt được các kỹ thuật khôi phục dữ liệu
- Tham gia kiểm tra và thử nghiệm các quy trình khôi phục

❖ **Quy trình sao lưu cụ thể đối với CSDL của đơn vị:**

Căn cứ vào điều kiện và yêu cầu thực tế để áp dụng việc sao lưu dữ liệu từng phần hay toàn phần theo kiểu đóng hay mở (có thể gọi theo cách khác là Online hay Offline). Sao lưu đóng có các ưu điểm như khái niệm đơn giản, dễ thực hiện, yêu cầu ít các thao tác qua lại và thực tế. Nhưng có nhược điểm là phải tiến hành đào tạo người quản trị hệ thống thật tốt và cũng có thể phát sinh nhiều lỗi hơn do phải tiến hành nhiều tác động qua lại với hệ thống

Việc sao lưu dữ liệu sang đĩa quang được tiến hành theo tháng và sao lưu toàn phần theo chế độ nối tiếp

Thông qua các điều kiện và yêu cầu thực tế về CSDL của đơn vị, có thể có giải pháp chọn kiểu sao lưu như sau:

- Sao lưu từng phần nối tiếp được đặt ở chế độ tự động theo từng ngày vào thời điểm nhất định
- Sao lưu ra bên ngoài được tiến hành bằng phương pháp sao lưu để bảo quản dữ liệu trong môi trường lưu trữ khác, thời điểm sao lưu vào cuối tuần

❖ **Khôi phục dữ liệu cụ thể đối với CSDL của đơn vị**

Tiến trình khôi phục dữ liệu được chạy tự động tại máy chủ, để phục vụ việc khôi phục dữ liệu khi xảy ra sự cố tại mỗi điểm của nó. Thông thường có các biện pháp khôi phục sau:

- *Khôi phục dữ liệu do lỗi của người sử dụng gây ra:*
 - *Các lỗi hay xảy ra do:*
 - Người sử dụng bất cẩn loại bỏ hay cắt bỏ bớt bảng dữ liệu
 - Người sử dụng xóa tất cả các hàng của bảng dữ liệu
 - Người sử dụng thực hiện chuyển giao dữ liệu, nhưng phát hiện thấy dữ liệu có lỗi cần sửa chữa
 - *Các biện pháp khắc phục:*
 - Khôi phục dữ liệu từ bản sao hợp lệ
 - Thực hiện việc nhập dữ liệu từ các bảng xuất
 - Khôi phục dữ liệu lại đúng thời điểm được đánh dấu trước khi xảy ra sự cố
- *Khôi phục dữ liệu từ lỗi do các câu lệnh gây ra:*
 - *Các nguyên nhân chung*
 - Lỗi về logic do chương trình ứng dụng gây ra

Người sử dụng cố ý đưa dữ liệu sai vào bảng dữ liệu

Người sử dụng cố ý thực hiện các thao tác không được phép

Người sử dụng thử tạo các bảng dữ liệu quá quyền hạn của mình

Người sử dụng cố ý thực hiện việc cập nhật và bổ sung dữ liệu vào bảng dẫn đến làm ảnh hưởng đến không gian nhớ của CSDL

▪ *Các biện pháp khắc phục:*

Tìm và khắc phục lỗi trong các chương trình ứng dụng, sửa các câu lệnh

Trở lại hệ thống tạo các quyền sử dụng dữ liệu trong CSDL đúng với đối tượng sử dụng

❖ **Khôi phục dữ liệu do hệ thống hỏng:**

- Các nguyên nhân chung:

Nguồn điện ngừng hoạt động

Phần cứng hỏng (CPU hay bộ nhớ hỏng)

Phần mềm hỏng

Một số tiến trình hỏng dẫn đến cả hệ thống bị treo

- Các biện pháp khắc phục

Khôi phục bằng các file đã sao lưu

Tiến trình khôi phục về điểm ổn định sau sự cố

Tiến trình khôi phục về điểm ổn định trước sự cố

Câu hỏi cuối chương:

1. Nêu vai trò của việc kiểm soát và bảo mật hệ thống
2. Có thể tránh được mọi sai sót, rủi ro đối với hệ thống không? Cách lựa chọn và khắc phục như thế nào?
3. Hãy chỉ ra nguyên tắc phân quyền và ủy quyền đối với hệ thống

CHƯƠNG X: THIẾT KẾ CHƯƠNG TRÌNH

I. Đại cương:

Thiết kế chương trình bao gồm việc thiết kế các hạng mục sau:

- Thiết kế giao diện.
- Thiết kế kiểm soát
- Thiết kế các tệp (file, CSDL)
- Thiết kế chương trình

Trong các chương trước đã trình bày ba hạng mục đầu tiên, để có thể lập trình được cần đề cập tới một số chức năng hỗ trợ khác nhau

- Chức năng đối ngoại
- Chức năng xử lý lỗi
- Chức năng xử lý vào/ra
- Chức năng tra cứu CSDL
- Chức năng Môđun điều hành

Đầu vào của thiết kế chương trình là BLD của từng hệ thống con, được xác định trong thiết kế tổng thể, cùng với các thiết kế về giao diện, kiểm soát và CSDL đã được thực hiện trong thiết kế trước đây

Đầu ra của thiết kế chương trình là một mô tả về nội dung của các chương trình cần cài đặt, bao gồm:

- Một lược đồ chương trình (LCT) cho mỗi hệ thống con, lược đồ này được trình bày dưới dạng một đồ thị có hướng, gọi là lược đồ cấu trúc (Structure chart) trong đó:

Mỗi nút là một môđun chương trình

Mỗi cung là một lời gọi

- Đặc tả nội dung của từng môđun trong LCT
- Phân định các môđun trong LCT thành các môđun tải
- Thiết kế các mẫu thử

II. Lược đồ cấu trúc

Lược đồ cấu trúc là một biểu diễn dưới dạng đồ thị của một tập hợp các môđun cùng với các giao diện giữa các môđun đó

1. Các môđun chương trình:

Môđun chương trình có thể được hiểu là:

- Một chương trình con: dạng Procedure, function, subroutine,...

- Một cụm lệnh trong chương trình

Môđun có các thuộc tính sau:

Đặc trưng ngoài:(Phương pháp thiết kế có cấu trúc thường lưu ý trước hết đến đặc trưng ngoài)

- Vào/ra:

Vào: Thông tin nhận được từ chương trình gọi

Ra: Thông tin trả lại cho chương trình gọi

- Chức năng biến đổi cái vào cái ra.

Đặc trưng trong:

- Cơ chế: Đó là phương thức cụ thể để biến đổi cái vào thành cái ra
- Dữ liệu cục bộ: Là chỗ nhớ hay cấu trúc dữ liệu dành riêng cho Môđun

Ngoài 4 thuộc tính cơ bản trên, Môđun chương trình còn có một số thuộc tính khác cũng cần lưu ý tới khi thiết kế.

- Tên của Môđun: Cho phép dùng tên để gọi nó
- Chỗ chiếm của Môđun trong bộ nhớ trong

Các loại Môđun chương trình thường có trong hệ thống quản lý:

- Chương trình đơn chọn (menu Program)
- Chương trình nhập dữ liệu (data entry program)
- Chương trình biên tập, kiểm tra dữ liệu vào (Edit program)
- Chương trình cập nhật dữ liệu (Update program)
- Chương trình hiển thị tra cứu (Display program)
- Chương trình tính toán (compute program)
- Chương trình in (Print program)

2. Các yếu tố hợp thành LCT

a. Các Môđun:

Một Môđun được biểu diễn bằng hình chữ nhật trên có ghi nhãn là tên Môđun

Tên Môđun

Tính lương

Chú ý:

- Tên Môđun phải phản ánh được chức năng của nó

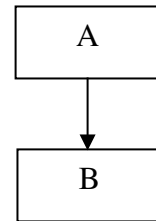
- Trường hợp nếu là Môđun đã có sẵn, thì các cạnh bên được vẽ kép



b. Kết nối các Môđun

Các Môđun được kết nối với nhau bằng các lời gọi, diễn tả bởi một cung định hướng

- Môđun A gọi Môđun B
- Môđun B thực hiện chức năng của mình rồi trả điều khiển cho A ở sau vị trí lời gọi



Trong cách biểu diễn lời gọi bằng các cung, ta cần hiểu thêm như sau:

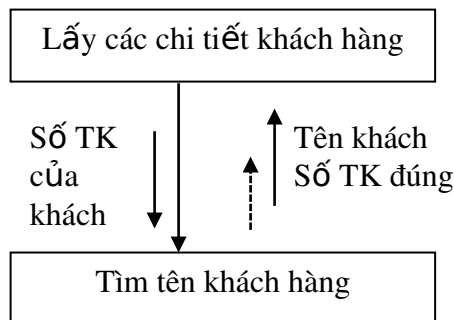
- Vị trí của lời gọi cũng như số lần gọi không được chỉ rõ. Điều này cần thiết vì trong giai đoạn lập LCT thì các chi tiết này tạm lược bỏ để giảm bớt sự phức tạp
- Trật tự trước sau của nhiều lời gọi xuất phát từ cùng một Môđun được thể hiện bởi trật tự từ trái qua phải

c. Thông tin chuyển giao giữa các Môđun

Các Môđun chuyển giao thông tin với nhau bằng dữ liệu và điều khiển:

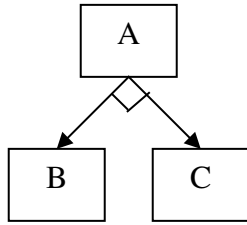
- Dữ liệu chuyển giao ký hiệu bằng mũi tên:
- Những thông tin điều khiển (không là đối tượng để xử lý mà dùng trong quá trình điều khiển thực hiện chương trình), ký hiệu bằng mũi tên đứt nét:

Ví dụ:

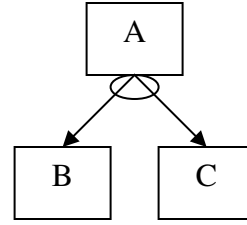


3. Một số trường hợp đặc biệt

Trong sơ đồ LCT có thể xảy ra sự chuyển giao từ Môđun này tới Môđun kia cần sự lựa chọn phù hợp hoặc chuyển giao ấy lặp lại nhiều lần, chúng ta ký hiệu riêng

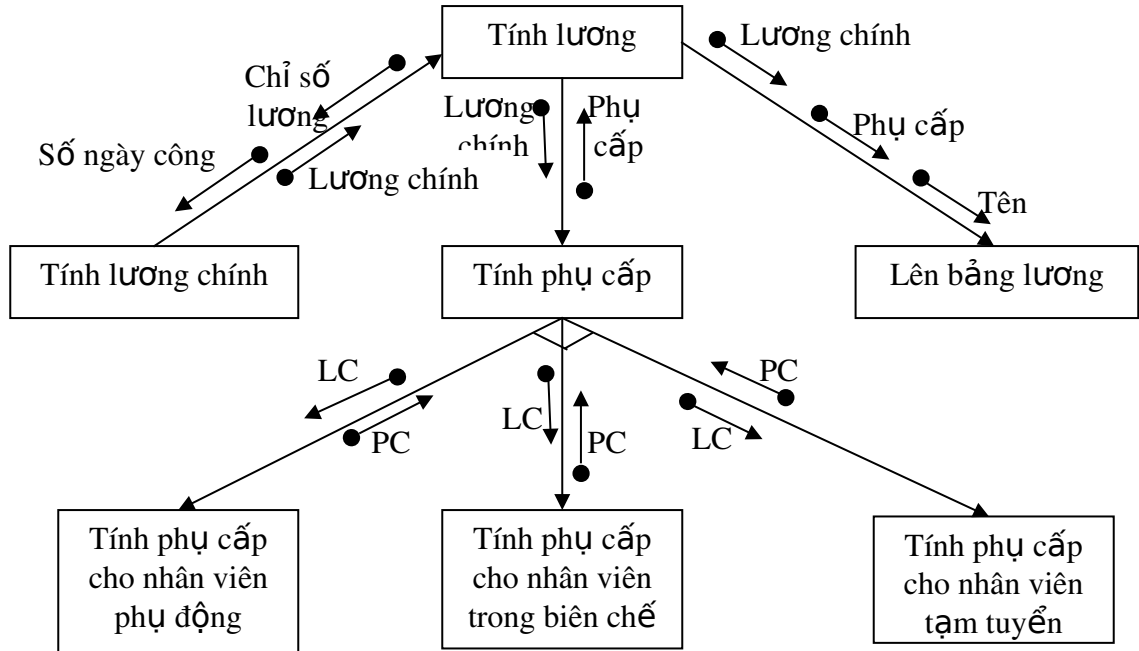


Ký hiệu thể hiện chọn lựa giữa B và C



Ký hiệu thể hiện gọi B và C nhiều lần

Một LCT diễn tả qui trình tính lương



Các môđun ở trên là các môđun điều khiển, còn các môđun ở dưới là các thao tác

4. Chất lượng của lược đồ cấu trúc

Một trong những phạm vi chất lượng thiết kế là sự tương tác, tức là sự phụ thuộc giữa hai môđun với nhau. Vấn đề cần bàn ở đây là sự tương tác tối thiểu, tức là tạo một môđun có độ độc lập có thể được, độ tương tác thấp giữa các môđun chỉ ra sự phân chia tốt trong hệ thống và các môđun có thể đạt được theo một trong ba cách sau:

- Lược bỏ những mối quan hệ không cần thiết
- Giảm bớt các quan hệ không cần thiết
- Bỏ đi các mối quan hệ lỏng lẻo cần thiết

Tiêu chuẩn:

- ❖ Sự tương tác: Nói lên mức độ ảnh hưởng lẫn nhau giữa các môđun, càng lỏng lẻo càng tốt, càng đơn giản càng tốt

- *Tương tác nội dung*: Tức là một môđun chương trình này làm biến đổi nội dung chương trình khác. Đây là sự tương tác xấu nhất cần tránh
- *Tương tác điều khiển*: Một môđun này chuyển chương trình điều khiển cho một môđun (có điều kiện chọn), cố gắng sử dụng càng ít càng tốt
- *Tương tác dữ liệu*: Tương tác này chỉ trao đổi vào/ra (bọc kín), cần thiết chấp nhận được. Nhiều khi thay dữ liệu thật bằng các con trỏ (nên hạn chế con trỏ)

❖ **Sự cố kết**: Nói lên sự gắn bó giữa các phần trong nội bộ một môđun

Yêu cầu:

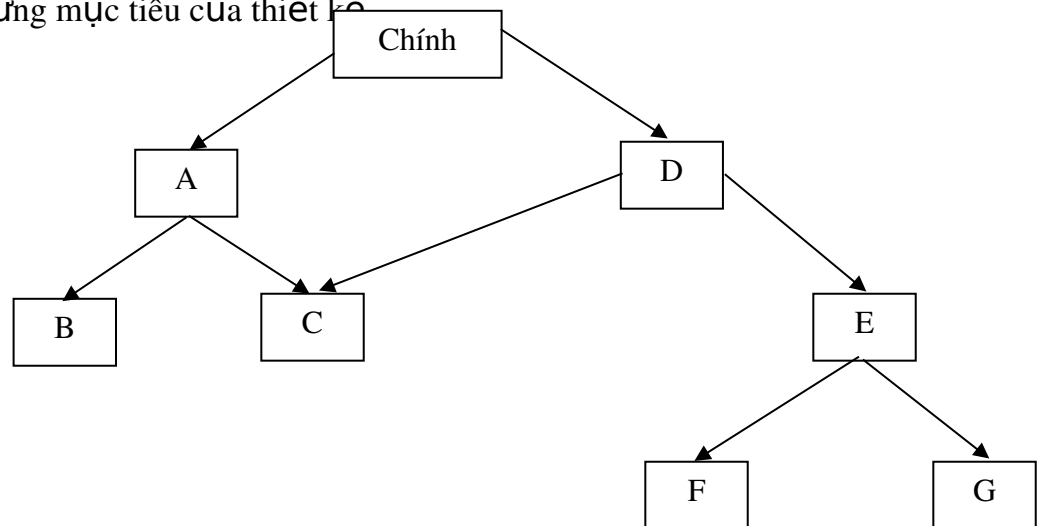
- Sự cố kết càng cao càng tốt
- Mỗi môđun chỉ thực hiện một chức năng logic

Môđun càng cố kết, thì chức năng của nó càng dễ thấy, logic của nó càng sáng sủa, do đó dễ phát hiện lỗi, dễ bảo trì, nâng cấp.

5. Hình thái của lược đồ cấu trúc:

Hình thái của lược đồ cấu trúc phản ánh chất lượng của nó

Một LCT, khi đi từ trên xuống, được coi là tốt nếu trên xòe ra và dưới chụm lại. Thông thường trong một LCT, các môđun càng ở phía trên thì chức năng của chúng càng nặng về điều khiển, nhẹ về xử lý. Trái lại, các môđun càng ở phía dưới thì chức năng của chúng bớt đi tính điều khiển, nhưng lại tăng tính xử lý. Bởi vậy trong một LCT, hiện tượng xuất hiện nhiều các điểm chụm ở dưới chứng tỏ rằng có nhiều môđun được sử dụng lại nhiều lần, mà sự sử dụng lại là một trong những mục tiêu của thiết kế.



Người ta gọi phạm vi điều khiển của một môđun là chính môđun đó cộng với những môđun được gọi trực tiếp hay gián tiếp từ nó. Chẳng hạn trong hình trên phạm vi điều khiển là A, B, C

Người ta gọi phạm vi ảnh hưởng của một quyết định là mọi môđun chịu ảnh hưởng của kết quả quyết định đó, chẳng hạn nếu trong B có một quyết định q và kết quả của quyết định này được dùng trong A, E, F thì phạm vi ảnh hưởng của quyết định q là A, E, F

Một thiết kế tốt phải tạo ra LCT, trong đó:

- Các quyết định phải có miền ảnh hưởng càng hẹp càng tốt
- Một phạm vi ảnh hưởng phải nằm trong phạm vi điều khiển tương ứng

III. Cách chuyển từ BLD thành LCT

1. Yêu cầu chung:

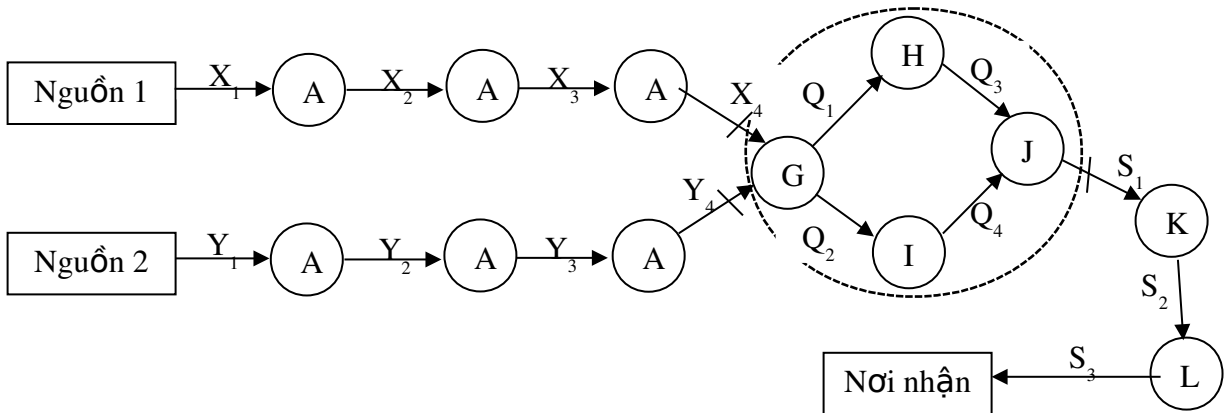
Đối với mỗi BLD của hệ thống con, lập một LCT tương ứng, LCT phải đạt các yêu cầu sau:

- Tất cả các nhiệm vụ của mọi chức năng xử lý trong BLD phải được chuyển hết vào các môđun của LCT
- Phải thêm các môđun vào/ra, đó chính là các giao diện với người dùng hay truy cập vào CSDL, và đặc biệt là các môđun điều khiển dẫn dắt quá trình xử lý
- Thiết lập các lời gọi giữa các môđun

2. Hai hướng thiết kế

a. Thiết kế theo trung tâm biến đổi

Phương án này áp dụng cho trường hợp BLD có nhiệm vụ biến đổi một số thông tin lấy từ nguồn phát, thành các thông tin gửi tới nơi nhận



Sơ đồ BLD theo trung tâm biến đổi

Qui trình diễn ra như sau:

Bước 1: Trước hết ta hãy xác định trung tâm biến đổi:

- Dõi theo các hướng thông tin vào cho đến khi thông tin được biến đổi trừu tượng nhất hoặc đến lúc chúng không được coi là dữ liệu vào nữa thì chúng ta đánh dấu

- Ngược theo các luồng dữ liệu ra cho đến khi chúng không được coi là dữ liệu ra nữa, ta dừng lại và đánh dấu
- Căn cứ vào các điểm đánh dấu, khoanh vùng để thể hiện các trung tâm biến đổi

Bước 2: Vẽ hai mức cao nhất của LCT, trong đó:

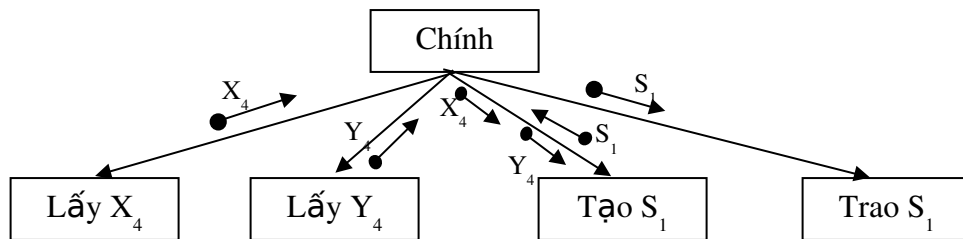
- Mức trên là môđun chính (đỉnh) gọi đến các môđun mức dưới
- Mức dưới bao gồm ba môđun:

Một môđun lấy thông tin cho mỗi tuyến vào

Một môđun tạo thông tin cho trung tâm biến đổi

Một môđun chuyển giao thông tin cho mỗi tuyến ra

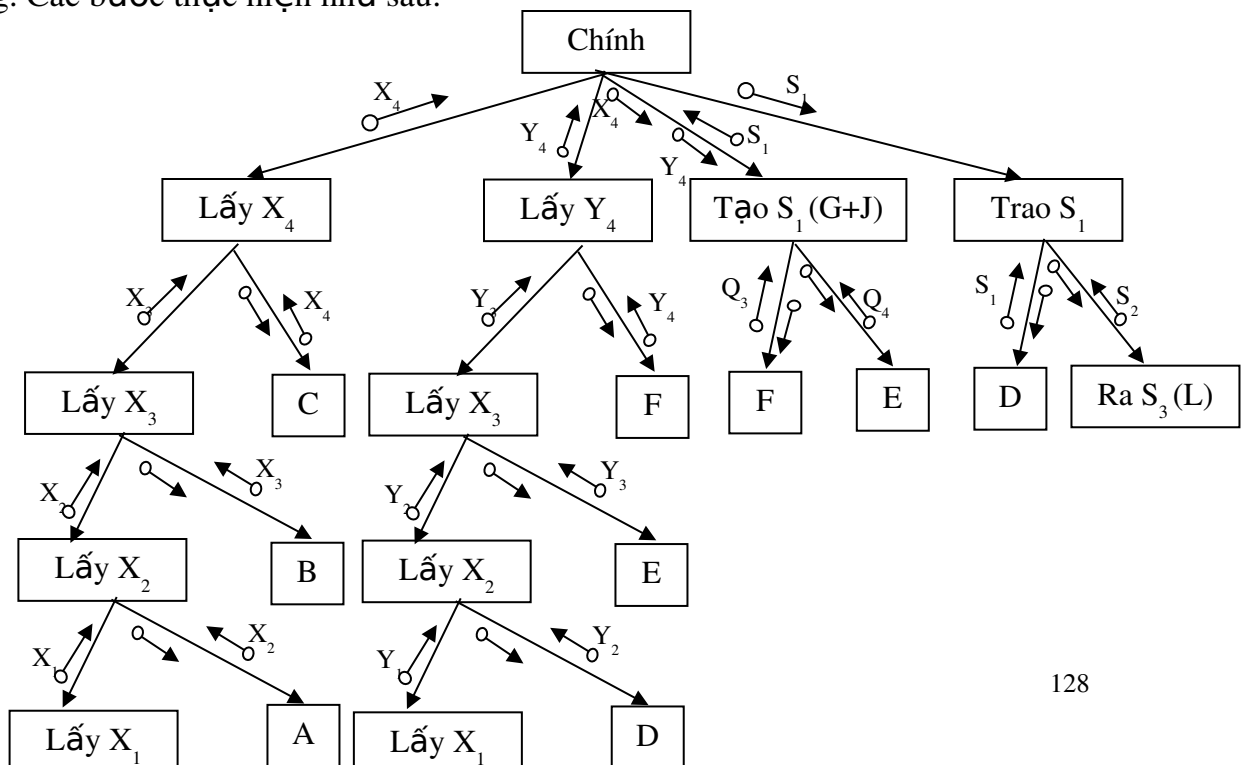
Chẳng hạn, thực hiện bước này đối với BLD cho hình trên ta thu được kết quả cho hình sau:



Bước 3: Triển khai mỗi môđun ở mức 2 (vào, ra hay biến đổi) thành các môđun ở các mức thấp hơn, ... cho tới khi mọi chức năng xử lý của BLD vào hết LCT

b. Thiết kế theo hướng giao tác:

Đặc trưng của cấu trúc giao tác là có một chức năng phân loại cho phép xác định loại dữ liệu vào, để rồi ứng với mỗi loại dữ liệu sẽ có một cách xử lý riêng. Các bước thực hiện như sau:



Sơ đồ theo phương pháp thiết kế giao tác

Bước 1: Phát hiện một giao tác trong BLD, đó chính là một chức năng xử lý nhận một luồng dữ liệu vào và xuất ra nhiều luồng dữ liệu loại trừ lẫn nhau

Bước 2: Xác định các chức năng khác nhau ứng với mỗi luồng dữ liệu ra (xác định các chức năng để khởi động mỗi lần ra của giao tác).

Bước 3: Vẽ hai mức cao nhất của LCT, trong đó môđun đỉnh (chính) gọi đến:

- Một môđun lấy thông tin
- Một môđun xử lý cho mỗi trường hợp của dữ liệu vào (được gọi qua phép chọn)
- Một môđun chuyển giao thông tin cho tuyến ra

Bước 4: Triển khai xuống các mức thấp hơn

Chú ý: Trong một BLD lớn nên phối hợp hai phương pháp như đã trình bày ở trên, một mặt để thích ứng với các cấu trúc xuất hiện, mặt khác để tạo hiệu quả trong công việc

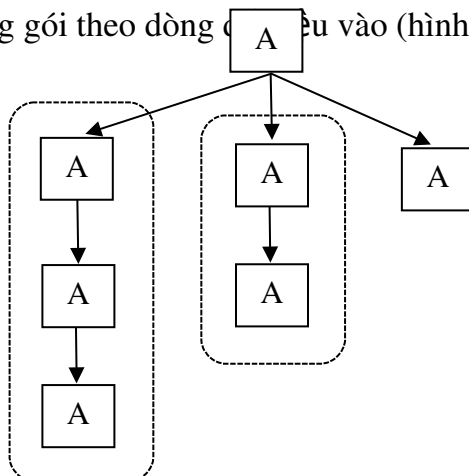
IV. Đóng gói thành môđun tải

Như đã biết một môđun chương trình muốn thực hiện phải đưa vào bộ nhớ trong (còn được gọi là nạp vào hay được tải vào). Mỗi lần nạp vào như vậy hệ thống cần phải có thời gian

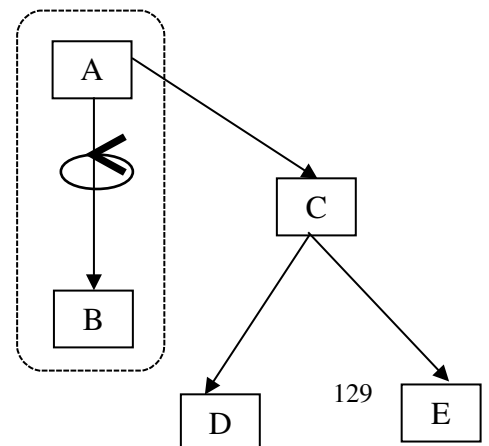
Một môđun tải là một nhóm môđun chương trình được nạp vào bộ nhớ một cách đồng thời. Vì bộ nhớ của máy tính là có giới hạn, nên ta không thể để tất cả các môđun chương trình của một LCT thành một môđun tải, vì sẽ mất nhiều thời gian cho công đoạn nạp

Một giải pháp trung gian sẽ là chia cắt LCT thành các môđun tải, vậy nên chia cắt như thế nào cho hợp lý? Có một số cách sau:

- Đóng gói theo dòng dữ liệu vào (hình 1)

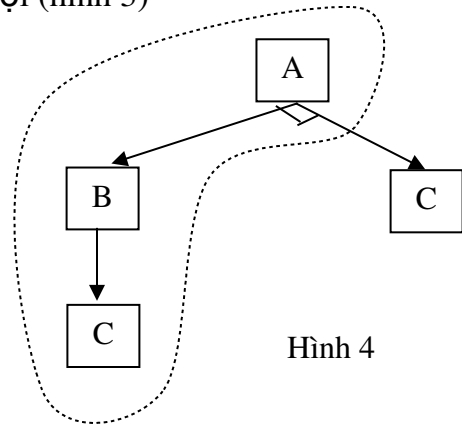
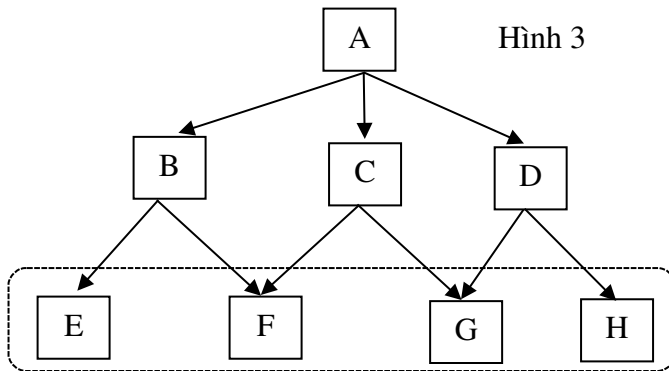


Hình 1



Hình 2

- Đóng gói các môđun ở mức thấp vào chung một môđun tải vì chúng được sử dụng bởi nhiều môđun khác. (hình 2)
- Môđun lập thường ghép chung với môđun gọi (hình 3)



- Đóng gói theo phép chọn không cân đối (hình 4)

CHƯƠNG XI: LẬP TRÌNH, CHẠY THỬ VÀ BẢO TRÌ

2. Lập trình:

a. Thành lập tổ lập trình:

Tổ lập trình là một nhóm tham gia việc viết các Môđun và được lắp ghép thành hệ thống. Việc thiết kế hệ thống càng chi tiết bao nhiêu và mang tính hệ thống cao sẽ giúp cho việc thực hiện cài đặt và phát triển hệ thống hoàn thiện bấy nhiêu

Một chương trình ứng dụng trung bình có từ 8000 đến 15000 câu lệnh và trung bình người ta có thể viết được 30 câu lệnh mỗi ngày.

Từ cơ sở trên tạo nhóm lập trình bao gồm bao nhiêu người trong khoảng thời gian bao lâu

b. Chọn ngôn ngữ lập trình

Những ngôn ngữ mang tính hệ thống viết được ra môi trường thường dùng là C, C++, Pascal, và môi trường chuyên dùng: Cobol, Access, VB,.... Môi trường điển hình hiện nay là HQTCSDDL (ORACLE)

c. Cài đặt các tệp, viết các đoạn chương trình chung

d. Soạn thảo chương trình cho từng đơn vị xử lý

Yêu cầu đối với các chương trình:

- Vào ra phải đúng đắn
- Dễ đọc, dễ hiểu để còn bảo trì
- Dễ sửa chữa, nâng cấp
- Chạy phải nhanh, tiết kiệm bộ nhớ có hiệu quả không gian, thời gian
- Tối ưu hóa về mã: Thể hiện ở thời gian và chỗ chiếm bộ nhớ

2. Chạy thử và ghép nối:

Chạy thử và ghép nối để cho ra một mẫu thử hệ thống

3. Thành lập các tài liệu hướng dẫn

Tài liệu hướng dẫn có vai trò quan trọng với người sử dụng

a. Đại cương:

Mục đích của tài liệu là để trao đổi, liên lạc. Nhà phân tích tham gia phát triển hệ thống cần trao đổi với một số người trước, trong và sau tiến trình phân tích và thiết kế. Thông tin thu được cần phải được ghi lại theo khuôn dạng làm thuận tiện cho việc thâm nhập và tìm kiếm. Kết quả của hoạt động phân tích và các ý tưởng được xem xét trong giai đoạn thiết kế (cả những ý tưởng được chấp nhận cũng như bị loại bỏ) đều cần được thu tóm dưới dạng văn bản nào đó, trước hết để giúp làm đầy đủ tiến trình phát triển rồi thứ nữa để hỗ trợ cho việc chạy và bảo trì hệ thống khi nó đi vào hoạt động.

Về cơ bản có hai khuôn dạng tài liệu. Chúng liên quan tới hai nhóm người tham gia trong việc phát triển và các nhu cầu thông tin khác nhau:

- Người dùng (thuật ngữ được dùng ở đây bao hàm cả nhà quản lý, người chủ và người vận hành hệ thống). Tài liệu cho những người này phải được chuẩn bị một cách chính thức bởi nhóm phát triển (một số trong họ cũng chính là người dùng). Tài liệu này được xem như một phần của việc bàn giao hệ thống. Trong phương pháp luận Systemscraft, các tài liệu bàn giao bao gồm:

Đặc tả yêu cầu nghiệp vụ

Đặc tả thiết kế hệ thống

Tài liệu cho người dùng

Hướng dẫn vận hành

- Người phát triển: (Thuật ngữ được dùng ở đây bao hàm cả nhà phân tích, người thiết kế, người làm bản mẫu, người lập trình, người quản lý dự án, chuyên gia CSDL, ... đã tham gia vào tiến trình phát triển. Ta cũng có thể kể cả một số người dùng có tham gia nhiều vào phát triển hệ thống). Tài liệu cho những người này trong suốt thời kỳ nghiên cứu, các tài liệu này thường được gọi là hồ sơ giấy tờ làm việc

b. Các hướng dẫn chung:

- Phần cứng và phần mềm ứng dụng
- Hướng dẫn về các phương thức khai báo
- Về các người sử dụng
- Các hướng dẫn dùng khác

c. Giới thiệu chương trình, trình tự khai thác:

- Danh sách các chương trình
- Mô tả chi tiết
- Trình tự khai thác

d. Đặc trưng của các đầu vào: Đưa ra các mẫu

e. Đặc trưng của các tệp

- Đặc trưng chung
- Cấu trúc tệp
- Các tệp chỉ dẫn

f. Đặc trưng của các đầu ra

- Đặc trưng chung
- Cấu trúc lúc trình bày

g. Hướng dẫn cho các nhân viên điều hành hệ thống

4. Bảo trì hệ thống:

Song song với qui trình kiểm tra thì ta phải tiến hành bảo trì hệ thống

- Sửa các lỗi
- Điều chỉnh theo yêu cầu mới

Cải thiện hiệu năng của hệ thống. Muốn vậy ta phải hiểu được chương trình từ những tài liệu để lại, phải lần ngược dấu vết khi phát hiện lỗi

❖ Bảo trì gồm 4 mức:

- Mức 0: Giới hạn trong chương trình
- Mức 1: Bảo trì mức vật lý (liên quan đến phần cứng)
- Mức 2: Mức truy nhập tổ chức
- Mức 3: Mức quan niệm, khái niệm hay logic

❖ Các loại bảo trì:

- Bảo trì sửa chữa: 17% - 20%
- Bảo trì thích ứng: 18% - 25%

- Bảo trì hoàn thiện: Cải tiến hệ thống để nó chạy tốt hơn, Ổn định hơn, nhanh hơn, ... chiếm từ 50% đến 60%

Tóm lại chu trình phát triển của hệ thống truyền thông như sau:

