



CHƯƠNG 3

THIẾT KẾ CSDL PHÂN TÁN

CHƯƠNG 3: THIẾT KẾ CSDL PHÂN TÁN

NỘI DUNG

3.1 Nội dung thiết kế các hệ thống phân tán

3.2 Các chiến lược phân tán dữ liệu

3.3 Phương pháp thiết kế CSDL phân tán

3.4 Phân mảnh dữ liệu

3.5 Cấp phát tài nguyên trong hệ phân tán

-

MỤC ĐÍCH

Cung cấp cho người lập trình ứng dụng các phương pháp thiết kế một cơ sở dữ liệu phân tán

CHƯƠNG 3: THIẾT KẾ CSDL PHÂN TÁN

Yêu cầu:

- Phải qua bước phân tích trước khi thiết kế. Các bước này phải độc lập với mọi giải pháp cài đặt
- Chọn những vị trí để cài đặt dữ liệu và các chương trình trên mạng máy tính.
- Đối với DBMS phân tán, việc phân tán các ứng dụng đòi hỏi hai điều:
 1. Phân tán DBMS
 2. Phân tán các chương trình ứng dụng chạy trên DBMS đó.

3.1 Nội dung thiết kế một hệ thống phân tán

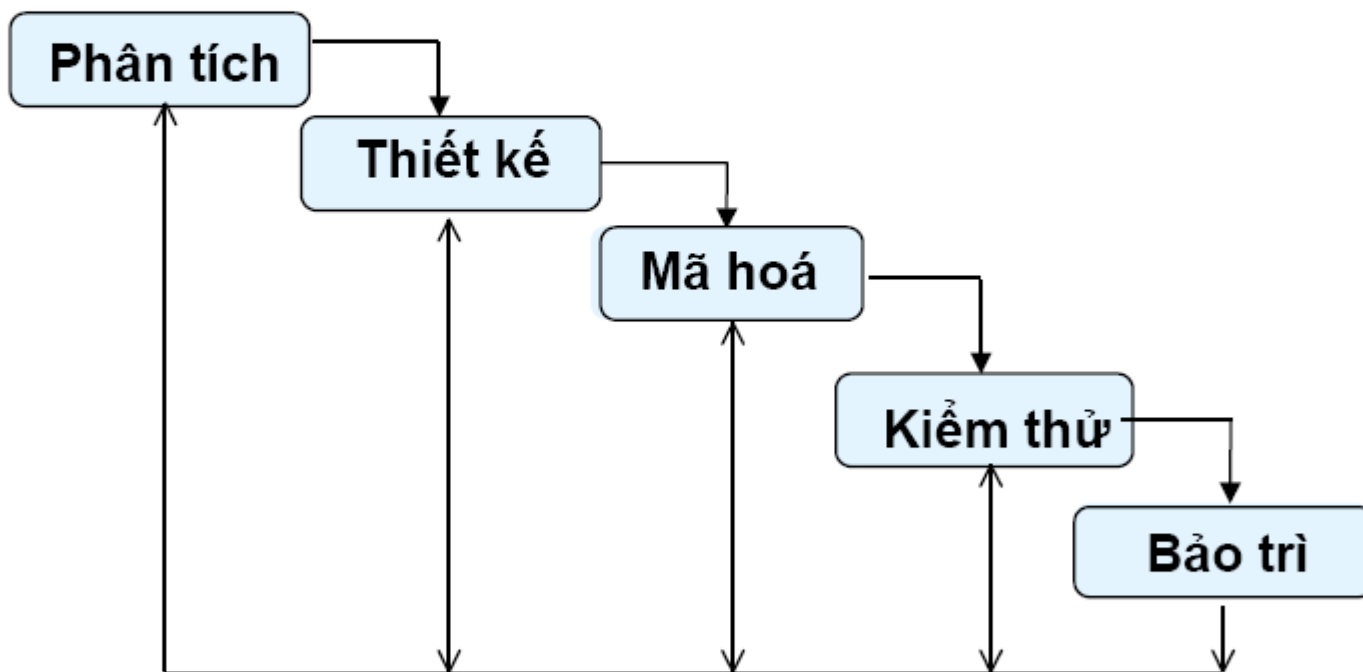
Nhận xét:

- Có nhiều điểm tương đồng với việc thiết kế hệ thống tập trung.
- Điều khác nhau cơ bản là hệ thống được phân bố trên một số địa điểm khác nhau
- Tính khả thi, chu kỳ sống, tính mở, tính sẵn sàng,...
- Thiết kế phần cứng: máy trạm, máy chủ, mạng ,...

Cụ thể?

MÔ HÌNH THÁC NƯỚC

(Water Fall Model)



3.1 Nội dung thiết kế một hệ thống phân tán

3.1.1 Các công việc cần phải làm để thiết kế HT phân tán:

- Xác định kiến trúc mô hình phân tán tổng thể
- Định vị các địa phương cần phân tán, loại hình phân tán sử dụng cho mỗi địa phương (toàn bộ, bản sao, lai,...).
- Tiến hành cân đối các yếu tố được phân tán bao gồm các phần tử dữ liệu và các hoạt động xử lý trên mỗi trạm.
- Thiết kế cơ sở dữ liệu phân tán.
- Thiết kế các chương trình ứng dụng.

3.1 Nội dung thiết kế một hệ thống phân tán

3.1.2 Các sản phẩm yêu cầu sau khi phân tích thiết kế

1. Mô tả các trạm

- . Thông tin địa lý
- . Thiết bị vật lý
- . Thông tin hạ tầng
- . Đặc trưng về con người (trình độ, kỹ năng,...)

2. Mô tả về sử dụng dữ liệu cho mỗi trạm

- . Các phần tử dữ liệu sử dụng từ hệ thống
- . Các phần tử dữ liệu cần phải tạo ra
- . Các phần tử dữ liệu cập nhật
- . Các phần tử dữ liệu xóa

3.1 Nội dung thiết kế một hệ thống phân tán

3.1.2 Các sản phẩm yêu cầu sau khi phân tích thiết kế(tiếp)

3. Mô tả quá trình nghiệp vụ cho mỗi trạm

- . Danh sách các xử lý (sơ đồ chức năng) ở các trạm
- . Mô tả các xử lý

4. Các thỏa thuận về phương án kiến trúc hệ thống cho mỗi trạm, cho nhu cầu về dữ liệu và xử lý của trạm đó

- . Có cần hay không về các trợ giúp không phải kỹ thuật
- . Có cần hay không về hệ thống địa phương, về nối mạng
- . Có cần hay không về các cấu hình phân tán khác

3.2 Các chiến lược phân tán dữ liệu

3.2 Các chiến lược phân tán dữ liệu

- Việc định vị và phân tán dữ liệu ở các nút trong một mạng máy tính sẽ quyết định tính hiệu quả và đúng đắn của hệ thống phân tán.
- Có 4 chiến lược phân tán dữ liệu cơ bản:
 - Tập trung dữ liệu
 - Chia nhỏ dữ liệu
 - Sao lặp dữ liệu
 - Phương thức lai

3.2 Các chiến lược phân tán dữ liệu

3.2.1 Tập trung dữ liệu:

Tất cả các dữ liệu được tập trung một chỗ. Cách này đơn giản nhưng có 3 nhược điểm:

- Dữ liệu không sẵn sàng cho người sử dụng truy nhập từ xa
- Chi phí truyền thông lớn, thường làm cực đại việc truy nhập dữ liệu tới nơi tập trung.
- Toàn bộ hệ thống ngừng khi cơ sở dữ liệu bị sự cố

3.2.2 Chia nhỏ dữ liệu:

- Cơ sở dữ liệu được chia thành các phần nhỏ liên kết nhau (không trùng lặp).
- Mỗi phần dữ liệu được đưa đến các trạm một cách thích hợp để sử dụng.

3.2 Các chiến lược phân tán dữ liệu

3.2.3 Sao lặ dữ liệu:

- CSDL được nhân thành nhiều bản từng phần hoặc đầy đủ và được đặt ở nhiều trạm trên mạng.
- Nếu bản sao của CSDL được lưu giữ tại mọi trạm của hệ thống ta có trường hợp sao lặ đầy đủ.
- Hiện nay có nhiều kỹ thuật mới cho phép tạo bản sao không đầy đủ phù hợp với yêu cầu dữ liệu ở mỗi trạm và một bản đầy đủ được quản lý ở server.
- Sau một khoảng thời gian nhất định các bản sao được làm đồng bộ với bản chính bằng một ứng dụng nào đó.

3.2 Các chiến lược phân tán dữ liệu

3.2.4 Phương thức lai:

- Cơ sở dữ liệu được phân thành nhiều phần: **quan trọng** và **không quan trọng**.
- Phần ít quan trọng được lưu giữ một nơi
- Phần quan trọng được lưu trữ ở nhiều nơi khác.

3.3 PHƯƠNG PHÁP THIẾT KẾ CSDL PHÂN TÁN

3.3.1 Sơ đồ thiết kế tổng thể cơ sở dữ liệu phân tán

Hiện nay chưa có một kỹ thuật cụ thể nào nói một cách chi tiết việc thiết kế một CSDL phân tán. Tuy nhiên, một cách tổng quát chúng ta có thể thiết kế CSDL phân tán theo các bước sau:



Sơ đồ thiết kế tổng thể

3.3 PHƯƠNG PHÁP THIẾT KẾ CSDL PHÂN TÁN

1. Thiết kế lược đồ quan hệ tổng thể:

- Thiết kế các quan hệ tổng thể
- Mô tả toàn bộ dữ liệu sẽ được dùng trong hệ thống

1. Thiết kế phân đoạn: thực hiện chia nhỏ dữ liệu thành các phần.

2. Thiết kế định vị các đoạn:

- là quá trình thực hiện ánh xạ các đoạn vào các trạm khác nhau
- Tạo các ảnh vật lý tại các trạm.
- Các đoạn dữ liệu được đưa vào các vị trí lưu trữ thích hợp với yêu cầu hoạt động thực tế của hệ thống.

1. Thiết kế cơ sở dữ liệu vật lý: thiết kế dữ liệu vật lý cho các quan hệ tại các trạm

3.3 PHƯƠNG PHÁP THIẾT KẾ CSDL PHÂN TÁN

3.3.2 Các phương pháp thiết kế CSDL phân tán

Có 2 phương pháp thiết kế CSDL phân tán

- Phương pháp tiếp cận từ trên xuống
- Phương pháp tiếp cận từ dưới lên.

3.3 PHƯƠNG PHÁP THIẾT KẾ CSDL PHÂN TÁN

a. Phương pháp thiết kế từ trên xuống

- Thiết kế từ tổng thể đến riêng biệt
- Phân rã một hệ thống lớn thành các hệ thống con
- Phân tích các yêu cầu nhằm định nghĩa môi trường hệ thống
- Thu thập các yêu cầu về dữ liệu và nhu cầu xử lý của các trạm có sử dụng CSDL.

3.3 PHƯƠNG PHÁP THIẾT KẾ CSDL PHÂN TÁN

- **Thiết kế view**: xây dựng khung nhìn dữ liệu cho người sử dụng ở các trạm.
- **Thiết kế mức quan niệm**: là một tiến trình kiểm tra và xác định rõ hai nhóm quan hệ: *phân tích thực thể* và *phân tích chức năng*.
 - + *Phân tích thực thể*: xác định các tập thực thể, các thuộc tính và các mối quan hệ giữa chúng.
 - + *Phân tích chức năng*: xác định các chức năng của hệ thống và đưa ra các chức năng cơ sở.

3.3 PHƯƠNG PHÁP THIẾT KẾ CSDL PHÂN TÁN

- **Thiết kế phân tán**: bao gồm hai phần:
 - + *Thiết kế phân đoạn*
 - + *Thiết kế định vị*
- **Thiết kế lược đồ quan niệm địa phương**: tạo ra các lược đồ mức quan niệm tại các địa phương
- **Thiết kế vật lý**: thực hiện ánh xạ lược đồ mức quan niệm tại các địa phương ra các đơn vị lưu trữ vật lý
- **Quan sát và kiểm tra**: kiểm tra các giai đoạn của quá trình thiết kế cơ sở dữ liệu

3.3 PHƯƠNG PHÁP THIẾT KẾ CSDL PHÂN TÁN

b. Phương pháp thiết kế từ dưới lên

Nhận xét

- Phương pháp thiết kế trên xuống thực sự có hiệu quả khi xây dựng một hệ thống mới.
- Trong thực tế, một số CSDL đã tồn tại trước, được tổ chức trong môi trường tập trung và CSDL phân tán được phát triển bằng cách liên kết chúng lại thành một CSDL mới thống nhất (Các DBMS địa phương khác nhau đã được sử dụng)

Cách thiết kế

1. Chọn một mô hình dữ liệu chung để mô tả lược đồ tổng thể
2. Chuyển mỗi lược đồ địa phương theo mô hình dữ liệu chung đã chọn
3. Tích hợp các lược đồ địa phương vào lược đồ tổng thể

3.4 PHÂN MẢNH DỮ LIỆU

0. Nhắc lại các phép toán đại số quan hệ và ngôn ngữ SQL
1. Tại sao cần phải phân mảnh?
2. Làm thế nào để thực hiện phân mảnh?
3. Phân mảnh nên thực hiện đến mức độ nào?
4. Có cách gì kiểm tra tính đúng đắn của việc phân mảnh?
5. Việc cấp phát các mảnh dữ liệu như thế nào?
6. Những thông tin nào sẽ cần thiết cho việc phân mảnh và cấp phát?

SQL

Sơ lược về ngôn ngữ SQL (**S**tructured **q**uery **l**anguage)

- SQL trước kia được gọi là SEQUEL
- IBM phát triển ở San Jose,
- Là một ngôn ngữ phi thủ tục
- Mục đích để sử dụng trong CSDL thử nghiệm System R

Câu lệnh SELECT

Cú pháp:

SELECT $R_1.A_1, R_2.A_2, \dots, R_i.A_i$

FROM R_1, R_2, \dots, R_k

WHERE Ψ

Ý nghĩa: $\Pi_{R_1.A_1, R_2.A_2, \dots, R_i.A_i} (\sigma_{\Psi} (R_1 \times \dots \times R_k))$

3.4 PHÂN MẢNH DỮ LIỆU

Phân mảnh quan hệ là gì?

Việc chia một quan hệ thành nhiều quan hệ nhỏ hơn được gọi là phân mảnh quan hệ.

3.4.1 Các lý do phân mảnh

- Khung nhìn hoặc đơn vị truy xuất của các ứng dụng không phải là toàn bộ quan hệ mà thường là một phần quan hệ.
- Việc phân rã một quan hệ thành nhiều mảnh, mỗi mảnh được xử lý như một đơn vị, sẽ cho phép thực hiện nhiều giao dịch đồng thời.
- Việc phân mảnh các quan hệ sẽ cho phép thực hiện song song một câu vấn tin bằng cách chia nó ra thành một tập các câu vấn tin con hoạt tác trên các mảnh.

3.4 PHÂN MẢNH DỮ LIỆU

- Nếu các ứng dụng có các khung nhìn được định nghĩa trên một quan hệ cho trước nằm tại những vị trí khác thì có hai cách chọn lựa đơn vị phân tán:
 - + hoặc là toàn bộ quan hệ
 - + hoặc quan hệ được lưu ở một vị trí có chạy ứng dụng.

Nhận xét: Chọn lựa thứ nhất gây ra một số lượng lớn các truy xuất không cần thiết đến dữ liệu ở xa. Chọn lựa sau sẽ gây ra nhiều vấn đề khi cập nhật và lãng phí không gian lưu trữ.

3.4 PHÂN MẢNH DỮ LIỆU

Khuyết điểm của việc phân mảnh:

- Nếu ứng dụng có những yêu cầu ngăn cản việc phân rã thành các mảnh để được sử dụng độc quyền, thì những ứng dụng có các khung nhìn được định nghĩa trên nhiều mảnh sẽ bị giảm hiệu suất hoạt động.
- Nếu một khung nhìn đòi hỏi thông tin ở nhiều mảnh thì việc truy xuất dữ liệu để nối lại sẽ có chi phí cao.
- Kiểm soát ngữ nghĩa dữ liệu (semantic data control): Do kết quả của phân mảnh, các thuộc tính tham gia vào một phụ thuộc có thể bị phân rã vào các mảnh khác nhau và được cấp phát cho những vị trí khác nhau. Trong trường hợp này, một nhiệm vụ đơn giản như kiểm tra các phụ thuộc cũng phải thực hiện truy tìm dữ liệu ở nhiều vị trí.

3.4 PHÂN MẢNH DỮ LIỆU

3.4.2 Các kiểu phân mảnh

- Phân mảnh ngang (horizontal fragmentation)
- Phân mảnh dọc (vertical fragmentation).
- Phân mảnh hỗn hợp (hibrid fragmentation)

Chú ý: Quá trình phân mảnh phải được gắn liền với vấn đề cấp phát dữ liệu và bài toán cụ thể như thế nào.

3.4 PHÂN MẢNH DỮ LIỆU

3.4.2.1 Phân mảnh ngang: phân mảnh ngang một quan hệ tổng thể n-bộ R là tách R thành các quan hệ con n-bộ R_1, R_2, \dots, R_k sao cho quan hệ R có thể được khôi phục lại từ các quan hệ con này bằng phép hợp: $R = R_1 \cup R_2 \cup \dots \cup R_k$

Có hai loại phân mảnh ngang:

Phân mảnh ngang nguyên thủy (primary horizontal fragmentation): phân mảnh ngang nguyên thủy của một quan hệ được thực hiện dựa trên các vị từ được định nghĩa trên quan hệ đó.

Phân mảnh ngang dẫn xuất (derived horizontal fragmentation): phân mảnh ngang dẫn xuất của một quan hệ được thực hiện dựa trên các vị từ được định nghĩa trên quan hệ khác.

Như vậy, trong phân mảnh ngang tập các vị từ đóng một vai trò quan trọng.

3.4 PHÂN MẢNH DỮ LIỆU

3.4.2.2 Phân mảnh dọc:

Phân mảnh dọc một quan hệ tổng thể n-bộ R là tách R thành các quan hệ con R_1, R_2, \dots, R_k sao cho quan hệ R có thể được khôi phục lại từ các quan hệ con này bằng phép nối:

$$R = R_1 \bowtie R_2 \bowtie \dots \bowtie R_k$$

3.4.2.3 Phân mảnh hỗn hợp:

là kết hợp cả phân mảnh ngang và phân mảnh dọc

3.4 PHÂN MẢNH DỮ LIỆU

3.4.3 Các yêu cầu của việc phân mảnh

Việc phân mảnh một quan hệ tổng thể cũng phải tuân theo một số quy tắc nhất định để khi tái thiết lại quan hệ cũ vẫn bảo đảm ngữ nghĩa của nó.

Một phương pháp thiết kế các phân mảnh đúng đắn phải thỏa mãn ba tính chất sau:

- a. *Tính đầy đủ (completeness)*
- b. *Tính tái thiết được (reconstruction)*
- c. *Tính tách biệt (disjointness):*

3.4 PHÂN MẢNH DỮ LIỆU

a. Tính đầy đủ: Nếu một quan hệ R được phân rã thành các mảnh R_1, R_2, \dots, R_k thì mỗi mục dữ liệu có trong R phải có trong ít nhất một mảnh R_i nào đó.

b. Tính tái thiết được:

- Nếu một quan hệ R được phân rã thành các mảnh R_1, R_2, \dots, R_k thì phải tồn tại một toán tử θ sao cho $R = \theta(R_i), \forall i$.
- Toán tử θ thay đổi tùy theo từng loại phân mảnh.
- Trong thực tế khi các mảnh được phân mảnh ngang thì θ là phép hợp, phân mảnh dọc thì θ là phép nối và phân mảnh hỗn hợp thì θ là phép nửa nối .

3.4 PHÂN MẢNH DỮ LIỆU

c. Tính tách biệt:

- Nếu một quan hệ R được phân mảnh ngang thành các quan hệ R_1, R_2, \dots, R_k và mục dữ liệu t_i nằm trong mảnh R_i thì nó sẽ không nằm trong một mảnh $R_k, k \neq i$.
- Tiêu chuẩn này bảo đảm các mảnh ngang phải được tách rời nhau.
- Nếu quan hệ được phân mảnh dọc thì thuộc tính chung phải được lặp lại trong mỗi mảnh. Do đó, trong trường hợp phân mảnh dọc tính tách biệt chỉ được định nghĩa trên các trường không phải là thuộc tính chung của quan hệ.

3.4 PHÂN MẢNH DỮ LIỆU

Ví dụ: Xét cơ sở dữ liệu của một công ty phần mềm được tổ chức như sau:

- **NHANVIEN (MANV, TENNV, CHUCVU):** quan hệ này chứa dữ liệu về nhân viên của công ty.
- **TLUONG (CHUCVU, LUONG):** quan hệ này chứa dữ liệu liên quan về lương và chức vụ của nhân viên.
- **DUAN (MADA, TENDA, NGANSACH):** quan hệ này chứa dữ liệu về các dự án mà công ty đang thực hiện.
- **HOSO (MANV, MADA, NHIEMVU, THOIGIAN):** quan hệ này chứa dữ liệu về hồ sơ của nhân viên được phân công thực hiện dự án).

Cơ sở dữ liệu của một công ty máy tính

NHANVIEN (E)

MANV	TENNV	CHUCVU
A1	Nam	Phân tích HT
A2	Trung	Lập trình viên
A3	Đông	Phân tích HT
A4	Bắc	Phân tích HT
A5	Tây	Lập trình viên
A6	Hùng	Kỹ sư điện
A7	Dũng	Phân tích HT
A8	Chiến	Thiết kế DL

HOSO (G)

MANV	MADA	NHIEMVU	THOIGIAN
A1	D1	Quản lý	12
A2	D1	Phân tích	34
A2	D2	Phân tích	6
A3	D3	Kỹ thuật	12
A3	D4	Lập trình	10
A4	D2	Quản lý	6
A5	D2	Quản lý	20
A6	D4	Kỹ thuật	36
A7	D3	Quản lý	48
A8	D3	Lập trình	15

DUAN (J)

MADA	TENDA	NGANSACH
D1	CSDL	20000
D2	CÀI ĐẶT	12000
D3	BẢO TRÌ	28000
D4	PHÁT TRIỂN	25000

TLUONG (S)

CHUCVU	LUONG
Kỹ sư điện	1000
Phân tích HT	2500
Lập trình viên	3000
Thiết kế DL	4000

3.4 PHÂN MẢNH DỮ LIỆU

Ví dụ về phân mảnh ngang: xét các phép toán đại số quan hệ sau:

DUAN 1 = $\sigma_{\text{NGANSACH} \leq 20000}$ (DUAN)

DUAN 2 = $\sigma_{\text{NGANSACH} > 20000}$ (DUAN)

DUAN 1

MADA	TENDA	NGANSACH
D1	CSDL	20000
D2	CÀI ĐẶT	12000

DUAN 2

MADA	TENDA	NGANSACH
D3	BẢO TRÌ	28000
D4	PHÁT TRIỂN	25000

Để thấy, các mảnh thỏa mãn *tính tái thiết được* và *tính đầy đủ*

$\text{DUAN 1} \subseteq \text{DUAN}$; $\text{DUAN 2} \subseteq \text{DUAN}$; $\text{DUAN} = \text{DUAN 1} \cup \text{DUAN 2}$

3.4 PHÂN MẢNH DỮ LIỆU

Ví dụ về phân mảnh dọc: xét các phép toán đại số quan hệ sau:

$$\text{DUAN 3} = \Pi_{\$1, \$3} \text{DUAN}; \quad \text{DUAN 4} = \Pi_{\$1, \$2} \text{DUAN}$$

MADA	NGANSACH	MADA	TENDA
D1	20000	D1	CSDL
D2	12000	D2	CÀI ĐẶT
D3	28000	D3	BẢO TRÌ
D4	25000	D4	PHÁT TRIỂN

Để thấy, các mảnh thỏa mãn *tính tái thiết được* và *tính đầy đủ*

$$\text{DUAN 3} \subseteq \text{DUAN}; \quad \text{DUAN 4} \subseteq \text{DUAN}; \quad \text{DUAN} = \text{DUAN 3} \bowtie \text{DUAN 4}$$

Ví dụ về phân mảnh hỗn hợp: xét các phép toán đại số quan hệ sau:

$$\text{DUAN 5} = \sigma_{\text{NGANSACH} \leq 20000} (\Pi_{\$1, \$3} \text{DUAN}) \quad \text{Xem lại}$$

$$\text{DUAN 6} = \sigma_{\text{NGANSACH} > 20000} (\Pi_{\$1, \$3} \text{DUAN})$$

DUAN 5		DUAN 6	
TENDA	NGANSACH	TENDA	NGANSACH
CSDL	20000	CSDL	28000
CÀI ĐẶT	12000	CÀI ĐẶT	25000

3.5 Cấp phát tài nguyên trong hệ phân tán

3.5.1 Bài toán cấp phát (allocation problem):

Giả sử có một tập các mảnh $F = \{F_1, F_2, \dots, F_k\}$ và một mạng máy tính bao gồm các vị trí $S = \{S_1, S_2, \dots, S_m\}$ trên đó có một tập các ứng dụng $Q = \{Q_1, Q_2, \dots, Q_q\}$ đang thực thi.

Hãy tìm một phân phối tối ưu các mảnh F cho các vị trí S .

Một phân phối được gọi là tối ưu nếu thỏa mãn hai yếu tố sau:

3.5 Cấp phát tài nguyên trong hệ phân tán

a. Chi phí nhỏ nhất: hàm chi phí bao gồm:

- Chi phí lưu mỗi mảnh dữ liệu F_i tại vị trí S_j
- Chi phí vận tin F_i tại vị trí S_j
- Chi phí cập nhật F_i tại tất cả các vị trí có chứa nó
- Chi phí truyền dữ liệu.

Bài toán cấp phát sẽ tìm một lược đồ cấp phát với hàm chi phí là cực tiểu.

b. Hiệu quả: chiến lược cấp phát được thiết kế nhằm cực tiểu hóa thời gian thực hiện và tăng tối đa lưu lượng hệ thống tại mỗi vị trí.

3.5 Cấp phát tài nguyên trong hệ phân tán

Bài toán cấp phát tổng quát, ký hiệu DAP (Database Allocation Problem), là một bài toán NP-đầy đủ. Vì thế hầu hết các nghiên cứu đã được dành cho việc tìm ra được các thuật giải heuristic để có được lời giải tối ưu cho loại bài toán này.

Hiện nay chưa có một mô hình heuristic tổng quát nào nhận một tập các mảnh và sinh ra một chiến lược cấp phát gần tối ưu ứng với các ràng buộc cho trước mà chỉ mới đưa ra một số giả thiết đơn giản hóa và dễ áp dụng cho một số cách đặt vấn đề đơn giản.

3.5 Cấp phát tài nguyên trong hệ phân tán

3.5.2 Thông tin cấp phát

Ở giai đoạn cấp phát, chúng ta cần các thông tin định lượng về cơ sở dữ liệu, về các ứng dụng chạy trên đó, về cấu trúc mạng, về khả năng xử lý và giới hạn lưu trữ của mỗi vị trí trên mạng.

- a. Thông tin về cơ sở dữ liệu
- b. Thông tin về ứng dụng
- c. Thông tin về vị trí
- d. Thông tin về mạng

Câu hỏi cuối chương

1. Các vấn đề cần phải làm để thiết kế một HT phân tán
2. Các sản phẩm yêu cầu sau khi phân tích thiết kế một HT phân tán
3. Các chiến lược phân tán dữ liệu
4. Nội dung của phương pháp thiết kế từ trên xuống
5. Các kiểu phân mảnh và các yêu cầu của việc phân mảnh.

Cho ví dụ.

- ~~6. Bài toán cấp phát~~

CHƯƠNG I. TỔNG QUAN VỀ CSDL PHÂN TÁN



HẾT CHƯƠNG 3