

Chương 4

Thiết kế CSDL phân tán



Nội dung

- ❖ Các bước thiết kế CSDL.
- ❖ Mục tiêu của thiết kế CSDL phân tán.
- ❖ Các cách tiếp cận thiết kế CSDL.
- ❖ Thiết kế phân mảnh ngang chính.
- ❖ Thiết kế phân mảnh ngang dẫn xuất.



Các bước thiết kế cơ sở dữ liệu

❖ Thiết kế CSDL tập trung

- ▶ Thiết kế lược đồ ý niệm.
- ▶ Thiết kế CSDL vật lý.

❖ Thiết kế CSDL phân tán

- ▶ Thiết kế lược đồ toàn cục.
- ▶ Thiết kế phân mảnh.
- ▶ Thiết kế định vị mảnh.
- ▶ Thiết kế CSDL vật lý cục bộ.

Các bước thiết kế cơ sở dữ liệu

- ❖ Thiết kế CSDL phân tán: cần phải hiểu biết thật chính xác về các yêu cầu của ứng dụng, nhất là đối với các ứng dụng **quan trọng hơn**.
- ❖ Cần quan tâm đến:
 - ▶ Nơi chạy ứng dụng.
 - ▶ Tần suất chạy ứng dụng.
 - ▶ Số lượng, loại và sự phân tán của các truy xuất trong mỗi ứng dụng đến mỗi đối tượng dữ liệu cần thiết.

Mục tiêu của thiết kế phân tán dữ liệu

❖ Tính cục bộ xử lý

- ▶ *processing locality*
- ▶ Phân tán dữ liệu để làm cực đại hóa tính cục bộ xử lý là đặt dữ liệu càng gần các ứng dụng sử dụng các dữ liệu này càng tốt.
- ▶ Một quan hệ không là một đơn vị phân tán.
- ▶ Tính cục bộ xử lý dựa vào các tham chiếu cục bộ và các tham chiếu từ xa.
- ▶ Tính cục bộ hoàn toàn (*complete locality*).

❖ Tính sẵn sàng và độ tin cậy của dữ liệu

- ▶ Tính sẵn sàng (*availability*).
- ▶ Độ tin cậy (*reliability*).

Mục tiêu của thiết kế phân tán dữ liệu

❖ Điều phối tải làm việc

- ▶ Cực đại hóa mức độ thực hiện song song các ứng dụng.
- ▶ Điều phối tải làm việc có thể ảnh hưởng ngược lại với tính cục bộ xử lý.
- ▶ Tính đồng thời nội truy vấn.

❖ Chi phí lưu trữ và khả năng lưu trữ có sẵn

- ▶ Khả năng lưu trữ có sẵn tại mỗi nơi.
- ▶ Chi phí lưu trữ dữ liệu là không đáng kể so với các chi phí CPU, nhập / xuất và truyền thông của các ứng dụng.



Cách tiếp cận từ trên xuống

❖ Thiết kế từ trên xuống

- ▶ *top-down design*
- ▶ Thiết kế lược đồ toàn cục.
- ▶ Thiết kế phân mảnh CSDL.
- ▶ Định vị các mảnh tại các nơi.
- ▶ Thiết kế dữ liệu vật lý đặt tại mỗi nơi.

Cách tiếp cận từ dưới lên

❖ Thiết kế từ dưới lên

- ▶ *bottom-up design*
- ▶ Chọn một mô hình CSDL chung để mô tả lược đồ toàn cục của CSDL.
- ▶ Chuyển đổi mỗi lược đồ cục bộ thành mô hình dữ liệu chung.
- ▶ Tích hợp các lược đồ cục bộ thành một lược đồ toàn cục chung.

Các yêu cầu thông tin

- ❖ Các yếu tố trong thiết kế tối ưu ảnh hưởng đến các quyết định phân tán.
 - ▶ Tổ chức luận lý của CSDL.
 - ▶ Vị trí của các ứng dụng.
 - ▶ Các đặc điểm truy xuất CSDL của các ứng dụng.
 - ▶ Các đặc tính của các hệ thống máy tính tại mỗi nơi.



Các yêu cầu thông tin

❖ Các loại thông tin để thiết kế phân tán

- ▶ Thông tin về CSDL
- ▶ Thông tin về ứng dụng
- ▶ Thông tin về mạng truyền thông
- ▶ Thông tin về hệ thống máy tính

Thiết kế phân mảnh ngang

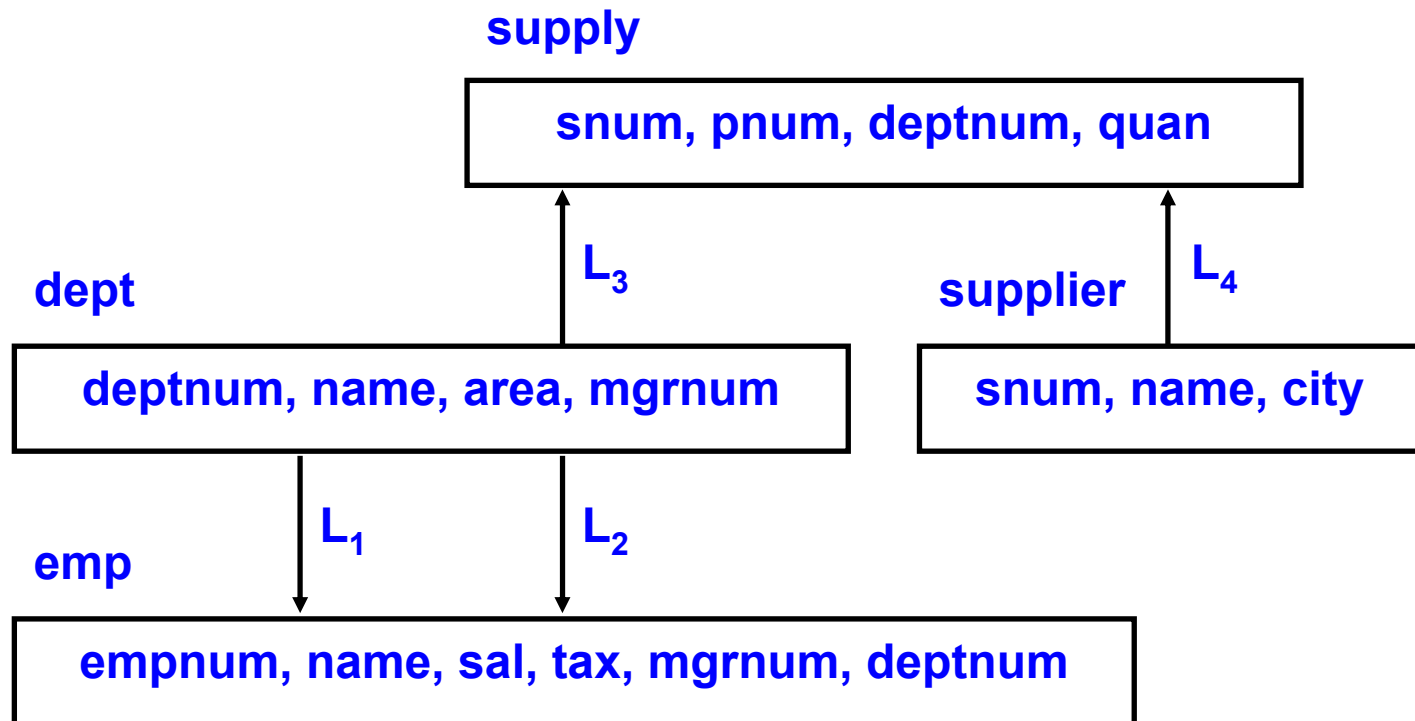
- ❖ Mỗi mảnh là một tập hợp con gồm các bộ của quan hệ.
- ❖ Phân mảnh ngang chính là phân chia một quan hệ dựa vào các vị từ định tính được định nghĩa trên quan hệ này.
- ❖ Phân mảnh ngang dẫn xuất là phân chia một quan hệ dựa vào các vị từ định tính được định nghĩa trên một quan hệ khác.

Thiết kế phân mảnh ngang

❖ Thông tin về CSDL

- ▶ Trong lược đồ ý niệm toàn cục, các quan hệ được kết với nhau.
- ▶ Trong mô hình liên kết thực thể (*ER model*):
 - Quan hệ chủ hoặc quan hệ nguồn
 - Quan hệ bộ phận hoặc quan hệ đích
 - Các hàm *owner* và *member*

Thiết kế phân mảnh ngang



$owner(L_1) = dept$

$member(L_1) = emp$

Hình 4.2. Biểu diễn các mối liên kết giữa các quan hệ dùng các đường liên kết .

Thiết kế phân mảnh ngang

❖ Thông tin về ứng dụng

- ▶ Các vị từ được sử dụng trong các truy vấn.
- ▶ Chỉ phân tích các ứng dụng quan trọng để xác định các vị từ này.
- ▶ Giả sử phân mảnh ngang quan hệ $R(A_1, A_2, \dots, A_n)$, với A_i là thuộc tính được định nghĩa trên miền D_i .

Thiết kế phân mảnh ngang

❖ Thông tin về ứng dụng

- ▶ **Vị từ đơn giản** (*simple predicate*) p_j được định nghĩa trên R có dạng:

$A_j \theta \text{ value}$

θ là một trong các phép so sánh $=, \neq, <, \leq, >, \geq$

value được chọn từ miền trị của A_j ($\text{value} \in D_j$)

- ▶ Ký hiệu P_r là tập các vị từ đơn giản được định nghĩa trên quan hệ R . Các phần tử của P_r được ký hiệu là p_j .

Thiết kế phân mảnh ngang

❖ Thông tin về ứng dụng

- ▶ **Vị từ giao tối thiểu** (*minterm predicate*) m_j đối với tập các vị từ đơn giản $P_r = \{p_1, p_2, \dots, p_m\}$ là một tổ hợp giao của tất cả các vị từ xuất hiện trong P_r (ở dạng thông thường hoặc ở dạng phủ định) sao cho m_j không bị mâu thuẫn.

$$m_j = \bigwedge p_i^*, 1 \leq i \leq m$$

với $p_i^* = p_i$ hoặc $p_i^* = \neg p_i$ và $m_j \neq \text{false}$

- ▶ Gọi tập các vị từ giao tối thiểu là:

$$M = \{m_1, m_2, \dots, m_z\}$$

Thiết kế phân mảnh ngang chính

- ❖ Mảnh ngang chính được xác định bằng phép chọn trên quan hệ toàn cục.

$$R_i = \sigma_{F_i}(R); \quad 1 \leq i \leq n$$

- ▶ F_i là điều kiện chọn của mảnh R_i
- ▶ Nếu F_i ở dạng chuẩn giao thì nó là một vị từ giao tối thiểu m_i
- ❖ Tính đúng đắn của phân mảnh ngang chính: mỗi bộ của quan hệ toàn cục được đưa vào trong một và chỉ một mảnh.

Thiết kế phân mảnh ngang chính

- ❖ Xác định phân mảnh ngang chính của một quan hệ toàn cục là xác định một tập các **vị từ chọn** (*selection predicate*) đầy đủ và tách biệt.
- ❖ Các bộ thuộc cùng một mảnh phải được tham chiếu giống nhau trong tất cả các ứng dụng.
- ❖ **Mảnh ngang** (*horizontal fragment*) hoặc **mảnh giao tối thiểu** (*minterm fragment*) R_i bao gồm tất cả các bộ của R thỏa mãn vị từ giao tối thiểu m_i .



Thiết kế phân mảnh ngang chính

❖ Các bước thiết kế phân mảnh ngang

- ▶ **Bước 1:** Tìm tập các vị từ chọn $P_{r'}$ là đầy đủ và tối thiểu.
- ▶ **Bước 2:** Tìm tập các vị từ giao tối thiểu có thể được định nghĩa trên các vị từ của $P_{r'}$

Thiết kế phân mảnh ngang chính

- ❖ Một vị từ đơn giản p_i được gọi là **thích hợp** (*relevant*) đối với một tập P_r các vị từ đơn giản, nếu tồn tại ít nhất hai vị từ giao tối thiểu m_i và m_j của P_r mà các biểu thức của chúng chỉ khác nhau ở p_i (tức là m_i chứa p_i và m_j chứa $\neg p_i$) và tồn tại ít nhất một ứng dụng tham chiếu khác nhau đến hai mảnh f_i và f_j (tương ứng với m_i và m_j).

Thiết kế phân mảnh ngang chính

- ❖ Một tập các vị từ đơn giản P_r được gọi là **đầy đủ** (*complete*) nếu và chỉ nếu bất kỳ hai bộ nào thuộc bất kỳ mảnh giao tối thiểu nào được định nghĩa theo P_r thì bất kỳ ứng dụng nào đều tham chiếu đến hai bộ này với cùng một xác suất.
- ❖ Một tập các vị từ đơn giản P_r được gọi là **tối thiểu** (*minimal*) nếu tất cả các vị từ của nó là các vị từ thích hợp.
- ❖ Cho $P_r = \{p_1, p_2, \dots, p_m\}$ là một tập các vị từ đơn giản. Để cho P_r biểu diễn phân mảnh đúng đắn và hiệu quả thì P_r phải **đầy đủ** và **tối thiểu**.

Thiết kế phân mảnh ngang dẫn xuất

- ❖ Phân mảnh ngang dẫn xuất được định nghĩa trên các quan hệ bộ phận của đường liên kết theo phép chọn trên quan hệ chủ của đường liên kết này.
- ❖ Đường liên kết giữa quan hệ chủ và quan hệ bộ phận được định nghĩa là một phép kết bằng.
- ❖ Một phép kết bằng có thể được thực hiện bằng các phép nửa kết.

Thiết kế phân mảnh ngang dẫn xuất

- ❖ Xét đường liên kết L với $owner(L) = S$ và $member(L) = R$, các mảnh ngang dẫn xuất của R được định nghĩa như sau:

$$R_i = R \triangleright \triangleleft_F S_i, \quad 1 \leq i \leq n$$

- ▶ n là số lượng lớn nhất các mảnh được định nghĩa trên R .
- ▶ $S_i = \sigma_{F_i}(S)$ với F_i là công thức dùng để định nghĩa mảnh ngang chính S_i
- ▶ F là điều kiện nửa kết.

Thiết kế phân mảnh ngang dẫn xuất

- ❖ Để thực hiện phân mảnh ngang dẫn xuất, cần có:
 - ▶ Tập các mảnh của quan hệ chủ
 - ▶ Quan hệ bộ phận
 - ▶ Tập các vị từ nửa kết giữa quan hệ chủ và quan hệ bộ phận.

Thiết kế phân mảnh ngang dẫn xuất

- ❖ **Phép kết phân tán** (*distributed join*) là một phép kết giữa các quan hệ được phân mảnh ngang.

$$R \triangleright \triangleleft_F S = (\cup_i R_i) \triangleright \triangleleft_F (\cup_j S_j)$$

$$R \triangleright \triangleleft_F S = \cup_{ij} (R_i \triangleright \triangleleft_F S_j)$$

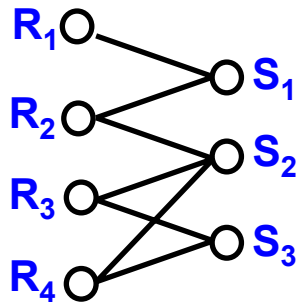
- ▶ Có thể suy diễn để xác định một số phép kết từng phần $R_i \triangleright \triangleleft_F S_j = \emptyset$.
- ❖ Phép kết phân tán được biểu diễn bằng **đồ thị kết** (*join graph*).



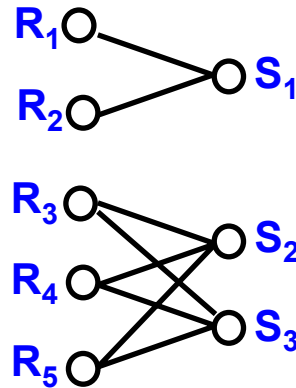
Thiết kế phân mảnh ngang dẫn xuất

- ❖ Đồ thị kết được gọi là **hoàn toàn** (*total*) nếu nó chứa tất cả các cạnh có thể có giữa các mảnh của R và S .
- ❖ Đồ thị kết được gọi là **suy giảm** (*reduced*) nếu không có một số cạnh giữa các mảnh của R và S .
 - ▶ Đồ thị kết suy giảm được gọi là **phân hoạch** (*partitioned*) nếu nó bao gồm hai hoặc nhiều đồ thị con và không có các cạnh giữa chúng.
 - ▶ Đồ thị kết suy giảm được gọi là **đơn giản** (*simple*) nếu nó là phân hoạch và mỗi đồ thị con có đúng một cạnh.

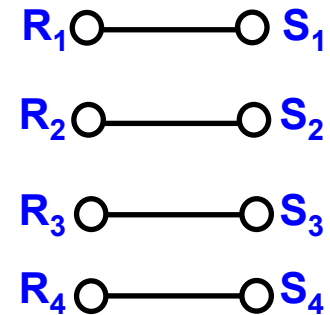
Thiết kế phân mảnh ngang dẫn xuất



(a) Đồ thị kết



(b) Đồ thị kết phân hoạch



(c) Đồ thị kết đơn giản

Hình 4.3. Các loại đồ thị kết.

Thiết kế phân mảnh ngang dẫn xuất

- ❖ Có thể có nhiều đường liên kết đến một quan hệ R và có nhiều cách phân mảnh ngang dẫn xuất cho R dựa trên hai tiêu chuẩn:
 - ▶ Sự phân mảnh có các đặc điểm kết tốt hơn.
 - ▶ Sự phân mảnh được sử dụng trong nhiều ứng dụng hơn.