CƠ SỞ DỮ LIỆU PHÂN TÁN

NGUYỄN THỊ THANH THỦY

Nội dung môn học

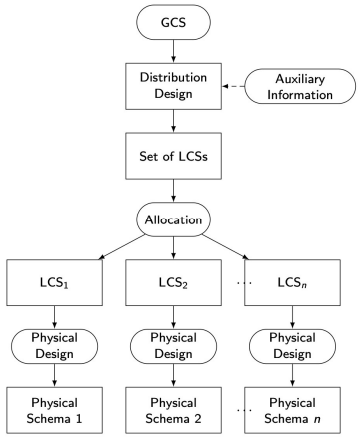
- Giới thiệu
- Thiết kế CSDL phân tán
- Điều khiển dữ liệu phân tán
- Xử lý truy vấn phân tán
- Xử lý giao dịch phân tán

- Nhân bản dữ liệu
- Tích hợp CSDL – Các hệ thống đa CSDL
- Các hệ thống CSDL song song
- Quản lý dữ liệu ngang hàng (Peer-to-Peer)
- Xử lý dữ liệu lớn
- NoSQL, NewSQL và Polystores
- Quản lý dữ liệu Web

Nội dung

- Thiết kế CSDL phân tán
  - Phân mảnh
  - Phân tán dữ liệu
  - Các phương pháp kết hợp

Thiết kế phân tán



Nội dung

- Thiết kế CSDL phân tán
  - Phân mảnh
  - Phân tán dữ liệu
  - Các phương pháp kết hợp

Phân mảnh

- Có thể phân tán các quan hệ hay không?
  - Quan hệ không phải là một đơn vị phân tán thích hợp
    - Các khung nhìn (view) của ứng dụng thường là các tập con của các quan hệ
    - Nếu khung nhìn được định nghĩa dựa trên các quan hệ cho trước tại các trạm khác nhau thì:
      - Hoặc quan hệ không được nhân bản và được lưu trữ tại một trạm: cần phải thực hiện rất nhiều các thủ tục truy nhập từ xa.
      - Hoặc quan hệ được nhân bản tại tất cả hoặc một số trạm có chạy ứng dụng: khó khăn khi thực thi cập nhật, lãng phí không gian lưu trữ.

Phân mảnh

- Đơn vị phân tán hợp lý là gì?
  - Các mảnh của các quan hệ (quan hệ con)
    - Thực hiện đồng thời một số giao dịch mà có thể truy nhập đến các phần khác nhau của một quan hệ
    - Có thể thực thi song song truy vấn đơn (thao tác trên các mảnh) ➔ nội truy vấn đồng thời
    - Nhược điểm:
      - Nếu các khung nhìn không thể được xác định trên một mảnh đơn sẽ được yêu cầu xử lý thêm ➔ giảm khả năng thực thi
      - Kiểm soát dữ liệu ngữ nghĩa (đặc biệt là thực thi tính toàn vẹn) khó khăn hơn do có thể phải thực hiện tìm kiếm dữ liệu tại nhiều trạm

CSDL ví dụ

EMP

ENO	ENAME	TITLE
E1	J. Doe	Elect. Eng.
E2	M. Smith	Syst. Anal.
E3	A. Lee	Mech. Eng.
E4	J. Miller	Programmer
E5	B. Casey	Syst. Anal.
E6	L. Chu	Elect. Eng.
E7	R. Davis	Mech. Eng.
E8	J. Jones	Syst. Anal.

ASG

ENO	PNO	RESP	DUR
E1	P1	Manager	12
E2	P1	Analyst	24
E2	P2	Analyst	6
E3	P3	Consultant	10
E3	P4	Engineer	48
E4	P2	Programmer	18
E5	P2	Manager	24
E6	P4	Manager	48
E7	P3	Engineer	36
E8	P3	Manager	40

PROJ

PNO	PNAME	BUDGET	LOC
P1	Instrumentation	150000	Montreal
P2	Database Develop.	135000	New York
P3	CAD/CAM	250000	New York
P4	Maintenance	310000	Paris

PAY

TITLE	SAL
Elect. Eng.	40000
Syst. Anal.	34000
Mech. Eng.	27000
Programmer	24000

Phân mảnh theo chiều ngang

$PROJ_1$: các dự án với ngân sách thấp hơn $200,000

$PROJ_2$: các dự án với ngân sách lớn hơn hoặc bằng $200,000

PROJ

PNO	PNAME	BUDGET	LOC
P1	Instrumentation	150000	Montreal
P2	Database Develop.	135000	New York
P3	CAD/CAM	250000	New York
P4	Maintenance	310000	Paris

9

Phân mảnh theo chiều ngang

$PROJ_1$: các dự án với ngân sách thấp hơn $200,000

$PROJ_2$: các dự án với ngân sách lớn hơn hoặc bằng $200,000

PROJ

PNO	PNAME	BUDGET	LOC
P1	Instrumentation	150000	Montreal
P2	Database Develop.	135000	New York
P3	CAD/CAM	250000	New York
P4	Maintenance	310000	Paris

$PROJ_1$

PNO	PNAME	BUDGET	LOC
P1	Instrumentation	150000	Montreal
P2	Database Develop.	135000	New York

$PROJ_2$

PNO	PNAME	BUDGET	LOC
P3	CAD/CAM	255000	New York
P4	Maintenance	310000	Paris

10

Phân mảnh theo chiều dọc

$PROJ_1$: thông tin về ngân sách dự án

$PROJ_2$: thông tin về tên và vị trí của dự án

PROJ

PNO	PNAME	BUDGET	LOC
P1	Instrumentation	150000	Montreal
P2	Database Develop.	135000	New York
P3	CAD/CAM	250000	New York
P4	Maintenance	310000	Paris

11

Phân mảnh theo chiều dọc

$PROJ_1$: thông tin về ngân sách dự án

$PROJ_2$: thông tin về tên và vị trí của dự án

PROJ

PNO	PNAME	BUDGET	LOC
P1	Instrumentation	150000	Montreal
P2	Database Develop.	135000	New York
P3	CAD/CAM	250000	New York
P4	Maintenance	310000	Paris

$PROJ_1$

PNO	BUDGET
P1	150000
P2	135000
P3	250000
P4	310000

$PROJ_2$

PNO	PNAME	LOC
P1	Instrumentation	Montreal
P2	Database Develop.	New York
P3	CAD/CAM	New York
P4	Maintenance	Paris

12

Tính đúng đắn của việc phân mảnh

- Tính đầy đủ
  - Việc phân rã quan hệ R thành các mảnh R_1, R_2, ..., R_n được coi là hoàn thành khi và chỉ khi mỗi mục dữ liệu trong R đều có thể được tìm thấy trong một số R_i nào đó.
- Tính phục hồi
  - Nếu quan hệ R được phân rã thành các mảnh R_1, R_2, ..., R_n, thì cần tồn tại một toán tử quan hệ nào đó ∇ sao cho:

    $R = \nabla_{1 \le i \le n} R_i$
- Tính tách biệt
  - Nếu quan hệ R được phân rã thành các mảnh R_1, R_2, ..., R_n, và mục dữ liệu d_i thuộc R_j, thì d_i không thuộc về bất kỳ mảnh R_k ($k \neq j$) nào khác.

13

Các phương pháp cấp phát (/định vị)

- Không nhân bản
  - Khi phân chia: Mỗi mảnh chỉ có tại một trạm
- Nhân bản
  - Nhân bản đầy đủ: mỗi mảnh có tại một trạm
  - Nhân bản một phần: mỗi mảnh có tại một số trạm
- Quy tắc:
  - Nếu số lượng truy vấn chỉ đọc lớn hơn nhiều số lượng truy vấn cập nhật, thì nhân bản là tốt
  - Ngược lại, nhân bản sẽ gây ra rất nhiều vấn đề.

14

So sánh các phương pháp nhân bản

	Nhân bản hoàn toàn	Nhân bản một phần	Phân mảnh
XỬ LÝ TRUY VẤN	Dễ	Cùng mức độ khó khăn	
QUẢN LÝ THƯ MỤC	Dễ hoặc không tồn tại	Cùng mức độ khó khăn	
ĐIỀU KHIỂN ĐỒNG THỜI	Vừa phải	Khó	Dễ
ĐỘ TIN CẬY	Rất cao	Cao	Thấp
TÍNH THỰC TẾ	Có thể áp dụng	Thực tế	Có thể áp dụng

15