

## Chương II Các hệ cơ sở dữ liệu phân tán

Nguyễn Kim Anh  
[anhnk-fit@mail.hut.edu.vn](mailto:anhnk-fit@mail.hut.edu.vn)

Bộ môn Hệ thống Thông tin, SoICT

1

### Nội dung

- Tổng quan về các hệ CSDLPT
- Phân đoạn dữ liệu
- Biểu diễn các yêu cầu với các mức trong suốt khác nhau
- Thiết kế CSDLPT
- Tối ưu hóa CSDL phân tán
- Quản trị giao dịch và điều khiển tương tranh

2

### Mở đầu

- Các hệ CSDL quan hệ cổ điển, tập trung đã thu được những thành tựu to lớn về cả phương diện lý thuyết và thực hành.
- Hạn chế của CSDL quan hệ tập trung: vấn đề tổ chức, lưu trữ và khai thác dữ liệu không phù hợp với các tổ chức xí nghiệp có quy mô lớn, rất lớn và hoạt động trên các địa bàn rộng lớn ở nhiều vị trí địa lý cách xa nhau

*Hướng nghiên cứu mở rộng được đề xuất:*

*Xây dựng hệ CSDL phân tán*



3

### Tài liệu tham khảo

- J.D.Ullman, Principle of Knowledge-base and Database Systems, 1988.
- S.Ceri&G.Pelagatti, Distributed Databases, Principles&Systems, 1984.
- M.T. Özsu & P. Valduriez, Principles of Distributed Database Systems. Prentice-Hall, 1999.

4

## Tổng quan về các hệ CSDL

- CSDL phân tán?
- Hệ quản trị CSDL phân tán
- Phân loại các hệ CSDL phân tán
- Kiến trúc của CSDL phân tán
- Các mức trong suốt khác nhau
- Ưu/nhược điểm của việc xây dựng hệ CSDL phân tán

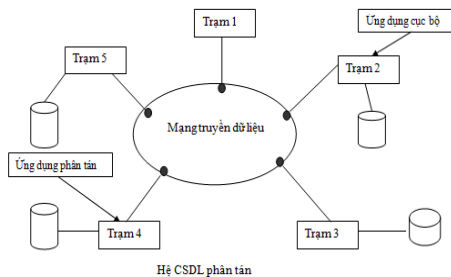
5

## CSDL phân tán?

- Một CSDL phân tán là một tập hợp nhiều CSDL có quan hệ logic với nhau và được phân tán trên một mạng máy tính
  - *Tính chất phân tán*: Toàn bộ dữ liệu của CSDL phân tán không được cư trú ở một nơi mà cư trú trên nhiều trạm thuộc mạng máy tính, điều này giúp chúng ta phân biệt CSDL phân tán với CSDL tập trung đơn lẻ.
  - *Quan hệ logic*: Toàn bộ dữ liệu của CSDL phân tán có một số các thuộc tính ràng buộc chúng với nhau, điều này giúp chúng ta có thể phân biệt một CSDL phân tán với một tập hợp CSDL cục bộ hoặc các tệp cư trú tại các vị trí khác nhau trong một mạng máy tính.

6

## CSDL phân tán?



7

## CSDL phân tán?

- -*Ứng dụng cục bộ*: yêu cầu và thực hiện tại một trạm trong hệ CSDL phân tán và chỉ liên quan đến CSDL tại nút đó.
- -*Ứng dụng phân tán (tổng thể)*: yêu cầu truy nhập dữ liệu tại nhiều trạm thông qua mạng truyền thông.

8

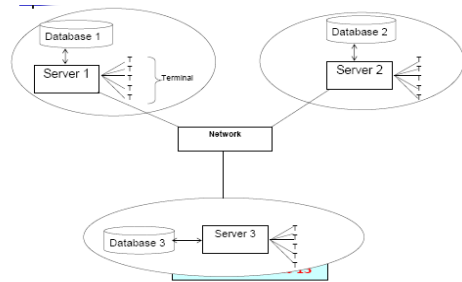
## Ví dụ:

Một ngân hàng có ba chi nhánh đặt tại các vị trí khác nhau:

- Tại mỗi chi nhánh có một máy tính điều khiển một số máy kế toán cuối cùng (teller, terminal) và cơ sở dữ liệu thống kê của chi nhánh đó.
- Mỗi máy tính với cơ sở dữ liệu thống kê địa phương của nó tại mỗi chi nhánh được đặt ở một vị trí của cơ sở dữ liệu phân tán. Các máy tính được nối với nhau bởi một mạng truyền thông. Các nút trong một mạng phân tán một mặt xử lý thông tin tại vị trí mà nó quản lý, mặt khác nó cũng tham gia vào việc xử lý các yêu cầu về thông tin cần truy cập qua nhiều địa điểm. Ví dụ như việc **lên danh sách tất cả nhân viên của ngân hàng**. Yêu cầu này đòi hỏi tất cả các máy tính ở các chi nhánh của công ty đều phải hoạt động để cung cấp thông tin.

9

## Ví dụ:



10

## Hệ quản trị CSDL phân tán

- Hệ quản trị CSDL phân tán (Distributed DataBase Management System-DDBMS) là một hệ thống phần mềm cho phép tạo lập CSDL phân tán và điều khiển mọi truy nhập đối với CSDL phân tán. Đặc biệt, DDBMS đảm bảo việc phân tán là trong suốt đối với người sử dụng.

11

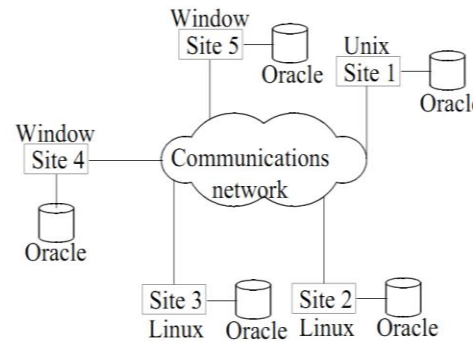
## Hệ quản trị CSDL phân tán

- Hệ quản trị CSDL phân tán gồm 1 tập các phần mềm (chương trình) sau đây:
  - Các chương trình quản trị các CSDL địa phương tại các trạm (DBMS địa phương)
  - Chương trình quản trị các dữ liệu phân tán
  - Các chương trình quản trị việc truyền thông dữ liệu
  - Các chương trình quản trị từ điển dữ liệu.

12

## Phân loại các hệ CSDLPT

- *Hệ CSDL phân tán thuần nhất:*
  - +Được thiết kế bằng cách chia nhỏ một CSDL hợp nhất thành một tập các CSDL cục bộ.
  - + Các CSDL cục bộ đều được mô hình hóa bởi cùng một mô hình dữ liệu
  - + Các CSDL cục bộ đều được quản trị bởi cùng một hệ quản trị CSDL địa phương.



13

## Phân loại các hệ CSDLPT

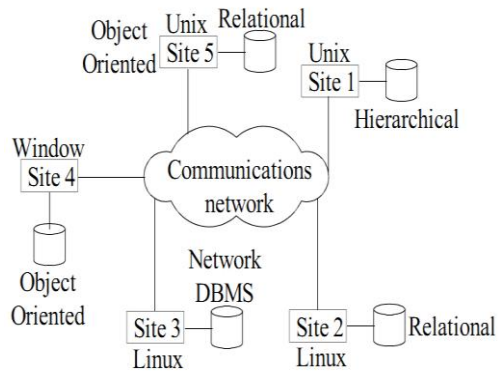
- *Hệ CSDL phân tán không thuần nhất:*
  - + Được thiết kế bằng cách tích hợp các CSDL cục bộ đã có, đang tồn tại, đang hoạt động.
  - + Các CSDL cục bộ có thể được mô hình hóa bởi các mô hình dữ liệu khác nhau
  - + Các CSDL cục bộ có thể được quản trị bởi các hệ quản trị CSDL địa phương khác nhau

15

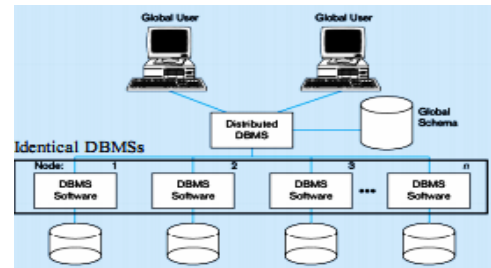
## Phân loại các hệ CSDLPT

- *Hệ CSDL liên hiệp:*
  - Một tập hợp các CSDL được ghép nối yếu (ghép nối lỏng) mà người dùng có thể truy cập và thao tác thông qua một ngôn ngữ đặc biệt (ngôn ngữ đa cơ sở) cho phép định nghĩa các DL đa cơ sở, các ràng buộc giữa các cơ sở và các thao tác DL đa cơ sở

16

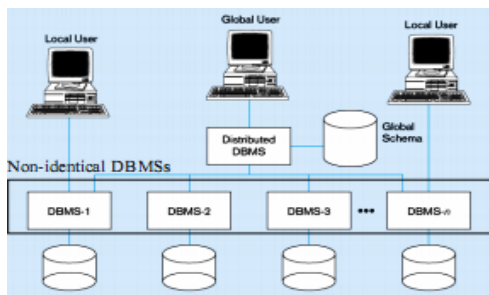


Hệ CSDL phân tán thuần nhất



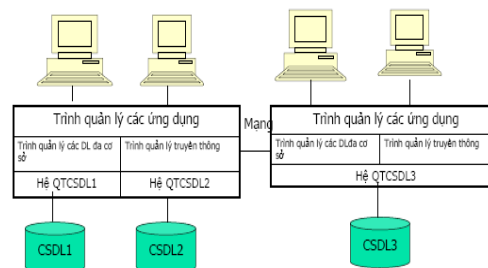
18

Hệ CSDL phân tán không thuần nhất



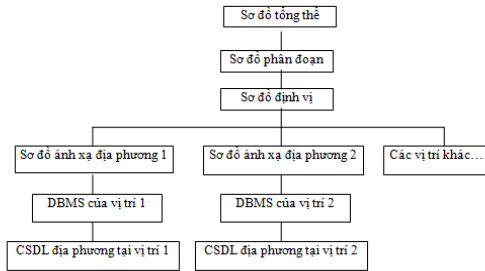
19

Hệ CSDL liên hiệp



20

## Kiến trúc của CSDL phân tán



21

## Kiến trúc của CSDL phân tán

- **Sơ đồ tổng thể:** Định nghĩa tất cả các dữ liệu sẽ được lưu trữ trong CSDL phân tán. Sử dụng mô hình quan hệ, sơ đồ tổng thể bao gồm một tập các định nghĩa quan hệ tổng thể.
- **Sơ đồ phân đoạn:** Mỗi quan hệ tổng thể có thể được chia thành một hay nhiều phần không bao trùm lên nhau được gọi là các đoạn (fragments). Có nhiều cách khác nhau để thực hiện việc phân chia này.
  - Ánh xạ (một - nhiều):  
 $\{\text{quan hệ tổng thể}\} \rightarrow \{\text{đoạn}\}$   
 được định nghĩa trong sơ đồ phân đoạn.
  - Ký hiệu  $R_i$  là đoạn thứ  $i$  của quan hệ tổng thể  $R$

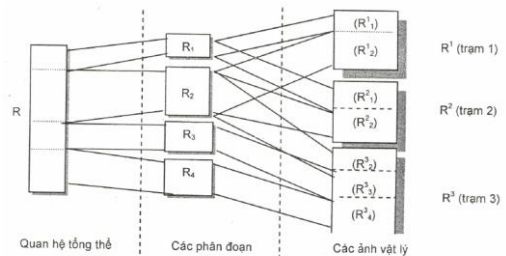
22

## Kiến trúc của CSDL phân tán

- **Sơ đồ định vị:** Các đoạn là các bộ phận logic của một quan hệ tổng thể được định vị vật lý tại một hoặc nhiều trạm trên mạng máy tính. Sơ đồ định vị xác định mỗi đoạn được định vị tại các trạm nào. Lưu ý rằng kiểu ánh xạ được định nghĩa trong sơ đồ định vị quyết định CSDL phân tán là dư thừa hay không.
  - Ký hiệu  $R_j^i$  là bản sao của đoạn thứ  $i$  của quan hệ tổng thể  $R$  tại trạm  $j$  và  $R_j$  là tập các bản sao của các đoạn của quan hệ tổng thể  $R$  được định vị tại trạm  $j$
  - $R_j$  được gọi là ảnh vật lý của quan hệ tổng thể  $R$  tại trạm  $j$
- Ánh xạ (một - một):  
 $\{\text{quan hệ tổng thể, trạm}\} \rightarrow \{\text{ảnh vật lý}\}$   
 được định nghĩa trong sơ đồ định vị.

23

## Kiến trúc của CSDL phân tán



24

## Kiến trúc của CSDL phân tán

- - Sơ đồ ánh xạ địa phương tại trạm j:

Ánh xạ:

{Ảnh vật lý tại trạm j} → {CSDL tại trạm j}

25

## Các mức trong suốt khác nhau

- Tính trong suốt của một hệ CSDL phân tán được hiểu là việc che giấu sự phân tán của dữ liệu đối với người sử dụng và những người lập trình ứng dụng.
  - **Mức trong suốt phân đoạn:** : Người sử dụng biểu diễn các yêu cầu trên các quan hệ tổng thể mà không cần biết về việc phân đoạn của các quan hệ tổng thể → việc phân đoạn là được che giấu đối với người sử dụng
  - **Mức trong suốt định vị:** Người sử dụng biểu diễn các yêu cầu trên các đoạn mà không cần biết về việc phân đoạn của các quan hệ tổng thể cũng như việc định vị các đoạn tại các trạm → việc định vị các đoạn là được che giấu đối với người sử dụng
    - → **Tính trong suốt lập lại:** việc lập lại các bản sao đoạn là độc lập đối với ứng dụng của người sử dụng
  - **Mức trong suốt ánh xạ địa phương:** Người sử dụng biểu diễn các yêu cầu trên các bản sao của các đoạn mà không cần biết về việc phân đoạn của các quan hệ tổng thể, việc định vị các đoạn tại các trạm cũng như ánh xạ địa phương tại các trạm → việc ánh xạ địa phương tại các trạm là được che giấu đối với người sử dụng

26

## Ưu điểm của hệ CSDL phân tán

- -Phù hợp với cơ cấu và qui mô của các tổ chức lớn phi tập trung
- -Nâng cao khả năng chia sẻ và tính tự trị địa phương.
- -Giảm chi phí truyền thông
- -Nâng cao tính sẵn sàng.
- -Nâng cao độ tin cậy.
- -Nâng cao hiệu năng.
- -Dễ mở rộng.

27

## Nhược điểm của hệ CSDL phân tán

- -Thiết kế CSDL phức tạp hơn.
- -Khó điều khiển tính nhất quán dữ liệu.
- -Khó phát hiện và khử lỗi.
- -Giá thành cao.
- -Thiếu chuẩn mực.
- -Khó đảm bảo an toàn và bảo mật dữ liệu.

28

## Nội dung

- Tổng quan về các hệ CSDLPT
- **Phân đoạn dữ liệu**
- Biểu diễn các yêu cầu với các mức trong suốt khác nhau
- Thiết kế CSDLPT
- Tối ưu hóa CSDL phân tán
- Quản trị giao dịch và điều khiển tranh

29

## Phân đoạn dữ liệu

### 1 - Phân đoạn quan hệ là gì?

Việc chia/tách một quan hệ tổng thể thành nhiều quan hệ nhỏ hơn hay nhiều đoạn được gọi là phân đoạn quan hệ.

30

## Phân đoạn dữ liệu

### 2 – Các lý do phải phân đoạn dữ liệu:

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

31

## Phân đoạn dữ liệu

### 3 – Các kiểu phân đoạn:

- Phân đoạn ngang (horizontal fragmentation)
    - **phân đoạn ngang đơn giản** (primary horizontal fragmentation)
    - **phân đoạn ngang suy diễn** (derived horizontal fragmentation)
  - Phân đoạn dọc (vertical fragmentation).
  - Phân đoạn hỗn hợp (híbrid fragmentation)
- *Chú ý: Việc phân đoạn dữ liệu phải được gắn liền với việc định vị dữ liệu*

32



## Phân đoạn dữ liệu

### 4- Các điều kiện đối với một phân đoạn

- Việc phân đoạn một quan hệ tổng thể phải tuân theo một số điều kiện nhất định để khi tái thiết lại quan hệ cũ vẫn bảo đảm ngữ nghĩa của nó.
- Một phân đoạn đúng đắn phải thỏa mãn ba điều kiện sau:
  - *Tính đầy đủ (completeness)*
  - *Tính tái thiết được (reconstruction)*
  - *Tính tách biệt (disjointness):*

33

## Phân đoạn dữ liệu

### \* *Tính đầy đủ:*

Nếu một quan hệ  $R$  được phân rã thành các đoạn

$R_1, R_2, \dots, R_k$  thì mỗi mục dữ liệu xuất hiện trong  $R$  phải xuất hiện trong ít nhất một đoạn  $R_i$  nào đó.

34

## Phân đoạn dữ liệu

### \* *Tính tái thiết được:*

- Nếu một quan hệ  $R$  được phân rã thành các đoạn  $R_1, R_2, \dots, R_k$  thì phải tồn tại một toán tử  $\theta$  sao cho  $R = \theta(R_1, R_2, \dots, R_k)$ .
- Toán tử  $\theta$  phụ thuộc vào từng kiểu phân đoạn.
- Trong thực tế, khi các đoạn được phân đoạn ngang thì  $\theta$  là phép hợp, phân đoạn dọc thì  $\theta$  là phép nối và phân đoạn hỗn hợp thì  $\theta$  là một biểu thức đại số bao hàm cả phép hợp và phép nối.

35

## Phân đoạn dữ liệu

### \* *Tính tách biệt:*

- Nếu một quan hệ  $R$  được phân rã thành các quan hệ  $R_1, R_2, \dots, R_k$  và mục dữ liệu  $t_i$  nằm trong đoạn  $R_i$  thì nó sẽ không nằm trong một đoạn  $R_k, k \neq i$ .
- Tiêu chuẩn này bảo đảm sự bao trùm giữa các đoạn là tối thiểu hay các đoạn phải được tách rời nhau.

36

## Phân đoạn ngang đơn giản

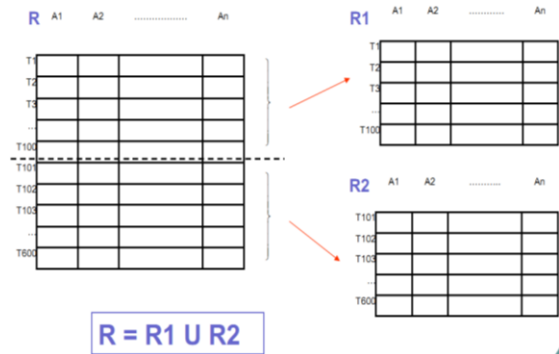
- Một đoạn ngang được xác định bằng phép chọn trên quan hệ tổng thể.

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

- $F_i$  là điều kiện chọn/tiêu chuẩn của đoạn  $R_i$
- Phân đoạn ngang đơn giản là đúng đắn nếu thỏa mãn ba điều kiện sau:
  - Tính đầy đủ*:  $\forall b \in R, \exists 1 \leq i \leq n, b$  thỏa  $F_i$
  - Tính tái thiết được*:  $R = R_1 \cup R_2 \cup \dots \cup R_n$
  - Tính tách biệt*:  $F_i$  và  $F_j$  là loại trừ nhau với  $i \neq j$

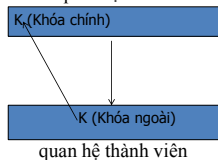
37

## Phân đoạn ngang đơn giản



## Phân đoạn ngang suy diễn

- Là phân đoạn ngang một quan hệ tổng thể (quan hệ thành viên) suy diễn ra từ sự phân đoạn ngang của một quan hệ tổng thể khác (quan hệ chủ)



39

## Phân đoạn ngang suy diễn

- Một đoạn ngang được xác định bằng phép nối kết nối của quan hệ thành viên (R) và đoạn tương ứng của quan hệ chủ (S).

$$R_i = R \bowtie S_i; 1 \leq i \leq n$$

- điều kiện nối  $F_i$  có dạng:  $K = K_i$ , trong đó  $K$  là khóa chính của  $S$  và  $K_i$  là khóa ngoài trong  $R$
- Phân đoạn ngang suy diễn là đúng đắn nếu thỏa mãn ba điều kiện sau:
  - Tính đầy đủ*:  $R$  chứa khóa ngoài tham chiếu đến khóa chính của  $S$  (ràng buộc toàn vẹn tham chiếu)
  - Tính tái thiết được*:  $R = R_1 \cup R_2 \cup \dots \cup R_n$
  - Tính tách biệt*:  $K$  là khóa chính của  $S$

40

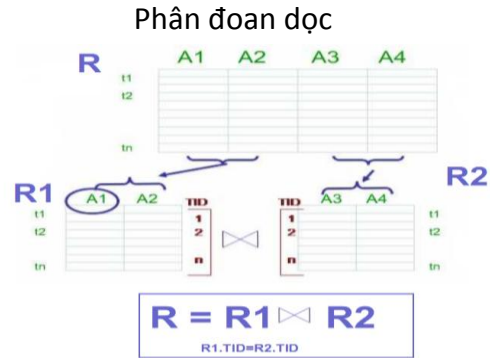
## Phân đoạn dọc

- Một đoạn dọc được xác định bằng phép chiếu trên quan hệ tổng thể.

$$R_i = \Pi_{U_i}(R); 1 \leq i \leq n$$

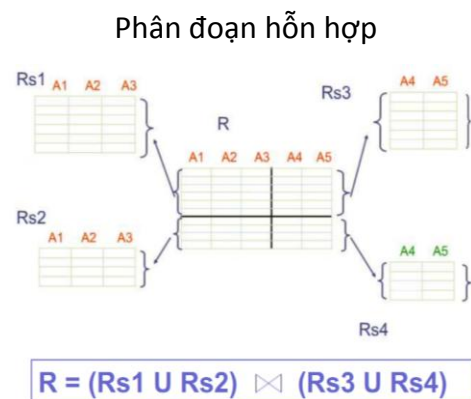
- $U_i$  là tập thuộc tính của đoạn  $R_i$
- Phân đoạn dọc là đúng đắn nếu thỏa mãn ba điều kiện sau:
  - Tính đầy đủ:  $U = U_1 \cup U_2 \cup \dots \cup U_n$
  - Tính tái thiết được:  $R = R_1 \bowtie R_2 \bowtie \dots \bowtie R_n$
  - Tính tách biệt:  $U_i \cap U_j$  là tối thiểu với  $i \neq j$

41



## Phân đoạn hỗn hợp

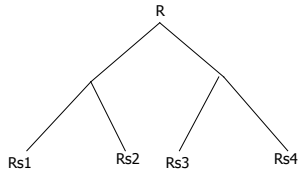
- Là phân đoạn được thực hiện bằng cách áp dụng một cách đệ quy/đồng thời cả phân đoạn ngang và phân đoạn dọc



43

## Phân đoạn hỗn hợp

- Cây phân đoạn



45

## Ví dụ:

- Xét cơ sở dữ liệu của một công ty máy tính 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 phát triển.
  - HOSO (MANV, MADA, NHIEMVU, THOIGIAN)**: quan hệ này chứa dữ liệu về hồ sơ của nhân viên được phân công thực hiện dự án.

46

## Ví dụ:

**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

47

**HOSO (G)**

MA NV	MA DA	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

## Ví dụ:

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

**DUAN (J)**

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

**TIENLUONG (S)**

48

## Ví dụ về phân đoạn dữ liệu:

- Ví dụ về phân đoạn ngang đơn giản quan hệ **TLUONG** (**CHUCVU**, **LUONG**):

$$TLUONG_1 = \sigma_{LUONG \leq 2500}(TLUONG)$$

$$TLUONG_2 = \sigma_{LUONG > 2500}(TLUONG)$$

- Ví dụ về phân đoạn ngang suy diễn quan hệ **NHANVIEN** (**MANV**, **TENNV**, **CHUCVU**) từ sự phân đoạn ngang của quan hệ **TLUONG**:

$$NHANVIEN_i = NHANVIEN \mid > TLUONG_i; 1 \leq i \leq 2$$

49

## Ví dụ về phân đoạn dữ liệu:

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

$$DUAN1 = \sigma_{NGANSACH \leq 20000}(DUAN)$$

$$DUAN2 = \sigma_{NGANSACH > 20000}(DUAN)$$

DUAN 1

MADA	TENDA	NGANSACH
D1	CSDL	20000
D2	CÁI ĐẶT	12000

DUAN 2

MADA	TENDA	NGANSACH
D3	BẢO TRÌ	28000
D4	PHÁT TRIỂN	25000

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

$$DUAN1 \subseteq DUAN; \quad DUAN2 \subseteq DUAN; \quad DUAN = DUAN1 \cup DUAN2$$

## Ví dụ về phân đoạn dữ liệu:

- Ví dụ về phân đoạn ngang suy diễn:
  - DUAN(MADA, TENDA, NGANSACH)**: quan hệ chủ chứa khóa chính **MADA**
  - HOSO(MANV, MADA, NHIEMVU, THOIGIAN)**: quan hệ thành viên chứa khóa ngoài **MADA** tham chiếu đến **DUAN**

$$\Rightarrow HOSO_i = HOSO \mid > DUAN_i; 1 \leq i \leq 2$$

51

## Ví dụ về phân đoạn dữ liệu:

- Ví dụ về phân đoạn ngang suy diễn:
  - NHANVIEN (MANV, TENNV, CHUCVU)**: quan hệ chủ chứa khóa chính **MANV**
  - HOSO(MANV, MADA, NHIEMVU, THOIGIAN)**: quan hệ thành viên chứa khóa ngoài **MANV** tham chiếu đến **NHANVIEN**

$$\Rightarrow HOSO_i = HOSO \mid > NHANVIEN_i; 1 \leq i \leq 2$$

52

### Ví dụ về phân đoạn dữ liệu:

- Ví dụ về phân đoạn dọc quan hệ  $HOSO(MANV, MADA, NHIEMVU, THOIGIAN)$ :

$$HOSO_1 = \Pi_{MANV, NHIEMVU, TID}(HOSO)$$

$$HOSO_2 = \Pi_{MADA, THOIGIAN, TID}(HOSO)$$

53

### Ví dụ về phân đoạn dữ liệu:

- Ví dụ về phân đoạn hỗn hợp quan hệ  $HOSO(MANV, MADA, NHIEMVU, THOIGIAN)$ :

$$HOSO_1 = \Pi_{MANV, NHIEMVU, TID}(HOSO) \mid >< NHANVIEN_1$$

$$HOSO_2 = \Pi_{MANV, NHIEMVU, TID}(HOSO) \mid >< NHANVIEN_2$$

$$HOSO_3 = \Pi_{MADA, THOIGIAN, TID}(HOSO) \mid >< DUAN_1$$

$$HOSO_4 = \Pi_{MADA, THOIGIAN, TID}(HOSO) \mid >< DUAN_2$$

54

### Ví dụ về định vị dữ liệu:

$$TLUONG_1 = \sigma_{LUONG \leq 2500}(TLUONG) \text{ -site } 1,3$$

$$TLUONG_2 = \sigma_{LUONG > 2500}(TLUONG) \text{ -site } 2,4,5$$

$$NHANVIEN_1 = NHANVIEN \mid >< TLUONG_1 \text{ -site } 1,3$$

$$NHANVIEN_2 = NHANVIEN \mid >< TLUONG_2 \text{ -site } 2,4,5$$

$$DUAN_1 = \sigma_{NGANSACH \leq 20000}(DUAN) \text{ -site } 1,2,5$$

$$DUAN_2 = \sigma_{NGANSACH > 20000}(DUAN) \text{ -site } 3,4$$

$$HOSO_1 = \Pi_{MANV, NHIEMVU, TID}(HOSO) \mid >< NHANVIEN_1 \text{ -site } 1,3$$

$$HOSO_2 = \Pi_{MANV, NHIEMVU, TID}(HOSO) \mid >< NHANVIEN_2 \text{ -site } 2,4,5$$

$$HOSO_3 = \Pi_{MADA, THOIGIAN, TID}(HOSO) \mid >< DUAN_1 \text{ -site } 1,2,5$$

$$HOSO_4 = \Pi_{MADA, THOIGIAN, TID}(HOSO) \mid >< DUAN_2 \text{ -site } 3,4$$

55

### Nội dung

- Tổng quan về các hệ CSDLPT
- Phân đoạn dữ liệu
- Biểu diễn các yêu cầu với các mức trong suốt khác nhau**
- Thiết kế CSDLPT
- Tối ưu hóa CSDL phân tán
- Quản trị giao dịch và điều khiển tương tranh

56

## Biểu diễn các yêu cầu với các mức trong suốt khác nhau

- Ngôn ngữ làm việc:
  - Các lệnh thao tác dữ liệu: SQL
  - Các lệnh lập trình: sử dụng ngôn ngữ lập trình tựa Pascal:
    - Read(*terminal*, *var*): đọc dữ liệu từ terminal
    - Write(*terminal*, *var*): ghi KQ tại terminal
    - Send DL from site <i> to site <j>: Truyền dữ liệu từ trạm i đến trạm j
    - Biến trao đổi giữa SQL và Pascal: \$<tên-biến>
    - Biến truyền thông tin KQ của lệnh truy vấn SQL: biến logic FOUND

57

## Biểu diễn các yêu cầu với các mức trong suốt khác nhau

- **Ứng dụng ‘chỉ đọc’**: các ứng dụng chỉ tra cứu dữ liệu mà không làm thay đổi dữ liệu trong CSDL  
→ nên lựa chọn bản sao đoạn tốt nhất để giảm chi phí truyền thông
- **Ứng dụng ‘cập nhật’**: các ứng dụng làm thay đổi dữ liệu trong CSDL (thêm, xóa, sửa) → phải cập nhật tất cả các bản sao đoạn để đảm bảo tính nhất quán dữ liệu

58

## Ứng dụng ‘chỉ đọc’

- Ví dụ ‘chỉ đọc’:  
MADA được đọc vào từ terminal, hãy tra TENDA và hiển thị tại terminal tại trạm 2
- Ứng dụng này thực hiện trên quan hệ DUAN(MADA, TENDA, NGANSACH)  
với thông tin phân đoạn và sắp chỗ:  

$$DUAN_1 = \sigma_{NGANSACH \leq 20000} (DUAN) \text{ -site } 1,2,5$$

$$DUAN_2 = \sigma_{NGANSACH > 20000} (DUAN) \text{ -site } 3,4$$

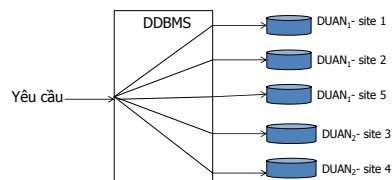
59

## Ứng dụng ‘chỉ đọc’

- **Mức trong suốt phân đoạn**:

```
read (terminal, $MaDA );
select TENDA into $TenDA
from DUAN
where MADA= $MaDA;
write (terminal, $TenDA );
```

} Yêu cầu



60

## Ứng dụng ‘chỉ đọc’

- **Mức trong suốt định vị:**

```

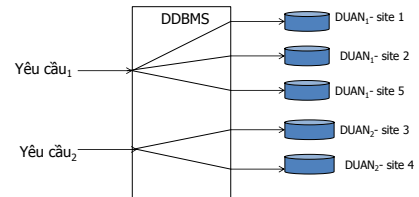
read (terminal, $MaDA );
select TENDA into $TenDA
from DUAN1
where MADA= $MaDA;
if not FOUND then
select TENDA into $TenDA
from DUAN2
where MADA= $MaDA;
write (terminal, $TenDA );
  
```

Yêu cầu<sub>1</sub>

Yêu cầu<sub>2</sub>

61

## Ứng dụng ‘chỉ đọc’



62

## Ứng dụng ‘chỉ đọc’

- **Mức trong suốt ánh xạ địa phương:**

```

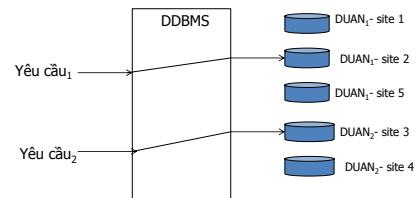
read (terminal, $MaDA );
select TENDA into $TenDA
from DUAN1 at site 2
where MADA= $MaDA;
if not FOUND then
select TENDA into $TenDA
from DUAN2 at site 3
where MADA= $MaDA;
send $TenDA from site 3 to site 2;
write (terminal, $TenDA );
  
```

Yêu cầu<sub>1</sub>

Yêu cầu<sub>2</sub>

63

## Ứng dụng ‘chỉ đọc’



64



Ứng dụng ‘cập nhật’

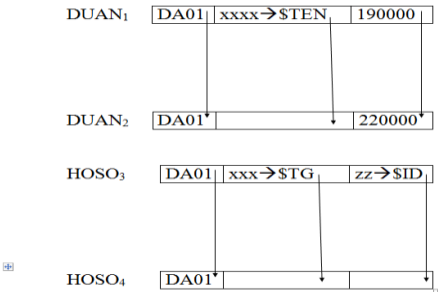
- Ví dụ ‘cập nhật’:  
Dự án có MADA là DA01 được tăng NGANSACH từ 19000 lên 22000. Hãy thực hiện tất cả các cập nhật cần thiết.
- Ứng dụng này thực hiện cập nhật trên quan hệ DUAN(MADA, TENDA, NGANSACH)  
với thông tin phân đoạn và sắp chỗ:  
 $DUAN_1 = \sigma_{NGANSACH \leq 20000}(DUAN)$  -site 1,2,5  
 $DUAN_2 = \sigma_{NGANSACH > 20000}(DUAN)$  -site 3,4  
Việc cập nhật quan hệ DUAN ảnh hưởng đến 2 đoạn của quan hệ HOSO được phân đoạn ngang suy diễn từ sự phân đoạn ngành đối với DUAN :  
 $HOSO_3 = \pi_{MADA, THOIGIAN, TID}(HOSO) \mid >> DUAN_1$  -site 1,2,5  
 $HOSO_4 = \pi_{MADA, THOIGIAN, TID}(HOSO) \mid >> DUAN_2$  -site 3,4

66

Ứng dụng ‘cập nhật’

- Mức trong suốt phân đoạn:  
update DUAN  
set NGANSACH = 22000  
where MADA = ‘DA01’

Ứng dụng ‘cập nhật’



67

Ứng dụng ‘cập nhật’

- Mức trong suốt định vị:  
select TENDA into \$TEN  
from DUAN<sub>1</sub>  
where MADA = ‘DA01’;  
select THOIGIAN, TID into \$TG, \$ID  
From HOSO<sub>3</sub>  
where MADA = ‘DA01’;  
delete HOSO<sub>3</sub> where MADA = ‘DA01’;  
delete DUAN<sub>1</sub> where MADA = ‘DA01’;  
insert into DUAN<sub>2</sub>: (‘DA01’, \$TEN, 22000)  
insert into HOSO<sub>4</sub>: (‘DA01’, \$TG, \$ID);

68

## Ứng dụng ‘cập nhật’

- Mức trong suốt ảnh xạ địa phương:

```

select TENDA into $TEN from DUANi at site 1
where MADA='DA01';
select THOIGIAN,TID into $TG, $ID from HOSO3 at site 1
where MADA = 'DA01';
delete HOSO3 at site 1 where MADA = 'DA01';
delete HOSO3 at site 2 where MADA = 'DA01';
delete HOSO3 at site 5 where MADA = 'DA01';
delete DUANi at site 1 where MADA = 'DA01';
delete DUANi at site 2 where MADA = 'DA01';
delete DUANi at site 5 where MADA = 'DA01';
insert into DUAN2 at site 3: ('DA01', $TEN, 22000);
insert into DUAN2 at site 4: ('DA01', $TEN, 22000);
insert into HOSO4 at site 3: ('DA01', $TG, $ID);
insert into HOSO4 at site 4: ('DA01', $TG, $ID);

```

69



70