



# Cơ sở dữ liệu phân tán

**Phạm Thị Anh Lê – ĐH Sư phạm Hà  
nội**

# Giới thiệu



---

- **Khái niệm và kiến trúc của hệ CSDLPT**
- **Mục tiêu của các hệ QTCSDL phân tán**
- **Kiến trúc tham chiếu**
- **Phân đoạn dữ liệu**
- **Đánh giá các câu hỏi phân tán**
- **Quản lý các giao tác phân tán**



## Khái niệm hệ CSDLPT

---

CSDLPT là một bộ các sưu tập các DL về mặt logic thuộc cùng một hệ thống nhưng về mặt vật lý được phân bố trên các trạm của một mạng máy tính.

Hệ QTCSDLPT là hệ thống phần mềm cho phép quản lý CSDLPT, làm cho việc phân tán trở nên “**trong suốt**” đối với người dùng

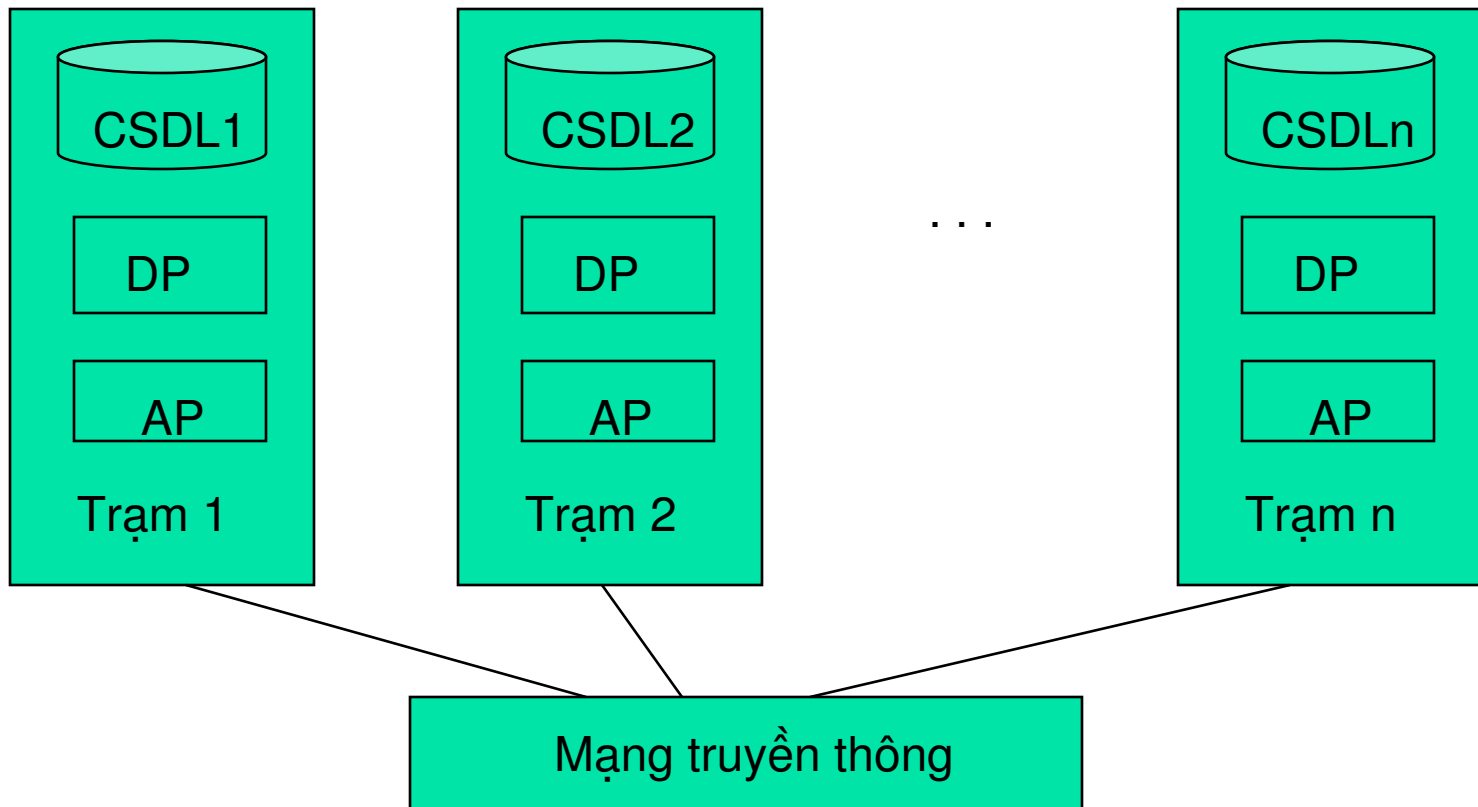


# Kiến trúc của hệ CSDLPT

---

- Có nhiều máy tính được gọi là các trạm (nút – node)
- Các trạm phải được kết nối bởi một kiểu mạng truyền thông để truyền DL và các lệnh giữa các trạm
- Phần mềm quản lý hệ CSDLPT:
  - Xử lý DL (DP – Data Processor): quản lý DL cục bộ (địa phương) tại một trạm
  - Xử lý ứng dụng (AP – Application Processor): thực hiện chức năng phân tán, truy cập thông tin phân tán từ thư mục CSDL phân tán và xử lý các yêu cầu truy cập đến nhiều trạm
  - Phần mềm truyền thông: cung cấp các nguyên thủy truyền thông được AP dùng để truyền lệnh và các trạm khi cần

# Kiến trúc của hệ CSDLPT



Kiến trúc đơn giản hoá của hệ thống vật lý cho hệ CSDLPT

Phạm Thị Anh Lê - ĐHSPTHN



# Kiến trúc của hệ CSDLPT

---

Một số chức năng đặc biệt của hệ QTCSDLPT:

- Quản lý một từ điển DL tổng thể lưu giữ thông tin liên quan đến các DLPT
- Định nghĩa các DLPT
- Kiểm tra ngữ nghĩa của các DLPT
- Định giá các câu hỏi phân tán của người dùng
- Quản lý các giao tác phân tán, các chương trình nguyên tố trong đó có các câu hỏi phân tán.



# Các cách tiếp cận cho việc phân bố dữ liệu

---

## 1. CSDL từ xa (remote database)

CSDL ở trên một máy tính khác với máy tính của người sử dụng và được truy nhập nhờ các lệnh truyền thông được xác định bởi người dùng

Cách tiếp cận CSDL từ xa cho một chức năng hạn chế nhưng không gặp phải những vấn đề khó nhất của CSDLPT



# Các cách tiếp cận cho việc phân bố dữ liệu

---

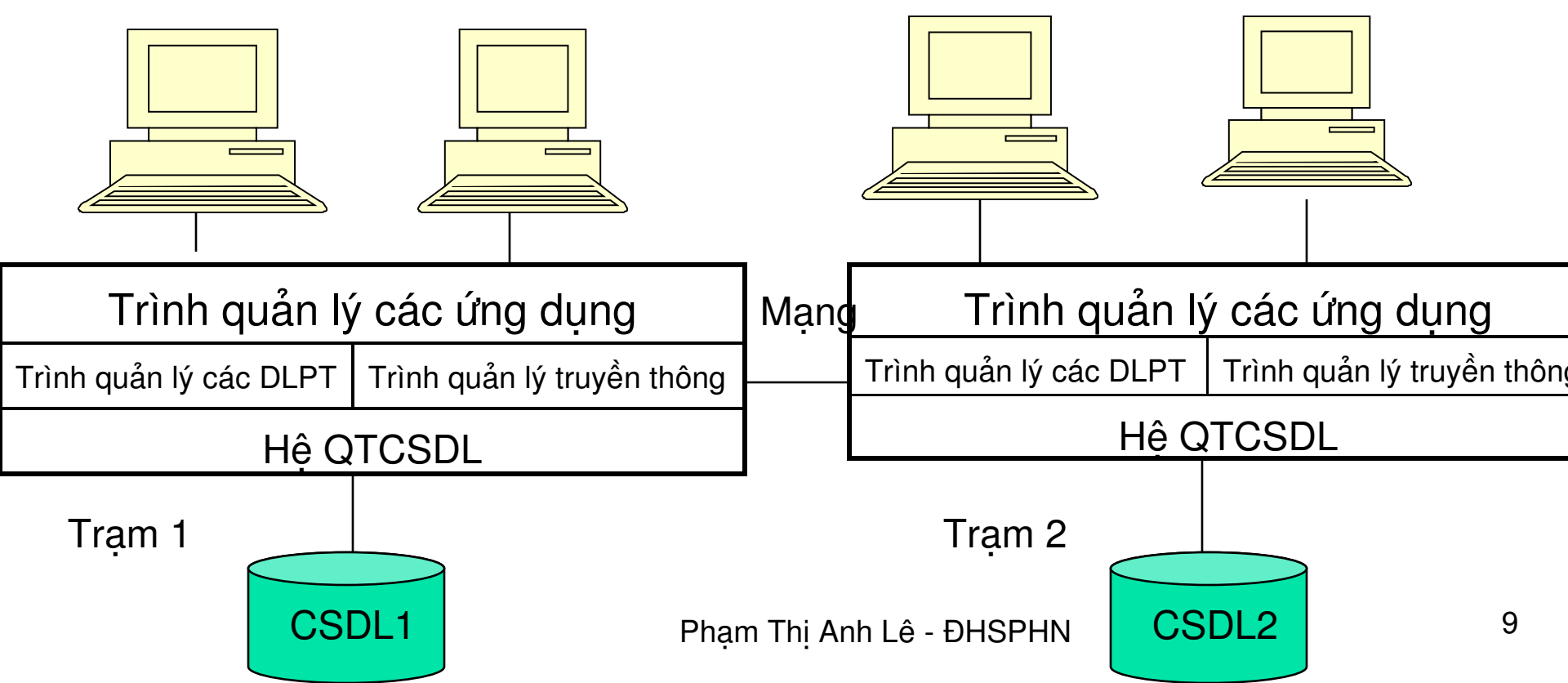
**2. CSDLPT:** tập các CSDL hợp tác, mỗi CSDL cư trú trên một trạm khác nhau, được nhìn thấy và thao tác bởi người dùng như chỉ là một CSDL tập trung duy nhất.

Như vậy, sự phân bố của dữ liệu là **trong suốt** đối với người dùng



# Các cách tiếp cận cho việc phân bổ dữ liệu

Việc quản lý các dữ liệu phân tán đòi hỏi ở mỗi trạm (site) lắp đặt các thành phần hệ thống sau:





# Các cách tiếp cận cho việc phân bố dữ liệu

---

Hệ thống quản lý một CSDL phân tán là một hệ QTCSDL phân tán

Tập hợp các phần mềm hệ thống bao gồm các trình quản lý các dữ liệu phân tán, các trình quản lý truyền thông và các CSDL địa phương cư trú trên mỗi trạm của CSDLPT



# Các cách tiếp cận cho việc phân bổ dữ liệu

---

- **CSDLPT thuần nhất:** CSDLPT có được bằng cách chia một CSDL thành một tập các CSDL địa phương (cục bộ), mỗi CSDL này được quản lý bởi cùng hệ QTCSDL

*(thuận lợi cho cách tiếp cận thiết kế trên-xuống)*

- **CSDLPT không thuần nhất:** CSDLPT có được bằng cách tích hợp vào một CSDL duy nhất một tập các CSDL địa phương được quản lý bởi các hệ QTCSDL khác nhau

*(thuận lợi cho cách tiếp cận thiết kế dưới-lên)*

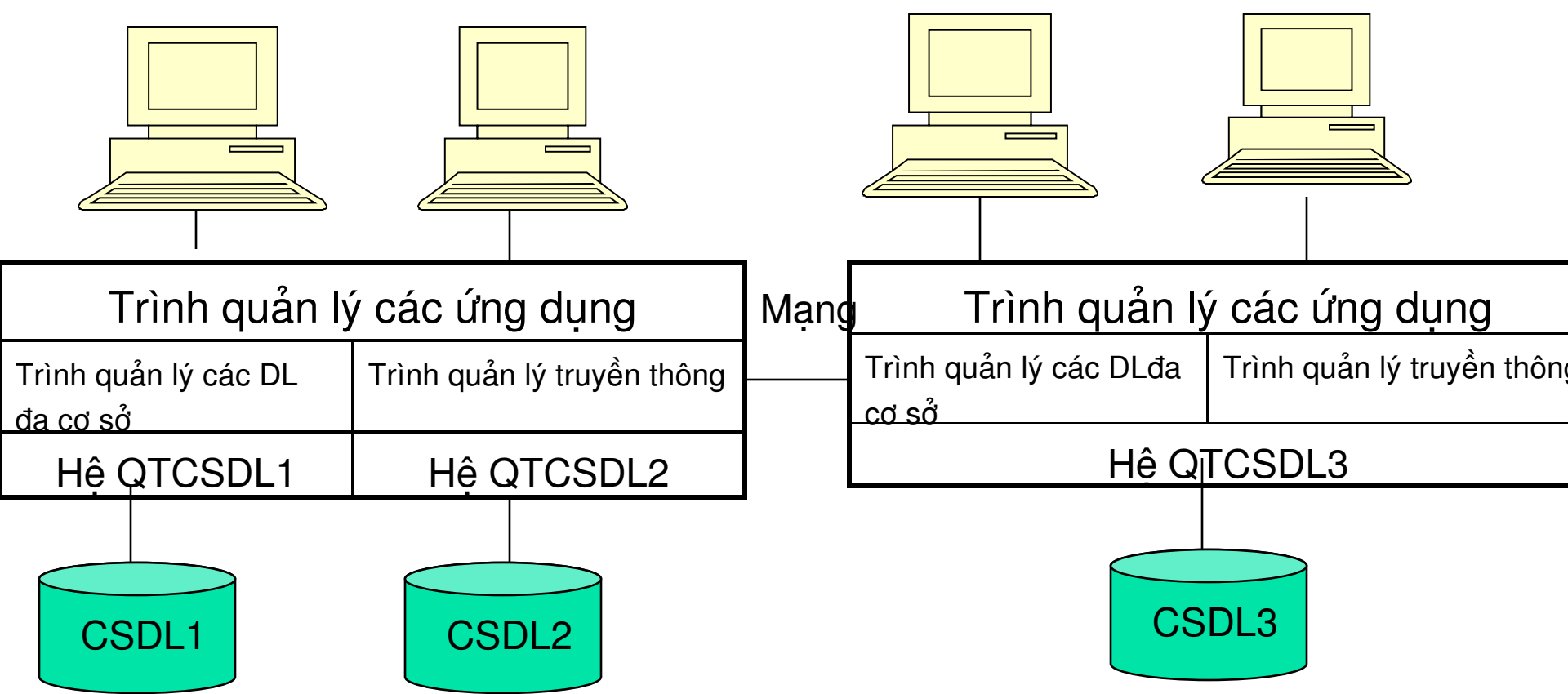


# Các cách tiếp cận cho việc phân bố dữ liệu

---

**3. CSDL liên hiệp (đa CSDL):** một tập hợp các CSDL được ghép nối yếu (lỏng lẻo) mà người dùng có thể thao tác nhờ vào một ngôn ngữ đặc biệt (ngôn ngữ đa cơ sở) cho phép định nghĩa các DL đa cơ sở, các phụ thuộc giữa các cơ sở và các thao tác DL đa cơ sở

# Các cách tiếp cận cho việc phân bố dữ liệu



# Các cách tiếp cận cho việc phân bố dữ liệu

---

## 4. CSDL song song

Các DL được phân bố trên tập các nút của một máy tính song song (bộ đa xử lý) để tăng hiệu quả việc quản lý DL.

Một CSDL song song là một CSDLPT thuần nhất mà các trạm là các nút của một máy tính song song và truyền đạt bằng thông báo.



# Mục tiêu của các hệ QTCSDLPT

---

- Tính độc lập đối với sự phân bố DL
  - Người dùng CSDLPT có thể không quan tâm tới sự phân tán của DL.
  - Thông tin về sự phân bố của DL được giữ trong từ điển DL và được hệ QTCSDLPT tham khảo để xác định vị trí các quan hệ có liên quan trong các câu hỏi của người dùng

*Sự trong suốt đối với sự định vị trí của DL cung cấp tính độc lập vật lý đối với môi trường phân tán.*



# Mục tiêu của các hệ QTCSDLPT

---

- Tính độc lập đối với sự phân đoạn
  - Việc truy nhập tới DL thường được xác định trên các quan hệ con (thu được từ việc chia nhỏ các quan hệ nguyên vẹn) được gọi là các **đoạn**
  - Các đoạn có thể được lưu trữ ở các trạm khác nhau
  - Việc phân đoạn (ngang, dọc, hỗn hợp) làm tăng tính hiệu quả của một CSDLPT vì nó cho phép làm dễ dàng các truy nhập địa phương.

*Tính độc lập đối với các phân đoạn đầu người dùng việc các DL đã được phân đoạn.*





# Mục tiêu của các hệ QTCSDLPT

---

- Tính độc lập đối với việc nhân (sao) bản
  - Một đoạn được nhân bản khi nó có tồn tại hai hay nhiều hơn các bản sao, mỗi bản sao đượ lưu giữ trên một trạm khác nhau
  - Sự sao bản được điều khiển bởi hệ thống làm DL có tính sẵn dùng cao, cải tiến hiệu năng truy nhập, tăng khả năng xử lý địa phương.
  - Tuy nhiên, việc duy trì các bản sao luôn phải giống nhau là phức tạp và tốn kém.

*Tính độc lập đối với nhân bản DL khiến người dùng không thấy là có nhân bản mà chỉ thấy các quan hệ không có nhân bản.*



# Mục tiêu của các hệ QTCSDLPT

---

- Tính độc lập đối với các hệ QTCSDL
  - Cho phép dấu sự kiện là các hệ QTCSDL địa phương có thể khác nhau
  - Mục tiêu này khó thực hiện hoàn toàn đối với hệ CSDLPT không thuần nhất



# Mục tiêu của các hệ QTCSDLPT

---

- Tính tự trị của các trạm
  - Cho phép mỗi trạm điều khiển và thao tác DL địa phương của nó độc lập với các trạm khác
  - Ưu việt là việc quản trị của CSDLPT có thể hoàn toàn phi tập trung.

# Mục tiêu của các hệ QTCSDLPT (tiếp)

- Tính mở rộng: Khả năng tăng trưởng bằng việc đưa thêm các trạm mới vào trong mạng với tác động tối thiểu trên các CSDLPT địa phương và các chương trình ứng dụng hiện có.
- Hiệu năng: là vấn đề sống còn của cách tiếp cận CSDLPT

Vấn đề cốt yếu là **phân đoạn** và **nhân bản DL** sao cho khai thác tốt tính song song



# Kiến trúc tham chiếu

---

- Kiến trúc các lược đồ:
  - Mở rộng kiến trúc 3 mức của nhóm ANSI/X3/SPARC cho các hệ CSDL tập trung.
  
- Kiến trúc chức năng



# Định nghĩa dữ liệu

---

- Mô tả các đối tượng
  - Việc mô tả DL thoát đầu được đưa vào trạm cục bộ (gọi là trạm tạo sinh), tại đây các DL được mô tả sẽ được tạo ra
  - Trạm tạo sinh và các trạm được phép truy cập các DL được mô tả sẽ hợp tác và trao đổi thông tin của mô tả DL
  - Trạm tạo sinh có thể biết danh sách các trạm được phép truy nhập các DL riêng của nó nhằm thông báo cho chúng mọi sửa đổi về mô tả của chúng



# Định nghĩa dữ liệu (tiếp)

---

- Sự phân đoạn: chia một quan hệ tổng thể thành các đơn vị logic của sự cấp phát (các đoạn) có thể được sắp đặt tối ưu trong CSDLPT

Sự phân đoạn phải được xác định bởi người quản trị CSDL và tuân thủ các qui tắc:

- Không mất thông tin: mỗi phần tử DL thuộc quan hệ tổng thể cũng thuộc một hay nhiều đoạn của nó
- Có khả năng khôi phục: xác định bằng phép toán ĐSQH
- Không trùng lặp (chỉ áp dụng cho phân đoạn ngang): buộc các đoạn phải rời nhau



# Định nghĩa dữ liệu (tiếp)

---

- Sự phân đoạn ngang
  - Phân đoạn ngang trực tiếp: phân hoạch một quan hệ thành các tập con các bộ, mỗi tập con được xác định bởi một phép chọn được áp dụng cho quan hệ
  - Phân đoạn ngang gián tiếp: phân hoạch một quan hệ thành các tập con các bộ, mỗi tập con được xác định bởi một quan hệ với mỗi đoạn của một quan hệ khác



# Định nghĩa dữ liệu (tiếp)

Ví dụ về phân đoạn ngang trực tiếp và gián tiếp

Ví dụ: Có các quan hệ sau:

RƯỢU\_VANG

SHRV	VÙNG_NHO	NIÊN_HIỆU	ĐỘ_RƯỢU	GIÁ
V1	Pháp	1982	13,2	?
V2	Bỉ	1987	12,9	7
V3	Pháp	1984	12,8	10
V4	Anh	1983	12,3	14

TIÊU\_THỤ

SHNU	SHRV	NGÀY	ĐỊA ĐIỂM	SỐ_LƯỢNG
B1	V1	12/04	Paris	5
B2	V3	25/09	Hà nội	2
B3	V3	12/04	Paris	7

# Định nghĩa dữ liệu (tiếp)

Ví dụ về phân đoạn ngang trực tiếp

Sự phân đoạn ngang sau

$$VANG\_1 = \sigma_{NIÊN\_HIỆU < 1984} RƯỢU\_VANG$$

$$VANG\_2 = \sigma_{NIÊN\_HIỆU \geq 1984} RƯỢU\_VANG$$

- không mất thông tin (nếu ta có giá trị null (kí hiệu bởi ?) là giá trị nhỏ nhất của mỗi miền)
- Khôi phục:  $RƯỢU\_VANG = VANG\_1 \cup VANG\_2$
- Không trùng lặp:  $VANG\_1 \cap VANG\_2 = \emptyset$

# Định nghĩa dữ liệu (tiếp)

Ví dụ về phân đoạn ngang gián tiếp

Sự phân đoạn ngang sau

$$\text{TIÊU\_THỤ}_1 = \text{TIÊU\_THỤ} \times \text{VANG}_1 = \{(B1, V1, 12-04, \text{Paris}, 5)\}$$

$$\text{TIÊU\_THỤ}_2 = \text{TIÊU\_THỤ} \times \text{VANG}_2 = \{(B2, V3, 25-09, \text{Hà Nội}, 2), \\ (B3, V3, 12-04, \text{Paris}, 7)\}$$

# Định nghĩa dữ liệu (tiếp)

- Phân đoạn dọc:
    - Phân hoạch một quan hệ thành các tập các bộ con, mỗi tập được xác định bởi một phép chiếu được áp dụng cho quan hệ
    - Để khôi phục: cần thêm vào mỗi đoạn một hay nhiều thuộc tính khoá, kết nối các đoạn theo các thuộc tính chung
- vi phạm quy tắc không trùng lặp (do thêm các thuộc tính chung)

# Định nghĩa dữ liệu (tiếp)

Ví dụ về phân đoạn dọc



---

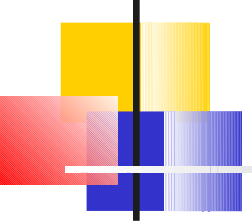
$$VANG1 = \pi_{SHRV, V\grave{U}NG\_NHO, NI\grave{E}N\_HI\grave{E}U} R\ddot{U}O\grave{U}\_VANG$$

$$VANG2 = \pi_{SHRV, ĐỘ\_R\ddot{U}O\grave{U}, GI\grave{A}} R\ddot{U}O\grave{U}\_VANG$$

Rõ ràng, khi đó:

$$R\ddot{U}O\grave{U}\_VANG = VANG1 \bowtie VANG2$$

# Định nghĩa dữ liệu (tiếp)

- 
- Sự phân đoạn hỗn hợp:
    - Phân đoạn một quan hệ thành các tập con các bộ con:
      - các bộ con được xác định bởi phân đoạn dọc
      - các tập con xác định bởi phân đoạn ngang
    - Cho phép tối ưu hoá các câu hỏi chọn\_chiều hay kết nối\_chiều

# Định nghĩa dữ liệu (tiếp)

Ví dụ về phân đoạn hỗn hợp

$$VANG1 = \pi_{SHRV, ĐỘ\_RƯỢU, GIÁ} RƯỢU\_VANG$$

$$VANG2 = \sigma_{NIÊN\_HIỆU < 1984} (\pi_{SHRV, VÙNG\_NHO, NIÊN\_HIỆU} RƯỢU\_VANG)$$

$$VANG3 = \sigma_{NIÊN\_HIỆU \geq 1984} (\pi_{SHRV, VÙNG\_NHO, NIÊN\_HIỆU} RƯỢU\_VANG)$$

Rõ ràng, khi đó:

$$RƯỢU\_VANG = VANG1 \bowtie (VANG2 \cup VANG3)$$

*Việc lựa chọn phân đoạn là khó vì nó phụ thuộc vào nhu cầu truy nhập của các ứng dụng*



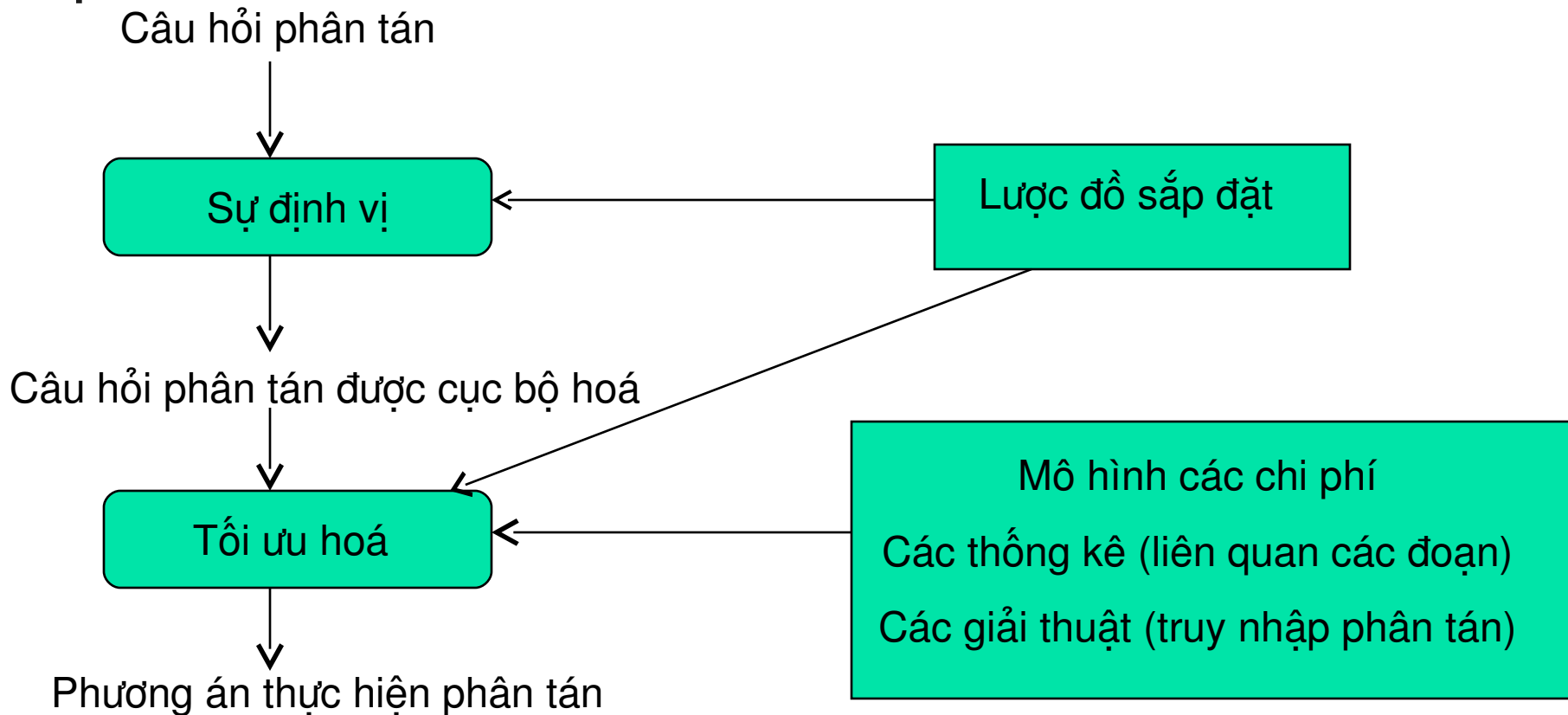
# Đánh giá các câu hỏi phân tán

---

- Phân tách: dịch câu hỏi trong ngôn ngữ bậc cao (SQL) sang một biến thái của đại số quan hệ
- Định vị (khoanh vùng)
  - Thông tin cần thiết cho sự định vị được lưu giữ trong lược đồ sắp chỗ
  - Định vị câu hỏi phân tán gồm hai giai đoạn:
    - Phát sinh câu hỏi chính tắc tương đương
    - sự giản lược: cho phép loại bỏ các cây con vô ích
- Tối ưu hóa:
- Thực hiện các thao tác (phép toán) của câu hỏi đã được tối ưu hóa để thu được kết quả



# Đánh giá các câu hỏi phân tán



*Các giai đoạn của việc đánh giá một câu hỏi phân tán*

Phạm Thị Anh Lê - ĐHSPHN



# Đánh giá các câu hỏi phân tán

---

- Sự định vị (khoanh vùng)
  - Thông tin cần thiết cho sự định vị được lưu giữ trong lược đồ sắp chỗ
  - Định vị câu hỏi phân tán gồm hai giai đoạn:
    - Phát sinh câu hỏi chính tắc tương đương
    - sự giản lược: cho phép loại bỏ các cây con vô ích



# Đánh giá các câu hỏi phân tán

---

Ví dụ, câu hỏi phân tán « các vùng nho của các rượu vang được tiêu thụ ở Paris » → các phân đoạn sau:

$$\text{VANG1} = \pi_{\text{SHRV, VÙNG\_NHO, NIÊN\_HIỆU}} \text{RƯỢU\_VANG}$$

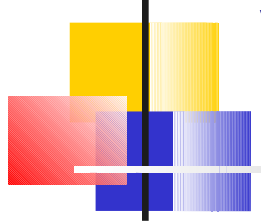
$$\text{VANG2} = \pi_{\text{SHRV, ĐỘ\_RƯỢU, GIÁ}} \text{RƯỢU\_VANG}$$

$$\text{TIÊU\_THỤ1} = \sigma_{\text{ĐỊA\_ĐIỂM}=\text{« Paris »}} \text{TIÊU\_THỤ}$$

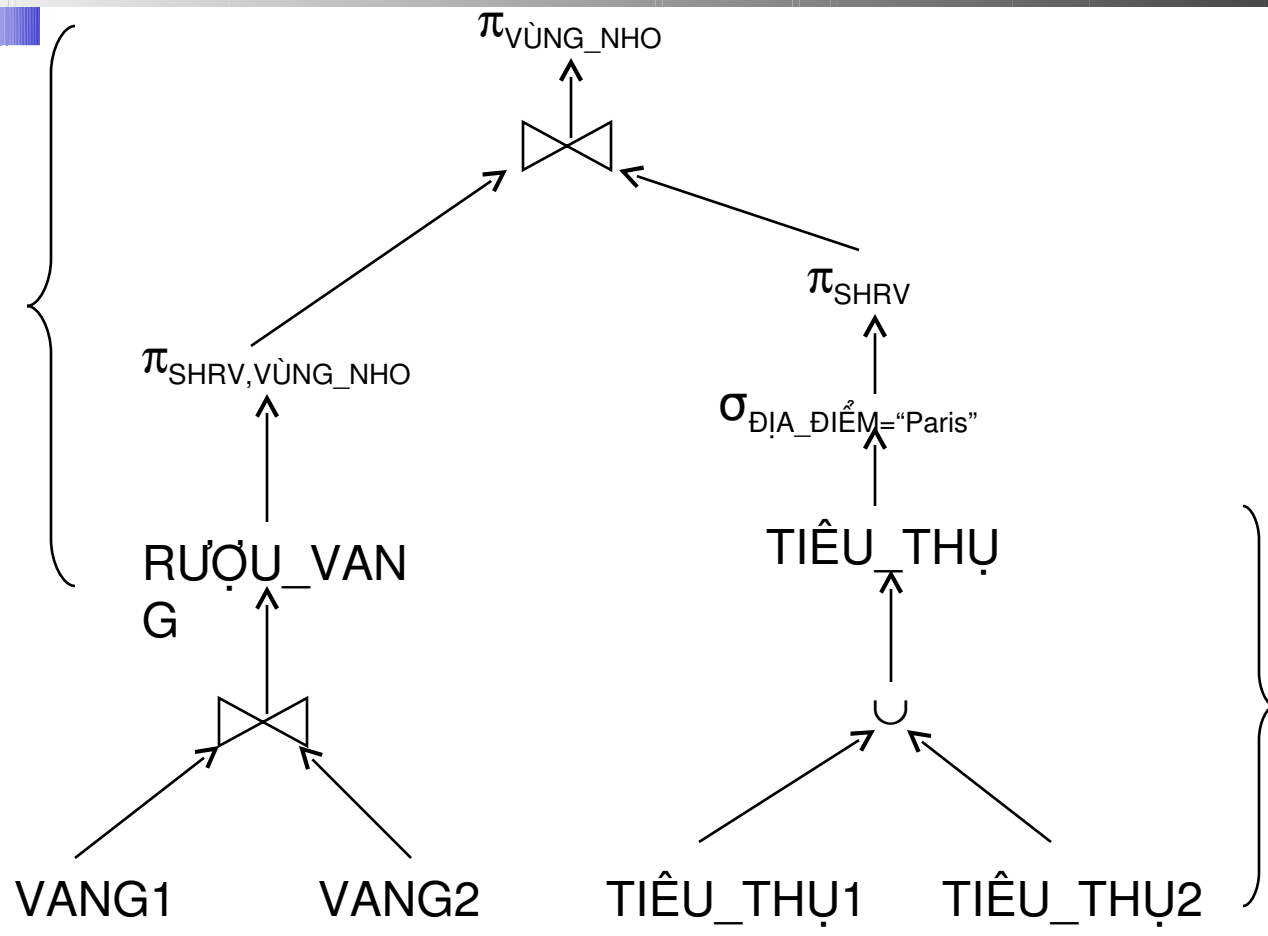
$$\text{TIÊU\_THỤ2} = \sigma_{\text{ĐỊA\_ĐIỂM} \neq \text{« Paris »}} \text{TIÊU\_THỤ}$$

# Đánh giá các câu hỏi phân tán

## Ví dụ về câu hỏi phân tán chính tắc



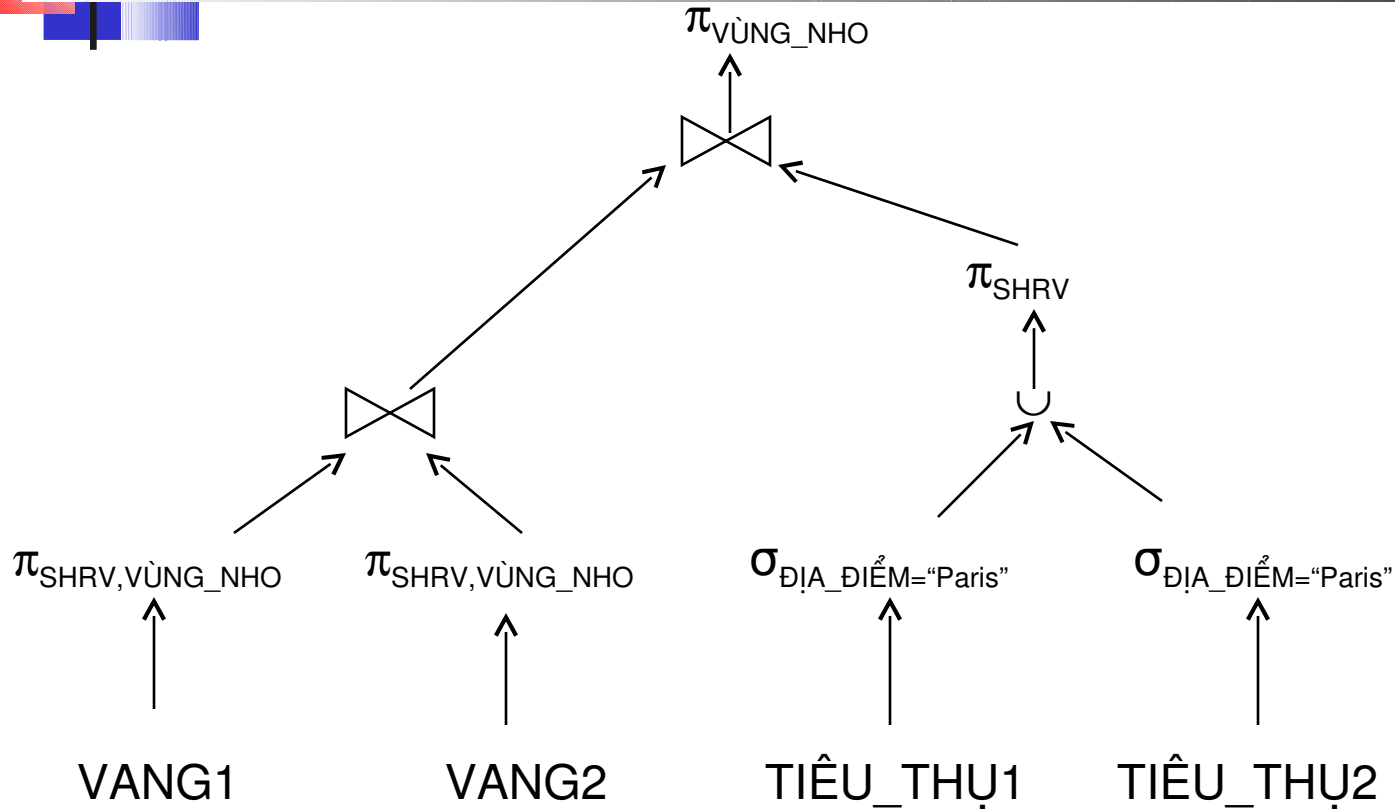
Câu hỏi tổng thể



các câu hỏi của sự khôi phục

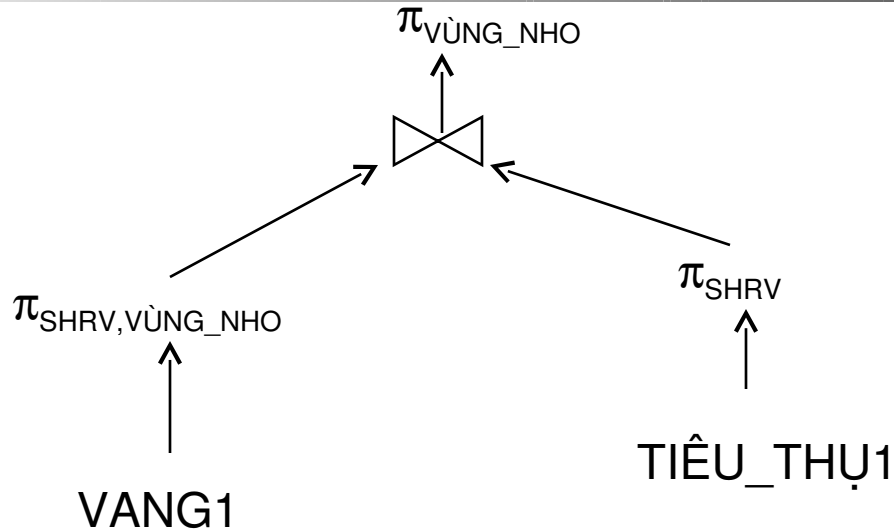
# Đánh giá các câu hỏi phân tán

## Ví dụ về câu hỏi sau khi cấu trúc lại



# Đánh giá các câu hỏi phân tán

## Ví dụ về câu hỏi sau khi giải lược





# Đánh giá các câu hỏi phân tán

---

- Tối ưu hoá: Xác định một chiến lược thực hiện câu hỏi làm cực tiểu một hàm chi phí
  - Hàm chi phí phải cực tiểu là **tổng thời gian** của việc thực hiện câu hỏi, và tổng tất cả thời gian thực hiện giành cho các trạm khác nhau tham gia vào câu hỏi
  - Hàm chi phí thường là **thời gian trả lời câu hỏi**, có tính đến các xử lý được tiến hành song song.



# Quản lý các giao tác phân tán

---

- Điều khiển tương tranh
  - Thời dẫu
  - Sự đặt khoá
- Sự hợp thức hoá giao tác



# Quản lý các giao tác phân tán

## Điều khiển tương tranh

---

- Điều khiển tương tranh phân tán nhằm ngăn chặn việc sản sinh ra các thực hiện không khả tuần tự của các giao tác phân tán

# Quản lý các giao tác phân tán

## Điều khiển tương tranh

---

- Thời dấu (time stamp):
  - Sắp xếp các giao tác phân tán khi cho thực hiện chúng và áp đặt các thao tác truy nhập tới DL tôn trọng thứ tự được xác định trước
  - Mỗi giao tác được dẫn trở bởi một số hiệu thứ tự duy nhất trong hệ thống, gọi là **thời dấu**
  - Mỗi trạm đánh thời dấu một cách tự trị theo đồng hồ địa phương và số hiệu của trạm. Ví dụ, một giao tác được khởi phát ở trạm  $i$  tại thời điểm  $t$  sẽ có thời dấu  $\langle t, i \rangle$ .

# Quản lý các giao tác phân tán

## Điều khiển tương tranh

- Sự đặt khoá (locking)
  - Việc truy nhập tới các hạt DL được chi phối bởi một giao thức gồm hai giao tác nguyên thủy:
    - LOCK: yêu cầu thao tác trên các hạt DL theo một cách thức nhất định (đọc và viết)
    - UNLOCK: báo hiệu sự kết thúc của thao tác trên hạt
  - Việc truy nhập tới một hạt bị khoá trong một lối không tương thích → giao tác truy nhập phải chờ cho tới khi mở khoá bởi giao tác tác động lên hạt.
  - Vấn đề khó nhất do việc đặt khoá là tình huống bế tắc:  $T_1 \rightarrow T_2, T_2 \rightarrow T_1$



# Quản lý các giao tác phân tán

---

## Sự hợp thức hoá giao tác

- Phương pháp trực tiếp để hợp thức hoá một giao tác phân tán: tích hợp thực sự các cập nhật của giao tác và CSDL
- Giao thức hợp thức hoá hai giai đoạn:
  - Giai đoạn chuẩn bị: trạm điều phối yêu cầu mỗi trạm tham gia chuẩn bị cho sự hợp thức hoá
  - Giai đoạn hợp thức: trạm điều phối ra lệnh cho tất cả các trạm tham gia hợp thức hoá các cập nhật của chúng nếu như tất cả chúng đều hoàn thành đúng giai đoạn thứ nhất hay nếu không thì huỷ bỏ chúng



**Xin cảm ơn !**

---

