TỐI ƯU HÓA TRUY VẤN TRONG CSDL PHÂN TÁN

Giảng viên: ThS. Nguyễn Hồ Duy Tri
tringuyen@uit.edu.vn

Nội dung

- Giới thiệu về xử lý truy vấn
- Xử lý truy vấn trong môi trường tập trung
- Xử lý truy vấn trong môi trường phân tán
- Tối ưu hoá truy vấn trong CSDL phân tán

Mục đích

- Giới thiệu một bức tranh tổng quát của bộ tối ưu hóa truy vấn trong môi trường tập trung và phân tán
- Đề cập đến vấn đề tối ưu hoá trong cơ sở dữ liệu phân tán nghĩa là: **giảm chi phí bộ nhớ trung gian**, **giảm thời gian truy vấn** cũng như **giảm thời gian truyền dữ liệu** trong các truy vấn phân tán.
- Trình bày các quy trình xử lý truy vấn trong hệ thống phân tán.

Giới thiệu về xử lý truy vấn

- **Mục đích của xử lý truy vấn:**
  - ✓ Giảm thiểu ***thời gian xử lý***
  - ✓ Giảm ***vùng nhớ trung gian***
  - ✓ Giảm ***chi phí truyền thông*** giữa các trạm
  - ✓ Sử dụng ít ***tài nguyên***.
- **Chức năng của xử lý truy vấn:**
  - ✓ Biến đổi một truy vấn ***phức tạp*** thành một truy vấn tương đương ***đơn giản*** hơn.
  - ✓ Phép biến đổi này phải đạt được cả về tính ***đúng đắn*** và ***hiệu quả***
  - ✓ Mỗi cách biến đổi dẫn đến việc sử dụng tài nguyên máy tính khác nhau, nên vấn đề đặt ra là ***lựa chọn phương án nào dùng tài nguyên ít nhất***.

Giới thiệu về xử lý truy vấn

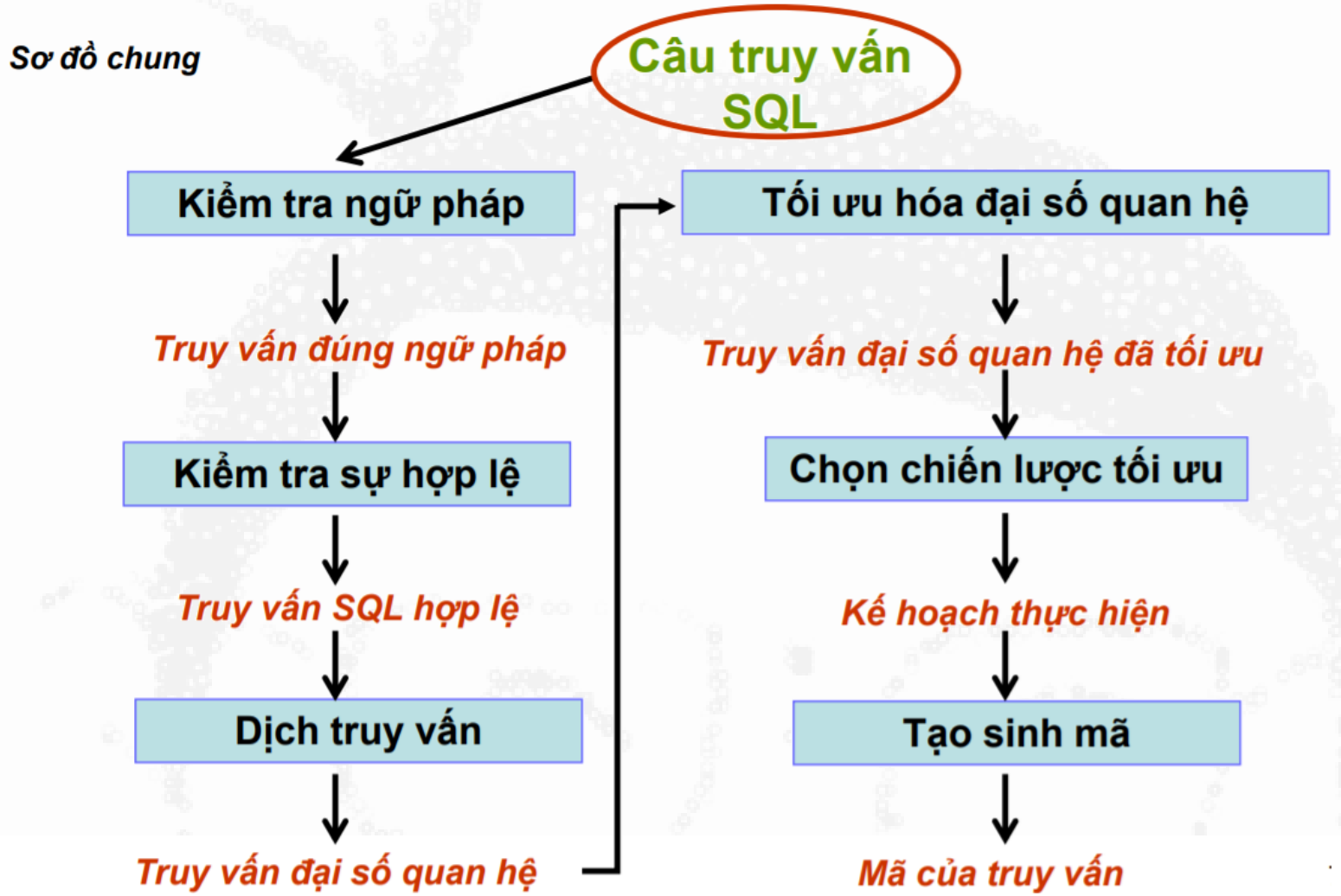
- **Các phương pháp xử lý truy vấn cơ bản**
  - ✓ *Phương pháp biến đổi đại số:*
    - ❖ Đơn giản hóa câu truy vấn nhờ các **phép biến đổi đại số tương đương** nhằm giảm thiểu thời gian thực hiện các phép toán.
    - ❖ Phương pháp này **không quan tâm đến kích thước** và **cấu trúc dữ liệu**.
  - ✓ *Phương pháp ước lượng chi phí:*
    - ❖ **Xác định kích thước dữ liệu**, **thời gian thực hiện** mỗi phép toán trong câu truy vấn.
    - ❖ Phương pháp này **quan tâm đến kích thước dữ liệu** và phải **tính toán chi phí thời gian** thực hiện mỗi phép toán.

Xử lý truy vấn trong môi trường tập trung

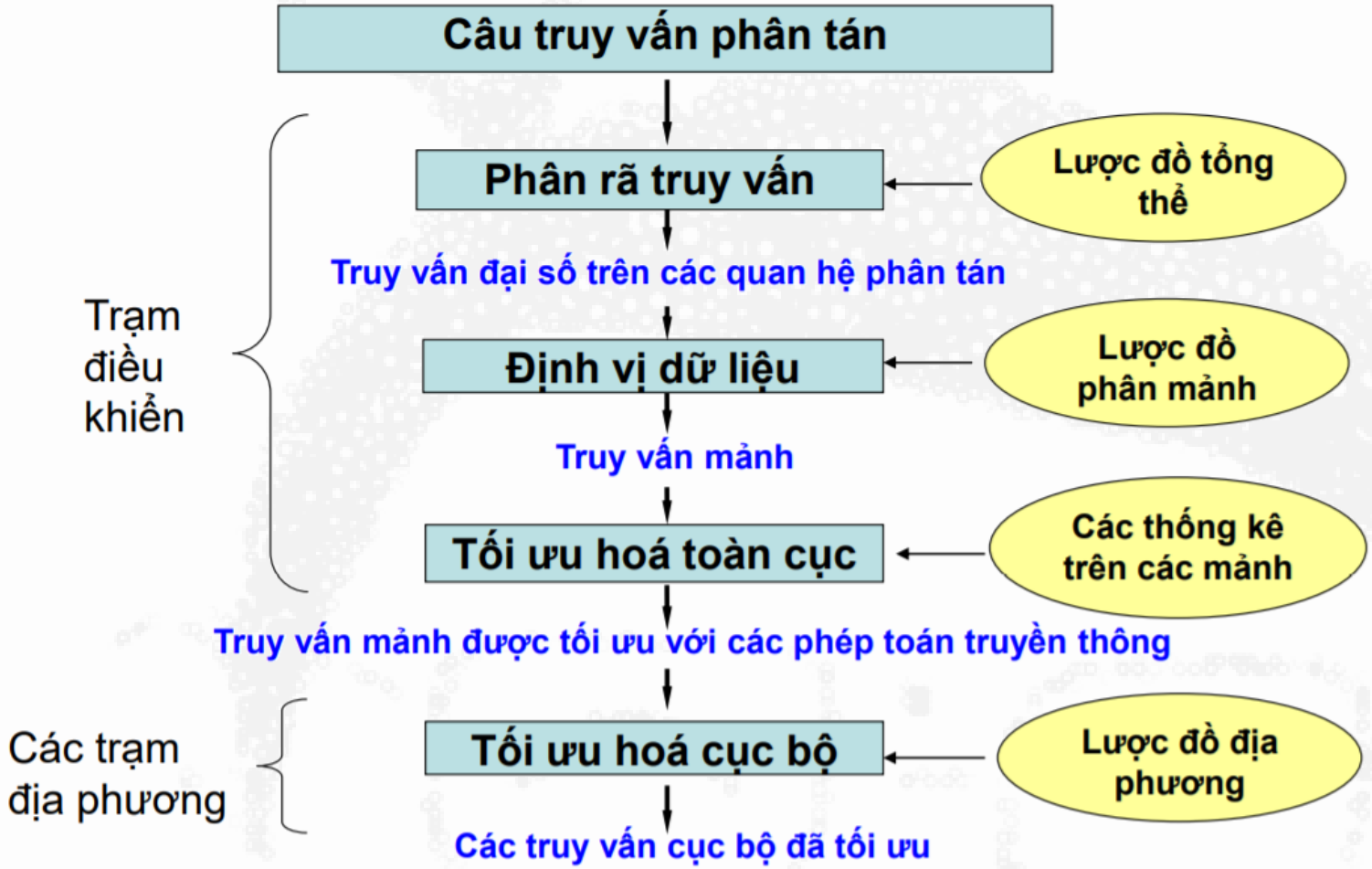
So sánh xử lý truy vấn tập trung và phân tán

- ✓ *Tập trung:*
  - ❖ Chọn một **truy vấn đại số quan hệ tốt nhất** trong số tất cả các truy vấn đại số tương đương.
  - ❖ Các chiến lược xử lý truy vấn có thể **biểu diễn trong sự mở rộng của đại số quan hệ**.
- ✓ *Phân tán*
  - ❖ Kế thừa chiến lược xử lý truy vấn như môi trường tập trung
  - ❖ Còn phải ***quan tâm thêm***
    - ❑ Các phép toán truyền dữ liệu giữa các trạm
    - ❑ Chọn các trạm tốt nhất để xử lý dữ liệu
    - ❑ Cách truyền dữ liệu

TỐI ƯU HÓA TRUY VẤN
Trong môi trường tập trung



TỐI ƯU HÓA TRUY VẤN
Trong môi trường phân tán



Sơ đồ phân lớp chung cho xử lý truy vấn phân tán

Xử lý truy vấn trong môi trường tập trung

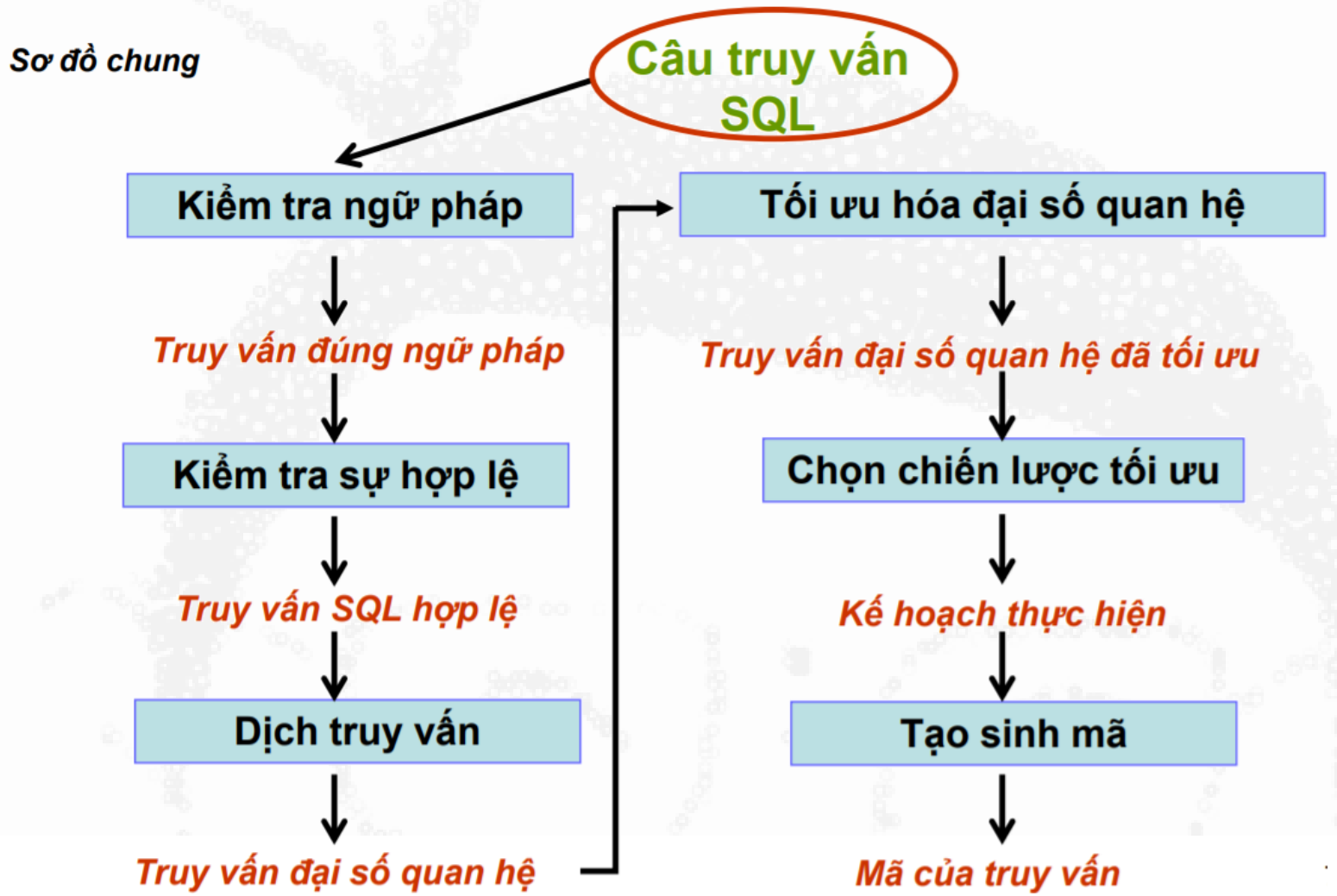
Tối ưu hóa trong CSDL tập trung

Tại sao phải nghiên cứu xử lý truy vấn tập trung?

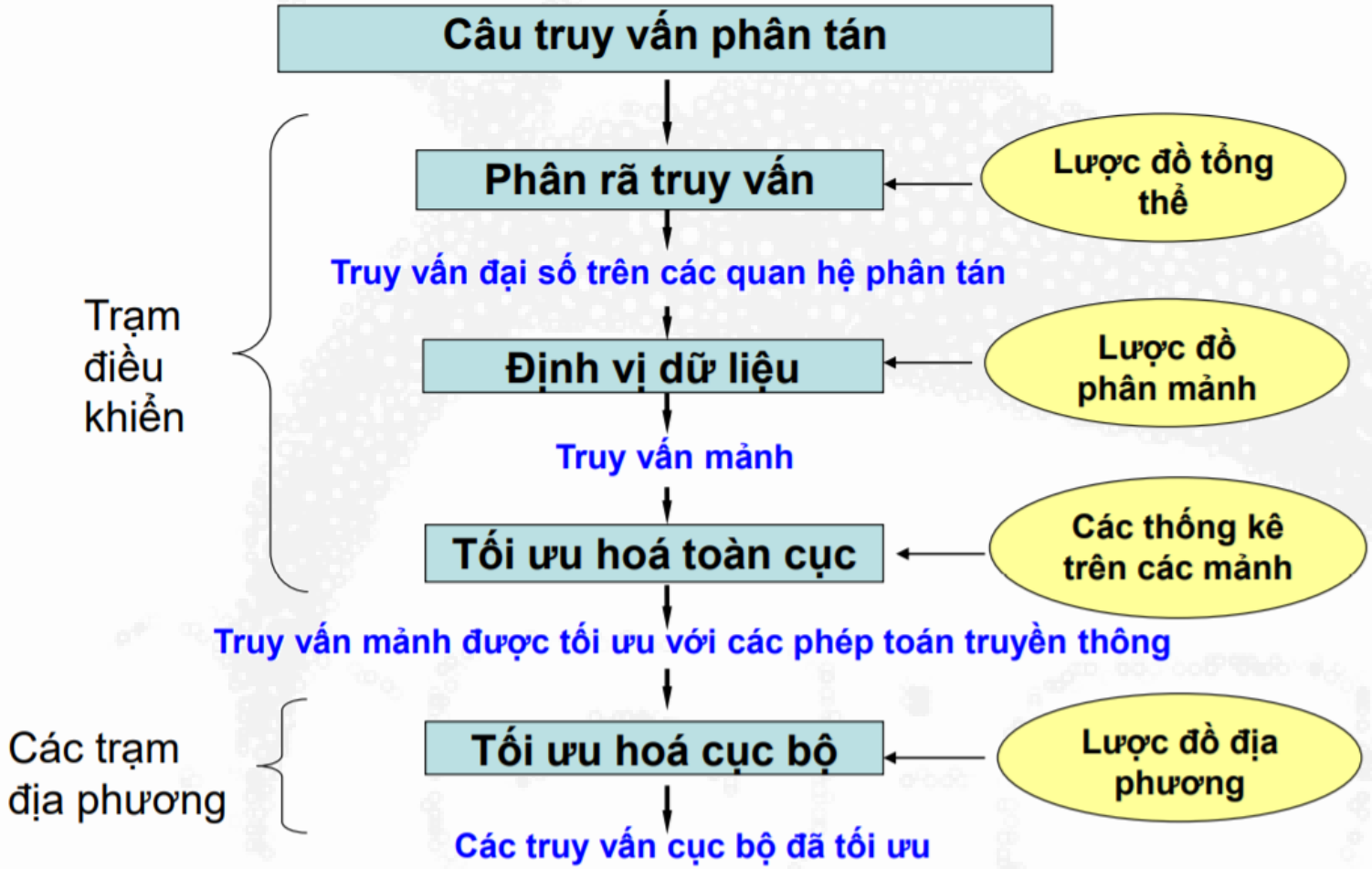
Để hiểu được các kỹ thuật tối ưu phân tán vì ba lí do:

- ✓ *Câu truy vấn phân tán **phải được dịch thành các câu truy vấn cục bộ** và **được xử lý theo phương pháp tập trung**.*
- ✓ *Các kỹ thuật tối ưu hoá phân tán thường là các **mở rộng của kỹ thuật tập trung**.*
- ✓ ***Tối ưu hoá tập trung thường đơn giản**.*

TỐI ƯU HÓA TRUY VẤN
Trong môi trường tập trung



TỐI ƯU HÓA TRUY VẤN
Trong môi trường phân tán



Sơ đồ phân lớp chung cho xử lý truy vấn phân tán

Xử lý truy vấn trong môi trường phân tán

- **Phân rã truy vấn - Tối ưu hóa toàn cục**

  Bước này **giống** với các **bước 1, 2, 3 và 4** của **tối ưu hóa truy vấn** trong cơ sở dữ liệu **tập trung**.

  Giai đoạn *Phân rã truy vấn* chia làm **bốn** bước:

  ✓ Chuẩn hoá

  ✓ Phân tích

  ✓ Loại bỏ dư thừa

  ✓ Viết lại

Xử lý truy vấn trong môi trường phân tán

✓ Chuẩn hoá

- ***Mục đích:*** Chuyển đổi truy vấn thành một dạng chuẩn để thuận lợi cho các xử lý tiếp theo.
- Với SQL, có **hai dạng chuẩn** cho các **vị từ** trong mệnh đề WHERE là:
  - ***Dạng chuẩn hội*** là **hội** ($\wedge$) của những phép toán **tuyển** ($\vee$):
    $(p_{11} \vee p_{12} \vee \cdots \vee p_{1n}) \wedge \cdots \wedge (p_{m1} \vee p_{m2} \vee \cdots \vee p_{mn})$
  - ***Dạng chuẩn tuyển*** là **tuyển** ($\vee$) của những phép toán **hội** ($\wedge$):
    $(p_{11} \wedge p_{12} \wedge \cdots \wedge p_{1n}) \vee \cdots \vee (p_{m1} \wedge p_{m2} \wedge \cdots \wedge p_{mn})$

  trong đó p_{ij} là các biểu thức nguyên tố.

Đại số mệnh đề

✓ Bảng các tương đương logic thường dùng

- Đặt T = hằng đúng, F = hằng sai

1	$p \wedge F \Leftrightarrow F$	Domination laws (luật thống trị)	6	$p \wedge p \Leftrightarrow p$	Idempotent laws (luật lũy đẳng)
2	$p \vee T \Leftrightarrow T$		7	$p \vee p \Leftrightarrow p$	
3	$p \vee F \Leftrightarrow p$	Identity laws (luật đồng nhất)	8	$p \wedge \neg p \Leftrightarrow F$	Cancellation laws (luật xóa bỏ)
4	$p \wedge T \Leftrightarrow p$		9	$p \vee \neg p \Leftrightarrow T$	
5	$\neg(\neg p) \Leftrightarrow p$	Double negation law (luật phủ định kép)	10	$p \wedge q \Leftrightarrow q \wedge p$	Commutative law (luật giao hoán)
			11	$p \vee q \Leftrightarrow q \vee p$	

Đại số mệnh đề

✓ Bảng các tương đương logic thường dùng (tt)

12	$(p \wedge q) \wedge r \Leftrightarrow p \wedge (q \wedge r)$	Associative laws (luật kết hợp)
13	$(p \vee q) \vee r \Leftrightarrow p \vee (q \vee r)$	
14	$p \wedge (q \vee r) \Leftrightarrow (p \wedge q) \vee (p \wedge r)$	Distributive laws (luật phân phối)
15	$p \vee (q \wedge r) \Leftrightarrow (p \vee q) \wedge (p \vee r)$	
16	$\neg(p \vee q) \Leftrightarrow \neg p \wedge \neg q$	De Morgan's laws (luật De Morgan)
17	$\neg(p \wedge q) \Leftrightarrow \neg p \vee \neg q$	
18	$(p \Rightarrow q) \Leftrightarrow (\neg p \vee q)$	Implication law (luật kéo theo)
19	$p \vee (p \wedge q) \Leftrightarrow p$	Absorption laws (luật hấp thu)
20	$p \wedge (p \vee q) \Leftrightarrow p$	

Xử lý truy vấn trong môi trường phân tán

✓ Ví dụ: xét CSDL công ty phần mềm đã cho

➢ Có các quan hệ: E = **NHANVIEN** (MANV, TENNV, CHUCVU) *và*
G = **HOSO** (MANV, MADA, NHIEMVU, THOIGIAN).

- Xét truy vấn: "*Tìm **tên** các nhân viên làm dự án có **mã số J1** với **thời gian 12** hoặc **24 tháng***".
- Truy vấn trên được biểu diễn trong SQL:

```
SELECT E. TENNV
FROM E, G
WHERE E.MANV = G.MANV
    AND G.MADA = "J1"
    AND (THOIGIAN = 12 OR THOIGIAN = 24)
```

- Điều kiện trong **dạng chuẩn hội** là:

$$E.MANV=G.MANV \wedge G.MADA="J1" \wedge (THOIGIAN=12 \vee THOIGIAN=24)$$

- Điều kiện trong **dạng chuẩn tuyển** là:

$$(E.MANV=G.MANV \wedge G.MADA="J1" \wedge THOIGIAN=12) \vee$$
$$(E.MANV=G.MANV \wedge G.MADA="J1" \wedge THOIGIAN=24)$$

Xử lý truy vấn trong môi trường phân tán

✓ Phân tích

- ***Mục đích:*** Phát hiện ra những thành phần không đúng (**sai kiểu** hoặc **sai ngữ nghĩa**) và loại bỏ chúng sớm nhất nếu có thể.
- ***Truy vấn sai kiểu:*** nếu một thuộc tính bất kỳ hoặc tên quan hệ của nó không được định nghĩa trong lược đồ tổng thể, hoặc phép toán áp dụng cho các thuộc tính sai kiểu.
  - *Ví dụ:* Truy vấn dưới đây là **sai kiểu**

```
SELECT E#
FROM E
WHERE E.TENNV > 200
```

vì hai lý do:

- Thuộc tính E# không khai báo trong lược đồ
- Phép toán “>200” không thích hợp với kiểu chuỗi của thuộc tính E.TENNV

Xử lý truy vấn trong môi trường phân tán

- ***Truy vấn sai ngữ nghĩa:*** nếu các thành phần của nó không tham gia vào việc tạo ra kết quả.

  Để xác định truy vấn có sai về ngữ nghĩa hay không, ta dựa trên việc biểu diễn truy vấn như một đồ thị gọi là **đồ thị truy vấn**.

  Đồ thị này được xác định bởi các truy vấn liên quan đến phép chọn, chiếu và nối. Nếu đồ thị truy vấn mà **không liên thông** thì truy vấn là **sai ngữ nghĩa**.

Xử lý truy vấn trong môi trường phân tán

- ***Đồ thị truy vấn:***
  - Có **một nút** dùng để biểu diễn cho quan hệ **kết quả**
  - Các **nút khác** biểu diễn cho các toán hạng trong câu truy vấn **(các quan hệ)**
  - **Cạnh nối** giữa hai nút mà **không phải là nút kết quả** thì biểu diễn một **phép nối**.
  - Cạnh có **nút đích là nút kết quả** thì biểu diễn một **phép chiếu**.
  - Một **nút không phải là nút kết quả** có thể được gán nhãn bởi **phép chọn** hoặc **phép tự nối** (seft-join: nối của quan hệ với chính nó).
- ***Đồ thị kết nối:***
  - Là một **đồ thị con của đồ thị truy vấn** (join graph), trong đó **chỉ có phép nối**.

Xử lý truy vấn trong môi trường phân tán

- ***Ví dụ:*** Từ các quan hệ:

  E = **NHANVIEN** (MANV, TENNV, CHUCVU),

  G = **HOSO** (MANV, MADA, NHIEMVU, THOIGIAN) và

  J = **DUAN** (MADA, TENDA, NGANSACH).

  Hãy xác định "**Tên** và **nhiệm vụ** của các nhân viên có **chức vụ** là **lập trình viên** làm **dự án CSDL** có **thời gian** lớn hơn **3 năm**."

- Truy vấn SQL tương ứng là: **SELECT** TENNV, NHIEMVU

  FROM E, G, J

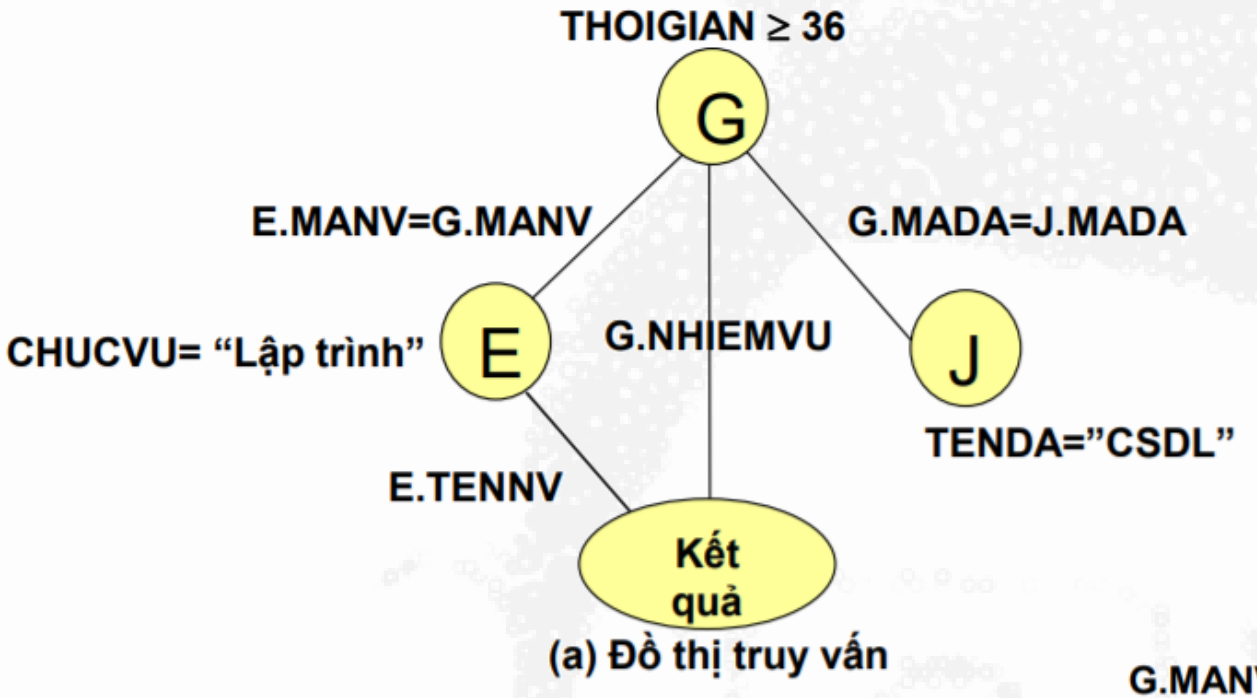
  WHERE E.MANV=G.MANV **AND** G.MADA=J.MADA

  AND TENDA="CSDL" **AND** THOIGIAN≥36

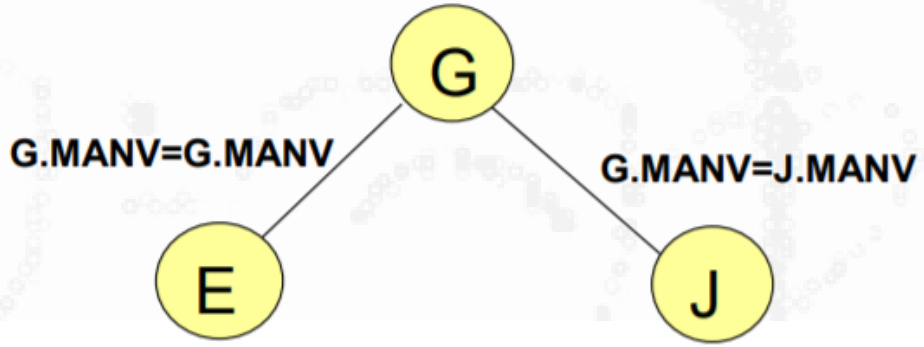
  AND CHUCVU="LTRINH"

Xử lý truy vấn trong môi trường phân tán

- Đồ thị truy vấn và đồ thị kết nối tương ứng:



(a) Đồ thị truy vấn



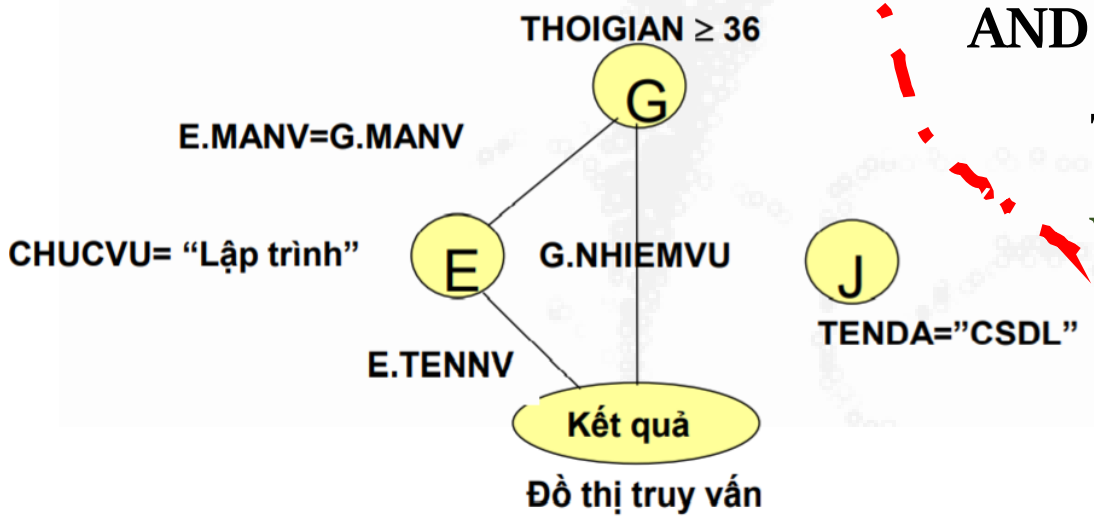
(b) Đồ thị kết nối tương ứng

Xử lý truy vấn trong môi trường phân tán

- Xét câu truy vấn SQL là:

thiếu **AND** G.MADA=J.MADA

```
SELECT TENNV, NHIEMVU
FROM   E, G, J
WHERE  E.MANV=G.MANV
   AND TENDA="CSDL"   AND THOIGIAN≥36
   AND CHUCVU="LTRINH"
```



Truy vấn này là **sai ngữ nghĩa** vì **đồ thị truy vấn** của nó **không liên thông**.

Xử lý truy vấn trong môi trường phân tán

✓ Loại bỏ dư thừa

- Điều kiện trong các truy vấn **có thể có chứa các vị từ dư thừa**.
- Một đánh giá sơ sài về một điều kiện dư thừa có thể dẫn đến **lặp lại** một số công việc.
- Sự dư thừa vị từ và dư thừa công việc có thể được loại bỏ bằng cách làm đơn giản hoá các điều kiện thông qua các **luật lũy đẳng** sau:

1. $p \wedge p \Leftrightarrow p$	2. $p \vee true \Leftrightarrow true$
3. $p \vee p \Leftrightarrow p$	4. $p \wedge \neg p \Leftrightarrow false$
5. $p \wedge true \Leftrightarrow p$	6. $p \vee \neg p \Leftrightarrow true$
7. $p \vee false \Leftrightarrow p$	8. $p_1 \wedge (p_1 \vee p_2) \Leftrightarrow p_1$
9. $p \wedge false \Leftrightarrow false$	10. $p_1 \vee (p_1 \wedge p_2) \Leftrightarrow p_1$

Xử lý truy vấn trong môi trường phân tán

- ***Ví dụ:*** Đơn giản hoá câu truy vấn sau:

```
SELECT E.CHUCVU
FROM   E
WHERE  (NOT(E.CHUCVU="Lập trình")
    AND  (E.CHUCVU="Lập trình" OR E.CHUCVU="Kỹ sư điện")
    AND NOT(E.CHUCVU="Kỹ sư điện"))
     OR E.TENNV="Dũng"
```

Xử lý truy vấn trong môi trường phân tán

- ***Ví dụ (tt):***

Đặt p1: E.CHUCVU="Lập trình", p2: E.CHUCVU="Kỹ sư điện", p3: E.TENNV="Dũng".

Các vị từ sau mệnh đề **WHERE** được mô tả lại:

p: $(\neg p1 \wedge (p1 \vee p2) \wedge \neg p2) \vee p3$

$\Leftrightarrow ((\neg p1 \wedge p1 \wedge \neg p2) \vee (\neg p1 \wedge p2 \wedge \neg p2)) \vee p3$

$\Leftrightarrow ((fasle \wedge \neg p2) \vee (\neg p1 \wedge false)) \vee p3$ (*áp dụng luật **4***)

$\Leftrightarrow (fasle \vee false) \vee p3$ (*áp dụng luật **9***)

$\Leftrightarrow$ **p3** (*áp dụng luật **7***)

Vậy câu truy vấn được biến đổi thành: **SELECT** E.CHUCVU

FROM E

WHERE E.TENNV="Dũng"

Xử lý truy vấn trong môi trường phân tán

✓Viết lại

Bước này được chia làm hai bước con như sau:

- **Biến đổi** trực tiếp truy vấn phép tính sang **đại số quan hệ**.
- **Cấu trúc lại** truy vấn đại số quan hệ để **cải thiện hiệu quả** thực hiện.

Đại số quan hệ là một **cây** mà **nút lá** biểu diễn một **quan hệ** trong CSDL, các **nút không lá** là các **quan hệ trung gian** được sinh ra bởi các phép toán đại số quan hệ.

Xử lý truy vấn trong môi trường phân tán

Cách chuyển một truy vấn phép tính quan hệ thành một cây đại số quan hệ:

- Các **nút lá** khác nhau được tạo cho mỗi biến bộ khác nhau (tương ứng một quan hệ). Trong SQL các nút lá chính là các **quan hệ** trong mệnh đề **FROM**.
- **Nút gốc** được tạo ra bởi một **phép chiếu** lên các **thuộc tính kết quả**. Trong SQL nút gốc được xác định qua mệnh đề **SELECT**.
- **Điều kiện** (mệnh đề **WHERE** trong SQL) được biến đổi thành **dãy các phép toán đại số** thích hợp (phép chọn, nối, phép hợp, v.v...) đi từ lá đến gốc, có thể thực hiện theo thứ tự xuất hiện của các vị từ và các phép toán.

Xử lý truy vấn trong môi trường phân tán

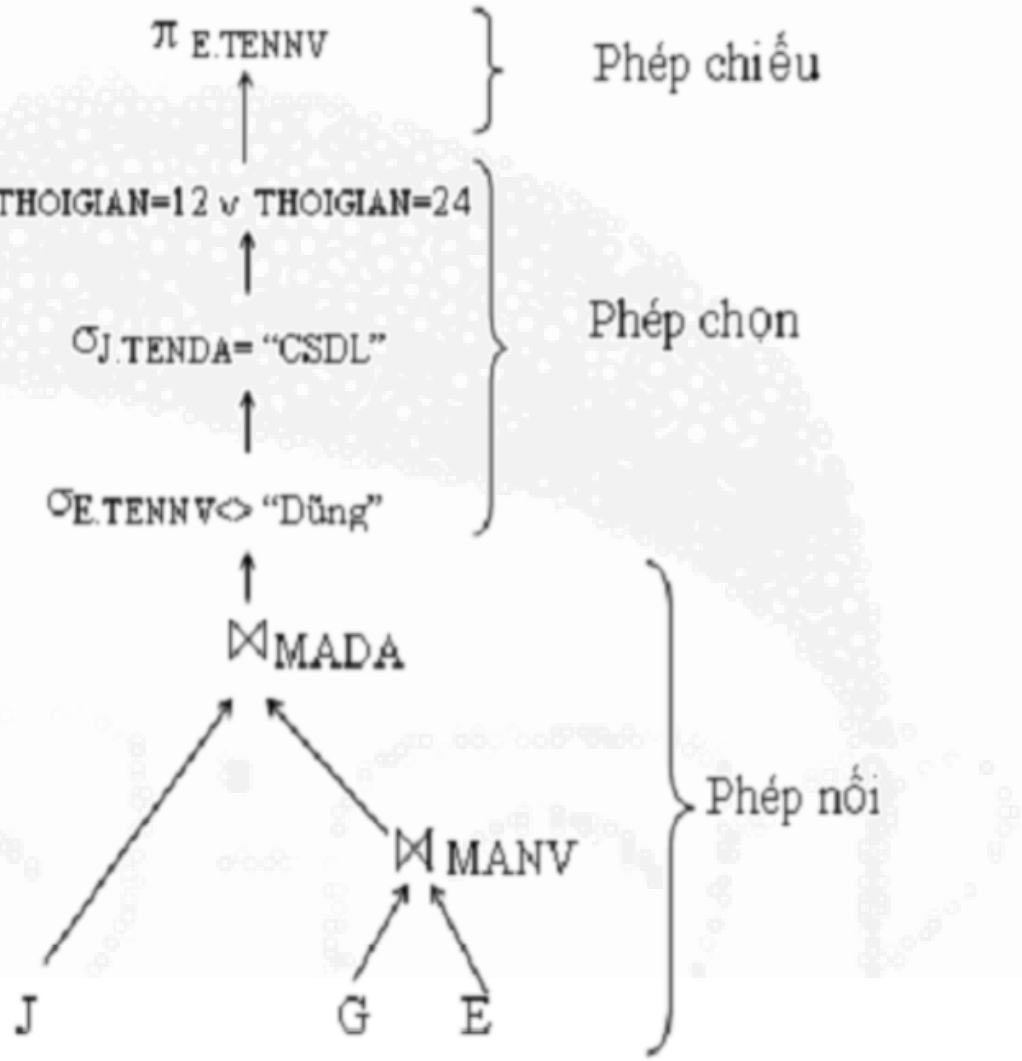
Ví dụ:

- Truy vấn "*Tìm tên các nhân viên không phải là "Dũng", làm việc cho dự án CSDL với thời gian một hoặc hai năm*".
- Biểu diễn truy vấn này trong SQL là:

```
SELECT E.TENNV
FROM   J, G, E
WHERE G.MANV=E.MANV
    AND G.MADA=J.MADA
    AND E.TENNV<>"Dũng"
    AND J.TENDA="CSDL"
    AND (THOIGIAN=12 OR THOIGIAN=24)
```

Xử lý truy vấn trong môi trường phân tán

```
SELECT E.TENNV
FROM   J, G, E
WHERE G.MANV=E.MANV
      AND G.MADA=J.MADA
      AND E.TENNV<>“Dũng”
      AND J.TENDA=“CSDL”
      AND (THOIGIAN=12
        OR THOIGIAN=24)
```



Xử lý truy vấn trong môi trường phân tán

o6 luật biến đổi phép toán đại số quan hệ:

- ***Mục đích:*** dùng để biến đổi cây đại số quan hệ thành các cây tương đương (trong đó có thể có cây tối ưu).

  Giả sử R, S, T là các quan hệ, R được định nghĩa trên toàn bộ thuộc tính $A=\{A_1, ..., A_n\}$, S được định nghĩa trên toàn bộ thuộc tính $B=\{B_1, ..., B_n\}$.

  ❖ ***Tính giao hoán của các phép toán hai ngôi:***

  Phép tích Decartes và phép nối hai quan hệ có tính giao hoán.

  $R \times S \Leftrightarrow S \times R$ $\qquad$ $R \bowtie S \Leftrightarrow S \bowtie R$

  ❖ ***Tính kết hợp của các phép toán hai ngôi:***

  Phép tích Decartes và phép nối hai quan hệ có tính kết hợp.

  $(R \times S) \times T \Leftrightarrow R \times (S \times T)$ $\qquad$ $(R \bowtie S) \bowtie T \Leftrightarrow R \bowtie (S \bowtie T)$

Xử lý truy vấn trong môi trường phân tán

❖ ***Tính luỹ đẳng của những phép toán một ngôi:***

Dãy các phép chiếu khác nhau trên cùng quan hệ được tổ hợp thành **một phép chiếu** và ngược lại:

$$\pi_{A'}\left(\pi_{A''}(R)\right) \Leftrightarrow \pi_{A'}(R), \qquad với\ A', A'' \subseteq R\ và\ A' \subseteq A''$$

Dãy các phép chọn khác nhau $\sigma_{p_i(A_i)}$ trên cùng một quan hệ, với p_i là một vị từ được gán vào thuộc tính A_i, có thể được tổ hợp thành **một phép chọn**.

$$\sigma_{p_1(A_1)}\left(\sigma_{p_2(A_2)}(R)\right) = \sigma_{p_1(A_1) \wedge p_2(A_2)}(R)$$

Xử lý truy vấn trong môi trường phân tán

❖ ***Phép chọn giao hoán với phép chiếu:***

$$\pi_{A_1,\ldots,A_n}\left(\sigma_{p(A_p)}(R)\right) = \pi_{A_1,\ldots,A_n}\left(\sigma_{p(A_p)}\left(\pi_{A_1,\ldots,A_n,A_p}(R)\right)\right)$$

Nếu A_p là thành viên của $\{A_1, \ldots, A_n\}$, biểu thức trên trở thành:

$$\pi_{A_1,\ldots,A_n}\left(\sigma_{p(A_p)}(R)\right) = \sigma_{p(A_p)}\left(\pi_{A_1,\ldots,A_n,A_p}(R)\right)$$

Xử lý truy vấn trong môi trường phân tán

❖ ***Phép chọn giao hoán với những phép toán hai ngôi***

- Phép chọn với phép nhân:

$$\sigma_{p(A_p)}(R \times S) \Leftrightarrow \sigma_{p(A_p)}(R) \times S$$

- Phép chọn với phép nối:

$$\sigma_{p(A_i)}(R \bowtie_{(A_i,B_k)} S) \Leftrightarrow \sigma_{p(A_i)}(R) \bowtie_{(A_i,B_k)} S$$

- Phép chọn với phép hợp: Nếu R và T cùng bộ thuộc tính.

$$\sigma_{p(A_i)}(R \cup T) \Leftrightarrow \sigma_{p(A_i)}(R) \cup \sigma_{p(A_i)}(T)$$

Xử lý truy vấn trong môi trường phân tán

❖ ***Phép chiếu giao hoán với những phép toán hai ngôi***

- Phép chiếu và tích Decartes:

Nếu $C = A' \cup B'$ với $A' \subseteq A$, $B' \subseteq B$, và A, B là tập các thuộc tính trên quan hệ R, S ta có:

$$\boldsymbol{\pi_C(R \times S) \Leftrightarrow \pi_{A'}(R) \times \pi_{B'}(S)}$$

- Phép chiếu và phép nối:

$$\boldsymbol{\pi_C\left(R \bowtie_{(A_i,B_j)} S\right) \Leftrightarrow \pi_{A'}(R) \bowtie_{(A_i,B_j)} \pi_{B'}(S)}$$

- Phép chiếu và phép hợp:

$$\boldsymbol{\pi_C(R \cup S) \Leftrightarrow \pi_C(R) \cup \pi_C(S)}$$

Xử lý truy vấn trong môi trường phân tán

o6 luật biến đổi phép toán đại số quan hệ:

- ***Chú ý:***
  - ✓ Việc sử dụng o6 luật trên có khả năng **sinh ra nhiều cây đại số quan hệ tương đương nhau**. Vấn đề là **xác định** cho được **cây tối ưu**.
  - ✓ Trong giai đoạn tối ưu, sự so sánh các cây có thể thực hiện dựa trên **chi phí** dự đoán của chúng. Tuy nhiên, nếu số lượng các cây quá lớn thì cách tiếp cận này sẽ không hiệu quả. Chúng ta có thể dùng o6 luật trên để cấu trúc lại cây, nhằm **loại bỏ** những **cây đại số quan hệ** "**tồi**".

Xử lý truy vấn trong môi trường phân tán

o6 luật biến đổi phép toán đại số quan hệ:

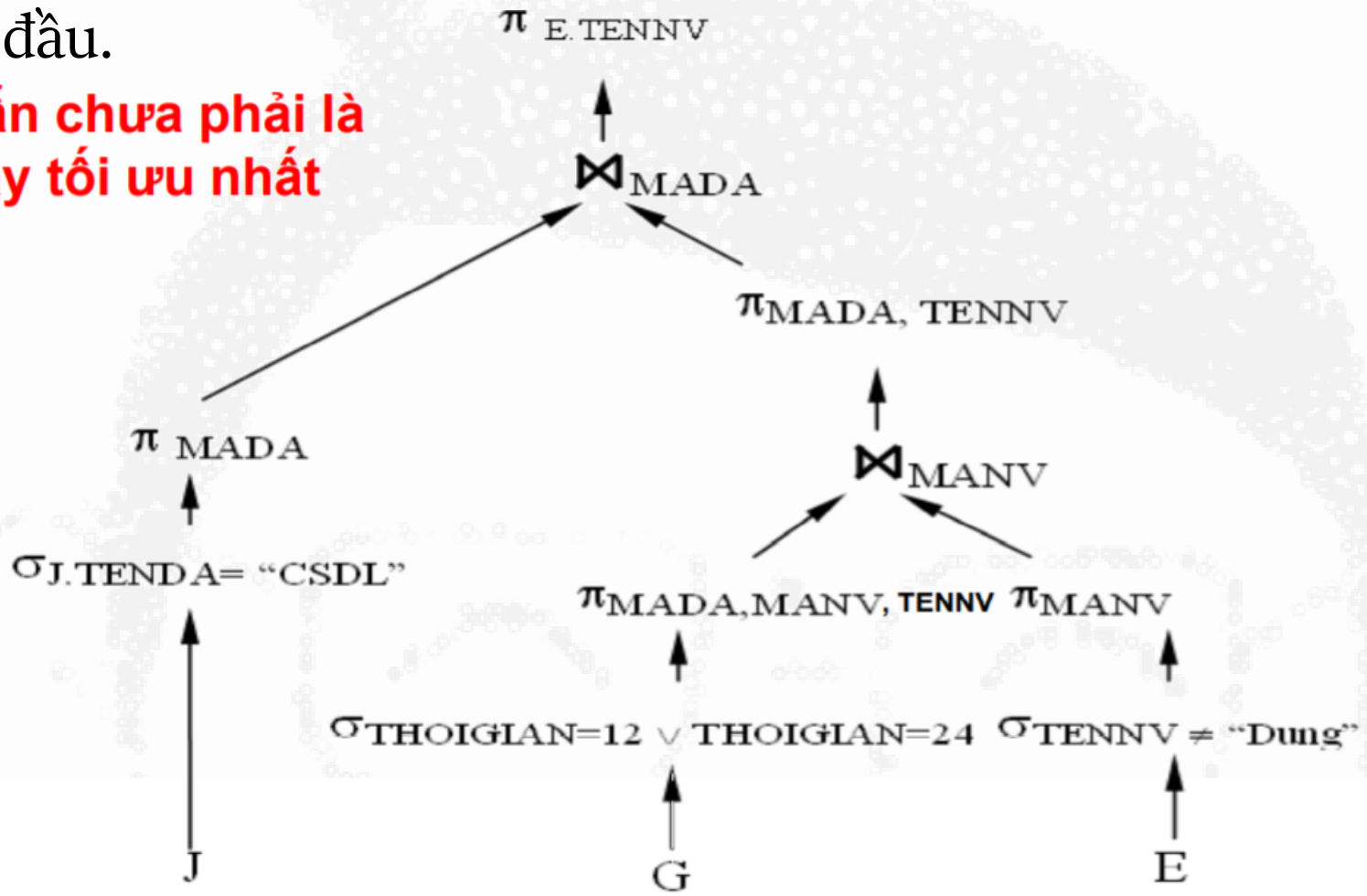
Các luật trên có thể **sử dụng theo bốn cách** như sau:

- ✓ **Phân rã** các phép toán một ngôi, **đơn giản hóa** biểu thức truy vấn .
- ✓ **Nhóm** các phép toán một ngôi trên cùng một quan hệ để giảm số lần thực hiện.
- ✓ **Giao hoán** các phép toán một ngôi với các phép toán hai ngôi để ưu tiên cho một số phép toán (chẳng hạn phép chọn).
- ✓ **Sắp thứ tự** các phép toán hai ngôi trong thực hiện truy vấn.

Xử lý truy vấn trong môi trường phân tán

Ví dụ: Cấu trúc lại cây truy vấn ở ví dụ trên, cho ra cây kết quả tốt hơn cây ban đầu.

Vẫn chưa phải là cây tối ưu nhất



Xử lý truy vấn trong môi trường phân tán

- **Định vị dữ liệu phân tán - Tối ưu hóa cục bộ**
  - Lớp định vị biến đổi một truy vấn đại số quan hệ tổng thể thành một truy vấn đại số được biểu thị trên các mảnh vật lý.
  - Sử dụng thông tin được lưu trữ trên các lược đồ phân mảnh để định vị.
  - Chương trình đại số quan hệ xây dựng lại quan hệ tổng thể từ các phân mảnh của nó gọi là **chương trình định vị**.
  - Truy vấn có được từ chương trình định vị gọi là truy vấn ban đầu.
  - ***Chú ý:*** Trong phần dưới đây, với **mỗi kiểu phân mảnh** chúng ta sẽ biểu diễn **một kỹ thuật rút gọn** để sinh ra truy vấn được tối ưu và đơn giản hoá.

Xử lý truy vấn trong môi trường phân tán

- **Tối ưu hóa truy vấn trên lược đồ phân mảnh bao gồm 2 bước sau:**
  - **Bước 1: Biến đổi** biểu thức đại số quan hệ trên lược đồ toàn cục (chứa các quan hệ toàn cục) **thành biểu thức đại số quan hệ trên lược đồ phân mảnh** (chứa các mảnh của quan hệ toàn cục) bằng cách **thay thế** các quan hệ toàn cục **bởi biểu thức tái lập** của chúng.
  - **Bước 2: Đơn giản hoá** (Rút gọn) biểu thức đại số quan hệ trên lược đồ phân mảnh để có được một **biểu thức hiệu quả** (loại bỏ các phép toán không cần thiết giảm vùng nhớ trung gian) bằng cách sử dụng các **phép biến đổi tương đương** của đại số quan hệ và các đại số quan hệ được tuyển chọn.

Xử lý truy vấn trong môi trường phân tán

- **Đơn giản hoá một biểu thức đại số quan hệ trên lược đồ phân mảnh được thực hiện dựa trên các luật sau:**
  - **Di chuyển các phép chọn xuống các nút lá của cây**, và sau đó áp dụng chúng bằng cách dùng đại số quan hệ được tuyển chọn; thay thế các kết quả chọn lựa bởi quan hệ rỗng nếu điều kiện chọn của kết quả bị mâu thuẫn.
  - Để phân phối các phép kết xuất hiện trong một truy vấn toàn cục, các **phép hợp** (biểu diễn tập hợp của các phân mảnh) phải được **di chuyển lên phía trên các phép kết** mà chúng ta muốn phân phối để loại bỏ các phép kết không cần thiết.
  - Dùng đại số quan hệ được tuyển chọn để **định vị điều kiện chọn** của các toán hạng của các phép kết; **thay thế** cây con, bao gồm phép kết và các toán hạng của nó **bằng quan hệ rỗng** nếu **điều kiện chọn** của kết quả của phép kết bị **mâu thuẫn**.

Xử lý truy vấn trong môi trường phân tán

✓ Rút gọn theo phân mảnh ngang nguyên thuỷ

Xét quan hệ **E (MANV,TENNV,CHUCVU)**.

Tách quan hệ này thành ba mảnh ngang E_1, E_2 và E_3 như sau:

$E_1 = \sigma_{MANV \leq "e3"}(E)$, $E_2 = \sigma_{"e3"<MANV \leq "e6"}(E)$, $E_3 = \sigma_{MANV > "e6"}(E)$

♪ **Chương trình định vị** cho quan hệ E: $\boldsymbol{E = E_1 \cup E_2 \cup E_3}$.

♪ **Dạng ban đầu** của bất kỳ truy vấn nào được xác định trên E là có được bằng cách **thay thế** nó bởi $\boldsymbol{E_1 \cup E_2 \cup E_3}$.

♪ Việc rút gọn các truy vấn trên các quan hệ đã được phân mảnh ngang bao gồm việc **xác định câu truy vấn**, sau khi đã **cấu trúc lại cây con**. Điều này sẽ sinh ra một số **quan hệ rỗng**, và sẽ **loại bỏ** chúng.

♪ Phân mảnh ngang có thể được khai thác để làm **đơn giản cả phép chọn và phép nối**.

Xử lý truy vấn trong môi trường phân tán

- **Rút gọn với phép chọn:** cho một quan hệ R được phân mảnh ngang thành R_1 , R_2 ,..., R_n với $R_j = \sigma_{p_j}(R)$

  Qui tắc 1: $\sigma_{p_i}(R_j) = \emptyset$ nếu $\forall x \in R: \neg(p_i(x) \wedge p_j(x))$.

  Trong đó:

  p_i, p_j là vị từ chọn, x là bộ dữ liệu, p(x) là vị từ p chiếm giữ x.

  ♪ **Di chuyển** các **phép chọn xuống các nút lá** của cây và sau đó áp dụng chúng bằng cách dùng đại số quan hệ được tuyển chọn;

  ♪ **Thay thế các kết quả chọn lựa** bởi quan hệ rỗng nếu **điều kiện chọn** của kết quả bị **mâu thuẫn**.

Xử lý truy vấn trong môi trường phân tán

- **Rút gọn với phép chọn:**

Ví dụ: Hãy rút gọn truy vấn

SELECT *

FROM E

WHERE MANV=”e5”

Với E được tách thành ba mảnh ngang E1, E2 và E3:

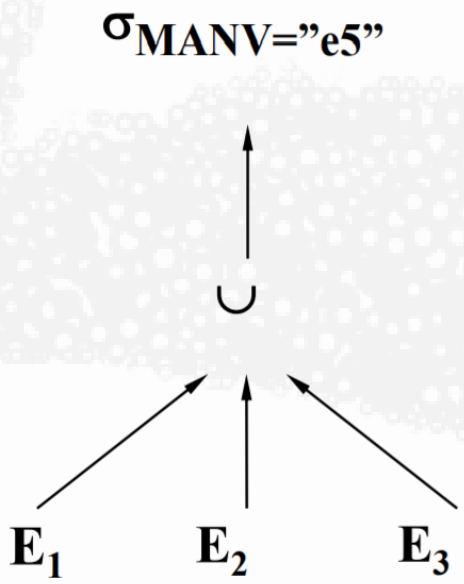
$$E_1 = \sigma_{MANV \leq "e3"}(E)$$

$$E_2 = \sigma_{"e3"<MANV \leq "e6"}(E)$$

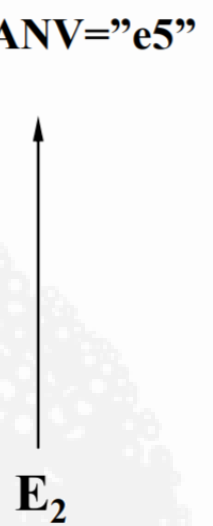
$$E_3 = \sigma_{MANV > "e6"}(E)$$

Xử lý truy vấn trong môi trường phân tán

$$E_1 = \sigma_{MANV \leq "e3"}(E)$$
$$E_2 = \sigma_{"e3"<MANV \leq "e6"}(E)$$
$$E_3 = \sigma_{MANV > "e6"}(E)$$



(a) Truy vấn ban đầu



(b) Truy vấn rút gọn

Rút gọn bằng cách sử dụng **tính chất giao hoán phép chọn với phép hợp**, chúng ta thấy **vị từ chọn đối lập** với **vị từ E1 và E3** nên sinh ra các **quan hệ rỗng**.

Xử lý truy vấn trong môi trường phân tán

Rút gọn với phép nối:

- Các **phép nối** trên quan hệ đã được phân mảnh ngang **có thể đơn giản** khi chúng **được phân mảnh theo thuộc tính nối**.
- Việc **rút gọn** được thực hiện dựa trên **tính phân phối giữa phép nối và phép hợp** và **loại bỏ các phép nối vô ích**.
- Với tính chất, $(R_1 \cup R_2) \bowtie R_3 = (R_1 \bowtie R_3) \cup (R_2 \bowtie R_3)$, R_i là các phân mảnh. Chúng ta có thể xác định được các phép nối vô ích của các mảnh khi các **điều kiện nối mâu thuẫn nhau**. Sau đó, dùng luật 2 dưới đây để loại bỏ các phép nối vô ích.

Xử lý truy vấn trong môi trường phân tán

❁ Rút gọn với phép nối:

♪ Qui tắc 2:

$$R_i \bowtie R_j = \emptyset \text{ nếu } \forall x \in R_i, \forall y \in R_j: \neg(p_i(x) \wedge p_j(y))$$

Trong đó Ri, Rj được xác định theo các vị từ pi, pj trên cùng thuộc tính.

Để phân phối các phép kết xuất hiện trong một truy vấn toàn cục, các **phép hợp** (biểu diễn tập hợp của các phân mảnh) phải được **di chuyển lên phía trên các phép kết** mà chúng ta muốn phân phối để **loại bỏ các phép kết không cần thiết**.

Xử lý truy vấn trong môi trường phân tán

❁ Rút gọn với phép nối:

♪ Nhận xét:

- ✓ Việc xác định các phép nối vô ích được thực hiện bằng cách **chỉ xem xét các vị từ mảnh**.
- ✓ Truy vấn rút gọn **không** phải **luôn tốt hơn** hoặc **đơn giản hơn** truy vấn ban đầu.
- ✓ Một **thuận lợi** của truy vấn rút gọn là những **phép nối** có thể **thực hiện song song**.

Xử lý truy vấn trong môi trường phân tán

♪ **Ví dụ:** Giả sử quan hệ E được phân mảnh thành các mảnh:

$$E_1 = \sigma_{MANV \leq "e3"}(E),\ E_2 = \sigma_{"e3" < MANV \leq "e6"}(E),\ E_3 = \sigma_{MANV > "e6"}(E)$$

Quan hệ G được phân làm hai mảnh:

$$G_1 = \sigma_{MANV \leq "e3"}(G) \text{ và } G_2 = \sigma_{MANV > "e3"}(G)$$

♪ **Nhận xét:**

- ✓ E_1 và G_1 được định nghĩa bởi cùng vị từ.
- ✓ Vị từ định nghĩa G_2 là hợp của các định nghĩa của những vị từ E_2 và E_3.

♪ **Xét truy vấn:**

```
SELECT *
FROM E, G
WHERE E.MANV=G.MANV
```

Xử lý truy vấn trong môi trường phân tán

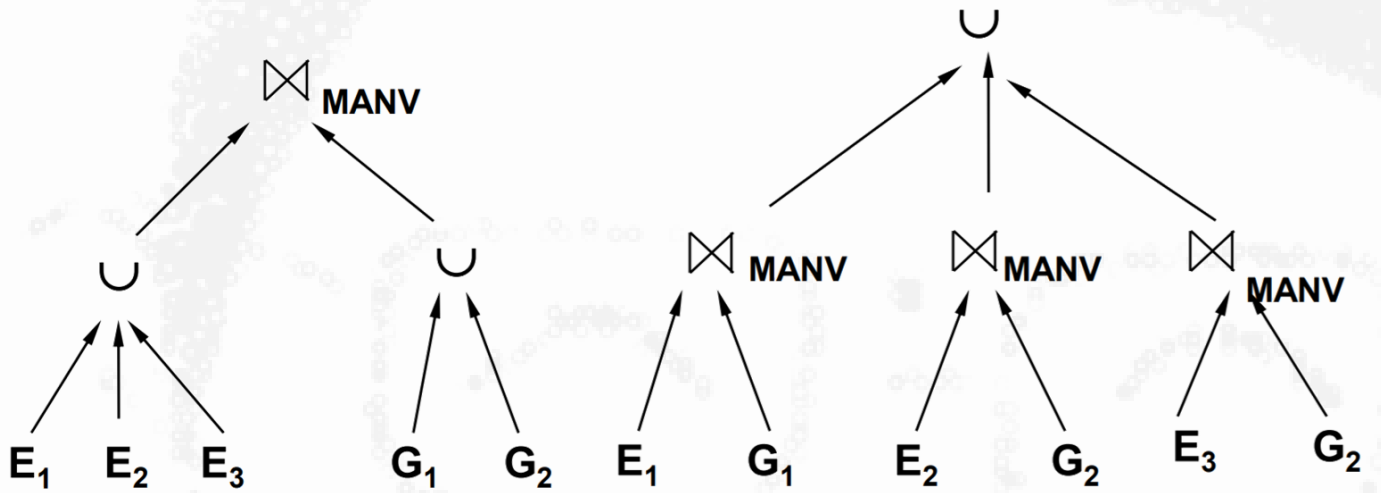
$E_1 = \sigma_{MANV \leq "e3"}(E), E_2 = \sigma_{"e3" < MANV \leq "e6"}(E), E_3 = \sigma_{MANV > "e6"}(E)$

$G_1 = \sigma_{MANV \leq "e3"}(G)$ và $G_2 = \sigma_{MANV > "e3"}(G)$

$E \bowtie G = (E_1 \cup E_2 \cup E_3) \bowtie (G_1 \cup G_2)$

$= (E_1 \bowtie G_1) \cup (E_1 \bowtie G_2) \cup (E_2 \bowtie G_1) \cup (E_2 \bowtie G_2) \cup (E_3 \bowtie G_1) \cup (E_3 \bowtie G_2)$

$= (E_1 \bowtie G_1) \cup \quad (E_2 \bowtie G_2) \cup \quad (E_3 \bowtie G_2)$



(a) Truy vấn ban đầu

(b) Truy vấn rút gọn

Xử lý truy vấn trong môi trường phân tán

✓ Rút gọn phân mảnh dọc

- Chức năng của việc phân mảnh dọc là tách quan hệ dựa vào thuộc tính của các phép chiếu.
- Vì phép toán xây dựng lại đối với phân mảnh dọc là nối, nên chương trình định vị một quan hệ đã được phân mảnh dọc là nối của các mảnh trong vùng thuộc tính chung.

♪ **Ví dụ:** Quan hệ E được phân mảnh dọc thành E1 , E2 , với thuộc tính khoá MANV được lặp lại như sau: $E_1 = \pi_{MANV, TENNV}(E)$ và $E_2 = \pi_{MANV, CHUCVU}(E)$

Chương trình định vị là: $E = E_1 \bowtie_{MANV} E_2$

- Các truy vấn trên phân mảnh dọc có thể rút gọn bằng cách **xác định các quan hệ trung gian vô ích** và **loại bỏ các cây con chứa chúng**.
- Các phép chiếu trên một phân mảnh dọc **không có thuộc tính chung với các thuộc tính chiếu** (ngoại trừ khóa của quan hệ) là **vô ích**, mặc dù các quan hệ là khác rỗng.

Xử lý truy vấn trong môi trường phân tán

❁ Rút gọn phân mảnh dọc:

♪ Qui tắc 3:

$$\pi_{D,K}(R_i) \text{ là vô ích nếu } D \cap A' = \emptyset$$

Trong đó, quan hệ R xác định trên A={A_1, ..., A_n }; $R_i = \pi_{A'}(R)$, $A' \subseteq A$, K là khoá của quan hệ, $K \subset A$, D là tập các thuộc tính chiếu, $D \subset A$.

- Dùng đại số quan hệ được tuyển chọn để **định vị điều kiện chọn** của các toán hạng của các phép kết;
- Thay thế cây con, bao gồm phép kết và các toán hạng của nó bằng quan hệ **rỗng** nếu **điều kiện chọn** của kết quả của phép kết bị **mâu thuẫn**.

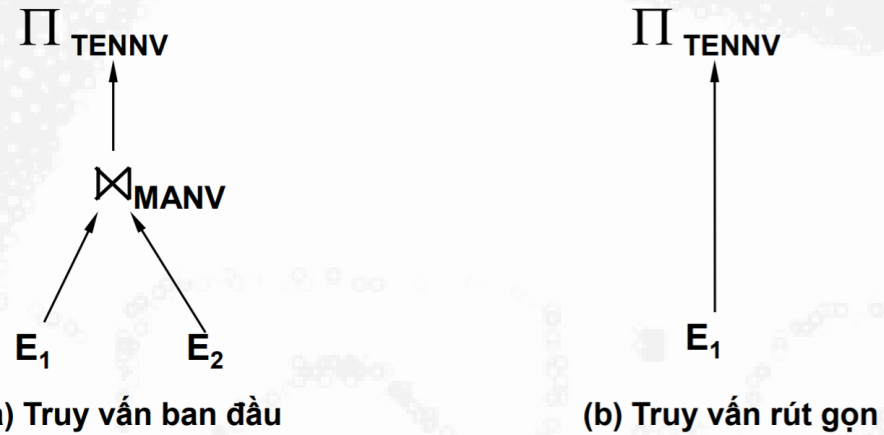
Xử lý truy vấn trong môi trường phân tán

♪ **Ví dụ:** Với quan hệ E được phân mảnh dọc như sau:

$$E_1 = \pi_{MANV,\,TENNV}(E) \text{ và } E_2 = \pi_{MANV,CHUCVU}(E)$$

Xét truy vấn SQL: **SELECT** TENNV **FROM** E

$$G_1 = \sigma_{MANV \leq "e3"}(G) \text{ và } G_2 = \sigma_{MANV > "e3"}(G)$$



♪ **Nhận xét:** phép chiếu trên E_2 là vô ích vì TENNV không có trong E_2, nên phép chiếu chỉ cần gán vào E_1.

Xử lý truy vấn trong môi trường phân tán

✓ Rút gọn theo phân mảnh ngang dẫn xuất

- Sự phân mảnh ngang dẫn xuất là một cách tách hai quan hệ để việc xử lý nối của các phép chọn và phép nối
- Nếu quan hệ R phụ thuộc vào sự phân mảnh ngang dẫn xuất nhờ quan hệ S, thì các mảnh của R và S, mà có cùng giá trị thuộc tính nối sẽ được định vị tại cùng trạm. Ngoài ra, S có thể được phân mảnh tùy thuộc vào vị từ chọn.
- Khi các bộ của R được đặt tuỳ theo những bộ của S, thì sự phân mảnh dẫn xuất chỉ nên sử dụng mối quan hệ một nhiều từ S → R (i.e. với một bộ của S có thể phù hợp với n bộ của R, nhưng với một bộ của R chỉ phù hợp với một bộ của S).
- Truy vấn trên các phân mảnh dẫn xuất cũng có thể rút gọn được, nếu các vị từ phân mảnh mâu thuẫn nhau thì phép nối sẽ đưa ra quan hệ rỗng.
- Chương trình định vị một quan hệ đã được phân mảnh ngang dẫn xuất là **hợp** của các mảnh.

Xử lý truy vấn trong môi trường phân tán

♪ **Ví dụ:** Cho mối quan hệ một nhiều từ E đến G, quan hệ G (MANV, MADA, NHIEMVU, THOIGIAN) được phân mảnh ngang dẫn xuất theo E (MANV, TENNV, CHUCVU) như sau:

$$G_1 = G \ltimes_{MANV} E_1 \text{ và } G_2 = G \ltimes_{MANV} E_2$$

Trong đó E được phân mảnh ngang như sau:

$$E_1 = \sigma_{CHUCVU="Lập\ trình"}(E) \text{ và } E_2 = \sigma_{CHUCVU\neq"Lập\ trình"}(E)$$

Chương trình định vị cho một quan hệ đã được phân mảnh ngang dẫn xuất là hợp của các mảnh $G = G_1 \cup G_2$.

Để rút gọn các truy vấn trên phân mảnh gián tiếp này, phép nối sẽ đưa ra quan hệ rỗng nếu các vị từ phân mảnh mâu thuẫn nhau.

Ví dụ vị từ G_1 và E_2 mâu thuẫn nhau, nên $G_1 \bowtie E_2 = \emptyset$

Xử lý truy vấn trong môi trường phân tán

$G_1 = G \ltimes_{MANV} E_1$ và $G_2 = G \ltimes_{MANV} E_2$

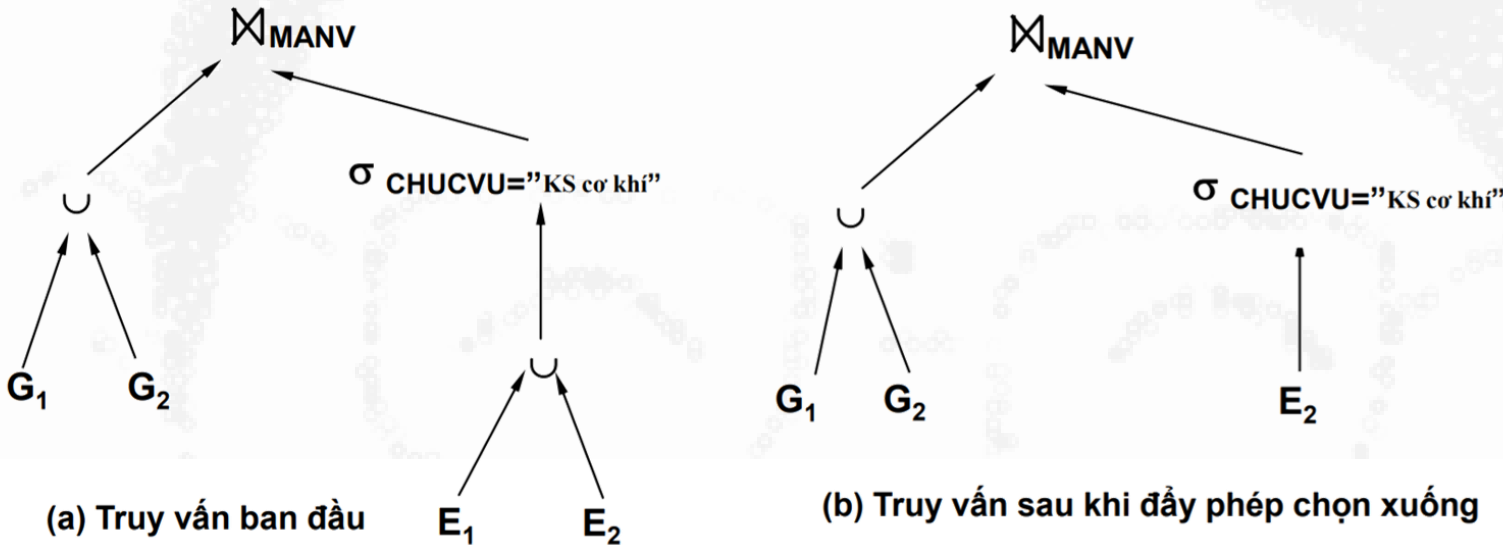
$E_1 = \sigma_{CHUCVU="Lập\ trình"}(E)$ và $E_2 = \sigma_{CHUCVU\neq"Lập\ trình"}(E)$

♪ **Ví dụ:** Xét truy vấn **SELECT** *

FROM E, G

WHERE G.MANV=E.MANV

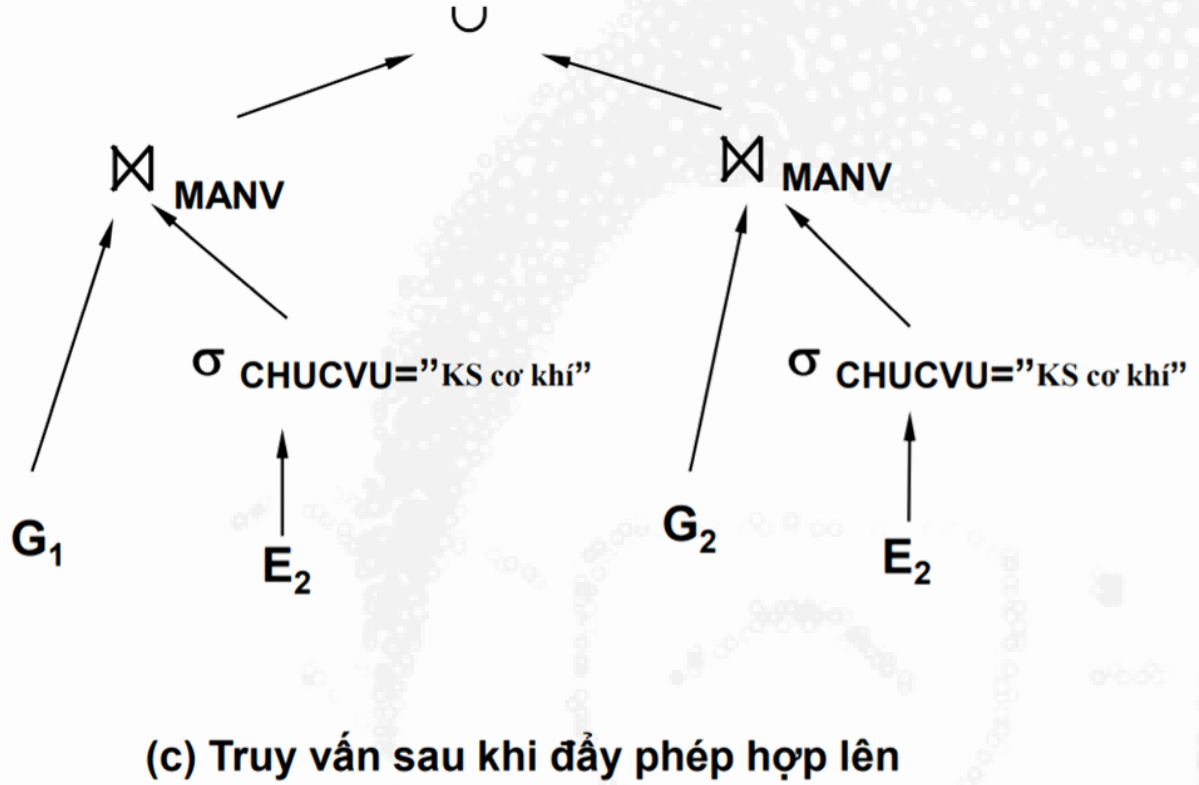
AND CHUCVU=”KS cơ khí”



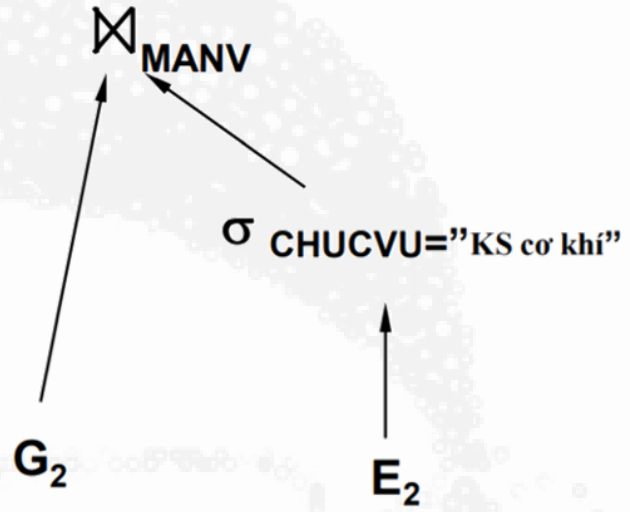
Xử lý truy vấn trong môi trường phân tán

Chú ý: $(G_1 \cup G_2) \bowtie \sigma_{CHUCVU="KS\ Cơ\ khí"}(E_2)$

$= (G_1 \bowtie \sigma_{CHUCVU="KS\ Cơ\ khí"}(E_2)) \cup (G_2 \bowtie \sigma_{CHUCVU="KS\ Cơ\ khí"}(E_2))$



(c) Truy vấn sau khi đẩy phép hợp lên



(d) Truy vấn đã rút gọn

Xử lý truy vấn trong môi trường phân tán

Nhận xét:

- ✓ Truy vấn ban đầu trên các mảnh E_1, E_2, G_1 và G_2 tương ứng hình (a).
- ✓ Bằng cách đẩy phép chọn xuống các mảnh E_1 và E_2, được truy vấn rút gọn ở hình (b).
- ✓ Phân phối các phép nối với phép hợp, chúng ta thu được cây hình (c).
- ✓ Cây con bên trái đưa ra một quan hệ rỗng, nên cây rút gọn có được trong hình (d).

Xử lý truy vấn trong môi trường phân tán

✓ Rút gọn theo phân mảnh hỗn hợp

- Sự phân mảnh hỗn hợp là sự kết hợp giữa phân mảnh dọc và phân mảnh ngang.
- Mục đích của phân mảnh hỗn hợp là hỗ trợ các truy vấn liên quan đến phép chiếu, phép chọn và phép nối
- Chương trình định vị cho một quan hệ đã phân mảnh hỗn hợp là phép hợp và phép nối của các mảnh.

♪ **Ví dụ:** Xét quan hệ E được phân mảnh hỗn hợp như sau:

$E_1 = \sigma_{MANV \leq "e4"}(\pi_{MANV, TENNV}(E))$, $E_2 = \sigma_{MANV > "e4"}(\pi_{MANV, TENNV}(E))$,

$E_3 = \pi_{MANV, CHUCVU}(E)$

Chương trình định vị là: $E = (E_1 \cup E_2) \bowtie_{MANV} E_3$

Xử lý truy vấn trong môi trường phân tán

✓ Rút gọn theo phân mảnh hỗn hợp

- Các truy vấn trên các mảnh hỗn hợp có thể được rút gọn bằng cách kết hợp các qui tắc sử dụng trong phân mảnh ngang nguyên thủy, phân mảnh dọc, phân mảnh ngang gian tiếp, tương ứng như sau:
  - **Loại bỏ các quan hệ rỗng** sinh bởi sự mâu thuẫn giữa các phép chọn trên các phân mảnh ngang.
  - **Loại bỏ các quan hệ vô ích** sinh bởi các phép chiếu trên các phân mảnh dọc.
  - **Phân phối các phép nối với các phép hợp** để tách và loại bỏ các phép nối vô ích.

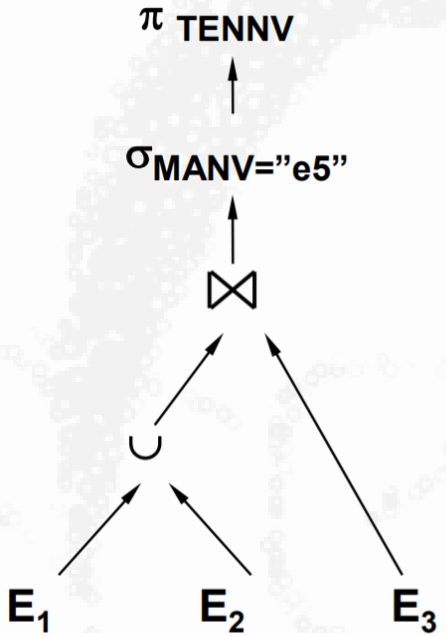
Xử lý truy vấn trong môi trường phân tán

♪ **Ví dụ:** Cho quan hệ E được phân mảnh:

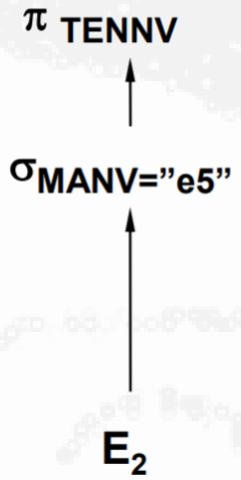
$E_1 = \sigma_{MANV \leq "e4"}(\pi_{MANV, TENNV}(E))$, $E_2 = \sigma_{MANV > "e4"}(\pi_{MANV, TENNV}(E))$,

$E_3 = \pi_{MANV, CHUCVU}(E)$

Câu truy vấn: **SELECT** TENNV **FROM** E **WHERE** MANV=“e5”



(a) Truy vấn ban đầu



(b) Truy vấn đã rút gọn

Biến đổi truy vấn toàn cục thành truy vấn đoạn

✓Kỹ thuật:

- Biểu diễn truy vấn toàn cục dưới dạng cây.
- Thay thế mối quan hệ bằng biểu thức xây dựng lại:
  - ❖ Phân đoạn ngang: $R = R_1 \cup R_2 \cup \cdots \cup R_n$
  - ❖ Phân đoạn dọc: $R = R_1 * R_2 * \cdots * R_n$
- Biến đổi truy vấn: Sử dụng các phép biến đổi tương đương.
- Bỏ các nhánh của cây trong đó điều kiện truy vấn và điều kiện phân đoạn mâu thuẫn với nhau.

Biến đổi truy vấn toàn cục thành truy vấn đoạn

♪ **Ví dụ:** Xét hai bảng NhanVien, DonVi

- NhanVien (MaNV, HoDem, Ten, Luong, MaDV)
- DonVi (MaDV, TenDV, MaNQL)
- NhanVien được phân mảnh ngang làm 2 theo MaDV = 1 hoặc 2:

$$NV_1 = \sigma_{MaDV=1}(NhanVien)$$
$$NV_2 = \sigma_{MaDV=2}(NhanVien)$$

- Cho truy vấn toàn cục: Tìm họ đệm, tên, tên đơn vị của các nhân viên có MaDV = 1.

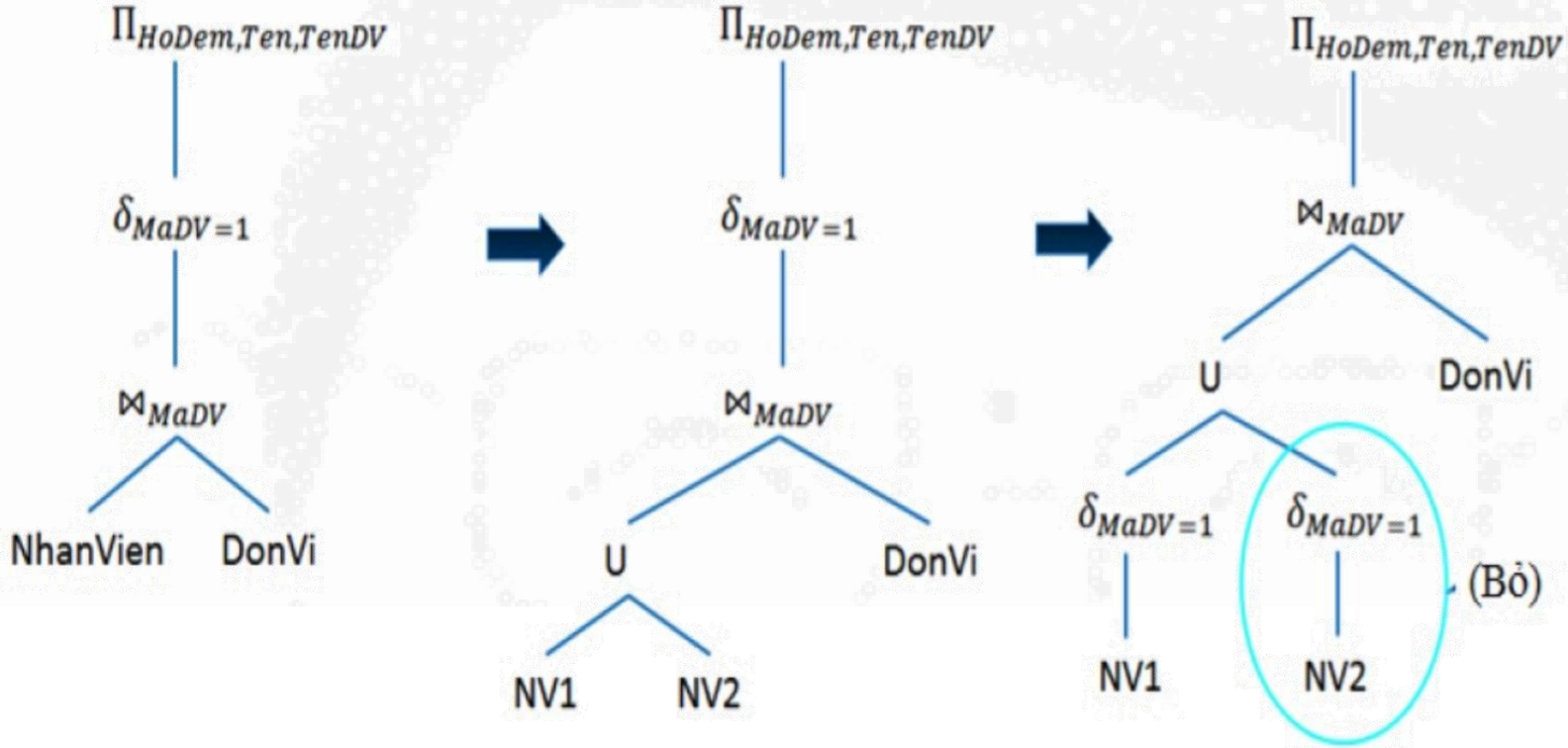
Biến đổi truy vấn toàn cục thành truy vấn đoạn

♪ **Ví dụ (tt):**

- Truy vấn toàn cục Q:

$$\pi_{HoDem,\,Ten,\,TenDV}(\sigma_{MaDV=1}(NhanVien \bowtie_{MaDV} DonVi))$$

- Chuyển Q thành truy vấn đoạn:



Bài tập

Cho các quan hệ toàn cục:

KhoHang (MaKH, TenKH, ViTri)

LoaiHang (MaLH, TenLH, MoTa, MaKH)

Giả sử KhoHang, LoaiHang được phân đoạn ngang theo MaKH = 1 và MaKH = 2 thành KhoHang1, KhoHang2, LoaiHang1, LoaiHang2.

- Viết biểu thức đại số quan hệ của các đoạn trên.
- Viết biểu thức xây dựng lại của 2 quan hệ toàn cục Khohang, LoaiHang.
- Hãy viết truy vấn toàn cục QTC: Lập danh sách gồm TenLH, ViTri của loại hàng có MaLH="LH01" và ở kho hàng có MaKH = 2.
- Chuyển truy vấn QTC ở trên thành truy vấn đoạn.