IS210 – Chương 6

Tối ưu hóa câu truy vấn

Trương Thu Thủy

Nội dung chi tiết

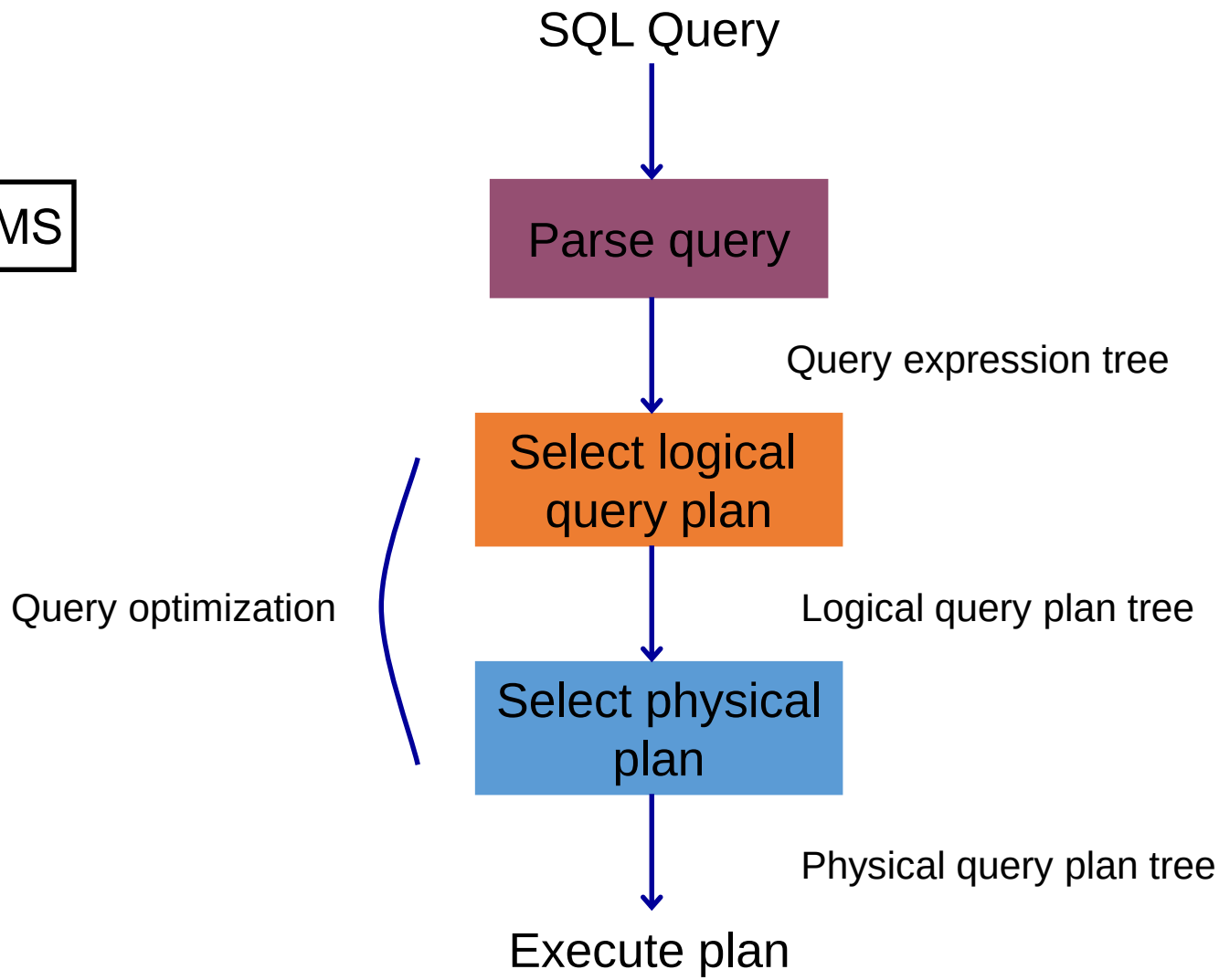
- Giới thiệu
- Phân tích cú pháp
- Chuyển đổi sang đại số quan hệ
- Tối ưu hóa câu truy vấn

Bộ biên dịch câu truy vấn – Query compiler

- Câu lệnh truy vấn (viết bằng ngôn ngữ SQL) được phân tích cú pháp và được biểu diễn dưới dạng cây
- Cây đã được phân tích cú pháp này sẽ được biểu diễn dưới dạng biểu thức đại số quan hệ (hoặc là một dạng tương tự) được gọi là *logical query plan*.
- Logical query plan sẽ được chuyển thành physical query plan: thể hiện cách thức hoạt động, thứ tự, giải thuật sử dụng ở mỗi bước, cách thức lưu trữ dữ liệu khi truy vấn

Bộ biên dịch câu truy vấn – Query compiler

Tập trung vào RDBMS



Phân tích cú pháp và parse tree

- Công việc của bộ phân tích cú pháp là chuyển câu lệnh được viết dưới dạng ngôn ngữ SQL sang cây phân tích cú pháp
- Câu truy vấn
  - <Query> ::= SELECT <SelList> FROM <FromList> WHERE <Condition>
- Select-Lists
  - <SelList> ::= < A ttribute> , <SelList>
  - <SelList> ::= <A ttribute>
- From –Lists
  - <FromList> ::= <Relation> , <FromList>
  - <FromList> ::= <Relation>
- Conditions
  - <Condition> ::= <Condition> AND <Condition>
    <Condition> ::= <A ttribute> IN (<Query>)
    <Condition> ::= <A ttribute> = <A ttribute>
    <Condition> ::= <A ttribute> LIKE <Pattern>

Phân tích cú pháp

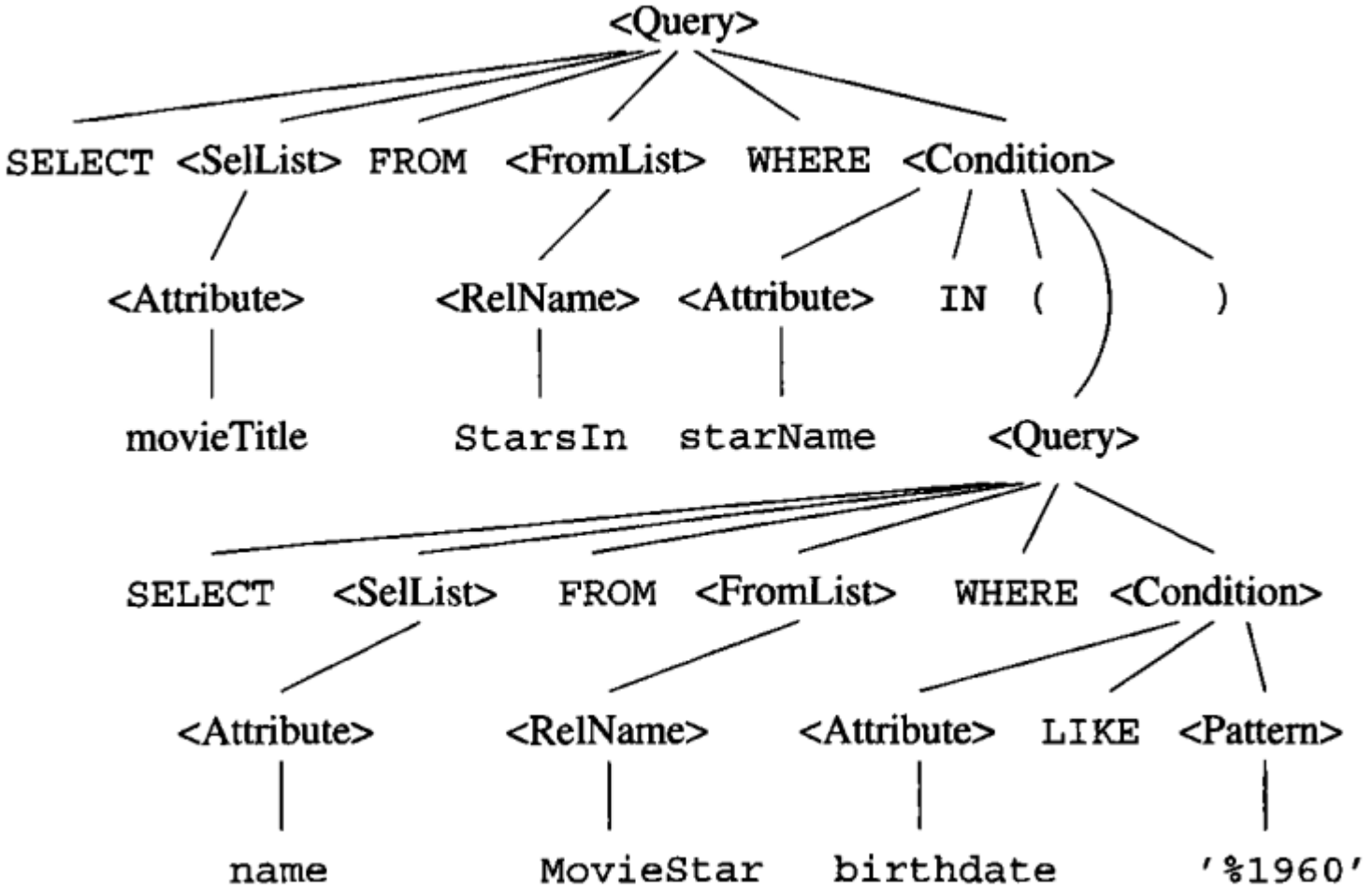
Ví dụ 1

- StarsIn (movieTitle, movieYear, starName)
- MovieStar (name, address, gender, birthdate)
- Và câu truy vấn:

```
SELECT movieTitle
FROM Starsln
WHERE starName IN (
        SELECT name
        FROM MovieStar
        WHERE birth date LIKE '%1960')
```

Phân tích cú pháp

Ví dụ 1



Phân tích cú pháp

Ví dụ 2

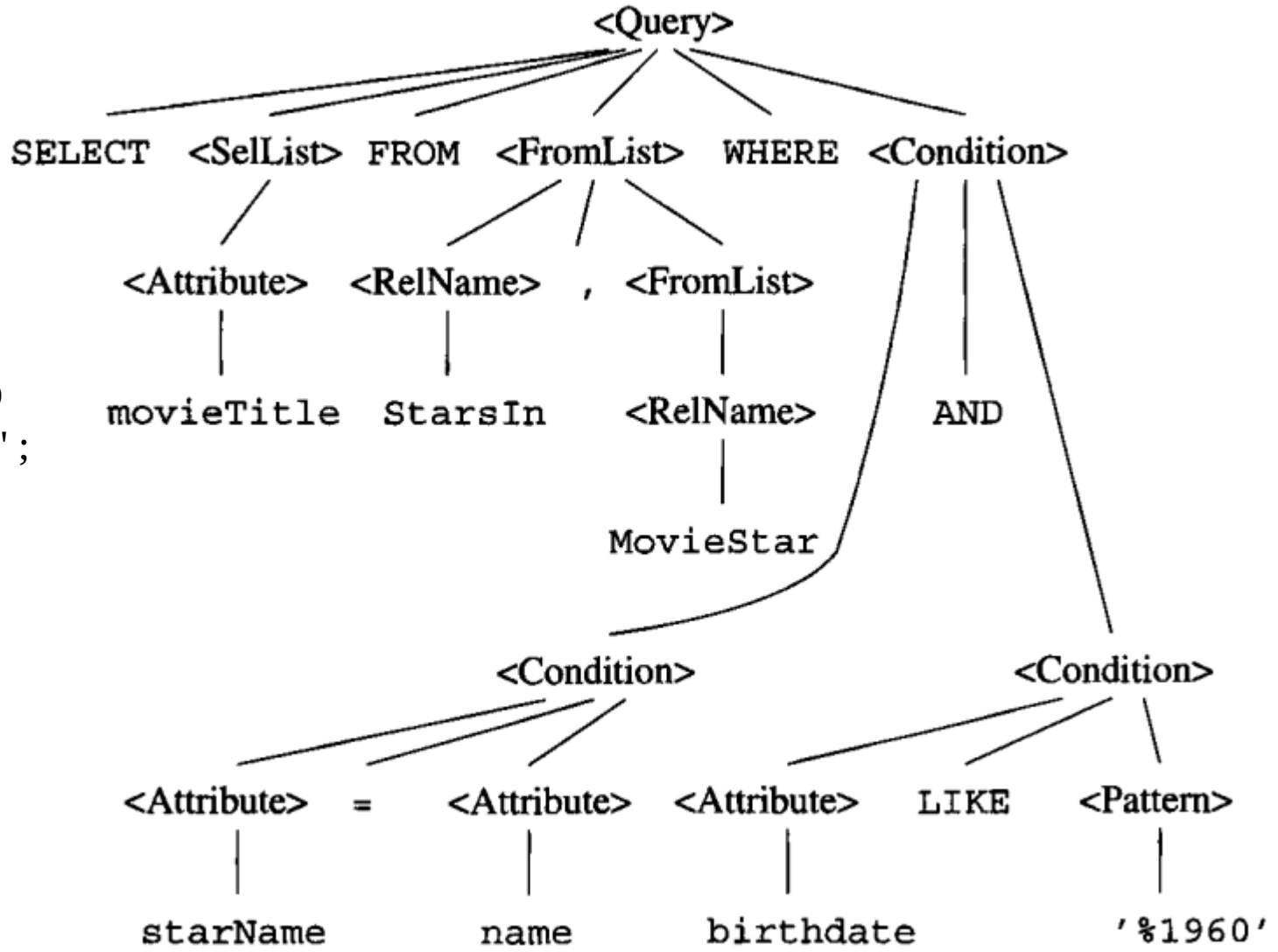
- StarsIn (movieTitle, movieYear, starName)
- MovieStar (name, address, gender, birthdate)
- Và câu truy vấn:

```
SELECT movieTitle
FROM   Starsln , MovieStar
WHERE  starName = name AND
       birthdate LIKE '%1960' ;
```

Phân tích cú pháp

- Ví dụ 2

```
SELECT movieTitle
FROM   Starsln , MovieStar
WHERE  starName = name AND
       birthdate LIKE '%1960' ;
```

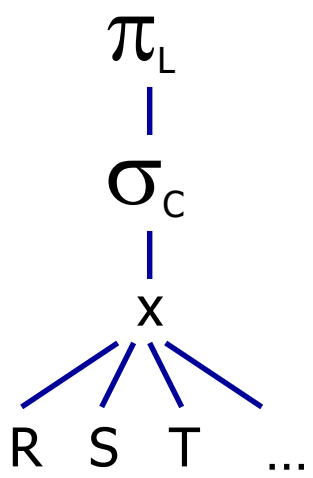


Kiểm tra ngữ nghĩa

- Kiểm tra quan hệ (bảng): mỗi Quan hệ được đề cập ở mệnh đề From phải là một Quan hệ (bảng) hoặc là View.
- Kiểm tra việc sử dụng các thuộc tính: mỗi thuộc tính được đề cập trong mệnh đề SELECT hay WHERE phải là thuộc tính của Quan hệ (bảng) trong phạm vi hiện tại.
- Kiểm tra kiểu dữ liệu: tất cả thuộc tính phải phù hợp khi sử dụng.

Biến đổi sang Đại số quan hệ

- Truy vấn đơn
  - Xét cấu trúc <SFW>
    - Thay thế <FromList> thành các biến quan hệ
      - Sử dụng phép tích cartesian cho các biến quan hệ
    - Thay thế <Condition> thành phép chọn σ_C
    - Thay thế <SelectList> thành phép chiếu π_L



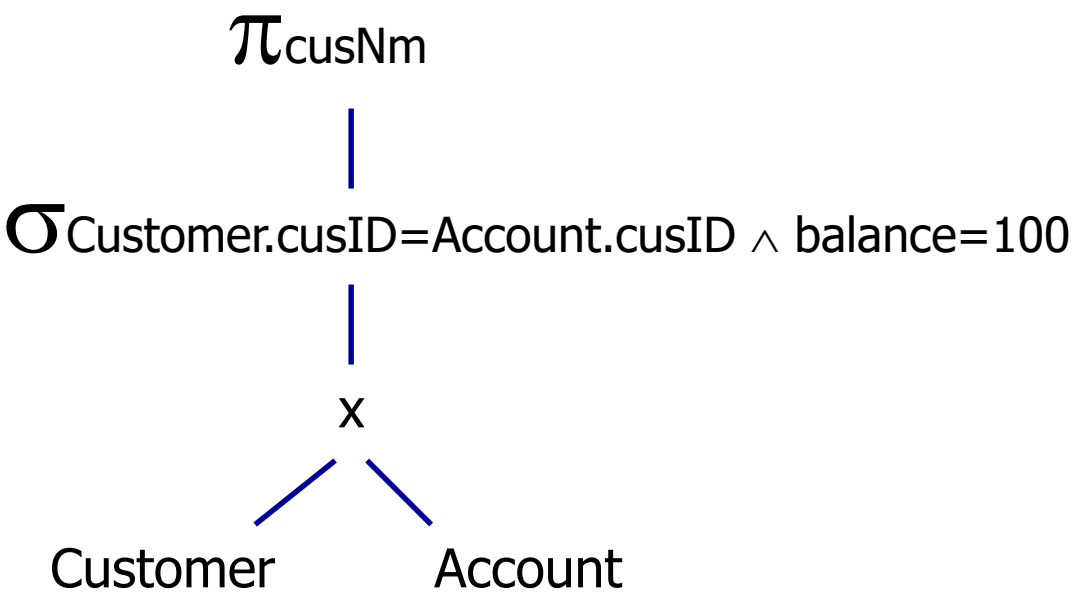
Cây truy vấn

Biến đổi sang Đại số quan hệ

Ví dụ 1

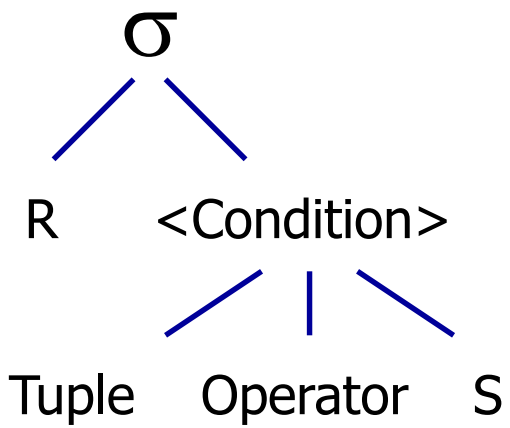
- Customer(cusID, cusNm, cusStreet, cusCity)
- Account(accID, cusID, balance)

```
SELECT  cusNm
FROM    Customer, Account
WHERE   Customer.cusID = Account.cusID
        AND balance = 100
```



Biến đổi sang Đại số quan hệ

- Truy vấn lồng
  - Tồn tại câu truy vấn con S trong <Condition>
  - Áp dụng qui tắc <**SFW**> cho truy vấn con **S**
  - Sử dụng *phép chọn 2 biến* (two-argument selection)
    - Nút là phép chọn không có tham số
    - Nhánh con trái là biến quan hệ R
    - Nhánh con phải là <condition> áp dụng cho **mỗi bộ trong R**

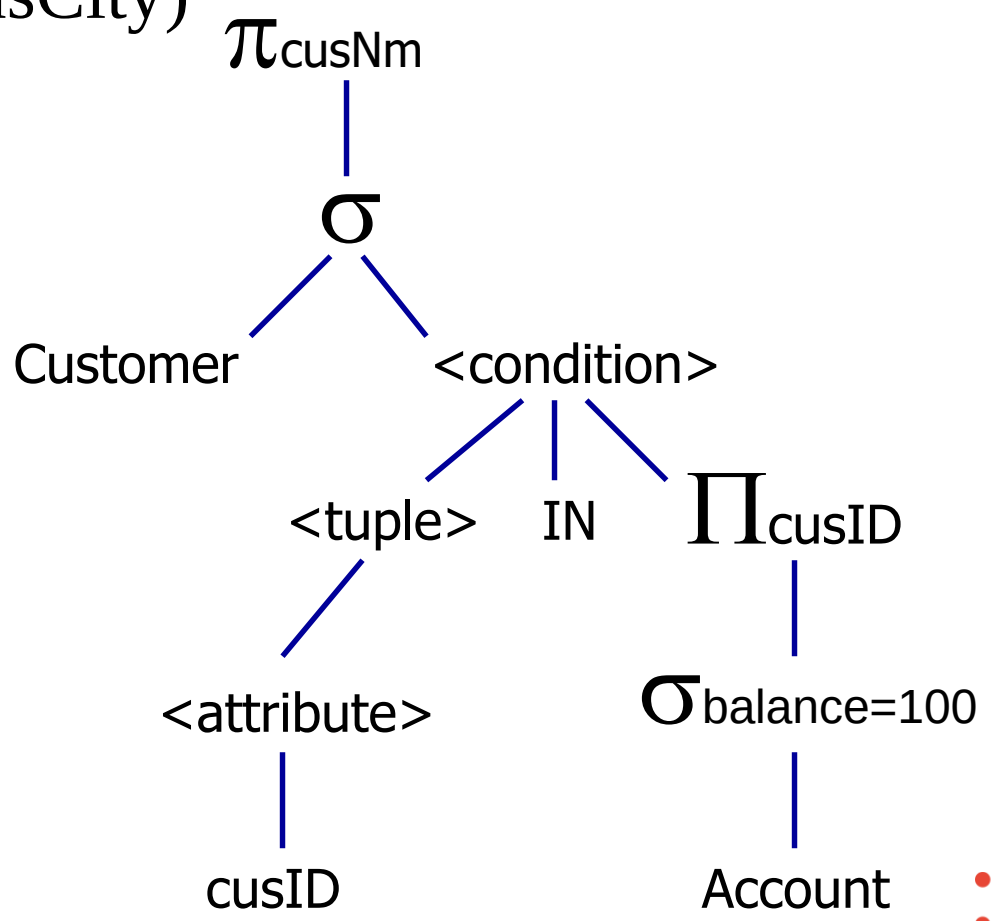


Biến đổi sang Đại số quan hệ

Ví dụ 2

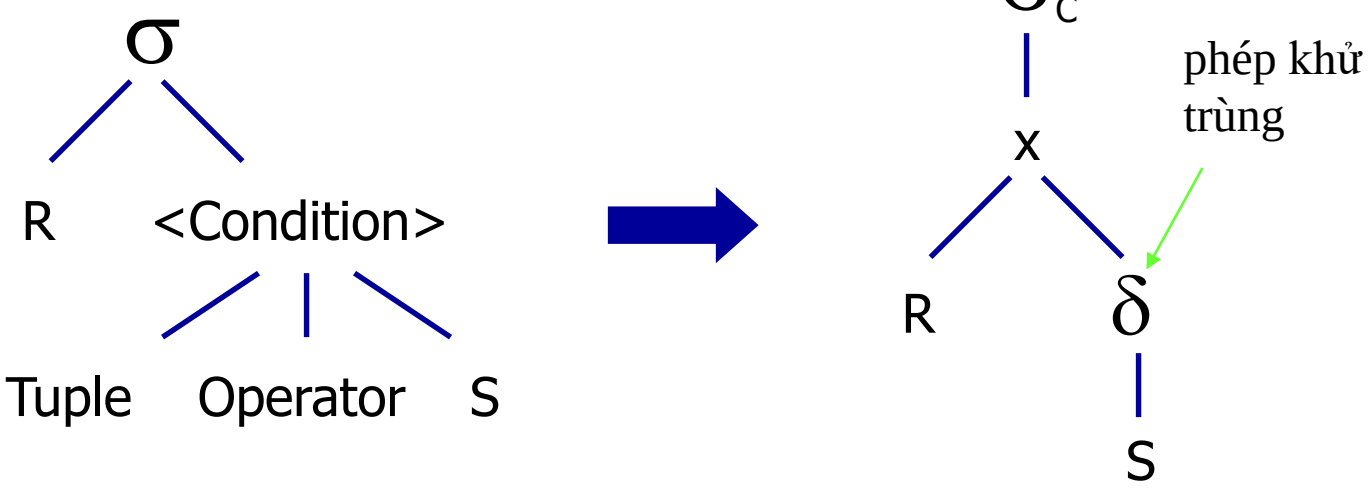
- Customer(cusID, cusNm, cusStreet, cusCity)
- Account(accID, cusID, balance)

```
SELECT cusNm
FROM   Customer
WHERE  cusID IN (
                  SELECT cusID
                  FROM   Account
                  WHERE  balance > 100)
```

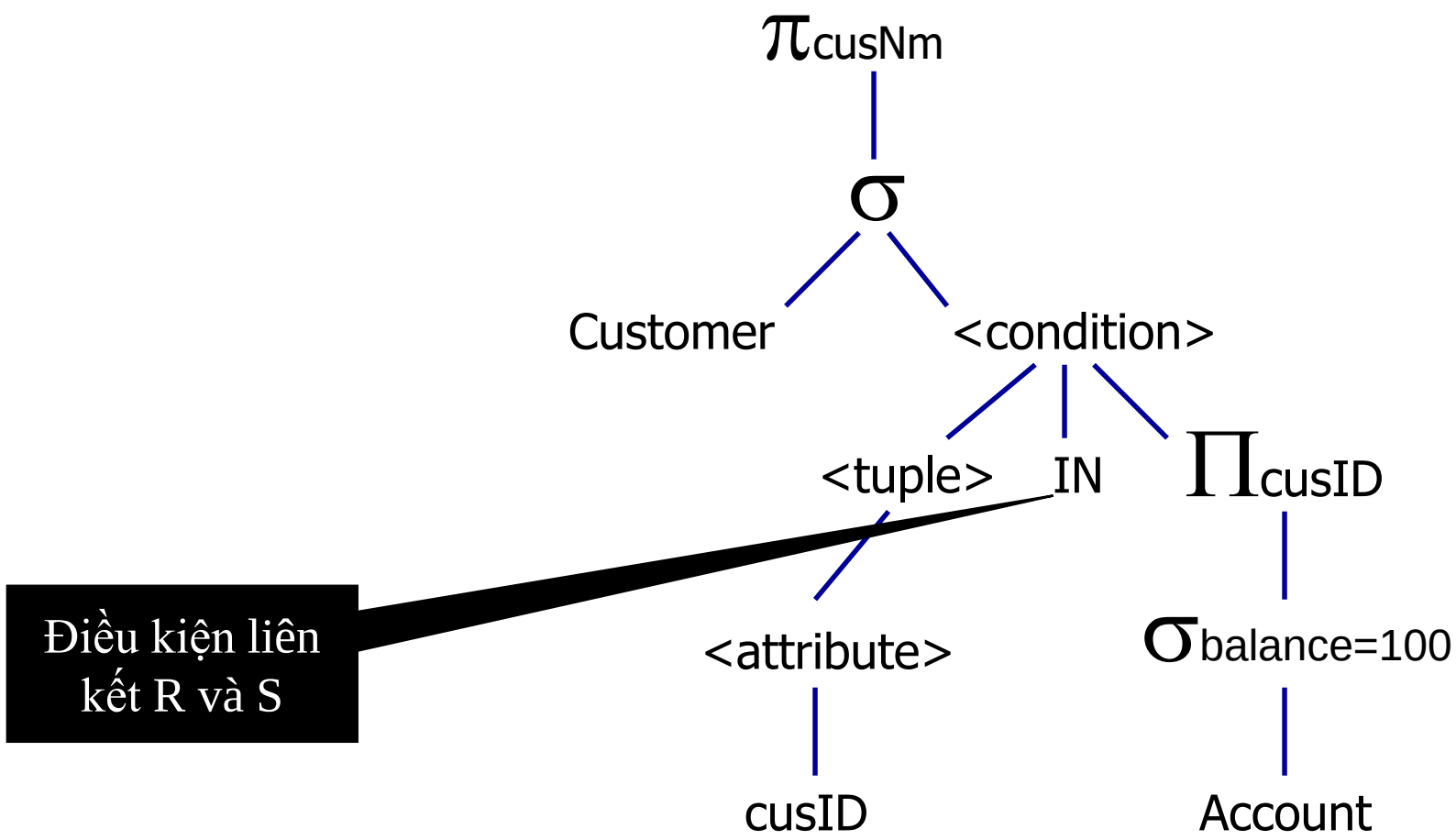


Biến đổi sang Đại số quan hệ

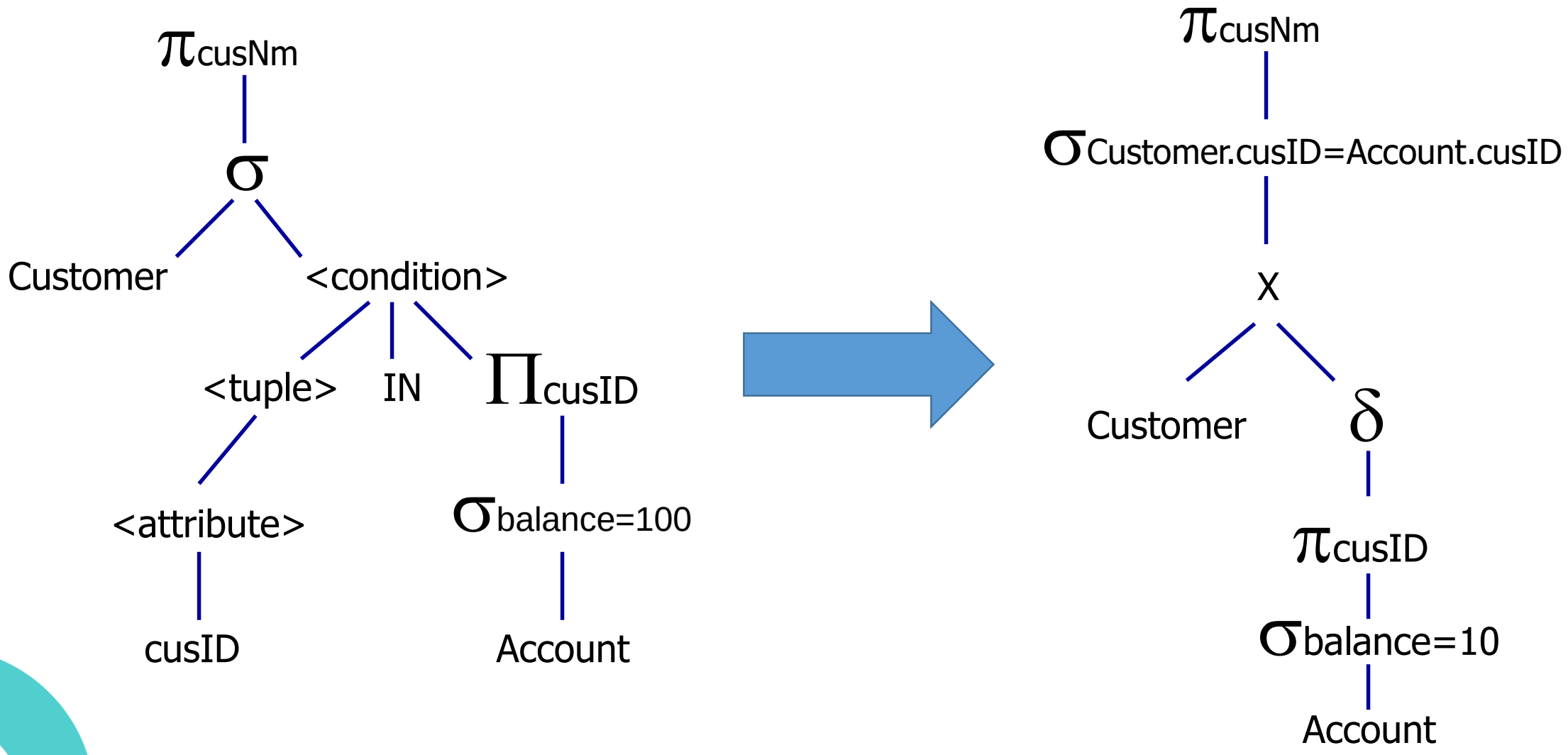
- Truy vấn lồng
  - Biến đổi phép chọn 2 biến
    - Thay thế <Condition> bằng 1 cây có gốc là S
      - Nếu S có các bộ trùng nhau thì phải lược bỏ bớt bộ trùng nhau đi
      - Sử dụng phép δ (loại bỏ dữ liệu trùng – giống distinct)
    - Thay thế phép chọn 2 biến thành σ_C - C là điều kiện liên kết (không đơn thuần là kết) R với S
    - σ_C làm trên kết quả của phép cartesian của R và S



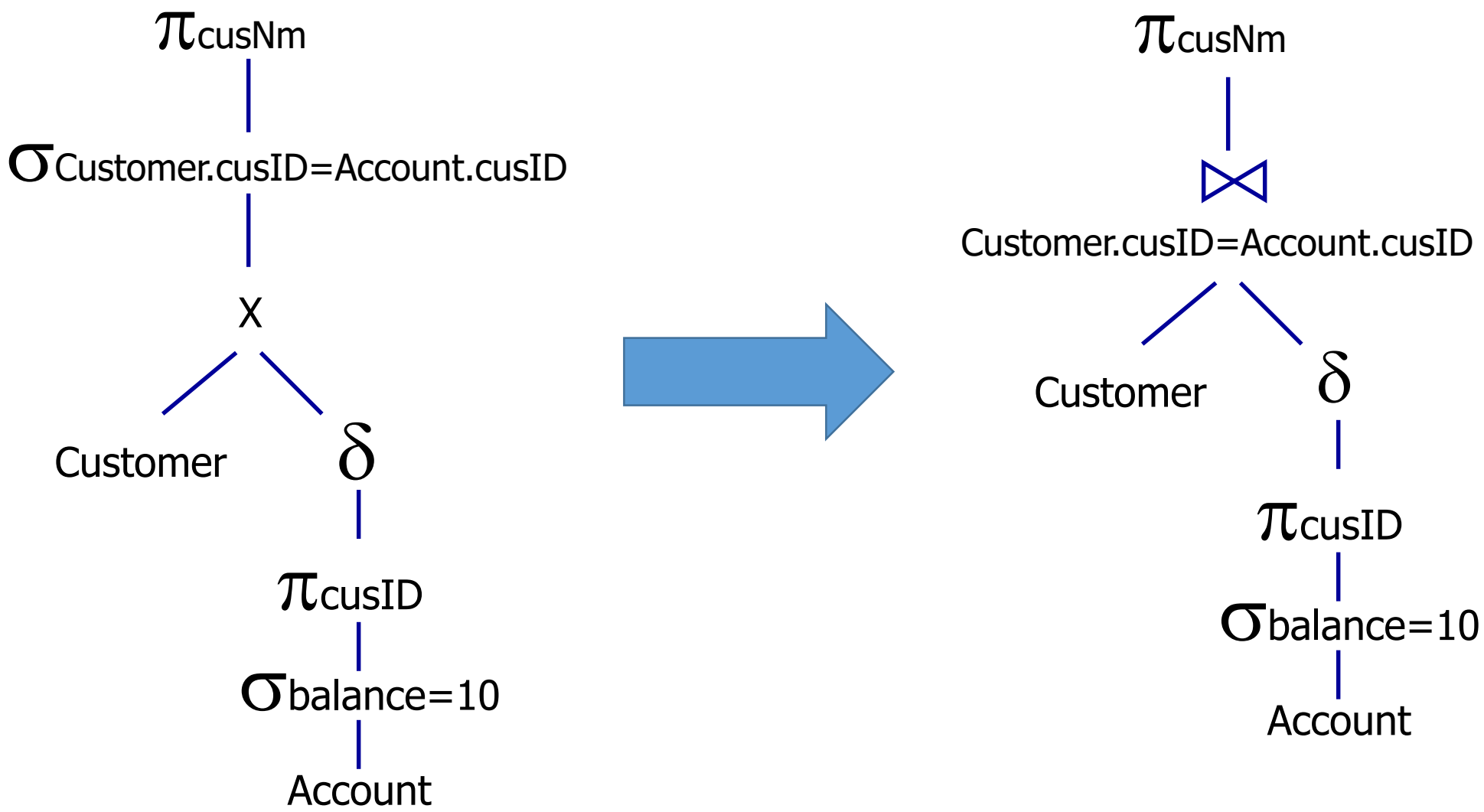
Biến đổi sang Đại số quan hệ



Biến đổi sang Đại số quan hệ



Biến đổi sang Đại số quan hệ



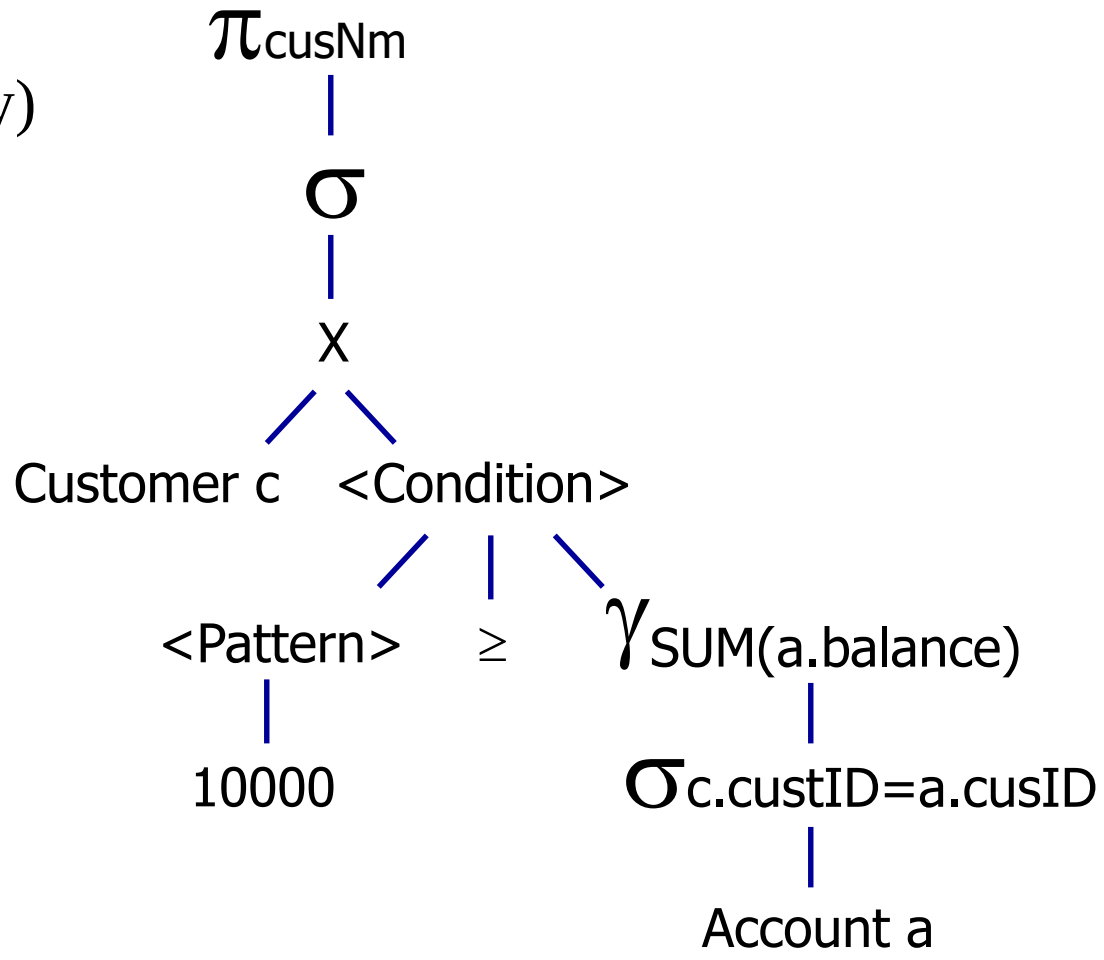
Biến đổi sang Đại số quan hệ

Ví dụ 3

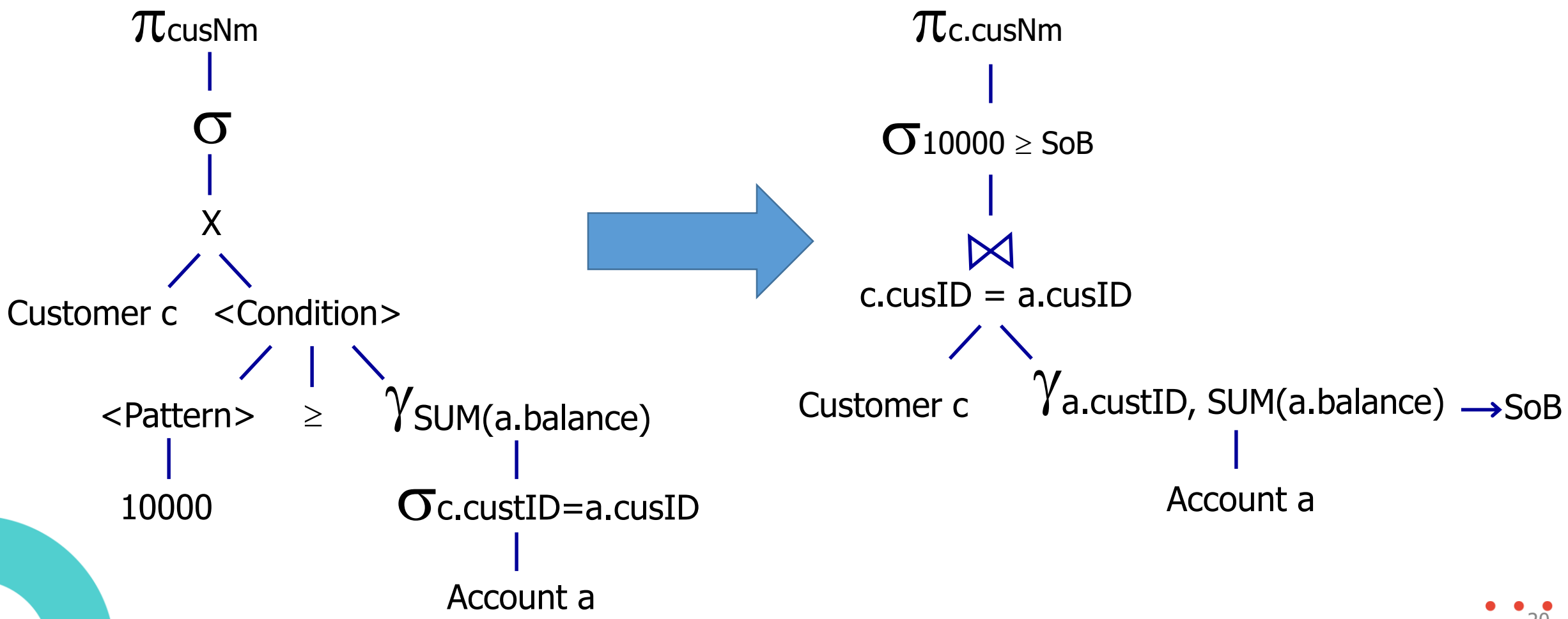
- Customer(cusID, cusNm, cusStreet, cusCity)
- Account(accID, cusID, balance)

Truy vấn lồng tương quan

```
SELECT c.cusNm
FROM   Customer c
WHERE 10000 >= (
                SELECT SUM(a.balance)
                FROM   Account a
                WHERE  a.cusID=c.cusID)
```



Biến đổi sang Đại số quan hệ



Tối ưu hóa câu truy vấn – các quy tắc

1. Quy tắc giao hoán & kết hợp
   - $R \times S = S \times R$
   - $(R \times S) \times T = R \times (S \times T)$
   - $R \bowtie S = S \bowtie R$
   - $(R \bowtie S) \bowtie T = R \bowtie (S \bowtie T)$
   - $R \cup S = S \cup R$
   - $R \cup (S \cup T) = (R \cup S) \cup T$

Tối ưu hóa câu truy vấn – các quy tắc

2. Quy tắc liên quan đến phép chọn σ

- Cho
  - p là vị từ chỉ có các thuộc tính của R
  - q là vị từ chỉ có các thuộc tính của S
  - m là vị từ có các thuộc tính của R và S
- $\sigma_{p1 \wedge p2}(R) = \sigma_{p1}[\sigma_{p2}(R)]$
- $\sigma_{p1 \vee p2}(R) = [\sigma_{p1}(R)] \cup [\sigma_{p2}(R)]$

Tối ưu hóa câu truy vấn – các quy tắc

3. Qui tắc σ, $\bowtie$

- $\sigma_p(R \bowtie S) = [\sigma_p(R)] \bowtie S$
- $\sigma_q(R \bowtie S) = R \bowtie [\sigma_q(S)]$
- $\sigma_{p \wedge q}(R \bowtie S) = [\sigma_p(R)] \bowtie [\sigma_q(S)]$
- $\sigma_{p \wedge q \wedge m}(R \bowtie S) = \sigma_m[\sigma_p(R) \bowtie \sigma_q(S)]$
- $\sigma_{p \vee q}(R \bowtie S) = [\sigma_p(R) \bowtie S] \cup [R \bowtie \sigma_q(S)]$

Phép chọn có tính chất quyết định trong vấn đề tối ưu hóa câu truy vấn, vì có khuynh hướng làm giảm kích thước truy vấn

Quy tắc: đưa phép chọn xuống càng sâu trong cây biểu diễn càng tốt mà không làm thay đổi kết quả - push selection down the tree

Tối ưu hóa câu truy vấn – các quy tắc

4. Quy tắc $\sigma, \cup$ và $\sigma, -$
   - $\sigma_c (R \cup S) = \sigma_c (R) \cup \sigma_c (S)$
   - $\sigma_c (R - S) = \sigma_c (R) - S = \sigma_c (R) - \sigma_c (S)$

Tối ưu hóa câu truy vấn – các quy tắc

- 5. Quy tắc phép chiếu π
- Cho
  - X = tập thuộc tính con của R
  - Y = tập thuộc tính con của R
  - XY = X ∪ Y
- Ta **KHÔNG** có
  - $\pi_{XY}(R) = \pi_X[\pi_Y(R)]$

Tối ưu hóa câu truy vấn – các quy tắc

6. Quy tắc π, $\bowtie$

- Cho
  - X=tập thuộc tính con của R
  - Y=tập thuộc tính con của S
  - Z=tập giao thuộc tính của R và S

Pushing projections

$$\pi_{XY}(R \bowtie S) = \pi_{XY}[\pi_{XZ}(R) \bowtie \pi_{YZ}(S)]$$

Tối ưu hóa câu truy vấn – các quy tắc

7. Quy tắc σ, π

- Cho
  - X=tập thuộc tính con của R
  - Z=tập thuộc tính con của R xuất hiện trong vị từ p
- Ta có

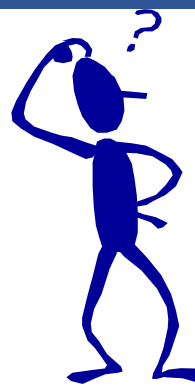
$$\pi_X [\sigma_p (R)] = \pi_X \{\sigma_p [\pi_{XZ} (R)]\}$$

Tối ưu hóa câu truy vấn – các quy tắc

- Nhận xét Quy tắc σ, π
- Ví dụ
  - R(A, B, C, D, E)
  - X={E}
  - p: A=3 ∧ B='a'

$\pi_x [\sigma_p (R)]$

Chọn trước tốt hơn???



$\pi_E \{\sigma_{A=3 \wedge B= 'a'}[\pi_{ABE}(R)]\}$

Chiếu trước tốt hơn???

- Bình thường
  - Chiếu trước
- Nhưng
  - Giả sử A và B được cài đặt chỉ mục (index)
  - Physical query plan dùng chỉ mục để chọn ra những bộ có A=3 và B=‘a’ trước
  - Nếu thực hiện chiếu trước $\pi_{ABE}(R)$ thì chỉ mục trên A và B là vô ích
  - Chọn trước

→Thông thường chọn trước tốt hơn

Tối ưu hóa câu truy vấn – các quy tắc

8. Quy tắc σ, π, ⋈

- Cho
  - X= tập thuộc tính con của R
  - Y=tập thuộc tính con của S
  - Z=tập giao thuộc tính của R và S
  - Z'=Z ∪ {các thuộc tính xuất hiện trong vị từ p}

$$\pi_{XY}[\sigma_p(R \bowtie S)] = \pi_{XY}\{\sigma_p[\pi_{XZ'}(R) \bowtie \pi_{YZ'}(S)]\}$$

Tối ưu hóa câu truy vấn – các quy tắc

9. Quy tắc $\times$, $\bowtie$

- $R \bowtie_c S = \sigma_c (R \times S)$
- $R \bowtie S = \pi_L [\sigma_c (R \times S)]$
  - Trong đó, $R \bowtie S$ là phép kết tự nhiên. L là danh sách các thuộc tính của R và S

Tối ưu hóa câu truy vấn – các quy tắc

10 Qui tắc δ - Loại bỏ dữ liệu trùng

$$\delta(R \bowtie S) = \delta(R) \bowtie \delta(S)$$

$$\delta(R \times S) = \delta(R) \times \delta(S)$$

$$\delta[\sigma_C(R)] = \sigma_C[\delta(R)]$$

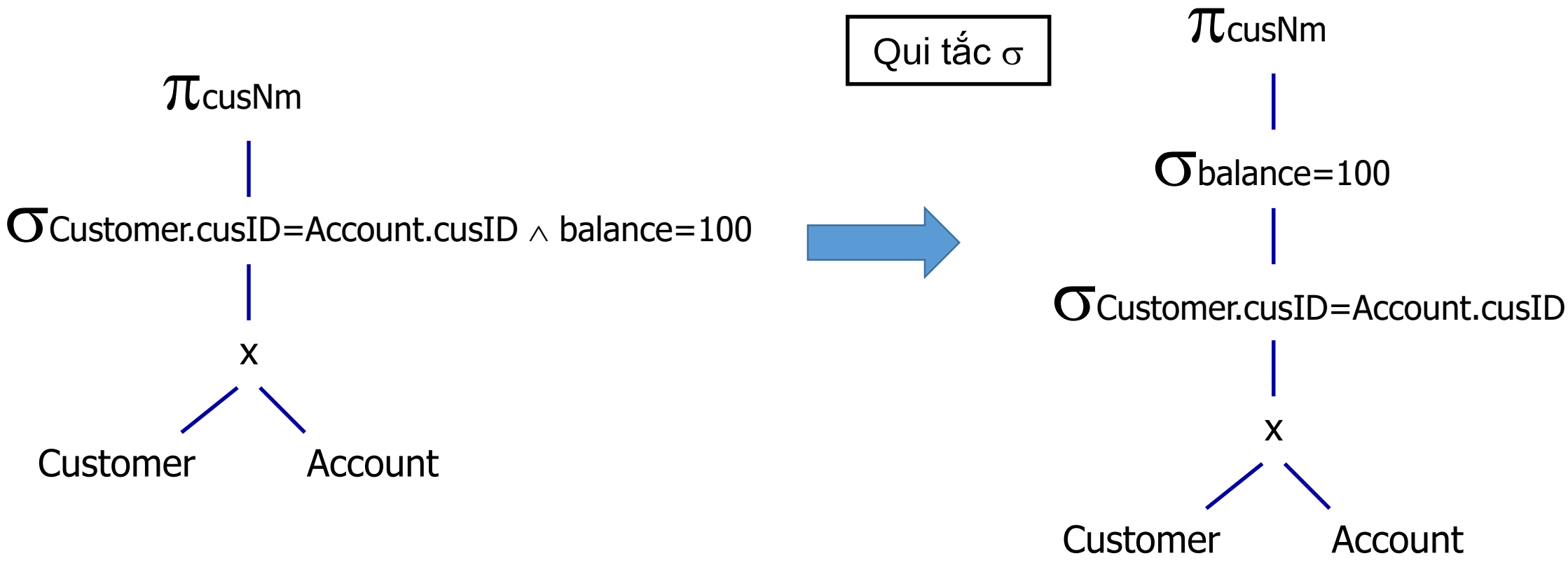
$$\delta(R \cap_B S) = \delta(R) \cap_B S = R \cap_B \delta(S) = \delta(R) \cap_B \delta(S)$$

Tối ưu hóa câu truy vấn – các quy tắc

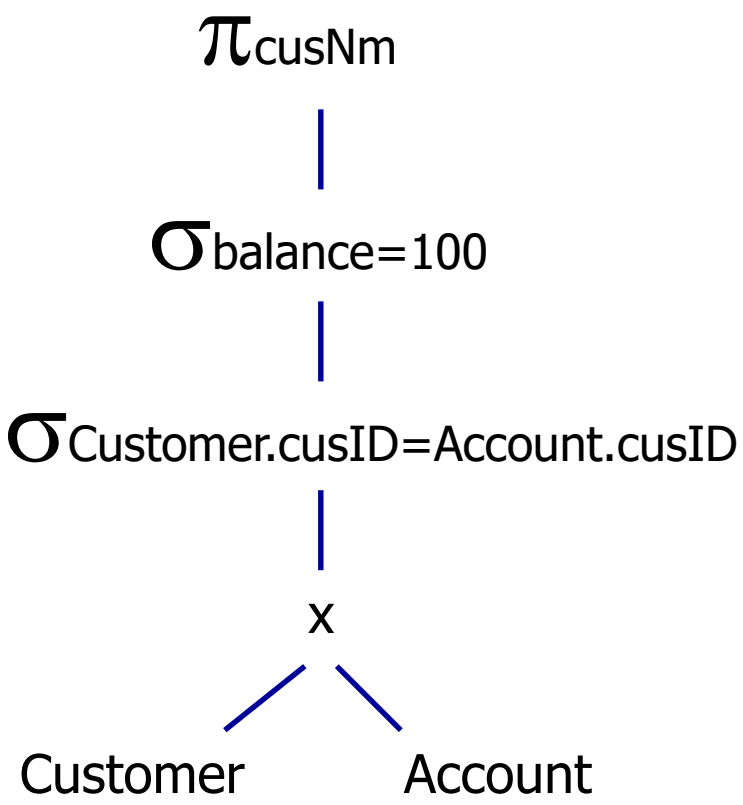
11 Quy tắc γ - gom nhóm

- Cho
  - X=tập thuộc tính trong R được gom nhóm
  - Y=X ∪ {một số thuộc tính khác của R}
  - $\delta[\gamma_X(R)] = \gamma_X(R)$
  - $\gamma_X(R) = \gamma_X[\pi_Y(R)]$

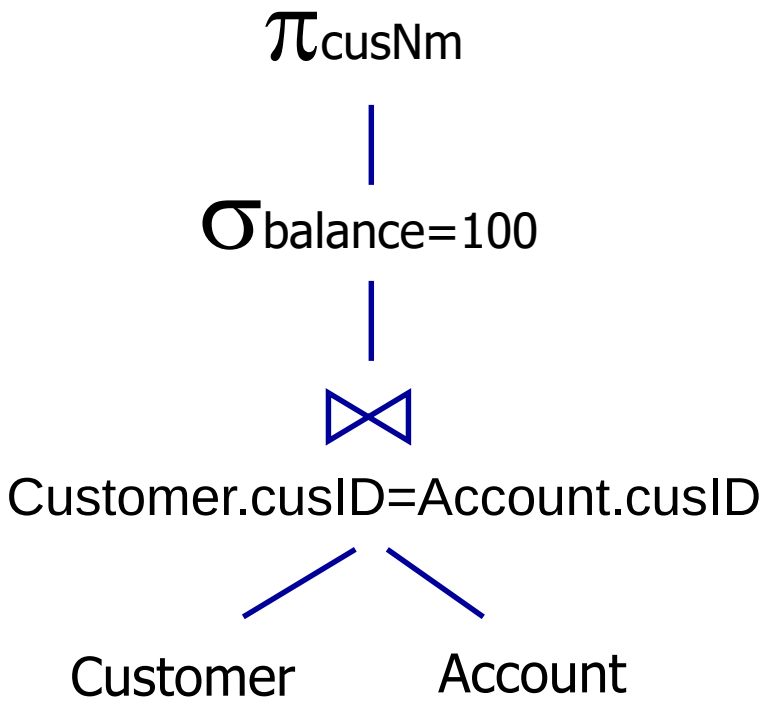
Tối ưu hóa câu truy vấn – các quy tắc



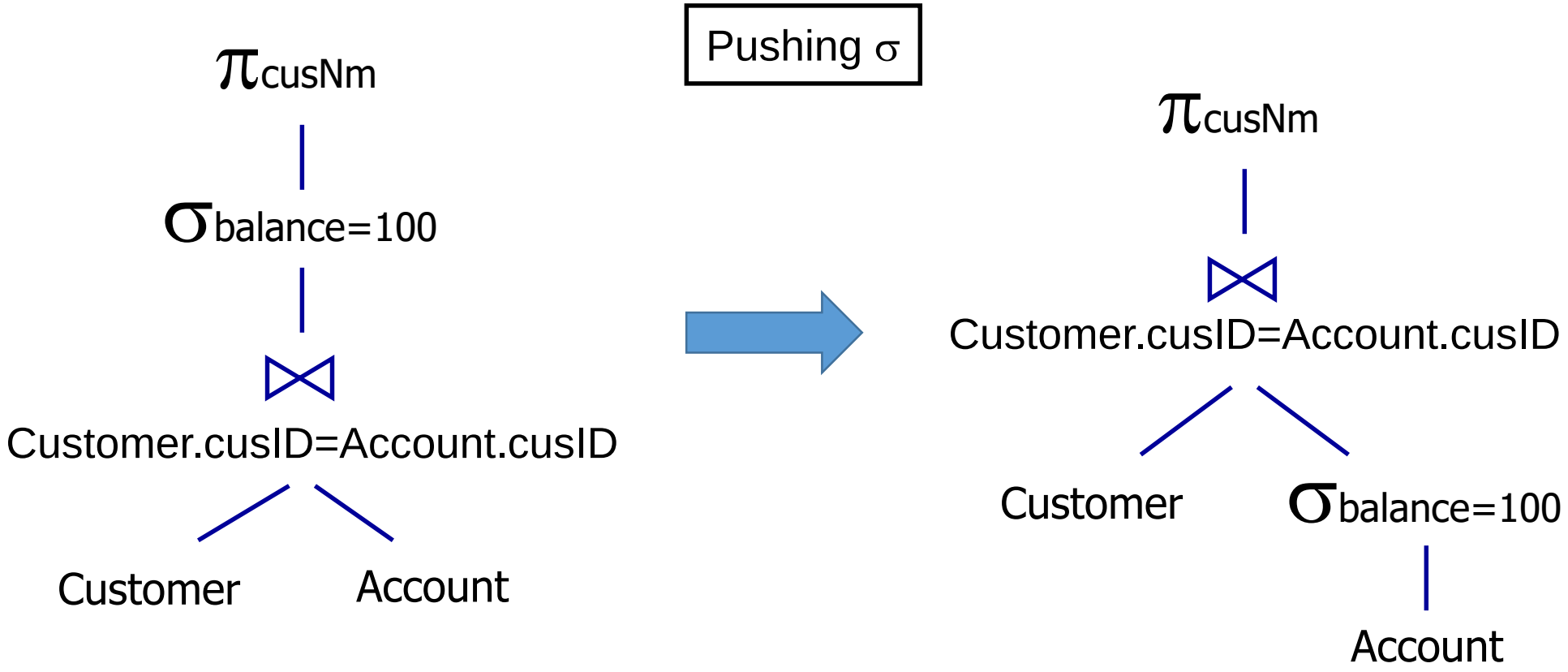
Tối ưu hóa câu truy vấn – các quy tắc



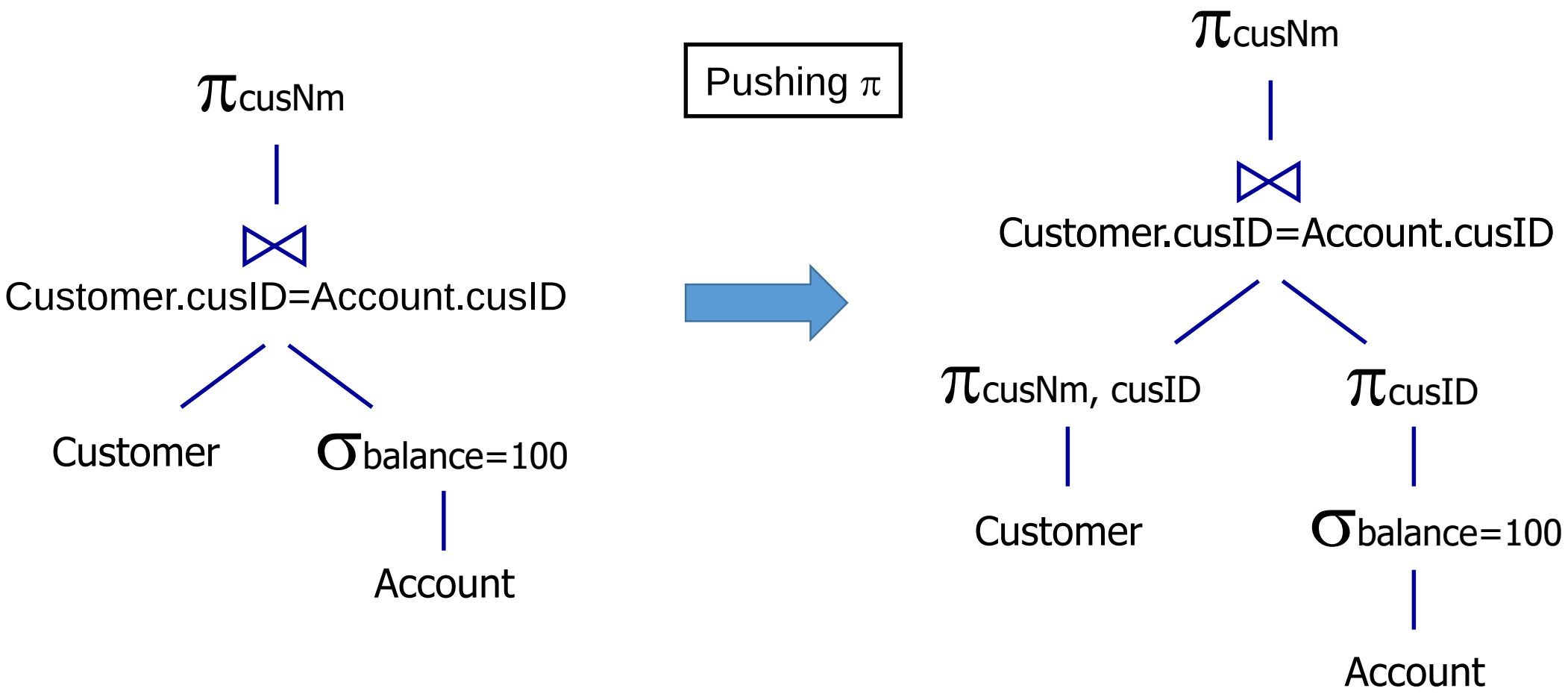
Qui tắc σ, ⋈



Tối ưu hóa câu truy vấn – các quy tắc



Tối ưu hóa câu truy vấn – các quy tắc



Tóm tắt các quy tắc biến đổi tương đương trong đại số quan hệ

- QT1: Xử lý các toán tử AND
  - $\sigma_{c1 \wedge c2 \ldots \wedge cn}(R) = \sigma_{c1}(\sigma_{c2}(\ldots(\sigma_{cn}(R))\ldots))$
- QT2: Thay đổi thứ tự các phép chọn
  - $\sigma_{c1}(_{c2}(R)) = \sigma_{c2}(_{c1}(R))$
- QT3: Xử lý các phép chiếu
  - $\pi_{ds1}(\pi_{ds2}(\ldots(\pi_{dsn}(R))\ldots)) = \pi_{ds1}(R)$
- QT4: Thay đổi thứ thự các phép chọn và phép chiếu
  - $\pi_{A1, A2, \ldots, An}(\sigma_c(R)) = \sigma_c(\pi_{A1, A2, \ldots, An}(R))$ – nếu $c \subset [A1, A2, \ldots, An]$
- QT5: Tính giao hoán của phép kết và tích Descartes
  - $R \bowtie_c S = S \bowtie_c R$
  - $R \times S = S \times R$

Tóm tắt các quy tắc biến đổi tương đương trong đại số quan hệ

- QT 6a: Thay đổi thứ tự giữa phép chọn và kết
  - $\sigma_c(R \bowtie S) = (\sigma_c(R)) \bowtie S$ – Nếu $c \subset R$
- QT 6b: phân phối giữa phép chọn và phép kết
  - $\sigma_c(R \bowtie S) = (\sigma_{c1}(R)) \bowtie (\sigma_{c2}(S))$ - Nếu $c = c1 \wedge c2$; $c1 \in R$ và $c2 \in S$
- QT 7: Phân phối giữa phép chiếu và phép kết
- $\pi_L(R \bowtie_c S) = (\pi_{A1\ldots,An,\, c}(R)) \bowtie_c (\pi_{B1,\ldots,Bm,\, c}(S))$
- QT 8: Giao hoán của phép hội và phép giao
- $R \cup S = S \cup R$
- $R \cap S = S \cap R$

Tóm tắt các quy tắc biến đổi tương đương trong đại số quan hệ

- QT 9: Kết hợp giữa phép kết, tích Descartes, hội và giao
  - $(R \theta S) \theta T = R \theta (S \theta T)$; Trong đó θ là một trong các phép toán $\bowtie$, $\times$, $\cap$, $\cup$
- QT 10: Phân phối của phép chọn đối với các phép toán
  - $\sigma_c(R \theta S) = (\sigma_c(R)) \theta (\sigma_c(S))$; Trong đó θ là một trong các phép toán $\cap$, $\cup$, -
- QT 11: Phân phối của phép chiếu đối với các phép toán
  - $\pi_L(R \theta S) = (\pi_L(R)) \theta (\pi_L(S))$; Trong đó θ là một trong các phép toán $\cap$, $\cup$, -
- QT12: chuyển các phép (σ, $\times$) thành phép kết
  - $\sigma_c(R \times S) = R \bowtie_c S$

Tối ưu hóa: Giải thuật heuristic

1. Áp dụng QT 1, tách các phép chọn liên tiếp thành 1 dãy các phép chọn.
2. Áp dụng QT 2,4,6 và 10, để đẩy phép chọn xuống càng sâu càng tốt.
3. Áp dụng QT 9 để tái tổ chức cây cú pháp sao cho phép chọn được thực hiện có lợi nhất (chọn ít nhất)→heuristic.
4. Phối hợp tích Decartes với các phép chiếu thích hợp theo sau.
5. Áp dụng QT 3, 4, 7 và 11 để đẩy phép chiếu xuống càng sâu càng tốt (có thể phát sinh phép chiếu mới).
6. Tập trung các phép chọn.
7. Áp dụng QT3 để loại những phép chiếu vô ích.

Ví dụ: giải thuật heuristic

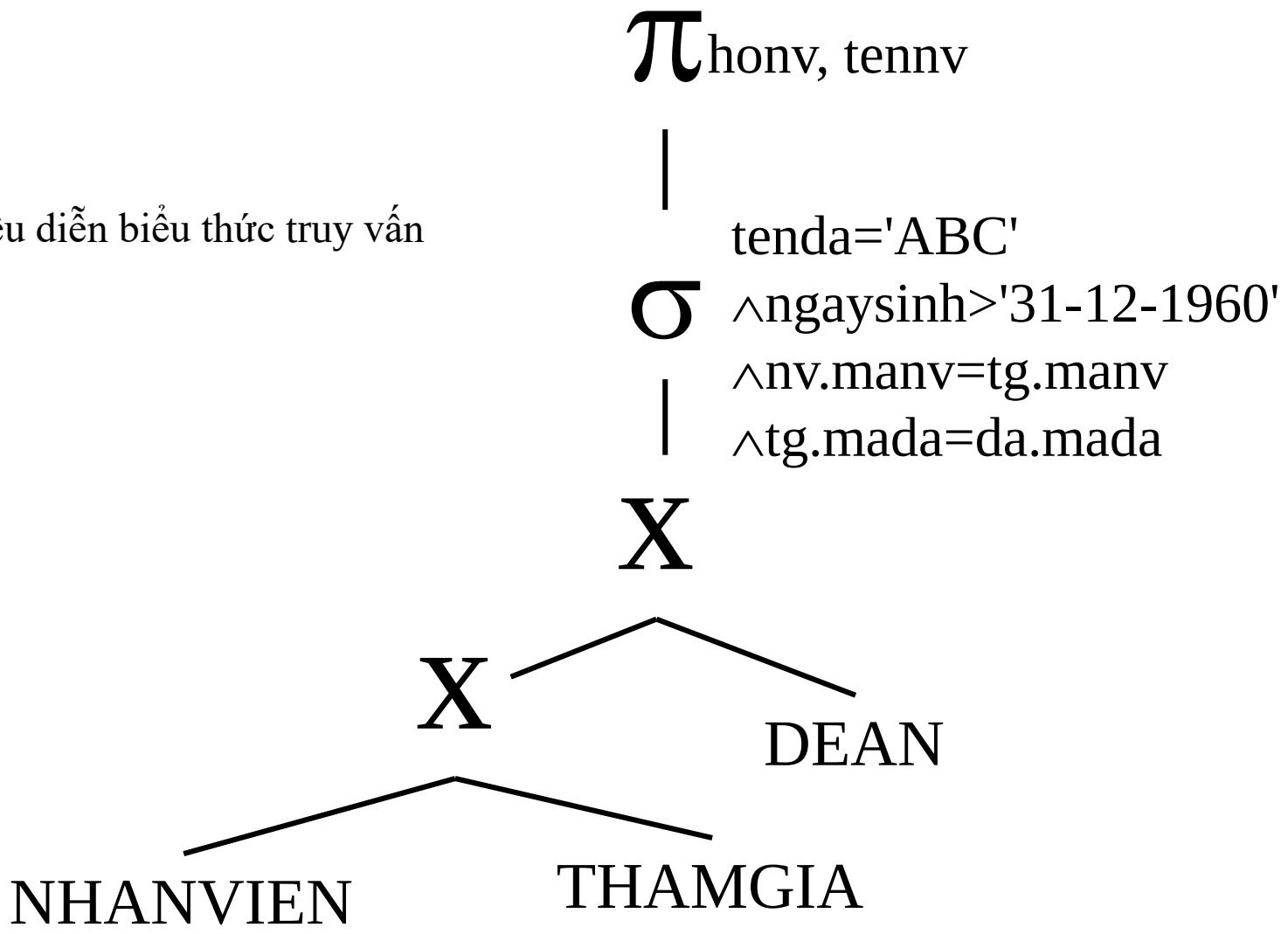
- Liệt kê họ tên NHANVIEN sinh sau năm 1960, có làm dự án 'ABC'
- Ngôn ngữ SQL:

```
SELECT    honv, tennv
FROM      NHANVIEN nv, THAMGIA tg, DEAN da
WHERE     tenda= 'ABC' AND ngaysinh> '31-12-1960'
          AND nv.manv=tg.manv
          AND tg.mada=da.mada
```

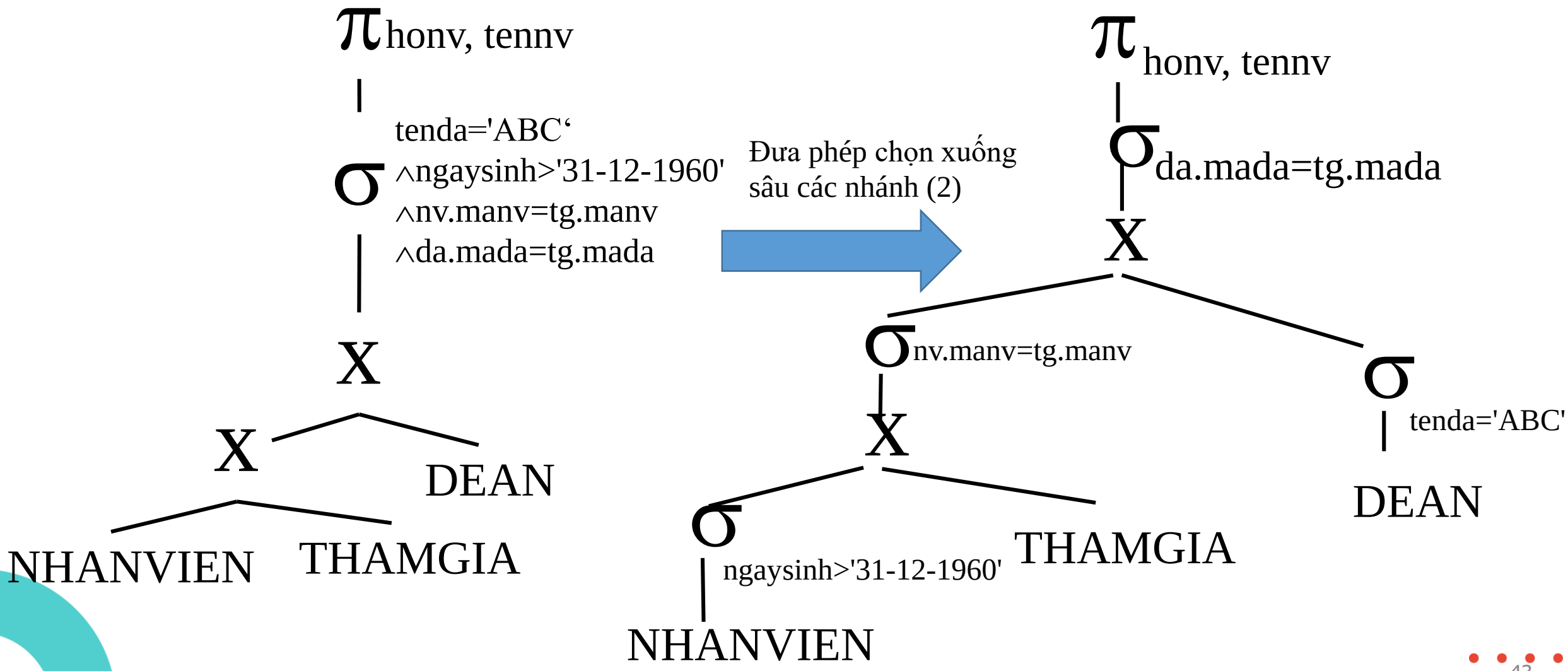
$$\pi_{honv,\ tennv}\sigma_{tenda='ABC'\wedge ngaysinh>\ '31\text{-}12\text{-}1960'\ \wedge nv.manv=tg.manv\ \wedge tg.mada=da.mada}(NHANVIEN \times THAMGIA \times DEAN)$$

$\pi_{honv,\ tennv}\sigma_{tenda='ABC' \wedge ngaysinh> '31\text{-}12\text{-}1960' \wedge nv.manv=tg.manv \wedge tg.mada=da.mada}(NHANVIEN \times THAMGIA \times DEAN)$

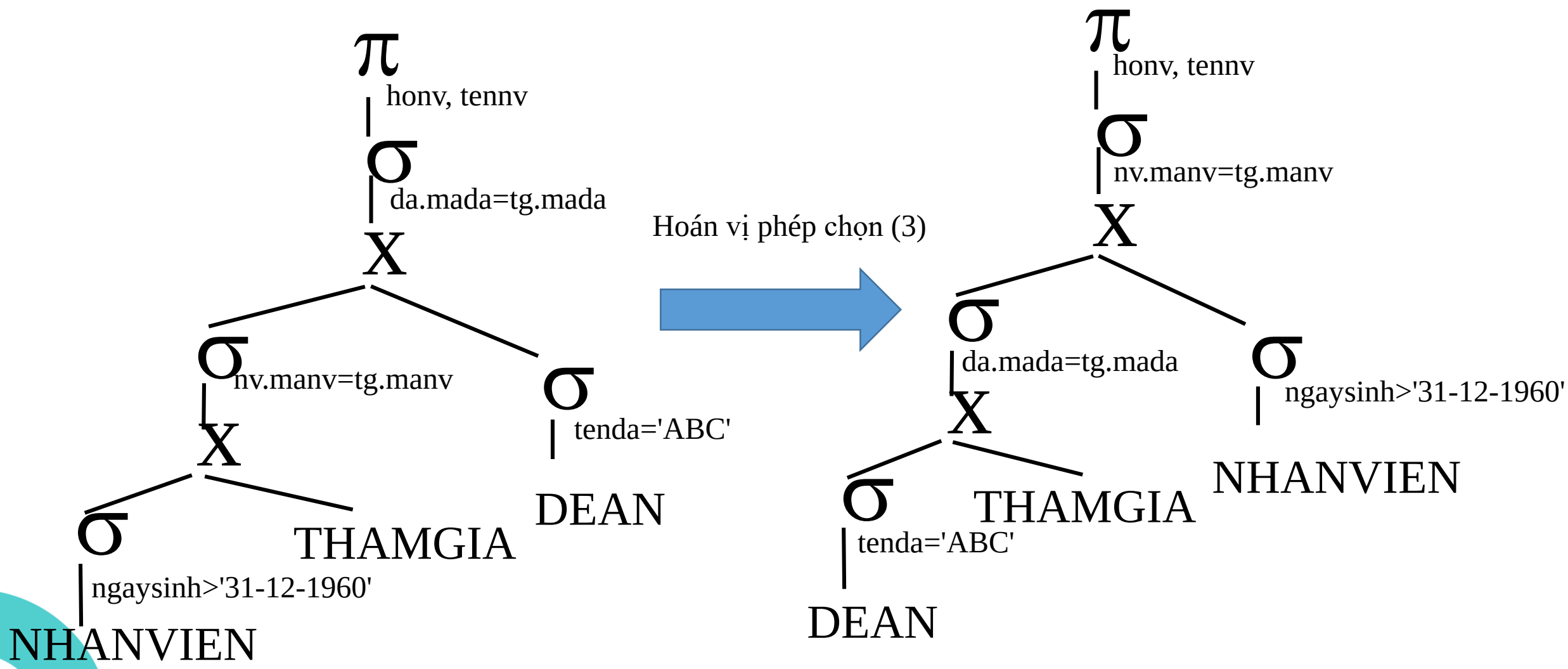
- Cây biểu diễn biểu thức truy vấn



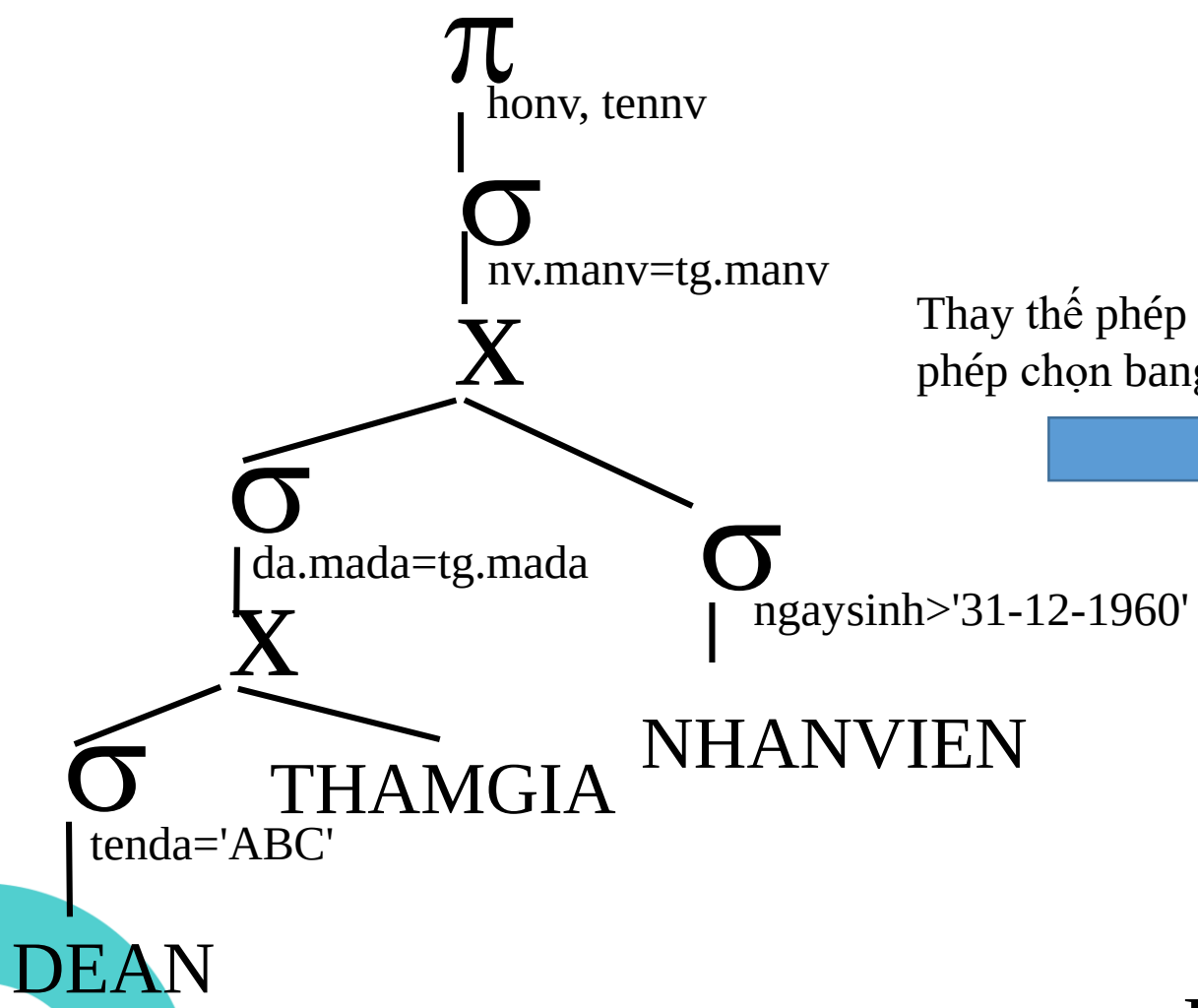
$\pi_{honv,\ tennv}\sigma_{tenda='ABC' \wedge ngaysinh> '31\text{-}12\text{-}1960' \wedge nv.MANV=tg.MANV \wedge da.mada=tg.mada}(NHANVIEN \times THAMGIA \times DEAN)$



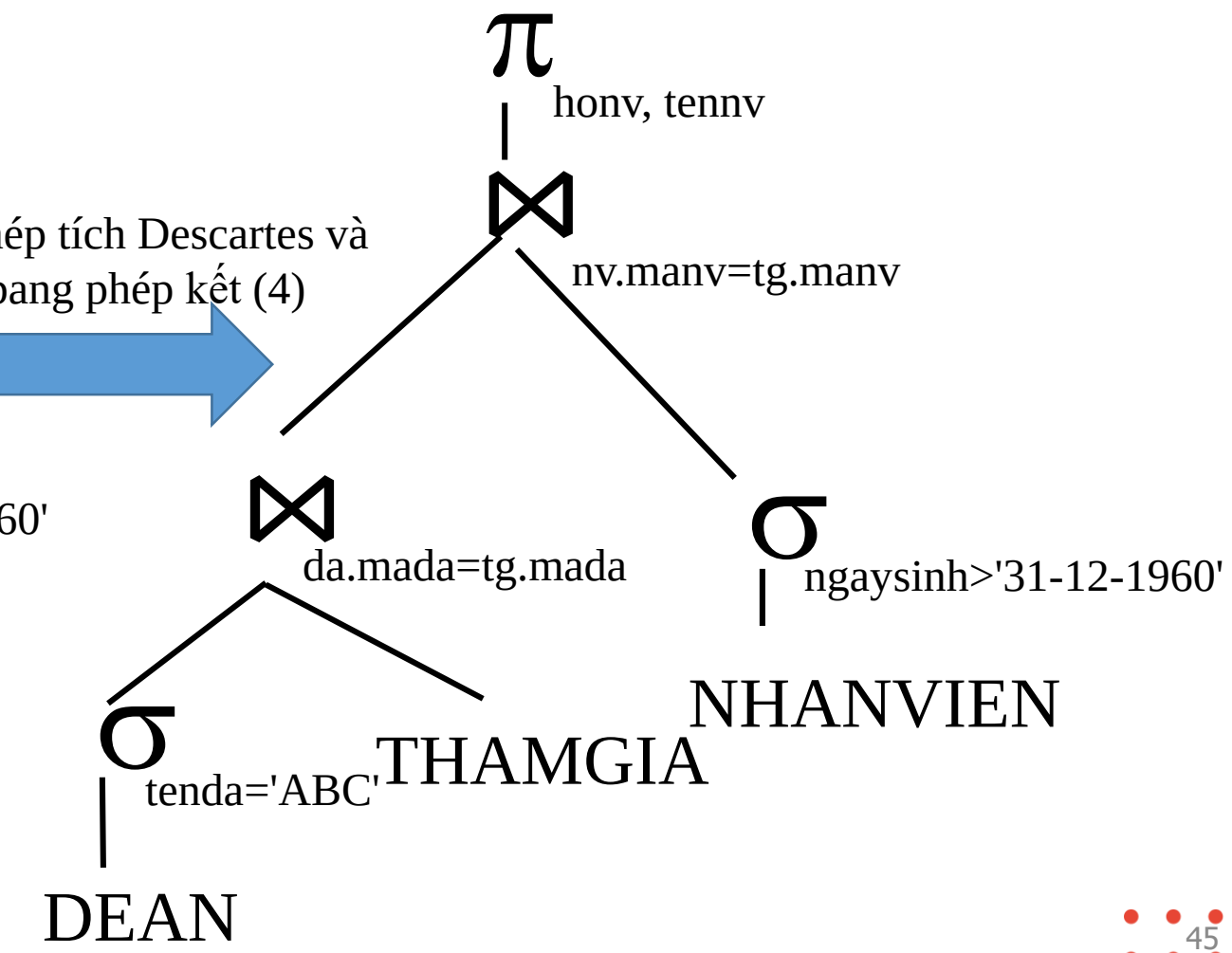
$\pi_{honv,\ tennv}\sigma_{tenda='ABC' \wedge ngaysinh> '31\text{-}12\text{-}1960' \wedge nv.MANV=tg.MANV \wedge da.mada=tg.mada}(NHANVIEN \times THAMGIA \times DEAN)$



$\pi_{honv,\ tennv}\sigma_{tenda='ABC' \wedge ngaysinh > '31\text{-}12\text{-}1960' \wedge nv.MANV=tg.MANV \wedge da.mada=tg.mada}(NHANVIEN \times THAMGIA \times DEAN)$



Thay thế phép tích Descartes và phép chọn bang phép kết (4)



$\pi_{honv,\ tennv}\sigma_{tenda='ABC' \wedge ngaysinh> '31\text{-}12\text{-}1960' \wedge nv.MANV=tg.MANV \wedge da.mada=tg.mada}(NHANVIEN \times THAMGIA \times DEAN)$