

# Chương 6: Tối ưu hoá câu truy vấn

- ☐ Giới thiệu
- ☐ Bộ biên dịch câu truy vấn (query compiler)
- ☐ Phân tích cú pháp
  - Cây phân tích (parse tree)
- ☐ Chuyển cây phân tích sang ĐSQH
  - Câu truy vấn đơn giản
  - Câu truy vấn lồng - lồng tương quan
- ☐ Quy tắc tối ưu cây truy vấn
- ☐ Ước lượng chi phí

## Giới thiệu

☐ R(A, B, C)

☐ S(C, D, E)

```
SELECT  B, D
FROM    R, S
WHERE   R.A='c' AND S.E=2 AND R.C=S.C
```

## Giới thiệu (tt)

☐ Câu truy vấn được thực hiện như thế nào?

R	A	B	C	S	C	D	E
	a	1	10		10	x	2
	b	1	10		20	y	2
	c	2	10		30	z	2
	d	2	10		40	x	1
	e	3	10		50	y	3

Kết quả

B	D
2	x

### □ Cách 1

- Tích cartesian
- Phép chọn (selection)
- Phép chiếu (projection)

$$\Pi_{B,D} [ \sigma_{R.A='c' \wedge S.E=2 \wedge R.C = S.C} (R \times S) ]$$

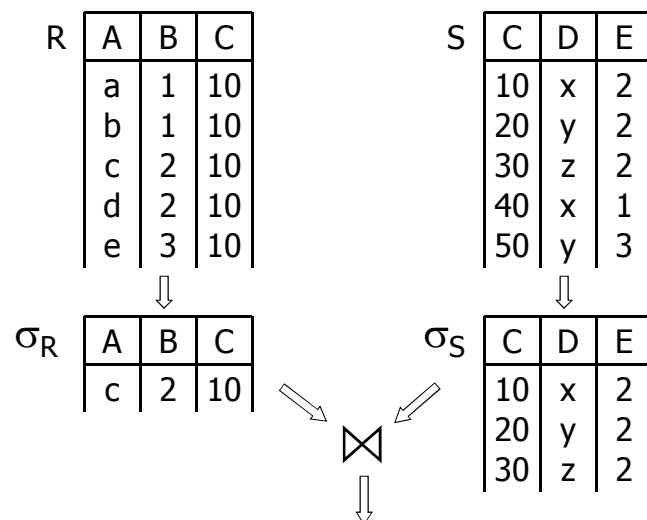
RxS

A	B	C	C	D	E
a	1	10	10	x	2
a	1	10	20	y	2
⋮					
c	②	10	10	ⓧ	2
c	2	10	20	y	2
c	2	10	30	z	2
⋮					

### ❑ Cách 2

- Phép chọn (selection)
- Phép kết (natural join)
- Phép chiếu (projection)

$$\Pi_{B,D} [ \sigma_{R.A='c'} (R) \bowtie \sigma_{S.E=2} (S) ]$$

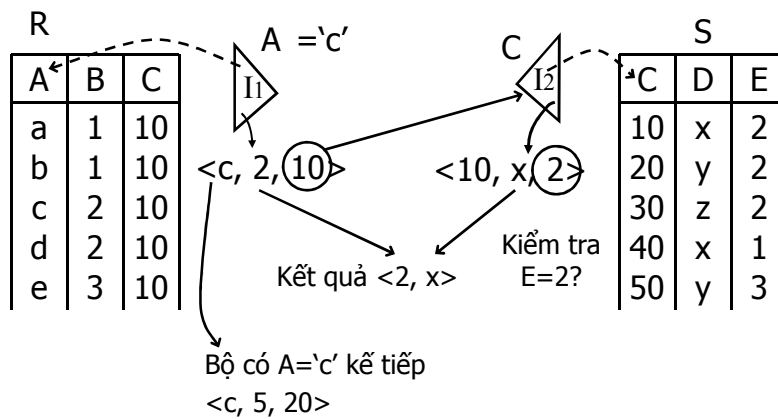


## Giới thiệu (tt)

### ❑ Cách 3 - sử dụng chỉ mục trên R.A và S.C

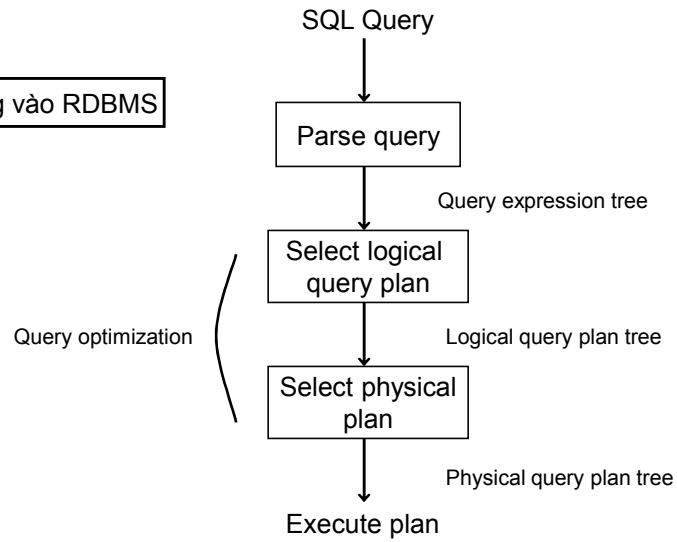
- Tìm các bộ trong R thỏa  $R.A = 'c'$
- Với mỗi bộ tìm thấy, tìm tiếp các bộ trong S thỏa  $R.C = S.C$
- Bỏ đi những bộ  $S.E \neq 2$
- Kết các bộ phù hợp của R và S
- Chiếu trên thuộc tính B và D

## Giới thiệu (tt)

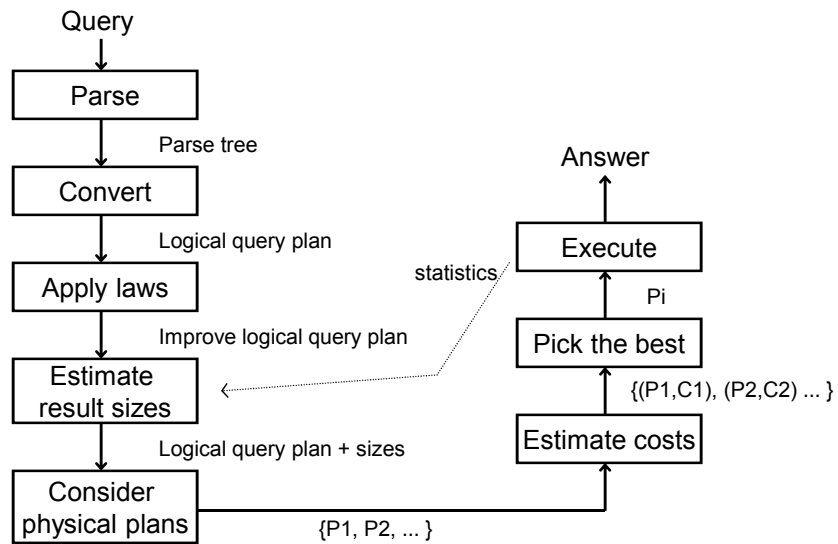


## Bộ biên dịch

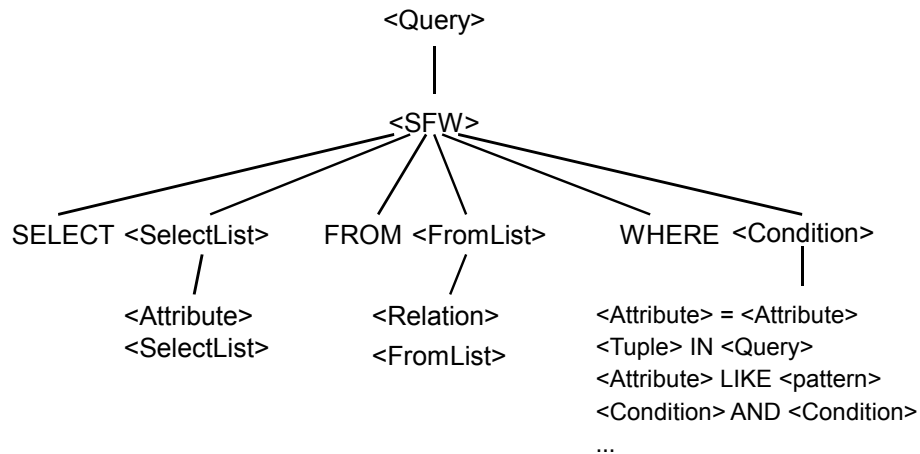
Tập trung vào RDBMS



## Quá trình biên dịch



## Cây phân tích



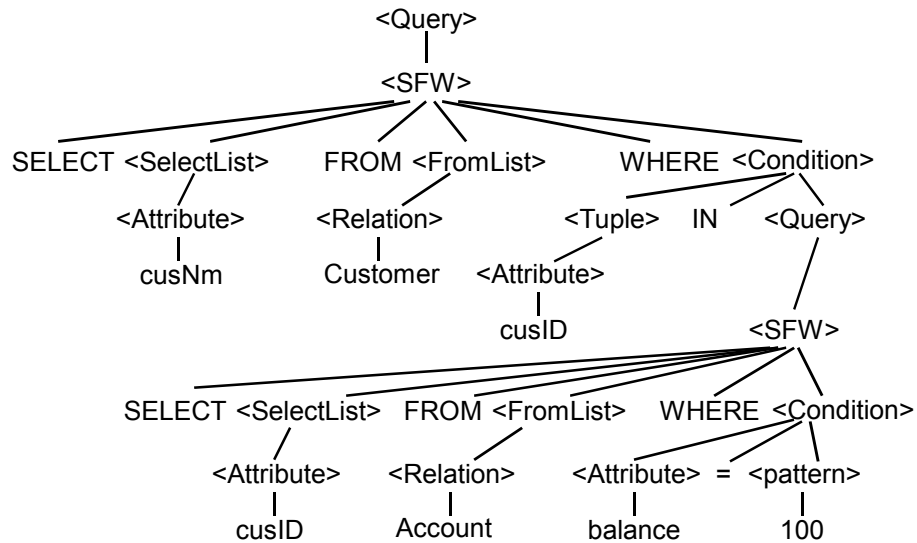
## Ví dụ 1

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

```

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

## Ví dụ 1 (tt)



## Ví dụ 2

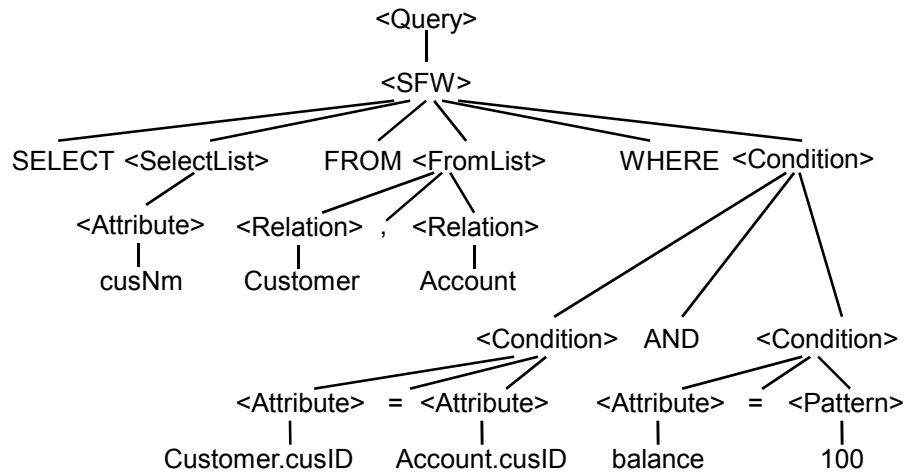
- ❑ Customer(cusID, cusNm, cusStreet, cusCity)
- ❑ Account(accID, cusID, balance)

```

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



## Ví dụ 2 (tt)



## Nhận xét

### ❑ Giới hạn

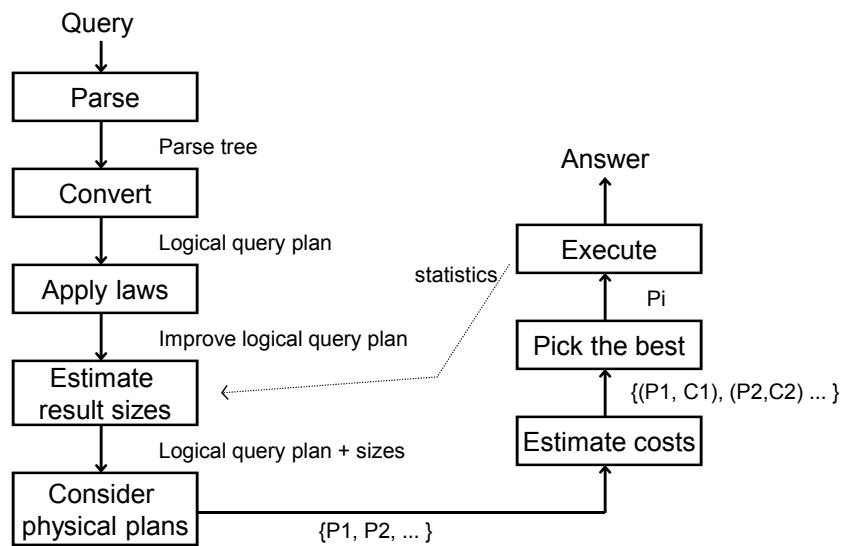
- GROUP BY
- HAVING
- ORDER BY
- DISTINCT
  
- Aggregation function (Max, Min, Count, Sum, Avg)
  
- Alias name

## Tiền xử lý (preprocessing)

### ❑ Kiểm tra ngữ nghĩa

- Quan hệ
  - Select
  - From
- Kiểu dữ liệu
  - Where

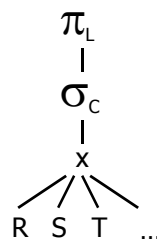
## Quá trình biên dịch



## Biến đổi sang ĐSQH

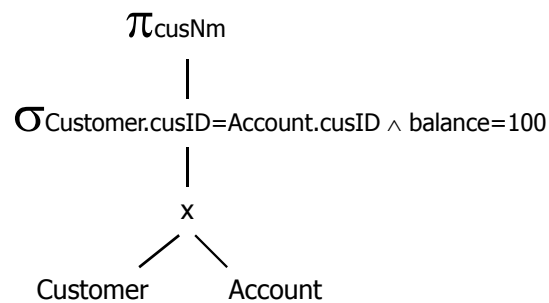
### ❑ Truy vấn đơn

- Xét câu 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  $\sigma_C$
  - Thay thế <SelectList> thành phép chiếu  $\pi_L$



Cây truy vấn

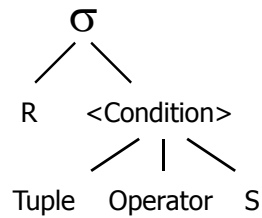
## Xét ví dụ 2



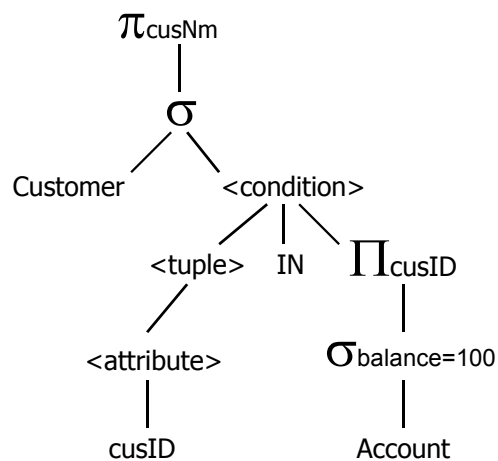
## Biến đổi sang ĐSQH (tt)

### ❑ Truy vấn lồng

- Tồn tại câu truy vấn con S trong <Condition>
- Áp dụng qui tắc <SFW> cho truy vấn con
- 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



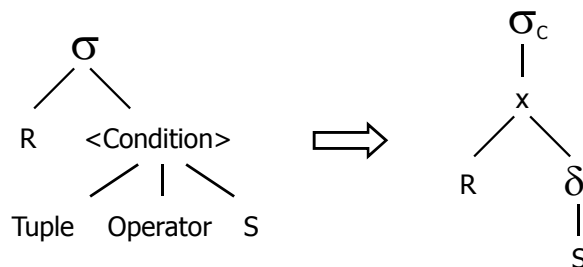
## Xét ví dụ 1



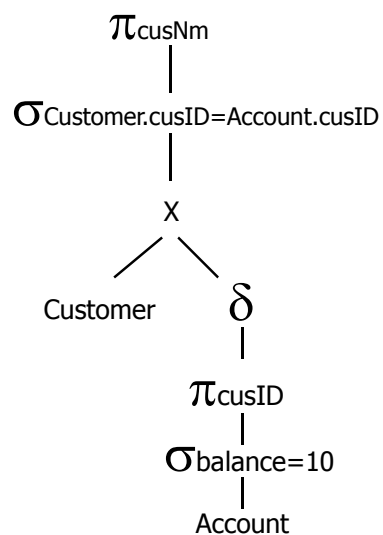
## Biến đổi sang ĐSQH (tt)

### ❑ Truy vấn lồng

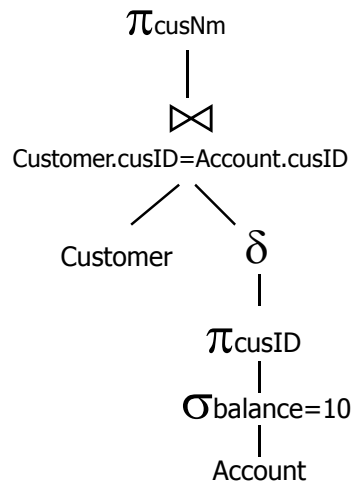
- Biến đổi phép chọn 2 biến
  - Thay thế <Condition> bằng 1 cây có gốc là S
    - Nếu S có các bộ trùng nhau thì phải lược bỏ bớt bộ trùng nhau đi
    - Sử dụng phép  $\delta$
  - Thay thế phép chọn 2 biến thành  $\sigma_C$
  - $\sigma_C$  là kết quả của phép cartesian của R và S



## Xét ví dụ 1 (tt)



## Xét ví dụ 1 (tt)



## Ví dụ 3

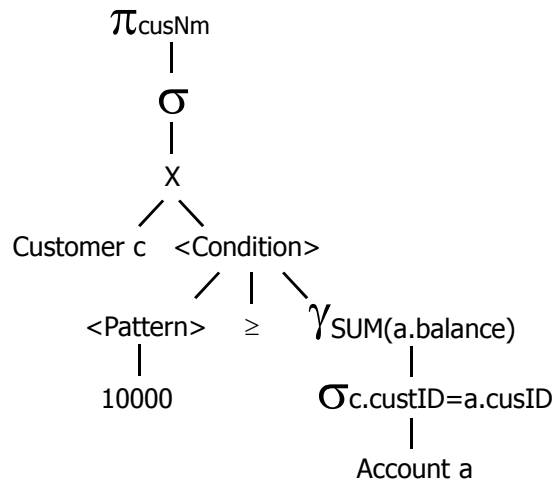
- ❑ Customer(cusID, cusNm, cusStreet, cusCity)
- ❑ Account(accID, cusID, balance)

```

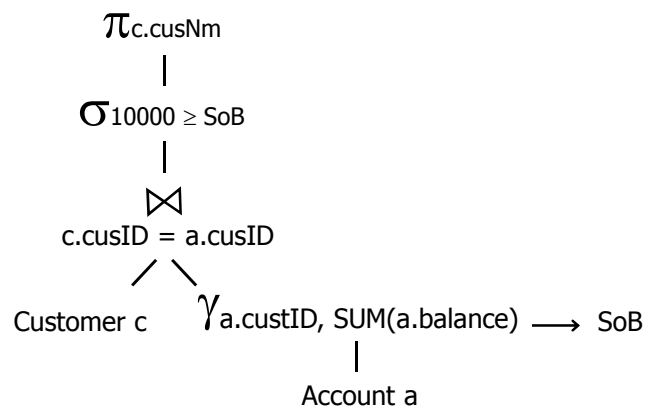
SELECT c.cusNm
FROM Customer c
WHERE 10000 >= (
    SELECT SUM(a.balance)
    FROM Account a
    WHERE a.cusID=c.cusID)
    
```

Truy vấn lồng tương quan

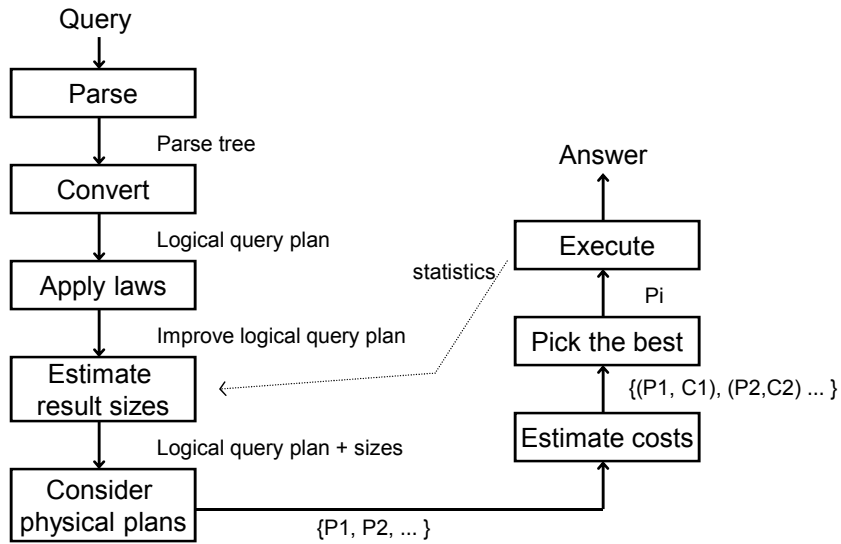
### Ví dụ 3 (tt)



### Ví dụ 3 (tt)



## Quá trình biên dịch



## Quy tắc: Kết tự nhiên, tích cartesian, hội

$$R \bowtie S = S \bowtie R$$

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

$$R \times S = S \times R$$

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

$$R \cup S = S \cup R$$

$$R \cup (S \cup T) = (R \cup S) \cup T$$



## Qui tắc: Phép chọn $\sigma$

- 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  $S$

Pushing selections

$$\sigma_{p1 \wedge p2}(R) = \sigma_{p1} [\sigma_{p2}(R)]$$

$$\sigma_{p1 \vee p2}(R) = [\sigma_{p1}(R)] \cup [\sigma_{p2}(R)]$$

Quan hệ  $R$  là tập hợp  
 $\cup_S$  là phép hội trên tập hợp

## Qui tắc: $\sigma, \bowtie$

$$\sigma_p(R \bowtie S) = [\sigma_p(R)] \bowtie S$$

$$\sigma_q(R \bowtie S) = R \bowtie [\sigma_q(S)]$$

### Qui tắc: $\sigma, \bowtie$ (tt)

$$\sigma_{p \wedge q}(R \bowtie S) = [\sigma_p(R)] \bowtie [\sigma_q(S)]$$

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

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

### Qui tắc: $\sigma, \cup$ và $\sigma, -$

$$\sigma_c(R \cup S) = \sigma_c(R) \cup \sigma_c(S)$$

$$\sigma_c(R - S) = \sigma_c(R) - S = \sigma_c(R) - \sigma_c(S)]$$

## Qui tắc: Phép chiếu $\pi$

- Cho
  - $X$  = tập thuộc tính con của  $R$
  - $Y$  = tập thuộc tính con của  $R$
- Ta có
  - $XY = X \cup Y$

$$\pi_{XY}(R) = \pi_X[\pi_Y(R)]$$

## Qui tắc: $\pi, \bowtie$

- 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

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

Except intersection and difference

## Qui tắc: $\sigma, \pi$

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

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

## Qui tắc: $\sigma, \pi, \bowtie$

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

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

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

## Nhận xét: $\sigma$ , $\pi$

- Ví dụ
  - $R(A, B, C, D, E)$
  - $X=\{E\}$
  - $p: A=3 \wedge B='a'$

$$\pi_X [\sigma_p (R)]$$

Chọn trước  
tốt hơn???



$$\pi_E \{ \sigma_p [\pi_{ABE}(R)] \}$$

Chiếu trước  
tốt hơn???

## Nhận xét: $\sigma$ , $\pi$ (tt)

- 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  $\pi_{AB}(R)$  thì chỉ mục trên A và B là vô ích
  - Chọn trước

→ Thông thường chọn trước tốt hơn

$$\sigma_C (R \bowtie S) = R \bowtie_C S$$

$$R \times S = \pi_L [\sigma_C (R \times S)]$$

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

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

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

$$\begin{aligned} \delta(R \cap_B S) &= \delta(R) \cap_B S = R \cap_B \delta(S) \\ &= \delta(R) \cap_B \delta(S) \end{aligned}$$

Except:  $\cup_B, \neg_B, \pi$

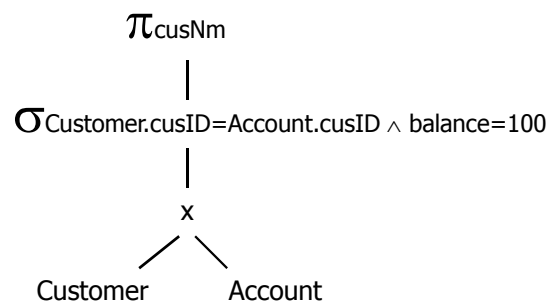
## Qui tắc: $\gamma$

- Cho
  - $X$  = tập thuộc tính trong  $R$  được gom nhóm
  - $Y = X \cup \{\text{một số thuộc tính khác của } R\}$

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

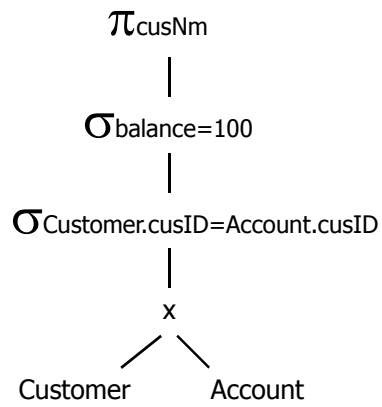
$$\gamma_X(R) = \gamma_X[\pi_Y(R)]$$

## Xét ví dụ 2



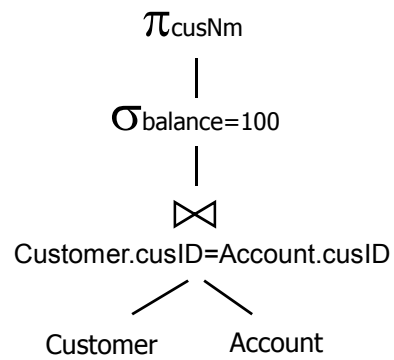
## Xét ví dụ 2

Qui tắc  $\sigma$



## Xét ví dụ 2 (tt)

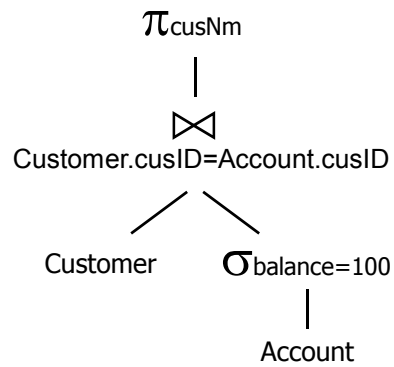
Qui tắc  $\sigma, \bowtie$





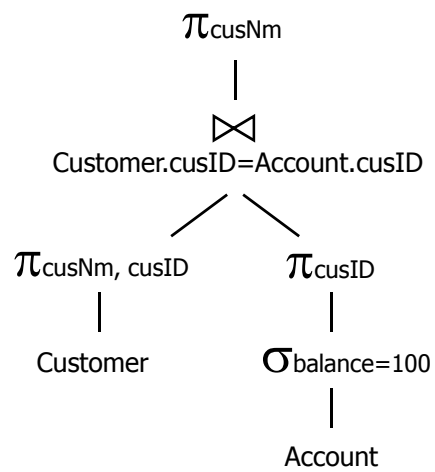
## Xét ví dụ 2 (tt)

Pushing  $\sigma$

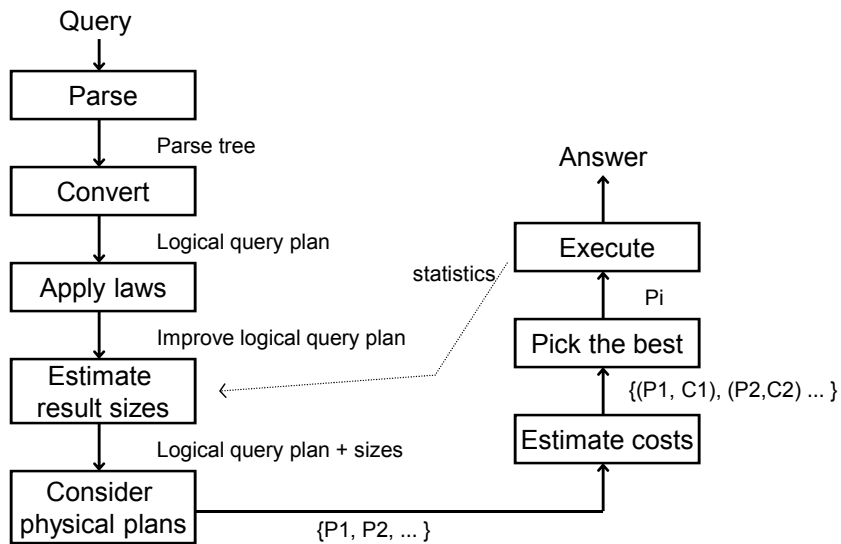


## Xét ví dụ 2 (tt)

Pushing  $\pi$



## Quá trình biên dịch



## Ước lượng chi phí

- Ước lượng kích thước cây truy vấn
  - Quan hệ
  - Các phép toán
- Ước lượng số lần truy xuất IOs
  - Số blocks được đọc hoặc ghi để thực hiện cây truy vấn

## Ước lượng kích thước

- Thống kê quan hệ R
  - $T(R)$ : số bộ trong R
  - $S(R)$ : tổng số byte của 1 bộ trong R
  - $B(R)$ : tổng số block chứa tất cả các bộ của R
  - $V(R, A)$ : số giá trị khác nhau mà thuộc tính A trong R có thể có

## Ví dụ

R	A	B	C	D
	x	1	10	a
	x	1	20	b
	y	1	30	a
	y	1	40	c
	z	1	50	d

A: chuỗi 20 bytes

B: số nguyên 4 bytes

C: ngày 8 bytes

D: chuỗi 68 bytes

1 block = 1024 bytes  
(block header: 24 bytes)

$$T(R) = 5$$

$$V(R, A) = 3$$

$$V(R, B) = 1$$

$$S(R) = 100$$

$$V(R, C) = 5$$

$$V(R, D) = 4$$

$$B(R) = 1$$

**Ước lượng:  $W = R_1 \times R_2$** 

$$S(W) = S(R_1) + S(R_2)$$

$$T(W) = T(R_1) \times T(R_2)$$

**Ước lượng:  $W = \sigma_{Z=val}(R)$** 

$$S(W) = S(R)$$

$$T(W) = \frac{T(R)}{V(R, Z)}$$

Số bộ trung bình thỏa điều kiện  $Z=val$

## Ước lượng: $W = \sigma_{Z \geq val} (R)$

$$T(W) = ???$$

- Cách 1

$$T(W) = \frac{T(R)}{2}$$

- Cách 2

$$T(W) = \frac{T(R)}{3}$$

## Ví dụ

- Cho
  - $R(A, B, C)$
  - $T(R) = 10000$
  - $V(R, A) = 50$
- Ước lượng kích thước biểu thức

$$S = \sigma_{A=10 \wedge B < 20} (R)$$

$$T(S) = \frac{T(R)}{V(R, A) \times 3} = \frac{10000}{50 \times 3} = 67$$

## Ví dụ (tt)

- Ước lượng kích thước biểu thức

$$S = \sigma_{A=10 \vee B < 20} (R)$$

- Giả sử
  - $n$  là  $T(R)$
  - $m_1$  là số bộ thỏa  $A=10$  trong  $R$
  - $m_2$  là số bộ thỏa  $B < 20$  trong  $R$

$$T(S) = n(1 - (1 - \frac{m_1}{n})(1 - \frac{m_2}{n}))$$

## Ước lượng: $W = R_1 \bowtie R_2$

- Cho
  - $X$  = tập thuộc tính của  $R_1$
  - $Y$  = tập thuộc tính của  $R_2$
- Xét trường hợp  $X \cap Y = \emptyset$

$$T(W) = ?$$

Tương tự  $R_1 \times R_2$

## Ước lượng: $W = R_1 \bowtie R_2$ (tt)

- Xét trường hợp  $X \cap Y = A$

 $R_1$ 

A	B	C

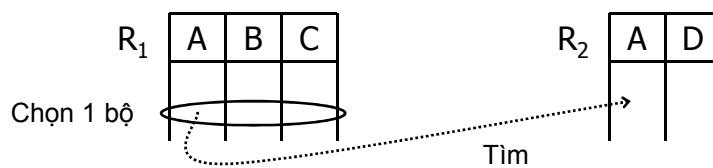
 $R_2$ 

A	D

- Giả sử

- $V(R_1, A) \leq V(R_2, A)$ 
  - Mọi giá trị của A trong  $R_1$  thì có trong  $R_2$
- $V(R_2, A) \leq V(R_1, A)$ 
  - Mọi giá trị của A có trong  $R_2$  thì có trong  $R_1$

## Ước lượng: $W = R_1 \bowtie R_2$ (tt)



1 bộ trong  $R_1$  sẽ thỏa với  $\frac{T(R_2)}{V(R_2, A)}$  bộ trong  $R_2$

$$T(W) = T(R_1) \times \frac{T(R_2)}{V(R_2, A)}$$

## Ước lượng: $W = R_1 \bowtie R_2$ (tt)

- $V(R_1, A) \leq V(R_2, A)$

$$T(W) = T(R_1) \times \frac{T(R_2)}{V(R_2, A)}$$

- $V(R_2, A) \leq V(R_1, A)$

$$T(W) = T(R_2) \times \frac{T(R_1)}{V(R_1, A)}$$

- Tổng quát

$$T(W) = \frac{T(R_1) T(R_2)}{\max\{V(R_1, A), V(R_2, A)\}}$$

## Ước lượng: $W = R_1 \bowtie R_2$ (tt)

- Xét trường hợp  $X \cap Y = A$

$R_1$	A	B	C

$R_2$	A	D

- $W(A, B, C, D)$

- Các thuộc tính không tham gia vào phép kết thì số lượng các giá trị vẫn giữ nguyên
- $V(W, A) = \min\{V(R_1, A), V(R_2, A)\}$
- $V(W, B) = V(R_1, B)$
- $V(W, C) = V(R_1, C)$
- $V(W, D) = V(R_2, D)$



## Ví dụ

$$Z = R_1(A, B) \bowtie R_2(B, C) \bowtie R_3(C, D)$$

$R_1$	$R_2$	$R_3$
$T(R_1) = 1000$	$T(R_2) = 2000$	$T(R_3) = 3000$
$V(R_1, A) = 50$	$V(R_2, B) = 200$	$V(R_3, C) = 90$
$V(R_1, B) = 100$	$V(R_2, C) = 300$	$V(R_3, D) = 500$

## Ví dụ (tt)

$$U = R_1(A, B) \bowtie R_2(B, C)$$

$$T(U) = \frac{1000 \times 2000}{200}$$

$$\begin{aligned} V(U, A) &= 50 \\ V(U, B) &= 100 \\ V(U, C) &= 300 \end{aligned}$$

## Ví dụ (tt)

$$Z = U \bowtie R_3(C, D)$$

$$T(Z) = \frac{1000 \times 2000 \times 3000}{200 \times 300}$$

$$V(Z, A) = 50$$

$$V(Z, B) = 100$$

$$V(Z, C) = 90$$

$$V(Z, D) = 500$$

## Nhận xét

- Phép chiếu
  - Phép tích
- } Ước lượng chính xác
- 
- Phép chọn
  - Phép kết
- } Ước lượng tương đối hợp lý
- số lượng bộ của các quan hệ tương đối lớn
  - giá trị của các thuộc tính phân bố đồng đều
- 
- Phép toán khác
    - Hội
    - Giao
    - Trừ
    - Lược bỏ trùng lặp
    - Gộp nhóm

## Ước lượng: $W = R_1 \cup R_2$

- $R_1$  và  $R_2$  là bag

$$T(W) = T(R_1) + T(R_2)$$

- $R_1$  và  $R_2$  là set


$$T'(W) = T(R_1) + T(R_2)$$


$$T''(W) \leq T(R_1) + T(R_2)$$

$$\rightarrow T(W) = \frac{T'(W) + T''(W)}{2}$$

## Ước lượng: $W = R_1 \cap R_2$

- Cách 1

- TH1:  $T'(W)=0$  

- TH2:  $T''(W)=T(R_1)$  hoặc  $T''(W)=T(R_2)$  

$$\rightarrow T(W) = \frac{T'(W) + T''(W)}{2}$$

- Cách 2

- Trường hợp đặc biệt của phép kết tự nhiên
  - Chỉ áp dụng cho  $\cap_s$

$$T(W) = \frac{T(R_1) T(R_2)}{\max\{V(R_1, Z), V(R_2, Z)\}}$$

### Ước lượng: $W = R_1 - R_2$

- TH1:  $T(W) = T(R_1)$
- TH2:  $T(W) = T(R_1) - T(R_2)$

$$\rightarrow T(W) = T(R_1) - \frac{1}{2} T(R_2)$$

### Ước lượng: $W = \delta(R)$

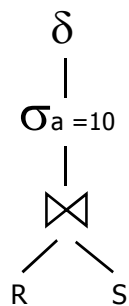
- TH1:  $T(W) = 1$ 
  - Nếu trong R không có bộ nào thì  $T(W)=0$
- TH2:  $T(W) = T(R)$ 
  - $R(a_1, a_2, \dots, a_n)$
  - Số bộ phân biệt tối đa của R là tích các  $V(R, a_i)$ ,  $i=1..n$

$$\rightarrow T(W) = \min\left\{\frac{1}{2} T(R), \text{tích các } V(R, a_i)\right\}$$

## Ước lượng: $W = \gamma(R)$

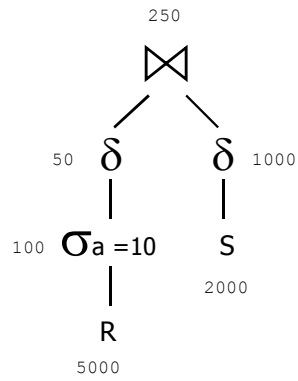
- $\gamma_L(R)$ 
    - Số lượng bộ trong W và cũng là số lượng nhóm
  - TH1:  $T(W) = 1$
  - TH2:  $T(W) = T(R)$ 
    - $R(a_1, a_2, \dots, a_n)$
    - Số lượng nhóm tối đa là tích các  $V(R, a_i), i=1..n$
- $\rightarrow T(W) = \min\{\frac{1}{2} T(R_1), \text{tích các } V(R, a_i)\}$

## Ví dụ

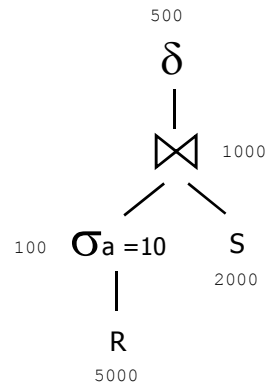


- $R(a, b)$ 
  - $T(R)=5000$
  - $V(R, a)=50$
  - $V(R, b)=100$
- $S(b, c)$ 
  - $T(S)=2000$
  - $V(S, b)=200$
  - $V(S, c)=100$

## Ví dụ (tt)



(1)



(2)

## Ví dụ (tt)

- ☐ Cộng kích thước sau khi thực hiện các phép toán, ngoại trừ
  - Các nút lá
  - Nút gốc

☐ (1):  $100 + 50 + 1000 = 1150$

☐ (2):  $100 + 1000 = 1100$

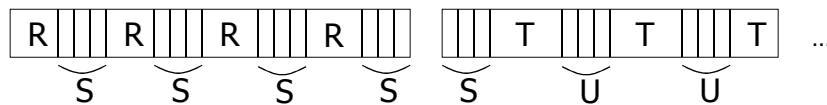
- ☐ Phép lược bỏ trùng lặp thực hiện sau thì tốt hơn

## Ước lượng số lần truy xuất IOs

- Các tham số thống kê
  - $B(R)$ : tổng số block chứa tất cả các bộ của  $R$
  - $f(R)$ : số bộ tối đa trong mỗi block
  - $M$ : số block trống trên bộ nhớ
- Quan tâm
  - Quan hệ  $R$  có được gom thành cụm không (clustered)?
  - Thuộc tính trong các phép toán có chỉ mục không (index)?
  - Chỉ mục có gom cụm không (clustering index)?
  - Kết quả cần được sắp thứ tự không?

## Clustering

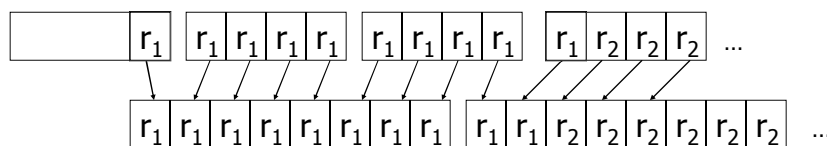
- Clustered-file



- Clustered relation



- Clustering index



## Ví dụ

- $R_1 \bowtie R_2$ 
  - $T(R_1) = 10000$
  - $T(R_2) = 5000$
  - $S(R_1) = S(R_2) = 1/10$  block
  - $M=101$  blocks
- Số block được đọc (bỏ qua việc ghi) để thực hiện phép kết tự nhiên trên là bao nhiêu?

## Nhận xét

- Ước lượng số lần truy xuất IOs không là cách tốt nhất
  - Bỏ qua chi phí CPU
  - Bỏ qua tham số thời gian
  - Xét trường hợp M đủ hoặc thiếu



