Elaborazione delle Query

Leggere le sezioni 8.1 e 8.2 di Riguzzi
et al. Sistemi Infomativi

Lucidi derivati da quelli di Hector Garcia-Molina

Query Processing

Come passare da una query ad un piano per eseguirla?

Esempio

Select B,D

From R,S

Where R.A = “c” $\wedge$ S.E = 2 $\wedge$ R.C=S.C

R	A	B	C
	a	1	10
	b	1	20
	c	2	10
	d	2	35
	e	3	45

S	C	D	E
	10	x	2
	20	y	2
	30	z	2
	40	x	1
	50	y	3

Risposta

B	D
2	x

- Come eseguire la query?

Un’idea

- Esegui prodotto cartesiano
- Seleziona le tuple
- Fai una proiezione

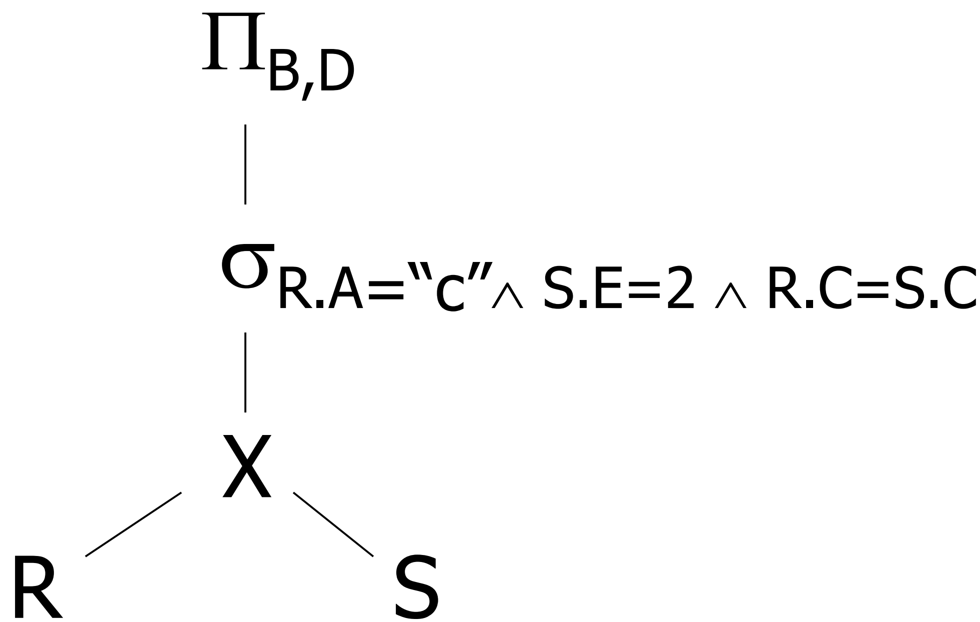
RXS

R.A	R.B	R.C	S.C	S.D	S.E
a	1	10	10	x	2
a	1	10	20	y	2
.					
.					
C	2	10	10	x	2
.					
.					

Trotvata!

Algebra relazionale- puo' essere usata per descrivere i piani.

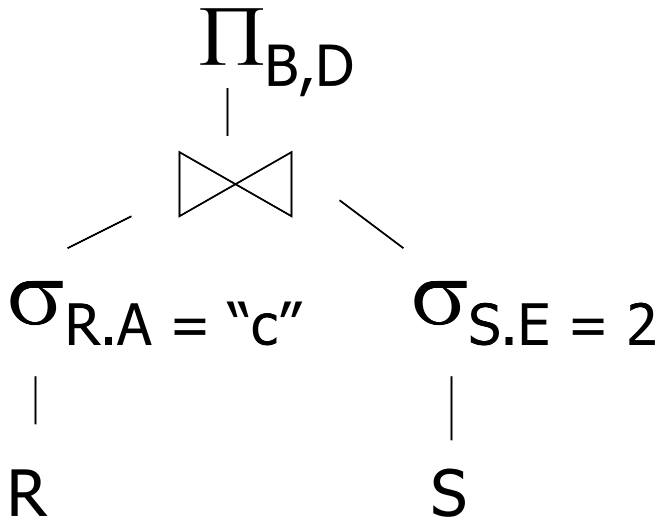
Es: Piano I



Oppure: $\Pi_{B,D}\,[\,\sigma_{R.A=\text{"c"} \wedge S.E=2 \wedge R.C = S.C}\,(RXS)]$

Un’altra idea:

Piano II



$\bowtie$

Join naturale

R

A	B	C
a	1	10
b	1	20
c	2	10
d	2	35
e	3	45

σ (R)

A	B	C
c	2	10

σ(S)

C	D	E
10	x	2
20	y	2
30	z	2

S

C	D	E
10	x	2
20	y	2
30	z	2
40	x	1
50	y	3

$\bowtie$

Piano III

Usa indici su R.A e S.C

(1) Usa l’indice su R.A per selezionare le tuple di R con R.A = “c”

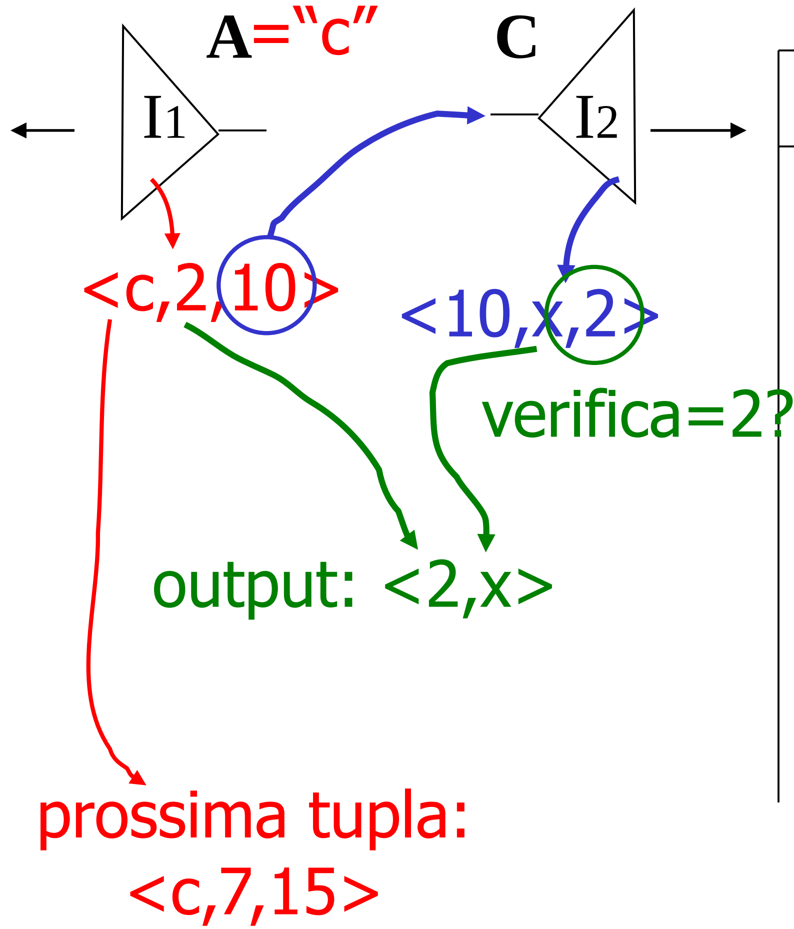
(2) Per ciascun valore R.C trovato, usa l’ndice su S.C per trovare le tuple corr.

(3) Elimina le tuple di S con S.E $\neq$ 2

(4) Esegui il join tra le tuple corrispodenti di R,S, proietta sugli attributi B,D e metti nel risultato

R

A	B	C
a	1	10
b	1	20
c	2	10
d	2	35
e	3	45



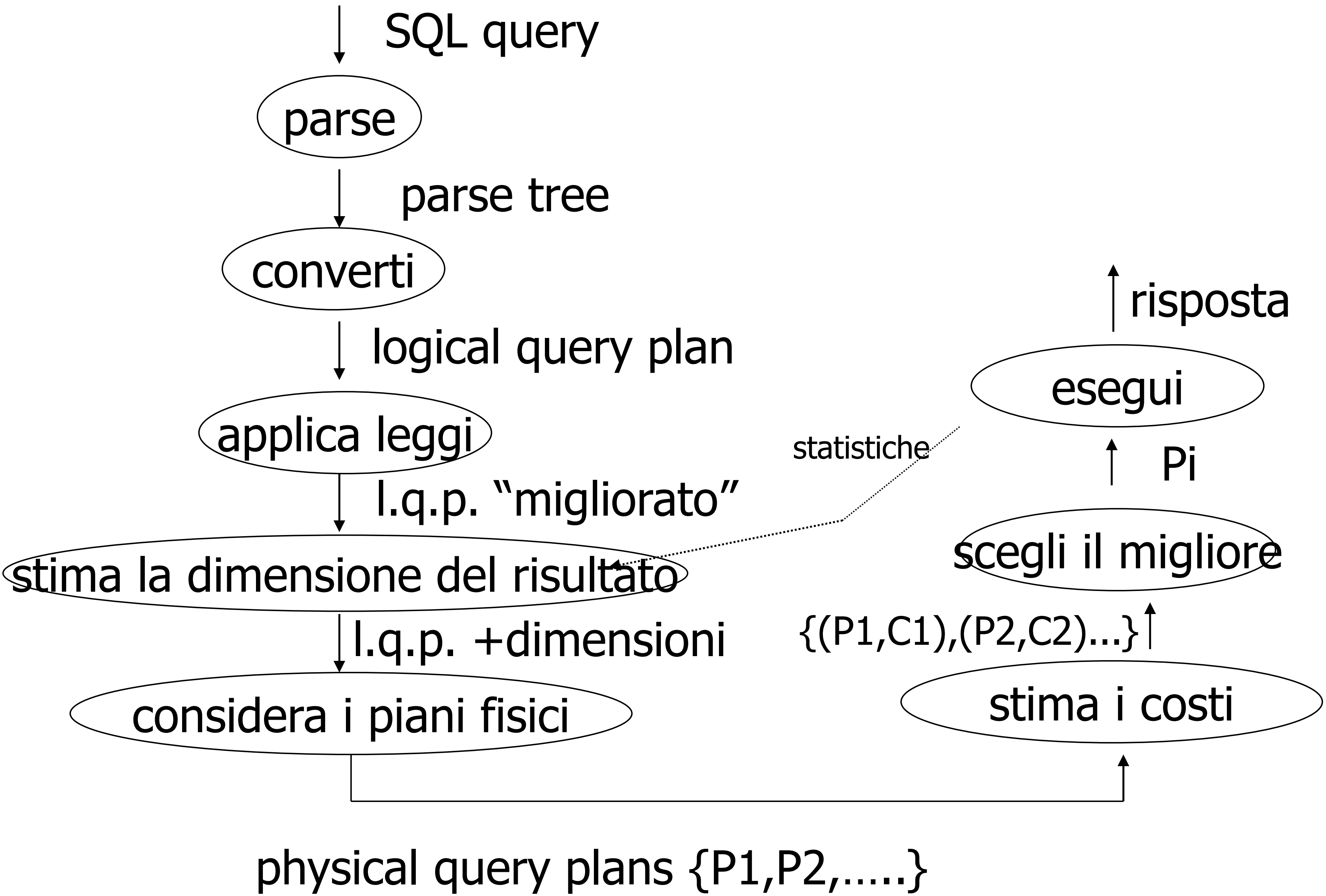
S

C	D	E
10	x	2
20	y	2
30	z	2
40	x	1
50	y	3

Ottimizzazione delle query

Piano descritto con l’algebra relazionale=logical query plan

Piano descritto con operatori fisici=physical query plan


SQL query
parse
parse tree
converti
logical query plan
applica leggi
l.q.p. “migliorato”
stima la dimensione del risultato
statistiche
l.q.p. +dimensioni
considera i piani fisici
physical query plans {P1,P2,…..}
stima i costi
{(P1,C1),(P2,C2)...}
scegli il migliore
Pi
esegui
risposta

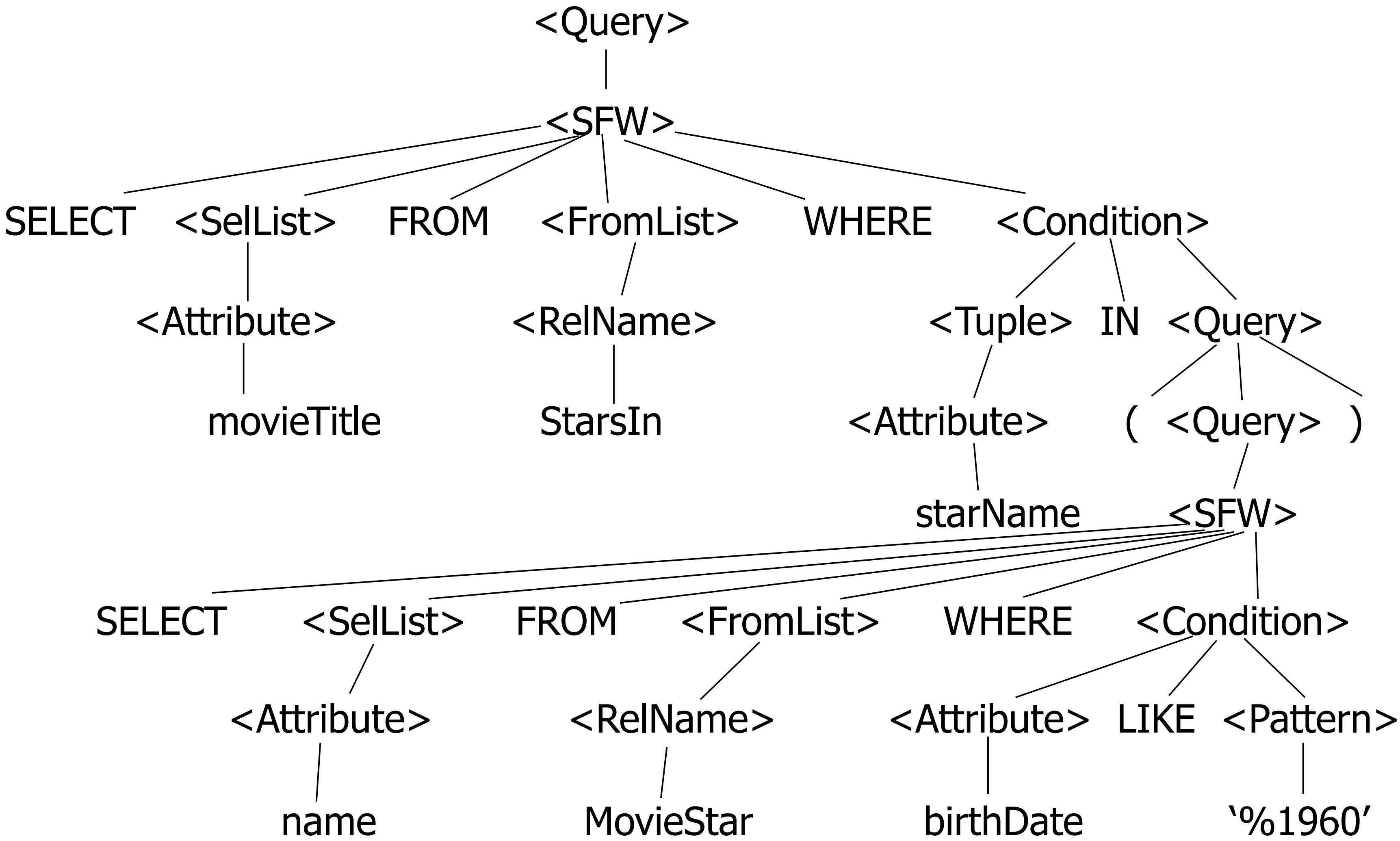
Esempio: SQL query

StarsIn(movieTitle,movieYear,starName)
MovieStar(name,address,gender,birthdate)

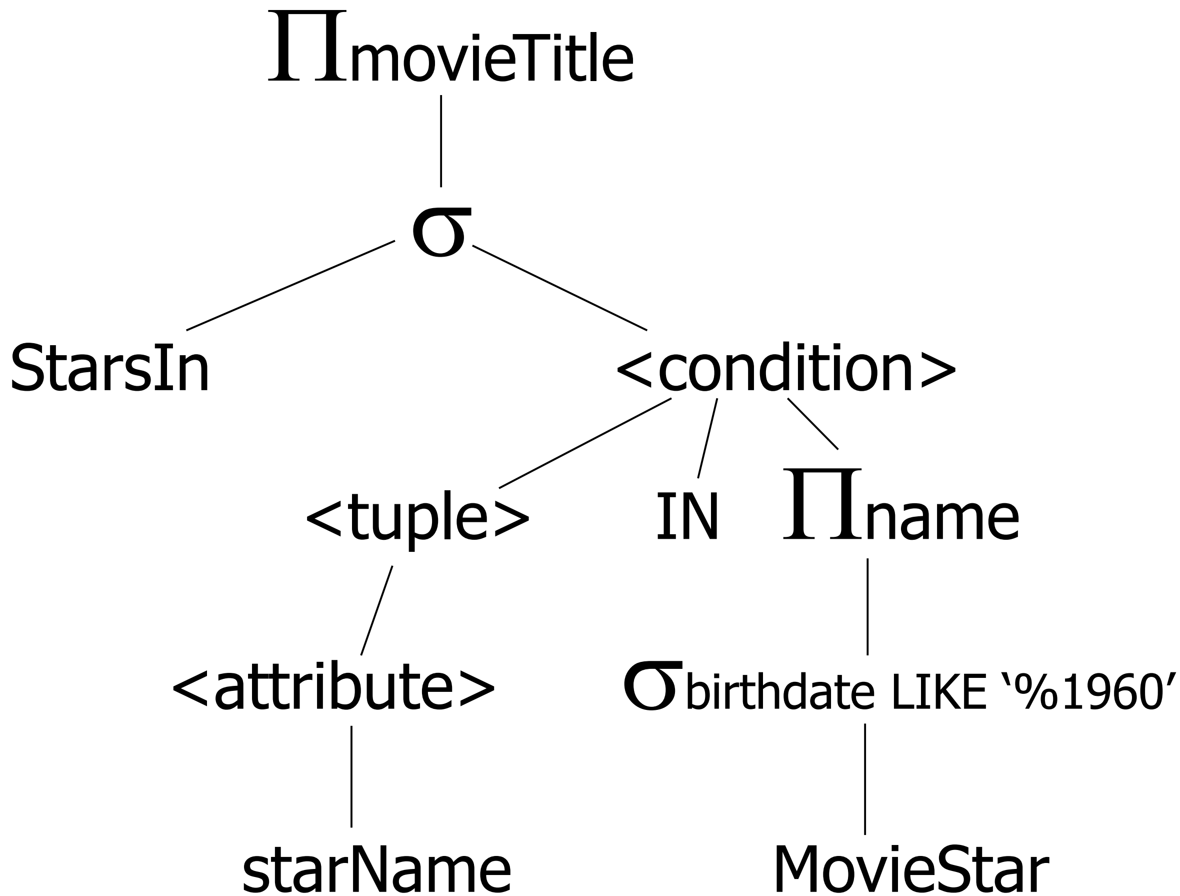
```
SELECT movieTitle
FROM StarsIn
WHERE starName IN (
        SELECT name
        FROM MovieStar
        WHERE birthdate LIKE '%1960'
);
```

(Trova i film con star nate nel 1960)

Esempio: Parse Tree

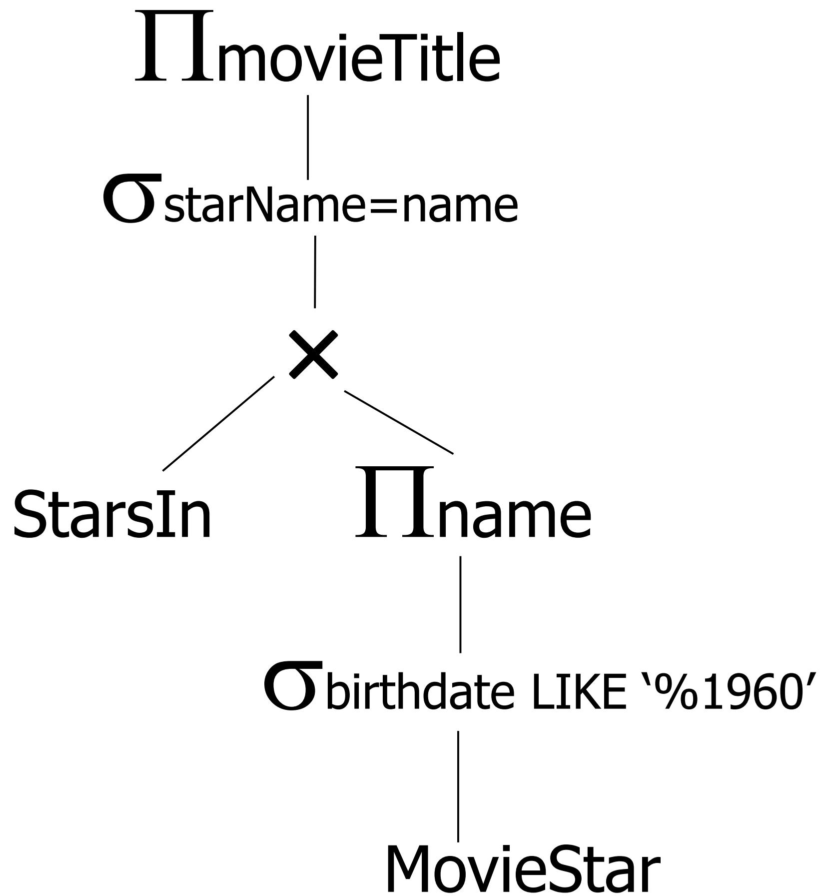


Esempio: Generazione dell'algebra relazionale



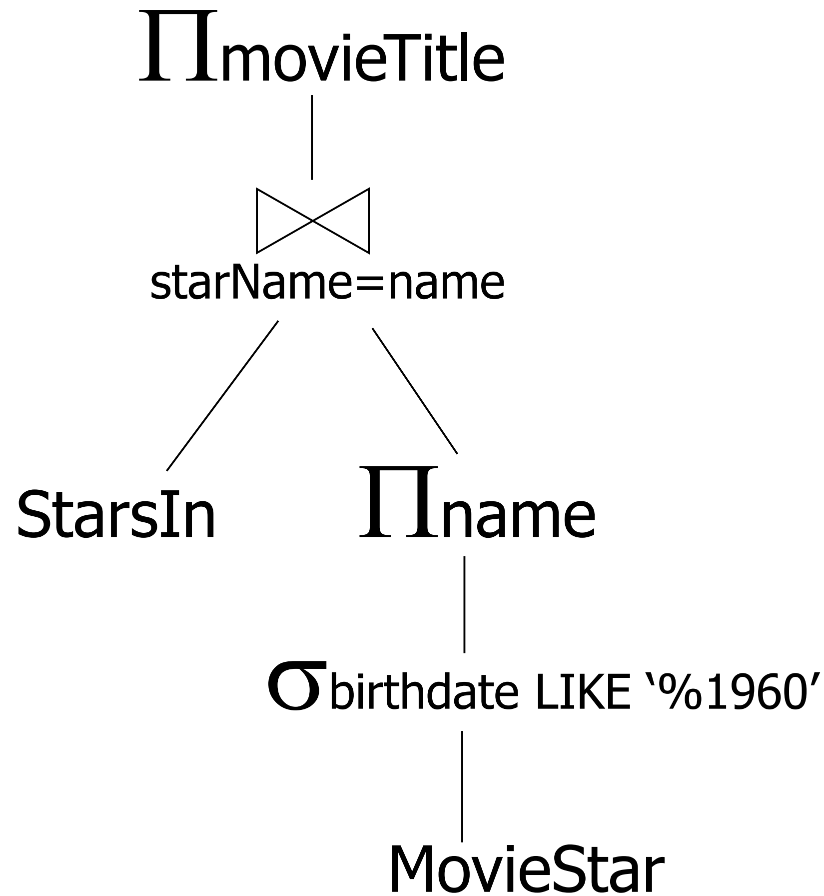
Il σ in questo caso si chiama "selezione a due argomenti" ed e' a mezza via tra il parse tree e l'algebra relazionale

Esempio: Piano logico della query



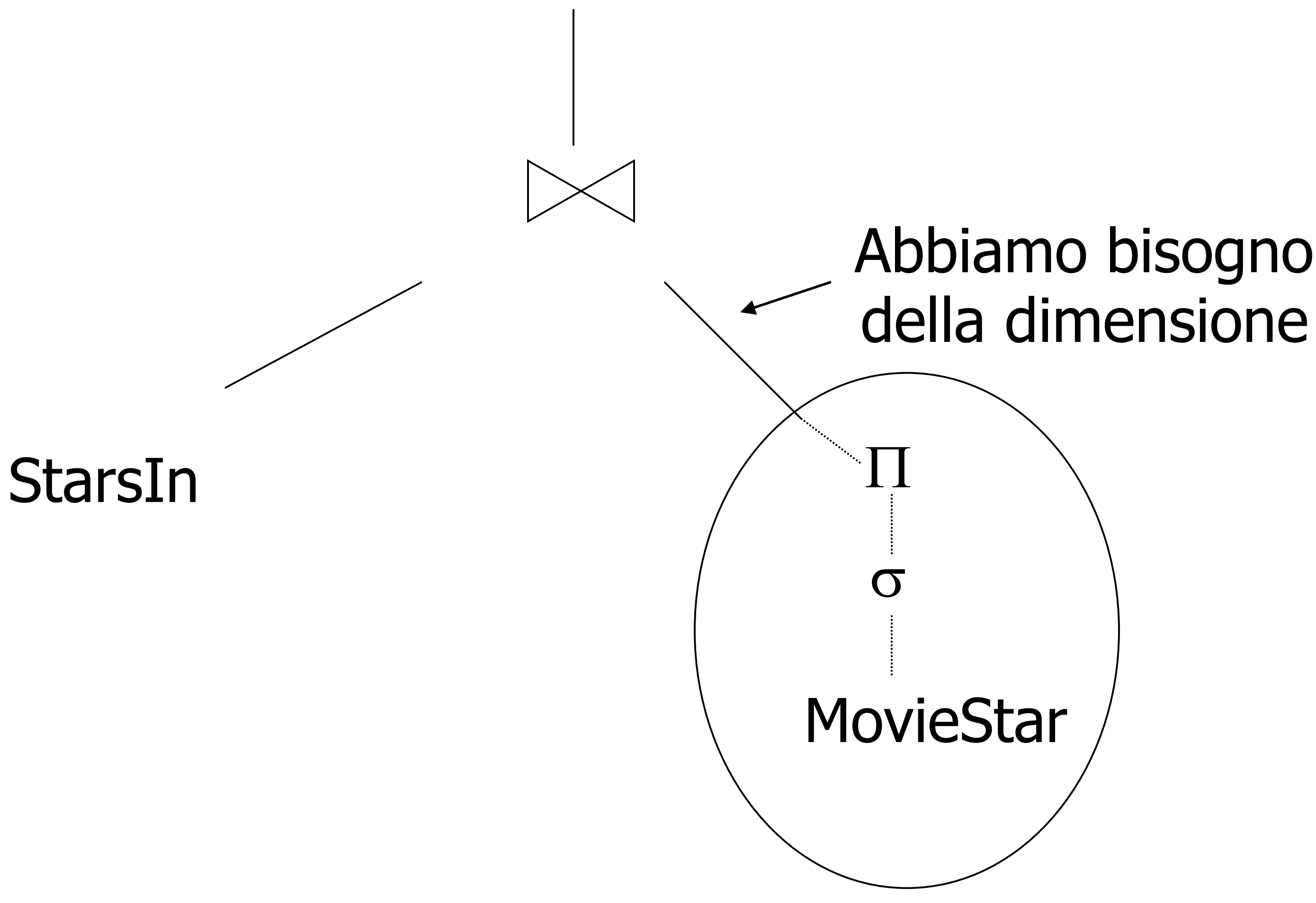
Trasformando la condizione IN

Esempio: Piano logico migliorato

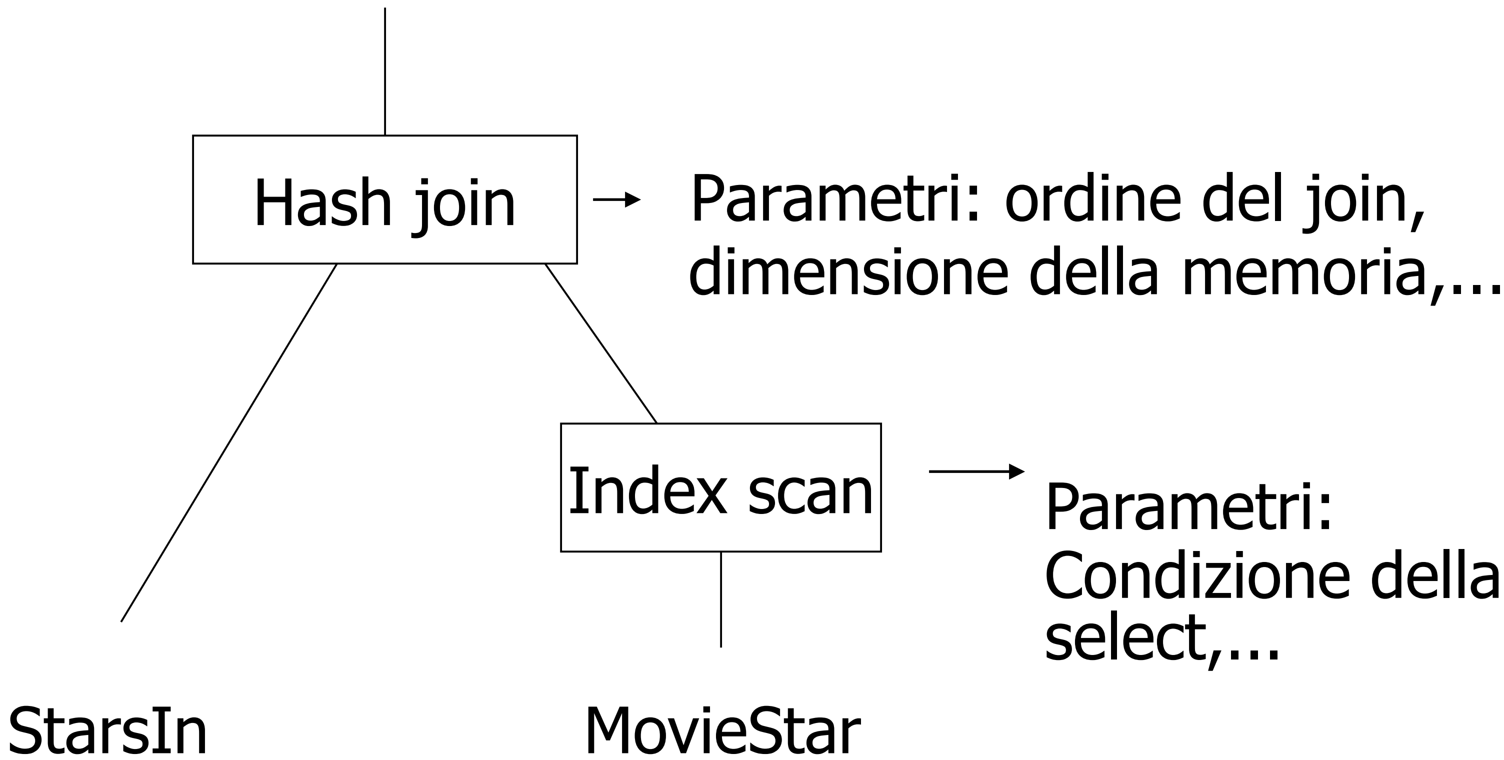


Un miglioramento sul piano precedente

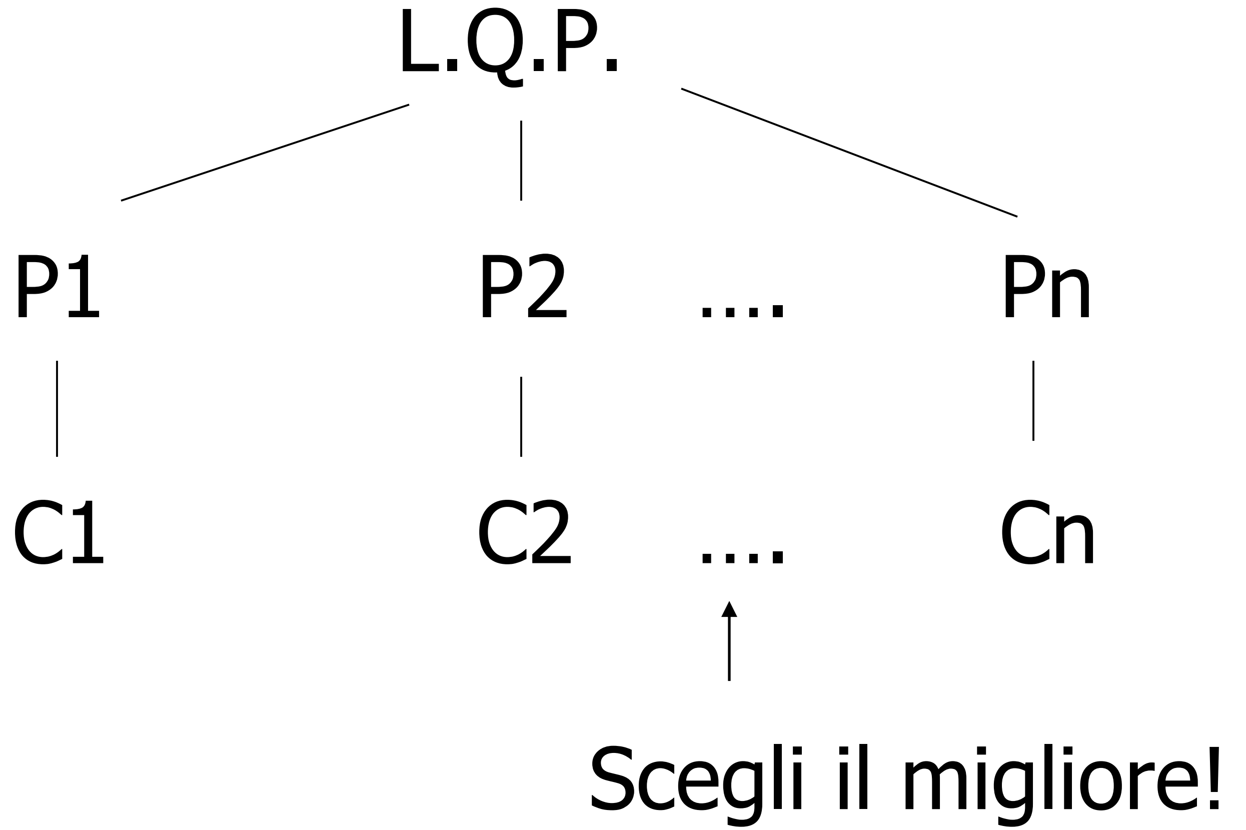
<u>Esempio:</u> Stima la dimensione del risultato



<u>Esempio:</u> Un piano fisico



Esempio: Stima i costi



Ottimizzazione delle Query

- Al livello dell'algebra relazionale
- Al livello del piano di query fisico
  - Genera i piani
  - Stima i costi, considerando i possibili algoritmi
    - Algoritmi di join
    - Uso o meno degli indici

Ottimizzazione dell’algebra relazionale

- Regole di trasformazione che preservano l’equivalenza

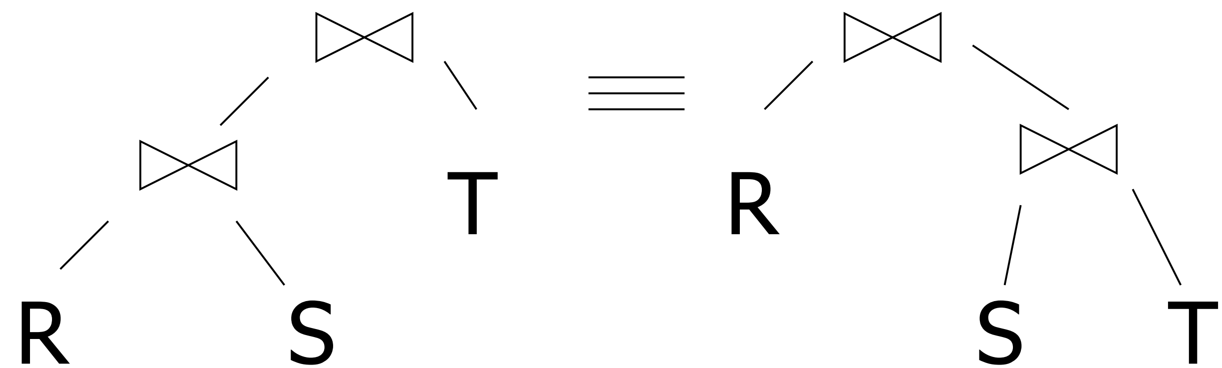
Regole: join naturale & prodotto cartesiano & unione & intersezione

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

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

Note:

- Dato che i nomi degli attributi sono contenuti nel risultato l'ordine degli operandi non e' importante
- Possiamo scriverli anche come alberi, ad es,:



Regole: join naturale & prodotto cartesiano & unione & intersezione

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

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

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

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

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

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

$$R \cap S = S \cap R$$

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

Regole: selezioni

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

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

La seconda regola funziona solo se R e' un insieme e non un bag

Esempio:

$R = \{a,b,c\}$

$S = \{b,c,d\}$

$R \cup_S S = \{a,b,c,d\}$

$R \cup_B S = \{a,b,b,c,c,d\}$

$R = \{a,a,b,b,b,c\}$

$S = \{b,b,c,c,d\}$

$R \cup_S S = \{a,b,c,d\}$

$R \cup_B S = \{a,a,b,b,b,b,b,c,c,c,d\}$

$\sigma_{p1 \mathbf{v} p2}(R) = \sigma_{p1}(R) \cup_S \sigma_{p2}(R)$

<u>Esempio:</u> R={a,a,b,b,b,c}

P1 soddisfatto da a,b; P2 soddisfatto da b,c

$\sigma_{p1 \mathbf{v} p2}(R) = \{a,a,b,b,b,c\}$

$\sigma_{p1}(R) = \{a,a,b,b,b\}$

$\sigma_{p2}(R) = \{b,b,b,c\}$

$\sigma_{p1}(R) \cup_S \sigma_{p2}(R) = \{a,b,c\}$

$\sigma_{p1}(R) \cup_B \sigma_{p2}(R) = \{a,a,b,b,b,b,b,b,c\}$

Regole $\sigma, \cup, -$ combinati:

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

$$\sigma_p(R - S) = \sigma_p(R) - S = \sigma_p(R) - \sigma_p(S)$$

<u>Regole:</u> $\sigma + \bowtie$ combinati

Siano p = predicati con solo attr. di R

q = predicati con solo attr. di S

m = predicati con solo attr. di R,S

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

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

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

Regole: $\sigma + \bowtie$ combinati (continua)

Alcune regole possono essere derivate:

$\sigma_{p \wedge q} (R \bowtie S) =$

$\sigma_{p \wedge q \wedge m} (R \bowtie S) =$

$\sigma_{p \vee q} (R \bowtie S) =$

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

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

$$\sigma_{p \vee q} (R \bowtie S) =$$

$$\left[(\sigma_p R) \bowtie S \right] \cup_S \left[R \bowtie (\sigma_q S) \right]$$

Solo se $(\sigma_p R) \bowtie S$ e $R \bowtie (\sigma_q S)$ sono insiemi e non bag

--> Derivazione per la prima:

$$\sigma_{p \wedge q} (R \bowtie S) =$$

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

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

$$[\sigma_{p} (R)] \bowtie [\sigma_{q} (S)]$$

Regole per σ, combinati con X

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

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

<u>Regole:</u> proiezione

Siano: X = insieme di attributi

Y = insieme di attributi

$XY = X \cup Y$

$$\pi_{xy}(R) = \cancel{\pi_x[\pi_y(R)]}$$

Regole: π, σ combinate

Sia x = sottoinsieme degli attributi di R

z = attributi nel predicato P

(sottoinsieme degli attributi di R)

$$\pi_x[\sigma_p(R)] = \pi_x\{\sigma_p[\cancel{\pi_x}^{\pi_{xz}}(R)]\}$$

Regole: $\pi, \bowtie$ combinati

Sia x = sottoinsieme degli attributi di R

y = sottoinsieme degli attributi di S

z = intersezione degli attributi di R,S

$$\pi_{xy}(R \bowtie S) =$$

$$\pi_{xy}\{[\pi_{xz}(R)] \bowtie [\pi_{yz}(S)]\}$$

$$\pi_{xy}\{\sigma_{p}\,(R \bowtie S)\} \;=$$

$$\pi_{xy}\{\sigma_{p}\,[\pi_{xz'}(R) \bowtie \pi_{yz'}(S)]\}$$

$$z' = z \cup \{\text{attributi usati in P}\}$$

Quali sono ‘buone’ trasformazioni?

- $\sigma_{p1 \wedge p2}(R) \rightarrow \sigma_{p1}[\sigma_{p2}(R)]$
- $\sigma_{p}(R \bowtie S) \rightarrow [\sigma_{p}(R)] \bowtie S$
- $R \bowtie S \rightarrow S \bowtie R$
- $\pi_{x}[\sigma_{p}(R)] \rightarrow \pi_{x}\{\sigma_{p}[\pi_{xz}(R)]\}$

Normalmente: esegui le proiezioni prima possibile

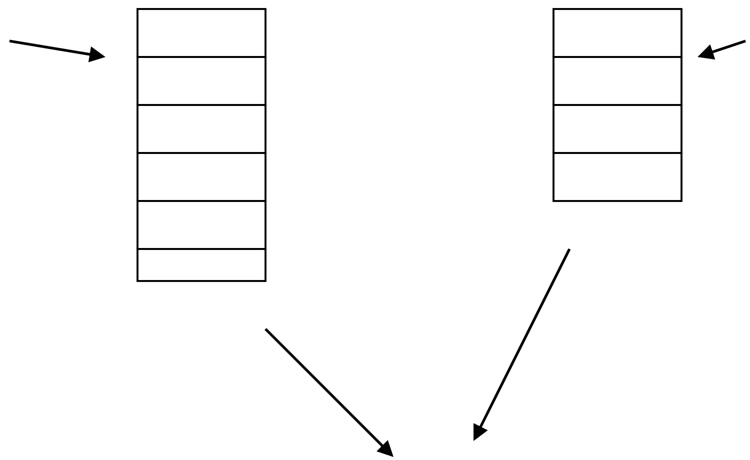
Esempio: $R(A,B,C,D,E)$ $\quad x=\{E\}$

$P: (A=3) \wedge (B=\text{“cat”})$

$\pi_x \{\sigma_p (R)\}$ rispetto. $\pi_E \{\sigma_p\{\pi_{ABE}(R)\}\}$

$\boxed{\text{Ma}}$ se A e B hanno indici?

B = “cat” A=3



Interseca i puntatori per ottenere i puntatori alle tuple richieste

In conclusione:

- Non sempre conviene anticipare le proiezioni

Regola per le selezioni

- Normalmente e' buono anticipare le selezioni, cioe' spingerle in basso nell'albero

Eccezione

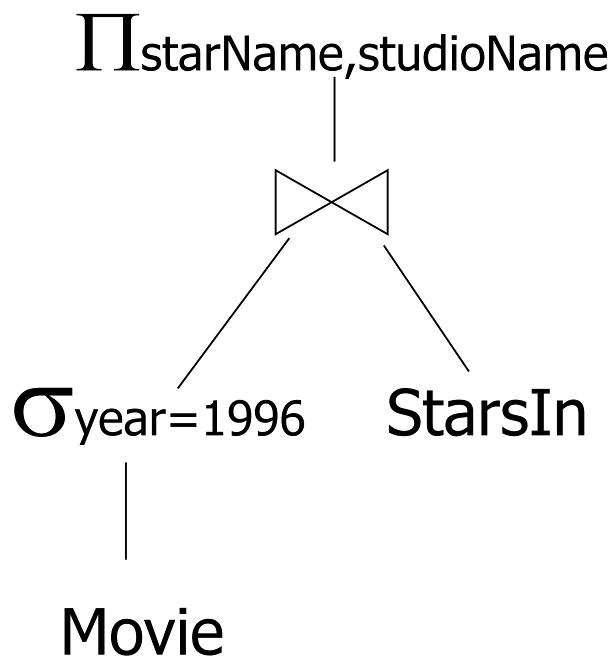
- StarsIn(title, year, starName)
- Movie(title,year,length, inColor,studioName, producerC#)

CREATE VIEW MovieOf1996 AS
    SELECT * FROM Movie WHERE
    year=1996

Eccezione

SELECT starName,
studioName

FROM MoviesOf1996 m
JOIN StarsIn s ON
m.title=s.title AND
m.year=s.year



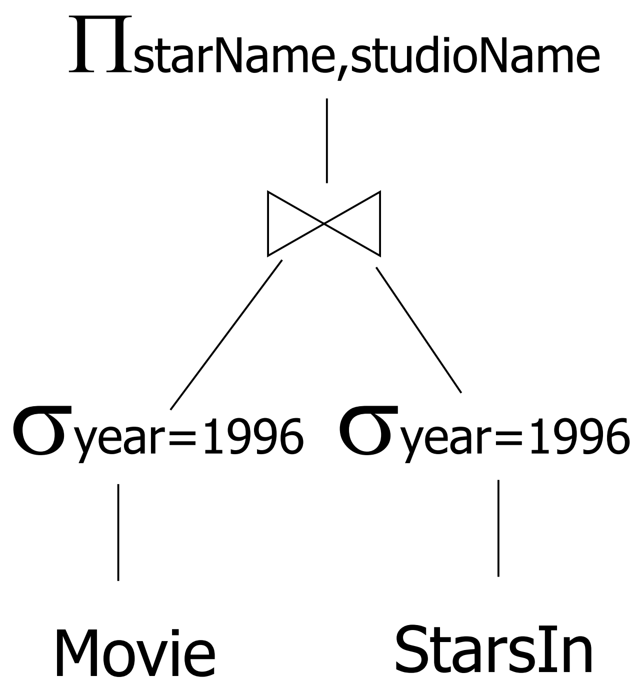
Eccezione

- Si usa la regola

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

- da destra verso sinistra e si porta la selezione sopra il join
- A questo punto, dato che year e’ anche attributo di StarsIn e il join contiene year, si puo’ spingere la selezione ad entrambi i figli del join

Eccezione



- Quindi in alcuni casi occorre far salire le selezioni il piu' possibile prima di farle scendere

Schema - Query Processing

- Livello dell’algebra relazionale (logical query plan)
  - Buone trasformazioni
- Livello del piano di query fisico
  - Stima dei costi
  - Genera e confronta i piani fisici dai piani logici

- Stima del costo di un piano di query

(1) Stima della dimensione dei risultati

(2) Stima del numero di IOs

Stima della dimensione del risultato

- Mantieni le statistiche per la relazione R
  - T(R) : # tuple in R
  - S(R) : # di bytes in ciascuna tupla di R
  - B(R): # di blocchi per tenere tutte le tuple di R
  - V(R, A) : # di valori distinti in R per l'attributo A

Esempio

R

A	B	C	D
cat	1	10	a
cat	1	20	b
dog	1	30	a
dog	1	40	c
bat	1	50	d

A: stringa di 20 byte

B: intero di 4 byte

C: data di 8 byte

D: stringa di 5 byte

$T(R) = 5 \qquad S(R) = 37$

$V(R,A) = 3 \qquad V(R,C) = 5$

$V(R,B) = 1 \qquad V(R,D) = 4$

Stima del numero di blocchi

- Ipotesi:
  - Record impaccati
  - Tutto lo spazio nei blocchi occupato da record (no strutture ausiliarie)

$B(R)=T(R)*S(R)/B$

Dove B e’ la dimensione in bytes del blocco

Stima della dimensione per W = R1 x R2

$T(W) = \quad T(R1) \times T(R2)$

$S(W) = \quad S(R1) + S(R2)$

<u>Stima della dimensione</u> per $W = \sigma_{A=a}(R)$

$S(W) = S(R)$

$T(W) = ?$

Esempio

R

A	B	C	D
cat	1	10	a
cat	1	20	b
dog	1	30	a
dog	1	40	c
bat	1	50	d

V(R,A)=3
V(R,B)=1
V(R,C)=5
V(R,D)=4

$W = \sigma_{z=val}(R) \quad T(W) = T(R)/V(R,Z)$

$W = \sigma_{A=val}(R) \quad T(W) = T(R)/3$

Assunzione:

I valori per l’attributo Z sono uniformemente distribuiti sui possibili V(R,Z) valori

Assunzione alternativa:

I valori per l’attributo Z sono uniformemente distribuiti sul dominio con DOM(R,Z) valori.

Esempio

R

A	B	C	D
cat	1	10	a
cat	1	20	b
dog	1	30	a
dog	1	40	c
bat	1	50	d

Assunzione alternativa

V(R,A)=3 DOM(R,A)=10

V(R,B)=1 DOM(R,B)=10

V(R,C)=5 DOM(R,C)=10

V(R,D)=4 DOM(R,D)=10

$$W = \sigma_{z=val}(R) \quad T(W) = \frac{T(R)}{DOM(R,Z)}$$

$C=val \Rightarrow T(W) = 5/10=0.5$

$B=val \Rightarrow T(W)= 5/10=0.5$

$A=val \Rightarrow T(W)= 5/10= 0.5$

Cardinalita’ della selezione

SC(R,A) = # medio di record che soddisfano una condizione di uguaglianza su R.A

$$SC(R,A) = \begin{cases} \dfrac{T(R)}{V(R,A)} \\ \\ \dfrac{T(R)}{DOM(R,A)} \end{cases}$$

Caso $\quad W = \sigma_{z \geq val}(R)$?

$T(W) = ?$

- Soluzione # 1:

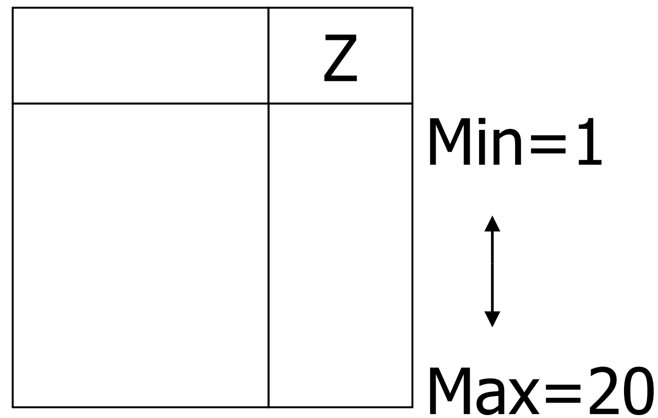
  $T(W) = T(R)/2$

- Soluzione # 2:

  $T(W) = T(R)/3$

- Soluzione # 3:   stima i valori nel range

<u>Esempio</u> R



$W = \sigma_{z \geq 15}(R)$

$$f = \frac{20-15+1}{20-1+1} = \frac{6}{20}$$ (frazione del range)

$T(W) = f \times T(R)$

Stima della dimensione per

$$W = R1 \bowtie R2$$

Sia X = attributi di $R1$

Y = attributi di $R2$

$\boxed{\text{Caso 1}}$

$$X \cap Y = \emptyset$$

Lo stesso di $R1 \times R2$

Caso 2 $\quad W = R1 \bowtie R2 \qquad X \cap Y = A$

R1

A	B	C

R2

A	D

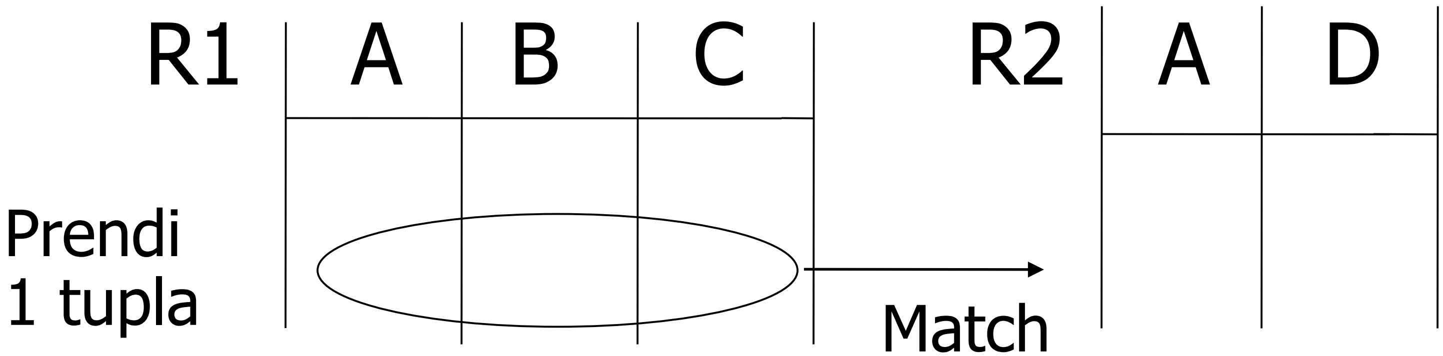
Assunzione:

$V(R1,A) \leq V(R2,A)$ e ogni valore per A in R1 e' in R2

$V(R2,A) \leq V(R1,A)$ e ogni valore per A in R2 e' in R1

"inclusione dell'insieme dei valori"

Calcolare T(W) quando $V(R1,A) \leq V(R2,A)$



1 tupla corrisponde a $\dfrac{T(R2)}{V(R2,A)}$ tuple...

Quindi $T(W) = \dfrac{T(R2)}{V(R2, A)} \times T(R1)$

- $V(R1,A) \leq V(R2,A) \quad T(W) = \dfrac{T(R2)\,T(R1)}{V(R2,A)}$

- $V(R2,A) \leq V(R1,A) \quad T(W) = \dfrac{T(R2)\,T(R1)}{V(R1,A)}$

[A e' l'attributo comune]

In generale $W = R1 \bowtie R2$

$$T(W) = \frac{T(R2)\, T(R1)}{\max\{ V(R1,A), V(R2,A) \}}$$

In tutti i casi:

$$S(W) = S(R1) + S(R2) - S(A)$$

dimensione dell'attributo A

Join naturale con attributi di join multipli

R1(x,y1,y2), R2(y1,y2,z)

W=R1$\triangleright\triangleleft$ R2

Date due tuple r1 di R1 ed r2 di R2, qual'e' la probabilita' che facciano join?

Prob che r1 ed r2 abbiano uguale y1 =1/max{V(R1,y1),V(R2,y1)}

Prob che r1 ed r2 abbiano uguale y2 =1/max{V(R1,y2),V(R2,y2)}

Attributi di join multipli

- Quindi, di T(R1)*T(R2) tuple, quelle che fanno match sono

$$\frac{T(R1)T(R2)}{\max\{V(R1,y1),V(R2,y1)\}\max\{V(R1,y2),V(R2,y2)\}}$$

Esempio

- $W=R(a,b,c) \bowtie_{R.b=S.d \text{ and } R.c=S.e} S(d,e,f)$
- T(R)=1000 T(S)=2000
- V(R,b)=20 V(S,d)=50
- V(R,c)=100 V(S,e)=50
- T(W)=1000*2000/(50*100)=400

Stima delle dimensioni

- $S = \sigma_{A=a \wedge B=b}(R)$
- $T(S)=T(R)/(V(R,A)xV(R,B))$
- $Q = \sigma_{A=a \text{ OR } B=b}(R)$
- $T(Q)=\min\{T(R),T(\sigma_{A=a}(R))+T(\sigma_{B=b}(R))\}$

Nota: per espressioni complesse, e' necessario calcolare i risultati T,S,V intermedi.

Es. $W = \underbrace{[\sigma_{A=a}(R1)]}_{\text{relazione } U} \bowtie R2$

$T(U) = T(R1)/V(R1,A) \qquad S(U) = S(R1)$

Quanto vale V (U, *) ?

Per stimare Vs

Es., $U = \sigma_{A=a}(R1)$
supponendo che R1 abbia gli attributi A,B,C,D

$V(U, A) = 1$

$V(U, B) = V(R1, B)$

$V(U, C) = V(R1, C)$

$V(U, D) = V(R1, D)$

Per i join $U = R1(A,B) \rhd\lhd R2(A,C)$

$V(U,A) = \min \{ V(R1, A), V(R2, A) \}$

$V(U,B) = V(R1, B)$

$V(U,C) = V(R2, C)$

[assunzione “conservazione dell’insieme dei valori”]

Esempio:

$$Z = R1(A,B) \bowtie R2(B,C) \bowtie R3(C,D)$$

$\boxed{R1}$	$T(R1) = 1000$	$V(R1,A)=50$	$V(R1,B)=100$
$\boxed{R2}$	$T(R2) = 2000$	$V(R2,B)=200$	$V(R2,C)=300$
$\boxed{R3}$	$T(R3) = 3000$	$V(R3,C)=90$	$V(R3,D)=500$

Risultato parziale: $U = R1 \bowtie R2$

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

$V(U,A) = 50$

$V(U,B) = 100$

$V(U,C) = 300$

$Z = U \bowtie R3$

$$T(Z) = \frac{1000 \times 2000 \times 3000}{200 \times 300} = 100.000$$

$V(Z,A) = 50$

$V(Z,B) = 100$

$V(Z,C) = 90$

$V(Z,D) = 500$

Altro ordine

$W = R2(B,C) \bowtie R3(C,D)$

$$T(W) = \frac{2000 \times 3000}{300}$$

$V(W,B) = 200$

$Z = R1 \bowtie W$

$$T(Z) = \frac{2000 \times 3000 \times 1000}{300 \times 200} = 100.000$$

Osservazione

- Le assunzioni di inclusione dell'insieme dei valori e di conservazione dell'insieme dei valori garantiscono che la stima della dimensione del risultato sia la stessa indipendentemente dall'ordine con cui si eseguono i join

Altro esempio

$R(a,b,c) \bowtie S(b,c,d) \bowtie U(b,e)$

$T(R)=1000$	$T(S)=2000$	$T(U)=5000$
$V(R,a)=100$		
$V(R,b)=20$	$V(S,b)=50$	$V(U,b)=200$
$V(R,c)=200$	$V(S,c)=100$	
	$V(S,d)=400$	
		$V(U,e)=500$

Altro esempio

$W = R \bowtie S$

$$T(W) = \frac{1000 \times 2000}{50 \times 200}$$

$V(W,b) = 20$

$Z = W \bowtie U$

$$T(U) = \frac{1000 \times 2000 \times 5000}{50 \times 200 \times 200} = 5000$$

In generale

- Se abbiamo un join di n relazioni
- $S=R1 \rhd\lhd R2 \rhd\lhd .... \rhd\lhd Rn$
- T(S) si ottiene moltiplicando il numero di tuple in ogni relazione e, per ciascun attributo A, dividendo per tutti i V(Ri,A) tranne il piu' piccolo
- Se A appare una volta sola, V(Ri,A) non compare nella formula
- Si veda il libro per la dimostrazione

- Da non dimenticare:

  le statistiche devono essere mantenute aggiornate…

Schema

- Stima del costo di una query
  - Stima della dimensione dell'output fatto!
  - Stima del # di IOs prossimo…
- Generare e confrontare i piani