

27/10/2020

Basi di Dati + Laboratorio - Informatica Triennale - Corso A

268

Equivalenza di espressioni

- Due espressioni sono **equivalenti** se producono lo stesso risultato qualunque sia l'istanza attuale della base di dati
- L'equivalenza è importante in pratica perché i DBMS cercano di eseguire espressioni equivalenti a quelle date, ma meno "costose"

Basi di Dati + Laboratorio - Informatica Triennale - Corso A

270

Tipi di equivalenza

- Assoluta
 - $\pi_{AB}(\sigma_{A>0}(R)) \equiv \sigma_{A>0}(\pi_{AB}(R))$
- Dipendente dallo schema
 - $\pi_{AB}(R_1) \bowtie \pi_{AC}(R_2) \equiv_R \pi_{ABC}(R_1 \bowtie R_2)$
- esempio: $R_1(ABC)$ e $R_2(AC)$
 - le due espressioni di prima non sono equivalenti

Basi di Dati + Laboratorio - Informatica Triennale - Corso A

271

Atomizzazione di selezioni

$$\sigma_{A \wedge B}(E) \equiv \sigma_A(\sigma_B(E))$$

- E è una espressione. Questa trasformazione consente l'applicazione di successive trasformazioni che operano su selezioni con condizioni atomiche

Basi di Dati + Laboratorio - Informatica Triennale - Corso A

272

Idempotenza delle proiezioni

$$\pi_X(E) \equiv \pi_X(\pi_{XY}(E))$$

- Una proiezione può essere trasformata in una cascata di proiezioni che 'eliminano' i vari attributi in fasi diverse

Basi di Dati + Laboratorio - Informatica Triennale - Corso A

273

Anticipazione selezione risp. al Join

$$\sigma_F(E_1 \bowtie E_2) \equiv E_1 \bowtie \sigma_F(E_2)$$

cond F su attributi di E_2

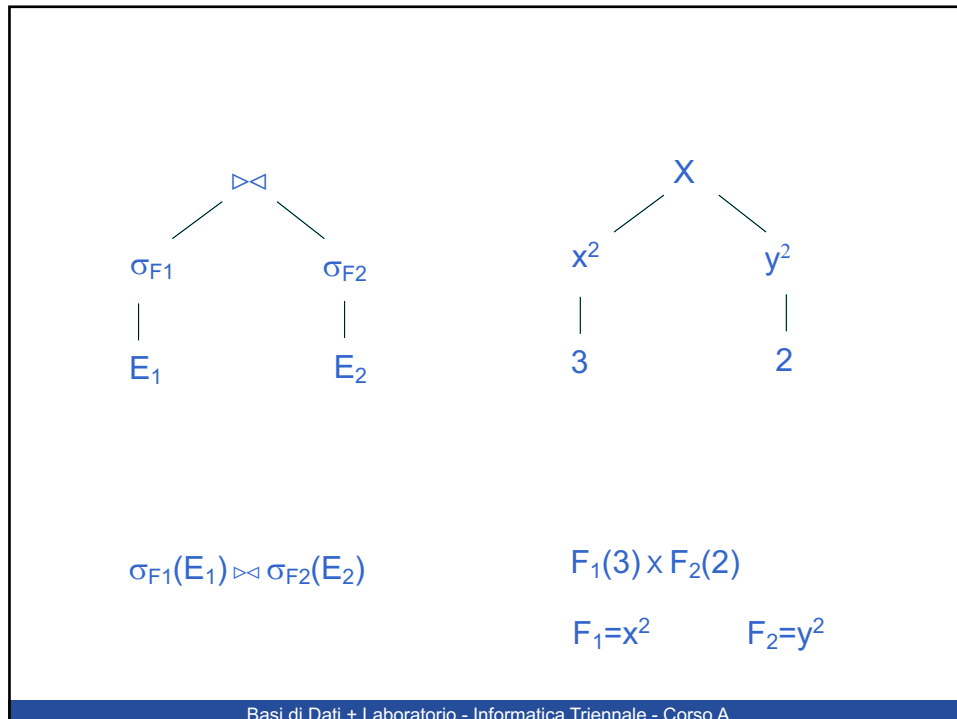
$$\sigma_{F_1 \wedge F_2}(E_1 \bowtie E_2) \equiv \sigma_{F_1}(E_1) \bowtie \sigma_{F_2}(E_2)$$

F_1 cond su attributi di E_1 e F_2 cond su attributi di E_2

- Detto anche *pushing selections down* riduce in modo significativo la dimensione del risultato intermedio (e quindi il costo dell'operazione)

Basi di Dati + Laboratorio - Informatica Triennale - Corso A

274



275

Eliminazione proiezioni superflue

$$\pi_Z(E) \equiv E$$

se Z sono tutti gli attributi di E

276

Anticipazione π rispetto al prodotto

$$\pi_{XY}(E_1 \times E_2) \equiv \pi_X(E_1) \times \pi_Y(E_2)$$

Basi di Dati + Laboratorio - Informatica Triennale - Corso A

277

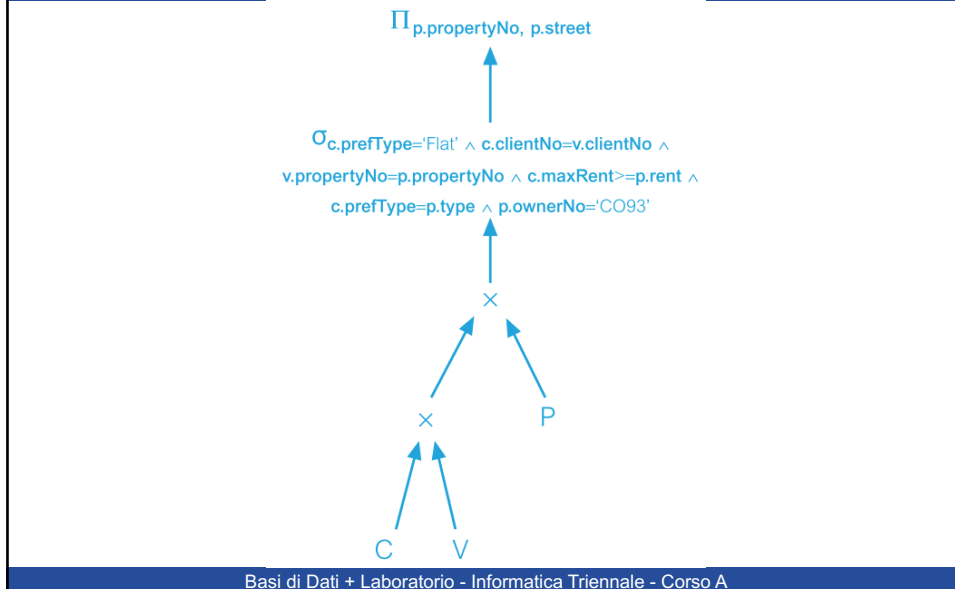
Possibile algoritmo di ottimizzazione

1. Si anticipa l'esecuzione delle selezioni sulle proiezioni (da sx verso dx)
2. Si raggruppano le selezioni
3. Si anticipa l'esecuzione delle selezioni sul prodotto (join)
4. Si ripete questi tre passi finché possibile
5. Si eliminano le proiezioni superflue
6. Si raggruppano le proiezioni
7. Se l'espressione è un prodotto si anticipa l'esecuzione delle proiezioni rispetto al prodotto

Basi di Dati + Laboratorio - Informatica Triennale - Corso A

278

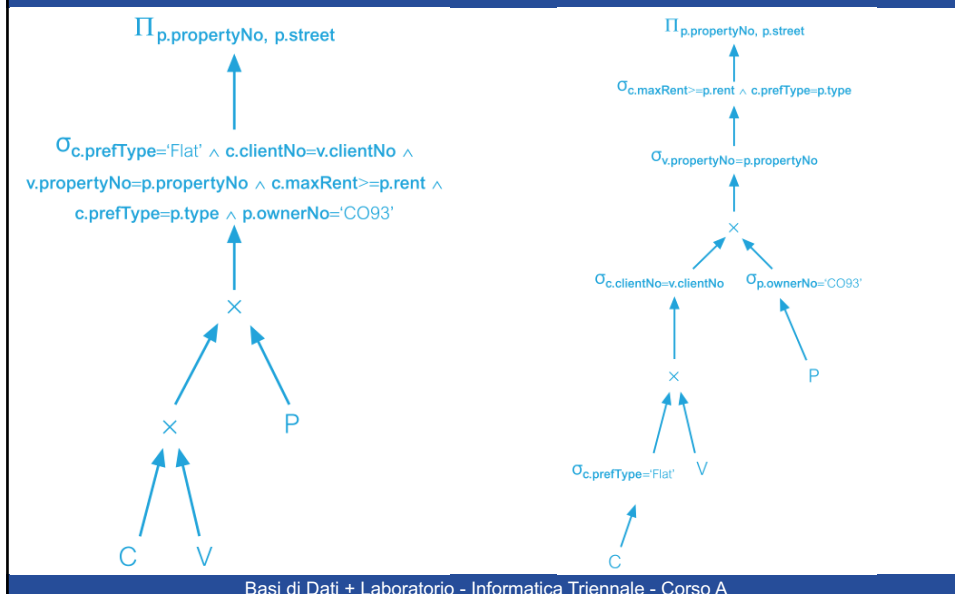
Esempio ottimizzazione



Basi di Dati + Laboratorio - Informatica Triennale - Corso A

279

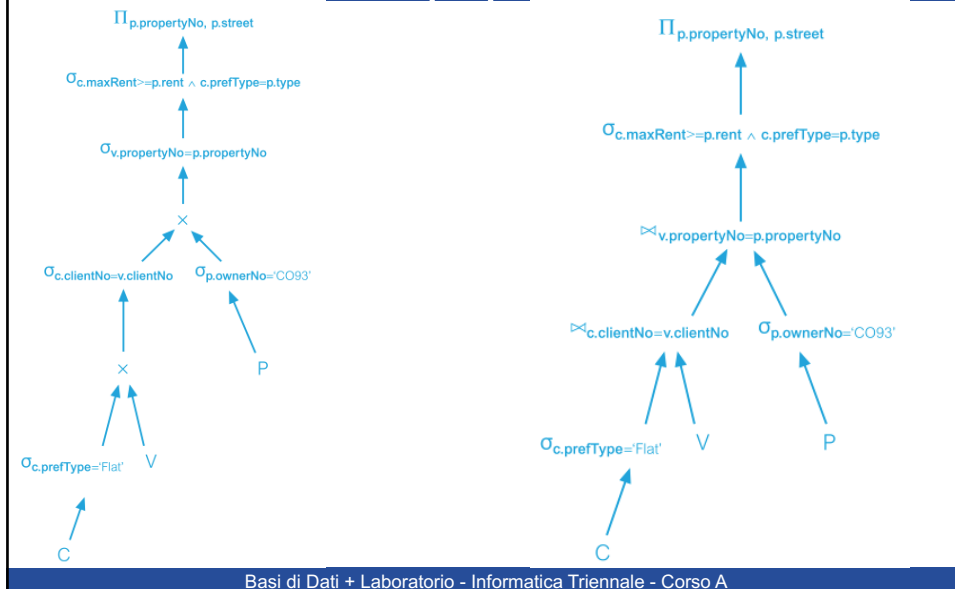
Pushing selections down



Basi di Dati + Laboratorio - Informatica Triennale - Corso A

280

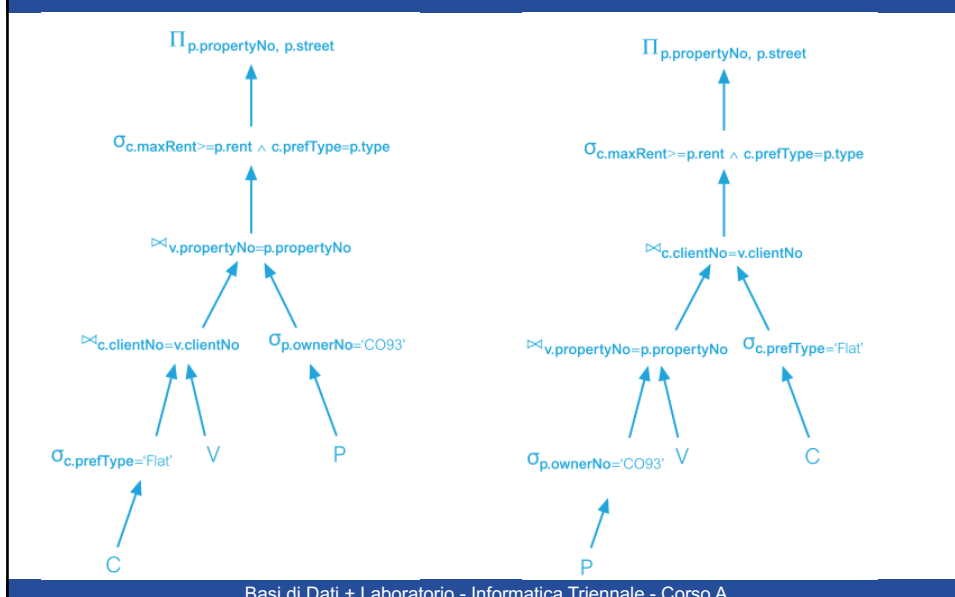
Selection/Cartesian products to Equijoins



Basi di Dati + Laboratorio - Informatica Triennale - Corso A

281

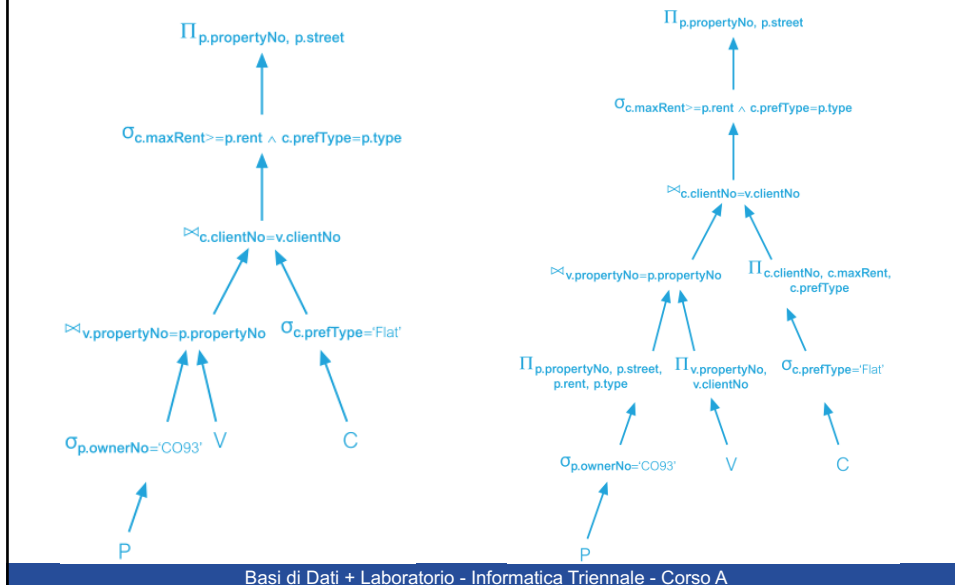
Using associativity of Equijoins



Basi di Dati + Laboratorio - Informatica Triennale - Corso A

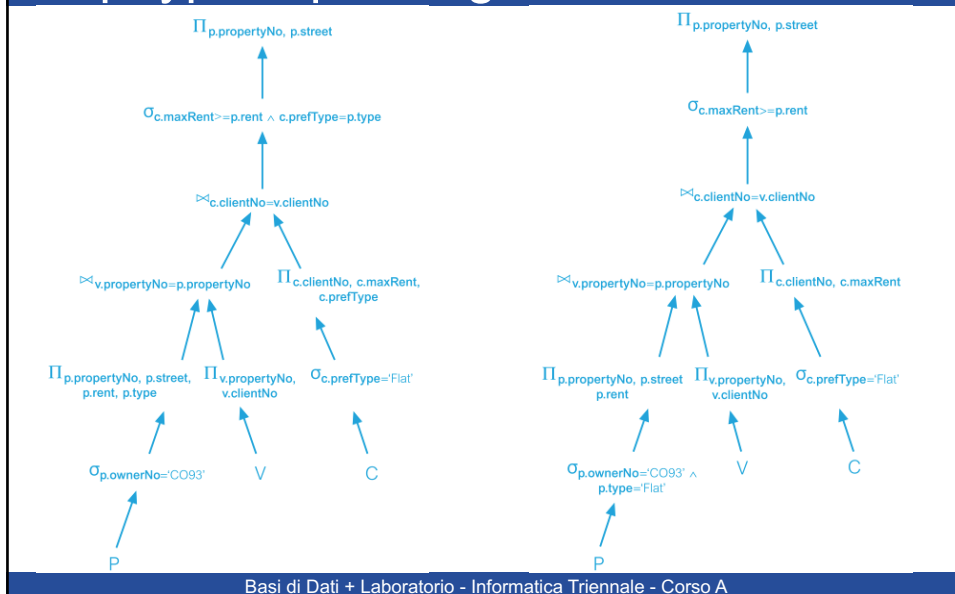
282

Pushing Projections down



283

c.prefType = 'Flat' in Selection on p.type & pushing Selection down



284

Selezione con valori nulli

Impiegati

Matricola	Cognome	Filiale	Età
7309	Rossi	Roma	32
5998	Neri	Milano	45
9553	Bruni	Milano	NULL

$\sigma_{\text{Età} > 40}(\text{Impiegati})$

- la condizione atomica è vera solo per valori non nulli

Basi di Dati + Laboratorio - Informatica Triennale - Corso A

285

Un risultato non desiderabile

$\sigma_{\text{Età} > 30}(\text{Persone}) \cup \sigma_{\text{Età} \leq 30}(\text{Persone}) \neq \text{Persone}$

- Perché? Perché le selezioni vengono valutate separatamente!

- Ma anche

$\sigma_{\text{Età} > 30 \vee \text{Età} \leq 30}(\text{Persone}) \neq \text{Persone}$

- condizioni atomiche valutate separatamente

Basi di Dati + Laboratorio - Informatica Triennale - Corso A

286

Selezione con valori nulli: soluzione

$\sigma_{Età > 40}$ (Impiegati)

- la condizione atomica è vera solo per valori non nulli
- per riferirsi ai valori nulli esistono forme apposite di condizioni:

IS NULL

IS NOT NULL

- si potrebbe usare (ma non serve) una "logica a tre valori" (vero, falso, sconosciuto)

Basi di Dati + Laboratorio - Informatica Triennale - Corso A

287

Selezione con valori nulli: soluzione

Quindi:

$$\begin{aligned} & \text{SEL}_{Età > 30}(\text{Persone}) \cup \text{SEL}_{Età \leq 30}(\text{Persone}) \cup \text{SEL}_{Età} \\ & \quad \text{IS NULL}(\text{Persone}) \\ & \quad = \\ & \quad \text{SEL}_{Età > 30 \vee Età \leq 30 \vee Età \text{ IS NULL}}(\text{Persone}) \\ & \quad = \\ & \quad \text{Persone} \end{aligned}$$

Basi di Dati + Laboratorio - Informatica Triennale - Corso A

288

Impiegati

Matricola	Cognome	Filiale	Età
5998	Neri	Milano	45
9553	Bruni	Milano	NULL

σ (Età > 40) OR (Età IS NULL) (Impiegati)

Basi di Dati + Laboratorio - Informatica Triennale - Corso A

289

29/10/2020

Basi di Dati + Laboratorio - Informatica Triennale - Corso A

290