

Selezione e proiezione

- Combinando selezione e proiezione, possiamo estrarre interessanti informazioni da una relazione

- matricola e cognome degli impiegati che guadagnano più di 50

Matricola	Cognome
7309	Rossi
5998	Neri
5698	Neri

PROJ_{Matricola,Cognome} (SEL_{Stipendio > 50} (Impiegati))

- Combinando selezione e proiezione, possiamo **estrarre** informazioni da **una** relazione
- Invece non possiamo
 - calcolare informazioni derivate
 - correlare informazioni presenti in relazioni diverse, né informazioni in ennuple diverse di una stessa relazione
- Vediamo altri operatori che permettono queste attività

Join

- il join è l'operatore più interessante dell'algebra relazionale
- permette di correlare dati in relazioni diverse

Il solito esempio

(con nomi di attributi modificati in "esami")

studenti

<u>Matricola</u>	Cognome	Nome	Data di nascita
6554	Rossi	Mario	05/12/1978
8765	Neri	Paolo	03/11/1976
9283	Verdi	Luisa	12/11/1979
3456	Rossi	Maria	01/02/1978

esami

<u>Matricola</u>	Voto	<u>Codice</u>
3456	30	04
3456	24	02
9283	28	01
6554	26	01

Quali esami ha superato
Maria Rossi e quali voti
ha ottenuto?

corsi

<u>Codice</u>	Titolo	Docente
01	Analisi	Mario
02	Chimica I	Bruni
04	Chimica II	Verdi

studenti

Matricola	Cognome	Nome	Data di nascita
3456	Rossi	Maria	01/02/1978

esami

Matricola	Voto	Codice
3456	30	04
3456	24	02

corsi

Codice	Titolo	Docente
02	Chimica I	Bruni
04	Chimica II	Verdi

(studenti JOIN esami) JOIN corsi

Matricola	Cognome	Nome	Data di nascita	Voto	Codice	Titolo	Docente
3456	Rossi	Maria	01/02/1978	30	04	Chimica II	Bruni
3456	Rossi	Maria	01/02/1978	24	02	Chimica I	Verdi

Join naturale

- operatore binario (generalizzabile)
- produce un risultato
 - sull'**unione** degli attributi degli operandi
 - con ennuple costruite ciascuna a partire da una ennupla di ognuno degli operandi

Join, sintassi e semantica

- alla lavagna

JOIN ~~×~~

JOIN NATURALE

$R_1(X_1) \quad R_2(X_2)$

$$R_1 \times R_2 = \{ t \text{ su } X_1 X_2 \mid$$

e esistono $t_1 \in R_1$ e $t_2 \in R_2$

$$t[X_1] = t_1 \text{ e } t[X_2] = t_2 \}$$

Esempi

- alla lavagna
 - join completo, non completo, vuoto, $m \times n$

Impiegato	Reparto
Rossi	A
Neri	B
Bianchi	B

Reparto	Capo
A	Mori
B	Bruni

Impiegato	Reparto	Capo
Rossi	A	Mori
Neri	B	Bruni
Bianchi	B	Bruni

- ogni ennupla contribuisce al risultato:
 - join **completo**

Un join non completo

Impiegato	Reparto
Rossi	A
Neri	B
Bianchi	B

Reparto	Capo
B	Mori
C	Bruni

Impiegato	Reparto	Capo
Neri	B	Mori
Bianchi	B	Mori

Un join vuoto

Impiegato	Reparto
Rossi	A
Neri	B
Bianchi	B

Reparto	Capo
D	Mori
C	Bruni

Impiegato	Reparto	Capo
-----------	---------	------

Un join completo, con $n \times m$ ennuple

Impiegato	Reparto
Rossi	B
Neri	B

Reparto	Capo
B	Mori
B	Bruni

Impiegato	Reparto	Capo
Rossi	B	Mori
Rossi	B	Bruni
Neri	B	Mori
Neri	B	Bruni

Cardinalità del join

- Il join di R_1 e R_2 contiene un numero di ennuple ...
 - alla lavagna (chiavi, vincoli integrità referenziale)

R_1 $|R_1|$

R_2 $|R_2|$

$$0 \leq |R_1 \bowtie R_2| \leq |R_1| \cdot |R_2|$$

R_1

<u>A</u>	B
1	0
2	4
3	4

R_2

<u>B</u>	C
1	x
2	y
3	z
4	w
...	

$$0 \leq |R_1 \bowtie R_2| \leq |R_1|$$

R_1

<u>A</u>	B
1	1
2	4
3	4

R_2

<u>B</u>	C
1	x
2	y
3	z
4	w
...	

VINCOLO DI INTEGRITA' REF.

FRA $R_1(B)$ E LA CHIAVE B DI R_2

$$|R_1 \bowtie R_2| = |R_1|$$

Cardinalità del join

- Il join di R_1 e R_2 contiene un numero di ennuple
 - compreso fra zero e il prodotto di $|R_1|$ e $|R_2|$
- se il join coinvolge una chiave di R_2 , allora il numero di ennuple è
 - compreso fra zero e $|R_1|$
- se il join coinvolge una chiave di R_2 e un vincolo di integrità referenziale verso di essa, allora il numero di ennuple è
 - pari a $|R_1|$

Cardinalità del join, 2

- $R_1(A,B)$, $R_2(B,C)$
- in generale

$$0 \leq |R_1 \text{ JOIN } R_2| \leq |R_1| \times |R_2|$$

- se B è chiave in R_2

$$0 \leq |R_1 \text{ JOIN } R_2| \leq |R_1|$$

- se B è chiave in R_2 ed esiste vincolo di integrità referenziale fra B (in R_1) e R_2 :

$$|R_1 \text{ JOIN } R_2| = |R_1|$$

Join, una difficoltà

Impiegato	Reparto
Rossi	A
Neri	B
Bianchi	B

Reparto	Capo
B	Mori
C	Bruni

Impiegato	Reparto	Capo
Neri	B	Mori
Bianchi	B	Mori

- alcune ennuple non contribuiscono al risultato: vengono "tagliate fuori"

Join esterno

- Il join **esterno** estende, con valori nulli, le ennuple che verrebbero tagliate fuori da un join (**interno**)
- esiste in tre versioni:
 - sinistro, destro, completo

Join, esterno

Impiegato	Reparto
Rossi	A
Neri	B
Bianchi	B

Reparto	Capo
B	Mori
C	Bruni

- Vedere anche su Relax (nella base di dati mostrata per gli esempi, usare R1 e R2)

Impiegati

Impiegato	Reparto
Rossi	A
Neri	B
Bianchi	B

Reparti

Reparto	Capo
B	Mori
C	Bruni

Impiegati $\text{JOIN}_{\text{LEFT}}$ Reparti

Impiegato	Reparto	Capo
Neri	B	Mori
Bianchi	B	Mori
Rossi	A	<i>NULL</i>

Impiegati

Impiegato	Reparto
Rossi	A
Neri	B
Bianchi	B

Reparti

Reparto	Capo
B	Mori
C	Bruni

Impiegati $\text{JOIN}_{\text{RIGHT}}$ Reparti

Impiegato	Reparto	Capo
Neri	B	Mori
Bianchi	B	Mori
<i>NULL</i>	C	Bruni

Impiegati

Impiegato	Reparto
Rossi	A
Neri	B
Bianchi	B

Reparti

Reparto	Capo
B	Mori
C	Bruni

Impiegati $\text{JOIN}_{\text{FULL}}$ Reparti

Impiegato	Reparto	Capo
Neri	B	Mori
Bianchi	B	Mori
Rossi	A	<i>NULL</i>
<i>NULL</i>	C	Bruni

- Su Relax il risultato ha una forma leggermente diversa, ma il concetto è lo stesso

Join e proiezioni

Impiegato	Reparto
Rossi	A
Neri	B
Bianchi	B

Reparto	Capo
B	Mori
C	Bruni

Impiegato	Reparto	Capo
Neri	B	Mori
Bianchi	B	Mori

Impiegato	Reparto
Neri	B
Bianchi	B

Reparto	Capo
B	Mori

Join e proiezioni

- $R_1(X_1), R_2(X_2)$

$$\text{PROJ}_{X_1}(R_1 \text{ JOIN } R_2) \subseteq R_1$$

Proiezioni e join

Impiegato	Reparto	Capo
Neri	B	Mori
Bianchi	B	Bruni
Verdi	A	Bini

Impiegato	Reparto
Neri	B
Bianchi	B
Verdi	A

Reparto	Capo
B	Mori
B	Bruni
A	Bini

Impiegato	Reparto	Capo
Neri	B	Mori
Neri	B	Bruni
Bianchi	B	Mori
Bianchi	B	Bruni
Verdi	A	Bini

Proiezioni e join

Impiegato	Reparto	Capo
Neri	B	Mori
Bianchi	B	Mori
Verdi	A	Bini

Impiegato	Reparto
Neri	B
Bianchi	B
Verdi	A

Reparto	Capo
B	Mori
A	Bini

Impiegato	Reparto	Capo
Neri	B	Mori
Bianchi	B	Mori
Verdi	A	Bini

Proiezioni e join

- $R_1(X_1), R_2(X_2)$

$$\text{PROJ}_{X_1}(R_1 \text{ JOIN } R_2) \subseteq R_1$$

- $R(X), X = X_1 \cup X_2$

$$(\text{PROJ}_{X_1}(R)) \text{ JOIN } (\text{PROJ}_{X_2}(R)) \supseteq R$$

Join e proiezioni

- $R_1(X_1), R_2(X_2)$

$$\text{PROJ}_{X_1}(R_1 \text{ JOIN } R_2) \subseteq R_1$$

- $R(X), X = X_1 \cup X_2$

$$(\text{PROJ}_{X_1}(R)) \text{ JOIN } (\text{PROJ}_{X_2}(R)) \supseteq R$$

- Quando c'è uguaglianza?

Proiezioni e join

Impiegato	Reparto	Capo
Neri	B	Mori
Bianchi	B	Mori
Verdi	A	Bini

Impiegato	Reparto
Neri	B
Bianchi	B
Verdi	A

Reparto	Capo
B	Mori
A	Bini

Join e proiezioni

- $R(X), \quad X = X_1 \cup X_2$

$$(\text{PROJ}_{X_1}(R)) \text{ JOIN } (\text{PROJ}_{X_2}(R)) \supseteq R$$

- Quando c'è uguaglianza?
 - quando gli attributi comuni alle due proiezioni $X_1 \cap X_2$ sono chiave (in effetti, basta superchiave) per una delle due proiezioni)

Prodotto cartesiano

- un join naturale su relazioni senza attributi in comune
- contiene sempre un numero di ennuple pari al prodotto delle cardinalità degli operandi (le ennuple sono tutte combinabili)

Impiegati

Impiegato	Reparto
Rossi	A
Neri	B
Bianchi	B

Reparti

Codice	Capo
A	Mori
B	Bruni

Impiegati JOIN Reparti

Impiegato	Reparto	Codice	Capo
Rossi	A	A	Mori
Rossi	A	B	Bruni
Neri	B	A	Mori
Neri	B	B	Bruni
Bianchi	B	A	Mori
Bianchi	B	B	Bruni

- Il prodotto cartesiano, in pratica, ha senso (quasi) solo se seguito da selezione:

$SEL_{Condizione} (R_1 JOIN R_2)$

- L'operazione viene chiamata **theta-join** e indicata con

$R_1 JOIN_{Condizione} R_2$

Equi-join

- Se l'operatore di confronto nel theta-join è sempre l'uguaglianza (=) allora si parla di **equi-join**

Nota: ci interessa davvero l'equi-join, non il theta-join più generale

- **Equi-join: prodotto cartesiano seguito da selezione di uguaglianza**

Impiegati

Impiegato	Reparto
Rossi	A
Neri	B
Bianchi	B

Reparti

Codice	Capo
A	Mori
B	Bruni

Impiegati JOIN_{Reparto=Codice} Reparti

Impiegato	Reparto	Codice	Capo
Rossi	A	A	Mori
Neri	B	B	Bruni
Bianchi	B	B	Bruni

Join naturale ed equi-join

- In pratica, ciò che ci interessa è l'equi-join
- Il join naturale lo abbiamo usato solo a fini didattici, perché i concetti sono più semplici
- Nelle interrogazioni "pratiche" useremo l'equi-join

Equivalenza di espressioni

- Due espressioni sono **equivalenti** se producono risultati uguali fra loro su ogni istanza della base di dati
- L'equivalenza è importante in pratica perché i DBMS cercano di eseguire espressioni equivalenti a quelle date, ma meno "costose"

Un'equivalenza importante

- Push selections (se A è attributo di R_1)

$$\text{SEL}_{A=10} (R_1 \text{ JOIN } R_2) = \text{SEL}_{A=10} (R_1) \text{ JOIN } R_2$$

Nota

- In questo corso, ci preoccupiamo poco dell'efficienza:
 - L'obiettivo è di scrivere interrogazioni corrette e leggibili
- Motivazione:
 - I DBMS si preoccupano di scegliere le strategie realizzative efficienti