



Modulo 2 Data Base 3

Università degli Studi di Salerno

Corso di Laurea in Scienze della comunicazione

Informatica generale

Docente: Angela Peduto

A.A. 2004/2005

Proiezione

Dati una relazione $r(X)$ e un sottoinsieme Y di X , la proiezione di r su Y si indica con

$$\Pi_Y(r) \text{ o } \text{PROJ}_Y(r)$$

ed è l'insieme di tuple su Y ottenute dalle tuple di r considerando solo i valori su Y .

$$\Pi_Y(r) = \{ \{Y\} \mid t \in r \}$$

Una proiezione ha un numero di tuple *minore o uguale* rispetto alla relazione r cui è applicata. Il numero di tuple è uguale se e solo se Y è superchiave per r .



Proiezione

- operatore monadico
- produce un risultato che
 - possiede parte degli attributi dell'operando
 - contiene tuple cui contribuiscono tutte le tuple dell'operando

- **sintassi**

PROJ_{ListaAttributi} (Operando)

- **semantica**

il risultato contiene le tuple che si ottengono restringendo tutte le tuple dell'operando agli attributi nella lista (ed eliminando gli eventuali duplicati)

Impiegati

Matricola	Cognome	Filiale	Stipendio
7309	Rossi	Roma	55
5998	Neri	Milano	64
9553	Milano	Milano	44
5698	Neri	Napoli	64

- visualizzare matricola e cognome di tutti gli impiegati

Matricola	Cognome
7309	Neri
5998	Neri
9553	Rossi
5698	Rossi

PROJ Matricola, Cognome **(Impiegati)**

- visualizzare cognome e filiale di tutti gli impiegati

Cognome	Filiale
Neri	Napoli
Neri	Milano
Rossi	Roma

PROJ Cognome, Filiale **(Impiegati)**



Join

Combinando selezione e proiezione, si possono estrarre informazioni da **una sola** relazione

Non si possono però correlare informazioni presenti in relazioni diverse

Il **join** è l'operatore più interessante (potente) dell'algebra relazionale in quanto permette di correlare dati presenti in relazioni diverse



Join

E' l'operatore più caratteristico dell'algebra relazionale, che evidenzia la proprietà del modello relazionale di essere basato su valori. Non ha un corrispettivo nella teoria degli insiemi.

L'operatore di join (naturale) correla dati contenuti in relazioni diverse.

Il suo risultato è una relazione definita sull'unione degli insiemi di attributi degli operandi, le cui tuple sono ottenute combinando le tuple degli operandi **che hanno valori uguali** su attributi comuni (con lo stesso nome).



Join

Il join naturale $r_1 \diamond r_2$ di $r_1(X_1)$ e $r_2(X_2)$ è una relazione definita su $X_1 \cup X_2$ (che si può scrivere $X_1 X_2$):

$$r_1 \diamond r_2 = \{ t \text{ su } X_1 X_2 \mid t[X_1] \in r_1 \text{ e } t[X_2] \in r_2 \}$$

Il grado della relazione ottenuta è minore o uguale al grado della somma dei gradi delle due relazioni in quanto gli attributi omonimi compaiono una sola volta.

Se $X_1 \cap X_2$ è vuoto il join naturale equivale al *prodotto cartesiano* fra le relazioni.

Se $X_1 = X_2$ il join naturale equivale all'intersezione fra le relazioni



Join

Se ciascuna tupla di ciascuno degli operandi contribuisce ad almeno una tupla del risultato il join si dice *completo*.

Se per alcune tuple non è verificata la corrispondenza e non contribuiscono al risultato, le tuple si dicono *dangling*.

Ai due estremi si pongono il **join vuoto**, in cui nessuna tupla degli operandi è combinabile, e quello in cui ciascuna delle tuple di un operando è combinabile con tutte le tuple dell'altro. In questo caso la cardinalità della relazione risultante è pari al prodotto della cardinalità degli operandi. E' ovviamente un caso limite, che si verifica solo se esistono attributi comuni che assumono lo stesso valore per ogni tupla di entrambi gli operandi (quindi se l'attributo, di fatto, *non contiene informazione significativa*).

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 tupla contribuisce al risultato: il join è **completo**

Un join non completo

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

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
------------------	----------------	-------------



Join

Proprietà

- Il join di r_1 e r_2 contiene un numero di tuple compreso fra zero e il prodotto di $|r_1|$ e $|r_2|$
- se il join di r_1 e r_2 è completo allora contiene un numero di tuple pari almeno al massimo fra $|r_1|$ e $|r_2|$
- se $X_1 \cap X_2$ contiene una chiave per r_2 , allora il join di $r_1(X_1)$ e $r_2(X_2)$ contiene al più $|r_1|$ tuple.
- se il join coinvolge una chiave di R_2 e un vincolo di integrità referenziale, allora il numero di tuple è pari a $|R_1|$
- $r_1 \diamond r_2 = r_2 \diamond r_1$ il join è *commutativo*
- $(r_1 \diamond r_2) \diamond r_3 = r_1 \diamond (r_2 \diamond r_3)$ il join è *associativo*
Quindi sequenze di join possono essere scritte senza parentesi



Join

Se si devono correlare attributi con nome diverso (cioè $X_1 \cap X_2$ è vuoto) è possibile fare il *theta-join*, definito come un prodotto cartesiano seguito da una selezione

$$r_1 \diamond_F r_2 = \sigma_F (r_1 \diamond r_2)$$

dove F è la condizione di selezione.

Se F è una condizione di uguaglianza fra un attributo della prima relazione e uno della seconda, allora siamo in presenza di un *equi-join*.

Sono importanti formalmente:

il join naturale è basato sui *nomi* degli attributi

equi-join e theta-join sono basati sui *valori*

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

Impiegati

Impiegato Reparto

Reparti

Codice Capo

Impiegati JOIN Reparti

PROJ_{Impiegato,Reparto,Capo} (**SEL**_{Reparto=Codice}
(Impiegati JOIN REN_{Codice ← Reparto} **(Reparti))**)

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 tuple non contribuiscono al risultato: vengono "tagliate fuori" (tuple *dangling*)



Join

Join Esterni

Il join naturale tralascia le tuple in cui non vi è corrispondenza fra gli attributi legati dal join. Si definiscono allora altri tipi di join, che fanno sì che anche quelle tuple vengano considerate, inserendo valori nulli dove non vi sia corrispondenza.

Join sinistro

Contribuiscono tutte le tuple del primo operando eventualmente estese con valori nulli

Join destro

Contribuiscono tutte le tuple del secondo operando eventualmente estese con valori nulli

Join completo

Contribuiscono tutte le tuple del primo e del secondo operando eventualmente estese con valori nulli

Impiegati

Impiegato	Reparto
Rossi	A
Neri	B
Bianchi	B

Reparti

Reparto	Capo
B	Mori
C	Bruni

Impiegati JOIN_{LEFT} Reparti

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

Impiegati

Impiegato	Reparto
Rossi	A
Neri	B
Bianchi	B

Reparti

Reparto	Capo
B	Mori
C	Bruni

Impiegati JOIN_{RIGHT} Reparti

Impiegato	Reparto	Capo
Neri	B	Mori
Bianchi	B	Mori
NULL	C	Bruni

Impiegati

Impiegato	Reparto
Rossi	A
Neri	B
Bianchi	B

Reparti

Reparto	Capo
B	Mori
C	Bruni

Impiegati JOIN_{FULL} Reparti

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



Interrogazioni (Query)

Un'interrogazione è una funzione $E(\mathbf{r})$ che applicata ad istanze \mathbf{r} di una base di dati produce una relazione su un dato insieme di attributi X .

Le interrogazioni su uno schema di base di dati \mathbf{R} in algebra relazionale sono espressioni i cui atomi (le variabili) sono relazioni in \mathbf{R} .

Esempi

Impiegati	Matricola	Nome	Età	Stipendio
	7309	Rossi	34	45
	5998	Bianchi	37	38
	9553	Neri	42	35
	5698	Bruni	43	42
	4076	Mori	45	50
	8123	Lupi	46	60

Supervisione	Impiegato	Capo
	7309	5698
	5998	5698
	9553	4076
	5698	4076
	4076	8123

- Trovare matricola, nome, età e stipendio degli impiegati che guadagnano più di 40 milioni

$SEL_{\text{Stipendio} > 40}(\text{Impiegati})$

Matricola	Nome	Età	Stipendio
7309	Rossi	34	45
5698	Bruni	43	42
4076	Mori	45	50
8123	Lupi	46	60

- Trovare matricola, nome ed età degli impiegati che guadagnano più di 40 milioni

PROJ_{Matricola, Nome, Età} (SEL_{Stipendio>40}(Impiegati))

Matricola	Nome	Età
7309	Rossi	34
5698	Bruni	43
4076	Mori	45
8123	Lupi	46

- Trovare le matricole dei capi degli impiegati che guadagnano più di 40 milioni

```
PROJ_Capo (Supervisione  
JOIN Impiegato=Matricola (SEL_Stipendio>40(Impiegati)))
```