

Join

Per correlare attributi con nome diverso (se 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*. E' il tipo di join più comune.

Sono importanti formalmente:

- il join naturale è basato sui *nomi* degli attributi
- equi-join e theta-join sono basati sui *valori*

Join

Il theta-join, espresso come prodotto cartesiano seguito da una selezione, è il tipo di join operativamente più generale. Infatti:

- Consente di associare (concatenare) tuple attraverso la corrispondenza fra attributi arbitrari e non necessariamente omonimi
- Consente di basare l'associazione su una arbitraria condizione e non solo su una condizione 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

Impiegati

Impiegato Reparto

Reparti

Codice Capo

Impiegati JOIN_{Reparto=Codice} **Reparti** (*equi-join*)

equivale a

SEL_{Reparto=Codice} (**Impiegati** JOIN **Reparti**)

Il risultato è definito sullo schema

Impiegato Reparto Codice Capo

con valori identici per ogni tupla per gli attributi
Reparto e Codice (*inutilmente duplicati*)

Join naturale ed equi-join

Impiegati

Impiegato Reparto

Reparti

Codice Capo

Se avessi usato ridenominazione + join naturale

Impiegati JOIN (REN_{Reparto} ← Codice (Reparti))

il risultato sarebbe stato definito sullo schema

Impiegato Reparto Capo

che equivale, in realtà, a *(gli attributi resi omonimi dalla ridenominazione compaiono una sola volta)*

PROJ_{Impiegato,Reparto,Capo} (SEL_{Reparto=Codice}

(Impiegati JOIN Reparti))

Join esterni

Il join naturale e il theta-join non includono nel risultato le tuple in cui non c'è corrispondenza fra gli attributi legati dal join (*tuple dangling*).

Questo fa sì che il risultato non contenga tutta l'informazione contenuta nelle relazioni di partenza

Si definiscono altri tipi di join, detti *join esterni*, in cui vengono considerate anche le tuple dangling, utilizzando valori nulli dove non vi sia corrispondenza.

Join esterni

Join sinistro ($r_1 \text{ JOIN}_{\text{LEFT}} r_2$)

Contribuiscono tutte le tuple di r_1 , eventualmente estese con valori nulli relativamente agli attributi di r_2 se non c'è corrispondenza con tuple di r_2

Join destro ($r_1 \text{ JOIN}_{\text{RIGHT}} r_2$)

Contribuiscono tutte le tuple di r_2 , eventualmente estese con valori nulli relativamente agli attributi di r_1 se non c'è corrispondenza con tuple di r_1

Join completo o full join ($r_1 \text{ JOIN}_{\text{FULL}} r_2$)

Contribuiscono tutte le tuple di r_1 e r_2 , eventualmente estese con valori nulli relativamente agli attributi dell'altra relazione se non c'è corrispondenza con tuple dell'altra relazione.

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

Join e proiezioni: problemi (perdite)

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: problemi (non reversibilità)

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
Bianchi	B	Bruni
Neri	B	Bruni
Bianchi	B	Mori
Verdi	A	Bini

Join e proiezioni: proprietà

- Date due relazioni $R_1(X_1)$ e $R_2(X_2)$

$$\Pi_{X_1}(R_1 \diamond R_2) \subseteq R_1$$

$$\Pi_{X_2}(R_1 \diamond R_2) \subseteq R_2$$

Se proietto il risultato del join sullo schema di una delle due relazioni ne ottengo un sottoinsieme.

Infatti, se non c'è corrispondenza fra alcune tuple di R_1 e di R_2 , tali tuple sono dangling e vengono escluse dal join. Se si tenta quindi di ricostruire le due relazioni originali si ha perdita di informazione.

Join e proiezioni: proprietà

- $R(X), \quad X = X_1 \cup X_2$
 $(\Pi_{X_1}(R)) \Join (\Pi_{X_2}(R)) \supseteq R$

Se scompongo lo schema di una relazione R in due sottoschemi X_1 e X_2 , R è contenuta nella relazione che si ottiene effettuando il join fra le proiezioni di R su di essi.

Equivalenze di espressioni

Due espressioni sono equivalenti se:

$E_1 \equiv_R E_2$ se $E_1(r) = E_2(r)$ per ogni istanza r di R
(equivalenza dipendente dallo schema) 

$E_1 \equiv E_2$ se $E_1 \equiv_R E_2$ per ogni schema R
(equivalenza assoluta)

L'equivalenza è importante in quanto consente di scegliere, a parità di risultato, l'operazione meno costosa.

Equivalenze

- Atomizzazione delle selezioni

$$\sigma_{F_1 \wedge F_2} (E) \equiv \sigma_{F_1} (\sigma_{F_2} (E))$$



- Idempotenza delle proiezioni

$$\Pi_X (E) \equiv \Pi_X (\Pi_{XY} (E)) \quad \text{qualunque sia } Y$$

- Anticipazione della selezione rispetto al join

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

Equivalenze

- Anticipazione della proiezione rispetto al join:

$$\Pi_{X_1 Y_2} (E_1 \diamond E_2) \equiv E_1 \diamond \Pi_{Y_2} (E_2)$$

(E_1 definita su X_1 , E_2 su X_2 ; vera se Y_2 contiene gli attributi coinvolti nella condizione di join)

Allora (combinando con idempotenza delle proiezioni):

$$\Pi_Y (E_1 \diamond_F E_2) \equiv \Pi_Y (\Pi_{Y_1} (E_1) \diamond_F \Pi_{Y_2} (E_2))$$

dove Y_1 e Y_2 contengono, rispettivamente, gli attributi di X_1 e X_2 compresi in Y o coinvolti nella condizione F .

In pratica è possibile ignorare, in ciascuna relazione, gli attributi non compresi in Y e non coinvolti nel join

Equivalenze

- Inglobamento di una selezione in un prodotto cartesiano a formare un join (*theta-join*) :

$$\sigma_F (E_1 \diamond E_2) \equiv E_1 \diamond_F E_2$$

- Tutti gli operatori binari eccetto la differenza godono delle proprietà associativa e commutativa.

Equivalenze

- Distributività della selezione rispetto all'unione:

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

- Distributività della selezione rispetto alla differenza:

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

- Distributività della proiezione rispetto all'unione:

$$\Pi_X (E_1 \cup E_2) \equiv \Pi_X (E_1) \cup \Pi_X (E_2)$$

NB La proiezione NON è distributiva rispetto alla differenza

Equivalenze

- Corrispondenze fra operatori insiemistici e selezioni complesse

$$\sigma_{F_1 \vee F_2}(R) \equiv \sigma_{F_1}(R) \cup \sigma_{F_2}(R)$$

$$\sigma_{F_1 \wedge F_2}(R) \equiv \sigma_{F_1}(R) \cap \sigma_{F_2}(R) \equiv \sigma_{F_1}(R) \diamond \sigma_{F_2}(R)$$

$$\sigma_{F_1 \wedge \neg F_2}(R) \equiv \sigma_{F_1}(R) - \sigma_{F_2}(R)$$

- Proprietà distributiva del join rispetto all'unione:

$$E \diamond (E_1 \cup E_2) \equiv (E \cup E_1) \diamond (E \cup E_2)$$

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

Esempi

- 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

Esempi

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

$\text{PROJ}_{\text{Matricola, Nome, Età}} (\text{SEL}_{\text{Stipendio} > 40} (\text{Impiegati}))$

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

Procedimento logico

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

$$\text{PROJ}_{\text{Capo}} \left(\text{SEL}_{\text{Stipendio} > 40} \left(\text{Supervisione JOIN}_{\text{Impiegato} = \text{Matricola}} \text{Impiegati} \right) \right)$$

- 1. Creo una singola tabella in cui ogni tupla contiene (fra gli altri) gli attributi Capo, Impiegato e Stipendio con un join fra Supervisione e Impiegati***
- 2. Seleziono le tuple con stipendio > 40***
- 3. Proietto il risultato sull'attributo Capo***