



PEGASO
Università Telematica



Indice

1. INTRODUZIONE.....	3
2. CARDINALITÀ.....	4
3. JOIN ESTERNO	5
3.1 JOIN SINISTRO	6
3.2 JOIN DESTRO	8
3.3 JOIN COMPLETO	8
4. SEMI-JOIN.....	10
BIBLIOGRAFIA	12
SITOGRAFIA	12

1. Introduzione

In questa unità didattica, dopo aver affrontato lo studio di un parametro importante come la cardinalità dell'operatore JOIN, si espongono altre tipologie di operazione di congiunzione:

1. Join esterno (sinistro, destro e completo);
2. Semi-Join.

2. Cardinalità

In questo paragrafo introduciamo un concetto importante, ovvero il concetto di Cardinalità. Tale concetto è strettamente legato al numero di ennuple che fanno parte della relazione risultante prodotta dall'operatore Join.

Definizione: Date due relazioni R_1 ed R_2 , definiamo **Cardinalità** del Join tra R_1 ed R_2 :

$|R_1 \bowtie R_2|$ = numero di ennuple appartenenti alla relazione risultante

Ne conseguono le seguenti considerazioni. Date due relazioni, R_1 e R_2 si ha che:

- 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, allora il numero di ennuple è:
 - pari a $|R_1|$

Quindi, date due relazioni $R_1(A,B)$, $R_2(B,C)$ si ha che:

- In generale:
$$0 \leq |R_1 \bowtie R_2| \leq |R_1| \times |R_2|$$
- Se B è chiave in R_2 si ha che:
$$0 \leq |R_1 \bowtie R_2| \leq |R_1|$$
- Se B è chiave in R_2 ed esiste vincolo di integrità referenziale fra B (in R_1) e R_2 si ha:

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

3. Join Esterno

Il join visto fino ad ora è il natural join, detto anche equi-join o join-interno o inner-join, in cui la congiunzione viene realizzata solo sulle tuple di due tabelle distinte che possiedono valori uguali per gli attributi corrispondenti. Ad esempio¹, date le due relazioni Persone(Codice, Nome, Cognome) e Automobili(Targa, Modello, CodiceID), illustrate in Figura 1

Codice	Nome	Cognome
1	Mario	Rossi
2	Luigi	Bianchi
3	Paolo	Neri

Targa	Modello	CodiceID
AASSGG	Tipo1	1
UUJJKK	Tipo2	1
PPLLBB	Tipo3	2
WWYYXX	Tipo4	NULL

Figura 1: le due relazioni Persone (sinistra) e Automobili (destra).

Effettuando il Join naturale o inner.Join, si ha la relazione risultante illustrata in Figura 2.

Targa	Modello	CodiceID	Nome	Cognome
AASSGG	Tipo1	1	Mario	Rossi
UUJJKK	Tipo2	1	Mario	Rossi
PPLLBB	Tipo3	2	Luigi	Bianchi

Figura 2: relazione risultante dall'operazione di Join naturale tra le relazioni Persone e Automobili.

L'operazione eseguita esclude le tuple aventi valore nullo per l'attributo in comune (l'auto con targa WWYYXX) e o che non hanno un elemento che gli corrisponde (la persona Paolo Neri). Quindi, alcune ennuple non contribuiscono al risultato e vengono «tagliate fuori». È possibile infatti che una tupla di una delle relazioni operande non faccia match con nessuna tupla dell'altra relazione; in tal caso tale tupla viene detta "dangling". Un altro esempio è quello delle due relazioni di Figura 3 con le due relazioni Impiegati (Impiegato, Reparto) e Reparti(Reparto, Capo).

¹ <http://www.senorbi.net/mod/resource/view.php?id=1420>

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

Figura 3: relazioni Impiegati e Reparti.

Come risultato otteniamo la relazione di Figura 4, dove anche qui non tutte le ennuple delle relazioni operande appaiono nella relazione risultato.

Impiegato	Reparto	Capo
Neri	B	Mori
Bianchi	B	Mori

Figura 4: risultato del Join naturale tra le due relazioni Impiegati e Reparti.

Per superare tale problema, si fa ricorso al **join esterno** il quale estende, con valori nulli, le ennuple che verrebbero tagliate fuori da un join (**naturale**). Tale join esiste in tre versioni: sinistro, destro e completo:

- **Join sinistro:** mantiene tutte le ennuple del primo operando, estendendole con valori nulli, se necessario;
- **Join destro:** mantiene tutte le ennuple del secondo operando, estendendole con valori nulli, se necessario;
- **Join completo:** mantiene tutte le ennuple di entrambi gli operandi, estendendole con valori nulli, se necessario;

3.1 Join Sinistro

Questo è il caso in cui esistono tuple appartenenti al primo operando formate da attributi che non hanno intersezione vuota con il secondo operando. Pertanto un Join naturale darebbe il risultato illustrato in Figura 4, lasciando fuori alcune tuple o ennuple. Il Join sinistro o Left-Join tra le due relazioni di Figura 5,

dove la tupla $t=(\text{Rossi},A)$ non appartenerebbe alla relazione prodotta da un Join naturale, permette di superare tale problema generando la relazione di Figura 6 dove è illustrata la relazione risultato ovvero il left-join.

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

Figura 5: la relazione Impiegati e la relazione Reparti. La tupla $t(\text{Rossi}, A)$ non ha riscontri nella relazione Reparti.

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

Figura 6: relazione risultato del Left-Join tra le due relazioni di Figura 5.

Come si può notare, la tupla $t=(\text{Rossi},A, \text{NULL})$ è stata inserita nel risultato con il valore NULL. Nel caso delle due relazioni Automobili e Persone di Figura 1, si ha che il Left- Join elenca tutte le righe della prima tabella (quella a sinistra dell'operatore) congiungendo tra le righe della seconda, solo quelle per le quali si trovano valori corrispondenti per gli attributi comuni, come illustrato in Figura 7.

Targa	Modello	CodiceID	Nome	Cognome
AASSGG	Tipo1	1	Mario	Rossi
UUJJKK	Tipo2	1	Mario	Rossi
PPLLBB	Tipo3	2	Luigi	Bianchi
WWYYXX	Tipo4	null	null	null

Figura 7: Persone Left-Join Automobili.

3.2 Join destro

Il Join destro o Right-Join, elenca tutte le righe della seconda tabella (quella a destra dell'operatore) congiungendo tra le righe della prima, solo quelle per le quali si trovano valori corrispondenti per gli attributi comuni. Nel primo esempio di *Figura 8*, vediamo il risultato del right-join tra le due relazioni Impiegati e Reparti, mentre in *Figura 9* il risultato del right-join tra le due relazioni Persone e Automobili.

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

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

Figura 8: esempio di right-join.

Targa	Modello	CodiceID	Nome	Cognome
AASSGG	Tipo1	1	Mario	Rossi
UUJJKK	Tipo2	1	Mario	Rossi
PPLLBB	Tipo3	2	Luigi	Bianchi
null	null	3	Paolo	Neri

Figura 9: right-join tra le relazioni Persone e Automobili.

3.3 Join completo

Il Join completo o Full-Join, combina le caratteristiche del left join con quelle del right join. Come si può notare dalla *Figura 10*, la relazione risultante contiene tutte le tuple che erano presenti o nel Left-Join o nel Right-Join.

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

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

Figura 10: Full-Join tra le relazioni Impiegati e Reparti.

Come secondo esempio, prendiamo sempre in considerazione le due relazioni Persone e Automobili. Il risultato del full-Join è quello di Figura 11.

Targa	Modello	CodiceID	Nome	Cognome
AASSGG	Tipo1	1	Mario	Rossi
UUJJKK	Tipo2	1	Mario	Rossi
PPLLBB	Tipo3	2	Luigi	Bianchi
null	null	3	Paolo	Neri
WWYYXX	Tipo4	null	null	null

Figura 11: full-Join tra le due relazioni Persone e Automobili.

4. Semi-join

Il Semi-Join è un operatore binario che agisce su due relazioni $R_1(X_1)$, $R_2(X_2)$ e restituisce una relazione su X_1 , con le ennuple di R_1 che contribuiscono al join con R_2 . Formalmente, il semi-Join R_1 con R_2 è una relazione su X_1 così definita:

$$\{\forall t \mid t \in R_1 \text{ ed esiste } t_2 \in R_2 \mid t[X_1 \cap X_2] = t_2[X_1 \cap X_2]\}$$

Es. vogliamo effettuare il semijoin tra le due relazioni Impiegati(Impiegato, Reparto) e Reparti(Reparto, Capo) di Figura 12. Il risultato è quello di Figura 13. Quindi Il semi-join di due relazioni A e B seleziona tutte le tuple di A “semanticamente legate” ad almeno una tupla di B le informazioni di B non compaiono nel risultato.

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

Figura 12: le due relazioni Impiegati e Reparti.

Impiegati ⋈ Reparti	
Impiegato	Reparto
Neri	B
Bianchi	B

Figura 13: semi-join tra Impiegati e Reparti.

Da notare che, se volessimo effettuare il semi-join tra Reparti(Reparto, Capo) e Impiegati(Impiegato, Reparto), il risultato sarebbe diverso come illustrato in Figura 14, confrontando tale risultato con la Figura 13.

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

Reparti ⋈ Impiegati	
Reparto	Capo
B	Mori

Figura 14: il semi-join tra Reparti e Impiegati.

Bibliografia

- Atzeni P., Ceri S., Fraternali P., Paraboschi S., Torlone R. (2018). Basi di Dati. McGraw-Hill Education
- Batini C., Lenzerini M. (1988). Basi di Dati. In Cioffi G. and Falzone V. (Eds). Calderini. Seconda Edizione

Sitografia

- <http://www-db.disi.unibo.it/courses/SIG/algebra2.pdf>
- <http://dbdmg.polito.it/wordpress/wp-content/uploads/2010/12/2.2-AlgebraRelazionale-PS.pdf>