



Basi di dati

Capitolo 3 – Algebra relazionale

Prof. Ivan Gentile

Operazioni su un Data Base (1/2)

- Creazione della base di dati e autorizzazioni agli utenti
 - Fatta una tantum (o quasi) all'inizio
 - Effettuata con dei linguaggi detti **DDL (Data Definition Language)**
- Interrogazione e aggiornamento della base di dati
 - Fatte spesso
 - Effettuate con dei linguaggi detti **DML (Data Manipulation Language)**

Operazioni su un Data Base (2/2)

- Spesso i linguaggi «reali» hanno entrambe le funzionalità (DDL, DML)
 - esempio **SQL**
- L'accesso ai dati può avvenire in vari modi
 - Interprete SQL (disponibile in tutti i DBMS)
 - Incastonando (*embedded*) comandi SQL in linguaggi tradizionali (C, C++, Java, etc.)
 - Incastonando (*embedded*) comandi simili a SQL in linguaggi di sviluppo *ad hoc*
 - Tramite interfacce grafiche «amichevoli»

Interrogazione e aggiornamento

- **Aggiornamento:** funzione che data un'istanza di base di dati produce un'altra istanza sullo stesso schema
- **Interrogazione:** funzione che data un'istanza di base di dati produce una relazione su un dato schema

Algebra relazionale

- Analogamente alla matematica: un insieme di relazioni e operazioni su di esse
- Tipi di operatori
 - *Insiemistici*: unione, intersezione, differenza
 - *Specifici*: ridenominazione, selezione, proiezione
 - *Join nelle varie forme*: naturale, prodotto cartesiano, theta-join.

Operatori insiemistici

- Le relazioni sono insiemi, quindi si possono applicare le operazioni tra insiemi
- Essendo le relazioni insiemi di tuple omogenee, si applicano tra relazioni con *stesso schema*
 - Per evitare di avere insiemi di tuple disomogenee

Esempio

LAUREATI

Matricola	Cognome	Età
7274	Rossi	37
7432	Neri	39
9824	Verdi	38

LAUREATI \cup DIRIGENTI

Matricola	Cognome	Età
7274	Rossi	37
7432	Neri	39
9824	Verdi	38
9297	Neri	56

Prof. Gentile Ivan

DIRIGENTI

Matricola	Cognome	Età
9297	Neri	56
7432	Neri	39
9824	Verdi	38

LAUREATI \cap DIRIGENTI

Matricola	Cognome	Età
7432	Neri	39
9824	Verdi	38

LAUREATI $-$ DIRIGENTI

Matricola	Cognome	Età
7274	Rossi	37

Operatori specifici

- Ridenominazione
- Selezione
- Proiezione
- Join (congiunzione)
 - Nelle varie forme

Ridenominazione (1/2)

- La limitazione imposta agli operatori insiemistici (stesso nome degli attributi), può essere pesante
- Nell'esempio a lato sarebbe sensato fare l'unione ottenendo le coppie «genitore-figlio»

PATERNITA'

Padre	Figlio
Adamo	Caino
Adamo	Abele
Abramo	Isacco
Abramo	Ismaele

MATERNITA'

Madre	Figlio
Eva	Caino
Eva	Set
Sara	Isacco
Agar	Ismaele

Ridenominazione (2/2)

- Operatore che cambia il nome agli attributi
- Al fine di facilitare operazioni insiemistiche
- Agisce solo schema di relazione (non sul corpo)

PATERNITA'

Padre	Figlio
Adamo	Caino
Adamo	Abele
Abramo	Isacco
Abramo	Ismaele

$\rho_{\{\text{Genitore} \leftarrow \text{Padre}\}}(\text{PATERNITA}')$

Genitore	Figlio
Adamo	Caino
Adamo	Abele
Abramo	Isacco
Abramo	Ismaele

Esempio

$$\rho_{\{\text{Genitore} \leftarrow \text{Padre}\}}(\text{PATERNITA}') \cup \rho_{\{\text{Genitore} \leftarrow \text{Madre}\}}(\text{MATERNITA}')$$

Genitore	Figlio
Adamo	Caino
Adamo	Abele
Abramo	Isacco
Abramo	Ismaele
Eva	Caino
Eva	Set
Sara	Isacco
Agar	Ismaele

Selezione e Proiezione

A	B	C

selezione

A	B	C

A	B	C
Prof. Gentile Ivan		

proiezione

A	B

Esempio

IMPIEGATI

Cognome	Nome	Età	Stipendio
Rossi	Mario	25	2000,00
Neri	Luca	40	3000,00
Verdi	Nico	36	4500,00
Rossi	Marco	40	3900,00

$\sigma_{\text{Età} > 30 \wedge \text{Stipendio} > 4000,00}(\text{IMPIEGATI})$

Cognome	Nome	Età	Stipendio
Verdi	Nico	36	4500,00

Esempio

IMPIEGATI

Cognome	Nome	Reparto	Capo
Rossi	Mario	Vendite	Gatti
Neri	Luca	Vendite	Gatti
Verdi	Mario	Personale	Lupi
Rossi	Marco	Personale	Lupi

$\pi_{\text{Cognome, Nome}}(\text{Impiegati})$

Cognome	Nome
Rossi	Mario
Neri	Luca
Verdi	Mario
Rossi	Marco

Osservazioni

- Una proiezione può contenere meno tuple della relazione operando
- Se Y è superchiave sicuramente contiene lo stesso numero di tuple

IMPIEGATI

Cognome	Nome	Reparto	Capo
Rossi	Mario	Vendite	Gatti
Neri	Luca	Vendite	Gatti
Verdi	Mario	Personale	Lupi
Rossi	Marco	Personale	Lupi

$\pi_{\text{Reparto, Capo}}(\text{Impiegati})$

Reparto	Capo
Vendite	Gatti
Personale	Lupi

Esercizio 1

- Data la seguente relazione, contenente i voti di studenti universitari a certi esami, esprimere in algebra relazionale ed esprimere il risultato in forma tabellare
 - l'operazione di selezione degli studenti che hanno avuto 30 senza Lode
 - L'operazione di selezione del CodCorso 729 o degli studenti che hanno avuto 30

Esami

Matricola	CodCorso	Voto	Lode
29323	483	28	NO
39654	729	30	Sì
29323	913	26	NO
35467	913	30	NO
31283	729	30	NO

Prof. Gentile Ivan

Esercizio 2

- Data la relazione in figura, esprimere in algebra relazionale e, il risultato, in forma tabellare le proiezioni su
 - Codice corso e docente
 - Codice corso e anno
 - Titolo del corso
 - Docenti

Corsi	CodCorso	Titolo	Docente	Anno
	483	Analisi	Biondi	1
	729	Analisi	Neri	1
	913	Sistemi Informativi	Castani	2

Join

- Operatore che permette di correlare dati in relazioni diverse
- La correlazione è basata sui valori
- Esistono fondamentalmente due versioni
 - **Join naturale:** utile per riflessioni astratte
 - **Theta-join** : più rilevante dal punto di vista pratico

Join naturale

- $r_1(X_1), r_2(X_2)$
- $r_1 \bowtie r_2 = \{t \text{ su } X_1X_2 \mid t[X_1] \in r_1 \wedge t[X_2] \in r_2\}$

r_1

Impiegato	Reparto
Rossi	Vendite
Neri	Produzione
Bianchi	Produzione

r_2

Reparto	Capo
Produzione	Mori
Vendite	Bruni

$r_1 \bowtie r_2$

Impiegato	Reparto	Capo
Rossi	Vendite	Bruni
Neri	Produzione	Mori
Bianchi	Produzione	Mori

Osservazioni

- Spesso il join si fa coinvolgendo la chiave di una relazione
- Se inoltre c'è un vincolo di integrità referenziale il join avrà esattamente il numero di righe della relazione primo argomento del vincolo (relazione slave)
- Se ciascuna tupla di ciascuno degli operandi contribuisce ad almeno una tupla del risultato il join si dice **completo**

Cardinalità

- $0 \leq |r_1 \bowtie r_2| \leq |r_1||r_2|$
- In particolare
 - Se il join è completo allora contiene un numero di tuple $\geq \max(|r_1|, |r_2|)$
 - Se $X_1 \cap X_2$ contiene una chiave per r_2 , allora il join ha al più $|r_1|$ tuple
 - Se $X_1 \cap X_2$ coincide con una chiave per r_2 , e sussiste il vincolo di riferimento tra $X_1 \cap X_2$ in r_1 e la chiave di r_2 , allora il join contiene esattamente $|r_1|$ tuple

Esercizio 3

- Consideriamo una prova scritta in un concorso pubblico, allo scopo di garantire una correzione anonima a ciascun compito viene posto in una busta con un numero mentre da parte è segnata l'associazione del numero con il nominativo del candidato. Supponendo che informazioni siano organizzate nelle tabelle seguenti, effettuare un Join naturale per ottenere l'indicazione del voto di ogni nominativo

Numero	Voto
1	25
2	13
3	27
4	28

Numero	Candidato
1	Mario Rossi
2	Nicola Russo
3	Mario Bianchi
4	Remo Neri

Esercizio 4

- Effettuare il Join Naturale tra le tabelle in figura

Esami

Matricola	CodCorso	Voto	Lode
29323	483	28	NO
39654	729	30	SÌ
29323	913	26	NO
35467	913	30	NO

Corsi

CodCorso	Titolo	Docente	Anno
483	Analisi	Biondi	1
729	Analisi	Neri	1
913	Sistemi Informativi	Castani	2

Join esterni (outer join)

- Tutte le tuple danno un contributo, eventualmente estese con valori nulli
- Tre varianti
 - **Esterno sinistro:** estende solo le tuple del primo operando
 - **Esterno destro:** estende solo le tuple del secondo operando
 - **Completo:** estende tutte

Esempio

r_1

Impiegato	Reparto
Rossi	Vendite
Neri	Produzione
Bianchi	Produzione

r_2

Reparto	Capo
Produzione	Mori
Acquisti	Bruni

$r_1 \bowtie_{\text{LEFT}} r_2$

Impiegato	Reparto	Capo
Rossi	Vendite	NULL
Neri	Produzione	Mori
Bianchi	Produzione	Mori

Prof. Gentile Ivan

$r_1 \bowtie_{\text{RIGHT}} r_2$

Impiegato	Reparto	Capo
Neri	Produzione	Mori
Bianchi	Produzione	Mori
NULL	Acquisti	Bruni

$r_1 \bowtie_{\text{FULL}} r_2$

Impiegato	Reparto	Capo
Rossi	Vendite	NULL
Neri	Produzione	Mori
Bianchi	Produzione	Mori
NULL	Acquisti	Bruni

Esercizio 5

- Date le seguenti tabelle effettuare il left join, right join e il full join

Ricercatori	Nome	CodProgetto
	Rossi	HK27
	Bianchi	HK27
	Verdi	HK28

Progetti	CodProgetto	Responsabile
	HK27	Bianchi
	HAL2000	Neri

Proprietà del join naturale

- Ci riferiamo al join naturale, non al join esterno, per il quale alcune proprietà non valgono
- Commutativa: $r_1 \bowtie r_2 = r_2 \bowtie r_1$
- Associativa: $r_1 \bowtie (r_2 \bowtie r_3) = (r_1 \bowtie r_2) \bowtie r_3$
- Se gli attributi delle relazioni sono uguali diventa un'intersezione
- Se gli attributi sono disgiunti, la condizione che definisce il join è sempre verificata, ogni tupla di una relazione è combinata con tutte le tuple dell'altra: **prodotto cartesiano**

Esempio: prodotto cartesiano

IMPIEGATI

Impiegato	Progetto
Rossi	A
Neri	A
Neri	B

PROGETTI

Codice	Nome
A	Venere
B	Marte

IMPIEGATI \bowtie PROGETTI

Impiegato	Progetto	Codice	Nome
Rossi	A	A	Venere
Neri	A	A	Venere
Neri	B	A	Venere
Rossi	A	B	Marte
Neri	A	B	Marte
Neri	B	B	Marte

Esercizio 6

- Effettuare il Join Naturale tra le tabelle in figura

VoliCharter	Codice	Data
	XY123	21/07/2001
	SC278	28/07/2001
	XX338	18/08/2001
VoliNoSmoking	Numero	Giorno
	SC278	28/07/2001
	SC315	30/07/2001

Theta-join

- Il prodotto cartesiano, ha poca utilità
- Concatena tuple non necessariamente correlate da un punto di vista semantico
- Su usa spesso allora il **theta-join**
- Operatore derivato, infatti dato da un prodotto cartesiano seguito da una selezione
 - $r_1 \bowtie_F r_2 = \sigma_F(r_1 \bowtie r_2)$
 - F formula proposizionale

Esempio: quello di prima

IMPIEGATI

Impiegato	Progetto
Rossi	A
Neri	A
Neri	B

PROGETTI

Codice	Nome
A	Venere
B	Marte

IMPIEGATI $\bowtie_{\text{Progetto=Codice}}$ PROGETTI

Impiegato	Progetto	Codice	Nome
Rossi	A	A	Venere
Neri	A	A	Venere
Neri	B	B	Marte

Equi-join

- Un theta-join in cui la condizione di selezione F sia una congiunzione di atomi di uguaglianza, con un attributo della prima relazione e uno della seconda
- Quindi come l'esempio di prima

Esercizio 7

- Date le due relazioni in figura
 - effettuare un equi-join sugli attributi CodProgetto e Sigla
 - effettuare un theta-join che vada a selezionare i CodProgetto con Sigla e Nome diverso da Responsabile

Ricercatori

Nome	CodProgetto
Rossi	HK27
Verdi	HAL2000
Bianchi	HK27
Verdi	HK28
Neri	HAL2000

Progetti

Sigla	Responsabile
HK27	Bianchi
HAL2000	Neri
HK28	Verdi

Nella pratica

- Il theta-join e l'equi-join hanno rilevanza pratica perché di solito i linguaggi di interrogazione dei database non usano i nomi di attributo per correlare relazioni
- Pertanto più diffusi sono il theta-join o l'equi-join piuttosto che il join naturale
- Tuttavia il join naturale è reso disponibile dalle ultime versioni di SQL
- Il join naturale si può ottenere da un ridenominazione, equi-join e proiezione

Interrogazione

- Data una base di dati di schema R
- **Interrogazione**: una funzione che per ogni istanza $r \in R$, produce una relazione su un dato insieme di attributi X
- In algebra relazionale, le interrogazioni sono espressioni su relazioni in R

Esempio (1/5)

IMPIEGATI

<u>Matr</u>	Nome	Età	Stipendio
101	Mario Rossi	34	40
103	Mario Bianchi	23	35
104	Luigi Neri	38	61
105	Nico Bini	44	38
210	Marco Celli	49	60
231	Siro Bisi	50	60
252	Nico Bini	44	70
301	Sergio Rossi	34	70
375	Mario Rossi	50	65

SUPERVISIONE

<u>Capo</u>	<u>Impiegato</u>
210	101
201	103
201	104
231	105
301	210
301	231
375	252



Esempio (2/5)

Trovare matricola, nome ed età degli impiegati che guadagnano più di 40 mila euro

$\pi_{\text{Matr, Nome, Età}}(\sigma_{\text{Stipendio} > 40}(\text{IMPIEGATI}))$

DB

Matr	Nome	Età
104	Luigi Neri	38
210	Marco Celli	49
231	Siro Bisi	50
252	Nico Bini	44
375	Mario Rossi	50

Esempio (3/5)

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

DB

$\pi_{\text{Capo}}(\text{SUPERVISIONE} \bowtie_{\text{Impiegato=Matr}} \sigma_{\text{Stipendio} > 40}(\text{IMPIEGATI}))$

Capo
210
301
375

Esempio (4/5)

Trovare nome e stipendio dei capi degli impiegati che guadagnano più di 40 mila euro

- Ogni tupla del risultato è costituita a partire da 3 tuple
 - La prima di IMPIEGATI (impiegati che guadagnano più di 40 mila euro)
 - La seconda SUPERVISIONE (matricola del capo dell'impiegato in questione)
 - La terza ancora IMPIEGATI (informazioni relative al capo)
- Bisogna fare il Join di IMPIEGATI con l'espressione precedente
 - Ma siccome in genere capo e impegnato differiscono, si devono cambiare i nomi degli attributi

DB

NomeC	StipC
Marco Celli	60
Sergio Rossi	7
Mario Rossi	65

$$\pi_{\text{NomeC}, \text{StipC}}(\rho_{\text{MatrC}, \text{NomeC}, \text{StipC}, \text{EtàC} \leftarrow \text{Matr}, \text{Nome}, \text{Stip}, \text{Età}(\text{IMPIEGATI}) \bowtie_{\text{MatrC} = \text{Capo}} \text{SUPERVISIONE} \bowtie_{\text{Impiegato} = \text{Matr}} \sigma_{\text{Stipendio} > 40}(\text{IMPIEGATI}))$$

Esempio (4/5)

IMPIEGATI

<u>Matr</u>	Nome	Età	Stipendio
101	Mario Rossi	34	40
103	Mario Bianchi	23	35
104	Luigi Neri	38	61
105	Nico Bini	44	38
210	Marco Celli	49	60
231	Siro Bisi	50	60
252	Nico Bini	44	70
301	Sergio Rossi	34	70
375	Mario Rossi	50	65

SUPERVISIONE

<u>Capo</u>	<u>Impiegato</u>
210	101
210	103
210	104
231	105
301	210
301	231
375	252



Esempio (4/5)

$$R_1 = \sigma_{\text{Stipendio} > 40}(\text{IMPIEGATI})$$

<u>Matr</u>	Nome	Età	Stipendio
104	Luigi Neri	38	61
210	Marco Celli	49	60
231	Siro Bisi	50	60
252	Nico Bini	44	70
301	Sergio Rossi	34	70
375	Mario Rossi	50	65



Esempio (4/5)

$$R_2 = \text{SUPERVISIONE} \bowtie_{\text{Impiegato=Matr}} R_1$$

<u>Matr</u>	Nome	Età	Stipendio	Capo	Impiegato
104	Luigi Neri	38	61	210	104
210	Marco Celli	49	60	301	210
231	Siro Bisi	50	60	301	231
252	Nico Bini	44	70	375	252

Capi degli
impiegati che
guadagnano
più di 40



Esempio (4/5)

$$R_3 = \rho_{\text{MatrC}, \text{NomeC}, \text{StipC}, \text{EtàC} \leftarrow \text{Matr}, \text{Nome}, \text{Stip}, \text{Età}} (\text{IMPIEGATI})$$

<u>MatrC</u>	NomeC	EtàC	StipendioC
101	Mario Rossi	34	40
103	Mario Bianchi	23	35
104	Luigi Neri	38	61
105	Nico Bini	44	38
210	Marco Celli	49	60
231	Siro Bisi	50	60
252	Nico Bini	44	70
301	Sergio Rossi	34	70
375	Mario Rossi	50	65

Esempio (4/5)

$$R_3 = \rho_{\text{MatrC}, \text{NomeC}, \text{StipC}, \text{EtàC} \leftarrow \text{Matr}, \text{Nome}, \text{Stip}, \text{Età}} (\text{IMPIEGATI})$$

MatrC	NomeC	EtàC	StipendioC
101	Mario Rossi	34	40
103	Mario Bianchi	23	35
104	Luigi Neri	38	61
105	Nico Bini	44	38
210	Marco Celli	49	60
231	Siro Bisi	50	60
252	Nico Bini	44	70
301	Sergio Rossi	34	70
375	Mario Rossi	50	65

$$R_2 = \text{SUPERVISIONE} \bowtie_{\text{Impiegato}=\text{Matr}} R_1$$

Matr	Nome	Età	Stipendio	Capo	Impiegato
104	Luigi Neri	38	61	210	104
210	Marco Celli	49	60	301	210
231	Siro Bisi	50	60	301	231
252	Nico Bini	44	70	375	252

Capi degli impiegati che guadagnano più di 40. Otteniamo le info sul capo

$$R_3 = \bowtie_{\text{MatrC}=\text{Capo}} R_2$$

MatrC	NomeC	EtàC	StipendioC	Matr	Nome	Età	Stipendio	Capo	Impiegato
210	Marco Celli	49	60	104	Luigi Neri	38	61	210	104
301	Sergio Rossi	34	70	210	Marco Celli	49	60	301	210
301	Sergio Rossi	34	70	231	Siro Bisi	50	60	301	231
375	Mario Rossi	50	65	252	Nico Bini	44	70	375	252

Esempio (4/5)

R_3

MatrC	NomeC	EtàC	StipendioC	Matr	Nome	Età	Stipendio	Capo	Impiegato
210	Marco Celli	49	60	104	Luigi Neri	38	61	210	104
301	Sergio Rossi	34	70	210	Marco Celli	49	60	301	210
301	Sergio Rossi	34	70	231	Siro Bisi	50	60	301	231
375	Mario Rossi	50	65	252	Nico Bini	44	70	375	252



$\pi_{NomeC, StipendioC}(R_3)$

NomeC	StipendioC
Marco Celli	60
Sergio Rossi	70
Mario Rossi	65

Esempio (5/5)

Trovare gli impiegati che guadagnano più del rispettivo capo, mostrando matricola, nome e stipendio di ciascuno di essi e del capo

- Simile alla precedente

DB

Matr	Nome	Stip	MatrC	NomeC	StipC
104	Luigi Neri	61	210	Marco Celli	60
252	Nico Bini	70	375	Sergio Rossi	7

$$\pi_{\text{Matr}, \text{Nome}, \text{Stip}, \text{MatrC}, \text{NomeC}, \text{StipC}} (\sigma_{\text{Stip} > \text{StipC}} \\ (\rho_{\text{MatrC}, \text{NomeC}, \text{StipC}, \text{EtàC} \leftarrow \text{Matr}, \text{Nome}, \text{Stip}, \text{Età} (\text{IMPIEGATI}) \bowtie_{\text{MatrC} = \text{Capo}} \\ \text{SUPERVISIONE} \bowtie_{\text{Impiegato} = \text{Matr}} \sigma_{\text{Stipendio} > 40} (\text{IMPIEGATI})))$$

Esercizio 8

- Esprimere in algebra relazionale e rappresentare in forma tabellare il risultato delle seguenti interrogazioni
 1. Nome, sede e stipendio degli impiegati che guadagnano più di 1300 Euro, definendo la vista ImpRicchi
 2. Sedi, responsabili e città degli impiegati che guadagnano più di 1300 Euro
 3. Progetti nelle città delle sedi degli impiegati che guadagnano più di 1300 Euro

Imp

CodImp	Nome	Sede	Ruolo	Stipendio
E001	Rossi	S01	Analista	2000
E002	Verdi	S02	Sistemista	1500
E003	Bianchi	S01	Programmatore	1000
E004	Gialli	S03	Programmatore	1000
E005	Neri	S02	Analista	2500
E006	Grigi	S01	Sistemista	1100
E007	Violetti	S01	Programmatore	1000
E008	Aranci	S02	Programmatore	1200

Sedi

Sede	Responsabile	Città
S01	Biondi	Milano
S02	Mori	Bologna
S03	Fulvi	Milano

Prog

CodProg	Città
P01	Milano
P01	Bologna
P02	Bologna

Considerazione

- Una selezione su relazioni con valori nulli produce come risultato le tuple per cui il predicato risulta vero
- Tuttavia comunque dei problemi ci sono
 - $\sigma_{\text{Età} > 30}(\text{PERSONE}) \cup \sigma_{\text{Età} \leq 30}(\text{PERSONE})$
 - Dovrebbe dare tutte le tuple, ma non da quelle con valore NULL su Età
 - Il problema è che si valutano singolarmente le selezioni e non nel suo complesso

Soluzione

- Soluzione: trattare i valori nulli da un punto di vista meramente sintattico
- A tale scopo si introducono due nuove forme di condizioni atomiche di selezione
 - $A \text{ IS NULL}$ assume vero su una tupla t se il valore di t su A è nullo e falso se è specificato
 - $A \text{ IS NOT NULL}$ il viceversa
- $\sigma_{\text{Età} > 30}(\text{PERSONE})$
 - Restituisce persone che hanno *certamente* $\text{Età} > 30$
- $\sigma_{\text{Età} > 30 \vee \text{IS NULL}}(\text{PERSONE})$
 - Restituisce persone che hanno più di trent'anno oppure la cui età non è nota