chiave esterna contiene un riferimento alla relazione stessa. Ad esempio, la relazione `Film` potrebbe contenere come chiave esterna su `Film` stessa una coppia di attributi `(titoloPre,registaPre)`, contenente titolo e regista del film di cui il film è eventualmente il seguito. Come evidenziato dall'Esempio 2.6, inoltre, una relazione può contenere più chiavi esterne, eventualmente anche sulla stessa relazione. Rimarchiamo inoltre che le chiavi esterne, come del resto le chiavi, devono essere esplicitamente specificate in uno schema di relazione. Il fatto di avere attributi con lo stesso nome e domini compatibili in relazioni diverse non offre di per sé alcuna garanzia relativamente al mantenimento dell'integrità referenziale. Notiamo infine che, se non esplicitamente impedito mediante la specifica di un apposito vincolo, le chiavi esterne possono assumere valore nullo.

Per concludere la trattazione del modello dei dati relazionale, sottolineiamo che i vincoli di integrità che il modello permette di rappresentare direttamente, e che abbiamo discusso in questo paragrafo, non sono sufficienti a garantire che il contenuto della base di dati rispecchi in modo fedele le informazioni del dominio applicativo da rappresentare. Discuteremo più in dettaglio nel Capitolo 3 (vedi Paragrafo 3.4) altre categorie di vincoli, oltre ai vincoli di dominio, di obbligatorietà, di chiave e di chiave esterna qui discussi, cui il modello relazionale non offre diretto supporto.

2.2 Algebra relazionale

L'algebra relazionale è costituita da varie operazioni per la manipolazione delle relazioni. Più precisamente, l'algebra è composta da cinque operazioni di base: *proiezione*, *selezione*, *prodotto cartesiano*, *unione* e *differenza*. Queste operazioni definiscono completamente l'algebra relazionale. Ogni operazione ha come argomento una o due relazioni (a seconda dell'operazione) e restituisce come risultato una relazione; è pertanto possibile applicare un'operazione al risultato di un'altra operazione (proprietà detta di *chiusura*). Esistono operazioni addizionali, che possono essere espresse in termini delle cinque operazioni di base. Tali operazioni non estendono il potere espressivo dato dalle cinque operazioni di base, ma sono utili come abbreviazione. Tra le operazioni addizionali, l'operazione detta di *join* è la più rilevante. È da notare che la definizione delle varie operazioni è leggermente diversa a seconda che si consideri la definizione del modello relazionale per posizione o la definizione per nome (vedi Paragrafo 2.1); la differenza principale consiste nel modo in cui vengono denotate le componenti delle tuple in alcune delle operazioni che hanno una o più di tali componenti come ulteriore argomento. Nella trattazione seguente, considereremo la definizione delle operazioni in base alla notazione per nome. In riferimento a tale notazione, viene introdotta un'ulteriore operazione, di *ridenominazione*, che permette di modificare i nomi degli attributi.

2.2.1 Operazioni di base

In quanto segue descriviamo le operazioni di base dell'algebra evidenziando eventuali restrizioni sugli argomenti e lo schema della relazione risultato di ogni operazione. Poiché facciamo riferimento alla notazione con nome, introduciamo innanzitutto l'operazione di ridenominazione.

Ridenominazione. La *ridenominazione* di una relazione R rispetto ad una lista di coppie di nomi di attributi $(A_1, B_1), (A_2, B_2), \ldots, (A_m, B_m)$, tale che $A_i \in U_R$ è un nome di attributo di R, indicata con $\rho_{A_1,A_2,\ldots,A_m \leftarrow B_1,B_2,\ldots,B_m}(R)$, ridenomina l'attributo di nome A_i con il nome B_i, $i = 1, \ldots, m$. La ridenominazione è corretta se il nuovo schema di relazione per R ha attributi con nomi tutti distinti. La relazione ottenuta ha lo stesso grado della relazione R ed ha lo stesso contenuto. Gli attributi della relazione risultato sono $U_R \setminus \{A_1, A_2, \ldots, A_m\} \cup \{B_1, B_2, \ldots, B_m\}$.

Esempio 2.8 Consideriamo la relazione `Noleggio` della Figura 2.2 e supponiamo di voler ridenominare i suoi attributi in `video`, `dataIn`, `cliente`, `dataFine`. L'operazione algebrica per eseguire tale ridenominazione è:

$$\rho_{\texttt{colloc,dataNol,codCli,dataRest} \leftarrow \texttt{video,dataIn,cliente,dataFine}}(\texttt{Noleggio})$$

il cui effetto è la modifica dello schema di `Noleggio` in `Noleggio(video,dataIn, cliente,dataFine)`, senza modificarne il contenuto. □

Proiezione. La *proiezione* di una relazione R su un insieme $A = \{A_1, A_2, \ldots, A_m\} \subseteq U_R$ di nomi di attributi di R, indicata con $\Pi_{A_1,A_2,\ldots,A_m}(R)$, è una relazione di grado m le cui tuple hanno come attributi solo gli attributi specificati in A. Pertanto la proiezione genera un insieme T di tuple con m attributi. Sia $t = [A_1 : v_1, A_2 : v_2, \ldots, A_m : v_m]$ una tupla in T; t è tale che esiste una tupla t' in R tale che $t[A] = t'[A]$. Nella relazione risultato gli attributi compaiono secondo l'ordine specificato in A; è pertanto possibile specificare operazioni di proiezione che permutano gli attributi di una relazione. La proiezione permette di estrarre da una relazione solo alcune delle informazioni in essa contenute, eliminando gli attributi al cui valore non siamo interessati. La relazione risultato, oltre ad avere un grado inferiore a quello della relazione argomento, può anche avere cardinalità inferiore a quella della relazione argomento, poiché eliminando alcuni attributi possono venire generate delle tuple duplicate che compariranno una sola volta nella relazione risultato.

Esempio 2.9 Consideriamo la relazione `Film` e supponiamo di essere interessati solo al titolo ed all'anno di uscita dei film presenti nella videoteca. L'espressione algebrica corrispondente è:

$$\Pi_{\texttt{titolo,anno}}(\texttt{Film})$$

il risultato della cui valutazione sulla relazione della Figura 2.1 è illustrato nella Figura 2.4(a). Supponiamo ora di essere interessati ai generi dei film presenti nella

titolo	anno
underground	1995
edward mani di forbice	1990
nightmare before christmas	1993
ed wood	1994
mars attacks	1996
il mistero di sleepy hollow	1999
big fish	2003
la sposa cadavere	2005
la fabbrica di cioccolato	2005
io non ho paura	2003
nirvana	1997
mediterraneo	1991
pulp fiction	1994
le iene	1992

(a) $\Pi_{\texttt{titolo,anno}}(\texttt{Film})$

genere
drammatico
fantastico
animazione
fantascienza
horror
commedia
thriller

(b) $\Pi_{\texttt{genere}}(\texttt{Film})$

Figura 2.4: Proiezione

videoteca. L'espressione algebrica corrispondente è:

$$\Pi_{\texttt{genere}}(\texttt{Film})$$

il risultato della cui valutazione sulla relazione della Figura 2.1 è illustrato nella Figura 2.4(b). □

Selezione. La *selezione* su una relazione R, dato un predicato F su R, indicata con $\sigma_F(R)$, genera una relazione che contiene tutte le tuple di R che verificano F. Un predicato F su R è un predicato semplice su R, oppure una combinazione booleana, ottenuta mediante gli operatori booleani $\wedge$, $\vee$, $\neg$,[3] di predicati semplici su R. Un predicato semplice su R ha la forma A *op* v oppure A *op* A', dove A ed A' sono nomi di attributi di R i cui domini devono essere compatibili; *op* è un *operatore relazionale di confronto* ed appartiene all'insieme $\{<, =, >, \geq, \leq, \neq\}$;[4] v è un valore costante compatibile con il dominio di A.

Esempio 2.10 I seguenti sono esempi di predicati definiti sulla relazione `Noleggio` introdotta nella Figura 2.2: `codCli=6635`; `dataNol=dataRest`; `codCli=6635` $\vee$ `dataNol=dataRest`. I primi due predicati sono predicati semplici. □

Se il grado della relazione operando R è k, la selezione $\sigma_F(R)$ genera un insieme T di tuple di grado k (sottoinsieme di R). Quindi, lo schema (ed il grado) della relazione risultato sono uguali a quelli della relazione operando. Sia $t = [A_1 :$

[3]Tali operatori sono spesso denotati rispettivamente con `AND`, `OR` e `NOT`.

[4]Nei linguaggi come SQL sono presenti numerosi altri operatori di confronto (vedi Capitolo 3).

(a) $\sigma_{\texttt{codCli}=\texttt{6635}}(\texttt{Noleggio})$

colloc	dataNol	codCli	dataRest
1111	01-Mar-2006	6635	02-Mar-2006
1115	01-Mar-2006	6635	02-Mar-2006
1117	02-Mar-2006	6635	06-Mar-2006
1118	02-Mar-2006	6635	06-Mar-2006
1119	08-Mar-2006	6635	10-Mar-2006
1120	08-Mar-2006	6635	10-Mar-2006
1121	15-Mar-2006	6635	18-Mar-2006
1122	15-Mar-2006	6635	18-Mar-2006
1113	15-Mar-2006	6635	18-Mar-2006
1129	15-Mar-2006	6635	20-Mar-2006
1127	22-Mar-2006	6635	?
1125	22-Mar-2006	6635	?

(b) $\sigma_{\texttt{dataRest}=\texttt{dataNol}}(\texttt{Noleggio})$

colloc	dataNol	codCli	dataRest

Figura 2.5: Selezione

$v_1, A_2 : v_2, \ldots, A_k : v_k]$ una tupla in T; t è tale che $F_{A_1/t[A_1],A_2/t[A_2],\ldots,A_k/t[A_k]}$ è vera, dove la notazione $A_i/t[A_i]$, $i = 1, \ldots, k$, indica la sostituzione in F del nome di attributo A_i (se tale attributo compare in F) con il valore dell'attributo di nome A_i della tupla t. Se nessuna tupla verifica F, il risultato è una relazione vuota. La selezione, quindi, permette di filtrare alcune tuple dalla relazione argomento mediante la specifica di opportune condizioni sui suoi attributi. Il risultato è una relazione la cui cardinalità è minore od uguale a quella della relazione argomento.

Esempio 2.11 Consideriamo la relazione `Noleggio` di Figura 2.2 e supponiamo di essere interessati ai noleggi effettuati dal cliente con codice 6635. L'espressione algebrica corrispondente è:

$$\sigma_{\texttt{codCli}=\texttt{6635}}(\texttt{Noleggio})$$

il risultato della cui valutazione sulla relazione della Figura 2.2 è illustrato nella Figura 2.5(a). Supponiamo ora di essere interessati ai noleggi conclusi lo stesso giorno in cui sono iniziati. L'espressione algebrica corrispondente è:

$$\sigma_{\texttt{dataNol}=\texttt{dataRest}}(\texttt{Noleggio})$$

il risultato della cui valutazione sulla relazione della Figura 2.2 è illustrato nella Figura 2.5(b) e corrisponde ad una relazione vuota. □

Prodotto cartesiano. Il *prodotto cartesiano* di due relazioni R ed S, di grado rispettivamente k_1 e k_2, indicato con $R \times S$, è una relazione di grado $k_1 + k_2$ le cui tuple sono tutte le possibili tuple che hanno:

- come prime k_1 componenti tuple di R;

codCli	nome
6610	anna
6635	paola
6642	marco

(a) $\Pi_{\texttt{codCli,nome}}(\texttt{Cliente})$

titolo	anno
pulp fiction	1994
le iene	1992

(b) $\Pi_{\texttt{titolo,anno}}(\sigma_{\texttt{regista='quentin tarantino'}}(\texttt{Film}))$

codCli	nome	titolo	anno
6610	anna	pulp fiction	1994
6635	paola	pulp fiction	1994
6642	marco	pulp fiction	1994
6610	anna	le iene	1992
6635	paola	le iene	1992
6642	marco	le iene	1992

(c) $\Pi_{\texttt{codCli,nome}}(\texttt{Cliente}) \times \Pi_{\texttt{titolo,anno}}(\sigma_{\texttt{regista='quentin tarantino'}}(\texttt{Film}))$

Figura 2.6: Prodotto cartesiano

- come ultime k_2 componenti tuple di S.

Il prodotto cartesiano può essere applicato solo se le due relazioni R ed S hanno schemi disgiunti.[5] Nella relazione risultato i primi k_1 attributi sono gli attributi della relazione R, gli ultimi k_2 attributi sono gli attributi della relazione S. Il prodotto cartesiano permette di costruire nuove tuple combinando le informazioni presenti nelle tuple delle relazioni argomento. Poiché ogni tupla di R viene combinata con ogni tupla di S, la cardinalità del risultato è il prodotto delle cardinalità degli argomenti.

Esempio 2.12 Consideriamo la relazione $\Pi_{\texttt{codCli,nome}}(\texttt{Cliente})$, contenente i codici ed i nomi dei clienti della videoteca, e la relazione contenente il titolo e l'anno dei film di Quentin Tarantino, cioè $\Pi_{\texttt{titolo,anno}}(\sigma_{\texttt{regista='quentin tarantino'}}(\texttt{Film}))$. Il prodotto cartesiano di queste due relazioni, espresso come:

$$\Pi_{\texttt{codCli,nome}}(\texttt{Cliente}) \times \Pi_{\texttt{titolo,anno}}(\sigma_{\texttt{regista='quentin tarantino'}}(\texttt{Film}))$$

valutato sulle relazioni delle Figure 2.1 e 2.2 produce il risultato illustrato nella Figura 2.6 (c). $\square$

Unione. L'*unione* delle relazioni R ed S, indicata con $R \cup S$, è l'insieme delle tuple che sono in R od in S. L'unione delle relazioni può essere eseguita solo se le relazioni hanno lo stesso schema (cioè stesso grado e, per ogni attributo,

[5] Se le due relazioni hanno attributi con lo stesso nome, è necessario ridenominare gli attributi in una delle due relazioni mediante l'operazione di ridenominazione ρ.

stesso nome[6] e domini compatibili). La relazione risultato ha lo stesso schema delle relazioni argomento. Essendo l'unione un'operazione tra insiemi (di tuple), le tuple duplicate vengono eliminate dal risultato.

Esempio 2.13 Consideriamo $R = \Pi_{\texttt{anno,genere}}(\sigma_{\texttt{regista='tim burton'}}(\texttt{Film}))$ e $S = \Pi_{\texttt{anno,genere}}(\sigma_{\texttt{regista='gabriele salvatores'}}(\texttt{Film}))$, relazioni contenenti anno e genere dei film di Tim Burton e Gabriele Salvatores, rispettivamente. L'unione $R \cup S$ di queste due relazioni, valutata sulle relazioni delle Figure 2.1 e 2.2, produce il risultato illustrato nella Figura 2.7(a), che contiene l'anno di produzione ed il genere dei film di Tim Burton o Gabriele Salvatores. La Figura 2.7(d) mostra invece il risultato dell'espressione $\Pi_{\texttt{genere}}(R) \cup \Pi_{\texttt{genere}}(S)$, che restituisce i generi dei film girati da Tim Burton o Gabriele Salvatores. □

Differenza. La *differenza* delle relazioni R ed S, indicata con $R - S$, è l'insieme delle tuple che sono in R ma non in S. La differenza, come l'unione, di due relazioni può essere eseguita solo se le relazioni hanno lo stesso schema e produce una relazione con lo stesso schema. Se le relazioni hanno attributi con nomi diversi, si applica quanto già detto per l'unione.

Esempio 2.14 Consideriamo nuovamente le relazioni R ed S dell'Esempio 2.13. La differenza $R-S$ di queste due relazioni produce il risultato illustrato nella Figura 2.7(b). La Figura 2.7(e) mostra invece il risultato dell'espressione $\Pi_{\texttt{genere}}(R) - \Pi_{\texttt{genere}}(S)$, che restituisce i generi di cui Tim Burton ha girato un film ma Gabriele Salvatores non ne ha girato. □

2.2.2 Operazioni derivate

Tramite le operazioni di base è possibile definire altre operazioni, dette *operazioni derivate*. Alcune delle più rilevanti sono descritte in quanto segue.

Intersezione. L'*intersezione* di due relazioni R ed S è denotata con $R \cap S$ ed è espressa come $R - (R - S)$. L'intersezione di R ed S restituisce le tuple che sono sia in R che in S. L'intersezione, come l'unione e la differenza di due relazioni, può essere eseguita solo se le relazioni hanno lo stesso schema e produce una relazione con lo stesso schema.

Esempio 2.15 Consideriamo nuovamente le relazioni R ed S dell'Esempio 2.13. L'intersezione $R \cap S$ di queste due relazioni produce il risultato illustrato nella Figura 2.7(c). La Figura 2.7(f) mostra invece il risultato dell'espressione $\Pi_{\texttt{genere}}(R) \cap \Pi_{\texttt{genere}}(S)$, che restituisce i generi di cui sia Tim Burton che Gabriele Salvatores hanno girato un film. □

[6] Se i nomi degli attributi nelle due relazioni sono diversi è sufficiente applicare l'operazione di ridenominazione ρ.

anno	genere
1990	fantastico
1993	animazione
1994	drammatico
1996	fantascienza
1999	horror
2003	fantastico
2005	animazione
2005	fantastico

R

anno	genere
2003	drammatico
1997	fantascienza
1991	commedia

S

anno	genere
1990	fantastico
1993	animazione
1994	drammatico
1996	fantascienza
1999	horror
2003	fantastico
2005	animazione
2005	fantastico
2003	drammatico
1997	fantascienza
1991	commedia

(a) $R \cup S$

anno	genere
1990	fantastico
1993	animazione
1994	drammatico
1996	fantascienza
1999	horror
2003	fantastico
2005	animazione
2005	fantastico

(b) $R - S$

anno	genere

(c) $R \cap S$

genere
fantastico
animazione
drammatico
fantascienza
horror

$GR = \Pi_{\texttt{genere}}(R)$

genere
drammatico
fantascienza
commedia

$GS = \Pi_{\texttt{genere}}(S)$

genere
fantastico
animazione
drammatico
fantascienza
horror
commedia

(d) $GR \cup GS$

genere
fantastico
animazione
horror

(e) $GR - GS$

genere
drammatico
fantascienza

(f) $GR \cap GS$

Figura 2.7: Unione, differenza ed intersezione

codCli	nome	titolo	anno
6610	anna	pulp fiction	1994
6635	paola	pulp fiction	1994
6642	marco	pulp fiction	1994
6610	anna	le iene	1992

(a) $\Pi_{\texttt{codCli,nome}}(\texttt{Cliente}) \bowtie_{\texttt{nome<titolo}} \Pi_{\texttt{titolo,anno}}(\sigma_{\texttt{regista='quentin tarantino'}}(\texttt{Film}))$

titolo	regista
la sposa cadavere	tim burton
la fabbrica di cioccolato	tim burton
big fish	tim burton
le iene	quentin tarantino

(b) $\Pi_{\texttt{titolo,regista}}(\sigma_{\texttt{codCli=6635}\wedge\texttt{dataNol='15-Mar-2006'}}(\texttt{Noleggio}) \bowtie_{\texttt{colloc=c}} \rho_{\texttt{colloc}\leftarrow\texttt{c}}(\texttt{Video}))$

codCli	nome	titolo	anno
6635	paola	le iene	1992
6635	paola	pulp fiction	1994
6642	marco	pulp fiction	1994

(c) $\Pi_{\texttt{codCli,nome,titolo,anno}}(\texttt{Cliente} \bowtie \texttt{Noleggio} \bowtie \texttt{Video} \bowtie \sigma_{\texttt{regista='quentin tarantino'}}(\texttt{Film}))$

Figura 2.8: Theta-join, equi-join e join naturale

Join. Il *join* (detto anche *theta-join*) di due relazioni R ed S sugli attributi A di R ed A' di S, indicato con $R \bowtie_{A\theta A'} S$, dove θ è un operatore relazionale di confronto, è definito dall'espressione algebrica $\sigma_{A\theta A'}(R \times S)$. Il join è pertanto un prodotto cartesiano seguito da una selezione; il predicato $A\theta A'$ è detto *predicato di join*. Come per il prodotto cartesiano, gli schemi delle due relazioni argomento devono essere disgiunti e lo schema della relazione risultato è dato dalla loro unione.

Esempio 2.16 Consideriamo le relazioni dell'Esempio 2.12. Se vogliamo collegare le informazioni di un cliente a quelle di un film solo se il nome del cliente precede in ordine alfabetico il titolo del film, dobbiamo formulare la seguente espressione:

$$\Pi_{\texttt{codCli,nome}}(\texttt{Cliente}) \bowtie_{\texttt{nome<titolo}} \Pi_{\texttt{titolo,anno}}(\sigma_{\texttt{regista='quentin tarantino'}}(\texttt{Film}))$$

che, valutata sulle relazioni delle Figure 2.1 e 2.2, produce il risultato illustrato nella Figura 2.8(a). □

Il join è un'operazione estremamente importante in quanto permette di collegare tuple di relazioni diverse e di attraversare le associazioni rappresentate nelle relazioni della base di dati mediante il meccanismo delle chiavi esterne. In particolare, il join prende il nome di *equi-join* quando l'operatore θ usato nel predicato di join è l'operatore di uguaglianza (=).

Esempio 2.17 Consideriamo le relazioni `Video` e `Noleggio` della base di dati della videoteca. Se vogliamo determinare il titolo ed il regista dei film noleggiati il 15 Marzo 2006 dal cliente di codice 6635 dobbiamo formulare la seguente interrogazione:

$$\Pi_{\texttt{titolo,regista}}(\sigma_{\texttt{codCli=6635}\wedge\texttt{dataNol='15-Mar-2006'}}(\texttt{Noleggio}) \bowtie_{\texttt{colloc=c}} \rho_{\texttt{colloc}\leftarrow\texttt{c}}(\texttt{Video}))$$

che, valutata sulle relazioni delle Figure 2.1 e 2.2, produce il risultato illustrato nella Figura 2.8(b) Notiamo che è necessario applicare la ridenominazione poiché l'equi-join (così come il theta-join ed il prodotto cartesiano) richiede che gli schemi dei suoi argomenti siano disgiunti. □

Join naturale. L'operazione di *join naturale* rappresenta una "semplificazione" del join. È un'operazione che tuttavia, a differenza delle altre finora introdotte, ha senso solo nella notazione con nome del modello relazionale. Il join naturale di due relazioni R ed S, denotato come $R \bowtie S$, è definito come segue. Sia $\{A_1, A_2, \ldots, A_k\} = U_R \cap U_S$ l'insieme dei nomi di attributo presenti sia nello schema di R sia in quello di S. Sia inoltre $\{I_1, I_2, \ldots, I_m\} = U_R \cup U_S$ l'insieme dei nomi di attributo unione dell'insieme dei nomi degli attributi di R e dell'insieme dei nomi degli attributi di S. Siano infine $\{B_1, B_2, \ldots, B_k\}$ nomi di attributo non appartenenti né ad R né ad S. L'espressione che definisce il join naturale è:

$$R \bowtie S = \Pi_{I_1,I_2,\ldots,I_m}(\sigma_C(R \times (\rho_{A_1,A_2,\ldots,A_k \leftarrow B_1,B_2,\ldots,B_k}(S))))$$

dove C è un predicato della forma $A_1 = B_1 \wedge A_2 = B_2 \wedge \ldots \wedge A_k = B_k$. Pertanto il join naturale esegue un join eguagliando gli attributi con lo stesso nome delle due relazioni argomento dell'operazione e poi elimina gli attributi duplicati dalla relazione risultato. Si noti che, affinché l'operazione di join sia ben definita, gli attributi con nomi uguali nelle relazioni argomento dell'operazione devono avere domini compatibili. Notiamo che, nel caso particolare $U_R = U_S$, il join naturale degenera nell'intersezione, mentre, nel caso $U_R \cap U_S = \emptyset$, degenera nel prodotto cartesiano.

Il join naturale permette di navigare in modo molto più agevole, rispetto al join, attraverso le associazioni rappresentate tramite il meccanismo delle chiavi esterne, nel caso in cui per chiave e chiave esterna sia stato utilizzato lo stesso nome.

Esempio 2.18 Consideriamo le relazioni dell'Esempio 2.12. Supponiamo di voler abbinare le informazioni del cliente a quelle del film solo se il cliente ha noleggiato un video di quel film, il che corrisponde a determinare, per ogni cliente che ha noleggiato un film di Quentin Tarantino, nome e codice del cliente e titolo ed anno del film. Dobbiamo formulare l'interrogazione seguente:

$$\Pi_{\texttt{codCli,nome,titolo,anno}}(\texttt{Cliente} \bowtie \texttt{Noleggio} \bowtie \texttt{Video} \bowtie \sigma_{\texttt{regista='quentin tarantino'}}(\texttt{Film}))$$