Note sul modello relazionale

Nicola Vitacolonna
Dipartimento di Matematica e Informatica
Università degli Studi di Udine

8 ottobre 2014

1 Il modello relazionale

Il **modello relazionale** dei dati fu proposto in [**?**] e adottato dai sistemi commerciali a partire dagli anni '80. Attualmente, è il modello piú diffuso nei DBMS, probabilmente grazie alla sua semplicità (l'utente percepisce la base di dati come un insieme di tabelle) e al carattere dichiarativo dei linguaggi di manipolazione e interrogazione associati.

1.1 Nozioni preliminari

Il modello relazionale si fonda sui concetti di "relazione" e "tupla" — variazioni, rispettivamente, delle nozioni matematiche di relazione e n-upla.

Definizione 1.1 *Dati n insiemi $D_1, \ldots, D_n$, una **relazione** R (in senso matematico) su $D_1, \ldots, D_n$ è un sottoinsieme del prodotto cartesiano $D_1 \times \cdots \times D_n$, ossia $R \subseteq D_1 \times \cdots \times D_n$. Il **grado** di R è n. La **cardinalità** $|R|$ di R è il numero di elementi di R.*

Gli elementi di una relazione matematica R sono n-uple ordinate della forma $(a_1, \ldots, a_n)$, dove $a_i \in D_i$ per $i \in \{1, \ldots, n\}$. Nel contesto delle basi di dati, tuttavia, l'ordine degli elementi in una n-upla non è importante: risulta conveniente essere in grado di *nominare* un particolare elemento a_i piuttosto che individuarlo mediante un indice numerico i. Inoltre, le relazioni che occorrono nelle basi di dati hanno cardinalità finita. Diamo quindi le seguenti definizioni.

Definizione 1.2 *Un **dominio** è un insieme non vuoto (finito o infinito) di valori atomici. Un **attributo** è un nome associato a un dominio.*

Indichiamo con $\mathsf{dom}(A)$ il dominio dell'attributo A.

Definizione 1.3 *Uno **schema di tupla** τ è un insieme finito di attributi e dei domini associati, denotato come segue*

$$\tau \stackrel{def}{=} \{A_1 : \mathsf{dom}(A_1), \ldots, A_n : \mathsf{dom}(A_n)\}.$$

Denotiamo con $\mathsf{attr}(\tau)$ l'insieme degli attributi che occorrono nello schema di tupla τ, ossia $\mathsf{attr}(\tau) = \{A_1, \ldots, A_n\}$.

Definizione 1.4 *Una **tupla** t sullo schema $\tau = \{A_i : \mathsf{dom}(A_i) \mid 1 \leq i \leq n\}$ è una funzione $t : \mathsf{attr}(\tau) \to \bigcup_{i=1}^{n} \mathsf{dom}(A_i)$ che associa a ciascun attributo di τ un valore del corrispondente dominio, ossia, per $i \in \{1, \ldots, n\}$,*

$$t(A_i) = v_i, \text{ con } v_i \in \mathsf{dom}(A_i).$$

Una funzione è un caso particolare di relazione matematica, perciò la tupla t, avendo un dominio finito, può anche essere denotata in modo estensionale come segue:

$$t = \{(A_1, v_1), \ldots, (A_n, v_n)\}.$$

Denotiamo con $\mathsf{dom}(t)$ il dominio di t, ossia $\mathsf{dom}(t) = \{A_1, \ldots, A_n\}$, e denotiamo con $t[A_{i_1}, \ldots, A_{i_k}]$ la restrizione di t allo schema $\{A_{i_1} : \mathsf{dom}(A_{i_1}), \ldots, A_{i_k} : \mathsf{dom}(A_{i_k})\}$.

Definizione 1.5 *Uno **schema di relazione** $R(\tau)$ è costituito da un **nome di relazione** R e da uno schema di tupla τ.*

Denotiamo con $\mathsf{attr}(R)$ l'insieme degli attributi che occorrono nello schema di relazione R (dato uno schema $R(\tau)$, ovviamente si ha $\mathsf{attr}(R) = \mathsf{attr}(\tau)$). Per semplicità, dato uno schema di tupla $\{A_1 : \mathsf{dom}(A_1), \ldots, A_n : \mathsf{dom}(A_n)\}$ denotiamo lo schema di relazione

$$R(\{A_1 : \mathsf{dom}(A_1), \ldots, A_n : \mathsf{dom}(A_n)\})$$

con

$$R(A_1 : \mathsf{dom}(A_1), \ldots, A_n : \mathsf{dom}(A_n))$$

o, ancor piú semplicemente, con

$$R(A_1, \ldots, A_n),$$

qualora non vi sia rischio d'ambiguità. Non bisogna dimenticare tuttavia che a uno schema di relazione con n attributi *sono sempre associati n domini* (non necessariamente distinti).

Definizione 1.6 *Una **(istanza di) relazione** (nel contesto delle basi di dati) sullo schema $R(\tau)$ è un insieme finito di tuple su τ.*

Il grado di una relazione r su $R(\tau)$ è il numero di attributi di τ. La cardinalità di r è il numero di tuple che r contiene. Diverse istanze di relazioni su $R(\tau)$ hanno lo stesso grado, ma possono avere cardinalità diverse.

Esempio 1.7 *Applichiamo le definizioni date a un caso limite. Lo schema di tupla piú piccolo (in senso insiemistico) è lo schema vuoto $\epsilon = \emptyset$ (si noti che ϵ è compatibile con la Definizione 1.3, che richiede soltanto che gli schemi siano insiemi* finiti*). Ovviamente,* $\mathtt{attr}(\epsilon) = \emptyset$*. L'unica tupla possibile su ϵ è la tupla vuota, cioè la funzione $e : \emptyset \to \emptyset$ (che scritta in modo estensionale, corrisponde all'insieme vuoto). Lo schema di relazione $R(\epsilon)$ ha allora due sole istanze possibili: $\emptyset$ (l'istanza vuota) ed $\{e\}$ (l'istanza contenente la tupla vuota). Tali istanze hanno grado* 0 *e cardinalità, rispettivamente,* 0 *e* 1.

Osservazione 1.8 *Poiché le relazioni sono insiemi, per definizione in una stessa relazione non possono essere presenti due tuple uguali. Informalmente, possiamo dire che relazioni non ammettono duplicati. Ciò è in contrasto con le implementazioni correnti basate su SQL, in cui invece le tabelle possono contenere duplicati.*

Definizione 1.9 *Siano r e r' due relazioni su $R(\tau)$ e $R'(\tau')$ rispettivamente. Diciamo che r e r' sono* ***compatibili*** *se $\tau = \tau'$.*

Due relazioni sono compatibili se i loro schemi sono identici, ossia hanno gli stessi attributi *associati agli stessi domini.* La nozione di compatibilità tornerà utile quando parleremo di algebra relazionale (si veda la Sezione 2).

Definizione 1.10 *Uno* ***schema di base di dati*** **B** *è un insieme finito di schemi di relazione con nomi diversi:*

$$\mathbf{B} \stackrel{def}{=} \{R_1(\tau_1), \dots, R_n(\tau_n)\}, \text{ con } R_i \neq R_j \text{ per } i \neq j.$$

Si noti che gli schemi di tupla $\tau_1, \dots, \tau_n$ non sono necessariamente distinti.

Definizione 1.11 *Una* ***(istanza di) base di dati*** *su $\{R_1(\tau_1), \dots, R_n(\tau_n)\}$ è un insieme di relazioni $\{r_1, \dots, r_n\}$ tali che r_i è un'istanza di relazione su $R_i(\tau_i)$, per $i \in \{1, \dots, n\}$.*

La definizione completa di uno schema di base di dati include anche la specificazione di un insieme di *restrizioni d'integrità*, di cui parleremo in seguito. La definizione precedente, conseguentemente, sarà raffinata piú avanti.

Esempio. Dato lo schema di base di dati[1]

```
{
 STUDENTE(matricola, nome),
 CORSO(codice, nome),
 ISCRIZIONE(studente, corso)
}
```

[1]Per semplicità, i domini sono stati omessi.

un'istanza di base di dati su tale schema è

$$
\begin{array}{l}
\{ \\
\quad \{\{(\text{matricola}, 37791), (\text{nome}, \text{'Ugo'})\}, \{(\text{matricola}, 37821), (\text{nome}, \text{'Anna'})\}\}, \\
\quad \{\{(\text{codice}, 382), (\text{nome}, \text{'Inf'})\}, \{(\text{codice}, 272), (\text{nome}, \text{'Mat'})\}\}, \\
\quad \{\{(\text{studente}, 37791), \text{corso}, 382)\}, \{(\text{studente}, 37791), (\text{corso}, 272)\}, \\
\quad \{(\text{studente}, 37821), (\text{corso}, 382)\}\} \\
\}
\end{array}
$$

Una rappresentazione piú intuitiva dell'istanza della base di dati può essere data in forma tabulare:

STUDENTE	
matricola	nome
37791	Ugo
37821	Anna

CORSO	
codice	nome
382	Inf
272	Mat

ISCRIZIONE	
studente	corso
37791	382
37791	272
37821	382

Si noti che la relazione ISCRIZIONE consente di individuare gli studenti iscritti ai vari corsi. Questa è una caratteristica fondamentale del modello relazionale: *le corrispondenze tra i dati in relazioni diverse sono stabilite attraverso il confronto tra i valori assunti dai dati stessi* (e non, per esempio, usando puntatori). Approfondiremo oltre questo aspetto.

L'esempio precedente esemplifica la seguente corrispondenza tra il concetto formale di istanza di relazione e il concetto informale di tabella:

Concetto relazionale	Equivalente informale
relazione	tabella
attributo	colonna
grado	numero di colonne
tupla	riga
cardinalità	numero di righe

1.2 Assenza di informazione

La definizione di tupla che abbiamo dato non contempla la situazione in cui qualche dato non sia disponibile. Ad esempio, dato lo schema

IMPIEGATO(cognome, nome, ufficio, telefono, fax),

può accadere che il numero di telefono di qualche impiegato non sia noto (assenza di informazione), oppure che l'impiegato non abbia un fax (valore inesistente), oppure non si sappia se un dato impiegato ha un fax oppure no.

In linea di principio, i tre casi potrebbero essere trattati in modo distinto. La soluzione comunemente adottata nel contesto del modello relazionale, tuttavia, consiste nell'assumere l'esistenza, in ogni dominio, di un valore speciale, chiamato **valor nullo** e denotato da NULL. Ad esempio, un'istanza di impiegato potrebbe essere:

IMPIEGATO				
cognome	nome	ufficio	telefono	fax
Rossi	Paolo	A1	3487	3499
Verdi	Mario	B3	3488	NULL
Bianchi	Lucio	A2	NULL	NULL

Laddove l'informazione non sia disponibile (per qualunque ragione), il valore assunto dall'attributo corrispondente è NULL.

L'introduzione del valor nullo nel modello relazionale ha, come vedremo, una serie di conseguenze rilevanti dovute alla sua semantica particolare.

1.3 Il problema dell'identificazione di tuple

Abbiamo già notato che dati in relazioni diverse possono essere posti in corrispondenza tra loro attraverso l'uguaglianza dei valori: ad esempio, la tupla (37821, 382) nella relazione ISCRIZIONE indica che Anna (la studente con matricola 37821) è iscritta al corso di Informatica (il cui codice è 382). È di fondamentale importanza, dunque, essere in grado di confrontare valori e di identificare univocamente ciascuna tupla di una relazione.

1.3.1 Confronto di valori

Mentre ciò non costituisce un problema in assenza di valori nulli (le relazioni sono *insiemi* di tuple e, pertanto, due tuple in una relazione sono sempre distinguibili), la loro presenza complica la situazione: ad esempio, quale valore di verità dev'essere assegnato a confronti del tipo $v = v'$ o $v \neq v'$ quando v, v' o entrambi i valori sono NULL? La soluzione che adottiamo è essenzialmente quella proposta in [**?**], ed è coerente con il comportamento adottato dalla maggior parte dei DBMS.[2]

Il principio fondamentale è che, se il valor nullo denota un'assenza di informazione, allora il valore di verità appropriato per un confronto che coinvolga un valor nullo è "indeterminato" piuttosto che "vero" o "falso". Di conseguenza,

[2]Mette conto osservare che Codd, nell'articolo citato, sottolinea: *We shall concern ourselves with only the "value at present unknown" type of null [...] The following treatment should be regarded as preliminary and in need of further research.*. Tuttavia, la soluzione proposta è usata oggi senza far distinzione tra le diverse tipologie di assenza d'informazione.

una logica a tre valori sembra appropriata per trattare l'assenza di informazione da un punto di vista semantico. Siano a e b due valori non nulli e distinti appartenenti allo stesso dominio; allora, il valore di verità dei confronti di (dis)uguaglianza è definito dalla seguente tabella:

Espressione	Valore di verità
$a = a$	vero
$a = b$	falso
$a = \text{NULL}$	indefinito
$\text{NULL} = \text{NULL}$	indefinito

Espressione	Valore di verità
$a \neq a$	falso
$a \neq b$	vero
$a \neq \text{NULL}$	indefinito
$\text{NULL} \neq \text{NULL}$	indefinito

La tabella può essere estesa ad altri tipi di confronto. Se D è un dominio su cui è definito un ordinamento lineare stretto $<$ (e, di conseguenza, è possibile definire i corrispondenti ordinamenti $>, \leq, \geq$), allora, per $\theta \in \{<, >, \leq, \geq\}$, il confronto $x\ \theta\ y$ è "indefinito" ogniqualvolta x o y o entrambi i valori sono NULL.

Analogamente, hanno valore di verità "indefinito" le espressioni $\text{NULL} \in R$ e $\{\text{NULL}\} \subseteq R$ per ogni R relazione unaria non vuota.

È altresí utile definire un tipo di confronto puramente "sintattico", che possa assumere solo i valori di verità "vero" e falso" — rispetto al quale, dunque, il valor nullo si comporti come un qualunque altro valore.

Definizione 1.12 *Sia D un dominio. Due elementi $a, b \in D$ sono* ***sintatticamente uguali****, in simboli $a =_{sint} b$, se e solo se si verifica una qualunque delle due seguenti possibilità:*

- *a non è il valor nullo, b non è il valor nullo, e $a = b$;*
- *a e b sono entrambi nulli.*

Se a e b non sono sintatticamente uguali, allora diremo che sono **sintatticamente diversi** e scriveremo $a \neq_{\text{sint}} b$. Rispetto a questa nozione di uguaglianza si ha che $\text{NULL} =_{\text{sint}} \text{NULL}$ e, se a non è il valor nullo, allora $a \neq_{\text{sint}} \text{NULL}$. Questa nozione di uguaglianza è utile nel trattamento dei duplicati all'interno di una relazione, come vedremo nella prossima sezione.

1.3.2 Identificazione di tuple

Intuitivamente, due tuple sono uguali quando assumono gli stessi valori sugli stessi attributi. Avendo dato due diverse definizioni di uguaglianza, una semantica (=) e una sintattica ($=_{\text{sint}}$), per il confronto di valori, conseguentemente possiamo confrontare le tuple in due modi.

Definizione 1.13 *Due tuple t e t' sono (semanticamente)* ***uguali****, in simboli $t = t'$, se e solo se hanno lo stesso dominio D e $t(A) = t'(A)$ è "vero" per ogni attributo $A \in D$; t e t' sono (semanticamente)* ***diverse****, in simboli $t \neq t'$, se e solo se hanno domini diversi, oppure se hanno lo stesso dominio D ed esiste $A \in D$ tale che $t(A) = t'(A)$ è "falso"; t e t' sono* ***inconfrontabili****, in simboli $t||t'$, se e solo se hanno lo stesso dominio D, non esiste alcun attributo A tale che $t(A) = t'(A)$ è "falso" ed esiste $A \in D$ tale che $t(A) = t'(A)$ è "indefinito".*

Esempio. In base alla definizione precedente, nella relazione

PRODOTTO		
nome	colore	prezzo
LCD	nero	100
LCD	NULL	100
LCD	NULL	150

la prima e la seconda tupla sono inconfrontabili perché coincidono sui valori di tutti gli attributi non nulli e il confronto nero = NULL ha valore indeterminato; la prima e la terza tupla sono invece diverse perché differiscono nel valore dell'attributo "prezzo" (lo stesso vale per la seconda e la terza tupla).

Definizione 1.14 *Due tuple t e t' sono* ***sintatticamente uguali****, in simboli $t =_{sint} t'$, se e solo se hanno lo stesso dominio D e $t(A) =_{sint} t'(A)$ per ogni attributo $A \in D$. Due tuple non sintatticamente uguali sono dette* ***sintatticamente diverse****.*

Esempio. Nella relazione PRODOTTO precedente, tutte le tuple sono sintatticamente diverse.

Si lascia per esercizio al lettore verificare che valgono le seguenti implicazioni:

- se $t = t'$ allora $t =_{\text{sint}} t'$;
- se $t \neq t'$ allora $t \neq_{\text{sint}} t'$;
- se $t =_{\text{sint}} t'$, allora $t = t'$ oppure $t||t'$;
- se $t \neq_{\text{sint}} t'$ allora $t \neq t'$ oppure $t||t'$.

Esempio. Date le tuple $t_0 = \{(A,1),(B,2)\}$, $t_1 = \{(A,2),(B,\text{NULL})\}$, $t_2 = \{(A,\text{NULL}),(B,2)\}$, si ha

- $t_0 = t_0$;
- $t_0 \neq t_1$;

- $t_0 || t_2$;
- $t_1 || t_2$;
- $t_2 || t_2$;
- $t_0 \neq_{\text{sint}} t_2$;
- $t_1 \neq_{\text{sint}} t_2$;
- $t_2 =_{\text{sint}} t_2$.

In un insieme non esistono, per definizione, due elementi uguali. Nel caso del modello relazionale, in cui una relazione è un insieme di tuple, dobbiamo assumere che ciò significhi, per definizione, che non esistono due tuple *sintatticamente* uguali nella stessa relazione (se usassimo la nozione di uguaglianza semantica, non sapremmo in generale nemmeno verificare se il dominio delle tuple è effettivamente un insieme, piuttosto che un multi-insieme). Questa è una proprietà desiderabile, perché, data una relazione r su uno schema R, è possibile identificare (sintatticamente) una tupla $t \in r$ senza ambiguità semplicemente elencando i valori assunti da t su (un sottoinsieme di) $\mathtt{attr}(R)$, anche in presenza di valori nulli. Tuttavia, questo tipo d'identificazione sintattica non è sufficiente, perché le associazioni tra i dati sono stabilite sulla base di confronti di tipo semantico (cioè, usando $=$ e non $=_{\text{sint}}$).

Esempio. Le tuple della seguente istanza di relazione sono tutte sintatticamente diverse tra loro:

STUDENTE	
matricola	nome
37428	NULL
NULL	Edgar
NULL	NULL

Tuttavia, le tre tuple sono semanticamente inconfrontabili. In particolare, non è possibile inferire se lo studente Edgar e lo studente 37428 siano la stessa persona, ossia se la prima e la seconda tupla corrispondano in realtà a un unico studente.

L'esempio precedente illustra una situazione che si vorrebbe escludere dal modello relazionale:[3] intuitivamente, affinché una base di dati sia un modello

[3] Si potrebbe argomentare che l'unico modo per escludere effettivamente dal modello una tale istanza sia rifiutare l'esistenza di valor nulli, ma lasciamo tali riflessioni all'iniziativa del lettore.

consistente con la realtà risulta indispensabile poter stabilire una corrispondenza biunivoca tra le tuple di una relazione e corrispondenti entità del mondo reale. Nell'istanza dell'esempio precedente, ciò non è possibile.

Il requisito che uno schema di relazione $R(\tau)$ debba essere una rappresentazione fedele della realtà che modella implica dunque che *non tutti i possibili insiemi finiti di tuple su* τ *siano, in generale, istanze di* $R(\tau)$ *accettabili.* In particolare, è ragionevole ritenere che debba almeno esistere un (sotto)insieme di attributi "identificatori", vale a dire i cui valori consentano di individuare una qualunque tupla di qualunque istanza ammissibile di $R(\tau)$. Ad esempio, l'istanza di STUDENTE dell'esempio precedente non è accettabile perché, nella realtà, il numero di matricola è usato per identificare in modo univoco uno studente nel contesto del sistema informativo di un ateneo e pertanto dev'essere sempre noto e unico per ciascun possibile studente.

Ciò porta alla necessità di individuare per ciascuno schema di relazione le corrispondenti "chiavi", in accordo alla definizione semi-formale seguente. Dato uno schema di relazione R, un insieme di attributi $K \subseteq \mathsf{attr}(R)$ è una **chiave**[4] (o anche, **chiave candidata**) per R se per ogni istanza r di R che sia una rappresentazione corretta della realtà valgono le seguenti proprietà:

Integrità delle entità: non esiste alcuna $t \in r$ tale che, per qualche $A \in K$, $t(A)$ è il valor nullo.

Unicità: per ogni $t, t' \in r$, con $t \neq_{\text{sint}} t'$, si ha $t[K] \neq t'[K]$;

Minimalità: per ogni $K' \subset K$ non vuoto, la precedente proprietà di unicità non sussiste.

La proprietà di unicità garantisce che i valori assunti dagli attributi di una chiave consentano sempre di discriminare due tuple qualunque di una relazione. La proprietà d'integrità delle entità (che è essenzialmente un vincolo di valor non nullo sulla chiave — si veda la Sezione 1.4) assicura la corrispondenza biunivoca con le entità del mondo reale rappresentate dalle tuple.[5] La proprietà di minimalità garantisce che l'insieme di attributi "identificatori" sia il piú piccolo possibile.

La determinazione delle chiavi è un problema di progettazione che non può prescindere dal contesto in cui la base di dati è sviluppata e dipende in modo fondamentale dal *significato* assegnato agli attributi.

Esempio. Una chiave per lo schema di relazione STUDENTE è {matricola}, perché sappiamo che in un'università non possono esistere due studenti con la

[4]La definizione data non è completamente formale dal momento che si appoggia a una nozione di insieme di istanze "consistenti con la realtà"; una definizione formale di "chiave" richiede l'uso delle "dipendenze funzionali", che saranno studiate piú avanti nel corso.

[5]Una tupla una cui chiave contenesse valori nulli rappresenterebbe infatti in modo ambiguo piú entità del mondo reale, o magari nessuna.

stessa matricola. L'attributo "nome" non può invece essere una chiave, perché, in generale, è possibile che esistano due studenti con lo stesso nome. L'unica chiave per ISCRIZIONE è invece {studente, corso}: infatti, assumendo che un corso abbia molti studenti e che uno studente possa seguire molti corsi (entrambe ipotesi del tutto ragionevoli), né {studente} né {corso} possono essere chiavi. Si noti che assunzioni differenti possono portare a determinare chiavi diverse: ad esempio, se all'attributo "corso" di ISCRIZIONE si associa il significato di "corso di laurea", allora {studente} diventa una chiave di ISCRIZIONE (e {studente, corso} non lo è piú), perché uno studente può essere iscritto, in un dato momento, a un solo corso di laurea.

Il modello relazionale impone che il progettista della base di dati scelga, per ciascuno schema di relazione, una chiave tra quelle candidate: tale chiave è detta **chiave primaria** della relazione. Dato uno schema di relazione, gli attributi della chiave primaria tipicamente sono sottolineati. Ad esempio, la chiave primaria di $R(\underline{A}, \underline{B}, C)$ è $\{A, B\}$. La scelta della chiave primaria tra le possibili chiavi candidate è del tutto arbitraria e dettata solitamente da motivi puramente pragmatici o di convenienza.

È lasciato per esercizio al lettore verificare che, se a uno schema R è associata una chiave primaria, allora $t = t'$ se e solo se $t =_{\text{sint}} t'$ per ogni coppia di tuple t, t' di una medesima istanza su R. La nozione di chiave consente quindi di riconciliare le nozioni sintattica e semantica di uguaglianza di tuple.

Concludiamo questa sezione accennando al fatto che la presenza di valori nulli è fonte di molte complicazioni di carattere concettuale, logico e fisico e che la soluzione di distinguere due tipi di uguaglianza (sintattica e semantica) non è completamente soddisfacente. Si consideri ad esempio, la seguente istanza sullo schema R(<u>ufficio</u>, telefono):

R	
<u>ufficio</u>	telefono
21	0432-12345
24	NULL
27	NULL

Vi sono uffici senza telefono? Vi sono uffici che hanno un telefono, ma il cui numero al momento non è noto? Quanti numeri di telefono esistono? Quanti e quali elementi ha la proiezione di R (si veda la Sezione 2.7) sull'attributo "telefono"? Quali sono le chiavi di tale proiezione (dopotutto, la proiezione di una relazione è ancora una relazione)? E cosí via...

1.4 Vincoli d'integrità

La nozione di chiave primaria costituisce un esempio di restrizione sulle possibili istanze di uno schema di relazione. In questa sezione trattiamo le restrizioni d'integrità in maniera sistematica.

Definizione 1.15 *Un **vincolo d'integrità** è una proprietà, associata a uno schema di base di dati, che dev'essere soddisfatta da tutte le istanze che rappresentano informazioni corrette della base di dati.*

I vincoli d'integrità si specificano congiuntamente allo schema della base di dati. È compito del DBMS verificare la consistenza dei dati rispetto ai vincoli specificati. Chiamiamo **intrarelazionali** i vincoli d'integrità che coinvolgono un unico schema di relazione, e **interrelazionali** quelli che coinvolgono piú schemi di relazione.

1.4.1 Vincoli intrarelazionali

I vincoli intrarelazionali piú importanti sono:

- vincoli di tupla: coinvolgono uno o piú attributi di una stessa tupla:
  - vincoli di dominio: restringono i valori attribuibili a un attributo;
  - vincoli su piú valori della stessa tupla;
  - vincoli di valor non nullo;
- vincoli d'unicità: restrizioni che vietano a due tuple distinte di una stessa istanza di coincidere sui valori di un dato sottoinsieme di attributi.

Esempio. Dato lo schema

ESAME(matr, corso, voto, lode)

alcune restrizioni d'integrità possono essere:

- $18 \leq$ voto ≤ 30 (vincolo di dominio);
- lode = sì solo se voto = 30 (vincolo su piú valori della stessa tupla);
- matr non può essere NULL (vincolo di valor non nullo);
- non esistono due tuple che coincidono contemporaneamente sui valori di matr e corso (vincolo d'unicità).

I vincoli di valor non nullo e i vincoli d'unicità sono particolarmente importanti. Definiamo perciò formalmente le circostanze in cui un'istanza di una relazione soddisfa tali vincoli.

Definizione 1.16 *Sia R uno schema di relazione e sia $K \subseteq \mathsf{attr}(R)$, con $K \neq \emptyset$, un insieme di attributi sottoposto a un vincolo di valor non nullo, in simboli*

VNN: K.

Un'istanza r su R ***soddisfa il vincolo di valor non nullo su*** *K se per ogni $t \in r$ e per ogni $A \in K$ si ha che $t(A) \neq_{sint}$* NULL.

Osservazione 1.17 *La coppia di vincoli $\mathcal{V} = \{VNN\colon X, VNN\colon Y\}$ è equivalente al singolo vincolo $\mathcal{W} = \{VNN\colon X \cup Y\}$ per ogni coppia di insiemi non vuoti di attributi X e Y. Qui, "equivalente" significa che $\mathcal{V}$ e $\mathcal{W}$ sono soddisfatti dalle medesime istanze.*

Definizione 1.18 *Sia R uno schema di relazione e sia $K \subseteq \mathsf{attr}(R)$, con $K \neq \emptyset$, un insieme di attributi sottoposto a un vincolo d'unicità, in simboli*

UNI: K.

Un'istanza r su R ***soddisfa il vincolo d'unicità su*** *K se per ogni $t_1, t_2 \in r$ tali che $t_1 \neq t_2$ esiste $A \in K$ tale che $t_1(A) \neq t_2(A)$ oppure $t_1(A) =_{sint}$* NULL *oppure $t_2(A) =_{sint}$* NULL.

Osservazione 1.19 *Si presti particolare attenzione al fatto che, in generale, dati due insiemi di attributi X e Y,*

UNI: X
UNI: Y

non è equivalente a

UNI: $X \cup Y$

Esempio 1.20 *È lasciato al lettore verificare se i vincoli UNI: $\{A, B\}$ e UNI: $\{B, C\}$ sono equivalenti a UNI: $\{A, B, C\}$. Piú in generale, due vincoli d'unicità su insiemi d'attributi con intersezione propria non vuota sono equivalenti a un vincolo d'unicità sull'unione degli attributi?*

Esempio. Si consideri la seguente istanza di relazione sullo schema $R(A, B, C, D)$:

		R	
A	*B*	*C*	*D*
a	*c*	1	3
b	*b*	2	2
c	*b*	NULL	3
d	*a*	NULL	NULL

Allora l'istanza data

- soddisfa il vincolo di valor non nullo su $\{B\}$;
- soddisfa il vincolo d'unicità e di valor non nullo su $\{A\}$;

- soddisfa il vincolo d'unicità e di valor non nullo su $\{A, B\}$;
- soddisfa il vincolo d'unicità su $\{C\}$;
- soddisfa il vincolo d'unicità su $\{A, B, C\}$;
- soddisfa il vincolo d'unicità su $\{C, D\}$;
- soddisfa il vincolo d'unicità su $\{B, D\}$;
- non soddisfa il vincolo di valor non nullo su $\{C\}$;
- non soddisfa il vincolo di valor non nullo su $\{D\}$;
- non soddisfa il vincolo di valor non nullo su $\{C, D\}$;
- non soddisfa il vincolo d'unicità su $\{B\}$.
- non soddisfa il vincolo d'unicità su $\{D\}$.

Selezionare un insieme di attributi K come chiave primaria per uno schema di relazione significa imporre su K sia un vincolo d'unicità sia un vincolo di valor non nullo, perciò non è necessario specificare esplicitamente tali vincoli sulla chiave primaria. È tuttavia necessario imporre tali vincoli su eventuali altre chiavi.

Esempio. Si consideri lo schema di relazione seguente:

PERSONA(carta_identità, cod_fiscale, cognome, nome).

Lo schema ha due chiavi (carta_identità e cod_fiscale). Se carta_identità è scelto come chiave primaria, allora bisogna imporre i vincoli d'unicità e di valor non nullo su cod_fiscale. Lo schema risultante è pertanto il seguente:

PERSONA(<u>carta_identità</u>, cod_fiscale, cognome, nome)
VNN: {cod_fiscale}
UNI: {cod_fiscale}

1.4.2 Vincoli interrelazionali

I vincoli interrelazionali piú importanti sono i vincoli d'integrità referenziale. Intuitivamente, una restrizione d'integrità referenziale richiede che i valori di uno o piú attributi di una relazione occorrano come valori di attributi corrispondenti di un'altra relazione. Poiché le corrispondenze tra relazioni diverse sono stabilite attraverso i valori, risulta chiaro che un vincolo interrelazionale di questo tipo è fondamentale per una corretta rappresentazione dell'informazione.

Esempio. Si considerino le seguenti relazioni:

STUDENTE	
matricola	nome
37791	Ugo
37821	Anna

CORSO	
codice	nome
382	Inf
272	Mat

ISCRIZIONE	
studente	corso
37791	382
36611	272
37821	402

Intuitivamente, la relazione ISCRIZIONE è usata per associare gli studenti ai corsi che frequentano. Nell'istanza precedente, tuttavia, esistono due incongruenze: non esiste alcuno studente con numero di matricola 36611 e non esiste alcun corso con codice 402. L'informazione di questa base di dati non è perciò corretta.

Con riferimento all'esempio precedente, due restrizioni sembrano necessarie: i valori della colonna "studente" di ISCRIZIONE devono essere valori presenti nella colonna "matricola" di STUDENTE, e i valori della colonna "corso" di ISCRIZIONE devono essere valori presenti nella colonna "codice" di CORSO.

La specificazione di un vincolo d'integrità referenziale avviene per mezzo dell'uso di "chiavi esterne". Consideriamo innanzitutto il caso piú semplice di chiavi esterne composte da un solo attributo.

Definizione 1.21 *Siano R e S due schemi di relazione appartenenti a uno schema di base di dati* **B**. *Una* ***chiave esterna (semplice) di*** R ***che fa riferimento a*** S *è definita da*

1. *un attributo $A \in \mathsf{attr}(R)$;*
2. *un attributo $B \in \mathsf{attr}(S)$ con le seguenti proprietà:*
   (a) *UNI: $\{B\}$;*
   (b) *$\mathsf{dom}(B) = \mathsf{dom}(A)$;*
3. *un* ***vincolo d'integrità referenziale*** *sulla coppia (A, B): per ogni istanza di* **B** *tale che r sia istanza di R e s sia istanza di S, e per ogni tupla $t \in r$ tale che $t(A) \neq_{sint}$ NULL, deve esistere $t' \in s$ tale che $t'(B) = t(A)$.*

Denotiamo una chiave esterna con

CE: $A \rightarrow S(B)$

e diciamo che A ***è chiave esterna verso*** *B.*

Si noti che la possibilità di definire una chiave esterna è subordinata a due condizioni:

1. gli attributi coinvolti devono avere lo stesso dominio;

2. l'attributo "destinazione" B (si noti che una chiave esterna non è simmetrica) dev'essere sottoposto a un vincolo d'unicità (il che è automaticamente verificato se B è la chiave primaria di S).

Esempio. Lo schema dell'esempio precedente, come abbiamo visto, richiede due chiavi esterne nella relazione ISCRIZIONE, perciò lo schema di relazione completo è

ISCRIZIONE(<u>studente</u>, <u>corso</u>)
CE: studente → STUDENTE(matricola)
CE: corso → CORSO(codice)

L'istanza dell'esempio precedente viola entrambi i vincoli d'integrità referenziale; l'istanza seguente viola solo la seconda delle due restrizioni:

ISCRIZIONE	
<u>studente</u>	<u>corso</u>
37791	382
37791	272
37821	402

mentre l'istanza seguente soddisfa entrambi i vincoli d'integrità referenziale:

ISCRIZIONE	
<u>studente</u>	<u>corso</u>
37791	382
37791	272
37821	382

Una chiave esterna può essere composta da piú di un attributo: in tal caso è possibile definire il vincolo d'integrità referenziale in tre modi diversi. Diamo la definizione generale di chiave esterna.

Definizione 1.22 *Dati due schemi di relazione R e S appartenenti a uno schema di base di dati* **B**, *una* ***chiave esterna di*** R ***che fa riferimento a*** S *è definita da*

1. *una lista ordinata di attributi $K = (A_1, \ldots, A_k)$ tale che $A_i \in \mathsf{attr}(R)$ per $i \in \{1, \ldots, k\}$;*

2. *una lista ordinata di attributi $J = (B_1, \ldots, B_k)$ tale che $B_k \in \mathsf{attr}(S)$ per $i \in \{1, \ldots, k\}$, con le seguenti proprietà:*

   (a) UNI: $\{B_1, \ldots, B_k\}$;

(b) $\mathsf{dom}(A_i) = \mathsf{dom}(B_i)$ *per ogni* $i \in \{1,\dots,k\}$;

3. *un **tipo di integrità referenziale**, che può essere:*

   (a) ***debole:*** *per ogni istanza di* $\mathbf{B}$ *tale che* r *sia istanza di* R *e* s *sia istanza di* S, *e per ogni* $t \in r$, *si verifica una delle seguenti circostanze:*

   - $t(A_i) =_{sint} \text{NULL}$ *per qualche* $i \in \{1,\dots,k\}$, *oppure*
   - *esiste* $t' \in s$ *tale che* $t(A_i) = t'(B_i)$ *per ogni* $i \in \{1,\dots,k\}$;

   (b) ***parziale:*** *per ogni istanza di* $\mathbf{B}$ *tale che* r *sia istanza di* R *e* s *sia istanza di* S, *e per ogni* $t \in r$, *si verifica una delle seguenti circostanze:*

   - $t(A_i) =_{sint} \text{NULL}$ *per ogni* $i \in \{1,\dots,k\}$, *oppure*
   - $t(A_i) \neq_{sint} \text{NULL}$ *per qualche* $i \in \{1,\dots,k\}$, *ed esiste una tupla* $t' \in s$ *tale che, per ogni* $i \in \{1,\dots,k\}$ *per cui* $t(A_i) \neq_{sint} \text{NULL}$ *si ha* $t(A_i) = t'(B_i)$;

   (c) ***totale:*** *per ogni istanza di* $\mathbf{B}$ *tale che* r *sia istanza di* R *e* s *sia istanza di* S, *e per ogni* $t \in r$, *si verifica una delle seguenti circostanze:*

   - $t(A_i) =_{sint} \text{NULL}$ *per ogni* $i \in \{1,\dots,k\}$, *oppure*
   - $t(A_i) \neq_{sint} \text{NULL}$ *per ogni* $i \in \{1,\dots,k\}$, *ed esiste* $t' \in s$ *tale che* $t(A_i) = t'(B_i)$ *per ogni* $i \in \{1,\dots,k\}$.

Scriviamo

$$CE\colon K \to S(J) \quad (i.r.\ T)$$

dove $T \in \{$*debole, parziale, totale*$\}$, *e diciamo che* K ***è chiave esterna verso*** J ***con tipo d'integrità referenziale*** T.

Tipicamente (ma non sempre), gli attributi di J costituiscono la chiave primaria di S.

I tre tipi di integrità referenziale (i.r.) corrispondono alle tre seguenti proprietà:

i.r. debole: se in una tupla di un'istanza di R tutti i valori degli attributi in K sono diversi dal valor nullo, allora deve esistere una tupla di un'istanza di S che assuma gli stessi valori nei corrispondenti attributi in J.

i.r. parziale: se in una tupla di R qualche attributo in K assume un valore diverso dal valor nullo, allora deve esistere una tupla di S che assuma gli stessi valori negli attributi di J corrispondenti agli attributi non nulli di K.

i.r. totale: ogni tupla di R soddisfa una delle due seguenti proprietà: tutti gli attributi in K assumono valor nullo oppure sono tutti non nulli ed esiste una tupla di S che assume gli stessi valori negli attributi corrispondenti in J.

È lasciato per esercizio al lettore verificare che i tre tipi d'integrità referenziale debole, parziale e totale, sono equivalenti in ciascuna delle seguenti situazioni:

- K e J sono costituiti da un solo attributo (nel qual caso si ottiene la Definizione 1.21);
- tutti gli attributi della chiave esterna sono sottoposti a un vincolo di valor non nullo.

Esempio (i.r. debole). Si consideri il seguente schema:

COMUNE(nome, provincia, abitanti)

CITTADINO(carta_identità, nome, cognome, comune, prov)

CE: (comune, prov) → COMUNE(nome, provincia) (i.r. debole)

Allora la seguente è un'istanza valida dello schema:

COMUNE		
nome	provincia	abitanti
Martignacco	UD	25324
Nimis	UD	12834
Sistiana	TS	17980

CITTADINO				
carta_identità	nome	cognome	comune	prov
CI1234	Bacco	Ramandulis	Nimis	UD
CI3857	Italo	Svevo	NULL	TS
CI2633	Regina	Neri	NULL	GO
CI7723	Eolo	Vento	Sistiana	TS
CI9822	Arcadio	Buendia	Macondo	NULL
CI9828	Ezechiele	Lupo	NULL	NULL

Esempio (i.r. parziale). Si consideri il seguente schema:

COMUNE(nome, provincia, abitanti)

CITTADINO(carta_identità, nome, cognome, comune, prov)

CE: (comune, prov) → COMUNE(nome, provincia) (i.r. parziale)

Allora l'istanza dell'esempio precedente non è piú valida, mentre la seguente è un'istanza valida dello schema:

COMUNE		
nome	provincia	abitanti
Martignacco	UD	25324
Nimis	UD	12834
Sistiana	TS	17980

CITTADINO				
carta_identità	nome	cognome	comune	prov
CI1234	Bacco	Ramandulis	Nimis	UD
CI3857	Italo	Svevo	NULL	TS
CI7723	Eolo	Vento	Sistiana	TS
CI3427	Ada	Pascal	Martignacco	NULL
CI9828	Ezechiele	Lupo	NULL	NULL

Esempio (i.r. totale). Si consideri il seguente schema:

COMUNE(nome, provincia, abitanti)

CITTADINO(carta_identità, nome, cognome, comune, prov)

CE: (comune, prov) → COMUNE(nome, provincia) (i.r. totale)

Allora l'istanza dell'esempio precedente non è piú valida, mentre la seguente è un'istanza valida dello schema:

COMUNE		
nome	provincia	abitanti
Martignacco	UD	25324
Nimis	UD	12834
Sistiana	TS	17980

CITTADINO				
carta_identità	nome	cognome	comune	prov
CI1234	Bacco	Ramandulis	Nimis	UD
CI7723	Eolo	Vento	Sistiana	TS
CI9828	Ezechiele	Lupo	NULL	NULL

1.4.3 Altri vincoli d'integrità

Le restrizioni analizzate nelle sezioni precedenti non esauriscono tutte le possibilità. Useremo il linguaggio naturale per esprimere le restrizioni che non rientrano nei casi precedenti.

Esempio. Si consideri il seguente schema:

DIPARTIMENTO(<u>cod</u>, nome)

PROFESSORE(<u>id</u>, nome, cod_dip)
CE: cod_dip → DIPARTIMENTO(cod)

"A ogni dipartimento deve afferire almeno un professore"

È lasciato per esercizio al lettore verificare che la restrizione d'integrità: "a ogni dipartimento deve afferire almeno un professore" non è esprimibile usando i vincoli delle sezioni precedenti.

Concludiamo la descrizione delle strutture dati del modello relazionale dando una definizione piú completa di (schema di) base di dati.

Definizione 1.23 *Uno* ***schema di base di dati*** *è composto da:*

- *un insieme di schemi di relazione con nomi diversi:*
- *le definizioni delle chiavi primarie di ciascuno schema;*
- *un insieme di ulteriori vincoli d'integrità.*

Definizione 1.24 *Una* ***(istanza di) base di dati*** *su* $\{R_1(\tau_1),\ldots,R_n(\tau_n)\}$ *(con i vincoli associati) è un insieme di relazioni* $\{r_1,\ldots,r_n\}$ *tali che*

- r_i *è un'istanza di relazione su* $R_i(\tau_i)$*;*
- r_i *soddisfa i vincoli d'integrità associati a* $R_i(\tau_i)$*.*

2 Algebra relazionale

Introduciamo ora un linguaggio per l'interrogazione di una base di dati, chiamato *algebra relazionale*. L'algebra relazionale è costituita da un insieme di operatori unari e binari su istanze di relazioni finite: ciascun operatore ha come argomenti una o due istanze di relazione e produce una nuova istanza di relazione. Si tratta di un linguaggio operazionale: un'espressione dell'algebra relazionale specifica la sequenza di operazioni necessaria per ottenere il risultato voluto. Distinguiamo operatori di due tipi:

- Operatori insiemistici:
  - unione
  - intersezione
  - differenza
  - prodotto cartesiano

- Operatori propriamente relazionali:
  - ridenominazione
  - selezione
  - proiezione
  - concatenazione (*join*)
  - divisione

Non tutti i precedenti operatori sono primitivi: l'intersezione, la concatenazione e la divisione, come vedremo, possono essere definiti mediante i restanti operatori.

2.1 Ridenominazione

L'operatore di ridenominazione ρ semplicemente cambia i nomi degli attributi di una relazione. Poiché una relazione è un insieme di tuple, ciò significa che la ridenominazione modifica i domini delle tuple. Data una relazione r sullo schema $R(A_1,\ldots,A_n)$, la **ridenominazione** $\rho_{A_i\to B_i,\ldots,A_j\to B_j}(r)$ **di** $A_i,\ldots,A_j$ con $B_i,\ldots,B_j$ rispettivamente è definita come segue:

$$\begin{aligned}\rho_{A_i\to B_i,\ldots,A_j\to B_j}(r) = \{\, &\{(A_1,v_1),\ldots,(B_i,v_i),\ldots,(B_j,v_j),\ldots,(A_n,v_n)\} \mid \\ &\{(A_1,v_1),\ldots,(A_i,v_i),\ldots,(A_j,v_j),\ldots,(A_n,v_n)\} \in r\,\}\end{aligned}$$

Esempio. Data la relazione

CORSO	
codice	nome
382	Inf
272	Mat

la ridenominazione di "codice" in "sigla" produce la relazione[6]

$\rho_{\text{codice}\to\text{sigla}}$(CORSO)	
sigla	nome
382	Inf
272	Mat

[6]Qui e nel seguito, con abuso di notazione, applicheremo gli operatori dell'algebra relazionale ai nomi degli schemi di relazione. Formalmente, dovremmo scrivere $\rho_{\text{codice}\to\text{sigla}}(r)$, con r istanza di CORSO, invece di $\rho_{\text{codice}\to\text{sigla}}$(CORSO). Ribadiamo tuttavia il concetto: l'algebra relazionale è un calcolo su *istanze* di relazioni.

2.2 Unione

Date due relazioni r_1 e r_2 compatibili, l'operatore di **unione** $\cup$ produce una relazione che contiene le tuple che appartengono a una delle due relazioni o a entrambe:

$$r_1 \cup r_2 \stackrel{\text{def}}{=} \{\, t \mid t \in r_1 \vee t \in r_2 \,\}$$

Esempio. Date le due relazioni seguenti

PUBBLICAZIONE		
autore	titolo	sommario
A. Bee	On DBs	…

SEMINARIO		
relatore	titolo	sommario
W. Zed	Unions	…

la loro unione è la relazione

PUBBLICAZIONE $\cup\, \rho_{\text{relatore}\to\text{autore}}$(SEMINARIO)		
autore	titolo	sommario
A. Bee	On DBs	…
W. Zed	Unions	…

Si noti la necessità di ridenominare un attributo per rendere le relazioni compatibili.

2.3 Differenza

Date due relazioni r_1 e r_2 compatibili, l'operatore di **differenza** $\setminus$ produce una relazione che contiene le tuple che appartengono alla prima relazione ma non alla seconda:

$$r_1 \setminus r_2 \stackrel{\text{def}}{=} \{\, t \mid t \in r_1 \wedge t \notin r_2 \,\}$$

Esempio. Date le relazioni seguenti

PIANO_DI_STUDI	
cod_corso	nome
A2	Matematica
C3	Basi di dati

ESAMI_SUPERATI	
cod_corso	nome
A2	Matematica

la differenza tra la prima e la seconda è

PIANO_DI_STUDI \ ESAMI_SUPERATI	
cod_corso	nome
C3	Basi di dati

2.4 Intersezione

Date due relazioni r_1 e r_2 compatibili, l'operatore di **intersezione** $\cap$ produce una relazione che contiene le tuple che appartengono a entrambe le relazioni:

$$r_1 \cap r_2 \stackrel{\text{def}}{=} \{\, t \mid t \in r_1 \wedge t \in r_2 \,\} = r_1 \setminus (r_1 \setminus r_2)$$

Si osservi che l'intersezione può essere definita usando la differenza.

Esempio. Data le relazioni seguenti:

GARAGE	
automobile	<u>proprietario</u>
106	Mario
C3	Antonio

CARROZZERIA	
<u>proprietario</u>	automobile
Antonio	C3
Sergio	307

la loro intersezione è

GARAGE ∩ CARROZZERIA	
automobile	proprietario
C3	Antonio

2.5 Prodotto cartesiano

Siano $R_1(\tau_1)$ e $R_2(\tau_2)$ due schemi di relazione tali che $\mathtt{attr}(\tau_1) \cap \mathtt{attr}(\tau_2) = \emptyset$. Date due istanze r_1 e r_2 di R_1 e R_2 rispettivamente, l'operatore di **prodotto cartesiano** $\times$ produce una relazione formata da tutte le tuple che è possibile ottenere unendo le tuple di r_1 e r_2:

$$r_1 \times r_2 \stackrel{\text{def}}{=} \{\, t \cup t' \mid t \in r_1 \wedge t' \in r_2 \,\}$$

Si noti che il prodotto cartesiano è definito solo per relazioni i cui schemi non condividono alcun attributo. La relazione $r_1 \times r_2$ è una relazione sullo schema $\tau_1 \cup \tau_2$.

Esempio. Date le relazioni

CORSO	
<u>codice</u>	nome
123	Musica
321	Latino
121	Geometria

AULA	
<u>num</u>	piano
A10	1
A20	2

il loro prodotto cartesiano è la relazione

CORSO × AULA			
codice	nome	num	piano
123	Musica	A10	1
123	Musica	A20	2
321	Latino	A10	1
321	Latino	A20	2
121	Geometria	A10	1
121	Geometria	A20	2

2.6 Selezione

L'operatore di selezione consente di selezionare le tuple di una relazione che soddisfano una determinata condizione. Le condizioni sintatticamente corrette sono combinazioni booleane di confronti e di un predicato speciale NULL() usato per verificare se un valore è nullo.

Definizione 2.1 *Siano A e B attributi, e sia $a \in \mathsf{dom}(A)$ tale che $a \neq_{sint}$ NULL. Un **confronto** è una delle seguenti espressioni:*

1. $A = B$, $A \neq B$, $A < B$, $A > B$, $A \leq B$, $A \geq B$ *(confronti tra attributi);*

2. $A = a$, $A \neq a$, $A < a$, $A > a$, $A \leq a$, $A \geq a$ *(confronti tra un attributo e un valore non nullo);*

3. NULL(A) *(verifica di valor nullo).*

Definizione 2.2 *Una **condizione** è definita induttivamente in accordo alle seguenti regole:*

1. *ogni confronto è una condizione;*

2. *se φ è una condizione, allora $\neg\varphi$ è una condizione;*

3. *se φ e ψ sono condizioni, allora $\varphi \wedge \psi$ e $\varphi \vee \psi$ sono condizioni.*

Per quanto riguarda la semantica di una condizione, adottiamo una logica a tre valori, cosicché la valutazione di una condizione può essere "vero", "falso" oppure "indefinito". Data una condizione φ denotiamo con $\varphi(t)$ il valore di verità assunto da φ quando φ è interpretata usando i valori di t.

Definizione 2.3 *Siano A e B attributi, $a \in \mathsf{dom}(A)$ tale che $a \neq_{sint}$ NULL, φ un confronto, t una tupla e $\alpha \in \{=, \neq, <, >, \leq, \geq\}$ un operatore di confronto. Il valore di verità di un confronto è dato dalle seguenti regole:*

1. *se φ è $A\,\alpha\,B$, allora $\varphi(t)$ è il valore di verità di $t(A)\,\alpha\,t(B)$;*

2. *se φ è $A\,\alpha\,a$, allora $\varphi(t)$ è il valore di verità di $t(A)\,\alpha\,a$;*

3. *se φ è* NULL(A)*, allora $\varphi(t)$ è "vero" se $t(A)$ è* NULL*; è "falso" altrimenti.*

Il valore di verità degli operatori booleani è riassunto nelle seguenti tabelle:

φ	ψ	$\varphi \wedge \psi$	$\varphi \vee \psi$
falso	falso	falso	falso
indefinito	falso	falso	indefinito
vero	falso	falso	vero
falso	indefinito	falso	indefinito
indefinito	indefinito	indefinito	indefinito
vero	indefinito	indefinito	vero
falso	vero	falso	vero
indefinito	vero	indefinito	vero
vero	vero	vero	vero

φ	$\neg\varphi$
falso	vero
indefinito	indefinito
vero	falso

Date una relazione r e una condizione φ, l'operatore di **selezione** σ produce una relazione costituita dalle tuple t di r tali che $\varphi(t)$ è "vero":

$$\sigma_{\varphi}(r) \stackrel{\text{def}}{=} \{\, t \mid t \in r \wedge \varphi(t) \}$$

Esempio. Data la seguente relazione:

ESPERIMENTO		
cod	T	output
7362	32	$8.9 \cdot 10^{-6}$
7362	27	NULL
982	27	$3.4 \cdot 10^{-4}$
2933	38	$5.5 \cdot 10^{-7}$

le tuple tali che l'output sia minore o uguale a 10^{-5} sono date dalla seguente selezione:

$\sigma_{\text{output} \le 10^{-5}}$(ESPERIMENTO)		
cod	T	output
7362	32	$8.9 \cdot 10^{-6}$
2933	38	$5.5 \cdot 10^{-7}$

Si noti che, a causa del carattere trivalente della logica impiegata, la relazione $\sigma_{\text{output} \leq 10^{-5} \vee \text{output} > 10^{-5}}(\textsc{Esperimento})$:

$\sigma_{\text{output} \leq 10^{-5} \vee \text{output} > 10^{-5}}(\textsc{Esperimento})$		
cod	T	output
7362	32	$8.9 \cdot 10^{-6}$
982	27	$3.4 \cdot 10^{-4}$
2933	38	$5.5 \cdot 10^{-7}$

non coincide con ESPERIMENTO.

2.7 Proiezione

Sia r un'istanza di uno schema di relazione R e sia $K \subseteq \mathsf{attr}(R)$. L'operatore di **proiezione** π produce una relazione le cui tuple corrispondono alle tuple di r il cui dominio è ristretto a K:

$$\pi_K(r) \stackrel{\text{def}}{=} \{\, t[K] \mid t \in r \,\}$$

Esempio. Data la relazione

AGENDA			
<u>nome</u>	età	indirizzo	telefono
Augusto	30	…	463782
Davide	56	…	393883
Geremia	98	…	338843

la proiezione di AGENDA su nome è

$\pi_{\text{nome}}(\textsc{Agenda})$
nome
Augusto
Davide
Geremia

La proiezione di AGENDA su età e telefono è

$\pi_{\text{età,telefono}}(\textsc{Agenda})$	
età	telefono
30	463782
56	393883
98	338843

2.8 Concatenazione naturale

Date due relazioni r_1 e r_2, l'operatore di concatenazione naturale[7] produce una relazione che le cui tuple sono ottenute "componendo" le tuple di r_1 e r_2 che hanno gli stessi valori sugli attributi comuni.

Siano $R_1(\tau_1)$ e $R_2(\tau_2)$ due schemi di relazione, e sia $J = \mathsf{attr}(R_1) \cap \mathsf{attr}(R_2)$ l'insieme (eventualmente vuoto) degli attributi comuni a R_1 e R_2. Siano inoltre r_1 un'istanza di R_1 e r_2 un'istanza di R_2 e si supponga che $\pi_J(r_1)$ e $\pi_J(r_2)$ siano compatibili.[8] La **concatenazione naturale** $\bowtie$ di r_1 e r_2 è

$$r_1 \bowtie r_2 \stackrel{\text{def}}{=} \{\, t \cup t' \mid t \in r_1 \wedge t' \in r_2 \wedge t[J] = t'[J] \,\}.$$

La relazione $r_1 \bowtie r_2$ è una relazione sullo schema $\tau_1 \cup \tau_2$. È lasciato per esercizio al lettore verificare che

- la concatenazione può essere definita in funzione di ridenominazione, proiezione, prodotto cartesiano e selezione;
- la concatenazione è associativa e commutativa;
- Se r_1 e r_2 non hanno attributi comuni, allora $r_1 \bowtie r_2 = r_1 \times r_2$;
- Se r_1 e r_2 sono compatibili, allora $r_1 \bowtie r_2 = r_1 \cap r_2$.

Esempio. Date le relazioni

LIBRO

ISBN	titolo	autore
88-11-11111-1	Il cane dei Baskerville	A. C. Doyle
88-22-22222-2	I delitti della Rue Morgue	E. A. Poe
88-33-33333-3	La bottiglia di Amontillado	E. A. Poe
88-55-55555-5	Ossi di seppia	E. Montale

e

BIOGRAFIA

autore	data_di_nascita
E. A. Poe	1809
E. Montale	1896
G. Leopardi	1798

[7] La specificazione "naturale" deriva dal fatto che in algebra relazionale è possibile definire altri tipi di concatenazione, che non tratteremo.

[8] Gli schemi di $\pi_J(r_1)$ e $\pi_J(r_2)$ hanno ovviamente gli stessi attributi; qui si richiede che gli attributi siano anche associati agli stessi domini.

la loro concatenazione naturale è

LIBRO ⋈ BIOGRAFIA			
ISBN	titolo	autore	data_di_nascita
88-22-22222-2	I delitti della Rue Morgue	E. A. Poe	1809
88-33-33333-3	La bottiglia di Amontillado	E. A. Poe	1809
88-55-55555-5	Ossi di seppia	E. Montale	1896

2.9 Divisione

Siano $R_1(\tau_1)$ e $R_2(\tau_2)$ due schemi di relazione tali che $\mathsf{attr}(\tau_2) \subset \mathsf{attr}(\tau_1)$ e siano r_1 e r_2 istanze di R_1 e R_2 rispettivamente. Si supponga inoltre che $\pi_{\mathsf{attr}(\tau_2)}(r_1)$ e r_2 siano compatibili. L'operatore di **divisione** $\div$ produce una relazione le cui tuple, se estese con una qualunque tupla della seconda relazione, producono una tupla della prima relazione, ossia:

$$\begin{aligned} r_1 \div r_2 &\overset{\text{def}}{=} \{\, t \mid \text{per ogni } t' \in r_2 \text{ si ha } t \cup t' \in r_1 \,\} = \\ &= \{\, t \mid \{t\} \times r_2 \subseteq r_1 \,\} \end{aligned}$$

La relazione $r_1 \div r_2$ è definita sullo schema $\tau_1 \setminus \tau_2$. La divisione è un operatore derivato; può essere infatti definita come segue:

$$r_1 \div r_2 = \pi_K(r_1) \setminus \pi_K\big((\pi_K(r_1) \times r_2) \setminus r_1\big)$$

dove $K = \mathsf{attr}(\tau_1 \setminus \tau_2)$.

Intuitivamente, la divisione è la controparte algebrica della quantificazione universale. Il seguente esempio illustra questo punto di vista.

Esempio. Date le relazioni

ISCRIZIONE		
matr	data	corso
123	10-12-05	BDD
283	10-12-05	BDD
123	20-12-05	INF
123	21-12-05	MAT
283	21-12-05	MAT
375	15-01-06	BDD
283	20-12-05	INF

SESSIONE	
data	corso
10-12-05	BDD
20-12-05	INF
21-12-05	MAT

la divisione di ISCRIZIONE per SESSIONE è

ISCRIZIONE ÷ SESSIONE
matr
123
283

Il risultato della divisione consiste nelle matricole iscritte a *tutte* le sessioni d'esame. La divisione corrisponde dunque, da un punto di vista logico, a una quantificazione universale.