



PEGASO
Università Telematica



Indice

1. INTRODUZIONE.....	3
2. IL MODELLO LOGICO DEI DATI.....	4
2.1 IL MODELLO GERARCHICO	4
2.2 IL MODELLO RETICOLARE	6
2.3 IL MODELLO RELAZIONALE	7
3. LA RELAZIONE.....	9
3.1 RELAZIONE MATEMATICA	9
3.2 LA <i>RELAZIONE</i> NEL MODELLO RELAZIONALE	10
4. LA TABELLA.....	11
BIBLIOGRAFIA	12

1. Introduzione

In questa unità didattica viene introdotto il modello relazionale dei dati. Inizialmente si illustra il concetto di modello logico dei dati. Successivamente si illustrano i due concetti di relazione e tabella, alla base del modello relazionale, con esempi di relazioni.

2. Il modello logico dei dati

La modellazione dei dati è un processo utilizzato quindi per definire e analizzare i requisiti dei dati necessari per supportare i processi aziendali nell'ambito dei corrispondenti sistemi informativi nelle organizzazioni. Pertanto, il processo di modellazione dei dati coinvolge progettisti che lavorano a stretto contatto con gli stakeholder aziendali, nonché i potenziali utenti del sistema informativo.

Un modello logico dei dati è un modello che non è specifico di un database e che descrive gli oggetti e i fatti sui quali un'organizzazione desidera raccogliere dati e le relazioni tra essi. Un modello logico contiene rappresentazioni di entità e attributi, relazioni, identificatori univoci, sottotipi e supertipi e vincoli tra le relazioni. Esso può anche contenere oggetti del modello di dominio o fare riferimento a uno o più modelli di domini o glossari. Dopo che gli oggetti logici e le relazioni sono stati definiti in un modello logico dei dati, è possibile utilizzare un ambiente per trasformare il modello logico in una rappresentazione fisica specifica del database sotto forma di un modello fisico dei dati. La fase di modellazione logica dei dati si concentra sulla descrizione delle entità e delle relazioni. La modellazione logica dei dati viene svolta indipendentemente dai requisiti e dalle funzionalità della piattaforma di database sottostante.

In letteratura si hanno tre tipologie di modelli logici *tradizionali*:

- Gerarchico
- Reticolare
- Relazionale

Mentre tra i più recenti modelli logici proposti in letteratura, si hanno:

- Modello a oggetti (poco diffuso)
- Modello basato su XML ("complementare" a quello relazionale)
- Modello NoSQL

2.1 Il modello gerarchico

Il modello di database gerarchico prevede che i dati siano organizzati secondo strutture ad albero, che si suppone riflettano una gerarchia esistente tra le entità (un esempio di entità è l'entità "Fornitori") che appartengono al database. Ogni albero è formato da un unico record radice (detto anche *segmento o padre*) e da un insieme di uno o più sotto-alberi dipendenti da esso, costruiti in modo simile. Ciascuna struttura ad albero costituisce, quindi, un insieme organizzato di segmenti o *rami* strutturati del database, in forma gerarchica. Al livello più alto (la radice della struttura ad albero)

può apparire solo una tipologia di entità (ad esempio "Fornitore X"). Tutte le altre tipologie di entità, possono avere un solo "padre", mentre il numero dei "figli" è variabile¹.

Il modello gerarchico consente di rappresentare informazioni usando la relazione tra segmenti "padre" e segmenti "figli": ogni padre può avere molti figli, ma ogni figlio può avere un solo padre. In questo caso si parla di relazioni 1:N, chiamate anche relazioni uno-a-molti.

Per fare un esempio di modello gerarchico, prendiamo un'organizzazione con un archivio di dipendenti memorizzati in una tabella chiamata appunto "Dipendenti". Nella tabella si trovano attributi come Nome, Cognome, Sesso, Livello, Reparto. Immaginiamo che l'organizzazione abbia dati sui figli dei dipendenti, memorizzati in una tabella "Figli" con attributi come Nome, Cognome, Sesso, Data di nascita. La tabella Dipendenti rappresenta il segmento padre e la tabella Figli il segmento figlio. I due segmenti formano una gerarchia dove un impiegato può avere diversi figli, mentre un figlio può avere un solo padre.

Un altro esempio di modello gerarchico è dato dai file system attualmente in uso sulla totalità dei computer presenti in aziende, uffici e abitazioni. Il modello gerarchico era ampiamente usato nei primi DBMS per mainframe, mentre successivamente i DBMS gerarchici sono stati sostituiti dai DBMS reticolari e in seguito da quelli relazionali chiamati anche RDBMS. Le relazioni di tipo gerarchico rendono molto semplice rispondere alle interrogazioni quando la realtà rappresentata è appunto gerarchica, ma molto difficile rispondere quando le relazioni rappresentate sono di tipo multi-a-molti (ad esempio nel caso classico di fornitori che vendono diversi prodotti e prodotti venduti da diversi fornitori)².

¹ https://it.wikipedia.org/wiki/Modello_gerarchico

² https://it.wikipedia.org/wiki/Modello_gerarchico



Figura 1: esempio di modello gerarchico.

La caratteristica del modello gerarchico di ammettere un solo genitore per ogni nodo, crea un problema nel descrivere l'associazione tra i record di Clienti e i record di Conti a causa del record relativo al numero di conto 5100. Questo record ha due genitori: il record dei clienti di nome Gialli e quello di nome Verdi. Per rispettare l'unicità del genitore del record di Conti si deve costruire l'albero di Figura 1.

2.2 Il modello reticolare

Un database reticolare consiste di due insiemi di dati, un insieme di record e un insieme di legami: i tipi record sono fatti di campi, mentre i legami sono realizzati collegando i record delle entità associate in entrambe le direzioni. Non esiste quindi una gerarchia predefinita tra le entità. La differenza principale rispetto al modello gerarchico consiste nel fatto che un record figlio può avere un numero qualsiasi di padri: in questo modo vengono evitate situazioni di ripetizione di dati uguali. Il problema illustrato in precedenza, in Figura 1 con il modello gerarchico, può essere rappresentato nel modello reticolare secondo lo schema di Figura 2.

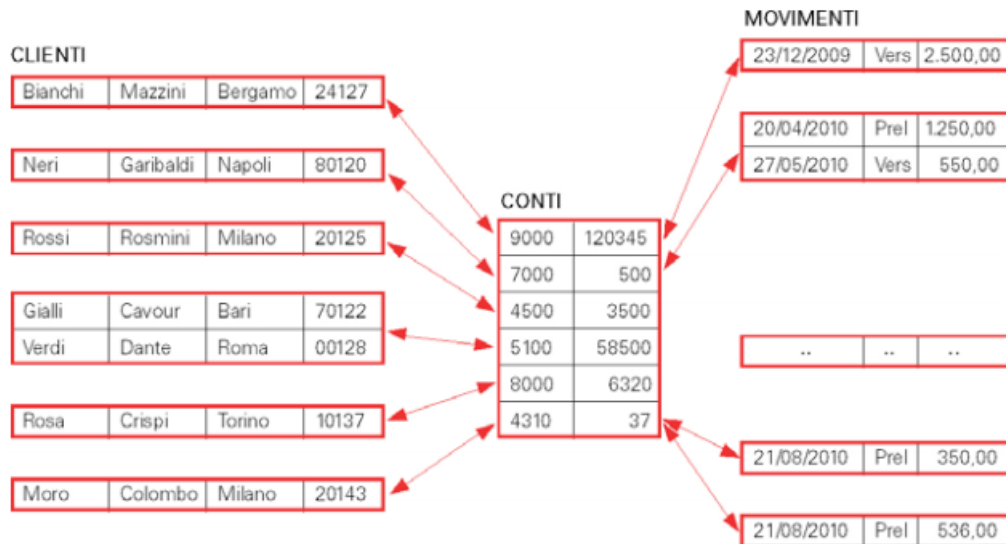


Figura 2: un esempio di modello reticolare.

2.3 Il modello relazionale

Questo modello, proposto da E. F. Codd **nel 1970** per favorire l'indipendenza dei dati, è disponibile in DBMS reali dal 1981 (non è facile implementare l'indipendenza con efficienza e affidabilità). Esso si basa sul concetto matematico di **relazione** (con una variante), la quale ha naturale rappresentazione per mezzo di **tabelle**.

Infatti, questo modello organizza i dati in una o più tabelle (o "relazioni") formate da colonne e righe, con una **chiave** univoca che identifica ciascuna riga (sul concetto di chiave torneremo in seguito). Le righe sono anche chiamate record o **tuple**. Le colonne sono anche chiamate **attributi**.

Matricola	Nome	Cognome	Data di nascita
01234	Roberto	Rossi	10-2-2000
03457	Antonio	Verdi	4-7-201
09876	Giuseppe	Gialli	11-1-2000

Figura 3: esempio di tabella/relazione.

Generalmente, ogni tabella/relazione rappresenta un tipo di entità (come cliente o prodotto). Le righe rappresentano istanze di quel tipo di entità mentre le colonne rappresentano valori attribuiti a tale istanza, come nell'esempio di Figura 3.

Il modello relazionale specifica che le tuple di una relazione non hanno un ordine specifico e che le tuple, a loro volta, non impongono alcun ordine sugli attributi. Le applicazioni accedono ai dati specificando le query, che utilizzano operazioni SQL come SELECT per identificare tuple. Le relazioni possono essere modificate tramite l'inserimento, l'eliminazione e l'aggiornamento mentre nuove tuple possono fornire valori espliciti o essere derivate da una query. Allo stesso modo, le query identificano le tuple per l'aggiornamento o l'eliminazione.

3. La relazione

La Relazione è alla base del modello relazionale dei dati. Esso si basa sul concetto di **Relazione**.

Esistono però tre accezioni di tale termine:

- Accezione 1: **Relazione matematica**: come nella teoria degli insiemi;
- Accezione 2: Relazione secondo il **modello relazionale** dei dati;
- Accezione 3: Relazione (dall'inglese **relationship**) che rappresenta una classe di fatti, nel modello Entity-Relationship; tradotto anche con associazione o correlazione.

Nei prossimi paragrafi approfondiremo tali definizioni.

3.1 Relazione matematica

Dati n insiemi, anche non distinti D_1, \dots, D_n , si forma il prodotto cartesiano $A = D_1 \times \dots \times D_n$, ottenendo in tal modo l'insieme di tutte le n -uple (d_1, \dots, d_n) tali che $d_1 \in D_1, \dots, d_n \in D_n$. Si definisce **relazione matematica** su D_1, \dots, D_n un sottoinsieme R di $D_1 \times \dots \times D_n$. D_1, \dots, D_n sono i **domini** della relazione R . In Figura 4 un esempio di relazione matematica.



Figura 4: un esempio di relazione matematica.

Quindi, una relazione matematica è un insieme che gode delle seguenti proprietà:

- è un insieme di n -uple ordinate (d_1, \dots, d_n) tali che: $d_1 \in D_1, \dots, d_n \in D_n$.
 - non c'è ordinamento fra le n -uple
 - le n -uple sono distinte

ciascuna n -upla è ordinata: l' i -esimo valore proviene dall' i -esimo dominio. In Figura 5 un esempio di relazione matematica chiamata *Partite*, che useremo in seguito.

Da quanto detto ne consegue che la struttura è di tipo **posizionale**, nel senso che le n-ple nella tabella devono rispettare l'ordine dei domini.

$\text{Partite} \subseteq \text{string} \times \text{string} \times \text{int} \times \text{int}$

Juve	Lazio	3	1
Lazio	Milan	2	0
Juve	Roma	0	2
Roma	Milan	0	1

Figura 5: la relazione matematica "Partite".

3.2 La Relazione nel modello relazionale

Il concetto di Relazione nel modello relazionale dei dati è leggermente diverso da quello matematico, anche se ne rappresenta il punto teorico di partenza. A ciascun dominio si associa un nome unico nella tabella (attributo), che ne descrive il "ruolo", come illustrato in Figura 6, riprendendo la relazione *Partite*.

Casa	Fuori	RetiCasa	RetiFuori
Juve	Lazio	3	1
Lazio	Milan	2	0
Juve	Roma	0	2
Roma	Milan	0	1

Figura 6: la relazione "Partite".

In questa accezione, la relazione è una struttura di tipo **non posizionale**, nel senso che, avendo identificato ciascuna colonna con il ruolo, anche scambiando le colonne, le informazioni rimangono comunque coerenti.

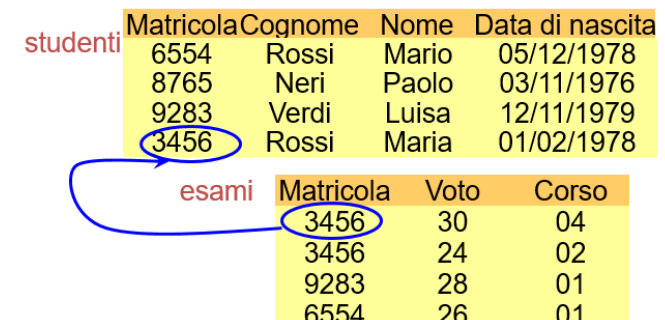
4. La Tabella

Una relazione, nel modello relazionale, viene rappresentata da una tabella, ovvero da una struttura formata da righe, le tuple, e colonne, gli attributi. Da quanto detto, ne consegue che in una tabella che rappresenta una relazione:

- l'ordinamento tra le righe è irrilevante;
- l'ordinamento tra le colonne è irrilevante.

Una tabella rappresenta una relazione se e solo se:

- Le righe sono diverse fra loro;
- Le intestazioni delle colonne sono diverse tra loro;
- I valori di ogni colonna sono fra loro omogenei;
- I riferimenti fra dati in relazioni diverse sono rappresentati per mezzo di valori dei domini che compaiono nelle n-uple. Un esempio di tale importante proprietà è illustrato in Figura 7, dove la due relazioni Studenti ed Esami, sono collegate tramite l'attributo Matricola.



studenti	Matricola	Cognome	Nome	Data di nascita
	6554	Rossi	Mario	05/12/1978
	8765	Neri	Paolo	03/11/1976
	9283	Verdi	Luisa	12/11/1979
	3456	Rossi	Maria	01/02/1978

esami	Matricola	Voto	Corso
	3456	30	04
	3456	24	02
	9283	28	01
	6554	26	01

Figura 7: un esempio di collegamento tra relazioni attraverso valori.

I vantaggi di basare le relazioni ed i collegamenti tra esse attraverso valori, presenta i seguenti vantaggi:

- Indipendenza dalle strutture fisiche (si potrebbe avere anche con puntatori di alto livello) che possono cambiare dinamicamente;
- Si rappresenta solo ciò che è rilevante dal punto di vista dell'applicazione;
- L'utente finale vede gli stessi dati dei programmatori;
- I dati sono portabili più facilmente da un sistema ad un altro;
- I puntatori sono direzionali.

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.