

Database (Base di dati)

Cos'è un database

Per comprendere appieno cos'è un database e quali sono i vantaggi legati al suo impiego, è necessario definire in modo esatto e preciso cosa si intende per:

- Database
- DBMS (Database Management System).

Definizione di database:

Un Database può essere definito come un insieme di informazioni strettamente correlate e memorizzate su un supporto di memoria di massa, costituenti un tutt'uno, che possono essere manipolate da più programmi applicativi.

Il data base, pertanto, è costituito da dati.

Finora abbiamo concepito i dati come file, record, campi, registrazioni su disco, ecc. Adesso interpretiamo questi concetti in modo differente.

Se prima il file era visto come l'unità fondamentale dell'elaborazione dei dati, adesso questo tipo di raggruppamento perde importanza a favore della base di dati, ovvero della **collezione globale** dei dati inerenti un certo soggetto.

Esempio:

Si pensi ad un'agenzia bancaria. L'insieme dei dati relativi ai clienti, alle transazioni, ai conti, ai dipendenti, ecc. costituiscono la base dati. Ciò significa che anche se le entità dei dati sono differenti tra loro (dati che riguardano i dipendenti sono di altro genere, ad esempio, dai dati che riguardano i clienti o i conti bancari) essi comunque appartengono alla medesima realtà, ovvero alla realtà di quella particolare agenzia bancaria.

Definizione di DBMS (Database Management System):

Un DBMS è un sistema software per la gestione di basi dati; esso si occupa dell'aggiornamento, della manutenzione e della consultazione di un insieme di registrazioni contenute in un supporto di memoria di massa.

Il DBMS, pertanto, è un insieme di programmi, che sono rivolti alla gestione di dati memorizzati in archivi. Ovviamente, tra Database e DBMS esiste una forte iterazione per cui spesso si tende a confonderli, ma sono comunque due cose ben diverse e distinte.

Esempio:

Si pensi ad un bambino che sta giocando con delle conchiglie disponendole in un certo ordine, le conchiglie (entità che vengono manipolate) rappresentano il database, il bambino (colui che opera) rappresenta il DBMS.

Livelli di astrazione di un DBMS

Abbiamo accennato al fatto che il DBMS ha tra le sue funzioni, quella di mantenere un modello astratto dei dati che consenta all'utente di concepire le informazioni secondo strutture riferite direttamente alle entità reali che i dati sono preposti a rappresentare.

Abbiamo inoltre visto che spetta sempre al DBMS, per mezzo del Database Manager, tradurre le operazioni ordinate dall'utente sulle strutture astratte in operazioni sui file.

Vediamo adesso quali sono i livelli di astrazione consentiti.

Livello fisico

Il livello fisico del database è rappresentato dalle strutture di memoria di massa usate per conservare i dati e per accedervi in modo rapido ed efficiente.

È necessario distinguere tra:

- I dati veri e propri
- Le strutture che li contengono e che ci permettono di accedere ai medesimi

Il livello fisico è, per l'utente, del tutto trasparente: egli, infatti, non si preoccupa affatto di come i dati vengano registrati sui supporti, tale funzione è compito esclusivo del DBMS. L'utente si occuperà principalmente del cosa vi è registrato: quali sono i dati e in quale relazione si trovano tra loro.

Essendo la gestione dei file realizzata dal sistema, l'utente non deve interessarsi della forma assunta dai dati in memoria di massa; non gli resta dunque che occuparsi del valore informativo dei dati.

Livello concettuale

Il livello concettuale rappresenta la struttura globale del database, relativa a tutte le informazioni in esso presenti, rappresentate in modo organizzato tramite un modello astratto.

Il modello è composto da *entità*, a ciascuna entità saranno riferite informazioni, rappresentate da *attributi* inerenti all'entità medesima. Si crea quindi un rapporto biunivoco tra un'entità del mondo reale, che costituisce l'oggetto di riferimento, e un insieme di dati omogenei e coerenti all'interno del database. Questi dati riporteranno le caratteristiche dell'oggetto a cui si riferiscono, realizzando per l'appunto un processo di *astrazione* dal livello reale al livello concettuale.

Un'entità è costituita da ciascun *oggetto* al quale si vogliono riferire informazioni: sono entità gli studenti di una scuola, i dipendenti di un'azienda, i correntisti di un istituto di credito, i libri di una biblioteca. A ciascuna entità fanno riferimento dati ben precisi: nome, data di nascita, reparto, classe, titolo, autore, ecc. Gli attributi sono collegati alle entità, costituendo un oggetto di potenziale informativo maggiore rispetto ad una collezione di informazioni non correlate.

Esempio:

Prendiamo in esame l'entità STUDENTI

STUDENTI (nome, classe, sezione)

Il dominio di <nome> sarà costituito da una stringa di caratteri, il dominio <classe> da un intero compreso tra 1 e 5, il dominio di <sezione> da un carattere alfabetico limitato alle sole sezioni presenti nella scuola.

Livello esterno

Una volta definito lo schema concettuale, è possibile limitare l'accesso ai dati ad alcuni utenti. Quindi sono realizzabili accessi e navigazioni personalizzate.

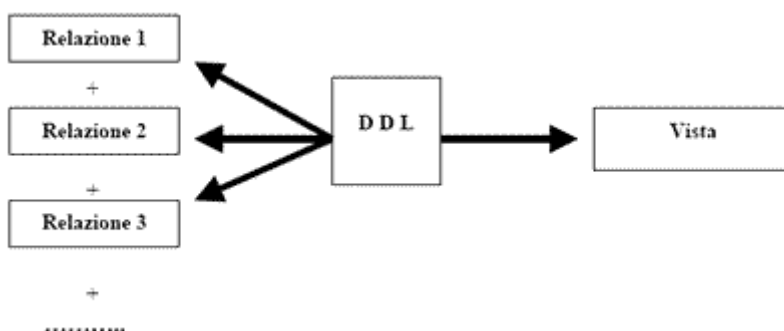
Ciò è reso possibile dalle funzioni del DBMS relative al *livello* esterno che danno la sensazione a ciascuno di essere l'unico utilizzatore dei dati che vede.

Accesi e navigazioni si attuano attraverso schemi detti *viste logiche* dei dati.

Definizione di vista:

Una **vista** è l'**astrazione** di una parte del data base concettuale che coinvolge i dati dell'istanza del data base limitatamente alla **porzione interessata**.

Attraverso il DDL, l'utente può descrivere viste logiche dei dati che abbiano tra loro informazioni appartenenti a entità diverse, creando così *nuove entità informative*.



Si può inoltre limitare l'accesso in *orizzontale*, cioè a una sola parte degli attributi di una entità, oppure in *verticale*, condizionando il dominio di uno o più attributi. In questo modo, si verrà a disporre di una nuova configurazione della base dei dati, rispondente in modo completo alle esigenze di quel utente.

Esempio:

Tornando alla entità di cui sopra:

STUDENTI (nome, classe, sezione)

Qualora ci interessi una stampa dei nomi, dei soli studenti che frequentano la classe quinta, è possibile descrivere una vista logica dei dati limitando il dominio di <classe> al valore 5 e gli attributi a <nome>. Se invece si desidera il listato dei soli alunni della sezione E, si limiterà il dominio di <sezione> a E appunto.

In sostanza, la vista logica è un modo di vedere, accedere e navigare sui dati che sono stati descritti in precedenza dallo schema concettuale e tramite questo registrati nell'istanza del data base a livello fisico.

A questo punto siamo in grado di riassumere quanto detto per i tre livelli:

- *Il livello fisico* riguarda l'effettiva memorizzazione dei dati, organizzati in file, record e strutture di accesso.
- *Il livello concettuale* riguarda la struttura logica assunta dai dati registrati, quindi il loro schema astratto.
- *Il livello esterno* si riferisce al modo in cui ciascun utente può vedere gli stessi dati, che mantengono l'organizzazione fisica e concettuale precedentemente descritte, ma vengono messe a disposizione secondo il formato richiesto.

Modello logico

Il modello logico discende dal modello concettuale e disegna un'architettura che tiene conto delle strutture proprie di quel particolare tipo di database. Ciò significa che è possibile realizzare diversi tipi di database a partire da uno stesso modello concettuale.

Il modello relazionale

Il modello relazionale, fu proposto per la prima volta, nel 1970. Fin da allora ha avuto un crescente successo, dovuto principalmente alla sua semplicità e alla sua flessibilità.

Su cosa si basa il modello relazionale:

Il *modello relazionale* si basa su due concetti *relazione* e *tabella*. La nozione di relazione proviene dalla teoria degli insiemi, mentre il concetto di tabella è semplice ed intuitivo.

La struttura del modello relazionale è molto semplice e potente. Al tempo stesso, essa impone però un certo grado di rigidità.

AUTO

Targa	Proprietario	Indirizzo
RM 1A2396	Verdi Piero	Via Tigli
FI 1A2300	Verdi Piero	Via Tigli
TV 2F4560	Bianchi Antonio	Via Tigli

PROPRIETARI AUTO

Persona	Indirizzo
Verdi Piero	Via Tigli
Bianchi Antonio	Via Tigli

Normalizzazione dei database

Per eliminare definitivamente la ridondanza dalle relazioni e quindi dal database, si ricorre ad un processo di normalizzazione: si tratta di un procedimento di tipo graduale, che realizza un'ottimizzazione progressiva a partire da relazioni non normalizzate fino a raggiungere un certo livello di normalizzazione.

Prima di entrare in merito alla normalizzazione è bene introdurre alcuni particolari concetti:

Definizione di ridondanza:

Si ha ridondanza dei dati ogni volta che vengono memorizzati inutilmente dei dati ripetuti.

Esempio:

Si supponga di disporre di una tabella in cui vengono registrati tutti gli acquisti di merci effettuati da un'impresa mercantile.

Acquisti (Cod_fornitore, Nome_fornitore, Codice_fiscale, Merce, Quantità)

In un caso come questo i dati di Fornitore e Codice_fiscale vengono ripetuti inutilmente ogni volta che compare la registrazione di un acquisto. Si può ovviare ad una tale problema separando le informazioni ridondanti della tabella acquisti:

Acquisti (Merce, Cod_fornitore, Quantità)

Fornitori (Cod_fornitore, Nome_fornitore, Codice_fiscale)

In questo modo se gli acquisti effettuati da un certo fornitore compaiono mille volte, non dovremo per mille volte ripetere inutilmente l'informazione Nome_fornitore, Codice_fiscale.

Linguaggio di interrogazione di un DBMS: SQL

SQL non identifica un prodotto commerciale, ma un linguaggio nello stesso modo in cui C e Basic indicano linguaggi generali e non compilatori particolari.

È un linguaggio, che serve per eseguire varie operazioni sia sui dati che sulle strutture che li contengono. La sigla, acronimo di *Structured Query Language*, è ormai diventata sinonimo di

Il termine SQL può generare confusione. La lettera S, iniziale di Structured (Strutturato), e la lettera L, iniziale di Language, sono abbastanza semplici, ma la lettera Q si presta a varie interpretazioni. Ovviamente Q sta per Query che se fosse interpretata alla lettera, limiterebbe il linguaggio SQL a uno strumento per interrogare il database. In effetti SQL fa molto di più che porre delle domande.