



L2 - Modelli e Astrazioni

Created	@September 22, 2022
Class	Basi di Dati
Materials	<u>L02.pdf</u>

Evoluzione del DBMS

- Dal monolitico al modulare e distribuito con architetture di tipo client-server
- Gli attuali ambienti di cloud computing consistono di migliaia di server dislocati geograficamente che gestiscono enormi moli di dati (Big Data) per conto degli utenti
- I sistemi moderni hanno almeno due livelli: un livello client e un modello server

Sistemi Client-Server

Nelle architetture client-server si hanno:

- Un modulo client, progettato per essere eseguito sul pc dell'utente
 - Spesso fornito di GUI user friendly
- Un modulo server che si occupa di permettere l'accesso ai dati del client

Modelli dei dati

I modelli dei dati servono per l'astrazione dei dati, e consistono di tre parti:

1. Struttura della base di dati (tipi dei dati, i vincoli e le relazioni)
2. Le operazioni per manipolare le strutture (inserzione, cancellazione, modifica, ricerca)
3. Un comportamento dinamico (o aspetto) per permettere all'utente di definire operazioni sui dati

Un modello deve:

- Rappresentare una certa realtà (es: mappe)
- Fornire un insieme di strutture simboliche per descrivere la rappresentazione della realtà (es: simboli per interpretare le mappe)

I modelli possono essere di diversi tipi in base alla fase di progettazione in cui ci troviamo

Astrazione

Procedimento mentale che sostituisce con un concetto un insieme di oggetti in base ad alcune loro proprietà

Es: Consente di definire il concetto di “automobile”. Grazie a esso è possibile descrivere e riconoscere tutte le automobili della realtà

L’astrazione è importante perché permette di comunicare ed elaborare una rappresentazione del miniworld

Per comunicare ed elaborare tale rappresentazione si utilizzano modelli concettuali e logici:

- Un modello concettuale fornisce simbolismi per rappresentare concetti astratti in modo indipendente dagli elaboratori
- Un modello logico traduce le strutture concettuali in logiche processabili da un DBMS

Categorie di data model

ALTO LIVELLO:

I data model di alto livello o concettuali forniscono concetti che sono vicini al modo di percepire i dati degli utenti utilizzando concetti quali entità, attributi e relazioni:

- Un’entità è un concetto da rappresentare del mondo reale
- Un attributo rappresenta qualche proprietà importante
- Una relazione rappresenta un’interazione tra le entità

BASSO LIVELLO:

I data model di basso livello o fisici forniscono concetti che descrivono dettagli su come i dati sono memorizzati

- Un percorso di accesso è una struttura che rende efficiente la ricerca

- Un indice è un esempio di percorso di accesso che garantisce l'accesso diretto ai dati

RAPPRESENTAZIONALI:

I data model rappresentazionali o implementazione forniscono concetti comprensibili agli utenti finali, ma che non sono troppo lontani dal modo in cui i dati sono fisicamente organizzati

- Relazionale → Incapsula i dati in strutture tabellari fisse
- Reticolare → Utilizzo di grafi, record e set
- Gerarchico → Utilizza strutture ad albero con i record come i nodi e le associazioni come archi

AUTODESCRITTIVI:

I data model auto-descrittivi combinano la descrizione dei dati con i valori dei dati stessi

Schemi e istanze di database

- Schemi → Descrizione di una base di dati corrispondente alla struttura logica del database (intensione della base di dati)
 - Detta anche “metadati”
 - Viene specificato in fase di progetto e non si cambia frequentemente
 - Per rappresentare uno schema si crea un diagramma di schema, che visualizza la struttura dei record ma non le reali istanze dei record
- Istanza → Contenuto del database in un particolare istante di tempo (estensione della base di dati)

La distinzione tra schema e istanza è molto importante:

- Nella creazione del DB è nello stato di “vuoto”
- Si passa nello “stato iniziale” quando si inseriscono i dati per la prima volta
- Da questo momento in poi ogni modifica del DB lo porterà in un nuovo stato

Architettura a tre livelli

Approccio:

- Supporto di viste multiple

- Uso di un catalogo per la descrizione del DB
- Isolamento tra programmi e dati

L'architettura a tre livelli aiuta a ottenere tali caratteristiche, definendo il database in tre livelli e separando quindi le applicazioni utente dal database fisico:

- Livello interno → Descrive le strutture di storing fisiche
 - Usa un data model fisico e descrive i dettagli completi del data storage e gli access paths del DB
- Livello concettuale → Descrive la struttura e i vincoli del database
 - Nasconde i dettagli delle strutture di storage fisico e si concentra sulla descrizione delle entità
- Livello esterno → Descrive le varie viste per gli utenti
 - Definisce un sottoinsieme del DB per una particolare applicazione

I mapping tra i livelli dell'architettura

I tre schemi sono delle descrizioni di dati: gli unici dati che realmente esistono sono a livello fisico

In un dbms basato sull'architettura a tre livelli, ogni gruppo di utenti utilizza una propria vista esterna

Un mapping è un processo di trasformazione delle richieste e dei risultati. Si trasforma una richiesta da un livello a quello successivo

Indipendenza dei dati

- Logica → Indica la capacità di cambiare lo schema concettuale senza dover cambiare lo schema esterno e gli applicativi correlati
 - Può essere cambiato per espandere oppure per ridurre il database
 - Se si elimina un tipo di record gli schemi esterni che si riferiscono solo ai dati restanti non devono essere alterati
 - Più complicata da realizzare, perché si dovrebbe apportare delle modifiche alla struttura dei dati e ai vincoli
- Fisica → Indica la capacità di cambiare lo schema interno senza dover cambiare lo schema concettuale

- Un cambiamento dello schema interno può essere dovuto alla riorganizzazione di qualche file fisico per migliorare l'esecuzione del ritrovamento o dell'aggiornamento
 - Si realizza quasi sempre perché i dati di basso livello sono più nascosti agli utenti
-

Architetture per i DBMS

In un primo momento si avevano sistemi mainframe, ai quali gli utenti accedevano con dei terminali “dumb”, i quali fornivano solo operazioni di visualizzazione

- Tutte le funzionalità del DBMS, l'esecuzione del programma e l'elaborazione dell'interfaccia utente sono eseguite su una sola macchina

In un secondo momento si è andato verso architetture modulari client-server, dove una macchina funge da fornitore di servizi e le altre accedono e forniscono strumenti di visualizzazione

- I server mantengono e gestiscono i file delle macchine dei client
- I client accedono ai dati nel server
- Architettura Two-Tier → Architettura logica e architettura fisica
 - Nel DBMS introduciamo tecniche per gestire le query e farle viaggiare su canali di comunicazione
- Architettura Three-Tier → Strati completamente separati (dati-utente-interfacce)