

Lezione I

Introduzione e Modellazione Concettuale

Tipi di Database ed Applicazioni

- Database Numerici e Testuali
- Database Multimediali
- Geographic Information Systems (GIS)
- Data Warehouses
- Real-time and Active Databases

Definizioni

- **Database:** Un insieme di dati correlati.
- **Dati:** Fatti noti che possono essere memorizzati e che hanno un significato intrinseco.
- **Mini-mondo:** Un certo aspetto del mondo reale (i dati del quale vengono archiviati in un database). Per esempio, i voti degli studenti agli esami, i libri di una biblioteca, ...
- **Database Management System (DBMS):** Programma (o insieme di programmi) che permette la creazione e la manutenzione di un database elettronico.
- **Database System:** Il DBMS ed i dati contenuti nel database. A volte anche le applicazioni (esterne al DBMS) per interagire con i dati.

Funzioni di un DBMS

- **Definire** un database: in termini di tipi di dati, strutture e vincoli
- **Costruire** un database: immagazzinare i dati in un mezzo di memorizzazione
- **Manipolare** un database: eseguire interrogazioni per recuperare dati, aggiornare, modificare e cancellare i dati, generare report,...
- **Condividere** un database: consentire a più utenti o applicazioni di accedere contemporaneamente ai dati (mantenendo i dati validi e consistenti)

Funzioni di un DBMS

Altre caratteristiche:

- Protezione o misure di sicurezza per prevenire accessi non autorizzati ai dati
- Procedure “attive” per eseguire operazioni sui dati internamente
- Sistemi di visualizzazione e presentazione dei dati

Esempio di database

- **Minimondo:** una parte dell'ambiente universitario
- **Entità:** ESERCIZIO ALLA LAVAGNA (PENSATECI ANCHE VOI!)
- **Associazioni:** ESERCIZIO ALLA LAVAGNA (PENSATECI ANCHE VOI!)

NOTA: Quanto sopra (e ciò che abbiamo scritto sulla lavagna) può essere espresso in termini di modello ER (Entità/Associazione)

Caratteristiche dell'approccio con database

- Natura autodescrittiva di un database system
- Isolamento tra programmi e dati
- Astrazione dei dati
- Supporto di viste multiple dei dati
- Condivisione dei dati e gestione delle transazioni con utenti multipli

Caratteristiche dell'approccio con database

- Natura autodescrittiva di un database system:
Esiste un **catalogo** in cui il DBMS memorizza una descrizione del database. Questa descrizione viene chiamata **meta-data**.
Ciò consente al DBMS di lavorare con diversi database.
Differenza enorme rispetto all'utilizzo di file.

Caratteristiche dell'approccio con database

- Isolamento tra programmi e dati:
Caratteristica chiamata anche **indipendenza tra programmi e dati**. Consente di cambiare le strutture o le operazioni di memorizzazione dei dati senza dover modificare i programmi di accesso del DBMS.
- Astrazione dai dati:
Viene utilizzato un **modello dei dati** per nascondere all'utente i dettagli sulla memorizzazione e per presentare una *rappresentazione concettuale* del database.

Caratteristiche dell'approccio con database

- Supporto di viste multiple dei dati:
Ciascun utente può vedere il database da prospettive diverse (**viste**). Ciascuna vista descrive *solamente* i dati di interesse per l'utente.
- Condivisione dei dati e gestione delle transazioni con utenti multipli:
Permette ad un insieme di utenti di operare in modo **concorrente** sul database. Il controllo della concorrenza del DBMS garantisce che ciascuna **transazione** sia eseguita correttamente oppure annullata. L'OLTP (Online Transaction Processing) è un aspetto di grande rilievo per le applicazioni con database!

Utenti del database

- Gli Utenti possono essere suddivisi tra quelli che effettivamente utilizzano e controllano il contenuto del database (*Attori sulla scena*) e quelli che lavorano per la manutenzione del DMBS ma che non sono direttamente interessati al database in sé (*Lavoratori dietro le quinte*).

Attori sulla scena

- **Amministratori**
Responsabili di autorizzare gli accessi al database, di coordinare e monitorarne l'uso, di acquistare (scegliere) il software e l'hardware necessario, di controllarne l'utilizzo e monitorare l'efficienza delle operazioni.
- **Progettisti**
Responsabili di definire il contenuto, la struttura, i vincoli e le funzioni o le transazioni del database. Devono comunicare con gli utenti finali e capire le loro necessità.
- **Utenti Finali**
Coloro che utilizzano effettivamente il database. Esistono diverse tipologie.

Tipologie di Utenti Finali

- **Utenti casuali**
Accedono occasionalmente al database. Esperti.
- **Utenti parametrici**
La maggior parte degli utenti finali. Usano *funzioni* o *transazioni predefinite* (programmate in precedenza e testate, con parametri inseribili dall'utente) per accedere al database in maniera ripetuta ed effettuare aggiornamenti, inserimenti, interrogazioni.
- **Utenti sofisticati**
Scienziati, ingegneri, analisti che usano direttamente le potenzialità del DBMS per soddisfare le proprie esigenze.
- **Utenti indipendenti**

Lavoratori dietro le quinte

- **Progettisti ed implementatori di DBMS**
Coloro che lavorano sui programmi che costituiscono il DBMS e ne forniscono funzionalità, interfacce, accesso ai dati, ...
- **Sviluppatori di strumenti**
Coloro che forniscono gli strumenti necessari a facilitare la progettazione, il controllo, la comunicazione e l'ottimizzazione delle prestazioni dei database.
- **Operatori per la manutenzione**
Coloro che si occupano di mantenere efficiente hardware e software per il DBMS.

Vantaggi dei database

- Controllo della ridondanza nella memorizzazione dei dati e nel lavoro di sviluppo e mantenimento.
- Condivisione dei dati tra più utenti.
- Controllo degli accessi ai dati.
- Strutture di memorizzazione tali da garantire un'efficiente interrogazione dei dati.

Vantaggi dei database

- Servizi di Backup e Recovery
- Interfacce multiple a diverse classi di utenti
- Rappresentazione di relazioni complesse tra i dati
- Garanzia sui vincoli di integrità

Ulteriori implicazioni

- Rafforzamento degli standard
- Tempi di sviluppo abbreviati
- Flessibilità
- Disponibilità di informazioni aggiornate
- Economia di scala

Storia

- Prime applicazioni di database:
Modelli **gerarchico** e **reticolare**, introdotti a metà degli anni 60 ed utilizzati per tutti gli anni settanta (ancora oggi)
- Sistemi basati sul modello relazionale:
Il modello **relazionale** è stato introdotto nel 1970. Il più usato al giorno d'oggi. Ricerche ed esperimenti iniziati in IBM ed università. Si è affermato negli anni 80.

Storia

- Applicazioni orientate agli oggetti:
Gli **OODBMS** sono stati introdotti alla fine degli anni 80, inizio anni 90 per affrontare le necessità di data processing complesso in sistemi CAD (ed altro) mediante l'uso di linguaggi e metodologie ad oggetti. L'uso non si è ancora affermato.
- Scambio di dati su Web:
Applicazioni specifiche per visualizzare su **Web** (in pagine html) dati estratti da un database. Numerose tecnologie. Uso per il commercio elettronico ma anche per applicazioni *semplici* (weblog, ...). Evoluzione rapida; **XML**.

Estensione delle capacità di un database

Aggiunta di nuove funzionalità ai database nei seguenti settori:

- Applicazioni scientifiche
- Memorizzazione e gestione delle immagini
- Gestione di dati audio e video
- Gestione di dati spaziali
- Gestione di dati storici e temporali
- Data Mining

Tutto ciò si basa su (e stimola) nuove ricerche e sforzi di sviluppo per incorporare nuovi tipi di dati, strutture complesse, nuove operazioni e schemi di indicizzazione nei sistemi di database.

Quando non usare un database

- Quando i costi lo impediscono (o sconsigliano):
 - costi iniziali per l'acquisto del DBMS, dell'hardware, della formazione
 - costi di gestione per assicurare sicurezza, concorrenza, ripristino ed integrità
- Quando non sono necessari per l'applicazione:
 - struttura dei dati semplice e non variabile nel tempo
 - esigenze di velocità che non possono essere garantite dalla struttura generale di un DBMS
 - singolo utente

Quando non usare un database

- Quando non sono sufficienti:
 - complessità dei dati talmente elevata da non poter essere gestita dal modello utilizzato dal DBMS
 - operazioni richieste dagli utenti non disponibili nel DBMS