



Politecnico  
di Torino



# Introduzione alle basi di dati

---

# Introduzione alle basi di dati

---

- Gestione delle informazioni
- Base di dati
- Modello dei dati
- Progettazione di una base di dati
- Indipendenza dei dati
- Accesso ai dati
- Vantaggi e svantaggi dei DBMS

# Gestione delle informazioni

---

Introduzione alle basi di dati

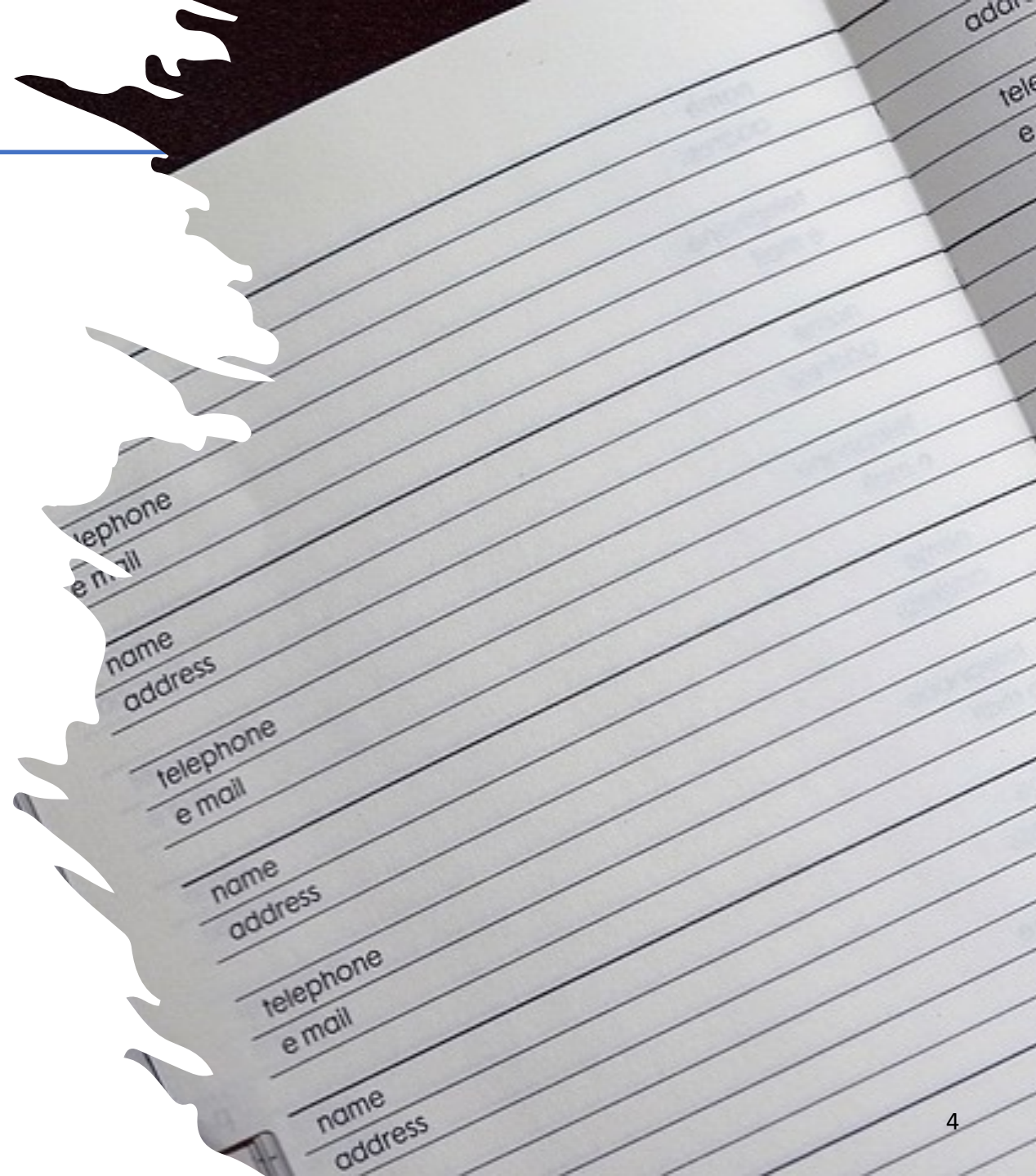
# Gestione delle informazioni

- Le informazioni sono registrate e scambiate in forme diverse
- Nel tempo, sono state introdotte forme di organizzazione e codifica delle informazioni



# Sistemi informatici

- Nei sistemi informatici, le informazioni sono rappresentate per mezzo di dati
  - i dati sono simboli grezzi che devono essere interpretati e correlati per fornire informazioni
- Esempio
  - dati: “Mario Rossi” e 424242
  - informazione: risultato della ricerca di un numero di telefono sulla propria agenda telefonica



# Caratteristiche dei dati

---



**I dati sono molto più stabili nel tempo dei processi che li gestiscono**

Esempio:

- i dati relativi alle applicazioni bancarie hanno una struttura invariata da decenni
- le procedure che li gestiscono variano di anno in anno



**I dati sono una risorsa importante dell'organizzazione che li gestisce**

# Base di dati

---

Introduzione alle basi di dati

# Base di dati

---

- Definizione generale
  - una base di dati è una collezione di dati che rappresenta le informazioni di interesse per un sistema informativo
- Definizione “tecnica”
  - una base di dati è una collezione di dati gestita da un DBMS (Database Management System)



# Tipi di basi di dati - DBMS

## Relazionali



## Ad oggetti

## Data warehouse



## In-memory



## NoSQL



## Documentali



## Serie temporali



## Real-time



## Grafo



*... e altri...*

# DataBase Management System - DBMS

---

- Un *sistema per la gestione di basi di dati* o DBMS (Data Base Management System) è un sistema software in grado di gestire collezioni di dati che siano
  - grandi
  - condivise
  - persistentiassicurando la loro affidabilità e privacy

# Caratteristiche dei DBMS

---

- Dimensioni molto maggiori della memoria centrale disponibile
  - gestione dei dati in memoria secondaria
- Condivisione dei dati tra applicazioni ed utenti: una base di dati è una risorsa *integrata*, condivisa da più settori aziendali
  - riduzione della ridondanza dei dati
  - riduzione delle inconsistenze tra i dati
  - meccanismo di controllo dell'accesso concorrente

# Caratteristiche dei DBMS



## Persistenza dei dati

tempo di vita non limitato a quello dell'esecuzione dei programmi che li utilizzano



## Affidabilità dei dati in caso di malfunzionamenti hardware e software

funzionalità di salvataggio (backup) e ripristino (recovery)



## Privatezza dei dati

meccanismi di autorizzazione per abilitare gli utenti



## Efficienza

capacità di svolgere le operazioni utilizzando un insieme di risorse (tempo e spazio) accettabile per gli utenti

sistema informatico adeguatamente dimensionato



## Efficacia

capacità di rendere produttive le attività degli utenti

# DBMS o file system?

---

- Approccio “semplificato” alla gestione dei dati: dati memorizzati in modo persistente nella memoria di massa all’interno di file
  - è possibile memorizzare e ricercare dati
  - semplici meccanismi di accesso (lettura sequenziale)
  - semplici meccanismi di condivisione (condivisione in lettura e blocco del file in scrittura)
- I DBMS estendono le funzionalità dei file system, fornendo più servizi in modo integrato

# Modello dei dati

---

Introduzione alle basi di dati

# Modello dei dati

---

- Un modello dei dati è un insieme di concetti utilizzati per organizzare i dati di interesse e descriverne la struttura in modo da essere comprensibile ad un elaboratore
  - tipi di dato elementari (intero, carattere, ...)
  - meccanismo di strutturazione per definire strutture più complesse (costruttore di record, array, ...)

# Tipi di modelli di dati

---

- Modello di dati relazionale
  - Modello di dati più diffuso
  - Dati organizzati in insiemi di record omogenei (a struttura fissa) e rappresentati come tabelle.
- Prima del modello relazionale, sono stati utilizzati altri modelli più vicini alle strutture fisiche di memorizzazione
  - modello gerarchico, modello di reticolare
- Dopo il modello relazionale
  - Modello a oggetti, XML, database NOSQL, ...



# Modello relazionale

Docenti

Nome	Dipartimento	Telefono
Verdi	Informatica	123456
Bianchi	Elettronica	636363
Neri	Informatica	414243

Corsi

Codice	Nome	Docente
M2170	Fondamenti di informatica	Verdi
M4880	Sistemi di elaborazione	Bianchi
F0410	Basi di dati	Neri

# Schema e istanza

- In una base di dati sono definiti
  - lo *schema*, che descrive la struttura dei dati
    - praticamente invariante nel tempo
    - è rappresentato dall'intestazione di ogni tabella (nome tabella e nomi colonne)

Docenti

Nome	Dipartimento	Telefono
Verdi	Informatica	123456
Bianchi	Elettronica	636363
Neri	Informatica	414243

SCHEMA

Corsi

Codice	Nome	Docente
M2170	Fondamenti di informatica	Verdi
M4880	Sistemi di elaborazione	Bianchi
F0410	Basi di dati	Neri

SCHEMA

# Schema e istanza

- In una base di dati sono definiti
  - l'*istanza*, costituita dal contenuto di ogni tabella, cioè dai valori effettivi dei dati
    - variabile nel tempo, anche molto rapidamente
    - è rappresentata dalle righe delle tabelle

Docenti

Nome	Dipartimento	Telefono
Verdi	Informatica	123456
Bianchi	Elettronica	636363
Neri	Informatica	414243



Corsi

Codice	Nome	Docente
M2170	Fondamenti di informatica	Verdi
M4880	Sistemi di elaborazione	Bianchi
F0410	Basi di dati	Neri



# Esempio di altri modelli di dati: Database NOSQL

- Un database è un insieme di raccolte
- Ogni collezione contiene un insieme di documenti
- Ogni documento è descritto da una lista di campi chiave-valore e ogni campo può contenere qualsiasi tipo di dato
- I documenti di una stessa collezione possono essere eterogenei
- Poiché la rappresentazione dei dati è priva di schema, non è necessario definire a priori lo schema dei documenti e gli oggetti di una stessa collezione possono essere caratterizzati da campi diversi.

Database relazionale	NOSQL database
Tabella	Collezione
Riga	Documento
Colonna	Campo

# Esempio di dati di un documento

- Le relazioni tra i documenti sono inefficienti e portano a una de-normalizzazione
  - Riferimento all'oggetto (ID), senza join nativo

```
{
  _id: <ObjectId>,
  username: "123xyz",
  contact: {
    phone: "123-456-7890",
    email: "xyz@example.com"
  },
  access: {
    level: 5,
    group: "dev"
  }
}
```

Embedded sub-document

Embedded sub-document

Source: <https://www.mongodb.com/docs/manual/core/data-model-design/>

# Tipi di modello

## *Modello concettuale*

- Permette di rappresentare i dati in modo indipendente dal modello logico
  - descrive concetti del mondo reale
  - utilizzato nella fase di progettazione
- Esempio: modello Entità-Relazione

## *Modello logico*

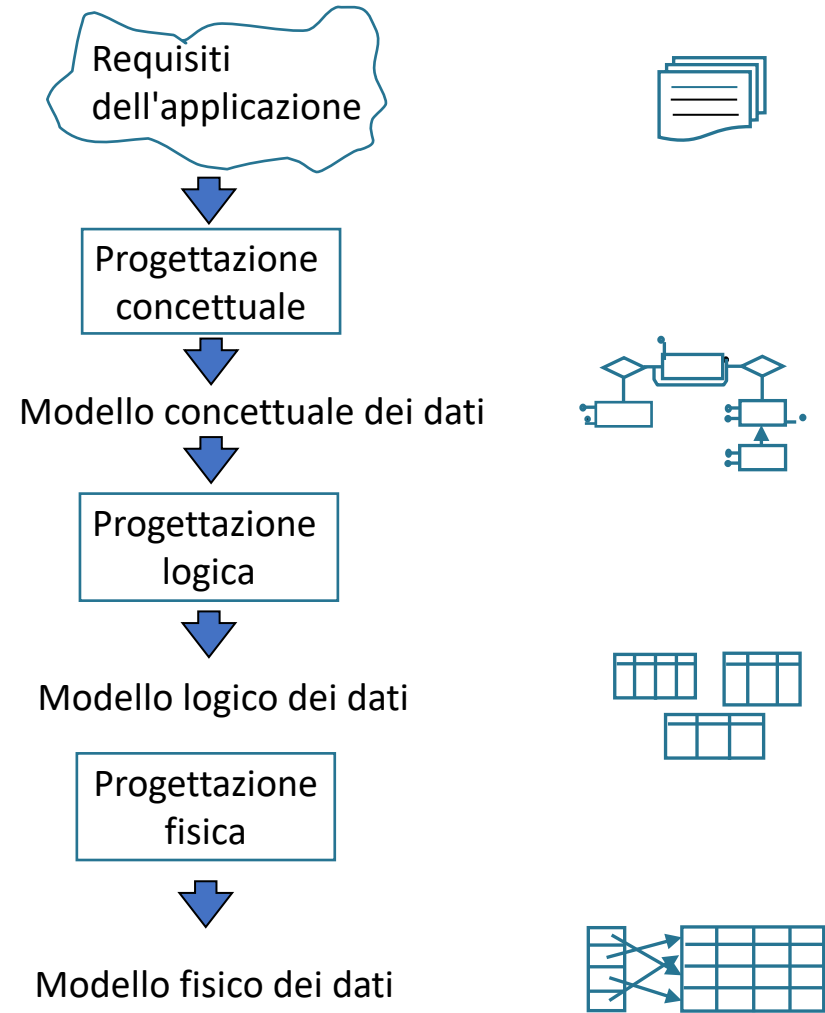
- Descrive la struttura dei dati nel DBMS
  - utilizzato dai programmi che accedono ai dati
  - indipendente dalle strutture fisiche
- Esempio: modello relazionale

# Progettazione di una base di dati

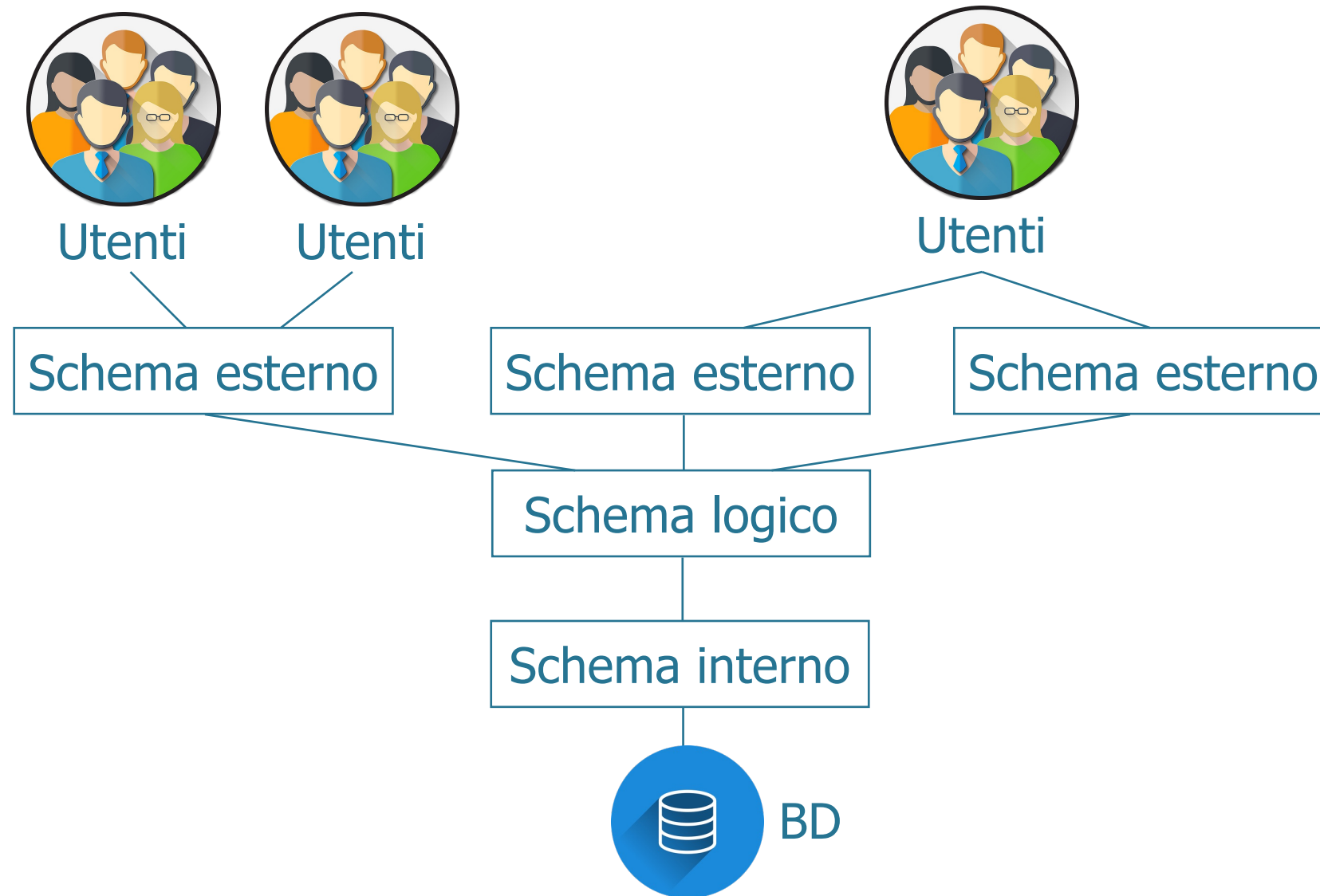
Rappresentare i requisiti informali di un'applicazione in termini di schema concettuale che fa riferimento a un modello concettuale di dati.

Traduzione dello schema concettuale definito nella fase precedente nello schema logico del database che fa riferimento a un modello logico di dati.

Lo schema logico è completato dai dettagli dell'implementazione fisica (organizzazione dei file e indici) su un determinato DBMS. Il prodotto è chiamato schema fisico e si riferisce a un modello fisico di dati.



# Livelli di astrazione in un DBMS





# Architettura standard ANSI/SPARC a tre livelli per DBMS

---

- Schema logico
  - descrizione della base di dati mediante il modello logico del DBMS
- Schema interno
  - rappresentazione dello schema logico mediante strutture fisiche di memorizzazione
- Schema esterno
  - descrizione di parti della base di dati, denominate “viste”, che riflette il punto di vista di particolari utenti
  - è definita sul modello logico

# Indipendenza dei dati

---

Introduzione alle basi di dati

# Indipendenza dei dati

---

- L'indipendenza dei dati garantisce che gli utenti e i programmi applicativi che utilizzano una base di dati possano ignorare i dettagli realizzativi utilizzati nella costruzione della base di dati
- È una conseguenza della suddivisione in livelli di astrazione
- Livelli di indipendenza dei dati:
  - Indipendenza fisica
  - Indipendenza logica

# Indipendenza dei dati

---

- Indipendenza fisica
  - consente di interagire con il DBMS in modo indipendente dalla struttura fisica dei dati
  - l'accesso a una relazione (livello logico o esterno) avviene sempre nello stesso modo, indipendentemente dalla modalità di memorizzazione
  - è possibile modificare il modo in cui i dati sono memorizzati fisicamente senza influire sui programmi che utilizzano i dati

# Indipendenza dei dati

---

- Indipendenza logica
  - consente di interagire con il livello esterno in modo indipendente dal livello logico
  - è possibile modificare il livello logico mantenendo inalterate le strutture esterne (purché le corrispondenze siano inalterate)
  - è possibile aggiungere nuove viste o modificare viste esistenti senza modificare lo schema logico

# Accesso ai dati

---

Introduzione alle basi di dati

# Linguaggi di accesso ai dati

---

- Interfacce amichevoli che permettono di specificare interrogazioni senza utilizzare un linguaggio testuale
- Linguaggi testuali interattivi (es., SQL, DSL)
- Comandi simili a quelli interattivi introdotti in linguaggi di programmazione tradizionale (C, C++, COBOL, Java, Python, ...), detti linguaggi ospite
- ORM (Object-Relational Mapping): mappatura degli oggetti nei linguaggi di programmazione con le tabelle/documenti dei database
- Comandi simili a quelli interattivi introdotti in linguaggi di sviluppo ad hoc, spesso con funzionalità specifiche (generazione di grafici, stampe complesse, videate)

# Linguaggi di accesso ai dati

---

- Sono divisi in due categorie
  - *linguaggi di definizione dei dati* (Data Definition Language o DDL) utilizzati per definire gli schemi logici, esterni e fisici e le autorizzazioni per l'accesso
  - *linguaggi di manipolazione dei dati* (Data Manipulation Language o DML) utilizzati per l'interrogazione e l'aggiornamento delle istanze della base di dati



# Utenti

---

- **Database administrator:** responsabile del controllo (centralizzato) e della gestione della base di dati
  - garantisce prestazioni sufficienti
  - assicura l'affidabilità del sistema
  - gestisce le autorizzazioni di accesso ai dati
- **Progettisti e programmatori:** definiscono e realizzano
  - la struttura della base di dati
  - i programmi che accedono alla base di dati
- **Utenti:** utilizzano la base di dati per le proprie attività
  - utenti finali: utilizzano transazioni, cioè programmi che realizzano attività predefinite
  - utenti casuali: formulano interrogazioni (o aggiornamenti) non predefinite mediante i linguaggi interattivi di accesso alla base di dati

# Transazioni

---

- Programmi che realizzano attività frequenti e predefinite
- Esempi
  - prenotazione di volo aereo
  - bonifico bancario
  - acquisto prodotti su e-commerce
- Generalmente realizzate con l'introduzione di SQL all'interno di un linguaggio ospite

# Vantaggi e svantaggi dei DBMS

---

Introduzione alle basi di dati

# Vantaggi dei DBMS

- Dati come risorsa comune di tutta l'organizzazione
  - riduzione di ridondanze e inconsistenze
- Modello dei dati unificato e preciso della realtà di interesse per l'organizzazione
- Possibile il controllo centralizzato dei dati
  - standardizzazione, economie di scala
- Indipendenza dei dati



# Svantaggi dei DBMS

- Sono prodotti costosi, complessi, che richiedono
  - investimenti diretti
    - acquisto del prodotto
  - investimenti indiretti
    - acquisizione delle risorse hardware e software necessarie
    - conversione delle applicazioni
    - formazione del personale
- Forniscono un insieme di servizi in forma integrata
  - non è possibile scorporare servizi inutilizzati che possono causare riduzioni di prestazioni

