

# FUNZIONALITÀ DEI DBMS

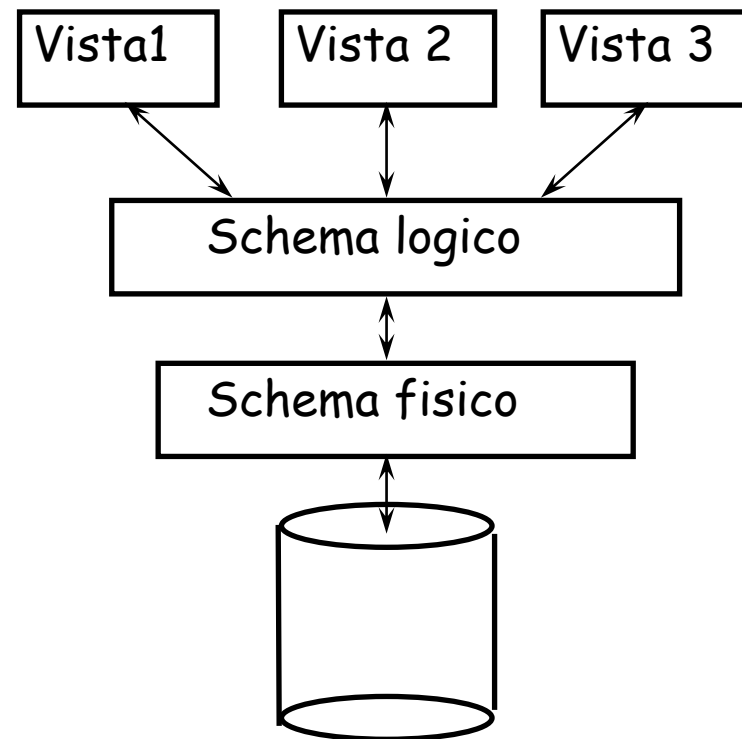
---

- Linguaggio per la definizione della base di dati;
- Linguaggi per l'uso dei dati;
- Meccanismi per il controllo dei dati;
- Strumenti per il responsabile della base di dati;
- Strumenti per lo sviluppo delle applicazioni

# LINGUAGGIO PER LA DEFINIZIONE DELLA BASE DI DATI (DDL)

---

- È utile distinguere tre diversi livelli di descrizione dei dati (**schemi**):
  - il livello di vista logica,
  - il livello logico,
  - il livello fisico.



## LIVELLO LOGICO

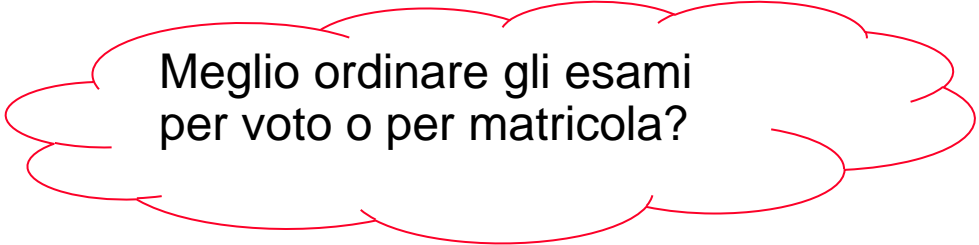
---

- **Livello logico**: descrive la **struttura degli insiemi di dati e delle relazioni fra loro**, secondo un certo **modello dei dati**, senza nessun riferimento alla **loro organizzazione fisica** nella memoria permanente (Schema logico).
- Esempio di schema logico:  
  
    Studenti(Matricola char(8), Nome char(20), login char(8),  
            AnnoNascita int, Reddito real)  
  
    Corsi(IdeC char(8), Titolo char(20), Credito int)  
  
    Esami(Matricola char(8), IdeC char(8), Voto int)

# LIVELLO FISICO

---

- **Livello fisico:** descrive come vanno organizzati **fisicamente i dati nelle memorie permanenti** e **quali strutture dati ausiliarie** prevedere per facilitarne l'uso (Schema fisico o interno).



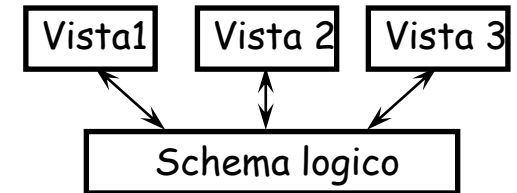
Meglio ordinare gli esami  
per voto o per matricola?

- Esempio di schema fisico
  - Relazioni Studenti e Esami organizzate in modo seriale, Corsi organizzata sequenziale con indice
  - Indice su Matricola, (Matricola, IdeC)

## LIVELLO VISTA LOGICA

---

- **Vista logica**: descrive come deve apparire la struttura della base di dati ad una certa applicazione (**Schema esterno** o **vista**).

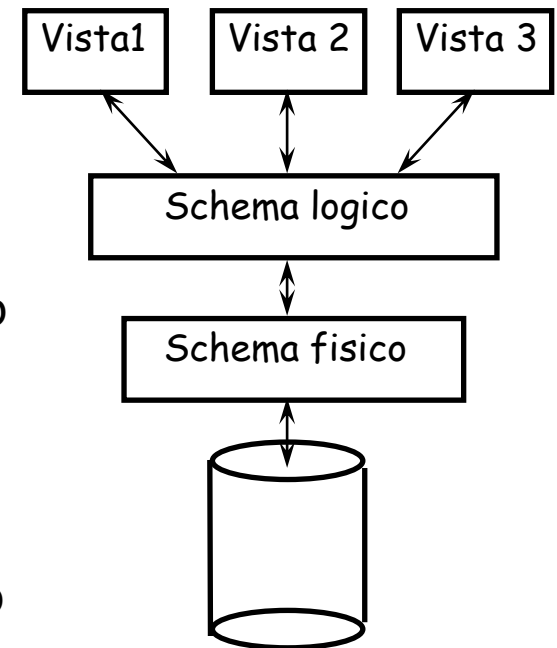


- Esempio di schema esterno
  - InfCorsi (IdeC char(8), Titolo char(20), NumEsami int)
- Esempio:
  - Nell'organizzazione di una banca, lo **schema logico** conterrà tutte le tabelle e i dati relativi ai conti correnti, ma anche al personale. Lo schema logico conserva tutte le informazioni della banca.
  - Nello **schema esterno**, ogni correntista potrà accedere solo ad alcune informazioni che sono per lui di interesse: quelle relative al **proprio** conto corrente.

## LINGUAGGIO PER LA DEFINIZIONE DELLA BD (cont.)

---

- L'approccio con tre livelli di descrizione dei dati è stato proposto come un modo per garantire le **proprietà di indipendenza logica e fisica dei dati**, che sono fra gli obiettivi più importanti dei DBMS.
- **Indipendenza logica:** i programmi applicativi non devono essere modificati in seguito a *modifiche dello schema logico*.
- **Indipendenza fisica:** i programmi applicativi non devono essere modificati in seguito a *modifiche dell'organizzazione fisica dei dati*.



# FUNZIONALITÀ DEI DBMS: LINGUAGGI PER L'USO DEI DATI

---

- Un **DBMS** deve prevedere **più modalità d'uso** per soddisfare le esigenze delle diverse categorie di utenti che possono accedere alla base di dati (dati e catalogo):
  - Un'interfaccia grafica per accedere ai dati
  - Un linguaggio di interrogazione per gli utenti non programmatori
  - Un linguaggio di programmazione per gli utenti che sviluppano applicazioni:
    - integrazione DDL e DML nel linguaggio ospite: procedure predefinite, estensione del compilatore, precompilazione
    - comunicazione tra linguaggio e DBMS
  - Un linguaggio per lo sviluppo di interfacce per le applicazioni

## Una distinzione terminologica (separazione fra dati e programmi)

---

Nei linguaggi di interrogazione di basi di dati  
distinguiamo tra

### Data Manipulation Language (DML)

per l'interrogazione e l'aggiornamento di  
(istanze di) basi di dati

Select insegnamento

From Orario

Where docente="Mario Rossi"

### Data Definition Language (DDL)

per la definizione di schemi (logici, esterni,  
fisici) e altre operazioni generali

```
CREATE TABLE orario (  
    insegnamento CHAR(20) ,  
    docente        CHAR(20) ,  
    aula           CHAR(4)  ,  
    ora            CHAR(5)  
)
```



## FUNZIONALITÀ DEI DBMS (cont.)

---

- **Strumenti per l'amministratore** della base di dati
  - Un linguaggio per la definizione e la modifica degli schemi logico, interno ed esterno.
  - Strumenti per il controllo e messa a punto del funzionamento del sistema.
  - Strumenti per stabilire i diritti di accesso ai dati e per ripristinare la base di dati in caso di malfunzionamenti.
- **Strumenti per lo sviluppo** delle applicazioni
  - Produzione di rapporti, grafici, fogli elettronici
  - Sviluppo di menu, forme, componenti grafici

## Linguaggi per le basi di dati

---

- Disponibilità di vari linguaggi e interfacce diverse
  - linguaggi testuali interattivi (es. **SQL**)
  - comandi (come quelli del linguaggio interattivo) immersi in un linguaggio **ospite** (es. C, Cobol, etc.)
  - comandi (come quelli del linguaggio interattivo) immersi in un linguaggio **ad hoc** (es. PL/SQL), con anche altre funzionalità (p.es. per grafici o stampe strutturate), anche con l'ausilio di strumenti di sviluppo (p. es. per la gestione di maschere)
  - con interfacce amichevoli (senza linguaggio testuale)

## SQL immerso in linguaggio ad hoc (Oracle PL/SQL)

---

```
declare Stip number;
begin
    select Stipendio into Stip from Impiegato where Matricola = '575488'
    for update of Stipendio;
    if Stip > 30 then
        update Impiegato set Stipendio = Stipendio * 1.1 where Matricola = '575488';
    else
        update Impiegato set Stipendio = Stipendio * 1.15 where Matricola = '575488';
    end if;
    commit;
exception
    when no_data_found then
        insert into Errori values('Non esiste la matricola specificata',sysdate);
end;
```

# SQL, un linguaggio interattivo

DS3 dove si trova?

## Corsi

Corso	Docente	Aula
Basi di dati	Rossi	DS3
Sistemi	Neri	N3
Reti	Bruni	N3
Controlli	Bruni	G

## Aule

NomeAula	Edificio	Piano
DS1	OMI	Terra
N3	OMI	Terra
G	Pincherle	Primo

```
SELECT Corso, Aula, Piano
FROM Aule, Corsi
WHERE NomeAula = Aula
AND Piano="Terra"
```

Visualizzare per ogni corso  
che si svolge al piano terra,  
l'aula e il piano

Corso	Aula	Piano
Sistemi	N3	Terra
Reti	N3	Terra

# FUNZIONALITÀ DEI DBMS: MECCANISMI PER IL CONTROLLO DEI DATI

---

- Una caratteristica molto importante dei DBMS è il tipo di meccanismi offerti per garantire le seguenti proprietà di una base di dati:
  - **Integrità**: mantenimento delle proprietà specificate in modo dichiarativo nello schema (vincoli d'integrità)
  - **Sicurezza**: *protezione dei dati da usi non autorizzati*
  - **Affidabilità**: *protezione dei dati da malfunzionamenti hardware o software (fallimenti di transazione, di sistema e disastri) e da interferenze indesiderate dovute all'accesso concorrente ai dati da parte di più utenti.*

# CONTROLLO DEI DATI: LE TRANSAZIONI

---

- *Definizione* Una **transazione** è una sequenza di azioni di lettura e scrittura in memoria permanente e di elaborazioni di dati in memoria temporanea, con le seguenti proprietà:
  - **Atomicità**: Le transazioni che terminano prematuramente (aborted transactions) sono trattate dal sistema come se non fossero mai iniziate; pertanto eventuali loro effetti sulla base di dati sono annullati.
  - **Persistenza**: Le modifiche sulla base di dati di una transazione terminata normalmente sono permanenti, cioè non sono alterabili da eventuali malfunzionamenti.
  - **Serializzabilità**: Nel caso di esecuzioni concorrenti di più transazioni, l'effetto complessivo è quello di una esecuzione seriale.

# SISTEMI PER BASI DI DATI (DATA BASE MANAGEMENT SYSTEMS - DBMS)

---

Un **DataBase Management System (DBMS)** è un sistema (**prodotto software**) in grado di gestire **collezioni di dati** che siano (anche):

- **Grandi**
- **Persistenti** (con un periodo di vita indipendente dalle singole esecuzioni dei programmi che le utilizzano)
- **Condivise** (utilizzate da applicazioni diverse)

garantendo **affidabilità** (resistenza a malfunzionamenti hardware e software-recovery) e **privatezza** (con una disciplina e un controllo degli accessi). Come ogni prodotto informatico, un DBMS deve essere **efficiente** (utilizzando al meglio le risorse di spazio e tempo del sistema) ed **efficace** (rendendo produttive le attività dei suoi utilizzatori).

## RIEPILOGO DEI VANTAGGI DEI DBMS

---

- Indipendenza dei dati
- Recupero efficiente dei dati
- Integrità e sicurezza dei dati
- Accessi interattivi, concorrenti e protetti dai malfunzionamenti
- Amministrazione dei dati
- Riduzione dei tempi di sviluppo delle applicazioni
- La riduzione dei costi della tecnologia e i possibili tipi di DBMS disponibili sul mercato facilitano la loro diffusione.



## SVANTAGGI DEI DBMS

---

- Prima di caricare i dati è necessario definire uno schema
- Possono gestire solo dati strutturati e omogenei
- Possono essere costosi e complessi da installare e mantenere in esercizi
- Sono ottimizzati per le applicazioni OLTP, non per quelle OLAP