

# Basi di Dati

prof. Letizia Tanca  
Politecnico di Milano

Introduzione al corso

# Basi di dati

- Titolare del corso: Prof. Letizia Tanca
  - ricevimento: v. sito PoliSelf
    - attualmente, su appuntamento, il mercoledì dalle 14:30 alle 16:30
  - email: [letizia.tanca@polimi.it](mailto:letizia.tanca@polimi.it)
  - tel: 02-2399-3531, fax: 02-2399-3411
  - Sito personale: <http://tanca.faculty.polimi.it/>
    - Vi si trovano:
      - le pagine del corso, ma anche...
      - informazioni sugli interessi di ricerca all'interno dei quali inserire progetti
- Esercitazioni: dott. Emanuele Rabosio:  
[emanuele.rabosio@polimi.it](mailto:emanuele.rabosio@polimi.it)

# Distribuzione indicativa dell'attività didattica

- Ore di lezione: 30
- Ore di esercitazione: 20
- 5 ore alla settimana non sempre tutte utilizzate
- Calendario sul sitoWeb

# Modalità di verifica

- L'esame consiste in una prova scritta.
- Sarà possibile sostenere 2 prove in itinere
- Per superare l'esame tramite le prove in itinere e' necessario che:
  - il risultato di ciascuna prova sia al di sopra di una soglia minima e
  - la media dei due voti sia superiore a 18/30
- Consegnando la prima prova in itinere, si perde il diritto a sostenere uno dei due appelli della sessione invernale
- Il contenuto delle due prove in itinere verra' precisato durante lo svolgimento del corso
- *Le modalità e gli appelli di esame saranno comunque in accordo con quanto previsto dalla Facoltà di Ingegneria Industriale e dell'Informazione, e riportato sul relativo sito Web.*

# Programma del corso

- 1. INTRODUZIONE AL CORSO: Il sistema informativo nell'azienda; Caratteristiche dei DBMS; Utenti, strumenti e moduli di un DBMS**
- 2. TEORIA RELAZIONALE: Il modello relazionale dei dati; Algebra relazionale; Interrogazioni in algebra relazionale e loro ottimizzazione; Calcolo relazionale: definizione, equivalenza fra algebra relazionale e calcolo relazionale; Datalog: interrogazioni ricorsive, vincoli di integrità, equivalenza fra tutti i linguaggi formali**
- 3. LINGUAGGI PER BASI DI DATI: Introduzione a SQL; SQL come DDL: integrità referenziale, definizione di schemi e loro modifica, cataloghi; Interrogazioni SQL: semplici, con ordinamenti e raggruppamenti, complesse; Comandi di modifica e viste in SQL; Aspetti evoluti del DDL: indici e vincoli di integrità generici, controllo dell'accesso, viste e controllo dell'accesso, transazioni (cenni); SQL per la programmazione: funzioni SQL, SQL Embedded, ODBC/ADO e JDBC, procedure; Integrazione Web e Basi di Dati (Cenni)**
- 4. PROGETTAZIONE DI BASI DI DATI: Fasi e astrazioni nella progettazione dei dati; Il modello Entità-Relazioni: definizione, associazioni, identificatori e gerarchie, proprietà delle gerarchie; Il ciclo di progettazione: fasi di progettazione, strategie, qualità di un progetto concettuale; Progettazione logica: fasi del progetto logico, cenni di normalizzazione; Progettazione fisica (cenni)**

# Libri di testo

- **Libro di testo: Basi di dati - Modelli e linguaggi di interrogazione 4/ed** di: Paolo Atzeni, Stefano Ceri, Piero Fraternali, Stefano Paraboschi e Riccardo Torlone  
**ISBN: 9788838668005**, giugno 2013  
In inglese, degli stessi autori:
- **Database Systems - Concepts, Languages and Architectures**  
Paolo Atzeni, Stefano Ceri, Stefano Paraboschi and Riccardo Torlone  
**ISBN 0077095006**
- **Eserciziari:**
  - Luciano Baresi, Chiara Francalanci, Fabio Schreiber, Letizia Tanca, “Progetto di dati e funzioni”, Esculapio, Bologna 2004.
  - D. Braga, M. Brambilla, A. Campi: Eserciziario di Basi di Dati, progetto leonardo, Esculapio, Bologna

# Altri materiali e risorse

- Lucidi delle lezioni nella pagina del corso
- **ATTENZIONE: I LUCIDI NON SONO ESAUSTIVI!**
- Soluzioni di tutti gli esercizi del libro sul sito della versione inglese del libro <http://www.mcgraw-hill.co.uk/atzeni>
- Esercizi risolti sul sito corsionline <http://corsionline.como.polimi.it>, sotto “Basi di dati 1”
- Altre informazioni e avvisi nella pagina del corso
- **Altra bibliografia di approfondimento**
  - **R. Elmasri, S.B. Navathe:** *Fundamentals of Database Systems*, Benjamin-Cummings, quarta edizione
  - **R. Ramakrishnan, J. Gehrke:** *Sistemi di Basi di dati*, Mc Graw –Hill, 2004
  - **C. Batini, S. Ceri, S.B. Navathe:** *Conceptual Database Design: an Entity-Relationship Approach*, Benjamin-Cummings, 1991

# L'informatica

E' la **scienza** che si occupa della  
**rappresentazione dell'informazione**  
e della sua **gestione**



# L'informatica e le basi di dati

- **Informazione:** e' parte di qualsiasi attivita' umana
- **Rappresentazione:** il problema di astrarre i concetti importanti da quelli trascurabili per poter **modellare** la realta' di interesse: **ci serve studiare metodi di rappresentazione appropriati all'elaborazione da parte di una macchina digitale**
- **Gestione:** uso e trasformazione dell'informazione in maniera funzionale agli obiettivi : **programmazione**

# Sistema Informativo

- Elemento del sistema organizzativo dell'azienda
- Insieme di risorse umane, strumenti di elaborazione, scambio, acquisizione di informazioni, procedure per il loro trattamento, regole organizzative, *che consentono il funzionamento* del sistema organizzativo
- Evoluzione dei sistemi informativi da settoriali a integrati

# Sistema informativo

- Il sistema informativo organizza e gestisce le informazioni necessarie per perseguire gli scopi di un'organizzazione

Se è gestito in modo automatizzato

⇒ **sistema informatico**

- Il concetto di sistema informativo è indipendente dalla sua gestione automatica ed esiste da secoli (servizi anagrafici, banche)

# Informazioni e dati

Nei sistemi informatici (e non solo in essi), le informazioni vengono rappresentate attraverso i **dati**.

- **informazione** : notizia, elemento che consente di avere conoscenza più o meno esatta di fatti, situazioni, modi di essere.
- **dato** : ciò che è immediatamente presente alla conoscenza, prima di ogni elaborazione; (in informatica) elementi di informazione costituiti da simboli che debbono essere elaborati.

# Base di dati

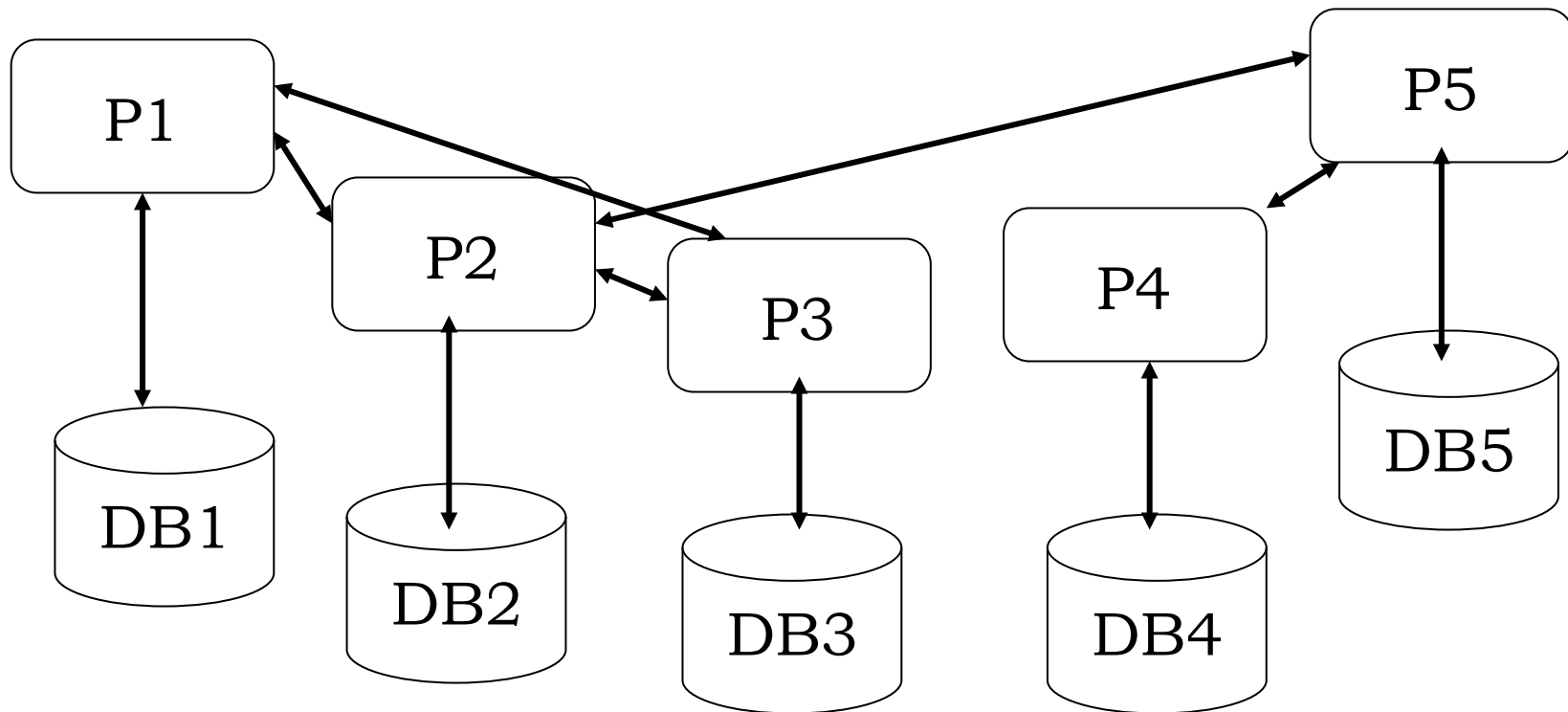
**Collezione di dati utilizzati per rappresentare le informazioni di interesse per una o più applicazioni di una organizzazione.**

Particolarmente efficace se **gestita da un sistema di gestione di basi di dati (SGBD)** o, in inglese, Data Base Management System **(DBMS)**

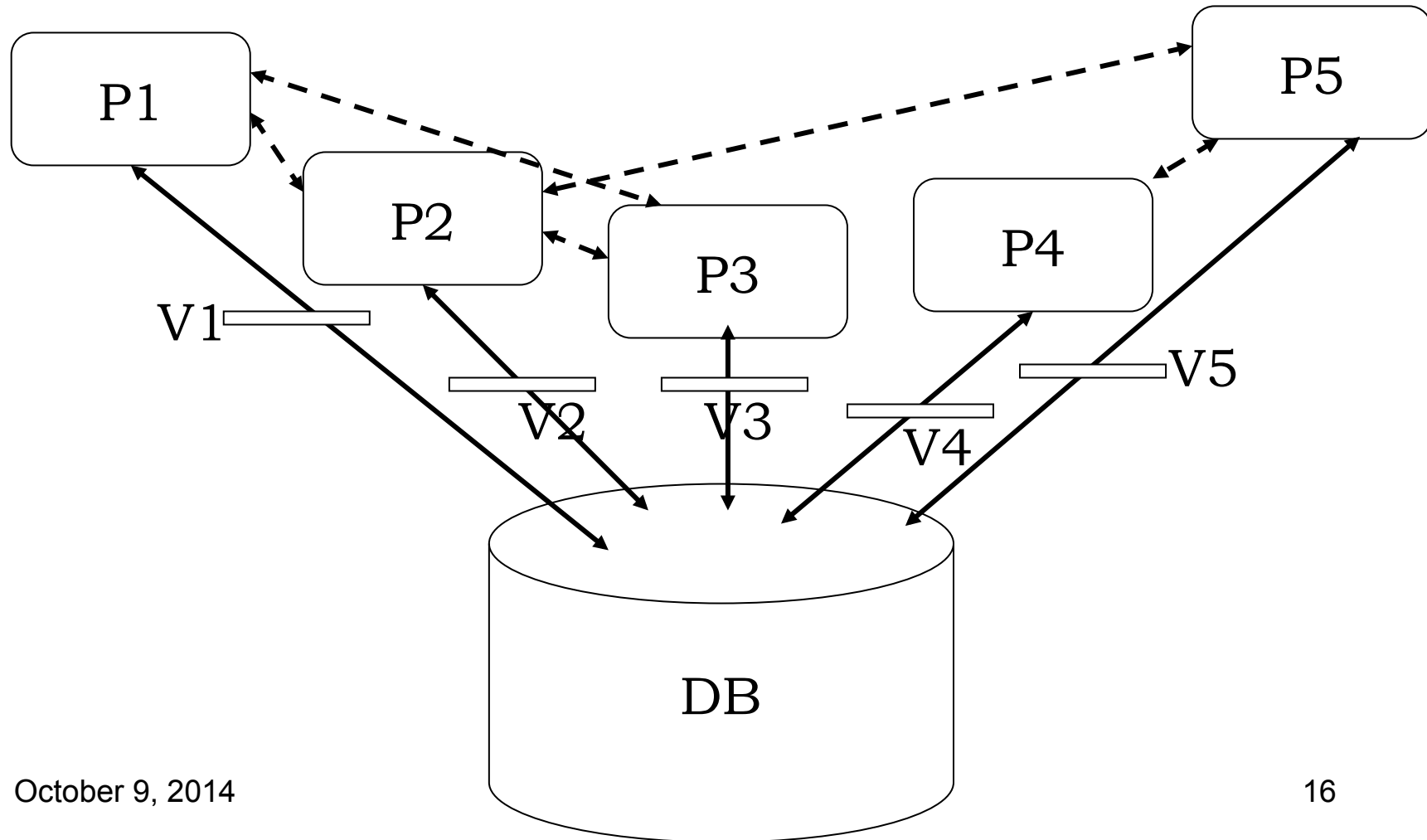
# Condivisione

- Ogni organizzazione (specie se grande) è divisa in settori o comunque svolge diverse attività. A ciascun settore o attività corrisponde un (sotto-) sistema informativo (privato o porzione di un sistema più grande).
- Possono esistere sovrapposizioni fra i dati di interesse dei vari settori.

# Da settoriali a integrati



# Da settoriali a integrati





# Una base di dati...

- è una risorsa **integrata**
- è **condivisa** tra i vari settori
- l'integrazione e la condivisione permettono di ridurre la **ridondanza**
- evitando le possibilità di **inconsistenza**
- occorrono meccanismi di definizione della **privatezza** dei dati e di limitazioni all'accesso (**autorizzazioni**).
- La condivisione richiede **controllo della concorrenza**.

# SGBD (DBMS) e file systems

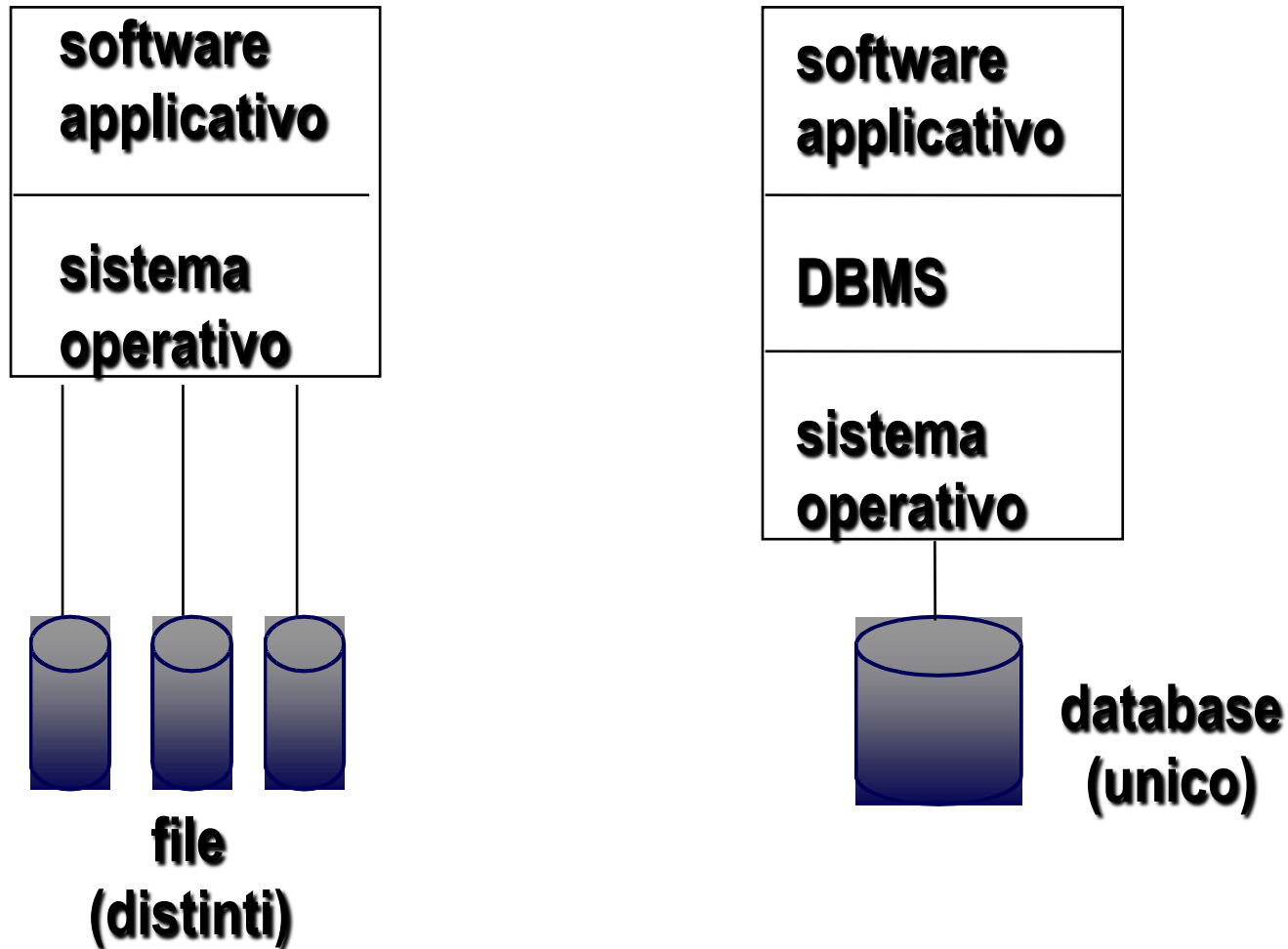
Un **SGBD** (Data Base Management System = DBMS):

- permette di accedere in modo efficiente ai dati con una **granularità più fine che il file system**
- permette di accedere in modo diretto ai dati **basandosi sulle loro proprietà**
- effettua il controllo della concorrenza **alla granularità del singolo record**
- realizza meccanismi sofisticati per il **controllo dell'affidabilità**
- realizza meccanismi di **controllo della privacy**
- realizza **l'atomicità delle transazioni**

# Efficienza

- Si misura (come in tutti i sistemi informatici) in termini di *tempo di esecuzione* (tempo di risposta) e *spazio di memoria* (principale e secondaria).
- I **DBMS**, a causa della varietà di funzioni, non sono necessariamente più efficienti dei file system.
- L'efficienza è il risultato della qualità del **DBMS** e delle applicazioni che lo utilizzano.

# File system e DBMS



# Descrizioni dei dati nei DBMS

- descrizioni e rappresentazioni dei dati **a livelli diversi**
- **indipendenza** dei dati dalla rappresentazione fisica
- concetto di **modello dei dati**.

# Modello dei dati

- insieme di costrutti utilizzati per organizzare i dati di interesse e descriverne la dinamica
- componente fondamentale: **meccanismi di strutturazione** (o costruttori di tipo)
- ad esempio, il **modello relazionale** prevede il costruttore **relazione (tabella)**, che permette di definire insiemi di record **omogenei**

# Tabelle: rappresentazione di relazioni

<b>CORSI</b>	<b>Corso</b>	<b>Docente</b>	<b>Aula</b>
	Basi di dati	Rossi	DS3
	Sistemi	Neri	N3
	Reti	Bruni	N3
	Controlli	Bruni	G

<b>AULE</b>	<b>Nome</b>	<b>Edificio</b>	<b>Piano</b>
	DS1	Ex-OMI	Terra
	N3	Ex-OMI	Terra
	G	Pincherle	Primo

# Schemi e istanze

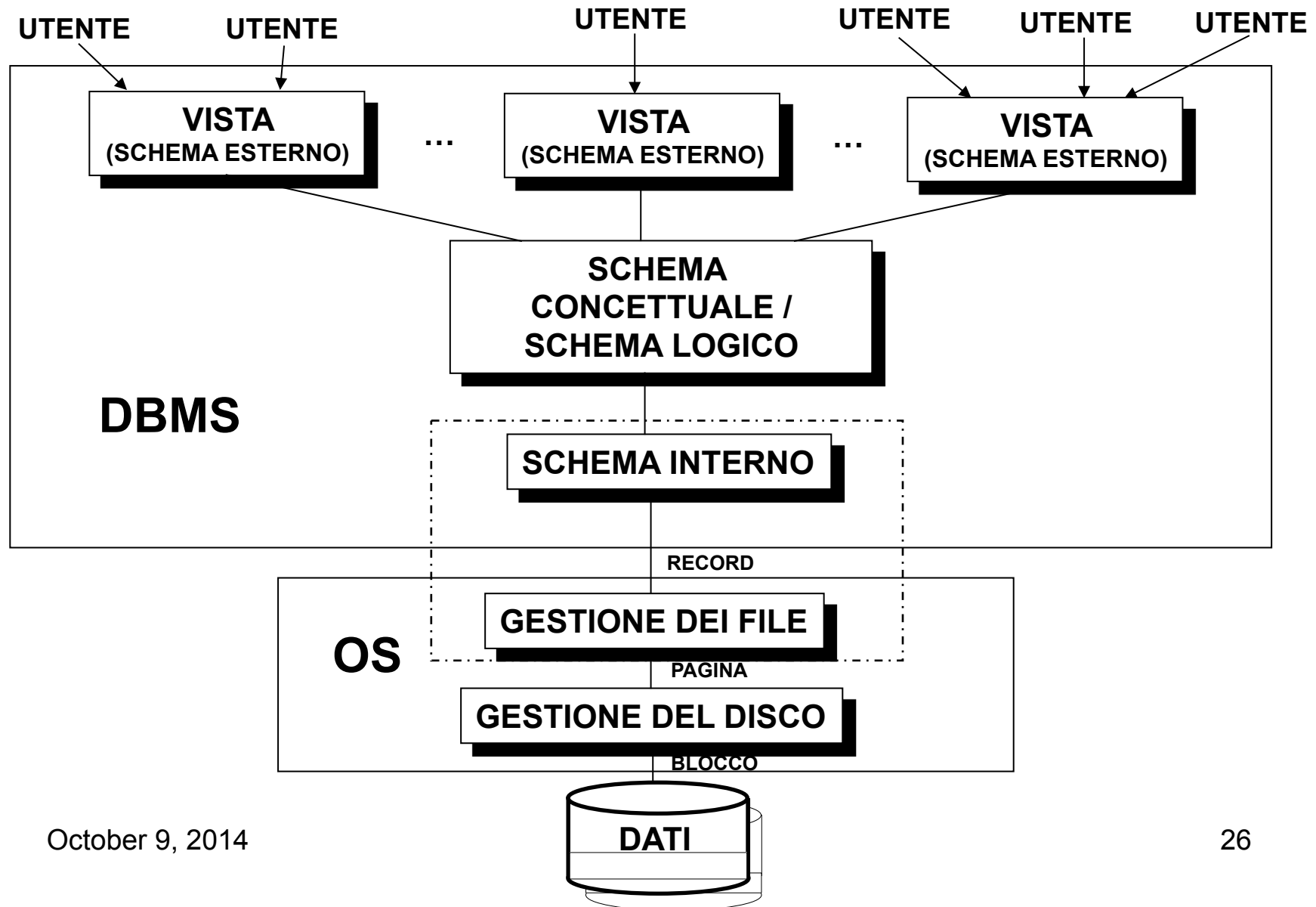
- **schema**, descrive la struttura (aspetto **intensionale**):  
    nell'esempio, le intestazioni delle tabelle
- **istanza**, costituita dai valori attuali (aspetto **estensionale**):  
    nell'esempio, il "contenuto" di ciascuna tabella



# Esempi

- **Modello dei dati:** collezione di concetti che possono essere usati per rappresentare la realta'. **Es:** alberi, grafi, tabelle
- **Schema dei dati:** rappresentazione di una specifica parte della realta', che usa un modello dei dati. **Es.** albero genealogico, sistema delle strade e delle citta', tabella degli studenti e dei voti nei singoli esami
- **Istanza:** collezione di valori dei dati che rispetta la struttura dello schema. **Es:** albero genealogico della mia famiglia *oggi*, rete viaria della Germania *il giorno 22 febbraio 2000*, dati sugli studenti e gli esami *in questo preciso istante*

# ARCHITETTURA STANDARD DI UN DBMS (ANSI/SPARC)



# Architettura ANSI/SPARC: schemi

**schema logico:** descrizione dell'intera base di dati nel modello logico "principale" del DBMS

**schema esterno:** descrizione di parte della base di dati in un modello logico ("viste" parziali, derivate, anche in modelli diversi)

**schema fisico:** rappresentazione dello schema logico per mezzo di strutture fisiche di memorizzazione.

# Una vista

Corsi

<b>Corso</b>	<b>Docente</b>	<b>Aula</b>
Basi di dati	Rossi	DS3
Sistemi	Neri	N3
Reti	Bruni	N3
Controlli	Bruni	G

Aule

<b>Nome</b>	<b>Edificio</b>	<b>Piano</b>
DS1	Ex-OMI	Terra
N3	Ex-OMI	Terra
G	Pincherle	Primo

CorsiSedi

<b>Corso</b>	<b>Aula</b>	<b>Edificio</b>	<b>Piano</b>
Sistemi	N3	Ex-OMI	Terra
Reti	N3	Ex-OMI	Terra
Controlli	G	Pincherle	Primo

# Indipendenza dei dati

L'accesso avviene solo tramite il livello esterno (che può coincidere con il livello logico)

- **indipendenza fisica**

- aggiunte o modifiche allo schema logico si riflettono sullo schema fisico in modo trasparente all'utente;
- modifiche (es. ottimizzazioni) allo schema fisico devono lasciare inalterato lo schema logico

- **indipendenza logica**

- aggiunte o modifiche alle viste non richiedono modifiche al livello logico;
- modifiche allo schema logico che lascino inalterato lo schema esterno sono trasparenti

# Linguaggi di interrogazione

## **Ricordiamo:**

- Permettono di trovare un dato basandosi **sulle sue proprietà**.
- Permettono di trovare dati basandosi su **confronti tra i contenuti** di più tabelle.

# Linguaggi di interrogazione

- **Es:** tabella STUDENTI → *trovare tutti gli studenti che hanno preso più di 28 a Fondamenti di Informatica*
- Permettono di trovare dati basandosi su *confronti* tra i contenuti di più tabelle.
- **Es:** tabella ELEZIONI, tabella PRESIDENTI → *Trovare gli anni in cui è stato eletto un presidente repubblicano proveniente dall'Illinois.*

# CLASSIFICAZIONE DEI LINGUAGGI

- LINGUAGGI FORMALI
  - Algebra relazionale
  - Calcolo relazionale
    - Delle tuple
    - Dei domini
  - Datalog
- LINGUAGGI “COMMERCIALI”
  - SQL (Structured Query Language) ←
  - QUEL
  - QBE (Query By Example)



# CLASSIFICAZIONE DEI LINGUAGGI

- **LINGUAGGI DI DEFINIZIONE DEI DATI (DDL)**
  - per creare gli schemi dei dati e definire le loro proprietà
- **LINGUAGGI DI MANIPOLAZIONE DEI DATI (DML)**
  - per aggiornare le istanze dei dati
  - per l'interrogazione dei dati

# Linguaggi commerciali per basi di dati

L'accesso ai dati può avvenire mediante:

1. linguaggi **testuali interattivi** (es. SQL)
2. comandi (come quelli del linguaggio interattivo) **immersi in un linguaggio ospite** (Pascal, C, Cobol, etc.)
3. comandi (come quelli del linguaggio interattivo) **immersi in un linguaggio ad hoc**, 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)
4. **interfacce amichevoli** (es. linguaggi grafici)

# SQL, un linguaggio interattivo

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

<b>Corso</b>	<b>Aula</b>	<b>Piano</b>
Reti	N3	Terra
Sistemi	N3	Terra

# SQL immerso in Pascal (o altro linguaggio ad alto livello)

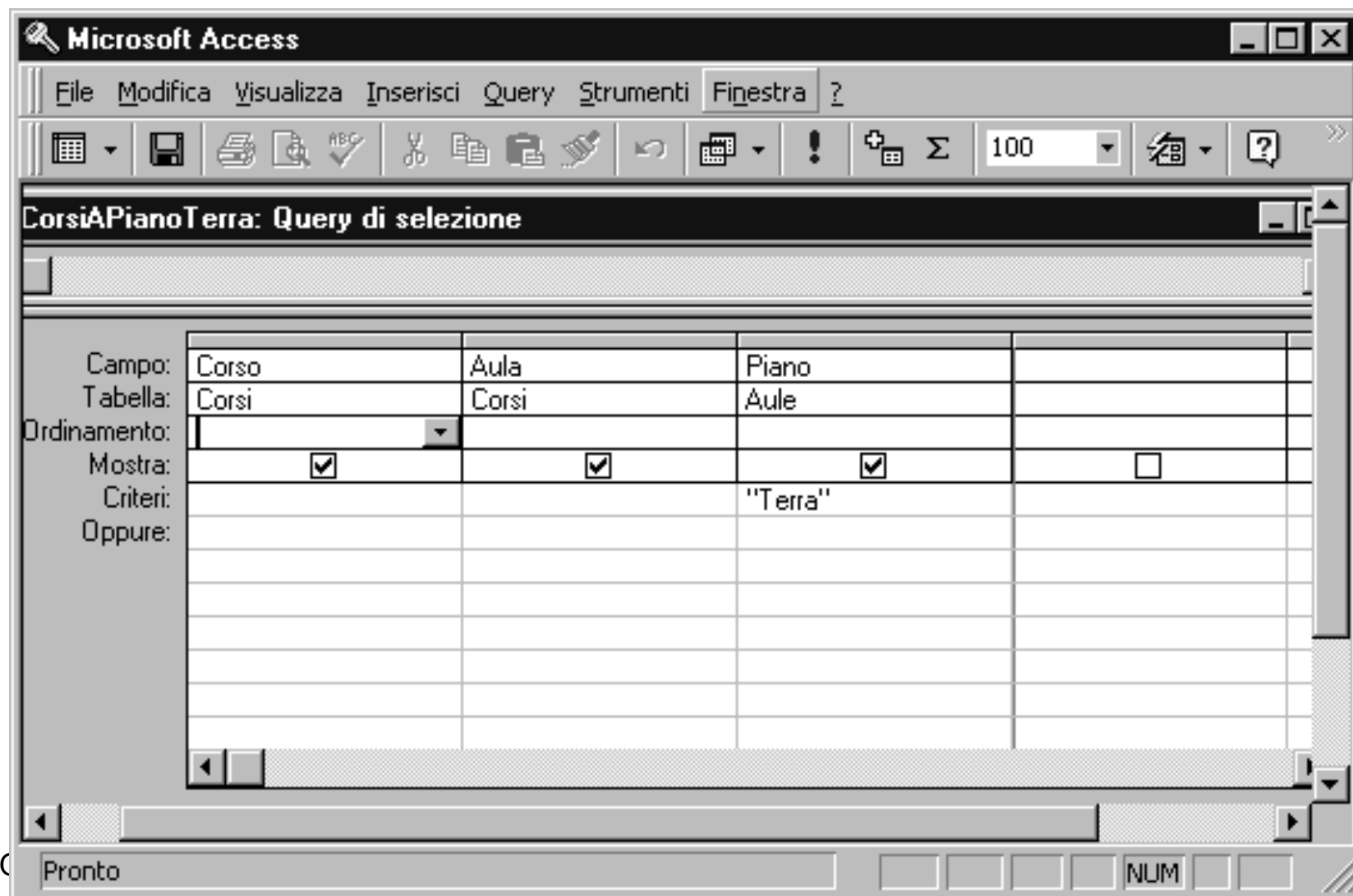
```
write('nome della citta' '?'); readln(citta);
EXEC SQL DECLARE P CURSOR FOR
    SELECT NOME, REDDITO
    FROM PERSONE
    WHERE CITTA = :citta ;
EXEC SQL OPEN P ;
EXEC SQL FETCH P INTO :nome, :reddito ;
while SQLCODE = 0 do begin
    write('nome della persona:', nome, ' aumento?');
    readln(aumento);
    EXEC SQL UPDATE PERSONE SET REDDITO = REDDITO + :aumento
        WHERE CURRENT OF P
    EXEC SQL FETCH P INTO :nome, :reddito
end;
EXEC SQL CLOSE CURSOR P
```

# 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;
```

October 9, 2014 10:41:04 AM EDT

# Access: query by example



# Modello dei dati

- insieme di costrutti utilizzati per organizzare i dati di interesse e descriverne la dinamica
- componente fondamentale: **meccanismi di strutturazione** (o costruttori di tipo)
- ad esempio, il **modello relazionale** prevede il costruttore **relazione (tabella)**, che permette di definire insiemi di record **omogenei**

# Tipi (principali) di modelli dei dati

## → modelli fisici:

- utilizzati nei DBMS esistenti;
- ad essi fa riferimento solo il DBMS;
- Realizzano le strutture fisiche di memorizzazione;

esempi: **strutture indicizzate come alberi bilanciati (B-trees, B+-trees), file sequenziali, file ordinati, file organizzati con strutture hash**

## → modelli logici:

- utilizzati nei DBMS esistenti;
- ad essi fanno riferimento i programmi;
- indipendenti dalle strutture fisiche;

esempi: **relazionale, reticolare, gerarchico, a oggetti**

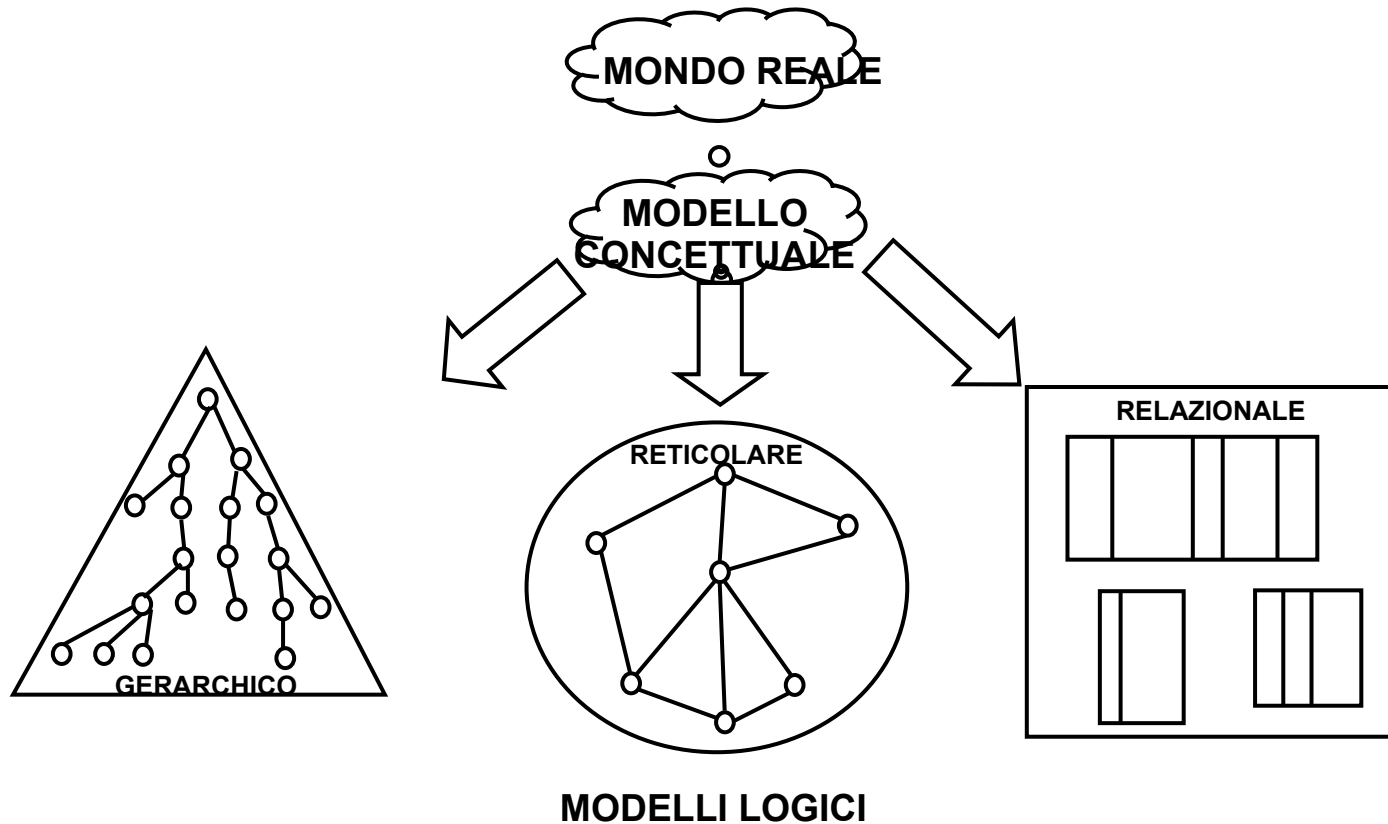
## → modelli concettuali:

- sono di ausilio alla progettazione
- permettono di rappresentare i dati in modo indipendente da ogni sistema, cercando di descrivere i concetti del mondo reale;
- il più noto è il modello Entità-Relazione, ma anche le ontologie, le reti semantiche, etc. , utilizzate nella rappresentazione della conoscenza



# Modelli dei dati e progettazione

i modelli costituiscono una strutturazione semplificata della realtà che ne accoglie aspetti specifici e aiuta a comprenderla meglio



# Modelli dei dati e progettazione

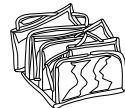
- Fatture



- Ordini dei clienti



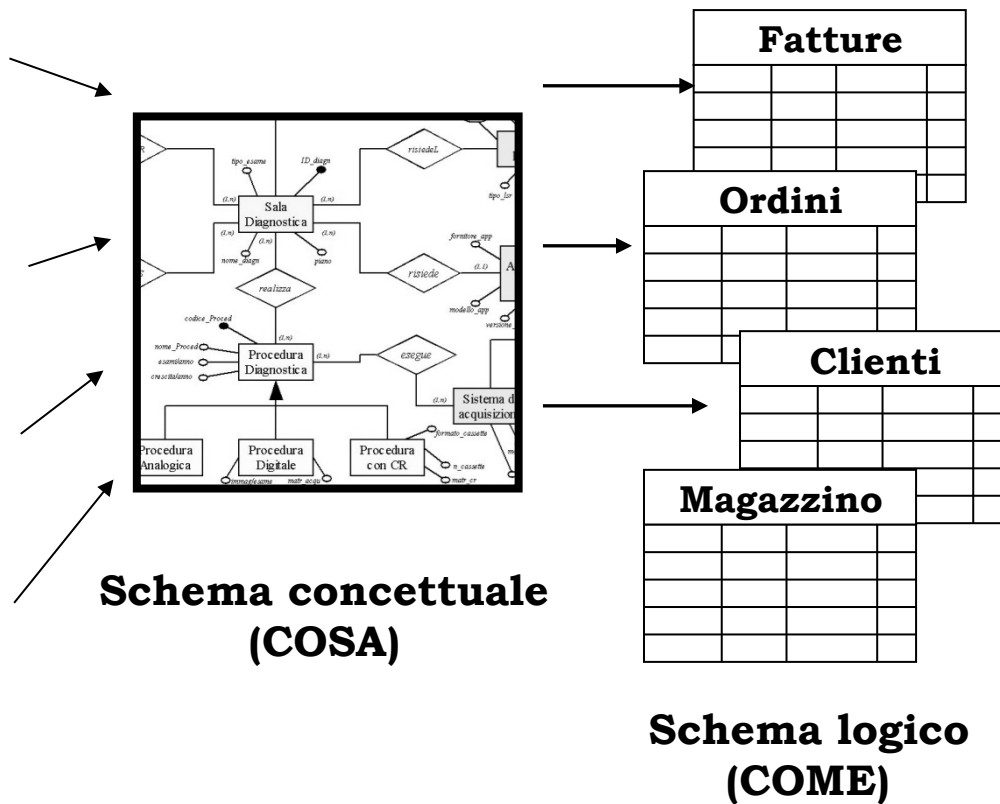
- Indirizzi dei clienti



- Archivio prodotti in magazzino



**Requisiti**



# Personaggi del mondo delle BD

- progettisti e realizzatori di DBMS
- progettisti della base di dati e amministratori della base di dati (DBA)
- progettisti e programmatori di applicazioni
- utenti
  - utenti finali: eseguono applicazioni predefinite (**transazioni**)
  - utenti casuali: eseguono operazioni non previste a priori, usando linguaggi interattivi

# Vantaggi e svantaggi dei DBMS

## **Pro**

- **dati come risorsa comune, base di dati come modello della realtà**
- **gestione centralizzata con possibilità di standardizzazione ed “economia di scala”**
- **disponibilità di servizi integrati**
- **riduzione di ridondanze e inconsistenze**
- **indipendenza dei dati (favorisce lo sviluppo e la manutenzione delle applicazioni)**

## **Contro**

- **costo dei prodotti e della transizione verso di essi**
- **non scorporabilità (spesso) delle funzionalità (con riduzione di efficienza)**