

DATABASE

MODELLAZIONE DATI

TRATTO DA CAMAGNI-NIKOLASSY, CORSO DI INFORMATICA, VOL 2, HOEPLI

Informatica

# Modello e strutture dati

- Un **modello di dati** consiste in una rappresentazione astratta delle strutture dei dati di un database. L'atto di creazione di un modello prende il nome di modellazione dei dati ( o data modeling)
- Le **strutture dei dati** sono tutti gli oggetti del database e le regole che regolano le operazioni tra i dati

# Modello e strutture dati: esempio

- Il *progetto* è il *modello di una casa*
- Progettazione e modellazione, costruzione dell'infrastruttura, costruzione del grezzo, finitura e collaudo
- Così come i disegni di una casa sono indipendenti dai materiali utilizzati per costruirla, il modello dei dati è indipendente dall'hardware o dal linguaggio con cui l'applicazione viene sviluppata

**La modellazione dei dati serve per trasportarli dal mondo reale al mondo informatico**

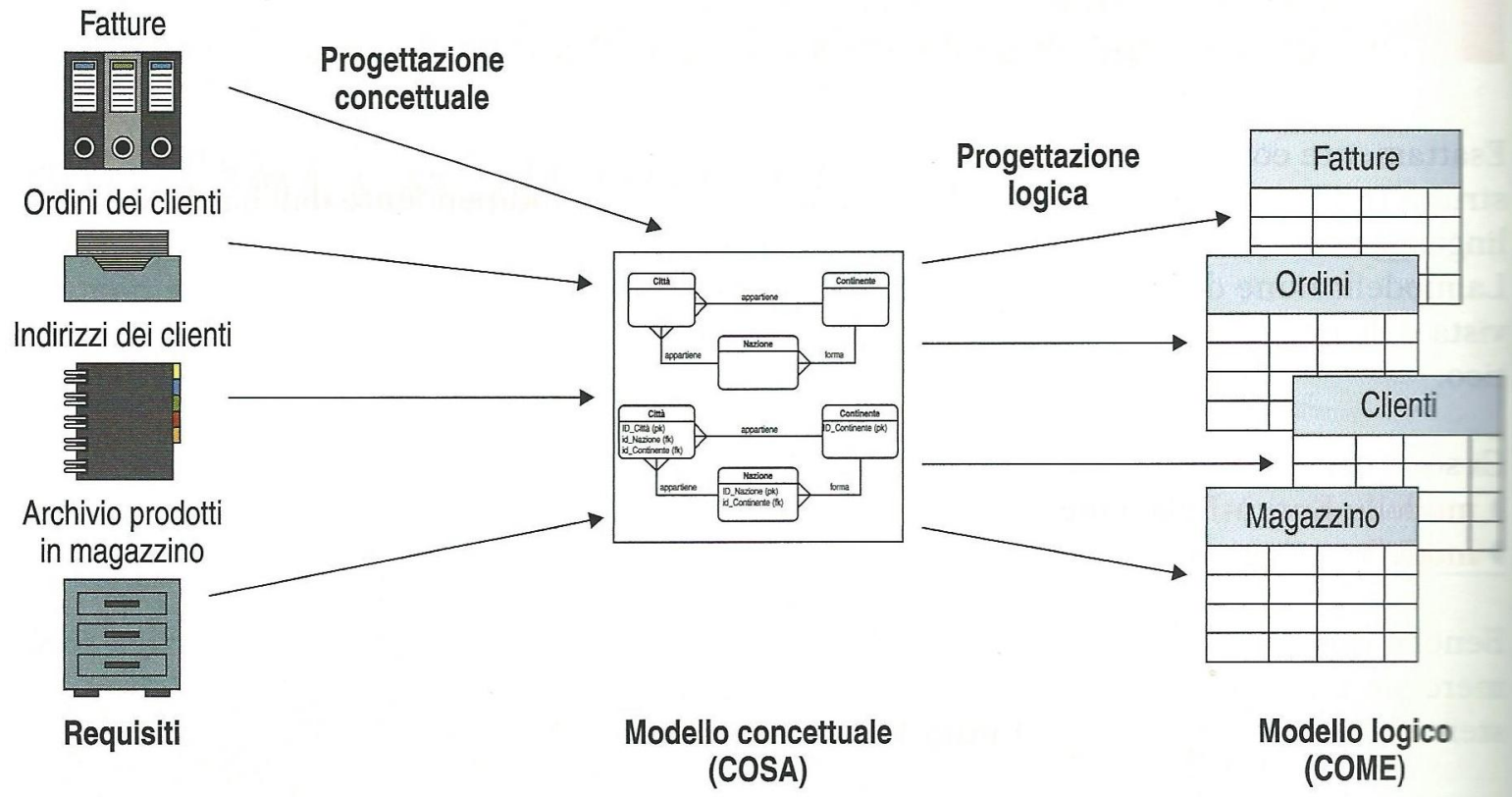
# Analisi, progettazione e implementazione

- Progettare un database significa progettare le strutture che possono accogliere i dati
  - Ciò viene fatto in diversi passi:
    - ▣ Analisi del problema
    - ▣ Progettazione concettuale (modello E-R)
    - ▣ Progettazione logica (schema logico)
    - ▣ Progettazione fisica e implementazione
    - ▣ Realizzazione delle applicazioni
- } Modellazione  
dei dati
- } Modellazione  
funzionale

Il **modello concettuale** descrive **cosa** deve essere rappresentato

Il **modello logico** descrive **come** sono organizzati i dati

# Schema di principio



- La modellazione dei dati consiste nella progettazione delle tabelle del DB
- La modellazione funzionale consiste nell'implementazione delle tabelle e nella creazione delle funzioni che accedono ai dati

# Analisi e progettazione concettuale

- L'analisi serve a capire l'esigenze del cliente
  - ▣ Quali informazioni devono essere considerate?
  - ▣ In che modo devono essere manipolate dal cliente?
- Il frutto della progettazione concettuale è il *diagramma Entità-Relazione* che rappresenta in modo grafico le strutture dati (indipendente dal linguaggio scritto o parlato)
- La fase di modellazione dei dati è una delle più difficili, laboriose e lunghe

Chi passerebbe su un ponte costruito senza piani di costruzione?

# Progettazione logica

- Il modello logico dei dati determina la struttura dei dati
- E' "più vicino" alla rappresentazione informatica dei dati e lo si ottiene con la traduzione dello schema concettuale
- Deve essere:
  - ▣ Indipendente dalle strutture fisiche
  - ▣ Utilizzato dai programmi applicativi

# Cronologia modelli logici

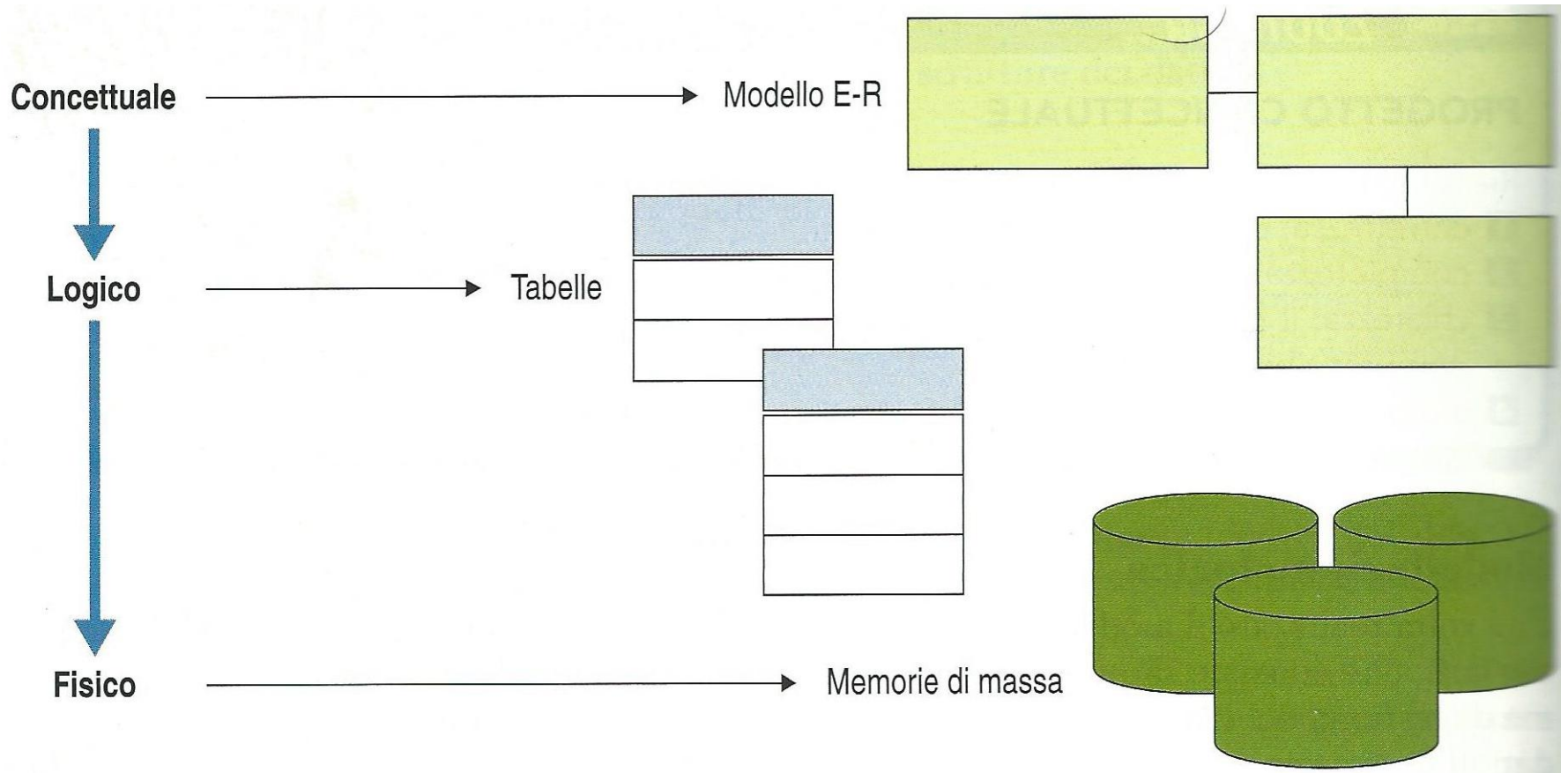
- **Gerarchico:** rappresentabile mediante un albero (anni '60)
- **Reticolare:** rappresentabile mediante un grafo (anni '60)
- **Relazionale:** attualmente il più diffuso, rappresentabile mediante tabelle e relazioni (anni '70)
- **A oggetti:** estensione alle basi di dati del paradigma “Object Oriented” (anni '80)
- **XML:** molto utilizzato come strumento per l'esportazione di dati tra diverse applicazioni (anni '90)



# Implementazione e realizzazione

- In questa fase viene fisicamente realizzato il database sul computer
  - ▣ Viene definita l'organizzazione fisica dei dati su un supporto di memoria e i meccanismi per operare su essi
  - ▣ Progetto e realizzazione delle procedure atte a utilizzare il database utilizzando gli strumenti informatici di alto livello (linguaggi di programmazione e di interrogazione)

# Riepilogo progettazioni

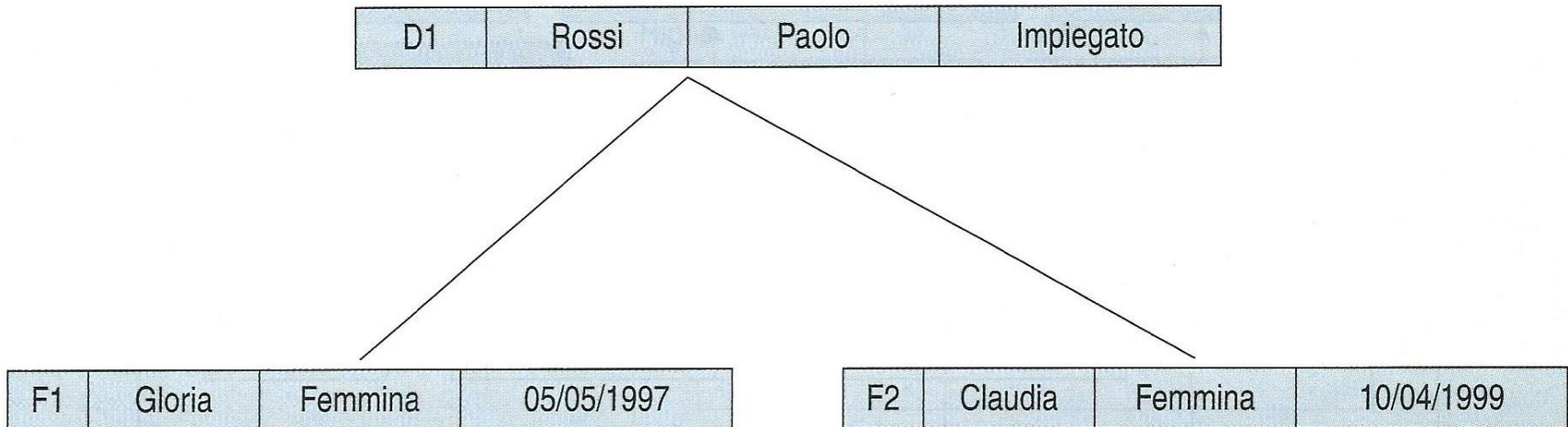


# Modello logico: gerarchico

- I dati sono organizzati secondo strutture ad albero
  - ▣ La **radice** è il record principale da cui partono uno o più sottoalberi ad esso simili
  - ▣ Ogni elemento prende il nome di **segmento**
- La rappresentazione dei dati avviene sfruttando la relazione gerarchica esistente tra segmenti padre e segmenti figlio (relazione 1:N, es. padre-figlio)

# Modello logico: gerarchico

## □ Esempio 1



## □ Esempio 2: Organizzazione del File System

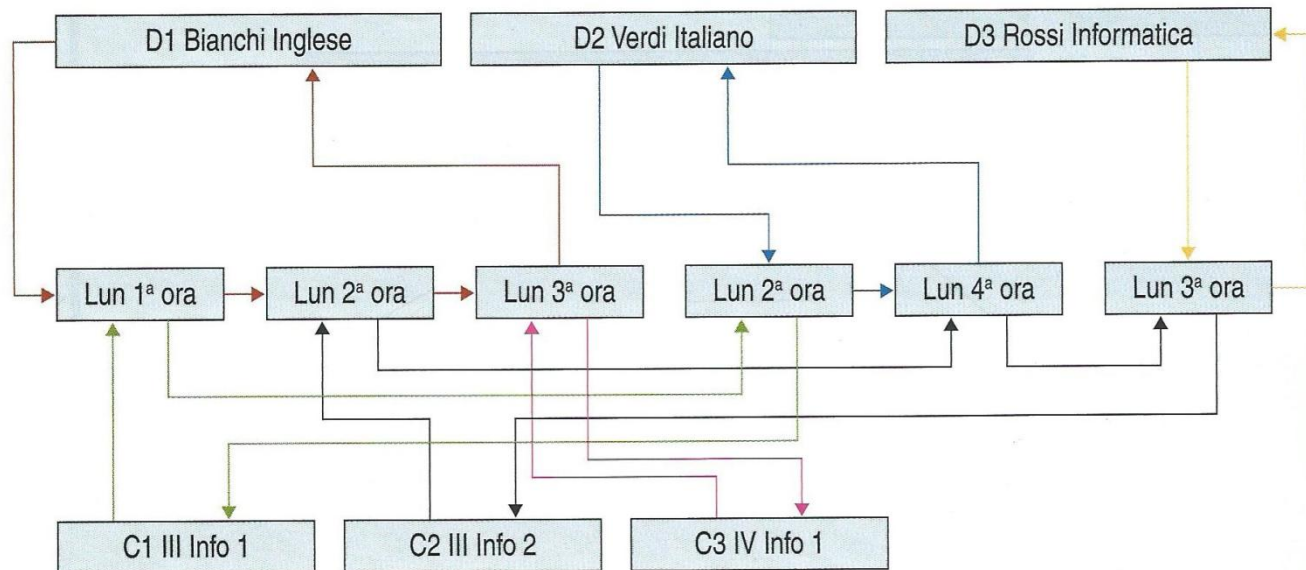
# Modello logico: gerarchico

## □ Svantaggi

- E' facile recuperare le informazioni quando i dati sono di natura gerarchica
- Tra schema logico e fisico esiste una dipendenza stretta e vincolante
- Le operazione di ricerca non sono efficienti

# Modello logico: reticolare

- È una struttura dati a **grafo**
- A differenza del modello gerarchico non esiste nessuna radice ma ogni nodo può essere punto di partenza per raggiungere un determinato campo
- Ogni elemento è costituito da un record che può connettersi con altri N record tramite i **record connettori**



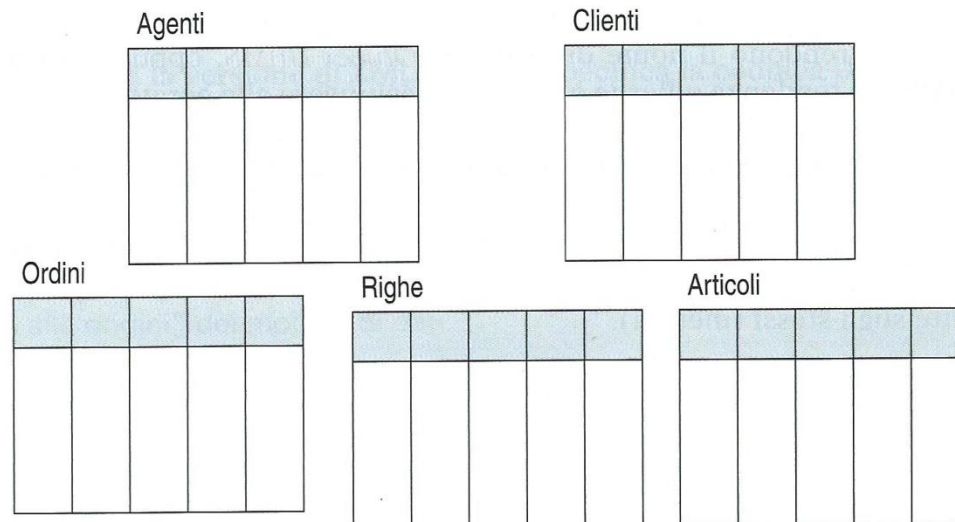
# Modello logico: reticolare

## □ Svantaggi

- È complesso apportare modifiche una volta che è stato realizzato
- I link sono realizzati con i puntatori, quindi esiste uno spreco di spazio di memoria
- Per realizzare due reticoli indipendenti bisogna duplicare i dati con conseguente ridondanza

# Modello logico: relazionale

- Consiste in un insieme di **relazioni** (o tabelle) e connessioni tra di esse
- Sono definite un numero ristretto di operazioni fondamentali
- Ogni tabella è composta da:
  - ▣ Colonne, che sono i diversi campi (o proprietà)
  - ▣ Righe, che sono i record





# Modello logico: relazionale

- Vantaggi
  - ▣ Non c'è duplicazione dei dati
  - ▣ Indipendenza dei dati (tra software applicativo e struttura fisica e logica)
  - ▣ La rappresentazione logica prescinde dalla realizzazione fisica
  - ▣ L'informazione è contenuta nei campi e non in strutture fisiche come i puntatori
- Esempi di DB relazionali: MySQL, SAPDB, Firebird, Access, Oracle.

# Modello logico: a oggetti

- Nasce dalla programmazione *Object-Oriented*
- Si possono definire nuovi tipi di dati e comportamenti che vengono inglobati nell'oggetto stesso
- Si utilizza per applicazioni multimediali, cioè in caso di dati non numerici (immagini, suoni o filmati)
- Attualmente si tende ad aggiungere alle caratteristiche dei sistemi a oggetto i vantaggi del modello relazionale, ottenendo un ibrido *Object-Relational DBMS*

# Modello logico: XML

- Non è un vero e proprio modello, ma è importante per scambiare le informazioni tra DBMS diversi
- E' un linguaggio di markup (tag) come HTML
- I tag possono essere scelti dal programmatore

```
<?xml version="1.0" encoding="UTF-8"?>
<utenti>
  <utente>
    <nome>Mario</nome>
    <cognome>Rossi</cognome>
    <indirizzo>via Pascoli,2</indirizzo>
    <cap>22100</indirizzo>
    <citta>Como</indirizzo>
  </utente>
  <utente>
    <nome>Giuseppe</nome>
    <cognome>Verdi</cognome>
    <indirizzo>via Da Vinci,3</indirizzo>
    <cap>20100</indirizzo>
    <citta>Milano</indirizzo>
  </utente>
</utenti>
```

# Conclusioni

- Un DBMS è un sistema software che gestisce efficientemente grandi quantità di dati, persistenti e condivisi
- Un DBMS deve garantire:
  - ▣ *Affidabilità e consistenza dei dati* (controllo degli accessi e della concorrenza)
  - ▣ *Sicurezza dei dati*, riguardo a danneggiamenti
  - ▣ *Segretezza e confidenzialità*, controllo degli accessi
  - ▣ *Integrità dei dati*, conservazione senza perdite