

Basi di dati - primo semestre
uniVR - Dipartimento di Informatica

Mattia Nicolis

Matteo Drago

A.A. 2025-26

Indice

Introduzione	5
Storia delle DBMS	5
Architettura di un DBMS	7
Progettazione di una base di dati	9
Metodologie di progettazione dei dati	10
Modello entità-relazione	11
Entità	11
Relazioni (associazioni)	12
Attributi	13

Introduzione

Storia delle DBMS

La nascita dei sistemi per la gestione di basi di dati ha visto due momenti principali:

anni '60: sviluppo di applicazioni negli **ambienti di ricerca scientifica**

anni '70: sviluppo di applicazioni informatiche in **ambito gestionale**

Si trattava di semplici dispositivi in cui gli algoritmi di elaborazione erano semplici e grandi quantità di dati erano **condivisi** da più applicazioni.

Queste caratteristiche erano specifiche per l'ambiente in cui erano state introdotte ovvero il **sistema informativo**.

Definizione - Sistema informativo

L'insieme delle attività umane e dei dispositivi di memorizzazione ed elaborazione che organizza e gestisce l'informazione di interesse per una organizzazione di dimensioni qualsiasi.

N.B: non per forza è contenuta tecnologia informatica

Il sistema informativo è costituito da *dati* e *informazioni*:

- **dato:** elemento di conoscenza di base costituito da simboli che devono essere elaborati
- **informazione:** interpretazione dei dati che permette di ottenere conoscenza più o meno esatta di fatti e situazioni

Lo studio del sistema informativo avviene attraverso i **diagrammi di flusso**, questi permettono di svolgere diverse operazioni tra cui:

- definizione archivi dati e delle sorgenti di dati
- definizione degli utenti
- definizione di procedure e processi
- definizione dei flussi dati

Come si è arrivati allo sviluppo dei DBMS?

Negli anni '70, i programmi comunicavano direttamente con i dati contenuti nel file system.

Era un'operazione scomoda in quanto l'accesso ai dati su file era scarso (**struttura ad accesso sequenziale**), c'era ridondanza nei dati (**uplicazioni dello stesso dato su più file**), inconsistenza dovuta ad aggiornamenti parziali e progettazione dei dati replicata per ogni programma.

Negli anni '80, la soluzione che venne proposta fu quella di inserire un **DBMS** (Data Base Management System) tra il **file system** e le **applicazioni**.

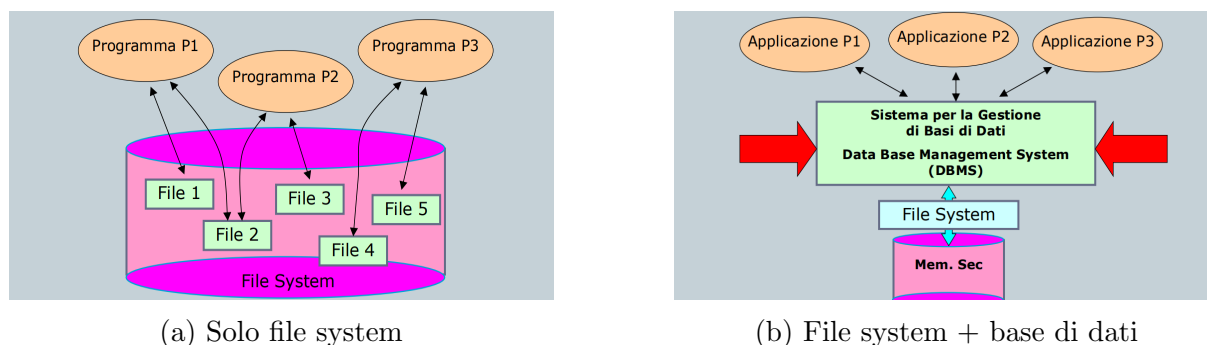


Figura 1: Prima e dopo l'innovazione

Definizione - Base di dati

Una collezione di dati utilizzati per rappresentare con tecnologia informatica le informazioni di interesse per un sistema informativo.

Definizione - DBMS

Sistema che gestisce su memoria secondaria collezioni di dati **grandi**, **condivise** e **persistenti**, assicurandone l'**affidabilità**, la **privatezza** e l'**accesso efficiente**.

Questo nuovo approccio ha portato numerosi vantaggi:

- **maggiore astrazione** e più potenza espressiva per descrivere le proprietà del dato. All'interno del **DMBS** i dati vengono interpretati come oggetti ovvero istanze di classi o righe di tabelle.

Prima erano interpretati come blocchi o pagine di memoria secondaria (sequenze di byte)

- operazioni di accesso ai dati più complesse basate su un linguaggio di interrogazione (SELECT FROM WHERE) anziché attraverso operazioni di READ/WRITE
- migliorata l'interazione uomo-informazione:
 - linguaggio per la definizione dei dati (Data Definition Language - DDL)

- linguaggio per l'interrogazione e aggiornamento dei dati (Data Manipulation Language – DML):
 - * linguaggio di interrogazione: estrae informazioni da una base di dati (esempio: SQL, algebra relazionale)
 - * linguaggio di manipolazione: popola la base di dati, modifica il suo contenuto con aggiunte, cancellazioni e variazioni sui dati (es. SQL)

Definizione - DBMS: modello dei dati

È l'insieme dei costrutti forniti dal DBMS per descrivere la struttura e le proprietà dell'informazione contenuta in una base di dati.

I costrutti permettono di definire le strutture dati che conterranno le informazioni e specificare le proprietà che dovranno soddisfare le istanze.

Nel passato esistevano modelli come quello reticolare o quello gerarchico.

Attualmente si parla di modello relazionale, ad oggetti, object-relational (SQL99 o SQL3), basato sui documenti(JSON) o NoSQL.

Definizione - Schema di una base di dati

È la descrizione della struttura e delle proprietà di una specifica base di dati fatta utilizzando i costrutti del modello dei dati (lo schema di una base di dati è invariante nel tempo).

Definizione - Istanza di una base di dati

È costituita dai valori effettivi che in un certo istante popolano le strutture dati della base di dati (l'istanza di una base di dati varia nel tempo).

Architettura di un DBMS

- **schema logico**: è la rappresentazione della struttura e delle proprietà della base di dati definita attraverso i costrutti del modello dei dati del DBMS
- **schema interno**: è la rappresentazione della base di dati per mezzo delle strutture fisiche di memorizzazione (file dati, file indice, ecc...)
- **schema esterno**: descrive una porzione dello schema logico di interesse per uno specifico utente o applicazione (attraverso viste sullo schema logico)

La caratteristica fondamentale di queste strutture è l'**indipendenza**.

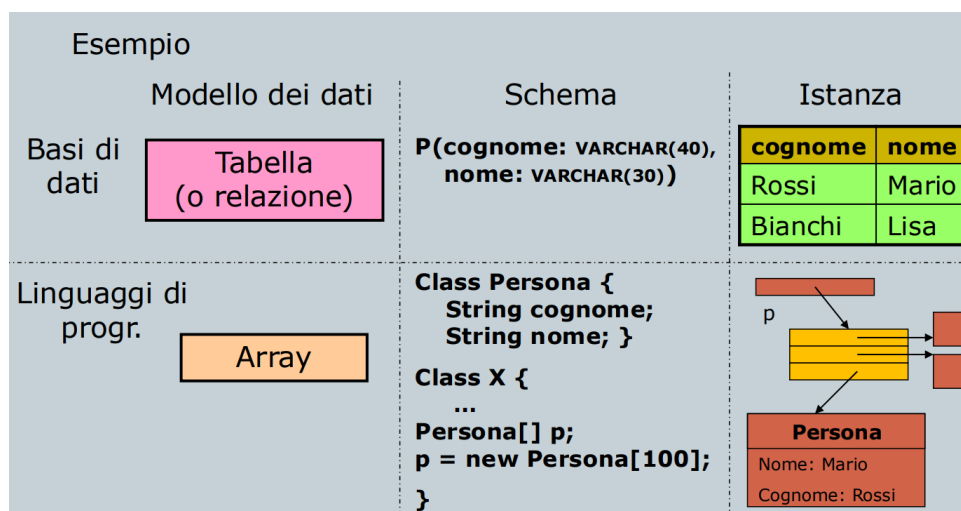


Figura 2: Distinzione tra modello, schema e istanza

indipendenza fisica: lo schema logico della base di dati è completamente indipendente dallo schema interno, ne consegue che le variazioni delle strutture fisiche non impattano sullo schema logico e quindi sulle applicazioni.

indipendenza logica: gli schemi esterni della base di dati sono indipendenti dallo schema logico, ne consegue che le variazioni dello schema logico (purchè non vengano rimossi dati) non impattano sugli schemi esterni e quindi sulle applicazioni (eventualmente è necessario solo ridefinire l'espressione di derivazione).

Progettazione di una base di dati

La progettazione di una base di dati costituisce solo una delle componenti del processo di sviluppo di un sistema informativo e va, quindi, inquadrata in un contesto più ampio: il **ciclo di vita** dei sistemi informativi.

Il *ciclo di vita* di un sistema informativo comprende:

1. **studio della fattibilità:** definisce (nel modo più preciso possibile) i *costi* delle *varie alternative* possibili e *stabilisce le priorità di realizzazione* delle varie componenti del sistema
2. **raccolta e analisi dei requisiti:** individua e studia le proprietà e le funzionalità del sistema, producendo una descrizione completa, ma informale, dei dati coinvolti e delle operazioni su di essi.
3. **progettazione:** si divide in
 - *progettazione dei dati:* descrive la struttura e l'organizzazione dei dati
 - *progettazione delle applicazioni:* descrive in modo formale le caratteristiche dei programmi applicativi

Le due attività sono complementari e possono procedere in parallelo o in cascata.

4. **implementazione:** realizza il sistema informativo secondo la struttura e le caratteristiche definite nella fase di progettazione.

Viene costruita e popolata la base di dati e viene prodotto il codice dei programmi.

5. **validazione e collaudo:** verifica il corretto funzionamento e la qualità del sistema informativo.

La sperimentazione deve prevedere, per quanto possibile, tutte le configurazioni operative.

6. **funzionamento:** entra in funzione il sistema informativo ed esegue i compiti per i quali era stato originariamente progettato.

Se non si verificano malfunzionamenti o revisioni delle funzionalità del sistema, questa attività richiede solo operazioni di gestione e manutenzione.

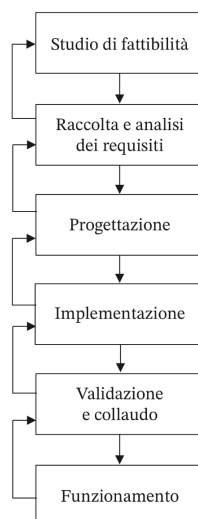


Figura 3: ciclo di vita

Metodologie di progettazione dei dati

Una **metologia di progettazione** è costituita da:

- **decomposizione** dell'intera attività di progetto in passi successivi indipendenti tra loro
- **insieme di strategie** da seguire nei vari passi e alcuni *criteri* per la scelta in caso di alternative
- **modelli di riferimento** per descrivere i dati di ingresso e uscita delle varie fasi

Le *proprietà* che una buona metologia deve avere sono:

- **generale** rispetto alle applicazioni e ai sistemi in gioco (essere indipendenti dal problema studiato e dagli strumenti utilizzati)
- **prodotto di qualità** in termini di *correttezza*, *completezza* ed *efficienza* delle risorse impiegate
- **facile da usare** nelle strategie e nei modelli di riferimento

Nell'ambito delle basi di dati, si è consolidata negli anni una metodologia di progetto che ha dato prova di soddisfare pienamente le proprietà descritte.

Tale metodologia è articolata in tre fasi principali da effettuare a cascata:

- **progettazione concettuale**: rappresenta le specifiche informali della realtà di interesse in termini di una **descrizione formale e completa**, ma *indipendente* dai criteri di rappresentazione utilizzati nei sistemi di gestione di basi di dati (*schema concettuale*)
- **progettazione logica**: traduce lo schema concettuale, definito nella fase precedente, in termini del modello di rappresentazione dei dati adottato dal sistema di gestione di base di dati a disposizione (*schema logico*).

Il modello logico, ci consente di descrivere i dati secondo una rappresentazione ancora indipendente da dettagli fisici, ma concreta perchè disponibile nei sistemi di gestione di base di dati.

- **progettazione fisica:** viene completato lo schema logico con la specifica dei parametri fisici di memorizzazione dei dati (*schema fisico*)

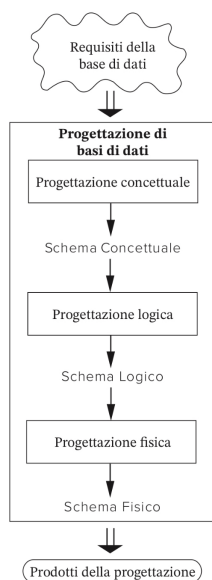


Figura 4: progettazione a cascata

Modello entità-relazione

Il **modello entità-relazione** è un *modello concettuale* di dati e fornisce una serie di strutture (*costrutti*), atte a descrivere la realtà di interesse in una maniera facile da comprendere e che prescinde dai criteri di organizzazione dei dati nei calcolatori.

Questi costrutti vengono utilizzati per definire *schemi* che descrivono l'organizzazione e la struttura delle *occorenze* dei dati, dei valori assunti dai dati al variare del tempo.

Entità

Un'entità **E** rappresenta una classe di oggetti che hanno proprietà comuni ed esistenza "autonoma" ai fini dell'applicazione di interesse (*Città, Dipartimento, impiegato, Acquisto, Vendita* ecc.).

Un'occorenza di un'entità è un oggetto della classe che l'entità rappresenta (città di Roma, Milano e Palermo sono occorrenze dell'entità *Città*).

Un'occorenza di entità ha un'esistenza indipendente dalle proprietà a esso associate.

In uno schema, ogni ha un nome che la identifica univocamente e viene rappresentata graficamente mediante un rettangolo con il nome dell'entità all'interno.

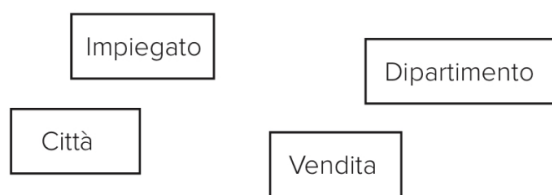


Figura 5: entità

Relazioni (associazioni)

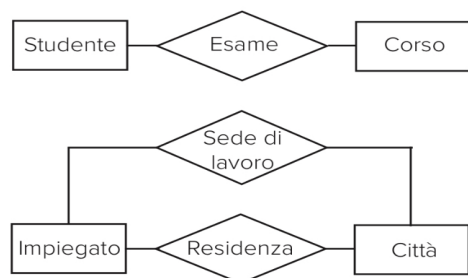
Una **relazione R** **rappresneta un legame logico tra due o più entità** (*Residenza* è una relazione che può sussistere tra le entità *Città* e *Impiegato*).

Un'occorenza di relazione è un'ennupla costituita da occorrenze di entità, una per ciascuna delle entità coinvolte.

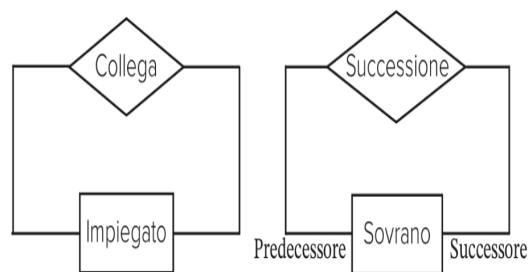
In uno schema E-R, ogni relazione ha un nome che identifica univocamente e viene rappresentata graficamente mediante un rombo con il nome delle realzione all'interno e da linee che connettono la relazione con ciascuna delle sue componenti.

L'insieme delle occorrenze di una relazione del modello E-R è, una relazione matematica tra le occorrenze delle enttà coinvolte, ossa, un sottoinsieme del loro prodotto cartesiano.

Questo significa che tra le occorrenze di una relazione del modello E-E non ci possono essere ennuple ripetute.



(a) relazioni tra entità



(b) relazione ricorsiva

E' anche possibile avere *relazioni ricorsive*: relazioni tra un'entità e se stessa.

E' possibile, infiene, avere *relazioni n-arie* che coinvolgono più di due entità.

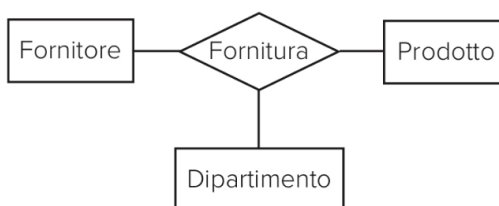


Figura 7: relazione ternaria

Attributi

Un **attributo** **descrive le proprietà elementari di un'entità (o relazione)** (*Cognome*, *Stipendio* ed *Età* sono possibili attributi dell'entità *Impiegato*).

Un attributo associa a ciascuna occorrenza di entità (o di relazione) un valore appartenente a un insieme (*dominio*), che contiene i valori ammissibili per l'attributo.

I domini non vengono riportati nello schema, ma sono generalmente nella documentazione associata.

Può risultare comodo raggruppare attributi della medesima entità (o relazione) che presentano affinità nel loro significato o uso (*attributo composto*), ad esempio *Via*, *Numero civico* e *CAP* possono formare l'attributo composto *Indirizzo* dell'entità *Persona*.

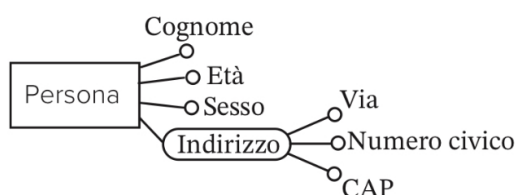


Figura 8: attributo composto

Un esempio completo di **modello E-R**, può essere il seguente:

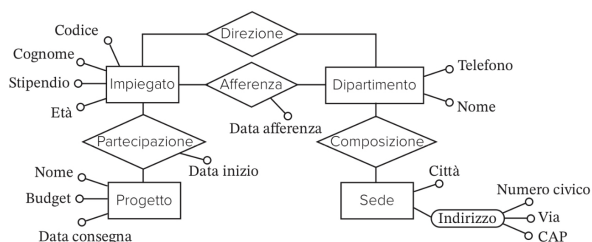


Figura 9: modello E-R completo