



ESERCIZI COMMENTATI
Segui la risoluzione passo passo



LABORATORI "CASE STUDIES"
Approfondisci con i casi pratici

Introduzione alle basi di dati

In questa unità imparerai...

- Che cosa caratterizza un sistema informativo e un sistema informatico
- Che cosa sono le basi di dati
- A riconoscere le differenze tra i vari modelli di dati

1 Il sistema informativo

Le informazioni necessarie a un'organizzazione sono gestite da un sistema informativo.

Un **sistema informativo** è un insieme organizzato di strumenti, procedure manuali, norme organizzative, risorse umane e materiali, orientato alla gestione delle informazioni rilevanti per un'organizzazione. Gestione intesa come raccolta, archiviazione, elaborazione e scambio di informazioni necessarie alle attività operative, di gestione, di preparazione, controllo e valutazione dell'organizzazione.

Esempio 1 – Sistema informativo di una compagnia di trasporti aerei

Il sistema informativo della compagnia deve essere in grado di fornire ai clienti un efficiente sistema di prenotazione dei posti sui voli, di gestire il personale di terra e il personale di volo, di gestire la manutenzione degli aerei, la contabilità, il magazzino dei pezzi di ricambio, il rifornimento degli aerei e così via.

Ovviamente, il flusso di informazioni da gestire (e quindi da memorizzare) è enorme, e dalla sua gestione dipendono l'efficienza e anche la qualità del servizio offerto dalla compagnia.

Esempio 2 – Sistema informativo della segreteria di una scuola

Anche in questa realtà lavorativa, le entità che devono essere previste per descrivere le informazioni da elaborare dipendono dalle funzionalità che il sistema informativo deve gestire.

Elementi essenziali sono le entità relative agli studenti, alle pagelle e agli insegnanti in modo da gestire le principali attività espletate da una segreteria didattica. Se si volesse ampliare il sistema informativo, si potrebbero prevedere le entità relative ai permessi e alle assenze degli insegnanti, quelle relative ai corsi di recupero e così via.

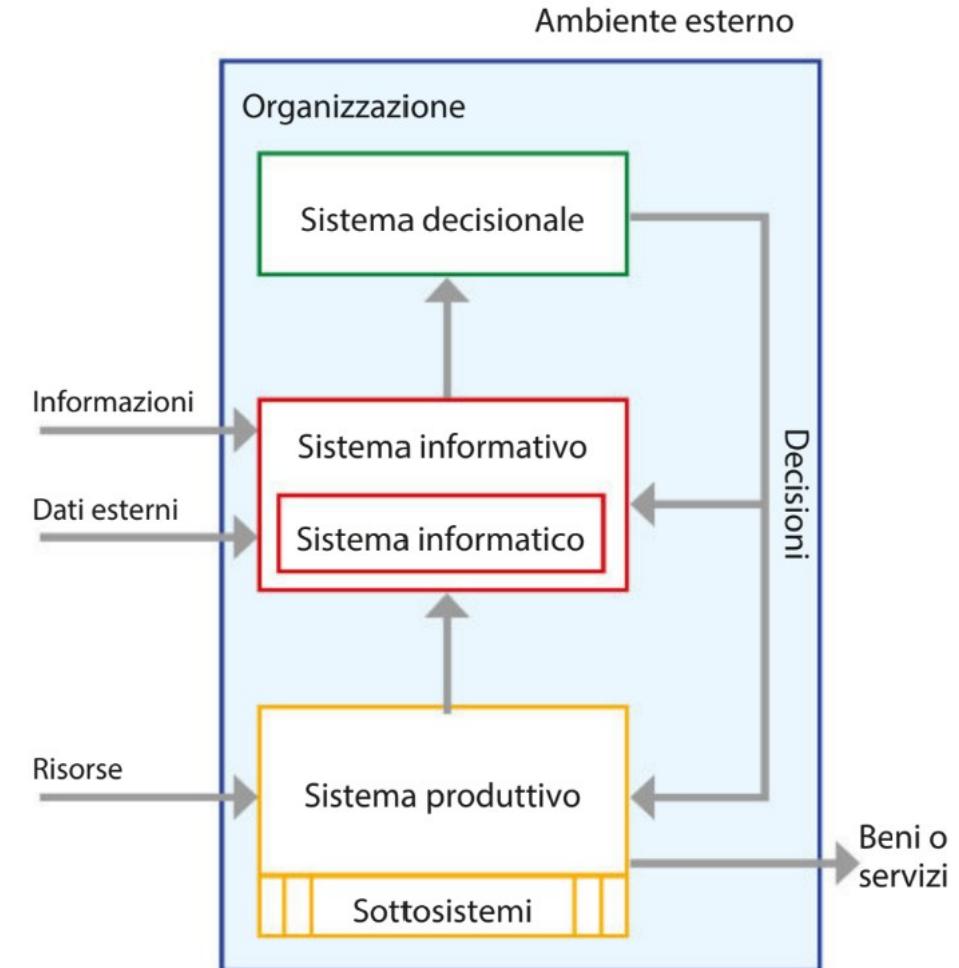
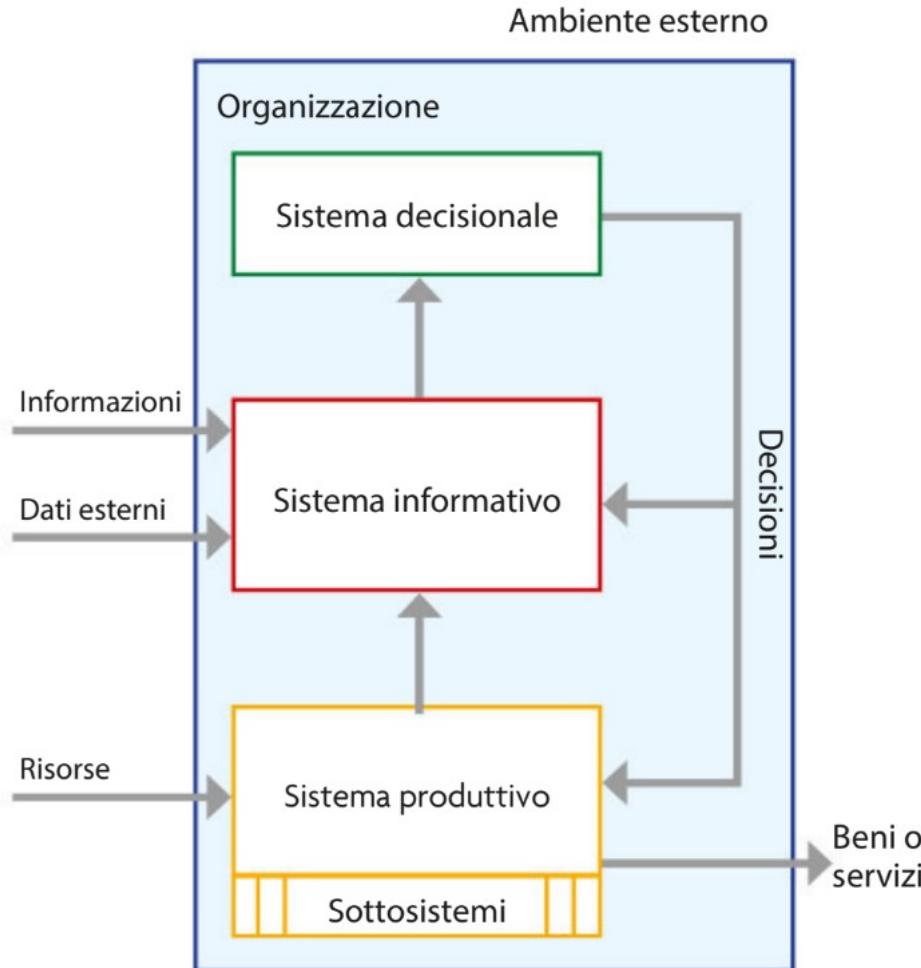
2 Il sistema informatico

La parte del sistema informativo di un'organizzazione che può essere automatizzata è chiamata sistema informatico.

Il **sistema informatico** è l'insieme degli strumenti informatici utilizzati per il trattamento automatico delle informazioni, rappresentate mediante dati digitali, al fine di agevolare le funzioni del suo sistema informativo.

In altri termini, il sistema informatico è un sistema informativo che si avvale di tecnologie informatiche.

Se consideriamo l'esempio della compagnia aerea, il sistema informatico è costituito dagli archivi elettronici in cui sono memorizzati i dati di interesse relativi ai clienti, agli aerei, al personale, dalle componenti fisiche che costituiscono il supporto di memorizzazione, dalle procedure di interrogazione per la ricerca delle informazioni, dalle reti di comunicazione tra i terminali degli operatori e così via.



3 Che cos'è una base di dati

In ogni modello di organizzazione della vita dell'uomo, dal più complesso al più semplice, vengono trattate **informazioni** che sono risorse preziosissime grazie alle quali la stessa organizzazione può sopravvivere.

Una volta individuate e raccolte, tali informazioni devono necessariamente essere memorizzate, in modo che si possa facilmente:

- recuperarle in base a determinati criteri di ricerca;
- aggiungerne di nuove;
- modificarle, apportando variazioni nel tempo;
- cancellare quelle non più necessarie.

Pensa a quante informazioni sono necessarie per gestire un ufficio anagrafico comunale, oppure per la contabilità di una grossa azienda, o per rendere efficiente un sistema di prenotazione aereo o ferroviario.

In campo informatico, la **teoria delle basi di dati** studia come organizzare al meglio grandi quantità di informazioni, per poterle gestire in modo:

- **semplice**, in quanto le applicazioni devono essere facilmente fruibili in applicazioni diverse e da parte di differenti utenti;

- **efficiente**, perché l'utilizzo delle risorse deve essere ottimizzato:
 - “in tempo” (efficiente utilizzo del microprocessore);
 - “in spazio” (efficiente utilizzo della memoria);

- **efficace**, nel senso che le informazioni devono essere rappresentative della realtà che si vuole analizzare (ad esempio, un’età non può essere negativa);

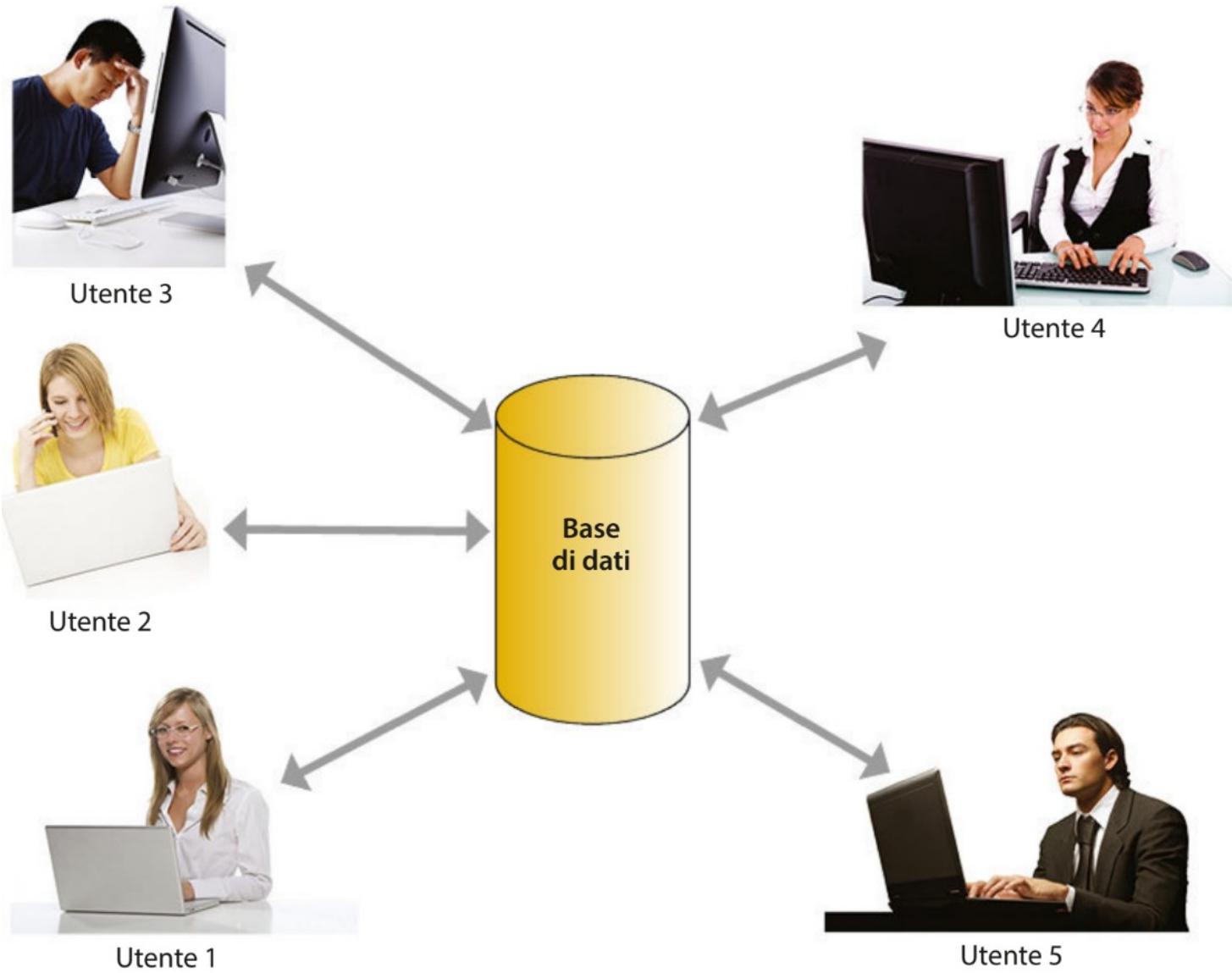
- **sicuro**, in quanto le operazioni sui dati sono permesse solo a soggetti identificabili e autorizzati.

Ora, però, la domanda nasce quasi spontanea: che cos'è una base di dati?
Da quanto detto sinora possiamo dare una prima definizione:

Una **base di dati** (in inglese **database**; da ora in poi utilizzeremo entrambi i termini) può essere considerata come una raccolta di dati logicamente correlati, utilizzata per modellare una realtà.

I dati sono memorizzati su un supporto di memoria di massa e sono progettati per essere fruiti in maniera ottimizzata da differenti applicazioni e utenti diversi.

Pensiamo a una banca. L'insieme dei dati relativi ai clienti, alle transazioni, ai conti, ai dipendenti e così via costituisce la base di dati. Ciò significa che, anche se le entità dei dati sono differenti tra loro (quelli che riguardano i conti bancari, ad esempio, sono di altro genere rispetto a quelli che riguardano i dipendenti e i clienti), appartengono comunque alla medesima realtà, ovvero alla realtà di quella particolare agenzia bancaria.



4 Dati e informazioni: schemi e istanze

In un sistema informatico, tutti i fatti vengono rappresentati per mezzo di dati che divengono informazioni solo dopo essere stati interpretati.

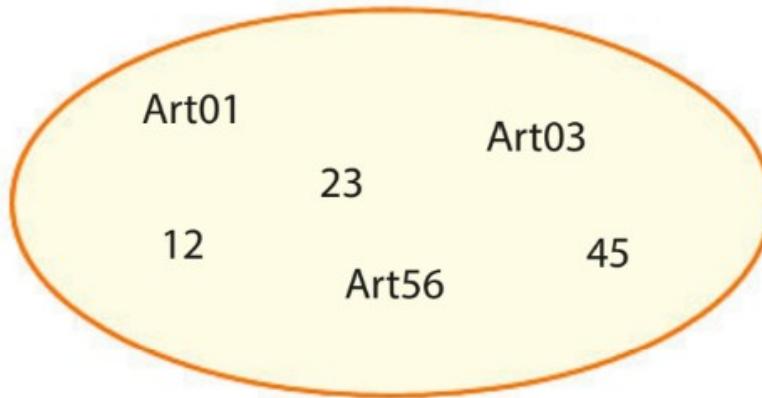
In informatica un **dato** (dal latino *datum*, plurale *data*) è la descrizione elementare di una cosa, di un avvenimento. Lo scopo dei dati è quello di codificare in vari modi i fatti (anche ipotetici) ritenuti importanti nell'ambito di un'organizzazione.

I dati sono una codifica, una rappresentazione dei fatti presenti in un'organizzazione. Un dato costituisce un'informazione per qualcuno se fornisce una nuova conoscenza.

L'**informazione** è, infatti, l'incremento di conoscenza che può essere acquisita (inferita) dai dati.

I dati sono utilizzabili (danno informazioni) se si attribuisce loro una **chiave di interpretazione** che consente di comprendere il loro **significato** (*semantica*), cioè i fatti che essi codificano.

I dati dell'insieme della figura seguente non forniscono alcuna informazione utile finché non si definisce che cosa rappresenta ciascun dato e le eventuali relazioni esistenti tra i dati stessi.



Il dato 23, senza chiave di interpretazione che gli dia un significato, non costituisce informazione per nessuno. Lo stesso dato 23, invece, con la chiave di interpretazione “Articoli in magazzino” assume il significato di “quantità disponibile per un determinato articolo”.

Possiamo dare un significato al precedente insieme con la seguente forma tabellare:

ARTICOLI IN MAGAZZINO	
Codice articolo	Quantità
Art01	23
Art03	12
Art56	45

Chiameremo **schema** o **intensione** la chiave di interpretazione dei dati, ovvero il significato (la semantica) che si attribuisce al dato per ricavare l'informazione da esso completata.

Chiameremo invece **istanza di uno schema** o **estensione** l'insieme dei valori assunti da uno schema in un certo istante di tempo.

Un'istanza dello schema precedente è quella composta dai seguenti tre articoli e dalle loro rispettive quantità:

Art01	23
Art03	12
Art56	45

Parleremo, quindi, di:

significato intensionale (o **schema**) riferendoci al contenuto informativo dei dati: la ricostruzione dei fatti (o significato completo dei dati) nasce dunque dall'interpretazione dei dati;
significato estensionale (o **istanza**) dei dati riferendoci ai valori che può assumere uno schema in un certo istante.

Chiamiamo **categoria** un gruppo di dati aventi la stessa chiave di interpretazione, ovvero lo stesso schema.

Nell'esempio precedente avevamo la categoria “Articoli in magazzino”. Di essa fa parte la seguente istanza, ovvero particolari valori assunti dalla categoria.

ARTICOLI IN MAGAZZINO		CATEGORIA
Codice articolo	Quantità	ATTRIBUTO
Art01	23	
Art03	12	ISTANZA
Art56	45	VALORE

Una **occorrenza** di una base di dati è l'insieme delle istanze delle categorie in un determinato istante di tempo.

Un **DBMS** (*Data Base Management System*) è un insieme di strumenti software che, sulla base delle specifiche dell'utente, è in grado di gestire dati strutturati che sono tanti, importanti, condivisi, sia interrogati che aggiornati. Provvede inoltre alla generazione dello schema, alla ricerca e all'aggiornamento dei dati. Nel caso specifico dei database relazionali, che tratteremo nelle prossime unità, si parla di **RDBMS** (*Relational Data Base Management System*).

Il DBMS rappresenta, quindi, un'interfaccia tra gli sviluppatori, gli utenti del database e il sistema di elaborazione. Ovviamente, fra DBMS e base di dati esiste una forte interazione: il DBMS è un sistema attivo, mentre la base di dati è l'oggetto passivo su cui esso opera. Quando si usa un DBMS, i sistemi informativi possono essere adeguati molto facilmente al cambiamento delle richieste informative dell'organizzazione. Si possono aggiungere al database nuove categorie di dati senza stravolgere il sistema esistente.

6 Livelli di astrazione di un DBMS

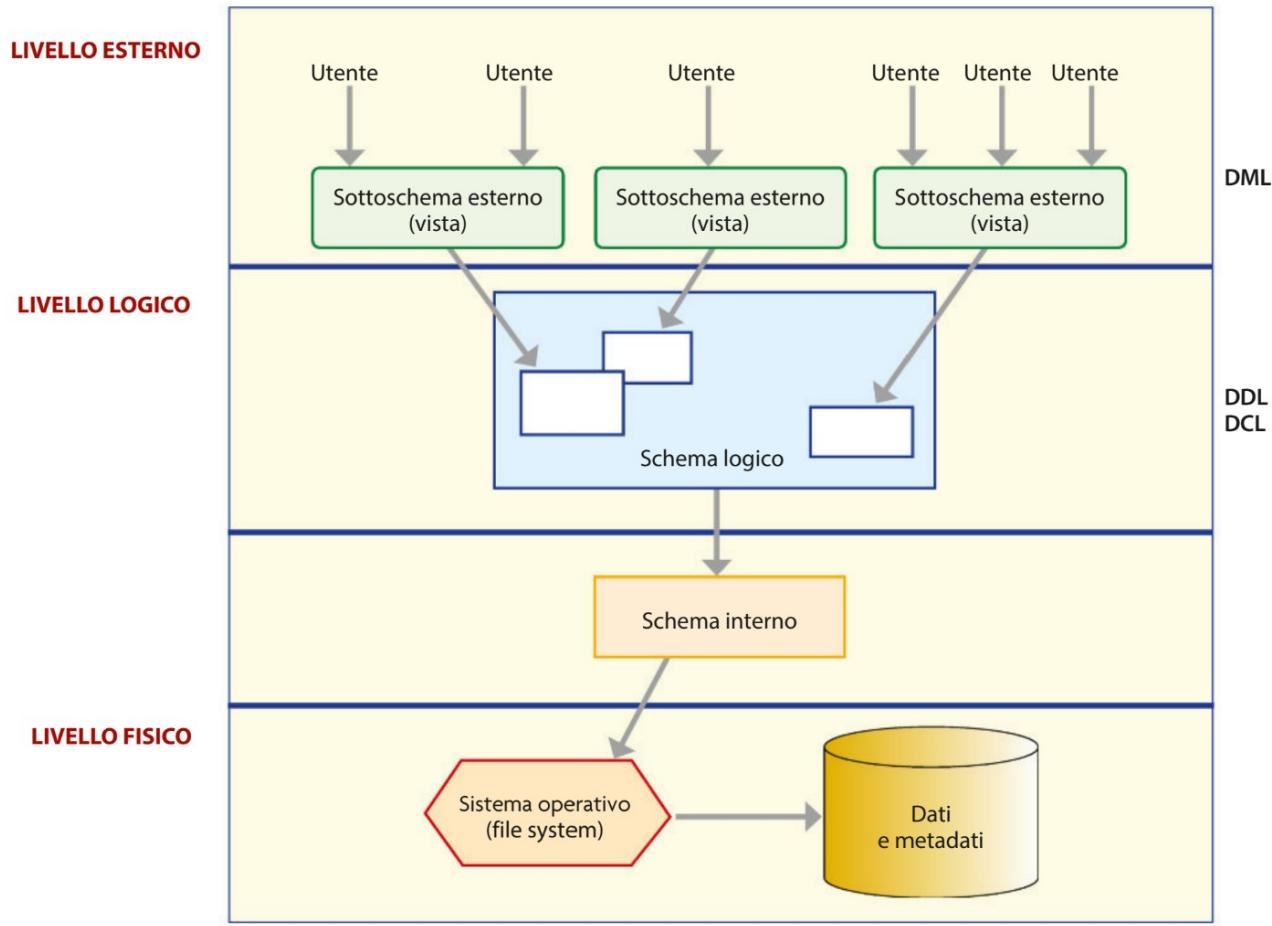


ORACLE



Una delle funzioni del DBMS è quella di mantenere un modello astratto dei dati che consenta all'utente di concepire le informazioni secondo strutture riferite direttamente alle entità reali che i dati sono preposti a rappresentare. Gli utenti di un DBMS vedono una macchina astratta (**macchina DBMS**) che consente loro di compiere specifiche operazioni per descrivere e manipolare i dati contenuti nel database, utilizzando appositi linguaggi. L'architettura di questa macchina virtuale e le sue relazioni con l'esterno sono rappresentate nella figura a pagina seguente.

- Il **livello esterno** è quello con cui interagiscono i singoli utenti del database, attraverso specifiche applicazioni, in quanto fornisce gli strumenti utili per accedere ai dati e modificarli. Come si evince dalla figura, ogni utente può essere interessato solo a un sottinsieme dei dati contenuti nel database. Ciò significa che è possibile limitare l'accesso ai dati ad alcuni utenti, realizzando così accessi e navigazioni personalizzate che danno a ciascuno la sensazione di essere l'unico utilizzatore dei dati che vede, anche se in realtà non è così. È compito dell'amministratore di sistema (**DBA, Data Base Administrator**) fornire a ciascun utente un sottoschema del database logico, ossia una *vista*, che contenga tutto e solo ciò che l'utente desidera o è abilitato a usare.



Una **vista** è l'astrazione di una parte del database concettuale che coinvolge i dati dell'istanza del database limitatamente alla porzione interessata o consentita.

- Il **livello concettuale o logico** rappresenta la struttura globale del database, relativa a tutte le informazioni in esso presenti. In questo livello troviamo il **database logico**, costituito dalla rappresentazione astratta del database indipendente dall'implementazione fisica, e che può essere visto come la definizione dell'intero schema dei dati. A questo scopo il DBMS mette a disposizione il linguaggio **DDL** (*Data Definition Language*), utilizzato per descrivere le caratteristiche delle categorie di dati presenti nel database e le corrispondenze esistenti tra di esse. A questo livello è disponibile anche il **DCL** (*Data Control Language*) che permette operazioni relative al controllo degli accessi degli utenti.

- Il **livello interno o fisico** è quello usato effettivamente per la memorizzazione dei dati. Qui troviamo il **database fisico**, costituito dall'implementazione del database logico. Esso considera i tipi di dati, i formati, le strutture di memorizzazione e i metodi di accesso; come tale, rappresenta la forma in cui il database viene memorizzato e usato. È facile intuire che l'efficienza del database dipende molto dal grado di ottimizzazione raggiunto nell'organizzazione dei dati all'interno di questo livello, operazione realizzata con un apposito linguaggio chiamato **DMCL** (*Device Media Control Language*). Per l'utente il livello fisico è del tutto trasparente: egli, infatti, non si preoccupa affatto di come i dati vengano registrati sui supporti, tale funzione è compito esclusivo del DBMS. L'utente si occuperà principalmente di ciò che vi è registrato: quali sono i dati e in quale relazione si trovano tra di loro. Poiché la gestione dei file è realizzata dal sistema, l'utente non deve interessarsi della forma assunta dai dati in memoria di massa; non gli resta dunque che occuparsi del loro valore informativo.

A questo punto siamo in grado di riassumere quanto detto per i tre livelli:

- il **livello fisico** riguarda l'effettiva memorizzazione dei dati, organizzati in file, record e strutture di accesso;
- il **livello concettuale** riguarda la struttura logica assunta dai dati registrati, quindi il loro schema astratto;
- il **livello esterno** si riferisce al modo in cui ciascun utente può vedere gli stessi dati, che mantengono l'organizzazione fisica e concettuale precedentemente descritte, ma vengono messi a disposizione secondo il formato richiesto.

Con l'organizzazione a tre livelli, un DBMS permette di ottenere **indipendenza fisica e logica** dei dati.

Con indipendenza fisica si intende la possibilità di modificare l'organizzazione fisica dei dati senza dover conseguentemente modificare l'organizzazione logica e i programmi applicativi che agiscono sul database.

Per indipendenza logica, invece, si intende la possibilità di modificare lo schema logico del database senza dover modificare i programmi applicativi non interessati alla modifica.

7 Il modello dei dati

La più diffusa classificazione delle basi di dati si basa sul modello di dati che essa supporta.

Un **modello di dati** è un insieme di concetti e di costrutti utilizzati per organizzare i dati di interesse e descriverne la struttura e la dinamica (associazioni e vincoli che devono rispettare).

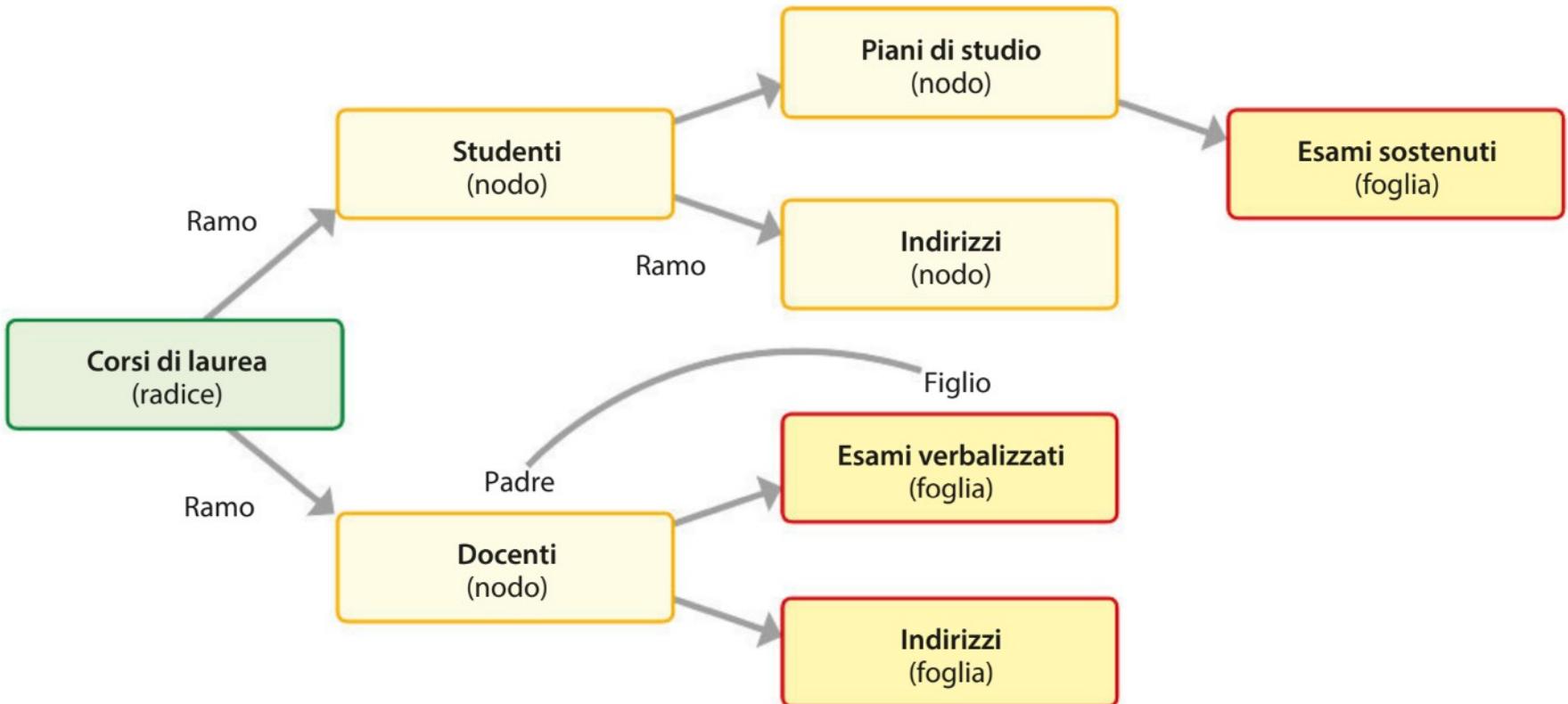
Nella teoria delle basi di dati, i modelli si distinguono in:

- **Modelli concettuali**, che permettono di rappresentare i concetti (dati) in modo indipendente da ogni sistema, cercando di descrivere i concetti del mondo reale. Come tali, mettono in evidenza i concetti presenti in una base di dati, piuttosto che la struttura con cui tali concetti possono essere rappresentati nella memoria del computer. Sono modelli utilizzati nella fase preliminare di progettazione e uno dei più noti è il modello **ER** (*Entity-Relationship*) che vedremo in seguito.
- **Modelli logici**, che consentono una specifica rappresentazione dei dati. Sono detti logici per sottolineare il fatto che le strutture utilizzate da questi modelli, pur essendo astratte, riflettono una particolare organizzazione (tabelle, alberi, grafi, oggetti e così via). Descrivono i dati in modo intermedio tra ciò che l'utente finale può vedere e il livello dell'implementazione. I principali modelli logici sono descritti di seguito.

Tipi di modelli logici

Modello gerarchico

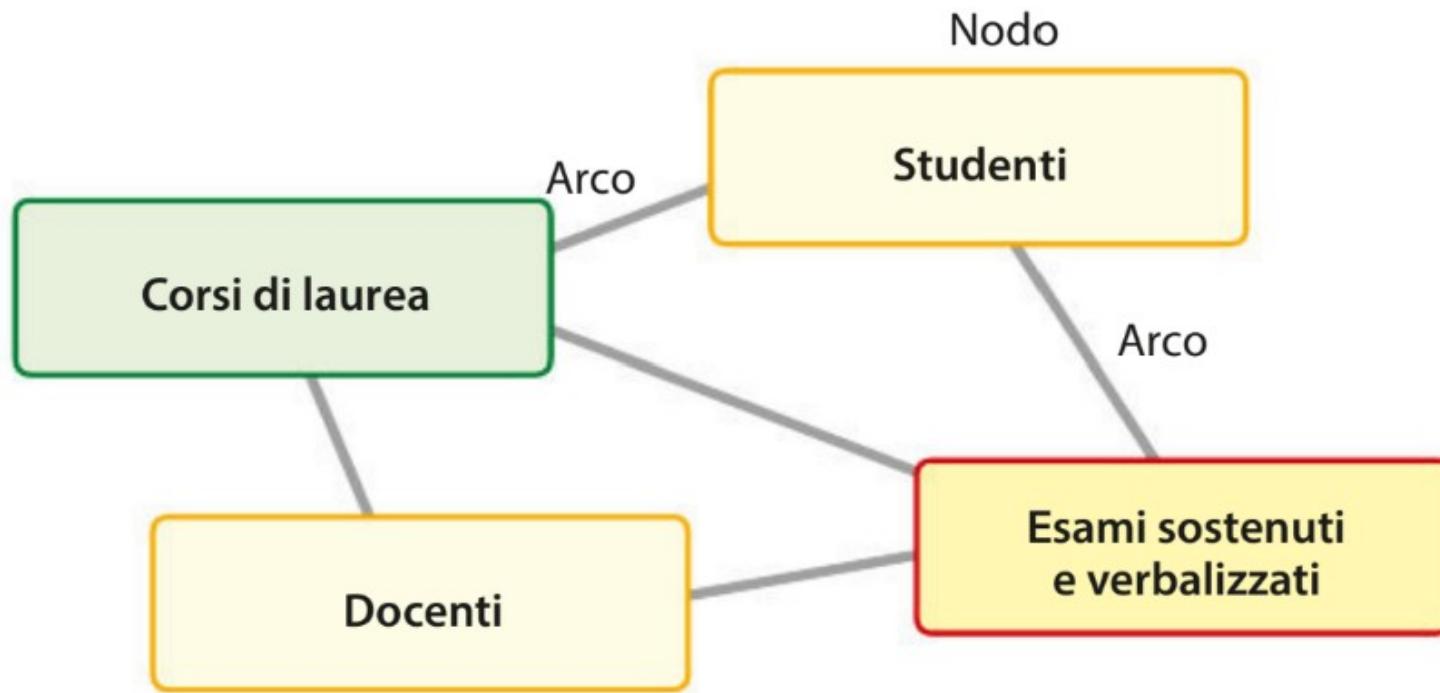
Questo modello ha caratterizzato i primi DBMS sviluppati verso la metà degli anni Sessanta. I dati sono organizzati in record connessi tra loro secondo strutture ad albero. Ogni record del database deve avere un unico padre. Possono esserci più record, su alberi diversi, che rappresentino la stessa informazione, anche se ciò può creare vari problemi come la ridondanza dei dati e continui controlli di consistenza (come ad esempio le informazioni sugli esami sostenuti e gli esami verbalizzati e l'impossibilità di inserire dati che non abbiano dei legami nel livello superiore, con l'esclusione della radice, come i dati sul personale tecnico-amministrativo). La manutenzione di un database gerarchico è piuttosto complessa poiché, ad esempio, la cancellazione di un record comporta l'eliminazione di tutti i record dipendenti da esso e l'aggiornamento di un dato richiede la modifica di altri record per assicurare la consistenza del database. Il modello gerarchico comporta la dipendenza dei programmi dalle strutture, quindi non possiamo modificare le strutture senza modificare i programmi.



Modello reticolare

È detto anche modello a rete o *modello Codasyl*, è stato ideato nel 1973 e perfezionato nel 1978; si basa sulle strutture dati a reticolo. Deriva da quello gerarchico, ma lo generalizza superando la rigidità della struttura ad albero: i record, infatti, sono legati tra loro con strutture ad anello che permettono all'utente finale di accedere ai dati in maniera più semplice, senza i vincoli rigidi della struttura gerarchica. Ogni nodo può essere un punto di partenza per raggiungere un campo. A differenza di quanto avviene nella struttura gerarchica, nel modello reticolare un record può avere uno o più record padre, evitando quindi i problemi relativi alla ridondanza dei dati. Non si escludono tuttavia problemi relativi alla dipendenza dei programmi dalle strutture e soprattutto alla complessità delle strutture stesse che aumentano in proporzione alla crescita dei dati. La gestione delle informazioni nei database costruiti secondo il modello reticolare è più complessa, in quanto deve essere utilizzata una struttura a grafo: risulta perciò più difficile l'implementazione e la costruzione del software applicativo. Un database costruito secondo il modello reticolare può produrre prestazioni più elevate, perché è più vicino alla struttura fisica delle memorie di massa, a fronte però di un notevole impegno di programmazione.

Per modificare un database con questa struttura, bisogna cancellarlo e ricrearlo.

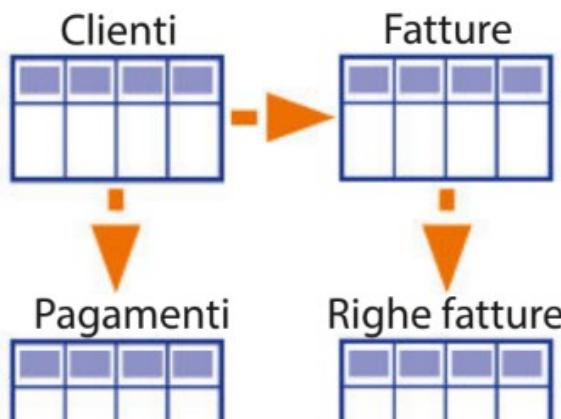


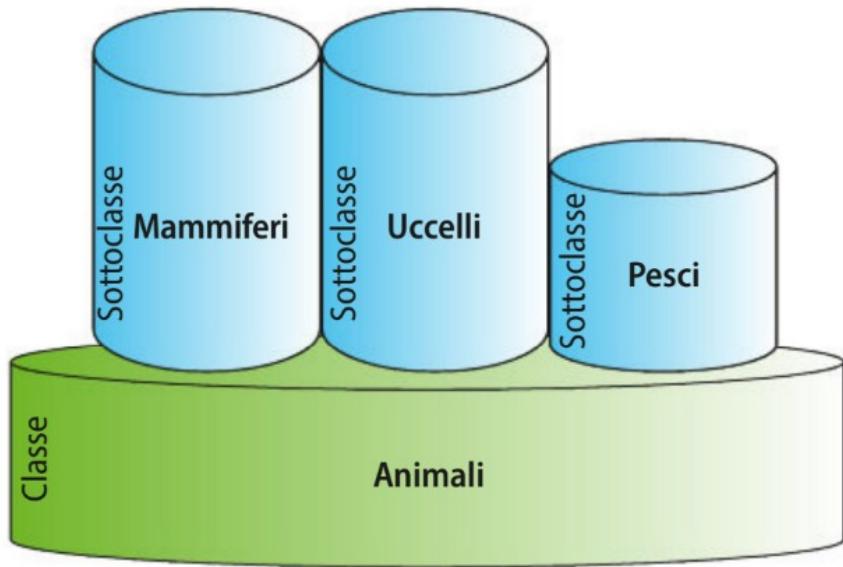
Modello relazionale

Questo modello, inventato da Codd nel 1970, è basato sul concetto di insiemi di record (tabelle) e le relazioni tra le informazioni derivano dalla corrispondenza di alcuni campi di record appartenenti a tabelle diverse. Il modello relazionale sarà trattato approfonditamente nelle prossime unità di apprendimento.

Le operazioni sui database gerarchici e reticolari sono macchinose: è necessario specificare i percorsi per ritrovare i singoli record coinvolti nell'elaborazione. Si dice che

l'approccio utilizzato nell'elaborazione dei dati è di tipo **procedurale**, ossia occorre indicare al computer come ritrovare i dati. I sistemi basati sul modello relazionale, invece, ammettono un approccio di tipo **dichiarativo** nell'elaborazione delle informazioni: ciò significa che è necessario specificare che cosa trovare, poiché del come (cioè del reperimento dei percorsi) si occupa il sistema.





Modello a oggetti

Questo modello, moderno e promettente, nasce come evoluzione del modello relazionale ed estende alle basi di dati il concetto di programmazione a oggetti. Lo sviluppatore del database gestisce tutti gli aspetti del database stesso, incluse le operazioni che manipolano i dati dall'interno del software di programmazione del database object oriented. Non esiste più una separazione netta tra il software di database e il software di programmazione dell'applicazione. Questo modello è stato introdotto per superare alcune caratteristiche dei database relazionali che si mostravano inadatte per alcune sfere applicative, quali, ad esempio, le applicazioni multimediali.

8 La progettazione di una base di dati

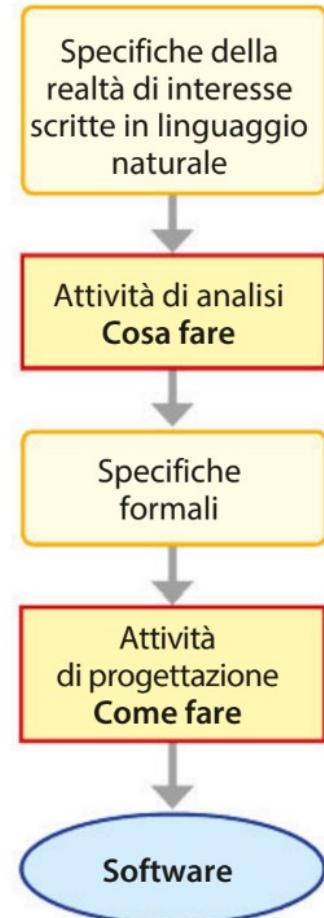
La **metodologia di progettazione** di una base di dati è un insieme di attività tra loro collegate, prodotti intermedi e finali di tali attività, criteri di verifica di qualità di tali fasi e prodotti, volti a realizzare una base di dati a partire da un insieme di specifiche che formalizzano le esigenze dell'utente.

Progettare una base di dati significa progettare la **struttura dei dati e le applicazioni**. La progettazione della struttura dei dati è l'attività fondamentale, in quanto richiede di specificare un modello della realtà di interesse (**universo del discorso**) quanto più possibile fedele.

Come si può osservare nella figura a lato, le specifiche della realtà di interesse vengono trasformate in **specifiche formali** dopo una prima fase di analisi. In questo caso “formali” si associa ad **astratte**, ovvero si descrivono in modo formale e astratto le tipologie di dati e le funzioni che opereranno su di essi.

Astratto, in questo caso, significa indipendente dalla tecnologia e dal particolare linguaggio di programmazione che verrà utilizzato per generare i programmi.

Si dice anche che la fase di analisi è volta a determinare **cosa** il programma deve fare (indipendentemente dai linguaggi che verranno utilizzati) e che la fase di progetto è volta a determinare **come** il programma dovrà fare quanto stabilito, cioè in che modo il documento di specifiche formali viene trasformato in programma.



PROGETTAZIONE

Progettazione concettuale

Progettazione logica

Progettazione fisica

- Il suo scopo è costruire e definire una rappresentazione corretta e completa della realtà di interesse.
- L'input di questa fase è il documento delle specifiche formali.
- L'output è uno schema concettuale, cioè una rappresentazione astratta e il più possibile formale della realtà; un esempio di output, riferendoci solo ai dati (e non alle funzioni), è il diagramma ER che vedremo nella prossima unità.

- Il suo scopo è quello di trasformare questa rappresentazione ancora astratta in uno schema logico, ovvero in una rappresentazione efficiente rispetto alle strutture del sistema di gestione che si intende utilizzare; un esempio è una descrizione relazionale tramite tabelle del diagramma ER.
- L'input di questa fase è il diagramma ER della fase di progettazione concettuale.
- L'output di questa fase è uno schema logico riassumibile con relazioni rappresentate tramite tabelle.

- Il suo scopo è quello di implementare lo schema logico definendo tutti gli aspetti fisici di memorizzazione e rappresentazione in memoria di massa.
- L'input di questa fase sono le tabelle della fase di progettazione logica.
- L'output di questa fase è l'implementazione in memoria di massa di tali tabelle.

