

Sistemi Informativi e Basi di Dati

PROF. DIOMAIUTA CRESCENZO

Organizzazione del corso

Orario lezioni:

- Lunedì dalle 16:00 alle 18:00
- Mercoledì dalle 11:00 alle 13:00

Ricevimento studenti

- Supporto via e-mail
- Da remoto
- Prima o dopo il corso

Modalità di esame

- Scritto e orale
 - oppure
- Progetto e orale

Organizzazione del corso

E-mail

- crescenzo.diomaiuta@unicampania.it

Libro di testo

BASI DI DATI 5/ED CON CONNECT

5ed

8838694451 · 9788838694455

di Paolo Atzeni, Stefano Ceri, Piero Frernali, Stefano Paraboschi, Riccardo Torlone



Informazione e Organizzazioni

- Nello svolgimento di attività è essenziale disporre di informazioni e di gestirle in maniera efficace;

- Conoscere significa poter competere a livello economico

- Conoscere significa essere in grado di gestire l'informazione

Sistema Informativo e Sistema Informatico

- Il Sistema Informativo organizza e gestisce le informazioni (ad esempio archivi delle banche, servizi anagrafici negli anni precedenti, ecc.)

- L'informazione rappresenta oggi uno dei beni più preziosi all'interno di una qualsiasi organizzazione

Sistema Informativo e Sistema Informatico

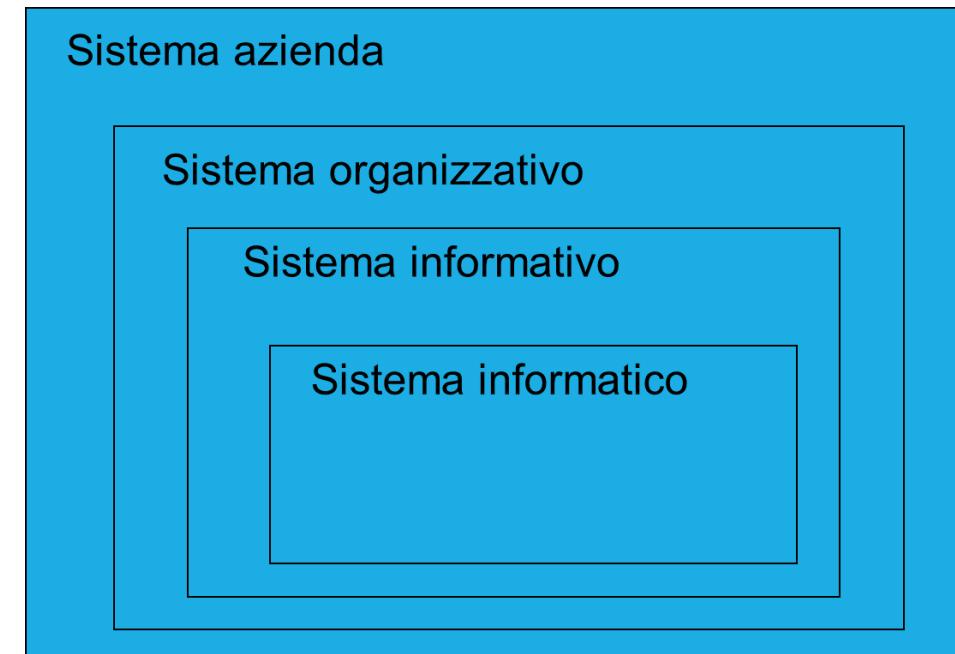
Definizione formale di Sistema Informativo:

- un sistema informativo è l'insieme delle componenti di un'organizzazione destinato alla
 - Raccolta, acquisizione
 - Archiviazione, conservazione
 - Recupero
 - Elaborazione, trasformazione, produzione
 - Condivisione, trasmissione , scambio, comunicazione

Sistema Informativo e Sistema Informatico

Definizione formale di Sistema Informatico è la tecnologia a supporto del sistema informativo:

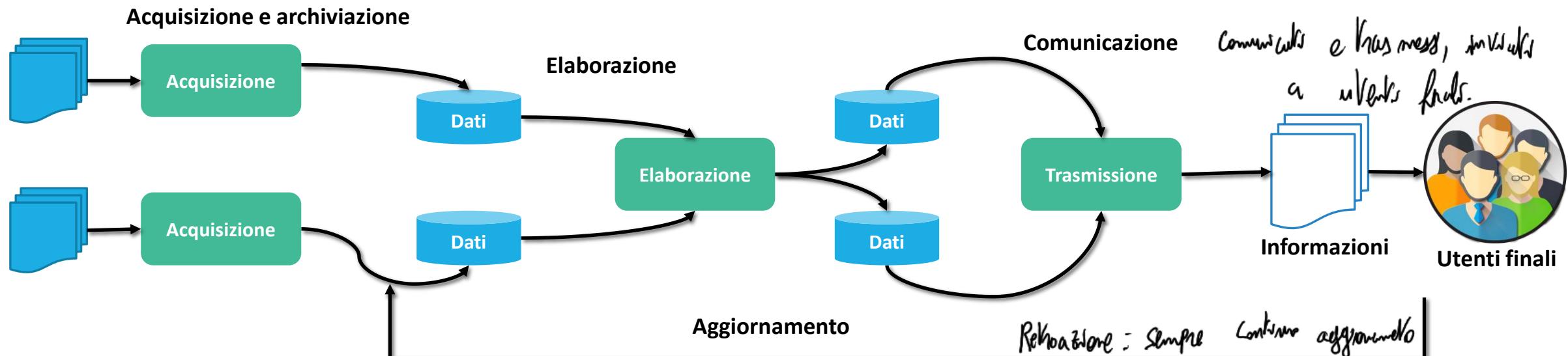
- Il sistema informatico rappresenta la porzione automatizzata del sistema informativo
 - Macchine hardware;
 - Programmi software;
 - Reti di comunicazione
 - Tutto quanto attiene alla ICT



Sistema Informativo e Sistema Informatico

Il sistema informativo deve permettere

- Acquisizione delle informazioni e il loro aggiornamento;
- Archiviazione tale da garantirne un reperimento efficace;
- Elaborazione per produrre aggregazioni e correlazioni utili per produrre statistiche e sintesi;
- Comunicazione alle persone che a vari livelli devono poterle utilizzare (reportistica).



Esempio

Comune dotato di sistema per il rilevamento del livello di inquinamento atmosferico.

Sistema **informativo**:

- Persone (dipendenti, sindaco, assessori)
- Procedure per la produzione e gestione dell' informazione relativa all'inquinamento

Sistema **informatico**:

- Gestione automatica – da parte di opportuni addetti- delle informazioni a livello di inquinamento atmosferico mediante: macchine hardware, programmi software, sensoristica e apparati di rete.

Alcuni esempi di Sistemi Informativi

- Sistemi informativi nella Pubblica Amministrazione
- Sistemi informativi ospedalieri
- Sistemi informativi per i trasporti
- Segreteria studenti
- Gestione supermercato
- ...

Alcuni esempi di Sistemi Informativi

I sistemi **informativi si basano sull'ICT per il trattamento delle informazioni**

- le **procedure sono automatizzate** e costituite da **programmi funzionanti su calcolatore e operanti in rete.**

In **alcuni contesti vi sono parti del sistema informativo aziendale che non sono “informatizzate”**

- pertanto le informazioni vengono prodotte senza l’ausilio delle tecnologie ICT.

Dati e informazioni: Definizione

Nelle attività umane, le informazioni vengono gestite in forme diverse:

- idee informali
- linguaggio naturale (scritto o parlato, formale o colloquiale, in varie lingue)
- disegni, grafici, schemi
- numeri e codici

e su vari supporti

- mente umana, carta, dispositivi elettronici

I dati devono essere opportunamente “interpretati” per dare luogo alle informazioni vere e proprie

Sia i dati grezzi che le regole per la loro interpretazione costituiscono pertanto il patrimonio informativo dell’azienda

Dati e informazioni: Definizione

Informazione: notizia, dato o elemento che consente di avere una conoscenza più o meno esatta di fatti, situazioni, modi di essere;

→ una serie di simboli

Dato: ciò che è immediatamente presente alla conoscenza, prima di ogni elaborazione, in informatica sono elementi di informazione costituiti da simboli che devono essere elaborati;

Devo fare in modo che dati messi insieme con le giuste interpretazioni mi dia una conoscenza.

Esempi

1. Esempio: le parole «Ferrari» e «8» annotate su un foglio di carta sono due dati che da soli non significano niente.



2. Esempio: la stringa (i dati) “Paolo Rossi 15486” non ha significato

1. se la ricevo in risposta alla domanda “Chi è il capo del personale e quale è il suo numero di ufficio?” allora i dati diventano informazione

Dati e informazioni

- Esempi di informazioni “elementari” ricavate interpretando dei dati appartenenti ad una certa organizzazione
- Si parla di informazione quando si comprende il dato
*→ valore assunto da attributo
in un certo istante*
- Un'informazione è definita da **(attributo, tipo e valore)**, una **tuple**
*↳ ripetizione di valori che può assumere
di tempo*

Attributo	Tipo	Valore
Cliente	Stringa	Paolo Rossi
Data di Nascita	Data	08/02/1964
Titolo libro	Stringa	Sistemi Web
Versione	Reale	1.0

Basi di Dati

- Una prima definizione di Base di Dati (DB, database)
 - insieme (collezione) di dati usati per rappresentare le informazioni di interesse di un certo contesto
 - dati tra di loro correlati e dotati di una opportuna descrizione
 - ↳ Fondamentale: aggregando dati posso ottenere conoscenza più approfondata
- Tipicamente, le basi di dati hanno caratteristiche più stabili rispetto alle procedure che operano su di essi.
- L'approccio convenzionale alla gestione di DB usa i files per memorizzare dati sulle memorie di massa.

Basi di Dati nella vita quotidiana

- Spesa in un supermercato
- Acquisto con carte di credito
- Prenotazione alberghiera
- Pagamento assicurazione
- Iscrizione ad un corso
- Richiedono
 - uso di “collezioni di dati”
 - memorizzate in modo persistente
 - relative ad una realtà di interesse
 - a servizio di una data organizzazione

devo sempre disporre di questi dati

Caratteristiche di una Base di Dati

- **Grandi**: possono avere anche dimensioni enormi, le più grandi hanno dimensioni dell'ordine di centinaia di TB.
 - il limite deve essere solo quello fisico dei dispositivi
 - esempi di dimensioni molto grandi
 - 500 Gigabyte (dati transazionali)
 - 10 Terabyte (dati decisionali)
 - 500 Terabyte (dati scientifici)
 - 100 miliardi di record
- **Condivise**: diverse applicazioni utenti devono poter accedere, secondo opportune modalità, a dati comuni.
- **Persistenti**: hanno un tempo di vita che non è limitato a quelle delle singole esecuzioni dei programmi che le utilizzano.

Problemi di una Base di Dati

- **Ridondanza:** anche con base dati perfette, posso comunque incorrere in ridondanze:
 - **informazioni ripetute** aumentano dimensione, ma non sono utile di info.

- **Rischio di incoerenza:**
 - le **versioni possono non coincidere**
 - Occhio a quanto modifichi la base dati
 - Manipolazione può portare ad anomalie

Base di Dati

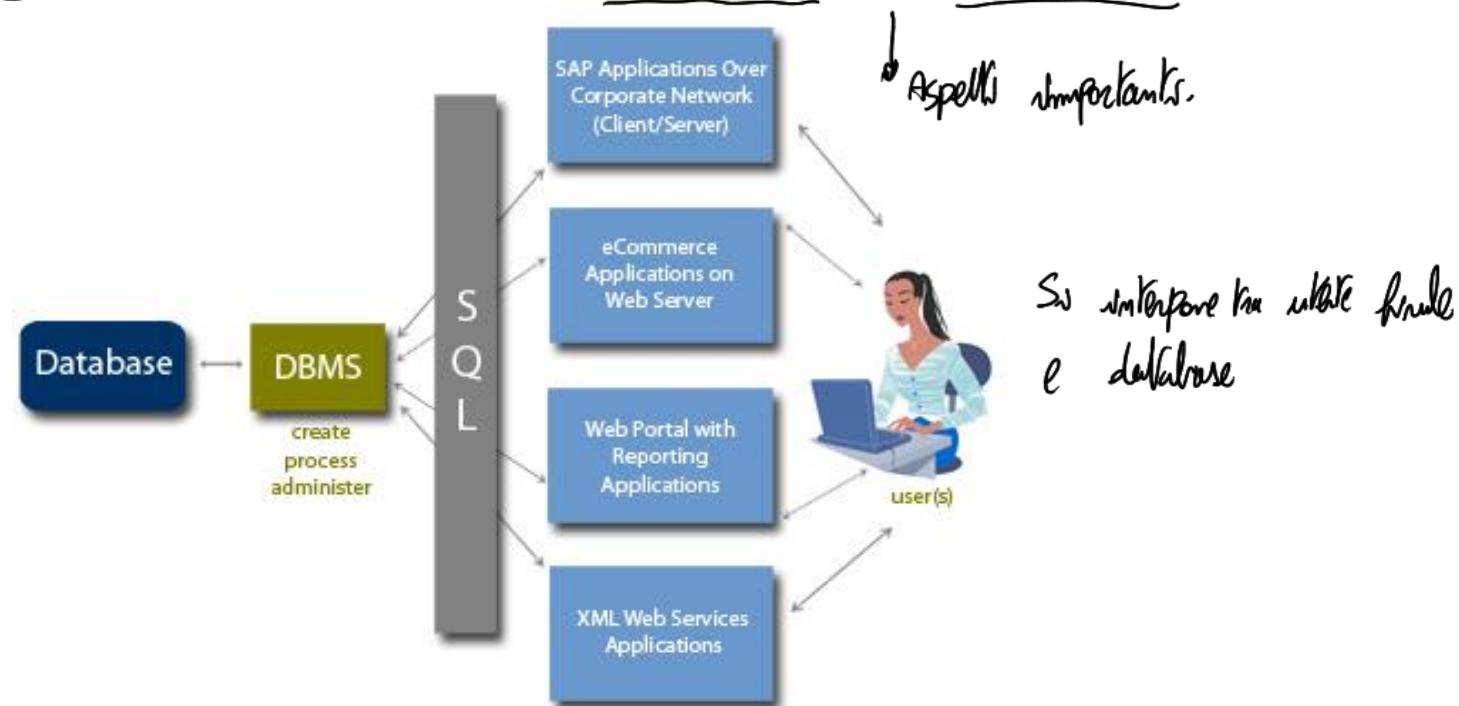
- La base di dati deve avere una propria descrizione
 - Mediante un catalogo o dizionario contenente un insieme di dati detti metadati che servono a descrivere i dati stessi.
Dato per descrivere dato

- Deve poter essere gestita
 - Grazie a una piattaforma software detta DBMS (Data Base Management System) che gestisce in maniera integrata tutti i dati in modo efficiente ed efficace.
Oracle fa da capo per il DBMS

↗ Skrumpfè SW che gestisce la base dati. L'è ottimizzata perché

DBMS

È un sistema di gestione di basi di dati (Data Base Management System – DBMS) è un sistema software in grado di gestire collezioni di dati che siano grandi, condivise e persistenti, assicurando la loro affidabilità e privatezza



DBMS

↳ few offre meccanismi: fai un dump del database x- es.

□ **Affidabilità:** ovvero la capacità del sistema di conservare sostanzialmente intatto il contenuto delle basi di dati.

- resistenza a malfunzionamenti hardware e software

□ **Privatezza:** ogni utente viene riconosciuto in base ad un nome utente (username) e può interagire con il DBMS. Viene abilitato solo a svolgere determinate azioni sui dati, attraverso meccanismi di autorizzazione => gestisce privilegi di accesso, per esempio

- l'utente A è autorizzato a leggere tutti i dati e a modificare X
- l'utente B è autorizzato a leggere dati X e a modificare Y

↳ Sûrlement c'est un DBA, che gère ce process.

DBMS

Come viene salvata info?

→ Come devono essere organizzati i dati?

- **Efficienza:** Cercano di utilizzare al meglio le risorse di spazio di memoria (principale e secondaria) e tempo (di esecuzione e di risposta)
▪ Buffer → Una memoria volatile, che ottimizza speed → quella da disco
 - L'efficienza è anche il risultato della qualità delle applicazioni
- **Efficacia:** Cercano di rendere produttive le attività dei loro utilizzatori, offrendo funzionalità articolate, potenti e flessibili

Compiti di un DBMS

Insieme di programmi che permette di:

- Definire la base di dati *DDL = data def. language*
 - Specificare tipi, strutture e vincoli sui dati
- Manipolare la base di dati *DML: data manipulation language*
 - Inserire, cancellare, aggiornare e recuperare i dati
- Controllare la base di dati *DCL = data control language*
 - Controllare l'accesso ai dati garantendo condivisione, protezione da guasti, da accessi indesiderati

↳ gestione permessi su utenti

Compiti di un DBMS

Il DBMS funge da interfaccia tra programmi applicativi e utenti e la base di dati

- supportando le operazioni di interrogazione, modifica e condivisione
- servendosi del catalogo per la comprensione dei dati

DBMS

Ogni applicazione gestisce propri file che consentono di memorizzare e cercare dati, con meccanismi «rozzi» per l'accesso e la condivisione (i dati sono logicamente privati)

Dati di interesse per più applicazioni devono essere replicati (ridondanza e incoerenza).

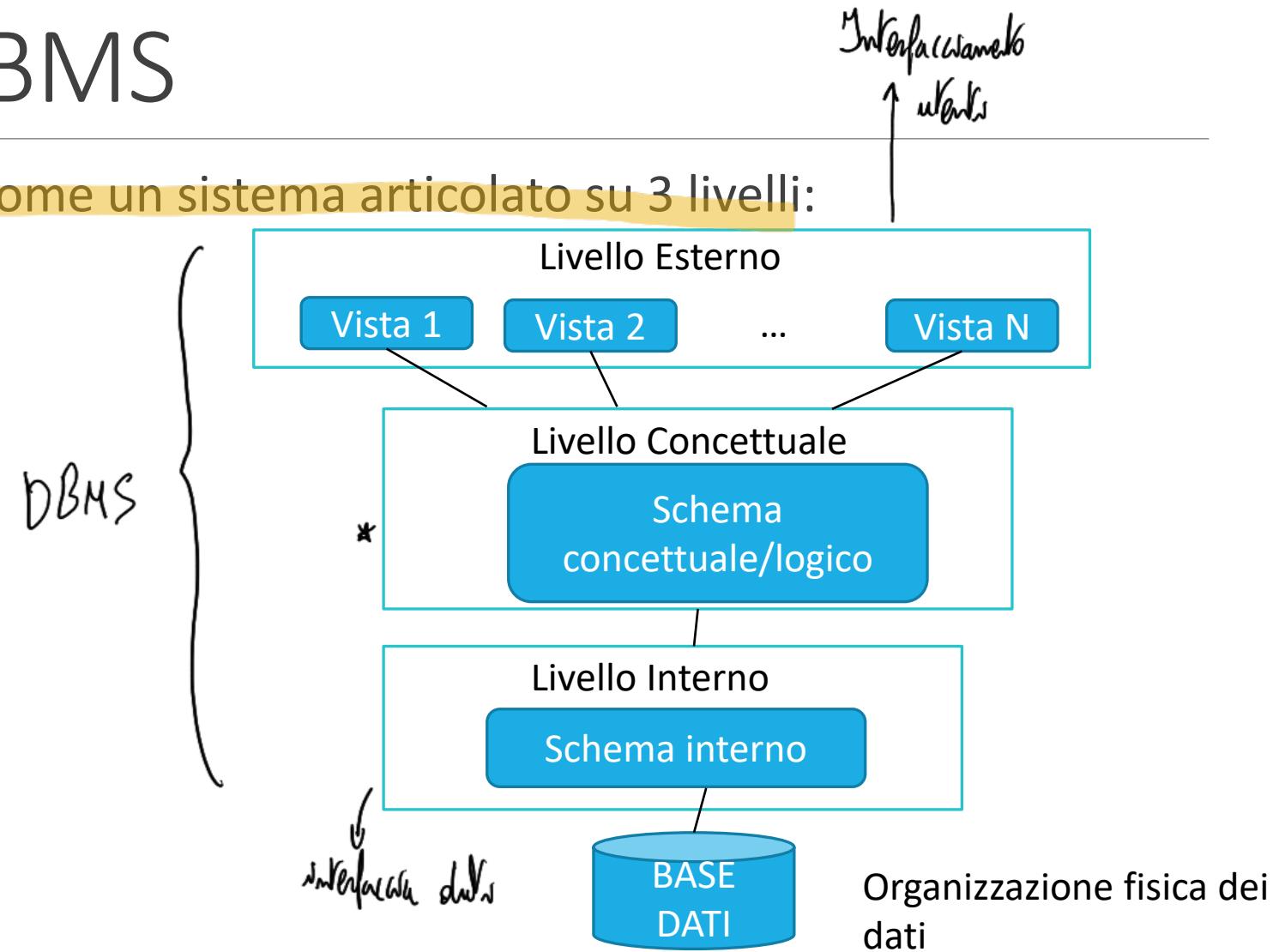
I DBMS sono nati per superare questi e molti altri problemi

Livelli di un DBMS

Il DBMS può essere visto come un sistema articolato su 3 livelli:

- Interno
- Logico
- Esterno

* Soprapposizione fra le 2 interfacce. Entrambe in gioco la programmazione



Livelli di un DBMS

↑ da base dati come viene memorizzata su disco?

- **Livello interno:** specifica come deve avvenire la memorizzazione fisica dei dati nel computer (interfaccia con il SO, file, record con puntatori, ordinati in un certo modo)

Come il DBMS vede i dati nei file

→ Posso avere diverse viste: posso vedere porzioni diverse di una base dati

- **Livello esterno:** personalizza l'accesso alle informazioni relativamente ad un singolo utente o ad un gruppo di utenti o ad una specifica applicazione

Come l'utente vede le informazioni

- **Livello logico:** descrive il modo in cui vengono organizzate le informazioni e come queste ultime sono correlate tra loro (ad esempio, la struttura della tabella)

Livello di mediazione tra utente e DBMS

↓
Rappresenta modo in cui organizzo le informazioni
Può essere influenzata dalle viste che ci sono, ma non dovrei fare questa cosa

Esempio

Matricola	Nome	Cognome	Indirizzo	CodFac
2340	Nino	Verdi	Milano	Ing
2360	Enrico	Bianchi	Venezia	Eco
2361	Francesco	Ceronte	Palermo	Eco
2362	Paolo	Merli	Torino	Ing

Base dati: insieme di tabella: la tabella facoltà ha info

Studenti

Facoltà

CodFac	NomeFacoltà	Indirizzo
Eco	Economia	Via De Gasperi
Ing	Ingegneria	Piazza Martini
Giu	Giurisprudenza	Via Carducci

Esempio

Matricola	Nome	Cognome	Indirizzo
2340	Nino	Verdi	Milano
2362	Paolo	Merli	Torino

StudentiIngegneria



Viste logiche

Per la selezione di Ingegneria

StudentiEconomia

Matricola	Nome	Cognome	Indirizzo
2360	Enrico	Bianchi	Venezia
2361	Francesco	Ceronte	Palermo

Per la selezione di Economia

Esempio

In riferimento all'esempio precedente possiamo dire che:

- Il livello esterno è composto dalle viste logiche
- Il livello logico è costituito dalla coppia di tabelle «Studenti e Facoltà»
- Il livello interno descrive come sono state realizzate le due tabelle «Studenti e facoltà» e dipende dal DMBS scelto

Al di sotto del livello interno ci sono i bit sul disco e sono di competenza del SO

Matricola	Nome	Cognome	Indirizzo	CodFac	Studenti	Facoltà
2340	Nino	Verdi	Milano	Ing		
2360	Enrico	Bianchi	Venezia	Eco		
2361	Francesco	Ceronte	Palermo	Eco		
2362	Paolo	Merli	Torino	Ing		

CodFac	NomeFacoltà	Indirizzo
Eco	Economia	Via De Gasperi
Ing	Ingegneria	Piazza Martini
Giu	Giurisprudenza	Via Carducci

Schemi

Punti di partenza



La descrizione di una base di dati è detta «schema della base di dati»

Si articola in:

Come le vado ho 3 schemi

- **schemi esterni**, che corrispondono alle differenti viste dei dati
- **schema logico**, che descrive le informazioni presenti nella base di dati e le loro proprietà
- **schema interno o fisico**, che fornisce una completa descrizione delle strutture fisiche di memorizzazione dei dati.

Schemi e istanze

Nei DB esistono due componenti:

schema della base di dati: caratteristiche dei dati, invariante nel tempo, che ne descrive la struttura (**aspetto intensionale**): dà indicazione del significato che assumono

- es.: le intestazioni delle tabelle

↳ Obbligatorie, a diff. delle istanze

un club db. Ha dà una semantica.

istanza o stato della base di dati: i valori attuali, che possono cambiare anche molto rapidamente (**aspetto estensionale**)

- es.: il "corpo" di ciascuna tabella

Foto della base
della db un certo
momento del tempo

Matricola	Nome	Cognome	Indirizzo
2340	Nino	Verdi	Milano
2362	Paolo	Merli	Torino

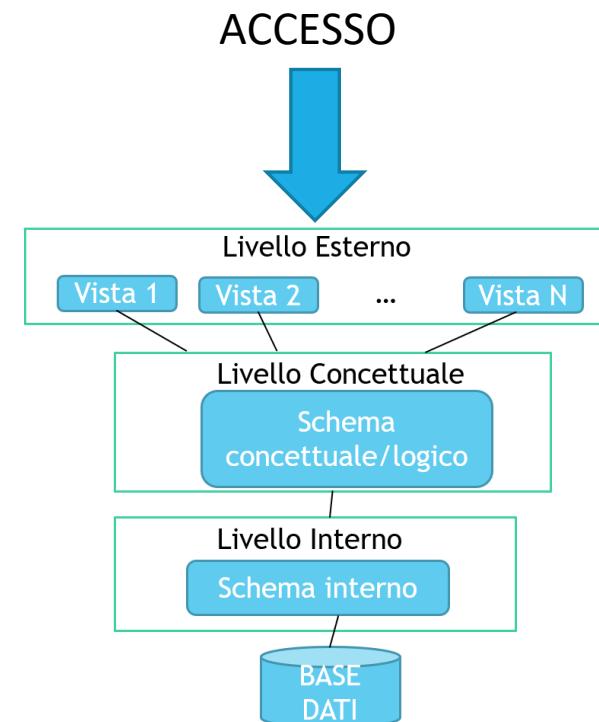
Schema

Istanza

→ Ciascuno dei livelli è indipendente dall'altro

Indipendenza dei dati nei DBMS

- Indipendenza dei dati dalle applicazioni
 - Le applicazioni sono indipendenti dal modo in cui i dati vengono organizzati. ⇒ A livello concettuale non mi interessa come i dati sono salvati
- Indipendenza logica dei dati:
 - È la capacità dello schema esterno di non essere influenzato dai cambiamenti apportati allo schema logico: se cambio cose a livello concettuale non debbo essere coinvolte le viste.
(Vero ma non troppo)
- Indipendenza fisica dei dati:
 - È la capacità dello schema logico di non essere influenzato dai cambiamenti apportati allo schema interno dei dati
 - Es: una tabella è utilizzata nello stesso modo qualunque sia la sua realizzazione fisica (che può anche cambiare nel tempo)



Modelli dei dati

- Un modello dei dati è un insieme di concetti utilizzati per organizzare i dati di interesse e descriverne la struttura in modo che essa sia comprensibile ad un elaboratore.
- Ogni modello fornisce meccanismi di strutturazione (o costruttori di tipo)
- Esistono diversi modelli disponibili su differenti DBMS esistenti, principalmente:
 - modelli logici
 - modelli concettuali

Modelli logici

- Adottati nei DBMS esistenti per l'organizzazione dei dati
 - utilizzati dai programmi
 - indipendenti dalle strutture fisiche
- Esempi: relazionale, reticolare, gerarchico, a oggetti, basato su XML
 - ↓
usiamo questo

Modelli concettuali

→ Rappresenta schema che facciamo per progettare base dati
La parte finale è modello logico. Ma serve modello concettuale per funz.

- Permettono di rappresentare i dati in modo indipendente da ogni sistema
 - cercano di descrivere i concetti del mondo reale
 - sono utilizzati nelle fasi preliminari di progettazione
- Il più diffuso è il modello **Entity-Relationship**

Modelli dei dati - Storia *(concrete)*

- Modello Gerarchico
 - Anni '60: dati organizzati in alberi
- Modello Reticolare
 - Anni '70: dati organizzati in grafi
- Modello Relazionale
 - Definito all'inizio degli anni '70, prime implementazioni fine anni 70. Struttura tabellare
 - Attualmente è il più diffuso ed il più consolidato.
 - Permette di definire tipi attraverso il concetto di relazione, organizzando i dati in record di lunghezza fissa.
 - Una relazione è tipicamente rappresentata attraverso una tabella.
- Modelli ad Oggetti
 - Object oriented: dati organizzati in “oggetti”
 - Object relational: tabella generalizzata

Modello gerarchico : Non ha vlr e proprie Valelle

Basato sull'uso di strutture ad albero (gerarchie)

Il modello gerarchico è particolarmente adatto per rappresentare situazioni nelle quali è possibile fornire all'insieme dei dati una struttura ad albero, nel quale i nodi rappresentano le entità e gli archi rappresentano le associazioni

I legami sono possibili grazie a puntatori a record

Si punte da record radice e si scende piano piano. Nelle foglie
CONTI hanno vlr e

Record radice

8000	12345
7000	500
4500	3500
5100	58500

CLIENTI

Bianchi	Mario	Padova	24127
Neri	Franco	Napoli	80120
Gialli	Dante	Bari	70122
Verdi	Marco	Roma	00128
Moro	Colombo	Milano	20143

Modello reticolare

Legami fatti tramite puntatori

Un database reticolare consiste di due insiemi di dati, un insieme di record e un insieme di legami. Nel modello le entità rappresentano i nodi e le associazioni rappresentano gli archi di uno schema a grafo orientato

I legami sono possibili grazie a puntatori a record

CLIENTI

Bianchi	Mario	Padova	24127
Neri	Franco	Napoli	80120
Gialli	Dante	Bari	70122
Verdi	Marco	Roma	00128
Moro	Colombo	Milano	20143

CONTI

8000	12345
7000	500
4500	3500
5100	58500

MOVIMENTI

23/12/2010	Vers	2500,00
25/04/2010	Prel	1250,00
03/04/2011	Vers	500,00
23/12/2010	Vers	736,00
23/12/2010	Prel	125,00

Le puntatori associati a record.

No concetto di tabella

Modello relazionale

Non ho più regole basate su puntatori a record ma un altro modo

Basato sulla definizione di relazione: essenzialmente una particolare tabella

Corrispondenze basate su valori che eliminano i problemi dati dalle posizioni dei record dei file

Matricola	Nome	Cognome	Data di nascita
2340	Nino	Verdi	05/12/2001
2360	Enrico	Bianchi	03/11/2003
2361	Francesco	Ceronte	01/02/2002

Studenti

Studente	Voto	Corso
3456	28	04
3456	24	02
9283	26	01

Esami

Codice	Titolo	Docente
01	Analisi	Mario
02	Chimica	Bruni
04	Fisica	Verdi

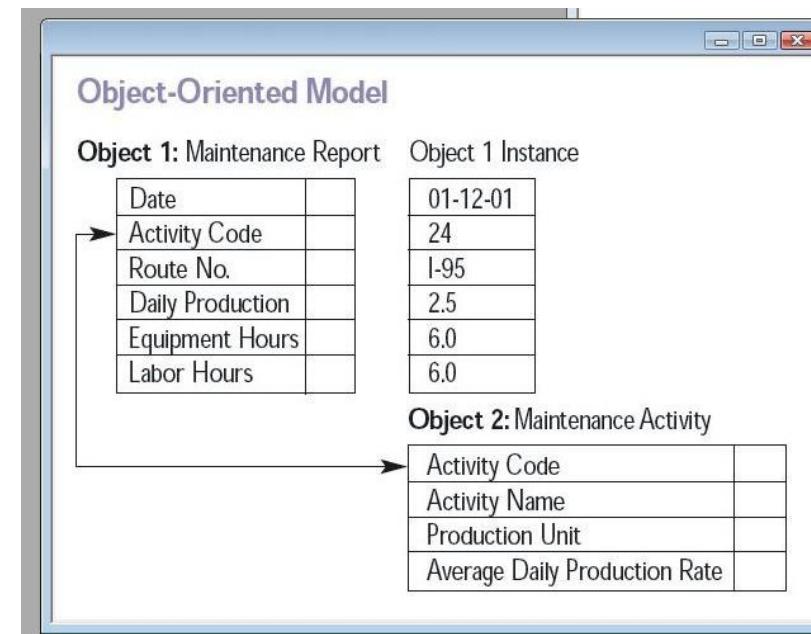
Corsi

Modello ad oggetti

Un database ad oggetti memorizza le informazioni sotto forma di oggetti.

La modalità di rappresentazione dei dati è la stessa di quella usata nella programmazione ad oggetti.

Poco performante su grosse quantità di dati



Vanno bene per quei valori & obietti, ma se devo memorizzare, x es., tabelle diverse? Ma sono file che vanno memorizzati (es. foto losche). Allora mi serve un database.

Database NoSQL (Not only SQL)

Assenza di una struttura rigida

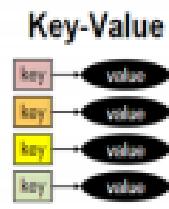
noSQL (no sequel).

Velocità di memorizzazione per grandi quantità di dati (Big Data)

Complessità nella lettura

Grande scalabilità orizzontale

Modello esistente da tempo ma reso attuale da Internet e dagli attuali server performanti



membase



Cassandra

mongoDB



Principali funzioni dei DBMS

L'implementazione del modello logico sul sistema di elaborazione implica:

- La definizione dei dati e delle strutture dati derivate dallo schema logico (tipicamente le tabelle del modello relazionale) e della relativa documentazione;
- La definizione dei sottoschemi (viste logiche), che consentono agli utenti di accedere ai soli dati che servono; una vista è una tabella virtuale, mentre le tabelle del database sono indicate come tabelle primarie.

Principali funzioni dei DBMS

Manipolazioni e interrogazioni

- Inserimento, modifica e cancellazione dei dati nel database.
- Interfaccia tra i programmi degli utenti e la base di dati, utilizzando le funzionalità del DBMS per migliorare l'organizzazione dei dati e le prestazioni dei programmi nelle operazioni di ritrovamento dei dati.
- Accesso ai dati attraverso interfacce grafiche o semplici comandi che facilitano l'utente non specialista.

Posso trattarvi su diversi punti di vista, ma ovviamente è anche importante proteggere i dati

Principali funzioni dei DBMS

Sicurezza e protezione dei dati

- Contro i danni causati da malfunzionamenti o da interventi dolosi.**
- Protezione da eventuali danneggiamenti con la possibilità di attivare procedure di recovery.**
- Autorizzazione degli utenti che accedono alla base di dati e protezione dagli accessi non autorizzati.**
- Controllo degli accessi simultanei da parte di più utenti.**

Principali funzioni dei DBMS

Controllo di integrità sui dati

- In relazione ai valori che possono assumere e alle interdipendenze tra dati di differenti tabelle.**
- Integrità definite dall'utente in conseguenza di politiche commerciali dell'impresa o di norme legislative e fiscali.**

Es: tabella che memorizza chiavi non può memorizzare un canale

Principali funzioni dei DBMS

Ottimizzazione delle prestazioni

- Ottimizzazione della memoria di massa occupata e dei tempi di accesso ai dati.
- Implementazione di strategie per ottimizzare le interrogazioni minimizzando i tempi di risposta o, più in generale, i costi dell'interrogazione

Es come un buffer che porta collegamenti a obietti più frequenti

Principali funzioni dei DBMS

Supporto alle transazioni

- Garanzia che tutte le operazioni di una transazione siano completate correttamente oppure che non ne sia eseguita alcuna.
- Una transazione consiste in un insieme di operazioni di interrogazione o di modifica del database che devono essere eseguite come se fossero un'unica operazione, come avviene nel caso di un'operazione di trasferimento di fondi tra conti correnti bancari. L'importo trasferito deve essere tolto da un conto e aggiunto a un altro: l'esecuzione della transazione implica il corretto completamento di entrambe le operazioni.
- Il supporto alle transazioni ne deve anche garantire il corretto completamento anche in caso di esecuzione concorrente.

Altre funzioni dei DBMS

- I DBMS dispongono spesso di componenti, report, che sono applicazioni specializzate per presentare i dati estratti da un database in un formato comodo per la lettura da parte di una specifica categoria di utenti.
- Un DBMS gestisce anche il dizionario dei dati (o catalogo del database). Il dizionario contiene i metadati, cioè i dati che descrivono i dati contenuti nel database. In genere un dizionario dei dati contiene informazioni su:
 - nomi delle tabelle e delle colonne
 - associazioni legame che esiste tra più tabell
 - viste logiche
 - vincoli di integrità
 - utenti e proprietari
 - autorizzazioni degli accessi

Proprietà di un DBMS

1. Controllo della ridondanza e consistenza

- Eliminazione dei duplicati (ridondanza).
- Riduzione del rischio di “inconsistenza”

2. Condivisione

- I dati di un’organizzazione sono condivisi da tutti gli utenti e da tutte le applicazioni di un’organizzazione.

3. Integrità dei dati

- Consistenza e alla validità dei valori contenuti nella DB gestendo le “modifiche” delle informazioni mediante opportune regole di vincolo conservate nel catalogo

Proprietà di un DBMS

4. Gestione efficiente delle operazioni

- Il DBMS dispone di funzionalità per l'esecuzione efficiente delle interrogazioni e degli aggiornamenti.

5. Gestione della concorrenza

- Il DBMS garantisce transazioni concorrenti sui dati senza interferenze reciproche

6. Affidabilità

- Il DBMS dispone di funzionalità per il ripristino della base dati anche in caso di guasti o malfunzionamenti durante l'esecuzione delle transazioni

Proprietà di un DBMS

7. Sicurezza

- Per sicurezza si intende prevenzione dall'accesso ai dati ad utenti non autorizzati. Il DBMS consente di definire politiche di accesso ai dati per utente o per profilo di utente.

Componenti di un DBMS

- Gestore degli **accessi**: sottocomponente che vede chi è che fa cosa: più specifiche se user fa operazione
 - Chi fa cosa
 - avvenuta con i log così cepriso
- Gestore delle **query**: effettua interrogazioni da user e programmi
 - esegue le richieste utenti
- Gestore della **memoria**: gestisce al meglio le memorie
 - strategie di accesso alle memorie di massa
- Gestore dei **file**: vede come base dati deve essere trasformata in file
 - Organizzazione fisica dei file di concerto con il file system
- Gestore della **concorrenza**: gestisce accessi simultanei a stesse porzioni
 - Accesso concorrente alle stesse informazioni
- Gestore dell'**integrità**: verifica che vincoli siano sempre rispettati: è critico. Garantisce assenza di incoerenze
 - Rispondenza ai vincoli che fissano i controlli di correttezza
- Gestore dell'**affidabilità**: parte del logging e recovery
 - Gestione dei log e delle procedure di ripristino

Utenti di un DBMS

- Progettisti e programmati di applicazioni, definiscono e realizzano i programmi che accedono al DB.
- Amministratore del DB: responsabile delle progettazione, del controllo e della amministrazione. In genere uno
- Utenti:
 - Utenti finali (o terminalisti): utilizzano programmi che realizzano attività standard definite
 - casuali: in grado di utilizzare DML che realizzano navigazioni non standard.
 - ↳ persona che usa abitualmente base dati (che in genere usa sempre la stessa tabella)
 - ↳ qualcuno che non conosce come è stata progettata la base dati.
 - ↳ Non è utente attuale o che accede alla stessa posizione

Vantaggi di un DBMS

- Permettono di vedere i dati come una risorsa comune a disposizione di tutti gli aventi diritto.
- Controllo centralizzato dei dati. Per forza di cose: se ho più user che ci accedono, deve essendo per forza
- Riduzione ridondanze e inconsistentezze, grazie alla condivisione.
↳ farà dei controlli
- Indipendenza dei dati
↓
è lo offre indipendenza a 3 livelli

Svantaggi di un DBMS

- Costo notevole (hw, sw, competenze) *(anche se oggi ho ruolo OS)*
- Solitamente non tutti i servizi offerti da un DBMS sono necessari
- Competenze diversificate per la gestione
- Sistema operativo, networking, web application, etc. *⇒ Limitazione del sistema operativo ecc.*
- Che rendono il ruolo del DBA affascinante

Linguaggi per DB: SQL oltre ...:

- Su un DBMS è possibile ovviamente specificare operazioni attraverso linguaggi:
 - DDL (Data Definition Language) utilizzati per la definizione di schemi. [Vedi slide 32] ↗ creazione
 - DML (Data Manipulation Language) utilizzati per interrogazioni ed aggiornamenti.
 - DCL (Data Control Language) utilizzati per definire utenti e privilegi.
- Disponibilità di vari linguaggi e interfacce:
 - linguaggi testuali interattivi (SQL) Nos usiamo lui
 - comandi (SQL) immersi in un linguaggio ospite (Pascal, Java, C ...)
 - comandi (SQL) immersi in un linguaggio ad hoc, con anche altre funzionalità (p.es. per grafici o stampe strutturate)
 - con interfacce amichevoli (senza linguaggio testuale)

Accesso ai Dati

- L'accesso ai dati può avvenire tramite:
 - Linguaggi testuali interattivi.
 - Comandi immersi in linguaggi tradizionali
 - Comandi simili a quelli interattivi immersi in linguaggi/ambienti di sviluppo ad hoc.
 - Interfacce amichevoli.

Transazioni

- Le **Transazioni sono** particolari programmi utente che operano sui **DBMS**.
- Una transazione è l'esecuzione di un **programma utente** in ambiente **DBMS** che costituisce sintatticamente e semanticamente un'unità atomica di modifiche "persistenti" fatte alla base di dati.
- Una **transazione**
 - o **termina con successo** nello stato finale previsto dal **programma (commit)**
 - o **riporta il sistema** nello **stato precedente** all'inizio della **transazione (abort)** per fallimento delle azioni richieste
- Sistemi di basi di dati basati su transazioni sono detti **On Line Transaction Processing**, o **OLTP**

Operazioni nelle transazioni

- Operazioni di interrogazione (SELECT) : garantiscono l'atomicità in quanto non modificano lo stato della base dei dati
- Operazioni di INSERT: per inserire un nuovo oggetto nella BD
- Operazioni di UPDATE: per modificare un oggetto preesistente nella base di dati.
- Operazioni di DELETE: per cancellare un oggetto all'interno della base di dati.

Proprietà ACID

- **Atomicità:** “tutto o niente”: o è eseguita nella sua interezza oppure non è eseguita per niente
- **Consistenza:** è una trasformazione di uno stato consistente della base di dati in un altro stato consistente
- **Isolamento:** le transazioni devono essere eseguite in modo indipendente l'una dalle altre
- **Durability (Persistenza):** gli effetti devono essere registrati in modo permanente

