

OPEN DATA E DBMS

INTRODUZIONI ALLE BASI DI DATI

MODELLO DEI DATI

SISTEMI INFORMATIVI E DBMS

CORSO DI LAUREA MAGISTRALE IN
MANAGEMENT E MONITORAGGIO DEL TURISMO SOSTENIBILE



PROF. ANDREA PINNA
AA 2020/2021

BASI DI DATI

Cos'è una base di dati?

Insieme organizzato di dati correlati, progettato, costruito e popolato con dati con lo scopo di supportare lo svolgimento di attività.

Rappresenta una realtà del “mondo reale”.

I dati sono tra loro collegati a livello logico, in base ad una semantica ben precisa.



BASI DI DATI: DBMS

Le basi di dati si possono gestire con il calcolatore. Per farlo si possono adottare due soluzioni:

- Usare un DataBase Management System (DBMS): un sistema dal funzionamento generale adattabile allo scopo specifico
- Usare programmi sviluppati appositamente per gestire i dati della specifica applicazione.



BASI DI DATI: DBMS

Generalmente, il sistema informativo di una azienda o di una qualunque organizzazione comprende una Base di Dati.

Tra gli elementi del sistema informativo si hanno:

Calcolatori

Sistemi di memorizzazione

Reti di trasmissione

Software applicativo

+ Il personale



BASI DI DATI: DBMS

Un DBMS consente di creare, popolare, manipolare e condividere una base di dati. L'uso di un DBMS garantisce che la base di dati goda delle seguenti proprietà:

- Privatezza (o riservatezza)
- Affidabilità (e robustezza)
- Efficienza
- Efficacia

Si utilizza per collezioni di dati **grandi, persistenti** e condivise e che possono essere aggiornate nel tempo (manutenibilità della base di dati)



BASI DI DATI: DBMS

Si parla di **grandi** dati quando questi sono di dimensioni molto maggiori della memoria del sistema (si intende la memoria RAM).

Tipicamente la memoria di un PC è 8GB. Una macchina da calcolo può avere oltre 36GB di memoria.

I dati si possono considerare grandi se superano di uno o due ordini di grandezza la dimensione memoria (ad esempio, dati di dimensione pari a 500 GB sono dati grandi)



BASI DI DATI: DBMS

Si parla di dati **persistenti** se i dati continuano ad esistere anche quando viene interrotta l'applicazione che li utilizza. Il dato ha una "vita indipendente" dal resto del sistema informativo.

Si parla di dati **condivisi** se è possibile utilizzare la stessa base di dati con diversi sistemi informativi. Ad esempio, un'organizzazione o un'impresa può avere diversi settori semi-indipendenti. Tramite una base di dati, tutti i settori possono accedere agli stessi dati, evitando di creare copie (ridondanza).



BASI DI DATI: DBMS

I DBMS permettono di evitare due problemi:

Ridondanza: informazioni registrate in più copie

Incoerenza: le versioni possono non coincidere

La condivisione dei dati può essere mantenuta solo se si implementano meccanismi di **autorizzazione** (chi può fare cosa) e di controllo della **concorrenza** (cosa fare se due utenti agiscono contemporaneamente sui dati)



BASI DI DATI: DBMS

I meccanismi di autorizzazione permettono di garantire la proprietà di **Privatezza** (o Riservatezza)

Ad esempio, si autorizza un certo gruppo di utenti a leggere una certa parte dei dati e a modificare un'altra parte dei dati. Uno dei meccanismi prevede l'uso delle "viste": sottoinsiemi della basi di dati che soddisfa le esigenze di un gruppo di utenti.

Oppure si autorizza un utente (ad esempio l'amministratore) a leggere e modificare tutto (super utente).



BASI DI DATI: DBMS

Affidabilità (e robustezza)

Le basi di dati devono resistere ad eventuali malfunzionamenti hardware e software e devono resistere a comportamenti inattesi degli utilizzatori o del sistema informativo.



BASI DI DATI: DBMS

Le operazioni di modifica della base di dati tramite DBMS possono essere eseguite tramite un meccanismo “transazionale”.

Una transazione è **insieme di operazioni** da effettuare atomicamente (ovvero in maniera indivisibile).

L'esecuzione di una transazione provoca effetti definitivi.

Le transazioni sono essenziali nella registrazione di operazioni relative a flussi monetari.



BASI DI DATI: DBMS

L'uso del DBMS consente di gestire la base di dati in modo da garantire **l'efficienza**.

Un DBMS gestisce le risorse della macchina per minimizzare lo spazio di memoria necessario e per ridurre il tempo di esecuzione delle operazioni.

L'efficienza complessiva del sistema informativo dipende sia dal DBMS che dalle applicazioni che lo utilizzano.



BASI DI DATI: DBMS

I DBMS consentono di garantire l'efficacia. I DBMS permettono di ottenere quello che si desidera dai dati evitandoci di dover implementare le modalità di elaborazione dei dati. Metodo utilizzato: separazione fra programmi e dati e astrazione dei dati. Questo si traduce nell'indipendenza tra i programmi e i dati e tra programmi e le operazioni sui dati (utilizzo di interfacce)

Efficacia: "Capacità di produrre pienamente l'effetto voluto, e l'ottenimento stesso dell'effetto".



BASI DI DATI: DBMS

Quando **non usare** un DBMS

- Applicazioni con vincoli rigidi di tempo reale (real-time).
- Applicazioni con alcune operazioni critiche che richiedono la scrittura ad hoc di un codice efficiente.
- Applicazioni che necessitano di tipi di manipolazione di dati non supportata dai DBMS.
- Analisi costi benefici con esito negativo.



MODELLO DEI DATI

I dati gestiti da un DBMS devono essere organizzati secondo un **modello**.

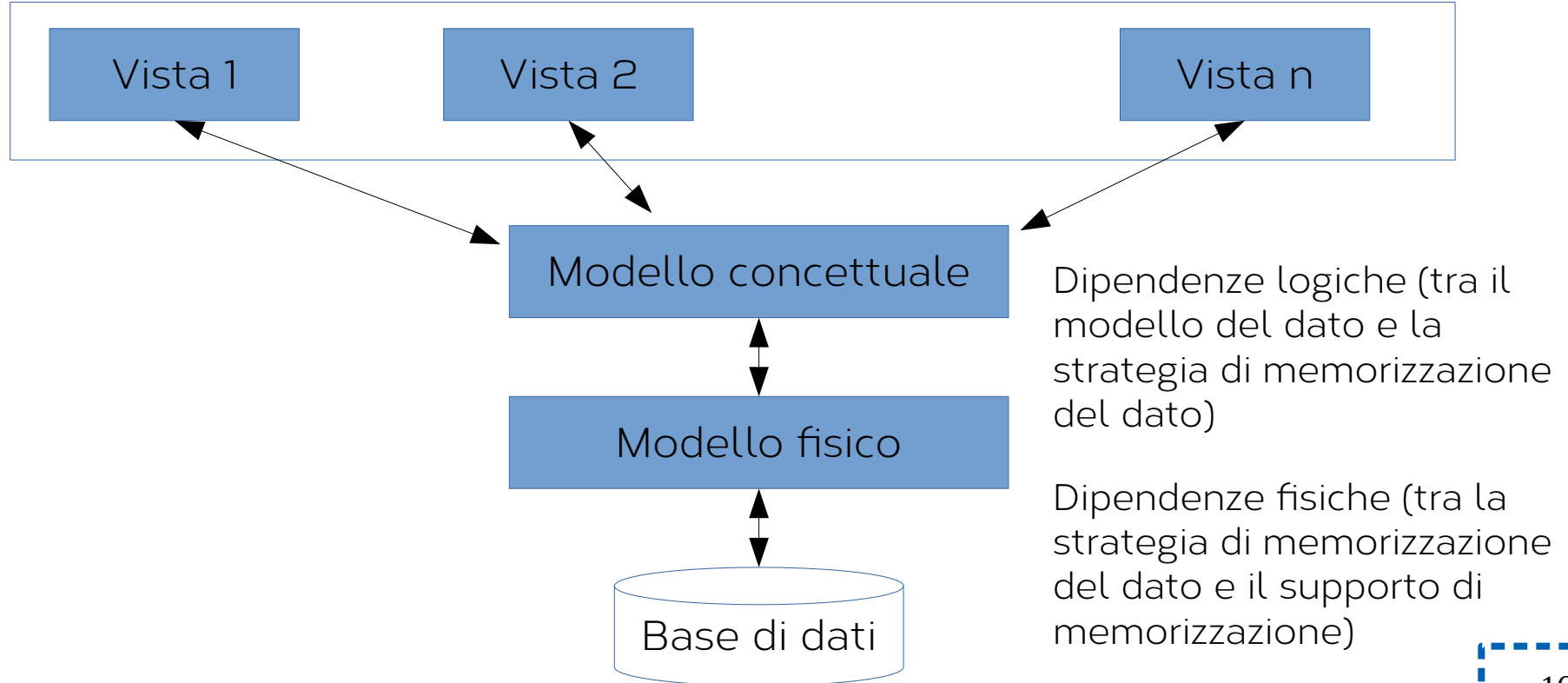
Il modello permette di astrarre (ovvero portare ad un livello più generale) ciò che i dati rappresentano.

Un modello è definito da un **insieme di costrutti** che permettono di strutturare i dati.



MODELLO DEI DATI: SCHEMA

Il modello dei dati è il modello concettuale di organizzazione dei dati.



MODELLO RELAZIONALE

Il **Modello Relazionale** è un modello basato sui valori.

Organizza i dati in strutture (tabelle) e consente di creare dei riferimenti tra i valori di diverse strutture. Ogni riga della tabella è chiamata ennupla (o tupla).



MODELLO RELAZIONALE

Il **modello relazionale** è fondato sui legami tra i dati che caratterizzano la base di dati. La relazione rappresenta una classe di fatti.

I riferimenti fra dati in relazioni diverse sono rappresentati per mezzo dei valori dei domini che compaiono nelle ennuple

Vedremo una metodologia di progetto che si basa sul modello Entity-Relationship.

Vedremo uno dei linguaggi per implementare e interrogare una base di dati relazionale: il linguaggio SQL



MODELLO RELAZIONALE

Il modello dei dati si basa sul concetto di relazione (una tabella) ed è basato sui valori. Mira all'indipendenza dalle strutture fisiche che possono cambiare dinamicamente

Si rappresenta solo ciò che è rilevante dal punto di vista dell'applicazione della base di dati.

L'utente finale utilizza gli stessi dati dei programmatori che realizzano le applicazioni. I dati sono portabili più facilmente da un sistema ad un altro (interoperabilità).



MODELLO RELAZIONALE

In una tabella che rappresenta una relazione:

- l'ordinamento tra le righe è irrilevante
- l'ordinamento tra le colonne è irrilevante

Una tabella rappresenta una relazione se:

- le righe sono diverse fra loro,
- le intestazioni delle colonne sono diverse tra loro,
- i valori di ogni colonna sono fra loro omogenei,

La struttura delle tabelle costituisce lo Schema.



MODELLO RELAZIONALE

Con il formalismo matematico si definisce:

Schema di relazione:

un nome R con un insieme di attributi A_1, \dots, A_n :

$$R(A_1, \dots, A_n)$$

Schema della base di dati:

insieme di schemi di relazione:

$$R = \{R_1(X_1), \dots, R_k(X_k)\}$$



MODELLO RELAZIONALE

Le ennuple e gli attributi secondo il formalismo matematico.

Una **ennupla** su un insieme di attributi X è una funzione che associa a ciascun attributo A in X un valore del dominio di A .

Con $t[A]$ viene denotato il valore della ennupla t sull'attributo A .



MODELLO RELAZIONALE

Le istanze: l'insieme del modello rappresentato dalle relazioni e dati rappresentati come ennuple (ovvero le tabelle con il loro contenuto)

- L'istanza di **relazione** su uno schema $R(X)$ è l'insieme r di ennuple su X .
- L'istanza di una **base di dati** su uno schema $R = \{R_1(X_1), \dots, R_n(X_n)\}$ è l'insieme di relazioni $r = \{r_1, \dots, r_n\}$ (con r_i relazione su R_i)



MODELLO E-R

Il **modello ER** (Entity-Relationship) è un modello concettuale ad alto livello che definisce e distingue due elementi base: le entità e le associazioni (relationship in inglese, tradotte spesso con “relazioni”, ma con un significato diverso rispetto al modello relazionale)

Il modello si può descrivere con i diagrammi ER (notazione grafica)

Il modello dei dati si ottiene partendo dalla **descrizione del mondo** che si intende rappresentare.



MODELLO E-R

Entità: sono elementi del mondo reale che esistono in modo indipendente.

Attributi: sono elementi che caratterizzano (o che descrivono) le entità. In una tabella corrispondono ai nomi degli elementi della ennupla.

Relationship (associazioni): quando un attributo di un tipo di entità fa riferimento a un altro tipo di entità.

Si rappresenta il riferimento come una “associazione” fra tipi di entità (e non come attributi)



MODELLO E-R

studenti

Matricola	Cognome	Nome	Data di nascita
6554	Rossi	Mario	05/12/1995
8765	Neri	Paolo	03/11/1995
9283	Verdi	Luisa	12/11/1996
3456	Rossi	Maria	01/02/1995

esami

Studente	Voto	Corso
3456	30	04
3456	24	02
9283	28	01
6554	26	01

corsi

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



MODELLO E-R

Associazioni sui singoli attributi

studenti

Matricola	Cognome	Nome	Data di nascita
6554	Rossi	Mario	05/12/1995
8765	Neri	Paolo	03/11/1995
9283	Verdi	Luisa	12/11/1996
3456	Rossi	Maria	01/02/1995

studenti lavoratori

Matricola
6554
3456



MODELLO E-R: VINCOLI

Il modello ER permette di definire gli attributi delle entità e delle associazioni.

Permette inoltre di definire i vincoli tra i valori per garantire l'integrità dell'informazione.



MODELLO E-R: VINCOLI

Esempio: una base di dati scorretta:

Esami	Studente	Voto	Lode	Corso
	276545	32		01
	276545	30	e lode	02
	787643	27	e lode	03
	739430	24		04

Studenti	Matricola	Cognome	Nome
	276545	Rossi	Mario
	787643	Neri	Piero
	787643	Bianchi	Luca



MODELLO E-R: VINCOLI

Un vincolo di integrità è una proprietà che deve essere soddisfatta dalle istanze che rappresentano informazioni corrette per l'applicazione.

Un vincolo è una funzione booleana (un **predicato**): associa ad ogni istanza il valore **vero** o **falso**



MODELLO E-R: VINCOLI

I vincoli di integrità fanno parte dei costrutti descrittivi del mondo reale.

Si usano per sia per ottenere una descrizione più accurata della realtà, che per dare un contributo alla “qualità dei dati”.

Si definiscono nella progettazione della base di dati e sono usati dai DBMS nella esecuzione delle interrogazioni.



MODELLO E-R: VINCOLI

I vincoli corrispondono a proprietà del mondo reale modellato dalla base di dati.

I vincoli interessano tutto lo schema (con riferimento cioè a tutte le istanze). Per cui associamo un insieme di vincoli allo schema e consideriamo **corrette** (valide, ammissibili) le istanze che soddisfano tutti i vincoli.

Un'istanza può soddisfare altri vincoli (“per caso”).



MODELLO E-R: CHIAVI

Una superchiave è un insieme di attributi che permettono di identificare univocamente le ennuple di una relazione (ovvero non esistono due ennuple con gli stessi valori di questo insieme di attributi).

Formalmente, un insieme K di attributi si definisce **superchiave** per r se r non contiene due ennuple distinte t_1 e t_2 con $t_1[K] = t_2[K]$

K è **chiave** per r se è una superchiave minimale per r (cioè non contiene un'altra superchiave).

Stabilire che un gruppo di attributi deve essere una chiave, significa imporre un vincolo.



MODELLO E-R: CHIAVI

Esempio: “Matricola” è una chiave perchè è superchiave e contiene un solo attributo (e quindi è minimale).

Matricola	Cognome	Nome	Corso	Nascita
27655	Rossi	Mario	Ing Inf	5/12/95
78763	Rossi	Mario	Ing Inf	3/11/95
65432	Neri	Piero	Ing Mecc	10/7/96
87654	Neri	Mario	Ing Inf	3/11/96
67653	Rossi	Piero	Ing Mecc	5/12/95



MODELLO E-R: CHIAVI

Esempio: impongo il vincolo che la terna Cognome, Nome e Nascita siano chiave.

Matricola	Cognome	Nome	Corso	Nascita
27655	Rossi	Mario	Ing Inf	5/12/95
78763	Rossi	Mario	Ing Civile	3/11/95
65432	Neri	Piero	Ing Mecc	10/7/96
87654	Neri	Mario	Ing Inf	3/11/96
67653	Rossi	Piero	Ing Mecc	5/12/95

La tabella è corretta: soddisfa i vincoli imposti.

Ne soddisfa anche altri ("per caso"): ad esempio la coppia **Cognome, Corso** è chiave.



MODELLO E-R: CHIAVI

L'esistenza delle chiavi garantisce l'accessibilità a ciascun dato della base di dati

Il modello ER è basato su valori e le chiavi permettono di correlare i dati (i valori) tramite le associazioni (relationship).



MODELLO E-R: CHIAVI

Due ennuple di una tabella che rappresenta una relationship non possono mai essere identiche. In altre parole, ogni relationship ha come superchiave l'insieme degli attributi su cui è definita.

Questo significa che ogni associazione deve avere una chiave.



MODELLO E-R: CHIAVI PRIMARIE

In presenza di valori nulli, i valori della chiave non permettono di identificare le ennuple e di realizzare facilmente i riferimenti da altre relazioni. La chiave primaria è una chiave che **non può avere valori nulli**.

<u>Matricola</u>	Cognome	Nome	Corso	Nascita
86765	NULL	Mario	Ing. Inf	05/12/95
78763	Rossi	Mario	Ing. Civile	03/11/95
65432	Neri	Piero	Ing. Mecc	10/07/96
87653	Neri	Mario	Ing. Inf	NULL
43289	Neri	Mario	NULL	05/12/95



MODELLO E-R: INTEGRITÀ

Vincoli di integrità referenziale

Impone che informazioni in relazioni diverse siano tra loro correlate attraverso valori comuni.

La correlazione tra i dati avviene tramite i valori delle chiavi (primarie). Le correlazioni devono essere "coerenti".



MODELLO E-R: INTEGRITÀ

Nel formalismo: un vincolo di **integrità referenziale** (“**foreign key**”) fra gli attributi X di una relazione R_1 e un'altra relazione R_2 impone ai valori su X in R_1 di comparire come valori della chiave primaria di R_2



MODELLO E-R: INTEGRITÀ

Esempio:

Infrazioni

<u>Codice</u>	Data	Vigile	<u>Prov</u>	Numero
34321	1/2/95	3987	MI	39548K
53524	4/3/95	3295	<u>TO</u>	E39548
64521	5/4/96	3295	<u>PR</u>	839548
73321	5/2/98	9345	<u>PR</u>	839548

Vigili

<u>Matricola</u>	Cognome	Nome
3987	Rossi	Luca
3295	Neri	Piero
9345	Neri	Mario
7543	Mori	Gino

