

Modelli dei dati

La progettazione di una base di dati si svolge attraverso una serie di fasi successive, identificabili come:

- Raccolta delle specifiche (identificazione dei concetti)
- Progettazione concettuale
- Progettazione logica
- Progettazione fisica (non compresa nel corso)

Ogni fase fa riferimento ad un *modello dei dati* che corrisponde ad un meccanismo di strutturazione dei dati atto a soddisfare le finalità della corrispondente fase della progettazione

Modelli dei dati

Un modello di dati è costituito dai *concetti* sulla base dei quali i dati sono strutturati e codificati. Ogni modello di dati fornisce meccanismi di strutturazione dati, analoghi ai costruttori di tipo dei linguaggi di programmazione

I ***modelli concettuali*** descrivono la realtà mediante concetti astratti, ma soggetti a precise regole. Non sono finalizzati alla rappresentazione dei dati nel calcolatore, ma a descrivere (e caratterizzare per ruolo) i concetti del mondo reale di cui i dati sono *istanze*

Si usano nella fase di ***progettazione concettuale***.

Nel corso useremo il *modello Entità/Relazione*

Il problema fondamentale

Dato un frammento della realtà che può essere identificato e isolato rispetto al «resto del mondo»:

- Descriverlo nel modo più completo possibile, identificando tutti gli elementi (*concetti*) che lo compongono (e che permettono di distinguerlo da tutto il resto)
- Descrivere, a sua volta, ogni elemento nel modo più corretto possibile, identificandone tutte le caratteristiche (*attributi*) che permettono di caratterizzarlo nel contesto considerato e ignorando quelle superflue rispetto a tale contesto

Il problema fondamentale

Se abbiamo identificato più concetti, questi dovranno risultare logicamente collegati fra loro: ogni concetto deve essere collegato ad almeno un altro concetto, in modo che non vi sia alcun concetto (o insieme di concetti) «isolato» dagli altri.

Se così fosse, infatti, questo costituirebbe un secondo frammento della realtà indipendente da quello considerato e non avrebbe senso descriverli insieme.

Il problema fondamentale

Fra i concetti, alcuni (*entità*) potranno essere descritti in modo indipendente dagli altri attraverso un insieme di *attributi* che gli sono propri e tipici, senza bisogno di fare riferimento ad altri concetti.

Ad es., se il contesto è la gestione di un negozio, posso descrivere un prodotto attraverso il nome, il peso, il prezzo, ecc. ... e posso descrivere un cliente attraverso i suoi dati anagrafici.

Il problema fondamentale

Però non basta... due concetti che posso descrivere in modo indipendente l'uno dall'altro come cliente e prodotto non possono costituire la descrizione completa di un contesto comune

Per descrivere il contesto del negozio, quindi, cliente e prodotto non possono fare parte della stessa descrizione (*schema*) senza che esista (almeno) un altro concetto che li colleghi

D'altra parte, ovviamente, un tale concetto non può, in alcun caso, essere indipendente e non può essere descritto senza *fare riferimento* ai concetti che collega

Il problema fondamentale

Un concetto che collega altri concetti, e che quindi non può esistere se non esistono i concetti che collega, si chiama *relazione* o *associazione*

Il problema fondamentale

Un concetto che collega altri concetti, e che quindi non può esistere se non esistono i concetti che collega, si chiama *relazione* o *associazione*

Nel caso del negozio, dato il concetto di cliente e di prodotto, una possibile associazione sufficiente a completare lo schema è ...

Il problema fondamentale

Un concetto che collega altri concetti, e che quindi non può esistere se non esistono i concetti che collega, si chiama *relazione* o *associazione*

Nel caso del negozio, dato il concetto di cliente e di prodotto, una possibile associazione sufficiente a completare lo schema è

VENDITA

Il problema fondamentale

Ancora non basta ...

Una relazione è un concetto che non è descrivibile soltanto attraverso i riferimenti agli altri concetti (anche più di due) che collega, ma è anch'essa caratterizzata da specifici attributi

Ad esempio, la descrizione di una vendita dovrà necessariamente contenere l'informazione relativa al cliente cui viene venduto un prodotto e al prodotto che ne è oggetto, ma potrà anche specificare ...

Il problema fondamentale

Data della vendita

Quantità del prodotto venduto

Prezzo unitario del prodotto o prezzo totale

...

Numero della fattura in cui viene riportata

(in questo caso sarà necessario definire un'entità Fattura a cui fare riferimento, quindi estendere lo schema)

Modelli dei dati

Modello (concettuale) Entità/Relazione



Si esprimono i **concetti** che devono essere rappresentati dai dati come concetti indipendenti (**entità**) e relazioni logiche che li collegano (**relazioni**) e che esistono solo in presenza delle entità.

Modelli dei dati

I DBMS si differenziano in base al *modello logico* che utilizzano.

Quindi i ***modelli logici***, seppure astratti come i modelli concettuali, non hanno solo finalità descrittive e di documentazione, ma riflettono la *struttura* con cui i dati sono organizzati all'interno del calcolatore

Modelli dei dati

I DBMS si differenziano in base al *modello logico* che utilizzano. I **modelli logici**, seppure astratti, riflettono la *struttura* con cui i dati sono organizzati.

- **Relazionale**

Il più diffuso, basato su un modello tabellare dei dati

- **Gerarchico**

Utilizzato nei primi DBMS (anni 60), tuttora utilizzato, basato su strutture ad albero

- **Reticolare**

Estensione del modello gerarchico, basato su grafi

- **A oggetti**

Estensione del modello relazionale basato sui paradigmi OOP.

- **XML**

(semistrutturato) Deriva dal modello gerarchico, ma è più flessibile

Modello Relazionale

E' un *modello logico*: definisce tipi attraverso il costruttore *relazione*, che organizza i dati secondo **record a struttura fissa**, rappresentabili attraverso tabelle.

Es. (relazioni INSEGNAMENTO e MANIFESTO)

Corso	Titolare
Basi di Dati e Web	Cagnoni
Fondamenti di Informatica	Tomaiuolo
Ingegneria Del Software	Poggi

CdL	Materia	Anno
IET-I	Basi di Dati e Web	3
IET	Fondamenti di Informatica	1
IET-I	Ingegneria del Software	3

Schemi

In ogni base di dati si possono distinguere:

- Lo **schema**, sostanzialmente invariante nel tempo. Ne descrive la struttura (*aspetto intensionale*)
Nell'esempio, le tabelle e le «intestazioni» delle tabelle
*Analogo del **prototipo** di una funzione in C/C++*
- Le **istanze**, cioè i valori attuali, che possono cambiare anche molto rapidamente
(*aspetto estensionale*)
Nell'esempio, i singoli dati contenuti in ciascuna tabella

Schemi

Lo *schema* di una base di dati è la parte *dichiarativa* ed *invariante* della base di dati e ne definisce la struttura.

Nel modello relazionale lo schema di una relazione è paragonabile alla definizione del prototipo di una funzione in C.

INSEGNAMENTO (Corso, Titolare)

è lo *schema* della relazione INSEGNAMENTO definita su due *attributi* (campi del record identificati da una etichetta che ne descrive il significato).

Le n-ple di *attributi* appartenenti alla relazione, a ciascuno dei quali deve essere assegnato un valore, sono dette *istanze* della relazione.

(Basi di dati, Cagnoni)

è una *istanza* di INSEGNAMENTO

Schemi

Gli schemi possono operare a diversi livelli di astrazione

Dalle specifiche è possibile derivare lo

- **Schema concettuale**
descrive i concetti che devono essere rappresentati, i dettagli che servono per descriverli e le relazioni logiche che li collegano

Schemi

Dallo schema concettuale si deriva lo

- **Schema logico**

descrive la struttura dell'intera base di dati
mediante il modello logico adottato dal DBMS
(reticolare, gerarchico, relazionale)

che viene poi realizzato attraverso lo

- **Schema interno (o schema fisico)**

implementa lo schema logico mediante strutture
fisiche di memorizzazione

Schemi

Nei confronti dell'utente è possibile definire anche uno

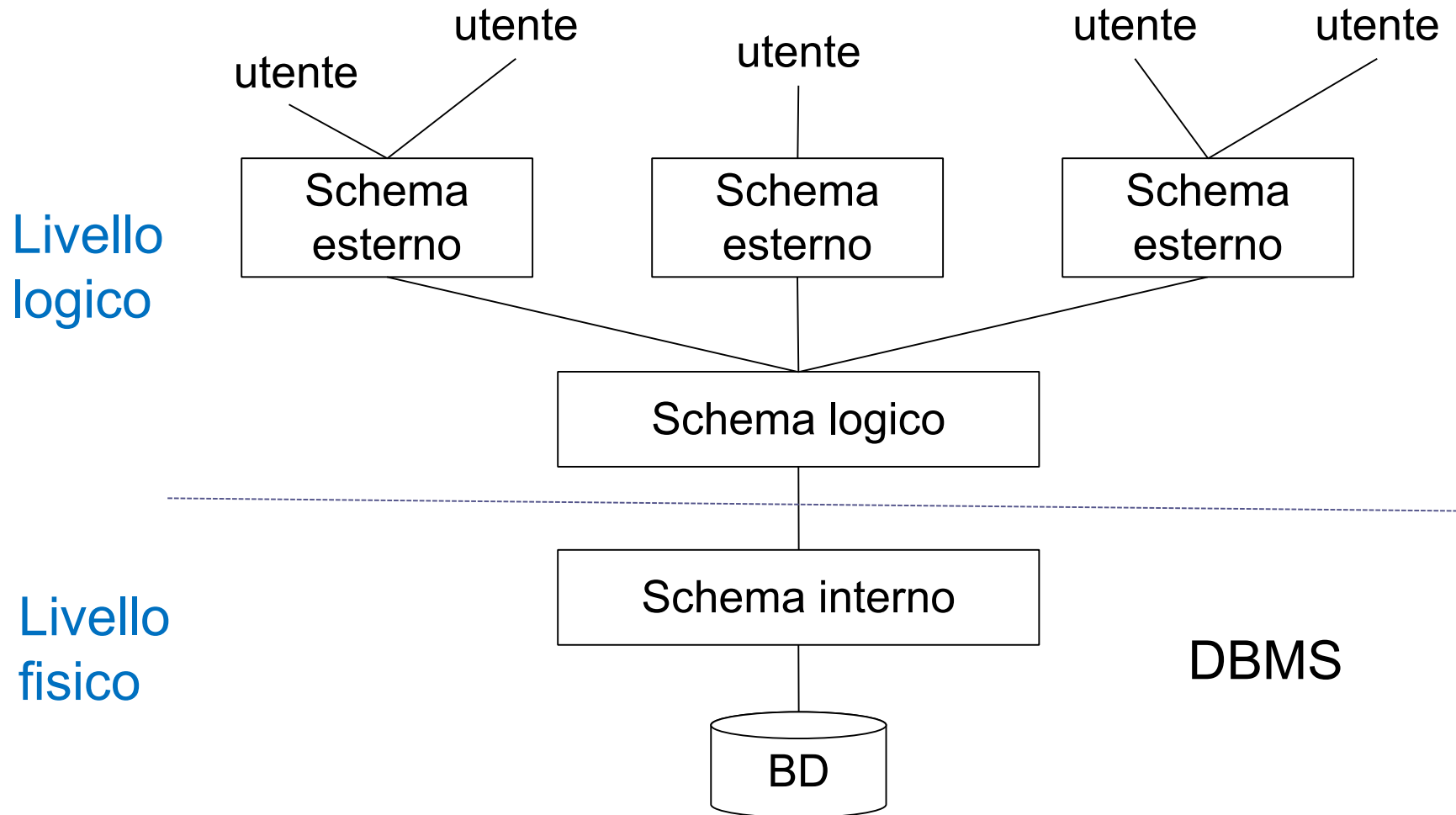
- **Schema esterno**

descrive la struttura di una porzione della base di dati attraverso il modello logico, dal punto di vista di una classe di utenti. Generalmente realizzato mediante *viste*, relazioni derivate da quelle che costituiscono lo schema logico.

Es. (i soli corsi del curriculum Ingegneria Informatica)

CdL	Materia	Anno
IET-I	Basi di Dati e Web	3
IET-I	Ingegneria del Software	3

Architettura standard (ANSI/SPARC) a tre livelli per DBMS



Progettazione di una base di dati

I diversi tipi di schema riflettono le diverse fasi della progettazione: date le specifiche del problema da risolvere

Progettazione concettuale

(dalle specifiche allo schema concettuale)

Definisce quali concetti rappresentare, per quanto riguarda sia la loro descrizione che le relazioni logiche che esistono fra di essi.

Progettazione logica

(dallo schema concettuale allo schema logico)

Definisce come rappresentare i concetti introdotti a livello concettuale in forma di strutture dati.

Progettazione fisica (dallo schema logico allo schema fisico)

Definisce come allocare e gestire fisicamente i dati all'interno del calcolatore.

Accesso alla base di dati

- Linguaggi testuali interattivi
- Comandi inclusi in estensioni di linguaggi tradizionali
- Comandi inclusi in linguaggi di sviluppo ad hoc
- Interfacce grafiche amichevoli

Linguaggi per basi di dati

- **Linguaggi di definizione dei dati**

Utilizzati per definire gli schemi e le autorizzazioni per l'accesso

- **Linguaggi di manipolazione dei dati**

Utilizzati per l'interrogazione e l'aggiornamento dei contenuti della base di dati

Alcuni linguaggi specializzati (es. SQL) presentano le caratteristiche di entrambi i tipi di linguaggio.

SQL, un linguaggio interattivo

```
SELECT Corso, Aula, Piano  
FROM Aule, Corsi  
WHERE Nome = Aula  
AND Piano = "Terra"
```

Corsi

Corso	Aula
Sistemi	N3
Reti	N2

Nome	Piano
N3	Terra
N2	Terra

Aule

Corso	Aula	Piano
Sistemi	N3	Terra
Reti	N2	Terra

SQL immerso in linguaggio ad alto livello

```
write('nome della citta"?'); readln(citta);
EXEC SQL DECLARE P CURSOR FOR
    SELECT NOME, REDDITO
    FROM PERSONE
    WHERE CITTA = :citta ;
EXEC SQL OPEN P ;
EXEC SQL FETCH P INTO :nome, :reddito ;
while SQLCODE = 0 do begin
    write('nome della persona:', nome, 'aumento?');
    readln(aumento);
    EXEC SQL UPDATE PERSONE SET REDDITO = REDDITO +
        :aumento
        WHERE CURRENT OF P
    EXEC SQL FETCH P INTO :nome, :reddito
end;
EXEC SQL CLOSE CURSOR P
```

PHP embedded in codice HTML

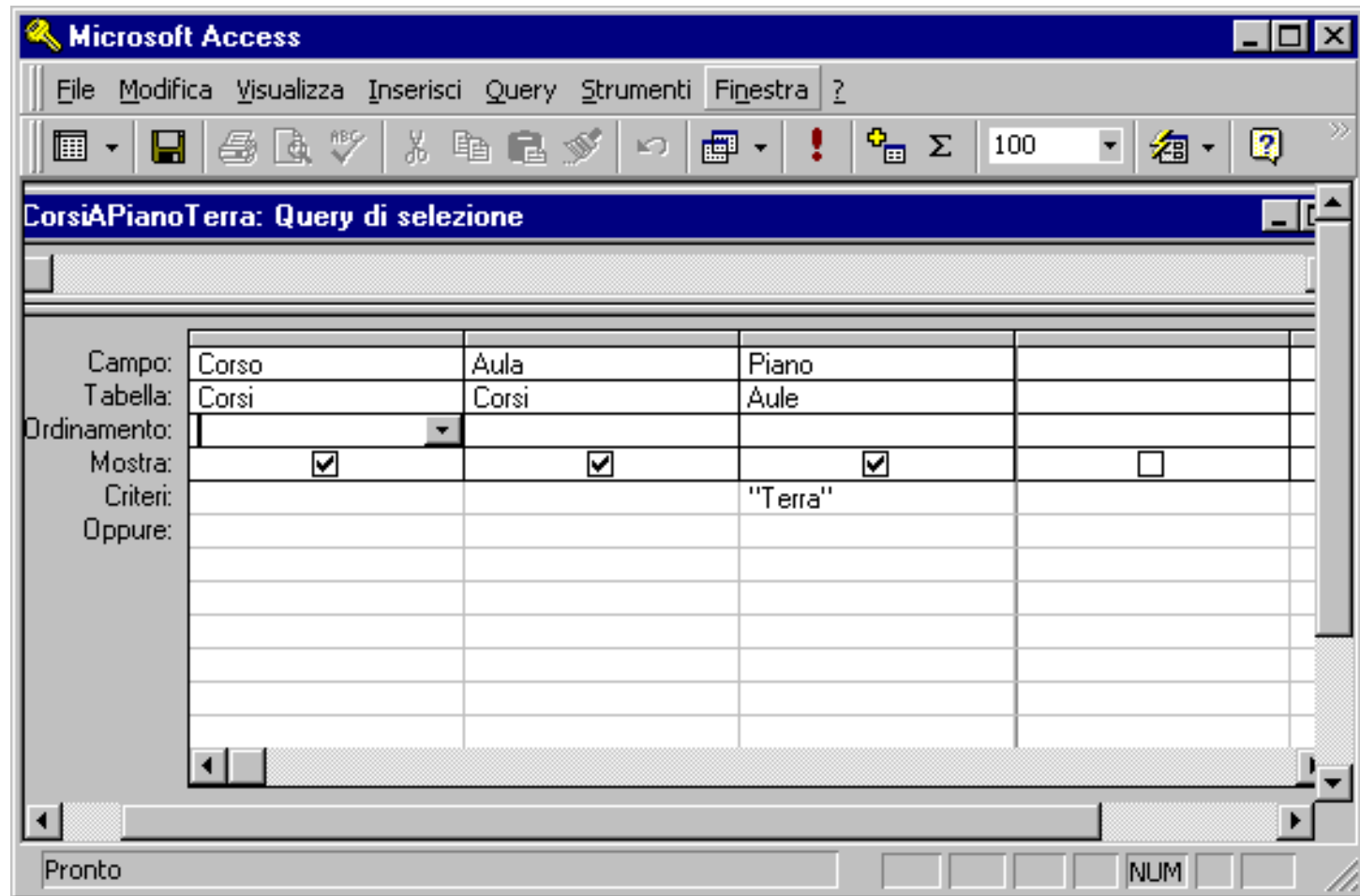
```
<html> <body>
<?php
$user="root"; $password=""; $host="127.0.0.1"; $database="prova";
mysql_connect($host,$user,$password);
@mysql_select_db($database) or die("Unable to select database");
$query="SELECT * from utenti";
$res=mysql_query($query);
mysql_close();

$row = mysql_fetch_assoc($res);
print '<table><tr>';
foreach($row as $name => $value) {
print "<th>$name</th>";}
print '</tr>';

while($row) {
print '<tr>';
foreach($row as $value) {
print "<td>$value</td>";}
print '</tr>';
$row = mysql_fetch_assoc($res);}
print '</table>';
?> </body> </html>
```

Connessione a mySQL
tramite API PHP

Interazione non testuale (in Access)



Realizza l'approccio detto "Query by Example"

Vantaggi dei DBMS

- Disponibilità dei dati a tutta una comunità
- Modello unificato e preciso della realtà di interesse
- Controllo centralizzato dei dati
- Condivisione
- Indipendenza dei dati

Svantaggi dei DBMS

- Prodotti costosi, complessi, che richiedono investimenti in hardware, software, personale.
- Forniscono un numero elevato di servizi, in modo integrato e difficilmente scorporabile se le esigenze dell'utente sono inferiori alle caratteristiche offerte