



UNIVERSITÀ DEGLI STUDI DI PALERMO

INFORMATIZZAZIONE DELLA DOCUMENTAZIONE SANITARIA

Anno Accademico 2019/2020

Docente: ing. Salvatore Sorce

Basi di dati II

DBMS

- Un **DBMS** (**D**ata **B**ase **M**anagement **S**ystem) è un sistema software in grado di gestire efficientemente le informazioni necessarie a un sistema informativo, rappresentandone i dati in forma integrata, e garantendone la persistenza.
- Un DBMS nasce allora con l'intento di gestire DB Grandi, Condivisi e Persistenti, garantendo loro
 - *Affidabilità e Privatezza*
- con
 - *Efficienza ed Efficacia.*
- ***Basi di Dati \Leftrightarrow collezione di dati gestita da un DBMS.***

Modelli dei Dati: il Modello Relazionale

- 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.
- Attualmente il modello più diffuso e più consolidato è il **Modello Relazionale**:
 - 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 dei Dati: il Modello Relazionale

- Il modello relazionale nella modellazione dei dati è stato applicato a partire dalla fine degli anni 60' da E.F. Codd
- Questo modello definisce:
 - il modo in cui i dati possono essere rappresentati (struttura dei dati)
 - il modo in cui i dati possono essere protetti (integrità dei dati)
 - le operazioni che si possono eseguire sui dati (manipolazione dei dati)
- Nei DB esiste una parte che è sostanzialmente invariante nel tempo (schema della basi di dati) ed un parte variabile nel tempo (istanza o stato della base di dati).

DBMS Relazionali

- Una **Tabella** corrisponde ad una relazione
- ogni tabella memorizza **informazioni su un singolo insieme di "oggetti"** ed è composta da:
 - **righe (record, ennuple o tuple)** ognuna corrispondente ad una singola "voce" (ad esempio un libro, una persona, una canzone)
 - **colonne (campi o attributi)**, corrispondenti a informazioni relative alle "voci" (ad esempio il titolo del libro, il suo numero di pagine oppure il nome della persona o il titolo del brano)
 - ogni record possiede almeno un campo, detto **campo chiave**, che lo identifica univocamente

DBMS Relazionali

- I sistemi di database relazionali devono possedere determinate e precise caratteristiche:
 - Tutti i dati sono rappresentati in strutture ordinate in righe e colonne (Relazioni)
 - Tutti i valori sono scalari cioè in ogni cella formata dall'incontro di una riga con una colonna deve esserci uno ed un solo valore
 - Ogni operazione è effettuata su un'intera relazione ed ha come risultato una relazione intera



DBMS Relazionali: creazione di un DB

- Per utilizzare un DBMS **bisogna innanzitutto costruire le tabelle che lo compongono.**
- Per prima cosa bisogna identificare il **tipo di informazione** che va memorizzata in ogni tabella e quali delle **sue caratteristiche** (campi) sono da considerare pertinenti per l'applicazione. Ad ogni campo va poi assegnato un **nome**, una **lunghezza** e una **categoria di dati**
- Un'applicazione semplice può presentare anche una sola tabella, ma più **spesso un DB è composto da più tabelle**, collegate tra di loro tramite valori di campi comuni.

Il modello relazionale

STUDENTI				
Matr	Cognome	Nome	Nato_il	Nato_a
3571	Banfi	Alessandro	19/02/1982	Milano
999	Bosio	Umberto	27/01/1983	Aosta
2805	Castelnuovo	Andrea	06/05/1982	Torino
3719	Colpi	Marco	15/01/1983	Genova
773	Izzo	Stefania	08/10/1982	Firenze
3672	Librandi	Silvia	12/03/1983	Bologna
1539	Longoni	Mauro	05/02/1983	Venezia
3500	Matta	Vera	26/04/1982	Roma
1886	Merlo	Andrea	05/05/1983	Trento
1427	Morelli	Riccardo	14/04/1982	Trieste
2608	Ornaghi	Gabriele	09/09/1982	Perugia
3711	Panico	Andrea	29/05/1982	Pescara
1940	Poretti	Stefania	20/02/1982	Ancona
1814	Quaglia	Andrea	13/08/1982	Napoli
1662	Salmoiraghi	Veronica	19/09/1982	Cagliari
2744	Sterlocchi	Elena	29/06/1982	Palermo
3024	Tarantola	Marcello	17/06/1982	Reggio Calabria
3527	Valentini	Samuele	10/07/1982	Bari
3615	Venturi	Anita	28/07/1982	Potenza
681	Zaccaretti	Carolina	23/02/1983	Campobasso

ISCRITTI	
Matr	Codice
2805	IG06
3527	BA03
1940	IG10
773	IG11
1539	IG05
1940	IG03
3672	ICT3
681	ICT2
1886	IG05
1940	ICT1
3500	BA08
1886	IG01
3024	BA01
3719	IG10
3672	IG08
773	ICT2
3719	IG06
1814	ICT2
2744	BA09
2744	IG03
2805	IG09

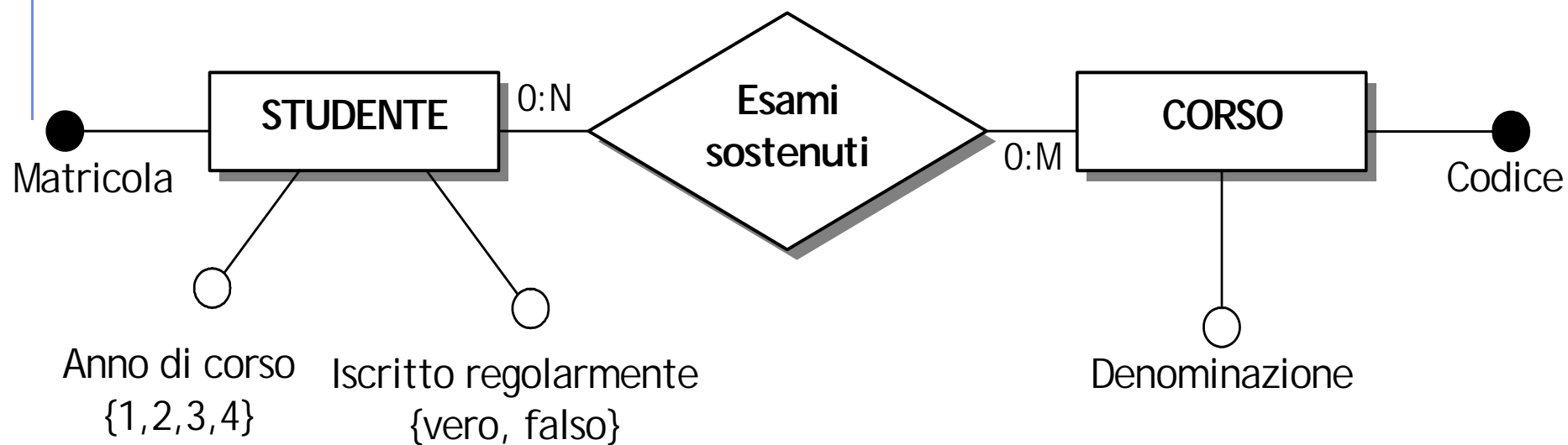
CORSI				
Codice	Titolo	Settore	Tipo	Crediti
BA01	Analisi Matematica I	MAT/05	Base	7,5
BA02	Analisi Matematica II	MAT/05	Base	7,5
BA03	Elettromagnetismo	FIS/01	Base	5
BA04	Fond. Meccanica Teorica e Applicata	ING-IND/13	Affine	5
ICT1	Fond. Informatica I (laboratorio)	ING-INF/05	Affine	4
IG01	Elettrotecnica	ING-IND/31	Affine	5
IG02	Fisica Tecnica	ING-IND/10	Affine	5
ICT2	Fond. Informatica II	ING-INF/05	Base	6
IG03	Fond. Automatica	ING-INF/04	Caratt.	5
...
IG04	Economia Organizzazione Aziendale	ING-IND/35	Caratt.	10
IG05	Gestione Produzione Industriale	ING-IND/17	Caratt.	10
BA09	Ricerca Operativa	MAT/09	Base	5
ICT3	Produzione Assistita Calcolatore	ING-IND/16	Caratt.	5
IG09	Sistemi di Controllo di Gestione	ING-IND/35	Caratt.	5
IG10	Logistica Industriale	ING-IND/17	Caratt.	5
IG11	Gestione Aziendale	ING-IND/35	Caratt.	5
IG12	Gestione della Qualità	ING-IND/17	Caratt.	5

Studenti (Matr, Cognome, Nome, Nato_il, Nato_a);
 Corsi (Codice, Titolo, Settore, Tipo, Crediti);
 Iscritti (Matr, Codice).

Modello concettuale dei dati

- Modello dei dati è la descrizione (mediante un linguaggio formale) della realtà che si vuole rappresentare.
- Tipicamente un modello dei dati è costituito da:
 - *entità*, organizzate in *insiemi di entità*
 - *attributi*, ognuno caratterizzato da un *insieme di valori* e valutabile su un insieme di entità, nel senso che a ogni entità dell'insieme è associato un valore dell'attributo
 - *relazioni*, particolari attributi caratterizzati da un insieme di valori che è in effetti un insieme di entità

Modello concettuale dei dati



Le chiavi

- Si definisce chiave di una tabella un campo, o un insieme di campi, il cui contenuto sia garantito unico tra i vari record della tabella.
- La chiave primaria viene impiegata anche per mettere in corrispondenza record appartenenti a tabelle diverse.

Query

- Una Query (interrogazione) è una frase scritta in un linguaggio speciale che permette
 - la **ricerca dei dati**: creazione di una nuova tabella con i dati (campi e attributi) di interesse
 - **l'aggiornamento dei dati**
- Tramite una Query si crea una **nuova tabella che contiene solo i campi e i record che interessano**



ID	Cognome	Nome	Via	Città	Telefono	ID	Data	Importo
1001	Jones	Frank	100 Elm	Oakland	555-1789	1001	1/10/99	10.00
1002	Smith	Susan	200 Main	Sarasota	555-1902	1002	1/11/99	15.00
1003	Ashley	Bill	100 Woods	San Jose	555-0232	1002	1/12/99	20.00

Cognome	Telefono	Data	Importo
Smith	555-1902	1/11/99	15.00
Smith	555-1903	1/12/99	20.00

Maschera Video

- Consente una **semplice interazione con le tabelle** del DB, alle quali è collegata direttamente.
- Tramite una maschera è possibile
 - visualizzare
 - immettere
 - cancellare
 - modificarei dati nelle tabelle del Database
- Una Maschera permette di
 - selezionare solo alcuni campi della tabella da visualizzare
 - visualizzare dati provenienti da tabelle diverse

Linguaggi per DB

- Su un DBMS è possibile ovviamente specificare operazioni attraverso linguaggi:
 - DDL (Data Definition Language) utilizzati per la definizione di schemi.
 - DML (Data Manipulation Language) utilizzati per interrogazioni ed aggiornamenti.
 - DCL (Data Control Language) utilizzati per definire utenti e privilegi.
- Alcuni linguaggi (SQL) integrano le varie funzionalità.

Esempio di query in SQL

```
SELECT      Matr, Cognome, Nome  
FROM        Studenti  
WHERE       Nato_a = 'Milano'
```

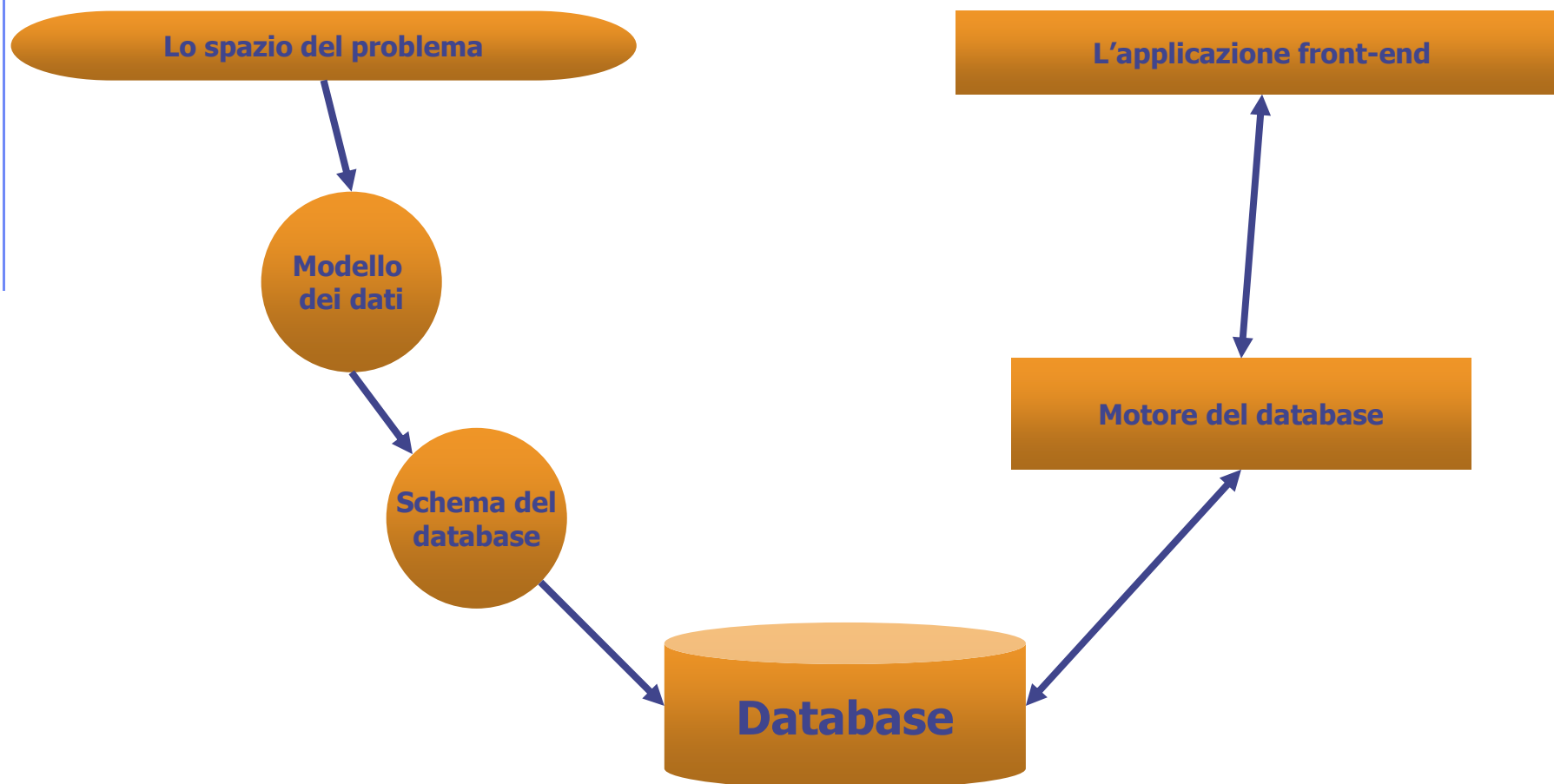
genera una nuova tabella il cui schema è (Matr, Cognome, Nome) e che comprende l'elenco degli studenti nati a Milano.

Esempio di query in SQL

```
SELECT      Studenti.Matr, Cognome, Nome
FROM        Studenti, Iscritti, Corsi
WHERE       Studenti.Matr = Iscritti.Matr
           AND Iscritti.Codice = Corsi.Codice
           AND Corsi.Titolo = "Fondamenti di Informatica II"
ORDER BY   Cognome
```

genera una tabella con l'elenco, ordinato per cognome, degli studenti iscritti al corso di "Fondamenti di Informatica II", e per ognuno di essi include il numero di matricola, il cognome e il nome.

Rappresentazione grafica della terminologia



Ambiente per lo sviluppo e l'utilizzo di un DB relazionale

The screenshot shows a database management system interface with several windows and components:

- Menu comandi:** Located at the top left, pointing to the menu bar (File, Modifica, Visualizza, Inserisci, Query, Strumenti, Finestra, Guida).
- Barra strumenti:** Located at the top right, pointing to the toolbar with various icons.
- Schema delle relazioni:** Located in the center-left, pointing to the 'Relazioni' window showing a database schema with tables like Fornitori, Prodotti, Ordini, and Clienti.
- Elementi del DB (Tabelle, Query, ...):** Located in the top right, pointing to the 'Northwind : Database' window showing a list of database objects.
- Struttura di una query SQL:** Located in the middle right, pointing to a window titled 'Clienti e fornitori per città: Query di unione' containing SQL code:


```
FROM Clienti
UNION SELECT Città, NomeSocietà, Contatto, "Fornitori"
FROM Fornitori
ORDER BY Città, NomeSocietà;
```
- Struttura di una query by example:** Located in the bottom left, pointing to a window titled 'Dettagli ordini complessivi: Query di selezione' showing a table structure and a grid for query criteria.
- Scheda di un cliente:** Located in the bottom right, pointing to a window titled 'Clienti' showing a form for a client record with fields like ID cliente, Nome società, Contatto, Posizione, Indirizzo, Città, Zona, CAP, Paese, Telefono, and Fax.

Architetture per un DBMS (1)

➤ **Architettura Centralizzata**

- Tutti i processi sono elaborati dall'host
- I terminali sono meri strumenti per l'immissione delle richieste e la visualizzazione dei risultati
- Molto usata (banche, anagrafe, voli...) per la forte specializzazione e alte prestazioni
- Non adatto per settori in rapido mutamento
- Rapporto prezzo/prestazioni alto

➤ **Architettura basata su LAN**

- Local Area Network (LAN) mette in comunicazione ad alta velocità molti PC
- Condivisione di risorse
- Flessibile, affidabile, espandibile, riconfigurabile

Architetture per un DBMS (2)

➤ **Architettura Client-Server**

- Divisione equa dei compiti tra client e nodi server
- Applicazioni client (front-end – interfaccia utente), raccoglie la logica delle richieste e la gestione dei risultati
- Server (back-end), elabora le richieste e fornisce i risultati
- Da client a server: richieste SQL
- Da server a client: insiemi di risposte (parti di tabelle)

Architettura Client-Server

➤ **Client:**

- Gestione dell'autenticazione
- Maschere progettate ad hoc per l'immissione
- Controllo della validità dei dati inseriti
- Gestione dei dati forniti dal server (display, stampa, ecc.)

➤ **Server:**

- Controllo della validità e integrità dei dati
- Aggiornamento di tabelle, indici e oggetti temporanei
- Gestione delle transazioni
- Esecuzione di query
- Restituzione dei risultati
- Controllo accessi e sicurezza

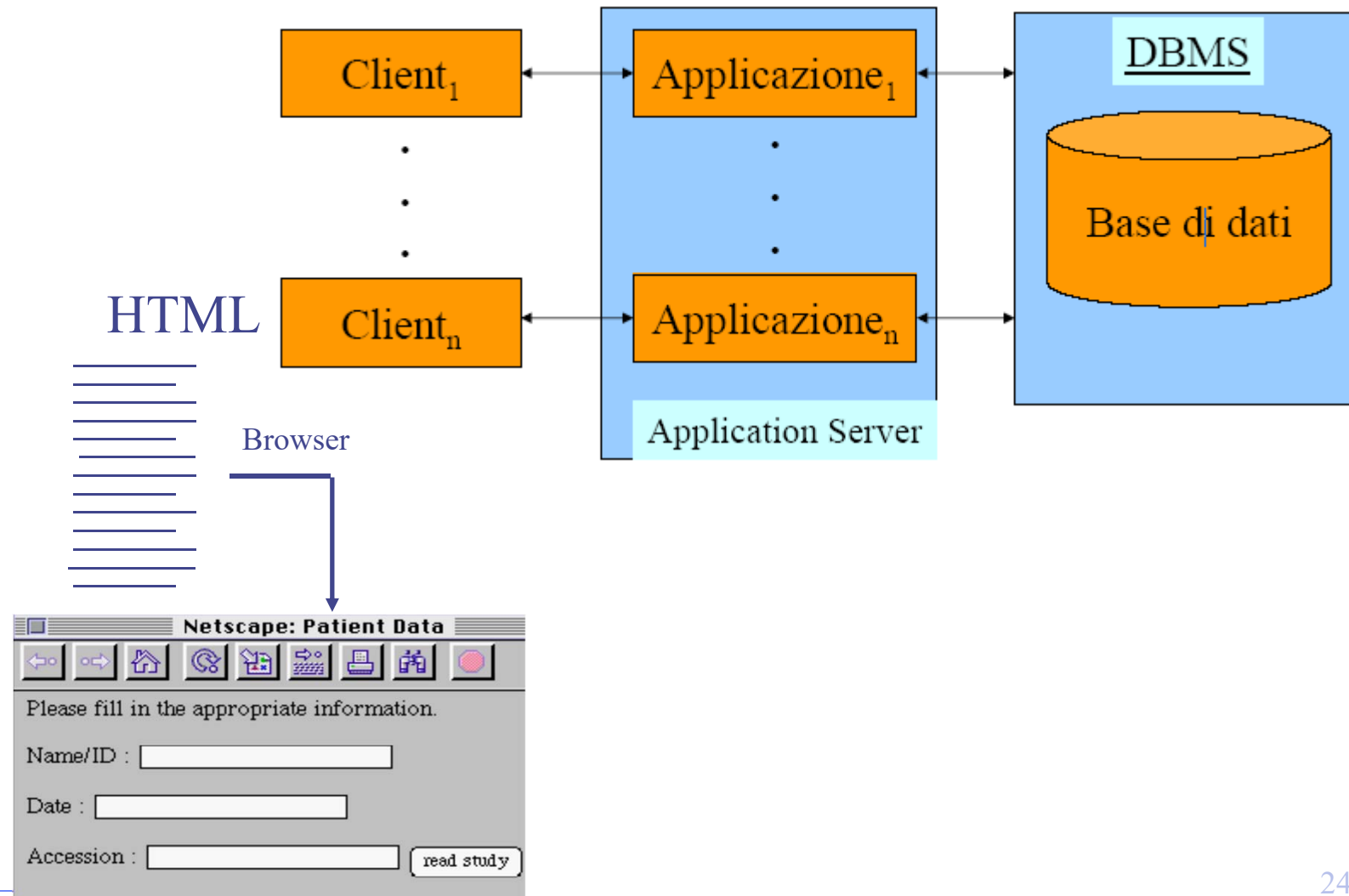
Applicazioni WEB

- Dimora ideale del client-server
- Internet per inviare e ricevere dati
- Enti coinvolti:
 - **Web browser:** contenitore dell'applicazione client
 - **Client:** oggetti attivi contenuti in pagine HTML accedono al DBMS via internet
 - **Web server:** consente di accedere alle pagine web memorizzate sul server
 - **Application server:** strato intermedio, interpreta le richieste del client e preleva i dati dal DB
 - **DBMS:** contenitore dei dati

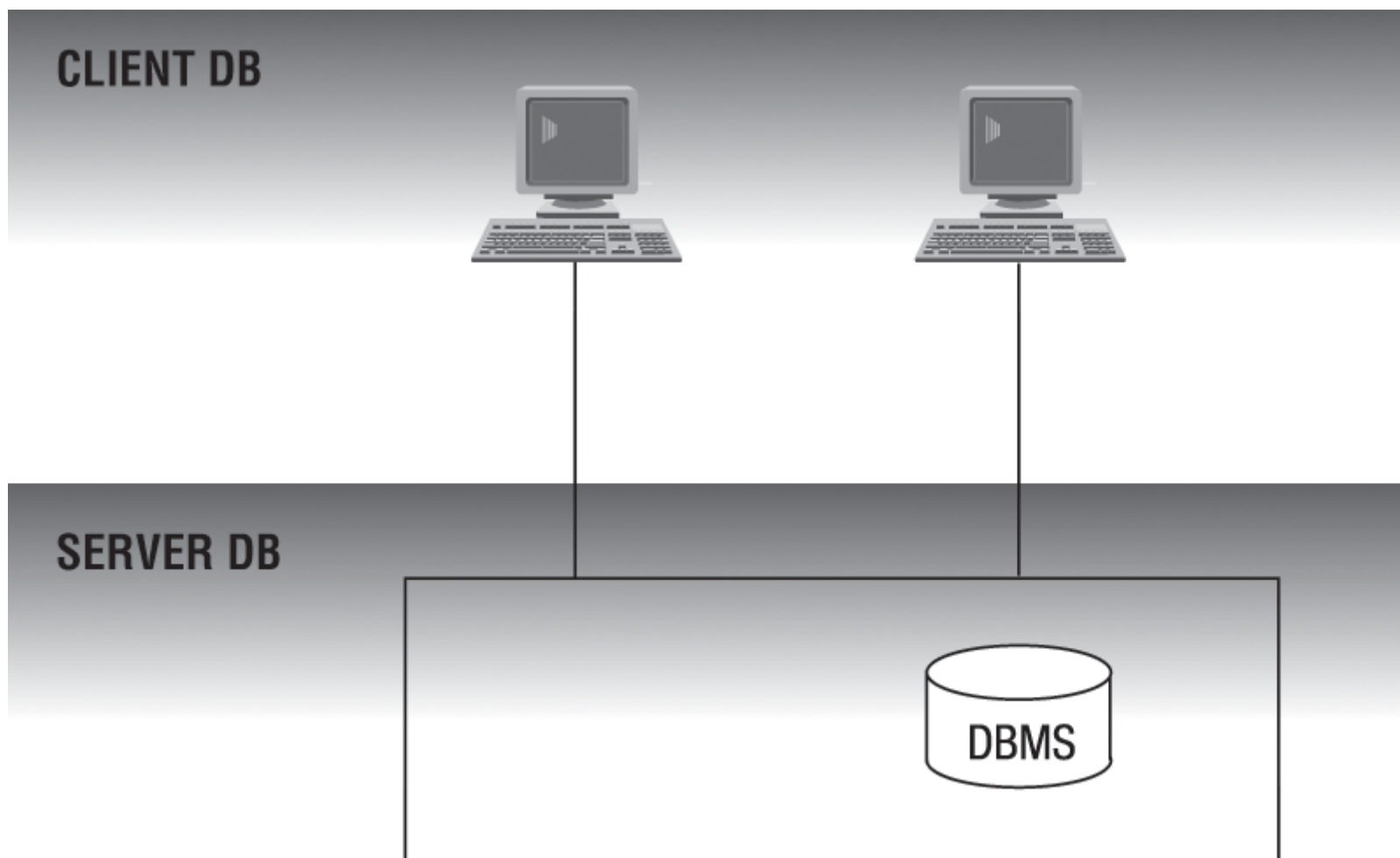
Applicazioni WEB

- **Vantaggi**
 - Non è necessaria l'installazione
 - Si scarica dalla rete e si esegue localmente
 - Con le pagine Web dinamiche la flessibilità di questi oggetti è aumentata
- **Punto critico**
 - efficienza della rete

Accesso ai Database: Applicazioni WEB



Elaborazione client/server



Elaborazione client/server su più livelli

