Caratteristiche delle Basi di Dati

* autodescrizione → linguaggi DDL, DML
* indipendenza (fisica, logica) → Astrazione
* visite multiple → multi utenza, linguaggio DQL
* controllo della concorrenza → condivisione
* controllo’dell’accesso → linguaggio DCL

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Problema: dei film abbiamo le recensioni. Una recensione è prodotta da un soggetto in una certa data. Lavalutazione è numerica su una certa scala (e.g. 8/10 , 4/5).

proposta dalla classe:

rating(punteggio, punteggio\_massimo, recensore\_nome, recensore, cognome, valore, movie\_id, genere, rating\_date)

genere → non serve perché con movie\_id posso passare alla tabella movie\_genere e reperire l’informazione da lì e con una singola colonna non potrei avere generi multipli per film.

Valore → si dice attributo derivato. É ricavato dai campi punteggio/punteggio\_valore. È sensato memorizzare un attributo derivato se la sua computazione è onerosa e quindi ha senso computarlo una volta e memorizzarlo.

Punteggio → non serve perché tenendo punteggio\_massimo e valore posso ricavarlo

recensore\_nome, recensore\_cognome → per il recensore non mi serve una tabella dedicata perché non ho altre informazioni da memorizzare e nomi ugali collasserebbero nello stesso record. Tenere nome e cognome separati serve solo se so che devo fare operazioni diverse con solo il nome o solo il cognome. In questo caso non mi serve quindi tengo solo un campo recensore.

movie\_id → tipica relazioni uno a molti: ogni recensione è collega a un solo film, ma ogni film può avere più recensioni. Nella tabella del “lato uno” aggiungo quindi l’identificatore del “lato molti”

arrivo alla tabella più sensata→ rating(punteggio\_massimo, valore, recensore, movie\_id, genere, rating\_date)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

DBMS

DataBase Management System

software in grado di gestire collezioni si dati che siano **grandi, condivise e persistenti**

Deve assicurare l’**affidabilità** e **coerenza** dei dati

DBMS facilita il processo di definire, costruire, manipolare e condividere basi di dati.

Architettura a tre livelli fisico→logico→esterno

Varie funzionalità:

* Istruzioni DDL: creare
* Comandi privilegiati
* Interrogazioni
* Programmi applicativi
* Transazioni predefinite: caratteristiche legate a funzionalità di sicurezza.
* Backup e recovery

visita → non tutti gli utenti sono interessati alle stesse tabelle.

Come interagisco? Devo dividere diverse categorie di utenza

* DBA (amministratore)
* DB user

I DBA sono utenti con privilegi di amministratore che crea (tipicamente) altri utenti per gestire/accedere ciascuna base di dati. Ogni base di dati ha almeno un DBA. Ha privilegi massimi.

I DB user sono utenti con privilegi limitati deteminati dal DBA.

* Devel user: utenti che possiamo assumere esperti che può agire con una certa libertà a cui il DBA può dare privilegi più ampi per aiutare nella gestione (e.g.)
* occasional user: utente di cui non si sa molto, ma che si suppone sappia utilizzare SQL e abbia un’idea dello schema della base di dati e può agire con una certa libertà.
* casual user: Utente che non conosce come usare una base di dati. Da questi utenti bisogna proteggersi in un certo senso, per evitare danni alla base di dati. Interagisce con la stessa tramite delle funzioni “black box” che devo mettere a disposizione.

Porta 5432 nel caso di connessione socket in rete, se uso sulla stessa macchina uso pipe del sistema operativo.

Al server postgress bisogna collegarsi tramite software client.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

IL MODELLO RELAZIONALE

* Concetto fondamentale: la relazione è una tabella.
* Una tabella è una funzione sui vari attributi che la compongono.
* Ogni tabella ha un nome e un insieme di proprietà. Ciascuna proprietà (o attributo) di una relazione ha un nome e un dominio (o tipo di dato). Il dominio è quindi l’insieme dei valori che l’ attributo può assumere. Il dominio quindi viene definito dal tipo di dato e un set di regole.

R nome della tabella/relazione

R: D1 x D2 x … x Dn Di è il dominio dell’attributo Ai

X = A1, A2, … , An I X è insieme degli attributi della mia relazione

R(X) la tabella

e.g.

R = luoghi\_abitati

X = nome\_città, area

D1 = {Roma, Milano}

D2 = {Nord, Centro, Sud}

R può avere le seguenti istanze

Roma, Nord

Roma, Centro

Roma, Sud

Milano, Nord

Milano, Centro

Milano, Sud

Questo è il prodotto cartesiano sui domini degli attributi della relazione R, cioè tutte le possibili istanze che possono essereci nella relazione.

R sarà un sottoinsieme di D1xD2

per esempio

Roma, Nord

Milano, Sud

o

Roma, Centro

Roma, Sud

Milano, Nord

Milano, Centro

* cardinalità: il numero di istanze contenute nella relazione.
* grado: il numero di attributi della relazione (numero delle colonne)

Una buona progettazione dovrebbe portare ad avere una tabella cui grado non varia a meno di variazioni dei requisiti iniziali.

Un’istanza è un record/riga/n-upla della tabella.

La struttura della tabella è rigida.

A1 A2 A3

T: D3 x D4 x D5

D3 = {x, y}

D4 = {a, b, c}

D5 = {k}

il grado di T è 3

prodotto cartesiano:

x, a, k

x, b, k

x, y, k

y, a, k

y, b, k

y, c, k

la cardinalità del prodotto cartesiano è 6

Notazione per indicare un valore (fissata una n-upla) all’interno della relazione sull’attributo Ak → T[Ak]

Le righe di una tabella non hanno un criterio specifico di ordinamento. Non hanno quindi un criterio di riferenziazione.

Schema di relazione: R(A1...An)

Schema di una base di dati: DB = {R1(X1), …, Rm(Xm)}

in cui X1 = {A1\_1, …, A1\_k} … Xm = {Am\_1, …, Am\_j}

Quindi lo schema di una base di dati è la collezione di relazioni contenute in una base di dati. Ogni relazione ha un proprio schema di relazione