

Basi di Dati e Conoscenza

Il anno

Loredana Vigliano

Stanza 0103 TEL. 0672594645

vigliano@mat.uniroma2.it

Obiettivi del corso

I parte

- Conoscere l' algebra su cui si basa la teoria dei Database Relazionali
- Imparare a progettare e a costruire un Database Relazionale
- Utilizzare al meglio l' SQL

Obiettivi del corso

II parte

- Approfondire la conoscenza delle basi di dati
- Ottimizzare la gestione di una base di dati migliorandone le prestazioni.
- Avere una panoramica delle altre realtà esistenti sul trattamento dei dati.

Propedeuticità obbligatorie

- Per poter essere ammessi all' esame di Basi di Dati bisogna aver superato :
 - **Matematica Discreta**
 - **Programmazione dei calcolatori con laboratorio**

Modalità d' esame

- Prova scritta
- Consegna di un progetto completo di Database
 - Solo se si supera la prova scritta
 - Seguendo le **linee guida** di progetto
- Prova orale
 - Solo se si supera la prova scritta e il progetto viene accettato

Programma del corso di Basi di dati - I parte

- Introduzione, Storia e Definizioni
- Algebra relazionale e cenni di Calcolo relazionale
- Flusso di progetto e visione dei dati
 - Dai concetti allo schema fisico
- Modello concettuale dei dati
 - Entità, relazioni, cardinalità, chiavi e attributi

Programma del corso di Basi di dati - I parte (2)

- Disegno logico DB
 - Schema Entity-Relationship ed esempi.
- Disegno fisico DB
 - Integrità, congruenza, consistenza e non ridondanza
- Forme normali
 - Le prime 5 forme normali
 - Dipendenze funzionali

Programma del corso

Basi di dati - I parte (3)

- Query language e implementazioni su MySQL
 - Creazioni tabelle e DML
 - Creazione indici e integrità semantica
 - SQL
 - Select
 - Opzioni, confronti, appartenenza, operatori ed ordinamenti
 - Join, equijoin e nonequijoin
 - Select nidificate e alias
 - Funzioni aritmetiche, di insieme, di stringa
 - Viste e gestione Sicurezza
- Realizzazione progetto (esame)

Programma del corso

Basi di dati - II parte

- Introduzione
- Organizzazione fisica dei dati
- Ottimizzazione degli indici
- Normalizzazione e Denormalizzazione
- Ottimizzazione delle interrogazioni
- Ottimizzazioni in MySQL

Programma del corso

Basi di dati - II parte (2)

- Transazioni
 - Proprietà ‘acide’
 - Controllo della concorrenza
 - 🕒 Locking a due fasi
 - MySQL e Storage Engine
 - MySQL e ottimizzazioni
- Basi di dati attive
 - Trigger e Stored Procedure
 - Cursor, Trigger e Stored Procedure in MySQL
- Altre caratteristiche per le performance di MySQL

Programma del corso

Basi di dati - II parte (3)

- Basi di dati geospaziali e GIS
- Basi di dati su architetture distribuite
 - Architettura client-server
 - Basi di dati distribuite
 - Commit a due fasi
 - Parallelismo
 - Basi di dati replicate

Programma del corso

Basi di dati - II parte (4)

- Database NoSQL e test su MongoDB/MySQL8
- Architetture per l'analisi dei dati
 - Data Warehouse
 - Data Mining

Libri di testo

I parte

- Atzeni, Ceri, Fraternali, Paraboschi, Torlone
“**Basi di dati – Modelli e Linguaggi di interrogazione**”,
McGraw-Hill **4th**
edition.



Libri di testo

Il parte

Elmasri R., Navathe S.
“**Sistemi di basi di dati –
Fondamenti e
Complementi**” ed.
Pearson **7th edition**



Definizione

- Nei sistemi informatici (e non solo) tutte le informazioni vengono rappresentate per mezzo di **dati**
- I sistemi informativi esistono da molto prima della diffusione dei calcolatori elettronici

Definizione

- **Base di dati o Database**
 - Insieme (organizzato) di tutte le informazioni utili presenti in un luogo (azienda, università, mente di qualcuno, ecc.)
- **Data independence**
 - La struttura di un DB deve dare garanzia che modifiche dei dati non richiedano modifiche ai programmi applicativi e/o alle tecniche di accesso ai dati stessi

Definizione

- **Base di dati o Database**
 - Collezione di informazioni registrate in formato leggibile dall' elaboratore elettronico e relativa ad un preciso dominio di conoscenze (azienda, università, mente di qualcuno, ecc.), organizzata allo scopo di poter essere consultata dai suoi utilizzatori.
(Brunella Longo - 1993)

Definizione

- Un **Modello dei dati** è un insieme di concetti utilizzati per organizzare i dati di interesse e descriverne la struttura in modo che essa risulti comprensibile ad un elaboratore (e non solo...).

Definizione di Ullman (2)

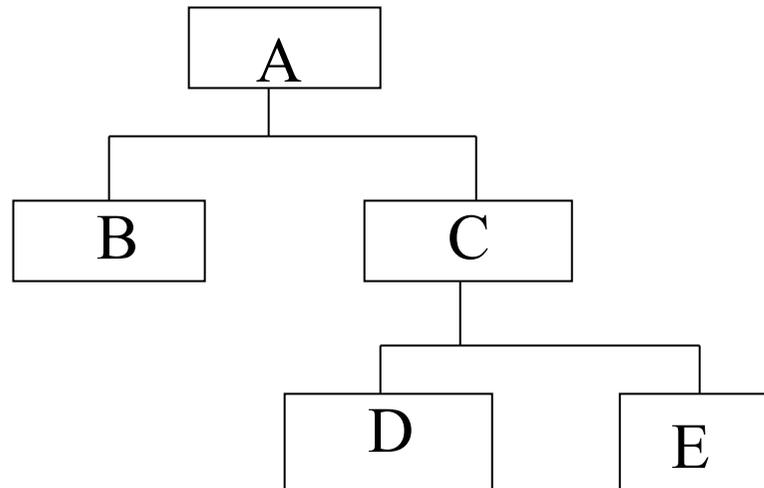
- o anche
- Un **Modello di dati** è un formalismo matematico composto da :
 - una notazione per descrivere i dati
 - un insieme di operazioni per manipolare tali dati

Storia dei database e dei Modelli di dati (1)

- Prima del 1960
 - Tanta, tanta, tanta carta e altro
- Anni '60
 - Computer
 - Due modelli di dati che dipendono dalla struttura fisica
 - Gerarchico
 - Reticolare (CODASYL)

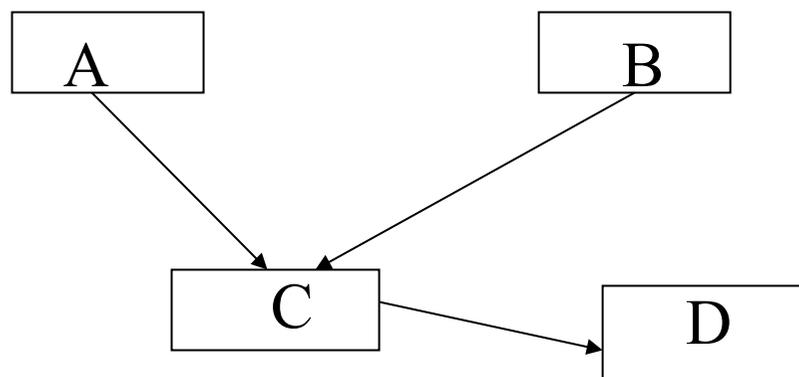
Storia dei database e dei Modelli di dati (2)

Gerarchico



Storia dei database e dei Modelli di dati (3)

Reticolare

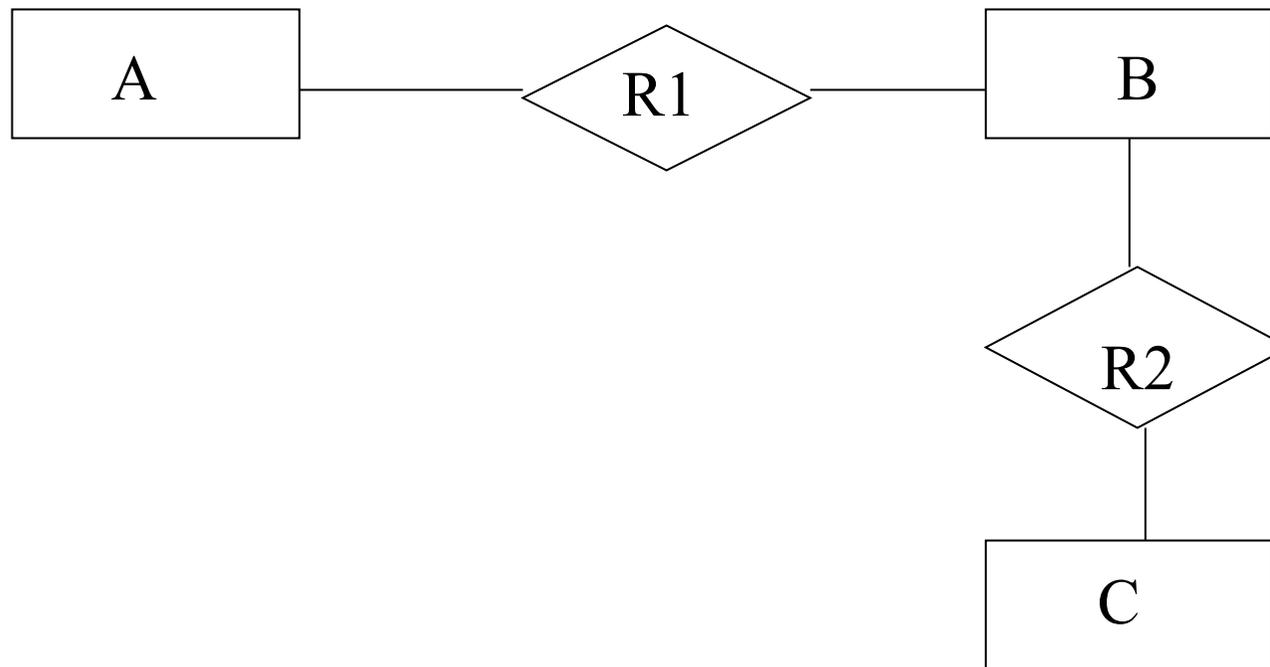


Storia dei database e dei Modelli di dati (4)

- Anni '60
 - Crescono un po' le capacità di memoria, ma tutti si concentrano a rendere efficienti i programmi di accesso ai dati.
- Anni '70 - '72
 - Edgar F. Codd separa l'organizzazione logica dai metodi di memorizzazione fisica

Storia dei database e dei Modelli di dati (5)

Relazionale E.F.Codd anni '70



Storia dei database e dei Modelli di dati (6)

- Anni '70 - '72
 - Codd rivoluziona il concetto di migliorare le prestazioni **NON** dalla programmazione ma con l'organizzazione dei dati.
 - Semplici tabelle
 - Il **modello relazionale** basato sui valori ha successo anche per le aumentate capacità di memoria.

Storia dei database e dei Modelli di dati (7)

- Dal 1973 primi prototipi di società
 - UCB di Berkeley --> INGRES porterà a Ingres Corp., Sybase, MS/SQL Server. Usava QUEL.
 - Relational Software Inc. di Larry Ellison --> un primo prodotto nel '77 o '79 porterà a Oracle
 - IBM --> progetto System R porterà a SQL/DS & DB2, AllBase di HP. Usava SEQUEL --> poi SEQUEL/2, poi SQL

Storia dei database e dei Modelli di dati (8)

- Dal 1973 primi prototipi di società
 - **System R** in particolare sviluppò diverse tecnologie
 - Structured Query Language (SQL) di Donald Chamberlin e Ray Boyce
 - Ottimizzatore di query di Pat Selinger
 - Compilatore di query di Raymond Lorie
 - La possibilità di ridefinire dati online.
- Boyce lavora con Codd sulla BCNF.
Viene coniato il termine RDBMS.

Storia dei database e dei Modelli di dati (9)

- 1976
 - Peter Chen propone il suo **Entity-Relationship Model for Database Design** (schemi E-R), altro importante passo in avanti per i modelli concettuali di dati.

Storia dei database e dei Modelli di dati (10)

- Anni '80

- Boom commercializzazione computer
- Primi DBMS relazionali
 - '77-' 79 Relational Software Inc.
 - 1981 IBM annuncia SQL/DS
 - 1983 IBM annuncia DB2

- Metà anni '80

- **SQL** come “standard intergalattico”
- Avvento PC  Rbase 5000, Paradox, OS/2, DbaseIII, DbaseIV.

Storia dei database e dei Modelli di dati (11)

- Primi anni '90
 - Grosse applicazione - grandi prezzi : Sybase, Oracle
 - Modello client-server
 - Inizi prototipi ODBMS
- Metà anni '90
 - Boom di Internet/WWW
 - Web/DB cresce esponenzialmente

Storia dei database e dei Modelli di dati (12)

- Tardi anni '90
 - Connettori Web/Internet/DB (Java Servlets, JDBC, Oracle Developer 2000)
 - Appare l' Open Source (Apache, MySQL)
- Inizio 21° secolo
 - le compagnie presenti sul mercato : IBM (compra Informix), Microsoft, Oracle, (Sun compra MySQL).

Storia dei database e dei Modelli di dati (13)

- Futuro

- Terabyte ????
- Datawarehousing, data mining e data mart
- Standard per i successori di SQL (fallimento attuale) ?
- La “next great thing” --> XML con Java per Database?
- “Object Oriented Everything” per Database
- Linked Open Data (Tim Berners-Lee) e Big Data

... vai a Modello relazionale

2 punti di vista

- Come costruisco un DB ?



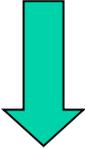
Flusso di progetto

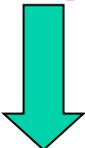
- Cosa vedo ?



Visione dei dati

Flusso di progetto di un database

- **Analisi dei requisiti** (definizione environment)

- **Modello concettuale DB** (Oggetti)

- **Disegno logico DB** (schema E-R, Yourdon)

- **Disegno fisico DB** (normalizzazione tabelle)

Strati della conoscenza

- Vista utente



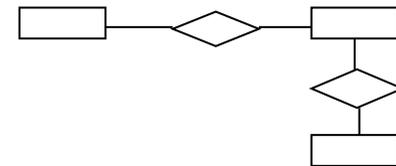
- Schema esterno : TABELLE



- Vista amministratore DB o azienda



- Schema concettuale : Schema E-R o RM/T



- Vista fisica

- Schema interno : catene, hash, indici

Organizzazione fisica dei dati

- L' organizzazione fisica dei dati di un database deve essere **efficiente**, ciò significa che il sistema di gestione di un database (DBMS) deve avere la capacità di rispondere alle richieste dell' utente il più velocemente possibile.

Organizzazione fisica dei dati (2)

- I dati sono organizzati in **file**
 - I file sono divisi in **record**
 - **Record logici**, quelli visibili
 - **Record fisici**, con informazioni del record e campi

INFORMAZIONI	CAMPI
---------------------	--------------

Organizzazione fisica dei dati (3)

- **INFORMAZIONI**
 - **Puntatore** al prossimo record
 - Tipo record
 - Lunghezza record
 - Bit di cancellazione
 - Eventuali Offset dei campi

Un **puntatore** è una coppia (b,k)

Organizzazione fisica dei dati (4)

- CAMPI

- Eventuale offset del campo
- Campo vero e proprio

Organizzazione fisica dei dati (5)

Tipi di file

- File Heap
 - Record inseriti nei blocchi senza ordine
 - Accesso ai record tramite directory di puntatori ai blocchi
 - Proliferazione di indirizzi indiretti

Organizzazione fisica dei dati (6)

Tipi di file

- File Hash
 - Record ripartiti in ‘bucket’ in base al valore della chiave
 - Ogni bucket ha 1 o più blocchi ed è organizzato come un Heap
 - Funzione Hash “buona”
 - Puntatori a lista collegano i blocchi di un bucket

Organizzazione fisica dei dati (7)

Tipi di file

- File con indice (sparso) ISAM
 - ISAM (indexed sequential access method)
 - Record ordinati in crescita (lasciando il 20% dei blocchi liberi) e non puntati
 - File indice formato da coppie (v,b)
 - Directory del file indice
 - Metodi di scansione
 - Ricerca binaria o dicotomica
 - Ricerca per interpolazione

Organizzazione fisica dei dati (8)

Tipi di file

- File B-tree
 - Generalizzazione del file con indice
 - Concetto dell' albero binario evita spreco di spazio