

Capitolo 1

Introduzione ai sistemi di basi di dati

Copyright © 2004 – The McGraw-Hill Companies srl

Che cos'è un DBMS?

- Una collezione integrata molto grande di dati
- Modella organizzazioni del mondo reale
 - Entità (ad esempio studenti, corsi)
 - Relazioni (ad esempio, Madonna segue il corso CS564)
- Un Database Management System (DBMS) è un pacchetto software progettato per memorizzare e gestire basi di dati

Vedi "Pagina Note"

Copyright © 2004 – The McGraw-Hill Companies srl

DBMS

- Prodotti software (complessi) disponibili sul mercato; esempi:
 - Proprietari: Access, DB2, Oracle, Informix, SQLServer, ...
 - Open source: MySQL, Postgres, SAP/DB, ...

Copyright © 2004 – The McGraw-Hill Companies srl

Perché usare un DBMS?

- Indipendenza dei dati e accesso efficiente
- Tempo ridotto di sviluppo dell'applicazione
- Integrità dei dati e sicurezza
- Amministrazione dei dati uniforme
- Accesso concorrente, ripristino da *crash*

Copyright © 2004 – The McGraw-Hill Companies srl

Descrizioni dei dati nei DBMS

- Rappresentazioni dei dati a livelli diversi
 - permettono l'*indipendenza dei dati* dalla rappresentazione fisica:
 - i programmi fanno riferimento alla struttura a livello più alto, e le rappresentazioni sottostanti possono essere modificate senza necessità di modifica dei programmi
 - Precisiamo attraverso il concetto di *modello dei dati*

Copyright © 2004 – The McGraw-Hill Companies srl

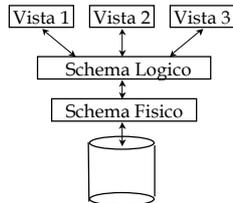
Modelli di dati

- Un modello di dati è una collezione di concetti per la descrizione dei dati
- Uno schema è una descrizione di una particolare collezione di dati, che fa uso del modello di dati fornito
- Il modello di dati relazionale è il modello oggi più usato
 - Concetto chiave: relazione, fondamentalmente una tabella con righe e colonne
 - Ogni relazione ha uno schema, che descrive le colonne, o campi

Copyright © 2004 – The McGraw-Hill Companies srl

Livelli di astrazione

- Molte viste, un singolo schema logico (concettuale) e uno schema fisico
 - Le viste descrivono i dati come vengono visti dagli utenti
 - Lo schema logico definisce la struttura logica
 - Lo schema fisico descrive i file e gli indici usati



Copyright © 2004 – The McGraw-Hill Companies srl

Esempio: La base di dati di una Università

- Schema logico:
 - Studenti(sid:string, nome:string, login: string, età:integer, media:real)
 - Corsi(cid:string, cnome:string, crediti:integer)
 - Iscritto(sid:string, cid:string, voto:string)
- Schema fisico:
 - Le relazioni sono memorizzate come file non ordinati
 - Indice sulla prima colonna di Studenti
- Schema esterno (vista):
 - info_corso(cid:string, iscritti:integer)

Copyright © 2004 – The McGraw-Hill Companies srl

Indipendenza dei dati*

- Le applicazioni sono separate dalla struttura e dalla memorizzazione dei dati
- Indipendenza logica dei dati: protezione dalle modifiche alla struttura logica dei dati
- Indipendenza fisica dei dati: protezione dalle modifiche alla struttura fisica dei dati

* Uno dei più importanti benefici dell'uso di un DBMS!

Copyright © 2004 – The McGraw-Hill Companies srl

Le basi di dati sono ... grandi

- dimensioni (molto) maggiori della memoria centrale dei sistemi di calcolo utilizzati
- il limite deve essere solo quello fisico dei dispositivi



Copyright © 2004 – The McGraw-Hill Companies srl

Le basi di dati sono ... persistenti

- hanno un tempo di vita indipendente dalle singole esecuzioni dei programmi che le utilizzano



Copyright © 2004 – The McGraw-Hill Companies srl

Le basi di dati sono ... condivise

- Ogni organizzazione (specie se grande) è divisa in settori o comunque svolge diverse attività
- Ciascun settore/attività ha un (sotto)sistema informativo (non necessariamente disgiunto)

Copyright © 2004 – The McGraw-Hill Companies srl

UNIVERSITA' DEGLI STUDI DI CHISSADOVE

Corso di Studi in Ingegneria Informatica

ORARIO DELLE LEZIONI PER L'ANNO ACCADEMICO 1999-2000

INSEGNAMENTO	Docente	Aula	Orario
Analisi matematica I	Luigi Neri	N1	8:00-9:30
Basi di dati	Piero Rossi	N2	9:45-11:15
Chimica	Nicola Mori	N1	9:45-11:30
Fisica I	Mario Bruni	N1	11:45-13:00
Fisica II	Mario Bruni	N3	9:45-11:15
Sistemi informativi	Piero Rossi	N3	8:00-9:30

Copyright © 2004 – The McGraw-Hill Companies srl

UNIVERSITA' DEGLI STUDI CHISSADOVE

Corso di Studi in Ingegneria Informatica

Orario di ricevimento dei docenti

DOCENTE	INSEGNAMENTI	ORARIO
Mario BRUNI	Fisica I Fisica II	Martedì 10-12
Luigi NERI	Analisi matematica I	Lunedì 12-13
Piero ROSSI	Basi di dati Sistemi informativi	Giovedì 11-13
Nicola MORI	Chimica	Martedì 16-18

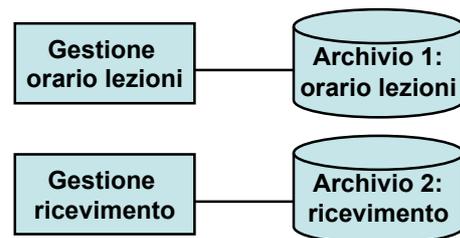
Copyright © 2004 – The McGraw-Hill Companies srl

Problemi

- **Ridondanza:**
 - informazioni ripetute
- **Rischio di incoerenza:**
 - le versioni possono non coincidere

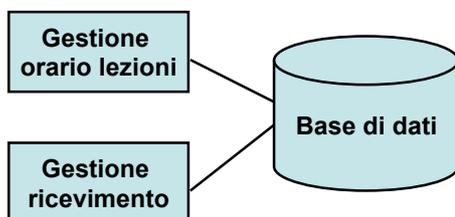
Copyright © 2004 – The McGraw-Hill Companies srl

Archivi e basi di dati



Copyright © 2004 – The McGraw-Hill Companies srl

Archivi e basi di dati



Copyright © 2004 – The McGraw-Hill Companies srl

Le basi di dati sono condivise

- Una base di dati è una risorsa **integrata, condivisa** fra applicazioni
- conseguenze
 - Attività diverse su dati condivisi:
 - meccanismi di **autorizzazione**
 - Accessi di più utenti ai dati condivisi:
 - controllo della **concorrenza**



Copyright © 2004 – The McGraw-Hill Companies srl

I DBMS garantiscono ... privatezza

- Si possono definire meccanismi di autorizzazione
 - l'utente A è autorizzato a leggere tutti i dati e a modificare quelli sul ricevimento
 - l'utente B è autorizzato a leggere i X e a modificare Y



Copyright © 2004 – The McGraw-Hill Companies srl

I DBMS garantiscono...affidabilità

- **Affidabilità** (per le basi di dati):
 - resistenza a malfunzionamenti hardware e software
- Una base di dati è una risorsa pregiata e quindi deve essere conservata a lungo termine
- Tecnica fondamentale:
 - gestione delle **transazioni**

Copyright © 2004 – The McGraw-Hill Companies srl

Transazione

- Insieme di operazioni da considerare indivisibile ("**atomico**"), corretto anche in presenza di **concorrenza** e con effetti **definitivi**

Copyright © 2004 – The McGraw-Hill Companies srl

Le transazioni sono ... atomiche

- La sequenza di operazioni (lettura/scrittura) sulla base di dati viene eseguita per intero o per niente:
 - trasferimento di fondi da un conto A ad un conto B: o si fanno il prelievo da A e il versamento su B o nessuno dei due

Copyright © 2004 – The McGraw-Hill Companies srl

Transazione: l'esecuzione di una azione sulla base di dati

- Ciascuna transazione, eseguita completamente, deve lasciare la base di dati in uno stato consistente se esso era consistente quando la transazione ha avuto inizio
 - Gli utenti possono specificare alcuni semplici vincoli di integrità sui dati, e il DBMS garantirà tali vincoli
 - Al di là di questo, il DBMS non capisce realmente la semantica dei dati (ad esempio non capisce come sono calcolati gli interessi su un conto bancario)
 - Quindi, garantire che una transazione (eseguita da sola) conservi la consistenza dei dati è, in ultima analisi, responsabilità dell'utente!

Copyright © 2004 – The McGraw-Hill Companies srl

Le transazioni sono ... concorrenti

- Le azioni interallacciate di diversi programmi utente possono portare a inconsistenza
 - se due assegni emessi sullo stesso conto corrente vengono incassati contemporaneamente si deve evitare di trascurarne uno
 - un assegno viene pagato mentre viene calcolato il bilancio del conto corrente
- L'effetto di transazioni concorrenti deve essere coerente (ad esempio "equivalente" all'esecuzione separata)

Copyright © 2004 – The McGraw-Hill Companies srl

Controllo di concorrenza

- L'esecuzione concorrente di programmi utente è essenziale per ottenere buone prestazioni dal DBMS
 - Poiché gli accessi al disco sono frequenti, e relativamente lenti, è importante tenere occupata la CPU lavorando su diversi programmi utente concorrentemente
- Il DBMS garantisce che tali problemi non si presentino: ogni utente può immaginare di essere l'unico utente del sistema

Copyright © 2004 – The McGraw-Hill Companies srl

I risultati delle transazioni sono permanenti

- La conclusione positiva di una transazione corrisponde ad un impegno (in inglese **commit**) a mantenere traccia del risultato in modo definitivo, anche in presenza di guasti e di esecuzione concorrente

Copyright © 2004 – The McGraw-Hill Companies srl

Garantire l'atomicità

- Il DBMS garantisce l'atomicità (proprietà tutto-o-niente) anche se il sistema va in crash durante una transazione
- Idea: mantenere un log (una storia) di tutte le azioni effettuate dal DBMS nell'eseguire un insieme di transazioni:
 - prima che una modifica sia fatta alla base di dati, la voce corrispondente del log viene messa al sicuro (protocollo WAL: spesso il supporto del sistema operativo in questo ambito è inadeguato)
 - dopo un crash, gli effetti delle transazioni eseguite parzialmente sono annullati usando il log (grazie al WAL, se una voce del log non era stata salvata prima del crash, le corrispondenti modifiche alla base di dati non erano state apportate!)

Copyright © 2004 – The McGraw-Hill Companies srl

I DBMS debbono essere...efficienti

- Cercano di utilizzando al meglio le risorse di spazio di memoria (principale e secondaria) e tempo (di esecuzione e di risposta)
- I DBMS, con tante funzioni, rischiano l'inefficienza e per questo ci sono grandi investimenti e competizione
- L'efficienza è anche il risultato della qualità delle applicazioni



Copyright © 2004 – The McGraw-Hill Companies srl

I DBMS debbono essere...efficaci

- Cercano di rendere produttive le attività dei loro utilizzatori, offrendo funzionalità articolate, potenti e flessibili:
 - il corso è in buona parte dedicato ad illustrare come i DBMS perseguono l'efficacia

Copyright © 2004 – The McGraw-Hill Companies srl

DBMS vs file system

- La gestione di insiemi di dati grandi e persistenti è possibile anche attraverso sistemi più semplici — gli ordinari **file system** dei sistemi operativi
- I file system prevedono forme rudimentali di condivisione: "tutto o niente"
- I DBMS estendono le funzionalità dei file system, fornendo più servizi ed in maniera integrata

Copyright © 2004 – The McGraw-Hill Companies srl

DBMS vs file system (2)

- Nei programmi tradizionali che accedono a file, ogni programma contiene una descrizione della struttura del file stesso, con i conseguenti rischi di incoerenza fra le descrizioni (ripetute in ciascun programma) e i file stessi
- Nei DBMS, esiste una porzione della base di dati (il [catalogo](#) o [dizionario](#)) che contiene una descrizione centralizzata dei dati, che può essere utilizzata dai vari programmi

Copyright © 2004 – The McGraw-Hill Companies srl

Personaggi e interpreti

- [progettisti](#) e realizzatori di [DBMS](#)
- [progettisti della base di dati](#) e amministratori della base di dati ([DBA](#))
- [progettisti](#) e programmatori di [applicazioni](#) (ad esempio programmatori di siti web)
- [utenti](#)
 - [utenti finali](#) (terminalisti): eseguono applicazioni predefinite ([transazioni](#))
 - [utenti casuali](#): eseguono operazioni non previste a priori, usando linguaggi interattivi

Copyright © 2004 – The McGraw-Hill Companies srl

Database administrator (DBA)

- Persona o gruppo di persone responsabile del controllo centralizzato e della gestione del sistema, delle prestazioni, dell'affidabilità, delle autorizzazioni
- Le funzioni del DBA includono quelle di progettazione, anche se in progetti complessi ci possono essere distinzioni



Copyright © 2004 – The McGraw-Hill Companies srl

Transazioni (per l'utente)

- Programmi che realizzano attività frequenti e predefinite, con poche eccezioni, previste a priori.
- Esempi:
 - versamento presso uno sportello bancario
 - emissione di certificato anagrafico
 - dichiarazione presso l'ufficio di stato civile
 - prenotazione aerea
- Le transazioni sono di solito realizzate in linguaggio ospite (tradizionale o ad hoc)

Copyright © 2004 – The McGraw-Hill Companies srl

Transazioni, due accezioni

- Per l'utente:
 - programma a disposizione, da eseguire per realizzare una funzione di interesse
- Per il sistema:
 - sequenza indivisibile di operazioni (cfr. [affidabilità](#))



Copyright © 2004 – The McGraw-Hill Companies srl

Linguaggi per basi di dati

- Un altro contributo all'efficacia: disponibilità di vari linguaggi e interfacce
 - ↳ linguaggi testuali interattivi ([SQL](#))
 - ↳ comandi (SQL) immersi in un linguaggio [ospite](#) (Pascal, Java, C ...)
 - ↳ comandi (SQL) immersi in un linguaggio ad hoc, con anche altre funzionalità (p.es. per grafici o stampe strutturate)
 - ↳ con interfacce amichevoli (senza linguaggio testuale)

Copyright © 2004 – The McGraw-Hill Companies srl

SQL, un linguaggio interattivo

- "Trovare i corsi tenuti in aule a piano terra"

Corsi

Corso	Docente	Aula
Basi di dati	Rossi	DS3
Sistemi	Neri	N3
Reti	Bruni	N3
Controlli	Bruni	G

Aule

Nome	Edificio	Piano
DS1	OMI	Terra
N3	OMI	Terra
G	Pincherle	Primo

Copyright © 2004 – The McGraw-Hill Companies srl

SQL, un linguaggio interattivo

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

Corso	Aula	Piano
Sistemi	N3	Terra
Reti	N3	Terra

Copyright © 2004 – The McGraw-Hill Companies srl

SQL immerso in linguaggio ospite

```
write('nome della città?'); 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
```



Copyright © 2004 – The McGraw-Hill Companies srl

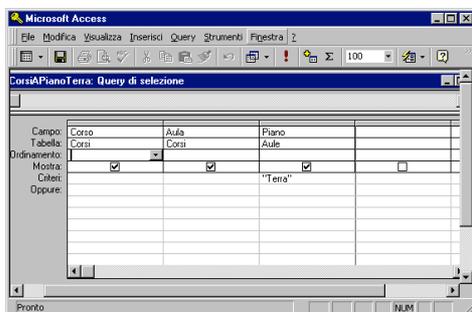
SQL in linguaggio ad hoc (Oracle PL/SQL)

```
declare Stip number;
begin
select Stipendio into Stip from Impiegato
where Matricola = '575488' for update of Stipendio;
if Stip > 30 then
update Impiegato set Stipendio = Stipendio * 1.1
where Matricola = '575488';
else
update Impiegato set Stipendio = Stipendio * 1.15
where Matricola = '575488';
end if;
commit;
exception
when no_data_found then
insert into Errori values('Matricola inesistente',sysdate);
end;
```



Copyright © 2004 – The McGraw-Hill Companies srl

Interazione non testuale (Access)



Copyright © 2004 – The McGraw-Hill Companies srl

Una distinzione terminologica (separazione fra dati e programmi)

data manipulation language (DML)
per l'interrogazione e l'aggiornamento di (istanze di) basi di dati

data definition language (DDL)

per la definizione di schemi (logici, esterni, fisici) e altre operazioni generali

Copyright © 2004 – The McGraw-Hill Companies srl

Un'operazione DDL (sullo schema)

```
CREATE TABLE orario (  
  insegnamento CHAR(20) ,  
  docente        CHAR(20) ,  
  aula           CHAR(4)  ,  
  ora            CHAR(5)  )
```

Copyright © 2004 – The McGraw-Hill Companies srl

Vantaggi e svantaggi dei DBMS, 1

Pro

- dati come risorsa comune, base di dati come modello della realtà
- gestione centralizzata con possibilità di standardizzazione ed "economia di scala"
- disponibilità di servizi integrati
- riduzione di ridondanze e inconsistenze
- indipendenza dei dati (favorisce lo sviluppo e la manutenzione delle applicazioni)

Copyright © 2004 – The McGraw-Hill Companies srl

Vantaggi e svantaggi dei DBMS, 2

Contro

- costo dei prodotti e della transizione verso di essi
- Necessità di acquisire le competenze di progettazione e amministrazione di un sistema complesso
- non scorporabilità delle funzionalità (con riduzione di efficienza)

Copyright © 2004 – The McGraw-Hill Companies srl

Sommario

- Un DBMS è usato per mantenere e interrogare grandi insiemi di dati
- Tra i benefici, il ripristino dai crash del sistema, l'accesso concorrente, il rapido sviluppo di applicazioni, l'integrità dei dati e la sicurezza
- I livelli di astrazione portano all'indipendenza dei dati
- I DBA svolgono un lavoro di responsabilità e ben pagato!

Copyright © 2004 – The McGraw-Hill Companies srl