Database Management Systems

CSE594

Instructor: Scott Vandenberg
vandenbe@cs.washington.edu
University of Washington

What Is a DBMS?

- A very large, integrated collection of data.
- Models real-world *enterprise.*
  - Entities (e.g., students, courses)
  - Relationships (e.g., Ken Griffey is taking CSE594)
- A *Database Management System (DBMS)* is a software package designed to store and manage databases easily and efficiently.

Why Use a DBMS?

- Data independence and efficient access.
- Reduced application development time.
- Data integrity and security.
- Uniform data administration and access.
- Concurrent access, recovery from crashes.

Why Study Databases??

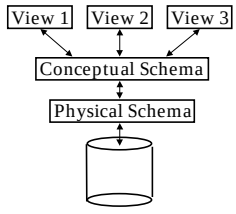
- Shift from *computation* to *information*
  - at the "low end": scramble to webspace (a mess!)
  - at the "high end": scientific applications
- Datasets increasing in diversity and volume.
  - Digital libraries, distance learning, Human Genome project, EOS project
  - ... need for DBMS exploding
- DBMS encompasses most of CS
  - OS, languages, theory, AI, multimedia, logic

Data Models

- A *data model* is a collection of concepts for describing data.
- A *schema* is a description of a particular collection of data, using the a given data model.
- The *relational model of data* is the most widely used model today.
  - Main concept: *relation*, basically a table with rows and columns.
  - Every relation has a *schema*, which describes the columns, or fields.

Levels of Abstraction

- Many *views*, single *conceptual (logical) schema* and *physical schema*.
  - Views describe how users see the data.
  - Conceptual schema defines logical structure using a data model
  - Physical schema describes the files and indices used.



☛ *Schemas are defined using DDL; data is modified/queried using DML.*

Example: University Database

- A Conceptual schema:
  - *Students(sid: string, name: string, login: string, age: integer, gpa:real)*
  - *Courses(cid: string, cname:string, credits:integer)*
  - *Enrolled(sid:string, cid:string, grade:string)*
- A possible Physical Schema:
  - Relations stored as unordered files.
  - Index on first column of Students.
- An External Schema (View):
  - *Course_info(cid:string,enrollment:integer)*

Data Independence

- Applications insulated from how data is structured and stored.
- *Logical data independence*: Protects views from changes in *logical* (conceptual) structure of data.
- *Physical data independence*: Protects conceptual schema from changes in *physical* structure of data.

☛ *One of the most important benefits of using a DBMS!*

Concurrency Control and Recovery

- Concurrent execution of user programs is essential for good DBMS performance.
  - Because disk accesses are frequent, and relatively slow, it is important to keep the cpu humming by working on several user programs concurrently.
- Interleaving actions of different user programs can lead to inconsistency: e.g., check is cleared while account balance is being computed.
- DBMS ensures such problems don't arise: users can pretend they are using a single-user system.

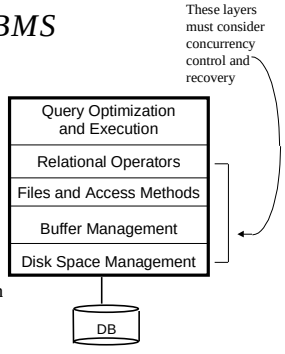
Databases make these folks happy ...

- DBMS implementers (???)
- End users and DBMS vendors
- DB application programmers
  - E.g. smart webmasters
- *Database administrator (DBA)*
  - Designs logical / physical schemas
  - Handles security and authorization
  - Data availability, crash recovery
  - Database tuning as needs evolve

Must understand how a DBMS works!

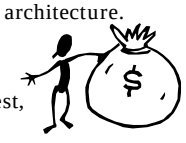
Structure of a DBMS

- A typical DBMS has a layered architecture.
- The figure does not show the concurrency control and recovery components.
- This is one of several possible architectures; each system has its own variations.



Summary

- DBMS used to maintain, query large datasets.
- Benefits include recovery from system crashes, concurrent access, quick application development, data integrity, and security.
- Levels of abstraction give data independence.
- A DBMS typically has a layered architecture.
- DBAs hold responsible jobs and are well-paid!
- DBMS R&D is one of the broadest, most exciting areas in CS.



IMPORTANT:

- Email majordomo@cs.washington.edu
  - subject line is always ignored by majordomo
  - body of message: subscribe cse594
- Email vandenbe@cs and tzoompy@cs:
  - name
  - workplace
  - email address
  - DB experience (academic and work)
  - lecture location (UW, Intel, Microsoft)
  - what you've done to prepare for Y2K

Important stuff, continued ...

- Get SQL Server 6.5 or 7.0
  - via your workplace (free)
  - use on-campus only (free) (v 7.0)
  - from us (free) (v 6.5, runs only on WinNT)
    - MUST email vandenbe@cs by Thursday 9/30/99
  - get "Special Edition Using Microsoft SQL Server 7.0", S. Wynkoop, Que Publishing, 1999 (comes with v 7.0 (runs on WinNT, Win9x)
  - via other means