COMP.3090/3100
Database I & II

Slides adapted from http://infolab.stanford.edu/~ullman/fcdb.html

Prof. Cindy Chen
cchen@cs.uml.edu
September 5, 2019

Textbook

- Required: *First Course in Database Systems, 3rd ed.*, by Jeffrey D. Ullman and Jennifer Widom
- Optional: *Database System Concepts, 6th ed.*, by Abraham Silberschatz, Henry F. Korth, and S. Sudarshan

Course Requirements

1. Homework: 10 sets, 2% each
2. Project: 10%
   - Two-person team
   - Use MS Access in COMP.3090 and MySQL/PHP in COMP.3100
3. Class Participation: 10%
4. Midterm: 20%
5. Final: 40%, covers everything discussed in the semester

Midterm and final are closed book exams. No make up exams for midterm and final will be offered. 3 of 13

Why do you take this class?

Choose all that applies.

A. I'm interested in database ---
   1. I want to be a database analyst/administrator.
   2. I want to develop better/faster database systems.
B. My friend has taken/is taking this course.
C. This is the only class that fits into my schedule.
D. I don't know.

4 of 13

The Course Pair

- COMP.3090 Database I
  - What everybody needs to know about databases – how to be an effective user of databases
    - Understand basics of database theory
    - Write SQL queries
- COMP.3100 Database II
  - What you need to know to be a database guru and make big bucks
    - Design and implement a web-based database
    - Create web interface for the database

Content of Database I & II

- Design of databases.
  - Relational model, E/R model, etc.
- Database programming.
  - Relational algebra, SQL, JDBC/ODBC, etc.
- Not DBMS implementation (that's COMP.5730 & COMP.5740).

What is a database?

- A database is a very large, integrated collection of data.
- A database management system (DBMS) is a software package designed to store and manage data efficiently, e.g., MS Access, MySQL, IBM DB2, Oracle, MS SQL Server.

A sample database

customer_id	*customer_name*	*customer_street*	*customer_city*
192-83-7465	Johnson	12 Alma St.	Palo Alto
677-89-9011	Hayes	3 Main St.	Harrison
182-73-6091	Turner	123 Putnam Ave.	Stamford
321-12-3123	Jones	100 Main St.	Harrison
336-66-9999	Lindsay	175 Park Ave.	Pittsfield
019-28-3746	Smith	72 North St.	Rye

(a) The *customer* table

account_number	*balance*
A-101	500
A-215	700
A-102	400
A-305	350
A-201	900
A-217	750
A-222	700

(b) The *account* table

customer_id	*account_number*
192-83-7465	A-101
192-83-7465	A-201
019-28-3746	A-215
677-89-9011	A-102
182-73-6091	A-305
321-12-3123	A-217
336-66-9999	A-222
019-28-3746	A-201

(c) The *depositor* table

Interesting Stuff About Databases

- It used to be about boring stuff: employee records, bank records, etc.
- Today, databases are behind almost everything you do on the Web.
  - Google searches.
  - Orders at Amazon, eBay, Paypal, etc.
  - Posts on Facebook, Instagram, LinkedIn, Snapchat, Twitter, etc.

9 of 13

More Interesting Stuff

- Database programming centers around limited programming languages.
  - Only area where non-Turing-complete languages make sense.
  - Leads to very succinct programming, but also to unique query-optimization problems.

10of 13

Databases vs File Systems

- Drawbacks of using file systems to store data:
  - Data redundancy and inconsistency
    - Multiple file formats, duplication of information in different files
  - Data isolation — multiple files and formats
  - Difficulty in accessing data
    - Need to write a new program to carry out each new task
  - Integrity problems
    - Integrity constraints (e.g., account balance>=0) become "buried" in program code rather than being stated explicitly
    - Hard to add new constraints or change existing ones

Databases vs File Systems

- Drawbacks of using file systems (cont.)
  - Atomicity of updates
    - Failures may leave database in an inconsistent state with partial updates carried out
    - Example: Transfer of funds from one account to another should either complete or not happen at all
  - Concurrent access by multiple users
    - Concurrent access needed for performance
    - Uncontrolled concurrent accesses can lead to inconsistencies
      - Example: Two people reading a balance (say 100) and updating it by withdrawing money (say 50 each) at the same time
  - Security problems
    - Hard to provide user access to some, but not all, data

Databases vs File Systems

- Database systems offer solutions to all the file system problems by providing:
  - Efficient storage
  - Ease of use
  - Integrity check
  - Atomicity of updates
  - Concurrent access by multiple users
  - Security/access control
- Database systems scale up well to handle petabytes of data