PURDUE
UNIVERSITY

CS34800
Information Systems

Course Overview

Prof. Chris Clifton

22 August, 2016

Why this Course?

- Managing Data is one of the primary uses of computers
- This course covers the foundations of organized data management
  - Database Management Systems (DBMS) of various flavors
- Learn how to
  - Model data in ways that protect data integrity
  - Store, retrieve, and manipulate data
  - Perform basic data analysis on large datasets

Course Outline

- Relational Databases
  1. Relational model overview
  2. Formal definitions, relational operations
  3. Query: SQL
- Database Design
  4. Entity-Relationship Model
  5. Relational Design
  6. Database Normalization
  7. Object Databases
  8. XML databases
- Integrity and Consistency
  9. Transactions
  10. Concurrency
  11. Constraints
- Advanced Topics
  12. Big Data: MapReduce, Hadoop, Spark
  13. Data Analysis / Data Mining
  14. Information Retrieval

Course Information

- Contact Information: Professor Clifton
  - Office: LWSN 2142F, x4-6005.
  - Office hours: TBD, for now generally 8-5
  - clifton@cs.purdue.edu
- Teaching Assistants:
  - Romila Pradhan
  - Denis Ulybyshev
  - Devesh Kumar Singh
- Course Web Page: http://www.cs.purdue.edu/homes/clifton/cs348/

Course Methodology

- Lectures to present the concepts
  - Unless otherwise noted, all the material you are expected to know will be covered in class
  - Interaction (questions/discussion/thinking) encouraged
- Reading will fill in the details. One of:
  - Database System Concepts, Avi Silberschatz, Henry F. Korth, and S. Sudarshan, McGraw-Hill (2010) ISBN 0-07-352332-1
  - Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe, Pearson (2016) ISBN 9780133970777
- Homework and Projects get you to *understand* what you've read and heard
  - 5-6 written homeworks
  - 4 programming projects

Communication Tools (see course page)

- Blackboard
  - Turning in assignments, managing grades
- Echo360
  - Recorded lectures will be available
  - *May not capture what is written on the board*
- In-class response: Vote on
  - iClicker
  - Hotseat
- Out-of-class discussion: Vote on
  - Mixable
  - Piazza

6

Evaluation and Grading

- Points earned as follows:
  - Midterm (22%)
    - One evening or two in-class?
  - Final Exam (Finals week) (30%)
    - *Do not book a flight out before the end of finals!*
  - Homeworks / projects (45%)
    - Larger projects may be given higher weight
  - Instructor's evaluation (3%)
    - In-class discussions/participation
    - Out of class discussions, email
    - Overall perception of quality of your work in ways that may not be reflected in your scores
- Late work penalized 10% per day

For more details see the course web page

Motivations for DB Technology

Automation of Information Systems

- The concept of "information system" is independent from IT technology: there are organizations, the goal of which is to manage information (like in the case of demographic services), and that have been in place for centuries

Motivations for DB Technology

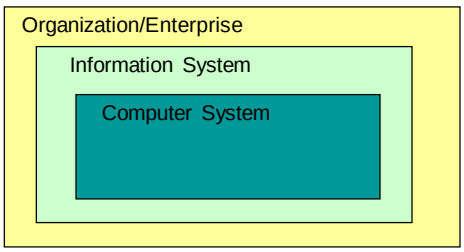
Organization/Enterprise

- Uses a set of resources, policies and regulation to execute the activities of interest for its own goals
  - Information and knowledge represent a key resource
- The *information system* is present in any organization we may think of
- The information system executes/manages information processes

Motivations for DB Technology

Computer system

- Automated portion of the information system
- Component of the information system that manages information through the use of computer systems



What is a Database?

- Collection of data, used to represent the information of interest to one or more applications in a given organization
  - Usually large
  - Organized for rapid search and retrieval
- Database Management System (DBMS): Tool to ease construction of databases
  - (Vendor) definition of database: Collection of data managed by a DBMS
- Desirable Properties:
  - Persistent Storage
    A File System does this
  - Query Interface
    Information retrieval system
  - Transaction Management

DBMS vs file system – an example

- Consider a company that needs to maintain information about its employees and its departments. Suppose that applications, managing data on employees and departments, directly use the file systems for storing and retrieving data
- According to such approach, the data concerning employees and department are stored in records collected in files. There is a file for the employee records and a file for the department records

DBMS vs file system – an example

- Assume that the following application programs are available:
  - A program to modify the salary of a given employee
  - A program to modify the department of a given employee
  - A program to insert and remove employee records
  - A program printing the list of all employee according to the lexicographic order

Drawbacks of using file systems to store data

- Data redundancy and inconsistency
  - Multiple file formats, duplication of information in different files
- Difficulty in accessing data
  - Need to write a new program to carry out each new task
- Data isolation
  - Multiple files and formats
- 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

Drawbacks of using file systems to store data (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

Database systems offer solutions to all the above problems

History of Database Systems

- 1950s and early 1960s:
  - Data processing using magnetic tapes for storage
    - Tapes provided only sequential access
  - Punched cards for input
- Late 1960s and 1970s:
  - Hard disks allowed direct access to data
  - Network and hierarchical data models in widespread use
  - Ted Codd defines the relational data model
    - Would win the ACM Turing Award for this work
    - IBM Research begins System R prototype
    - UC Berkeley begins Ingres prototype
  - High-performance (for the era) transaction processing

History (cont.)

- 1980s:
  - Research relational prototypes evolve into commercial systems
    - SQL becomes industrial standard
  - Parallel and distributed database systems
  - Object-oriented database systems
- 1990s:
  - Large decision support and data-mining applications
  - Large multi-terabyte data warehouses
  - Emergence of Web commerce
- Early 2000s:
  - XML and XQuery standards
  - Automated database administration
- Later 2000s:
  - Giant data storage systems
    - Google BigTable, Yahoo PNuts, Amazon, ..