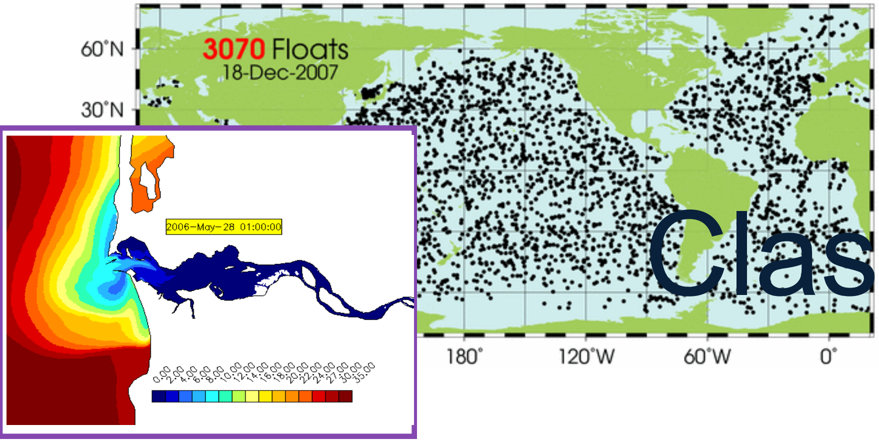
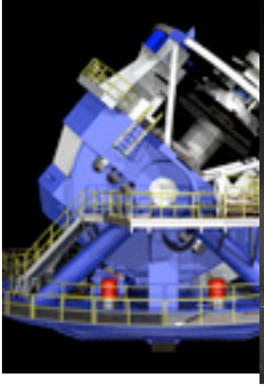
# Introduction to Data Management
# CSE 344

## Lecture 1: Introduction

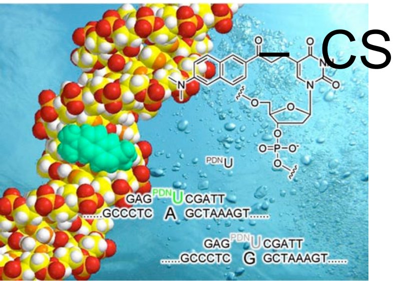
Couldn't register?
Signup on the overload list

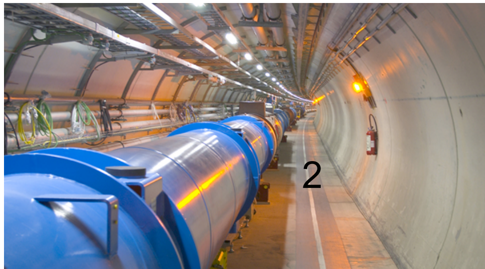


# Class Goals



- The world is drowning in data!
- Need computer scientists to help manage this data
  - Help domain scientists achieve new discoveries
  - Help companies provide better services (e.g. Facebook)
  - Help governments become more efficient
- Welcome to 344: Introduction to Data Management
  - Existing tools PLUS data management principles
- Next steps:
  - CSE 444: build data management systems
  - CSE 446: learn interesting facts from data





# Staff

- Instructor: Dan Suciu
  - suciu@cs.washington.edu
  - Office hours: Mon 1:30am-2:20pm in CSE 662
- TAs:
  - Lee Lee Choo, leelee@cs.washington.edu
  - Lewis (Brendan) Lee, lee33@cs.washington.edu
  - Tom Lehmann, tlehmann@cs.washington.edu
  - Vaspol Ruamviboonsuk, vaspol@cs.washington.edu
  - Office hours: see Website

# Course Format

- Lectures MWF, 9:30am-10:20am
- Sections: Th 8:30-9:20, 9:30-10:20
  - Content: exercises, tutorials, questions
  - Location: JHN 075

- 8 Homework assignments
- 7 Web quizzes

- Midterm and final

# Communications

- Web page: http://www.cs.washington.edu/344
  - Lectures will be available there (see calendar)
  - Homework assignments will be available there
  - Web quizzes will be available there
- Mailing list
  - Announcements, group discussions
  - You are already subscribed
- Discussion board
  - Great place to ask assignment-related questions

# Textbook

Main textbook, available at the bookstore:

- *Database Systems: The Complete Book*,
  Hector Garcia-Molina,
  Jeffrey Ullman,
  Jennifer Widom

  **Second edition**.

Most important: COME TO CLASS ! ASK QUESTIONS !

# Other Texts

Available at the Engineering Library
(not on reserve):

- *Database Management Systems*, Ramakrishnan
- *XQuery from the Experts*, Katz, Ed.
- *Fundamentals of Database Systems*, Elmasri, Navathe
- *Foundations of Databases*, Abiteboul, Hull, Vianu
- *Data on the Web,* Abiteboul, Buneman, Suciu

# Grading

| | |
|---|---|
| • Homeworks | 30% |
| • Web quizzes | 20% |
| • Midterm | 20% |
| • Final | 30% |

# Eight Homeworks

H1&H2: Basic SQL with SQLite

H3: Advanced SQL with SQL Server

H4: Relational algebra, Datalog

H5: XML and XQuery with Saxon

H6: Conceptual Design

H7: SQL in Java (JDBC)

H8: Parallel processing with MapReduce

Homework assignments are due Wednesday night – dropbox!

# About the Homeworks

- Homework assignments will take time but most time should be spent *learning*
- Very practical assignments
- Put everything on your resume!!!
  - SQL, SQLite, SQL Server, SQL Azure JDBC, XML, XQuery, Saxon, Amazon Elastic MapReduce, Hadoop, Pig Latin, …

# Seven Web Quizzes

- Class token on the white board: write it down
- Short online tests
- Can take many times: best score counts!
- Provide explanations for wrong answers
- Will help you
  - Test your knowledge
  - Stay in synch with class
  - Get ready for homeworks

Web quizes are due Friday night, but note exceptions

# Exams

- Midterm and Final
- Open book, open notes (no computers!)
- Check course website for dates
- Location: in class

# Outline of Today's Lecture

1. Overview of database management systems
    1. Why they are helpful
    2. What are some of their key features
    3. What are some of their key concepts

2. Course content

# Database

What is a database ?

Give examples of databases

# Database

What is a database ?

- A collection of files storing related data

Give examples of databases

- Accounts database; payroll database; UW's students database; Amazon's products database; airline reservation database

# Database Management System

What is a DBMS ?

Give examples of DBMSs

# Database Management System

What is a DBMS ?

- *A big program written by someone else that allows us to manage efficiently a large database and allows it to persist over long periods of time*

Give examples of DBMSs

- Oracle, IBM (DB2, Informix), Microsoft (SQL Server, Access)
- Sybase
- Open source: MySQL (Sun/Oracle), PostgreSQL
- Open source library: SQLite

We will focus on relational DBMSs most quarter

# An Example: Online Bookseller

- What data do we need?
  - Data: Information on books, customers, pending orders, order histories, trends, preferences, etc. Massive data: hundreds of GB and growing!
- What capabilities on the data do we need?
  - Add books, find a specific book, list all books in a certain category and price range, generate an order history, produce sales figures grouped by state, etc
- Data is persistent: outlives application
- Data is safe: from failures, malicious users, etc
- Multi-user access

# Multi-user discussion

- Jane and John both have ID number for gift certificate (credit) of $200 they got as a wedding gift
  - Jane @ her office orders "The Selfish Gene, R. Dawkins" ($80)
  - John @ his office orders "Guns and Steel, J. Diamond" ($100)

- Questions:
  - What is the ending credit?
  - What if second book costs $130?
  - What if system crashes?

# Summary Required Data Management Functionality

- Describe real-world entities in terms of data
- Store data persistently
- Query & update efficiently
- Change structure; e.g., add attributes
- Concurrency control: simultaneous updates
- Crash recovery
- Security and integrity

# Discussion

- Did you ever encounter a data management problem?
  - Experimental data from a homework?
  - Personal data?
  - Other data?

- How did you manage your data?

# DBMS Benefits

- Expensive to implement all these features inside the application

- DBMS provides these features (and more)

- DBMS simplifies application development

# Client/Server Architecture

- There is a single *server* that stores the database (called DBMS or RDBMS):
  - Usually a beefy system
  - But can be your own desktop…
  - … or a huge cluster running a parallel DBMS
- Many *clients* run apps and connect to DBMS
  - E.g. Microsoft's Management Studio
  - Or psql (for PostgreSQL)
  - More realistically some Java or C++ program
- Clients "talk" to server using JDBC protocol

# People

- **DB application developer**: writes programs that query and modify data (344)
- **DB designer**: establishes schema (344)
- **DB administrator**: loads data, tunes system, keeps whole thing running (344, 444)
- **Data analyst**: data mining, data integration (344, 446)
- **DBMS implementor**: builds the DBMS (444)

# Key Data Mngmt Concepts

- **Data models**: how to describe real-world data
  - Relational, XML, graph data (RDF)
- **Schema v.s. data**
- **Declarative query language**
  - Say what you want not how to get it
- **Data independence**
  - Physical independence: Can change how data is stored on disk without maintenance to applications
  - Logical independence: can change schema w/o affecting apps
- **Query optimizer** and compiler
- **Transactions**: isolation and atomicity

# What This Course Contains

- **Focus: Using DBMSs**
- Relational Data Model
  - SQL, Relational Algebra, Relational Calculus, datalog
- Semistructured Data Model
  - XML, XPath, and XQuery
- Conceptual design
  - E/R diagrams, Views, and Database normalization
- Transactions
- Parallel databases, MapReduce, and Pig-Latin
- Data integration and data cleaning

# What to Do Now

http://www.cs.washington.edu/344

- Webquiz 1 is open
  - Create account at http://newgradiance.com/
  - Use course token
  - Webquiz due this Friday
- Homework 1 is posted
  - Simple queries in SQL Lite
  - Homework due next Wednesday