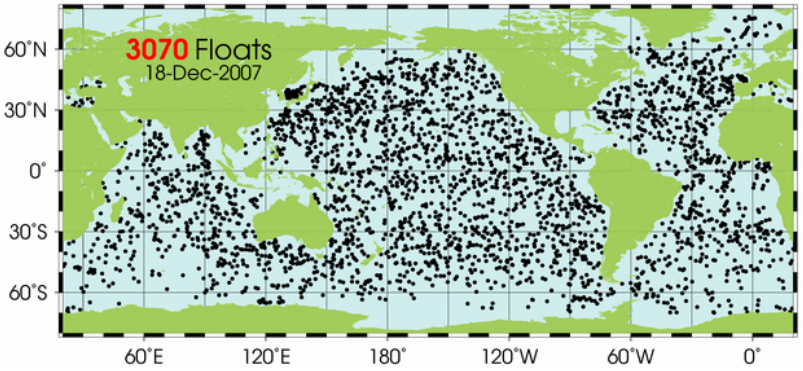
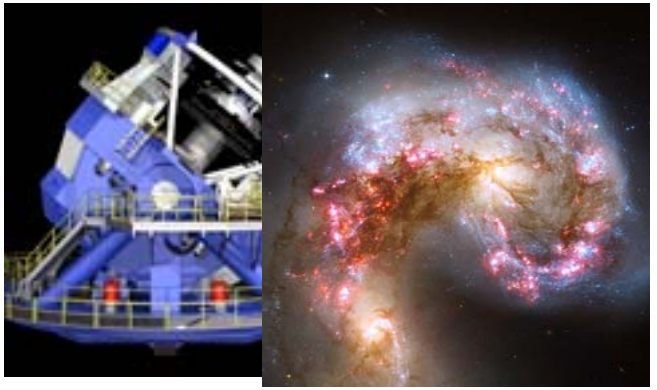
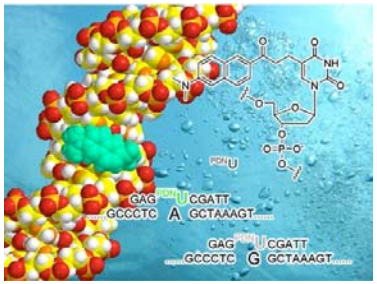
Introduction to Data Management (Database Systems) CSE 414

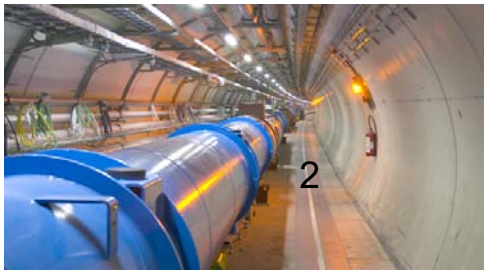
Lecture 1: Introduction



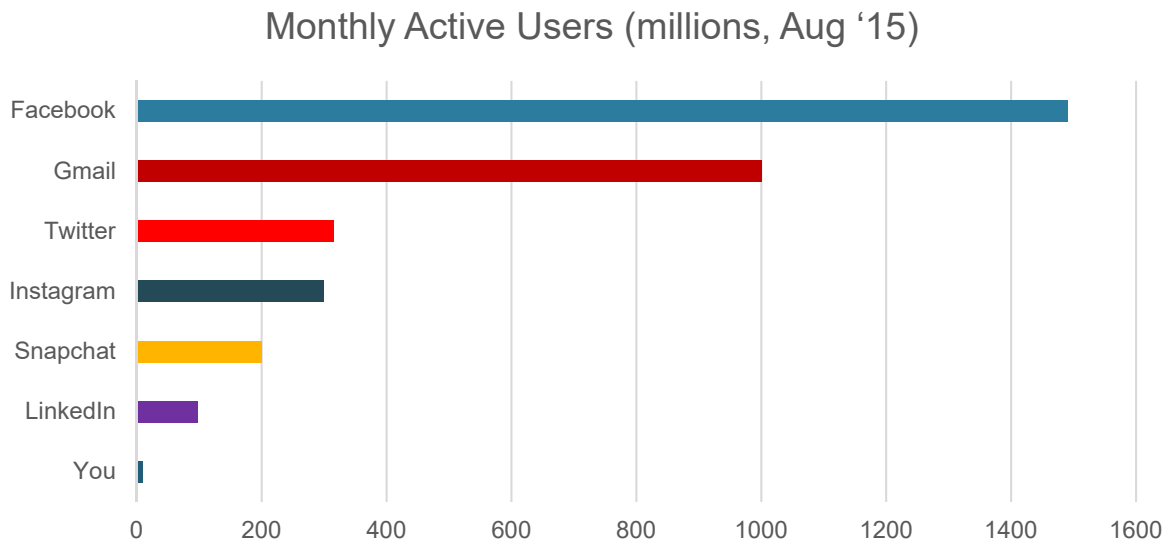


- The world is drowning in data!
- LSST produces 30 TB of data per night
  - Large Synoptic Survey Telescope
  - 9 PB per year
- LHC produced 25 PB in 2012 trying to find Higgs boson
  - Large Hadron Collider
- Affects almost every modern application…





Your New App…



- Suppose 10M monthly active, 2M daily active
- Record 20K bytes per page view / request
- 200 request per session
- Analyzing 3 months of data for trends: 1TB of data

Data Management is Universal

- Managing data is at the core of most apps / services
  - whether they store small or large amounts of data
  - whether they are modern systems or older ones
- Hard problems even with small amounts of data
  - we'll see examples later on…
- Doing it right typically makes everything else easier

Motivation

- The world is drowning in data
  - affects almost every app / service
- Need professionals to help manage it
  - help domain scientists achieve new discoveries
  - help companies provide better services
  - help governments become more efficient
- CSE 414: Introduction to Data Management
  - covers both *principles* and *tools*

Staff

- Faculty: Gang Luo
  - luogang at uw dot edu
- TAs:
  - Robert Thompson (AA), Ryan W Maas (AB), Amarpal Singh (AC)
- Office hours: check web site (under calendar)
- Contacting staff:
  - Discussion board for most things. Otherwise cse414-staff at cs

About Me

- Faculty member in the Department of Biomedical Informatics and Medical Education
- CS PhD in database from Univ. of Wisconsin
- Worked at IBM Research before
- Research interests: health informatics, big data, information retrieval, database systems, data mining, and machine learning

Course Format

- Lectures MWF, 3:30-4:20 pm
  - Location: here!
- Sections: Thursdays
  - Content: exercises, tutorials, questions
  - Locations: see web
- 8 homework assignments
  - submit via catalyst dropbox
- 6 web quizzes
  - http://www.newgradiance.com/
- Midterm and final

Communications

- Web page: https://courses.cs.washington.edu/courses/cse414
  - https://courses.cs.washington.edu/courses/cse414/17au/
  - Syllabus is there
  - Lecture slides will be available there
  - Homework assignments will be available there
  - Link to web quizzes is there
- Mailing list
  - Announcements (low traffic – must read)
  - Registered students automatically subscribed
- Discussion board
  - **THE** place to ask course-related questions
  - Today, go to board and enable notifications

Textbook

Main textbook, available at the bookstore:

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

  Second edition.

Covers most, but **not all**, of course content

Other Texts

Available at the Engineering Library:

- *Database Management Systems*, Ramakrishnan, Gehrke
- *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 Homework Assignments

H1&H2: Basic SQL with SQLite

H3: Advanced SQL with SQL Server

H4: Relational algebra, Datalog

H5: JSon and AsterixDB

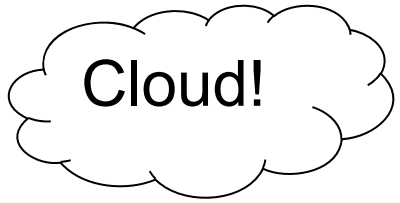
H6: Parallel processing

H7: Conceptual Design

H8: SQL in Java (JDBC)

About the Assignments

- Homework assignments will take time, but most time should be spent *learning*
- Do them on your own
- Very practical
- Put everything on your resume!!!
  - SQL, SQLite, SQL Server, **Azure**, JDBC, JSon, AWS, MapReduce, Hadoop, Spark, AsterixDB…



Deadlines and Late Days

- Assignments are expected to be done on time, but things happen, so…
- You have up to 4 late days
  - No more than 2 on any one assignment
  - Use in 24-hour chunks
- Late days = safety net, not convenience!
  - You should not plan on using them
  - If you use all 4, you are doing it wrong

Six Web Quizzes

- http://www.newgradiance.com/services/
- Create account, add class with token
  - Class token: write it down!
- Short tests
- Can take many times — best score counts
- No late days – closes at 11:00 deadline
- See explanations for wrong answers

Exams

- Midterm and Final
  - See course calendar for dates and times
- Allowed 1 letter-size paper (double-side) with notes
- Closed book. No computers, phones, watches, etc.
- Check course website for dates
- Location: in class

Academic Integrity

- Anything you submit for credit is expected to be your own work
  - encouraged to exchange ideas, but not detailed solutions
  - we all know difference between collaboration and cheating
  - attempt to gain credit for work you did not do is misconduct
- I trust you implicitly, but will come down hard on any violations of that trust

Outline of Today's Lecture

- Overview of database mgmt systems
  - Why they are helpful
  - What are some of their key features
  - What are some of their key concepts

- Course content

Database

What is a database?

- A collection of files storing related data

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 ?

- *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*

Examples of DBMSs

- Oracle, IBM DB2, Microsoft SQL Server, Vertica, Teradata
- Open source: MySQL (Sun/Oracle), PostgreSQL, AsterixDB
- Open source library: SQLite

We will focus on relational DBMSs in most of the quarter

An Example: Online Bookseller

- What data do we need?
  - Data about books, customers, pending orders, order histories, trends, preferences, etc.
  - Data about sessions (clicks, pages, searches)
  - Note: data must be persistent! Outlive application
  - Also note that data is large… won't fit all in memory
- What capabilities on the data do we need?
  - Insert/remove books, find books by author/title/etc., analyze past order history, recommend books, …
  - Data must be accessed efficiently, by many users
  - Data must be safe from failures, malicious users, and bugs!

Multi-User Issues

- 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 the server crashes?
  - What if the data center goes offline?

Required Functionality for Data Management

1. Describe real-world entities in terms of stored data
2. Persistently store large datasets
3. Efficiently query & update
   - Must handle complex questions about data
   - Must handle sophisticated updates
   - Performance matters (users can feel 200ms latency)
4. Easily change structure (e.g., add attributes)
5. Enable simultaneous updates
6. Crash recovery
7. Security and integrity

DataBase Management System (DBMS)

- Very difficult to implement all these features inside the application (correctly)
- DBMS provides these features (and more)
- DBMS simplifies application development

Client-Server Architecture

- One *server* that stores the database (DBMS):
  - 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 SQL Server Management Studio
  - Or psql (for PostgreSQL)
  - Or some Java/C++ program (very typical)
- Clients "talk" to server using JDBC protocol
  - Often phone/browser <~> web server <~> DBMS

Key People

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

Key Concepts

- **Data models**: how to describe real-world data
  - Relational, XML, JSon
- **Schema vs 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 affecting applications
  - Logical independence: can change schema w/o affecting apps
- **Query optimizer** and compiler
- **Transactions**: isolation and atomicity

Review this slide throughout the quarter!

What This Course Contains

- **Focus: Using DBMSs**
- Relational Data Model
  - SQL, Relational Algebra, Datalog
- Semistructured Data Model
  - JSon, NoSQL, AsterixDB
- Conceptual design
  - E/R diagrams, Views, and Database normalization
- Transactions
- Parallel databases, MapReduce, and Spark

What to Do Now

- https://courses.cs.washington.edu/courses/cse414/
  - https://courses.cs.washington.edu/courses/cse414/17au/

- Web quiz 1 is open
  - Create account at http://newgradiance.com/services/
  - Sign up for class (use token)
  - Due Oct. 10, 11 pm

- Homework 1 is posted
  - Simple queries in SQL Lite
  - Due Oct. 9, 11 pm

- Use discussion board if you have questions about HW

- The instructor will try to post HW and WQ early. You are strongly encouraged to finish them early and definitely should not drag to the last minute to do them

Announcements

- Bring your laptop to the lecture on Friday
  - With SQLite installed
- Bring your laptop and credit card to section on Thursday
  - To help you set up Azure and AWS accounts
  - you will be using Microsoft Azure
  - we will send out codes for free student use
    - good for 3 months and up to $100
  - look at HW1 for installing sqlite
  - can go through the examples yourself