Introduction to Database Systems
CSE 414

Lecture 1: Introduction
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 (and universities!) become more efficient
• Welcome to 414: Introduction to Database Systems
  – Existing tools PLUS data management principles
  – This is not just a class on SQL!
Turing Awards in Data Management

Charles Bachman, 1973
*IDS and CODASYL*

Ted Codd, 1981
*Relational model*

Jim Gray, 1998
*Transaction processing*

Michael Stonebraker, 2014
*INGRES and Postgres*

You could be next!!
Staff

• Instructor: Alvin Cheung
  – Office hour on Wednesdays, 1-2pm

From ACM Spring BBQ 15
Trusty TAs

Kodiak Conrad
Jack Khuu
Jonathan Leang
Boyan Li
Cindy Suripto
Vineeth Varghese
Ying Wang
Course Format

• Lectures
  – Location: here!
  – Please attend

• Sections:
  – Content: exercises, tutorials, questions, new materials (occasionally)
  – Locations: see web
  – Please attend
  – Bring your laptop

• 8 homework assignments
• 7 web quizzes

• Midterm and final

• Class and section participation
  – Post and answer questions (in class, piazza, etc)
Grading

- Homeworks 30%
- Web quizzes 10%
- Midterm 20%
- Final 30%
- Class participation 10%

- This is all subject to change
Communications

• **Web page:** [http://www.cs.washington.edu/414](http://www.cs.washington.edu/414)
  - Syllabus (course information)
  - Schedule: add to your calendar
  - Lecture/section notes will be available there
  - Homework assignments will be available there
  - Link to web quizzes is there

• **Piazza**
  - Sign up: [https://piazza.com/washington/spring2018/cse414](https://piazza.com/washington/spring2018/cse414)
  - **THE** place to ask course-related questions
  - Log in today and enable notifications
Textbook

Main textbook, available at the bookstore:


  **Second edition.**
Other Texts

Available at the Engineering Library (some on reserve):

• *Database Management Systems*, Ramakrishnan
• *Fundamentals of Database Systems*, Elmasri, Navathe
• *Foundations of Databases*, Abiteboul, Hull, Vianu
• *Data on the Web*, Abiteboul, Buneman, Suciu
Prerequisites

Formally: CSE143: Computer Programming II

Assume knowledge of:

• Java programming
• Basic data structures (lists, trees, objects)
• Unix (command line tools)
Eight Homework Assignments

H1: SQL+sqlite intro (1 week)
H2: SQL basics (1 week)
H3: Advanced SQL on Azure (1+ weeks)
H4: Datalog and Relational Algebra (1+ weeks)
H5: NoSQL: Json/SQL++ (1 week)
H6: Spark on AWS (1+ weeks)
H7: Schema Design (1 week)
H8: Transactional Application (1+ weeks)
About the Assignments

• You will learn/practice the course material:
  – SQL, RA, parallel db, transactions, ...

• You will also learn lots of new technology
  – Cloud computing: Azure, and Amazon web services
  – NoSQL: AsterixDB, Spark
  – Databases: sqlite, Microsoft SQL Server
  – Git

• Each ranges in its difficulty to setup and use
• Will require (non-trivial) time to fiddle and explore!
• The time spent learning the new technology is very useful: write everything on your CV!
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
Seven Web Quizzes

- [http://newgradiance.com/](http://newgradiance.com/)
- Create account; please use the same ID as your UW ID
- Course token will be posted on piazza
- Short tests, take many times, best score counts
- No late days – closes at 11:59pm deadline
- Provide explanations for wrong answers
Exams

• Midterm (May 2) and Final (June 7)

• You may bring letter-size piece of paper with notes
  – May write on both sides
  – Midterm: 1 sheet, Final: 2 sheets

• Closed book. No computers, phones, watches,...

• Location: in class
Academic Integrity

• Anything you submit for credit is expected to be your own work
  – Of course OK 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

• We trust you implicitly, but will come down hard on any violations of that trust
Lecture Notes

• Will be available before class online

• Feel free to bring them to class to take notes

• We can bring hard copies to class if needed
Using Electronics in Class

In the lectures:
• Opened laptops may disturb neighbors
• Please sit in the back if you take notes on laptop; pads / surfaces are OK
• Please don’t check your email / youtube / fb

In the sections:
• Always bring your laptop (starting Thursday)
Now onto the real stuff...
Outline of Today’s Lecture

• Overview of database management systems

• 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
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
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, Microsoft SQL Server, Vertica, Teradata
– Open source: MySQL (Sun/Oracle), PostgreSQL, CouchDB
– Open source library: SQLite

We will focus on relational DBMSs most quarter
An Example: Online Bookseller

• What data do we need?
  -
  -
  -
  -

• What capabilities on the data do we need?
  -
  -
  -
  -
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?
  –
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 and malicious users
Challenges for a DBMS

Alice and Bob receive a $200 gift certificate as wedding gift
Challenges for a DBMS

Alice and Bob receive a $200 gift certificate as wedding gift

Alice @ her office orders
"The Selfish Gene"

Bob @ home orders
“Guns, germs, and steel”
Challenges for a DBMS

Alice and Bob receive a $200 gift certificate as wedding gift.

Alice @ her office orders "The Selfish Gene" $80
Bob @ home orders "Guns, germs, and steel" $100

Questions:
What is the ending credit?
What if second book costs $130?
What if system crashes?
Challenges for a DBMS

Alice and Bob receive a $200 gift certificate as wedding gift

Alice @ her office orders "The Selfish Gene" for $80

Bob @ home orders "Guns, germs, and steel" for $100

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

Lesson: a DBMS needs to handle various scenarios.
What a DBMS Does

• Describe real-world entities in terms of stored data
• Persistently store large datasets
• Efficiently query & update
  – Must handle complex questions about data
  – Must handle sophisticated updates
  – Performance matters
• Change structure (e.g., add attributes)
• Concurrency control: enable simultaneous updates
• Crash recovery
• Security and integrity
The players

• **DB application developer**: writes programs that query and modify data (414)
• **DB designer**: establishes schema (414)
• **DB administrator**: loads data, tunes system, keeps whole thing running (414, 444)
• **Data analyst**: data mining, data integration (414, 446, CSED 516)
• **DBMS implementor**: builds the DBMS (444)
• **Research on new systems**: (544)
Data Management Concepts

• Data model
• Declarative query language
• Data independence
• Query optimization
• Physical design
• Transactions
What is this class about?

- **Data models**
  - Relational: SQL and Datalog
  - NoSQL: SQL++
- **RDMBS internals**
  - Relational algebra
  - Query optimization and physical design
- **Parallel query processing**
  - Spark and Hadoop
- **Conceptual design**
  - E/R diagrams
  - Schema normalization
- **Transactions**
  - Locking and schedules
  - Writing DB applications
What to Do Now

http://www.cs.washington.edu/414

• Homework 1 will be posted
  – Simple queries in SQL Lite
  – Due on Tuesday, 4/3

• Webquiz 1 will open tomorrow
  – Create account at http://newgradiance.com/
  – Sign up for class online
  – Due on Tuesday, 4/3

• First sections on Thursday
  – Tutorial on git and SQLite

• Post on Piazza if you have questions about HW and lecture