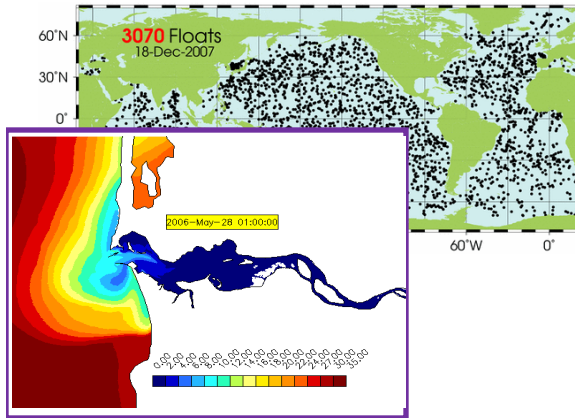# CSE 444: Database Internals
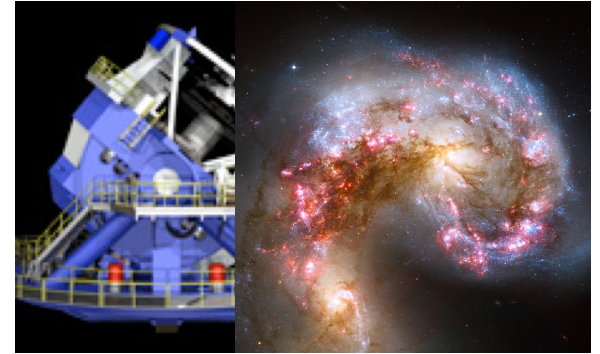
## Lecture 1
## Introduction

# Course Staff

- Instructors:
  - Ryan Maas
- TAs:
  - Josh Bean
  - Kevin Bi
  - Shumo Chu
  - James Wang
  - Shibin (Jack) Xu

  - Email addresses and office hour times and locations are on the course website
    - Every day one or more of us have office hours

# Course 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
  - Help governments become more efficient

- This class: **principles of building data mgmt systems**
  - Learn how classical DBMSs are built
  - Learn key principles and techniques
  - Get hands-on experience building a working DBMS
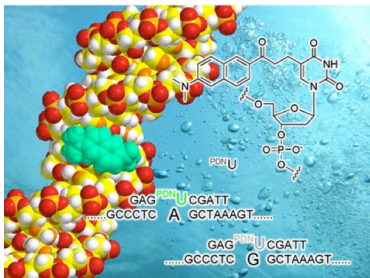
CSE 444 - Winter 2019

# Course Format

- Lectures MWF @ 10:30am

- Sections: Th 9:30, 10:30, 11:30, or 12:30

- Homeworks
  - 5 Labs + 6 Homeworks

- Quizzes:
  - 4 short quizzes in class

# Communication (part 1)

- Web page: http://www.cs.washington.edu/444
  - Lectures/Sections slides will be posted there
  - Homeworks/Labs will be available there

- Mailing list
  - Announcements, group discussions
  - Your @uw.edu address is already subscribed

# Communication (part 2)

**Message Board**:

- https://piazza.com/washington/winter2019/cse444/home

- Ask questions about the course, labs, homeworks
  - Feel free to answer questions too! If you think you know how to answer but are not sure, simply say so
  - Staff will check & answer questions regularly
    - If your question has not been answered in 12 hours, let me know

- Do not post any fragments of your code

# Communication (part 3)

- Do **not** send questions by email unless
  - You need to discuss a personal matter
  - You want to setup an appointment
  - A question has not been answered on the board
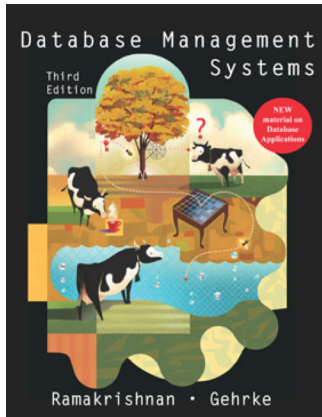
# Textbooks
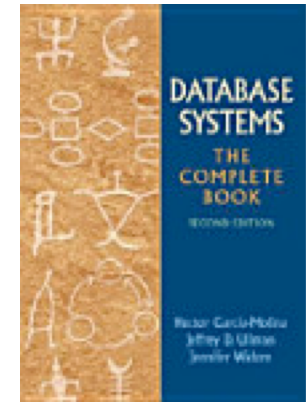
<span style="color:blue">Recommended textbook (pick one)</span>

- Database Management Systems. **Third Ed**. Ramakrishnan and Gehrke. McGraw-Hill.

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

See course website for recommended chapters

# Other Readings

- See Website

- There is a section on reading assignments for 544M only

# Grading CSE444

- Labs:   40%
  - Includes final project lab

- Final project report 10%

- Six written assignments: 30%

- Four lab quizzes 20%

# Grading CSE 544M

- Same as CSE 444 plus
- Another 10% for the 4 paper reviews
- Then re-normalize to add up to 100%

- Graded separately from CSE 444

Acks: SimpleDB lab series originally developed by Prof. Sam Madden at MIT. We work with them on improving/extending.

# Six Labs

- Lab 1: Build a DBMS that can scan a relation on disk
  - **RELEASED TODAY! Part 1 of this lab is due on Monday!**
- Lab 2: Build a DBMS that can run simple SQL queries and also supports data updates

- Lab 3: Add a lock manager (transactions)
- Lab 4: Add a write-ahead log (transactions)

- Lab 5: Add a query optimizer

# About the Labs

Managed on GitLab:

https://gitlab.cs.washington.edu/cse444-19wi/simple-db-[your gitlab id]

Logistics:

- To be done **INDIVIDUALLY!**
- Each lab will take a **significant** amount of time
- Labs build on each other

Purpose

- Hands-on experience building a DBMS
- Deepen your understanding significantly
- We will build a *classical* DBMS

13

# Six Homeworks

- <span style="color:red">Homework 1 released today. Due next week</span>

- Written assignments – **Print out pdf and fill in answers**

- Help review material learned in class

- Prepare you for the labs
  - One homework before each corresponding lab

- Go beyond what we implement in labs

- To be done **INDIVIDUALLY**

# Quizzes

- One quiz in class for each of labs 1-4

- Tests depth of your knowledge
  - **No notes. No code. Answer from memory**
  - Only one or two open-ended questions
  - Example: "Explain how data is stored in SimpleDB"
  - Grades:
    - 9-10: Strength! Exceptional understanding and explanations
    - 8: You got it!
    - 7 or less: Developing knowledge – some gaps
    - 0: Did not show up or wrote nothing
  - Important: We grade based on the depth of knowledge demonstrated in your answer

# Late Days

- Total of **4 late-days**
- Use in 24-hour chunks on hws or labs
- **At most 2 late-days per assignment**

- <span style="color:red">**No late-days can be applied to the final project due during finals week**</span>

# Outline (this lecture and next)

- Review of DBMS goals and features

- Review of relational model

- Review of SQL

# Review: DBMS

- **What is a database?** Give examples

  –

  –

- **What is a database management system?** Give examples

  –

  –

# Review: DBMS

- **What is a database?** Give examples
  - A collection of related files
  - E.g. payroll, accounting, products

- **What is a database management system?** Give examples
  - A program written by someone else that manages the database; PostgreSQL, Oracle, …
  - In 444 you are that "someone else", implementing SimpleDB

# Review: Data Model

- ## What is a data model?

  -

- ## What is the relational data model?

# Review: Data Model

- **What is a data model?**

  – A mathematical formalism for data


- **What is the relational data model?**

  – Data is stored in tables (aka relations)

  – Data is queried via relational queries

  – Queries are *set-at-a-time*

# Review: Transactions

- ## What is a transaction?
  - 

- ## What properties do transactions have?

# Review: Transactions

- ## What is a transaction?
  - A set of instructions that must be executed all or nothing

- ## What properties do transactions have?
  - ACID
  - Better: Serialization, recovery

# Review: Data Independence

The application should not be affected by changes of the physical storage of data

- Indexes
- Physical organization on disk
- Physical plans for accessing the data
- Parallelism: multicore, distributed

# Some Key Data Management Concepts

- Data models: Relational, XML, graph data (RDF)

- Schema vs. Data

- Declarative query languages
  - Say what you want not how to get it

- Data independence
  - Physical: Can change how data is stored on disk without maintenance to applications

- Query compiler and optimizer

- Transactions: isolation and atomicity

# Course Content

**Focus: how to build a classical relational DBMS**

- Review of the relational model (lecture 1 and 2)
- DBMS architecture and deployments (lecture 3)
- Data storage, indexing, and buffer mgmt (lectures 4-6)
- Query evaluation (lectures 7-8)
- Query optimization (lectures 9-12)
- Transactions (lectures 13-19)
- Parallel query processing (lectures 20-23)
- Replication and distribution (lectures 24-25)
- NoSQL and NewSQL (lectures 26-27)

# Relational Model...

- The foundation of our traditional database management system

- We'll continue our review of the relational model next lecture …