CMSC424: Database Design
Introduction/Overview

Professor: Pete Keleher
keleher@umd.edu

Today

- Administrivia
- Motivation: Why study databases? What is databases?
- Current Industry Outlook
- A typical DBMS at a glance
Logistics

- Professor: Peter Keleher
  - 5146 Iribe Bldg
  - keleher@umd.edu
  - Class Webpage:
    - http://sedna.cs.umd.edu/424

- Communication:
  - Piazza
  - Office hours
  - Email to me: include 424 in subject as a last resort.
  - Do not message me on ELMS, I will not read it.

---

Logistics

Grading

All grades will be on grades.cs.umd.edu.

<table>
<thead>
<tr>
<th>36% Weekly Assignments</th>
<th>12% Weekly Quizzes</th>
</tr>
</thead>
<tbody>
<tr>
<td>We have 9 graded assignments:</td>
<td>12 weekly quizzes:</td>
</tr>
<tr>
<td>• Each are worth 4%.</td>
<td>• Each is worth 1%</td>
</tr>
<tr>
<td>• All are due Sunday at midnight.</td>
<td>• All are due Friday at midnight</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>52% Exams</th>
</tr>
</thead>
<tbody>
<tr>
<td>We have (3) exams:</td>
</tr>
<tr>
<td>• Exam 1 is 16%</td>
</tr>
<tr>
<td>• Exam 2 is 16%</td>
</tr>
<tr>
<td>• The Final Exam is 20%</td>
</tr>
</tbody>
</table>
Logistics

Grading
- Whole class is curved: avg is B min, stdev up or down for A, C
- Approximate cut-offs *last year* (not guaranteed)
  - 85+: A
  - 75+: B
  - 65+: C
  - 60-: D/F
- Most had 40+ points (out of 50) on non-exams last year
  - Exams are usually somewhat harder (no curves)
  - *Must average a passing grade on the total exam score*

Logistics

- Web site: [https://sedna.cs.umd.edu/424](https://sedna.cs.umd.edu/424)
- Discussion: [https://piazza.com/class/l6xsiaxujqx3nf](https://piazza.com/class/l6xsiaxujqx3nf)
- Grades: [https://grades.cs.umd.edu](https://grades.cs.umd.edu)
- Gradescope: [https://www.gradescope.com/courses/424744](https://www.gradescope.com/courses/424744)
  - quizzes, assignment submissions, graded exams
- Office Hours
  - Pete (me) IRB 5146, by appt
  - TAs (hours TBD):
    - Pooja Nilangekar
    - Charles Cassel
    - Johnny Rajala
- ELMS
  - *Nope!*
Some To-Dos

- Sign up for Piazza!
  - If not already added

- Set up the computing environment (Assign. 0), and make sure you can run Vagrant+VirtualBox, PostgreSQL, IPython, etc.

- Upcoming:
  - Quiz 1 (due next Friday),
  - Assign 0: Environment. (this Sunday, but not graded/no submission)
  - Assign 1: SQL. (next Sunday, midnight)

Motivation: Data Overload

- Explosion of data, in pretty much every domain
  - Sensing devices and sensor networks that can monitor everything 24/7 from temperature to pollution to vital signs
  - Increasingly sophisticated smart phones
  - Internet, social networks makes it easy to publish data
  - Scientific experiments and simulations produce astronomical volumes of data
  - Internet of Things
  - Datafication: taking all aspects of life and turning them into data (e.g., what you like/enjoy turned into a stream of your "likes")

- How to handle that data? How to extract interesting actionable insights and scientific knowledge?
- Data volumes expected to get much worse
Four V’s of Big Data

- Increasing data Volumes
  - **Scientific data**: 1.5GB/genome -- can be sequenced in .5 hrs; LHC generates 100TB of data a day
  - 500M tweets per day (as of 2013)
  - As of 2012: 2.5 Exabytes of data created every day
  - EBay: Two data warehouses with 7.5PB and 40PB
  - Walmart: 583 terabytes of sales and inventory data
  - FICO monitors 2.5 billion active accounts worldwide

- Variety:
  - Structured data, spreadsheets, photos, videos, natural text, ...

- Velocity

- Veracity

Sensors everywhere -- can generate tremendous volumes of "data streams"
- Real-time analytics requires data to be consumed as fast as it is generated

- Veracity
  - How do you decide what to trust? How to remove noise? How to fill in missing values?
**Big Data and Data Science to the Rescue**

- Terms increasingly used synonymously: also data analytics, data mining, business intelligence
  - Loosely used for any process where interesting things are inferred from data
  - Google search: “How Big Data Will Change”
- Data scientist called the sexiest job of the 21st century
  - The term has becoming very muddled at this point
- Overhyped words
  - We are headed toward the trough of Disillusionment

---

**Is it all hype?**

- No: Extracting insights and knowledge from data very important, and will continue to increase in importance
  - Big data techniques are revolutionizing things in many domains like Education, Food Supply, Disease Epidemics, ...
- But: it is not much different from what we, especially statisticians, have been doing for many years
- What is different?
  - Much more data is digitally available than was before
  - Inexpensive computing + Cloud + Easy-to-use programming frameworks = Much easier to analyze it
  - Often: large-scale data + simple algorithms > small data + complex algorithms
    - Changes how you do analysis dramatically
Motivation: Data Overload

- How do we do anything with this data?

- Where and how do we store it?
  - Disks are doubling every 18 months or so -- not enough
  - In many cases, the data is not actually recorded as it is; *summarized* first

- What if the disks crash?
  - Very common, especially with 10,000’s of disks

- How do we ensure “correctness”?
  - What if the system crashes in the middle of an ATM transaction?
    - Can’t have money disappearing
  - What happens when a million people try to buy tickets to *<your favorite artist>*’s concert at the same time?

- What to do with the data? How to process/analyze it?
  - text search?
    - Very limited
  - “find the stores with the maximum increase in sales in last month”
    - We can’t expect the users to write Java programs
  - “how much time from here to Pittsburgh if I start at 2pm?”
    - Data is there; more will be soon (GPS, live traffic data)
    - Requires predictive capabilities
  - Increasing need to convert “information” to “knowledge”: *Data mining*
    - “How should we replicate different movies?” (Netflix)
    - Find videos with this type of an event (say car break-ins)
    - Mine the “blogs” to detect “buzz”
Motivation: Data Overload

- **Speed !!**
  - With TB’s of data, just finding something (even if you know what), is not easy
    - Reading a file with TB of data can take hours
  - Imagine a bank and millions of ATMs
    - How much time does it take you to do a withdrawal?
    - The data is not local

- How do we guarantee the data will be there 10 years from now?

- **Privacy and security !!!**
  - Every other day we see some database leaked on the web
    - identity fraud, influencing elections...
  - How to make sure different users’ data is protected from each other

Why not use file systems?

- **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
**Why not use file systems?**

- **Drawbacks of using file systems to store data:**
  - **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

---

**Today**

- **Administrivia**
  - Quiz 1, Assign 1 due Friday/Sunday 11:59pm next week
  - M1 macs: “brew install postgresql”
DBMSs to the Rescue

- Provide a systematic way to answer many of these questions...
- Aim is to allow easy management of high volumes of data
  - Storing, Updating, Querying, Analyzing ....

- What is a Database?
  - A large, integrated collection of (mostly structured) data
  - Typically models and captures information about a real-world enterprise
    - Entities (e.g. courses, students)
    - Relationships (e.g. John is taking CMSC 424)
  - Usually also contains:
    - Knowledge of constraints on the data (e.g. course capacities)
    - Business logic (e.g. pre-requisite rules)
    - Encoded as part of the data model (preferable) or through external programs

Massively successful for highly structured data

- Why? Structure in the data (if any) can be exploited for ease of use and efficiency
  - If there is no structure in the data, hard to do much
  - Contrast managing emails vs managing photos
- Much of the data we need to deal with is highly structured
- Some data is semi-structured
  - E.g.: Resumes, Webpages, Blogs etc.
- Some has complicated structure
  - E.g.: Social networks
- Some has no structure
  - E.g.: Text data, Video/Image data etc.
Structured vs Unstructured Data

- A lot of the data we encounter is structured
  - Some have very simple structures
    - E.g. Data that can be represented in tabular forms
    - Significantly easier to deal with
    - We will focus on such data for much of the class

<table>
<thead>
<tr>
<th>Account</th>
<th>Customer</th>
</tr>
</thead>
<tbody>
<tr>
<td>bname</td>
<td>cname</td>
</tr>
<tr>
<td>acct_no</td>
<td>cstreet</td>
</tr>
<tr>
<td>balance</td>
<td></td>
</tr>
<tr>
<td>Downtown</td>
<td>Jones</td>
</tr>
<tr>
<td>Mianus</td>
<td>Smith</td>
</tr>
<tr>
<td>Perry</td>
<td>Hayes</td>
</tr>
<tr>
<td>R.H</td>
<td>Curry</td>
</tr>
<tr>
<td>A-101</td>
<td>Lindsay</td>
</tr>
<tr>
<td>500</td>
<td></td>
</tr>
<tr>
<td>A-215</td>
<td></td>
</tr>
<tr>
<td>700</td>
<td></td>
</tr>
<tr>
<td>A-102</td>
<td></td>
</tr>
<tr>
<td>400</td>
<td></td>
</tr>
<tr>
<td>A-305</td>
<td></td>
</tr>
<tr>
<td>350</td>
<td></td>
</tr>
</tbody>
</table>

Structured vs Unstructured Data

- Some data has a little more complicated structure
  - E.g graph structures
    - Map data, social networks data, the web link structure etc.
  - Can convert to tabular forms for storage, but may not be optimal
  - Queries often reason about graph structure
    - *Find my “Erdos number”*
    - *Suggest friends based on current friends*
  - Growing importance in recent years in a variety of domains: Biological, social networks, web...
Structured vs Unstructured Data

- Increasing amount of data in a semi-structured format
  - XML – Self-describing tags (HTML ?)
  - Complicates a lot of things
  - We will discuss this toward the end

- A huge amount of data is unfortunately unstructured
  - Books, WWW
  - Amenable to pretty much only text search... so far
    - Information Retrieval research deals with this topic
  - What about Google search ?
    - Google search is mainly successful because it uses link structure (in its original incarnation)

- Video ? Music ?
  - Can represent in DBMS’s, but can’t really operate on them

circle size == page importance == pagerank
  more incoming links → higher pagerank
  incoming links from important pages → higher pagerank