W4111
Introduction to Databases
Spring 2016

Evan Jones

Computer Science Department

Columbia University

Adjunct (not a “real” professor)

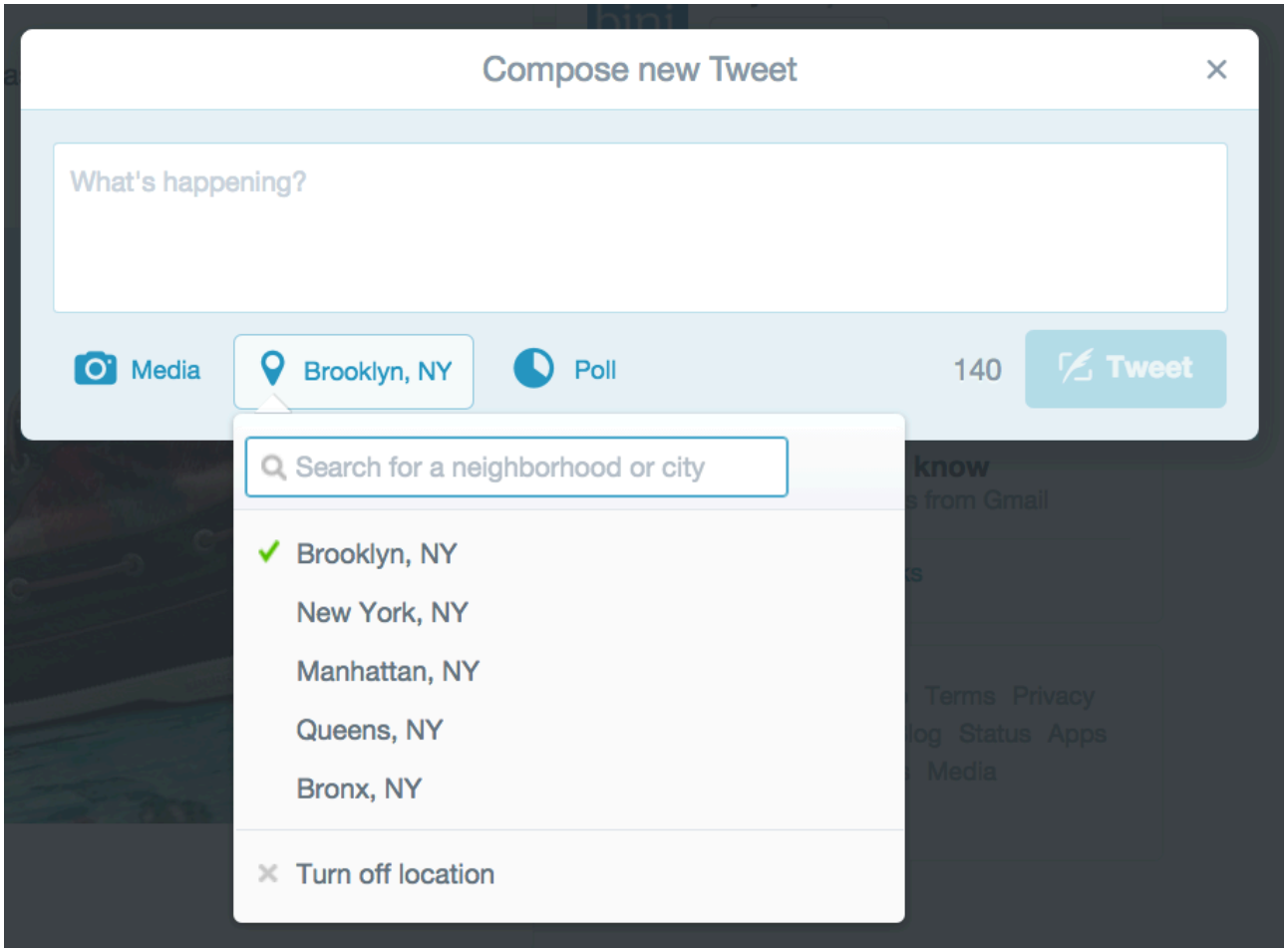
2006-2007: Google NYC

2007-2011: MIT databases

2012-2014: Mitro (failed startup)

2014-Now: Twitter

Twitter Geo data



Administrative details

http://www.cs.columbia.edu/~coms4111/

https://github.com/w4111/syllabus

Q&A: On Piazza

Office hours: To be announced (Evan: Thursdays)

Homework 0: Due Thursday

Data

Data

is for serious business

IT Executive

Deliver the Real-Time Enterprise

With growing data volumes and aggressive service level expectations, maximize the potential of your IT organization while delivering the real-time enterprise.

More for IT Executives ›

- **Are you unlocking the full potential of your database infrastructure?**
- **Are you unleashing the full potential of your database professionals?**
- **Does your data seamlessly meet availability, security, and compliance requirements?**
- **Does your database enable competitive business operations and analytics?**

Data

is at the center of most things.

Data

is at the center of *everything*

Betting worth billions. Elite players. Violent threats. Covert messages with Sicilian gamblers. And suspicious matches at Wimbledon. Leaked files expose match-fixing evidence that tennis authorities have kept secret for years.

A BUZZFEED NEWS / BBC INVESTIGATION

The investigation into men's tennis by BuzzFeed News and the BBC is based on a cache of leaked documents from inside the sport – the Fixing Files – as well as an original analysis of the betting activity on 26,000 matches and interviews across three continents with gambling and match-fixing experts, tennis officials, and players.

Data Sizes

Bit
(⅛ byte)

1/8 of a letter

Nibble
(½ byte)

1/2 of a letter

Byte
(1 byte)

1 letter

Megabyte
(1,024 kilobytes)

1 book

Gigabyte
(1,024 megabytes)

1600 books

Terabyte
(1,024 gigabytes)

1,600,000 books

Petabyte
(1,024 terrabytes)

160,000,000 books

1 exabyte is equivalent toabout **3000 times** the entire content of the Library of Congress.

Exabyte
(1,024 petabytes)

Only 10,000 miles short of reaching the moon!

1,600,000,000,000 books

2.5 quintillion bytes of data are created daily, produced by everything from photos uploaded to social media websites, to weather balloons, to the Curiosity rover currently exploring Mars.

Bigger Than Big Data

How did we get here?

Data was *Expensive*

Data is *Cheap*

U32 Shadow™ 1TB External USB 3.0 Portable Hard Drive

by Oyen Digital

$84.95 Prime

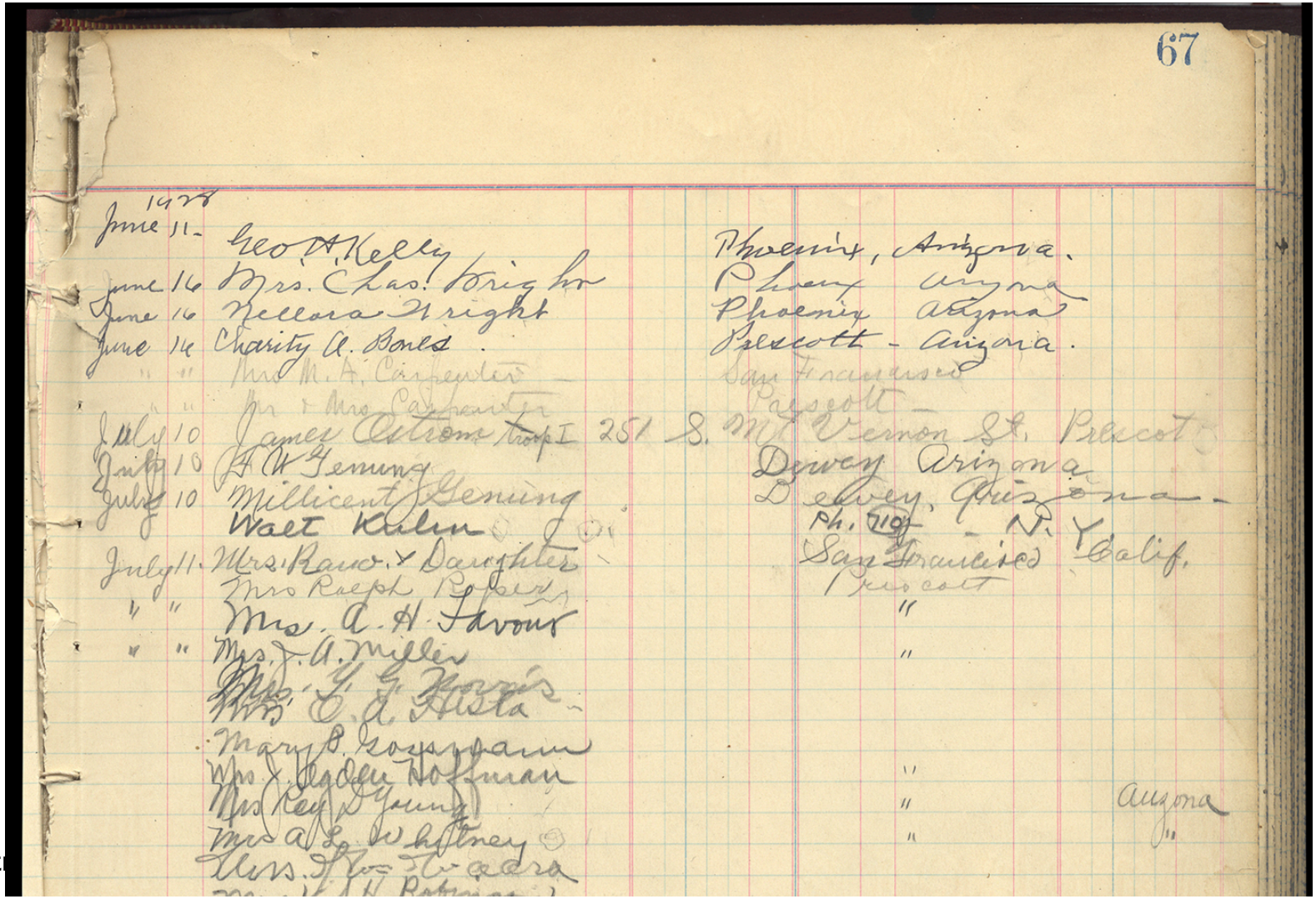
Only 4 left in stock - order soon.

More Buying Choices

$84.95 new (2 offers)

$74.33 used (1 offer)

Data was *Manual*



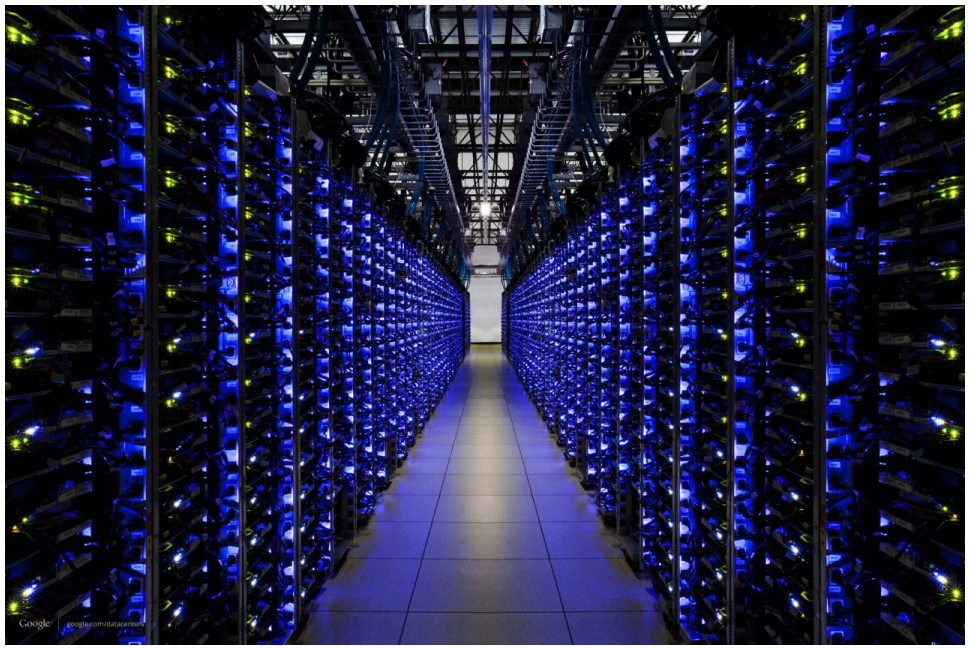
Data is *Automated*

Physical devices

Data is *Automated*

Physical devices

Software logs



Data is *Ubiquitous*

Physical devices

Software logs

Phones

Data is *Ubiquitous*

Physical devices

Software logs

Phones

GPS/Cars

Data is *Everywhere*

Physical devices

Software logs

Phones

GPS/Cars

Internet of *Things*

All this data, what are we doing with it?

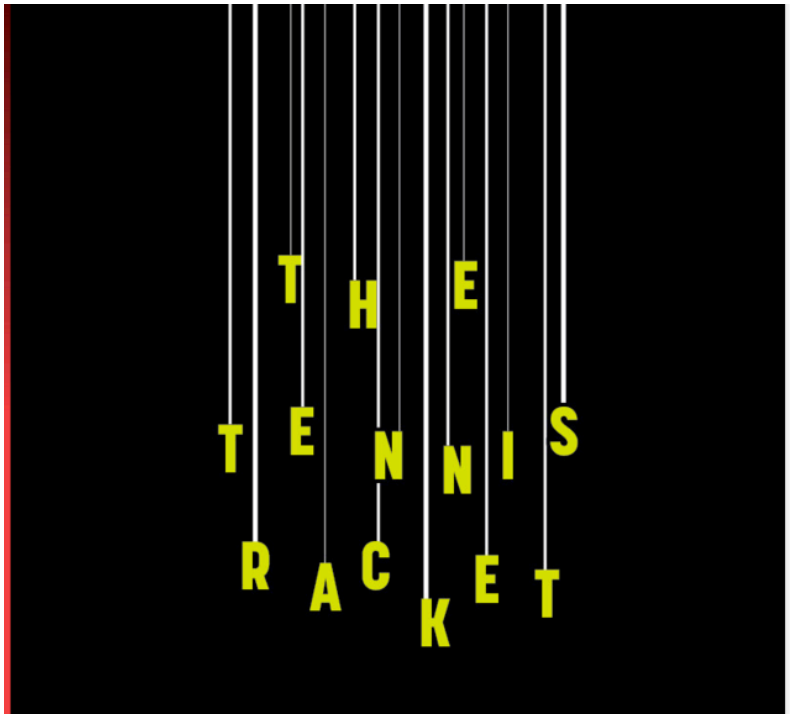
What are we doing with data?

Health

What are we doing with data?

Health

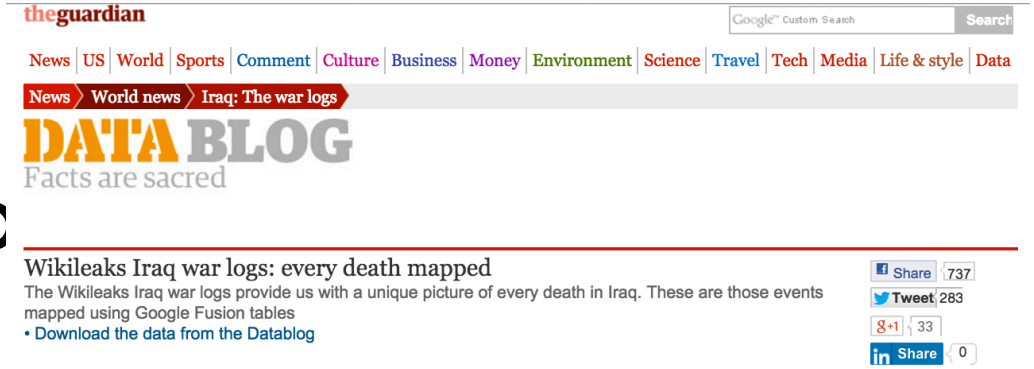
Investigative Journalism

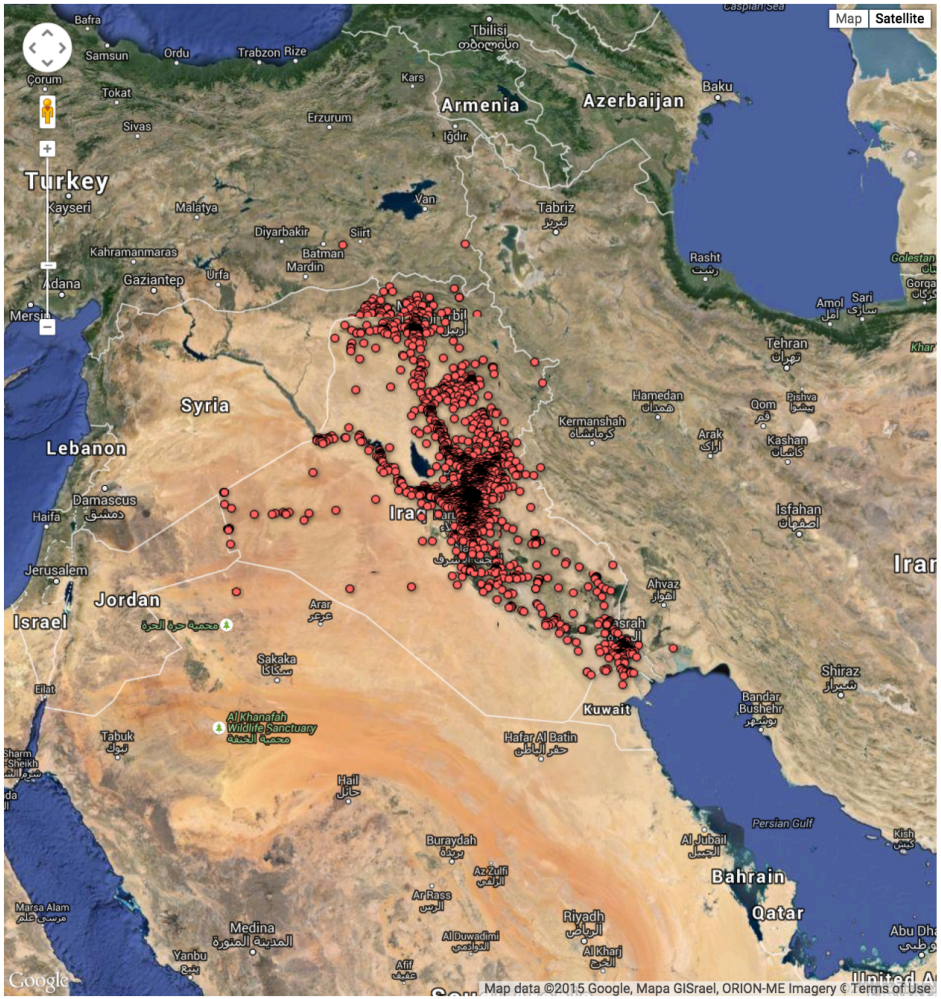


What are we do

Health

Investigative Journalism





What are we doing with data?

Health

Investigative Journalism

Recommendations

What are we doing with data?

Health

Investigative Journalism

Forbes / Tech

2 FREE Issues of Forbes

FEB 16, 2012 @ 11:02 AM 2,814,982 VIEWS

How Target Figured Out A Teen Girl Was Pregnant Before Her Father Did

Kashmir Hill, FORBES STAFF

Welcome to The Not-So Private Parts where technology & privacy collide

FOLLOW ON FORBES (2079)

Opinions expressed by Forbes Contributors are their own.

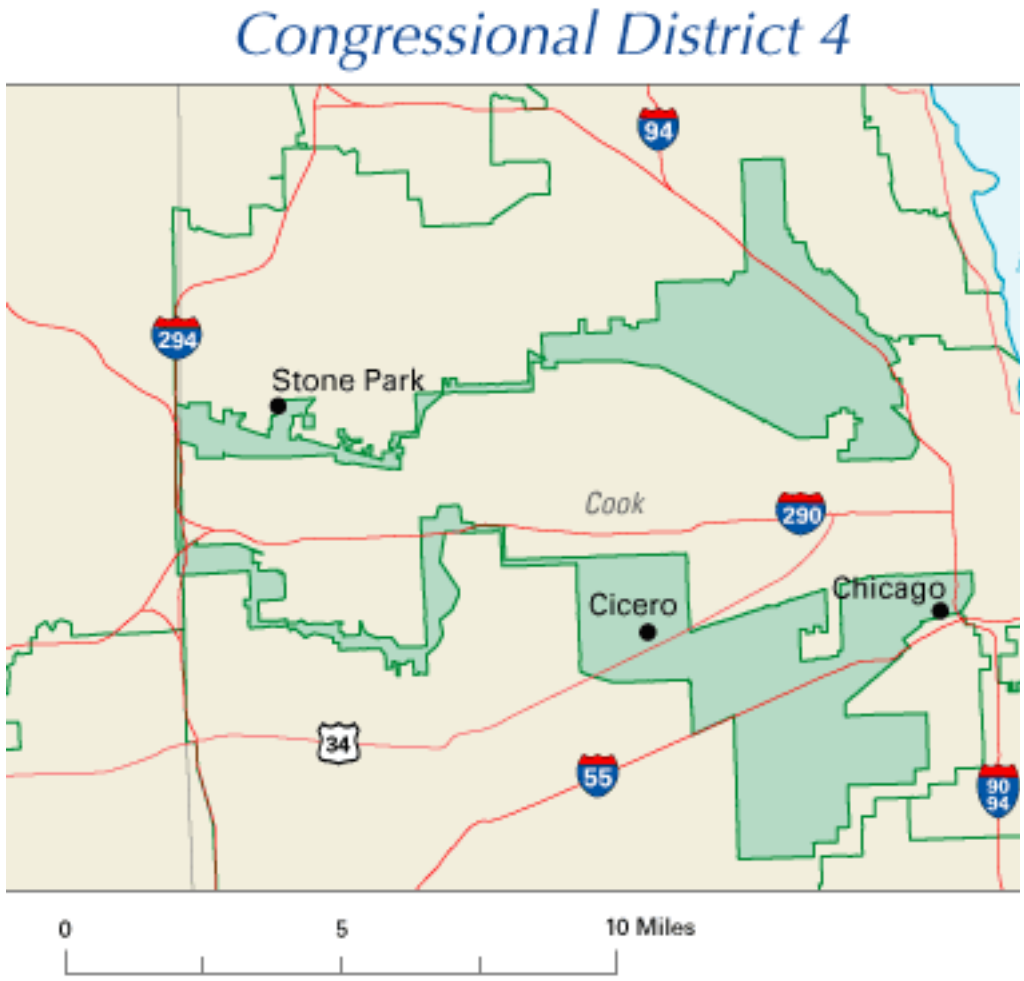
What are we doing with data?

Health

Investigative Journalism

Recommendations

Politics



What are we doing with data?

Health

Investigative Journalism

Recommendations

Politics


TIME Subscribe

2012 ELECTION

Inside the Secret World of the Data Crunchers Who Helped Obama Win

Data-driven decisionmaking played a huge role in creating a second term for the 44th President and will be one of the more closely studied elements of the 2012 cycle

What are we doing with data?

Health

Investigative Journalism

Recommendations

Politics

Surveillance

What are we doing with data?

Health

Investigative Journalism

Recommendations

Politics

Surveillance

Identity

HARVARD | BUSINESS | SCHOOL

30 APR 2012 RESEARCH & IDEAS

India's Ambitious National Identification Program

Comments 30 Email Print Download Share Recommend Share 92

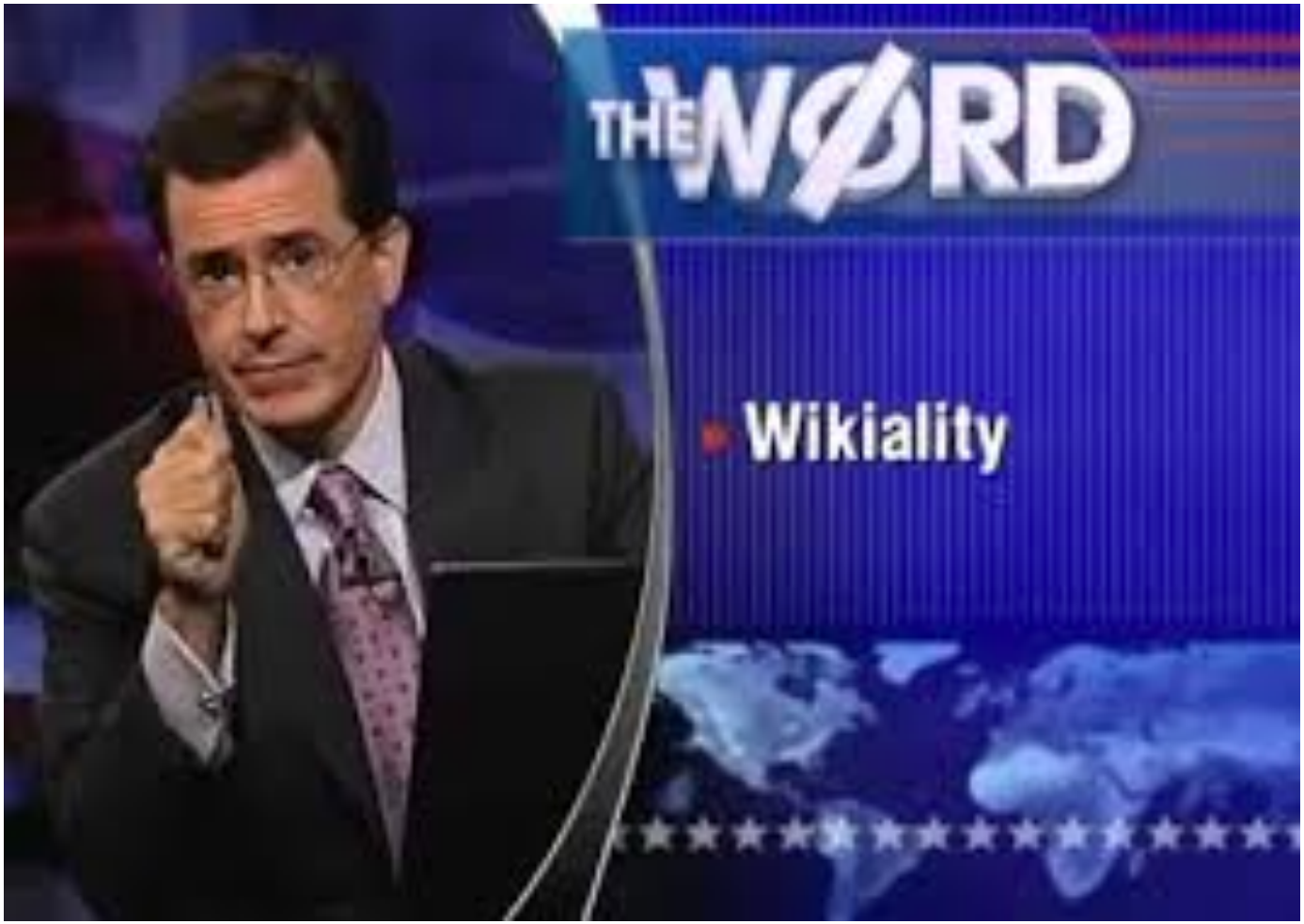
The Unique Identification Authority of India has been charged with implementing a nationwide program to register and assign a unique 12-digit ID to every Indian resident—some 1.2 billion people—by 2020. In a new case, Professor Tarun Khanna and HBS India Research Center Executive Director Anjali Raina discuss the complexities of this massive data management project.

"YOU ARE BASICALLY DENIED ALMOST EVERYTHING IF YOU CAN'T PROVE WHO YOU ARE."

What data?

What data?

Fake data



What data?

Fake data

Biased data

What data?

Fake data

Biased data

Incorrect data

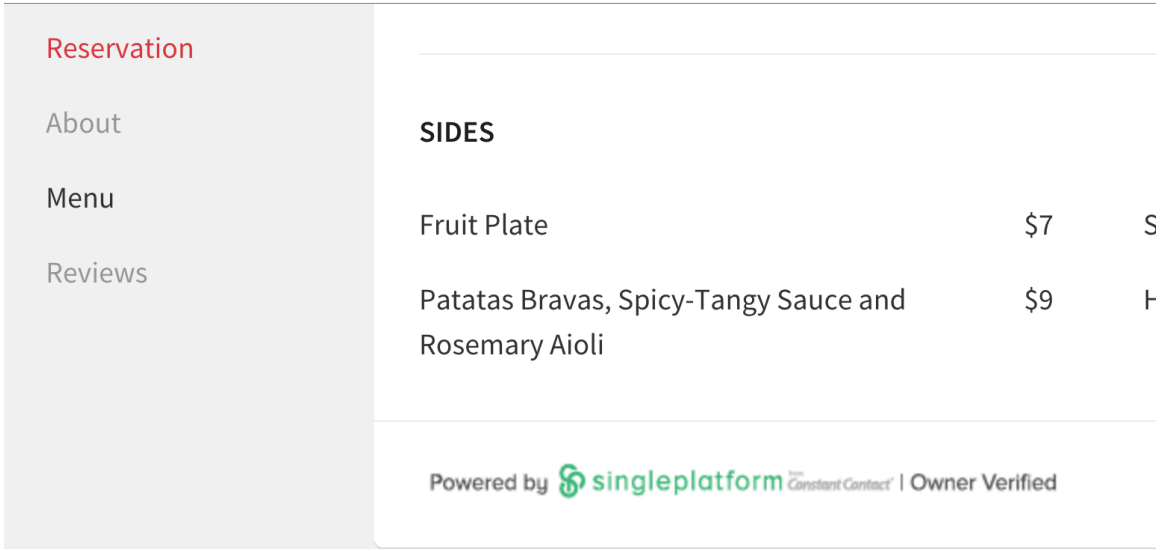
What data?

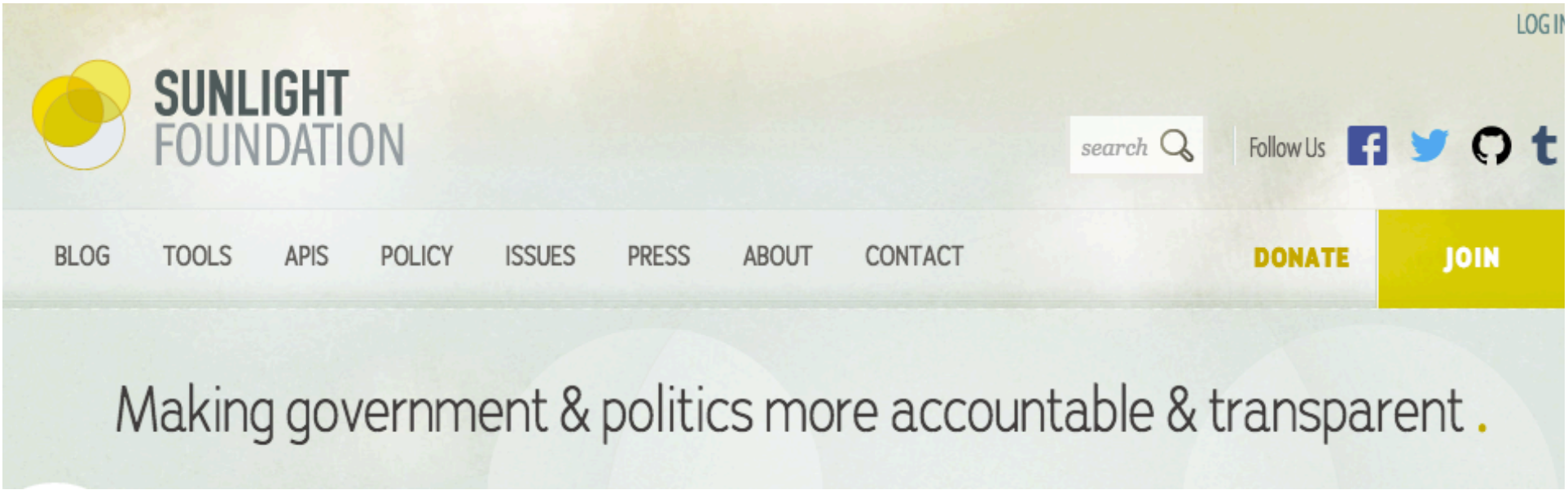
Fake data

Biased data

Incorrect data

Mixed data



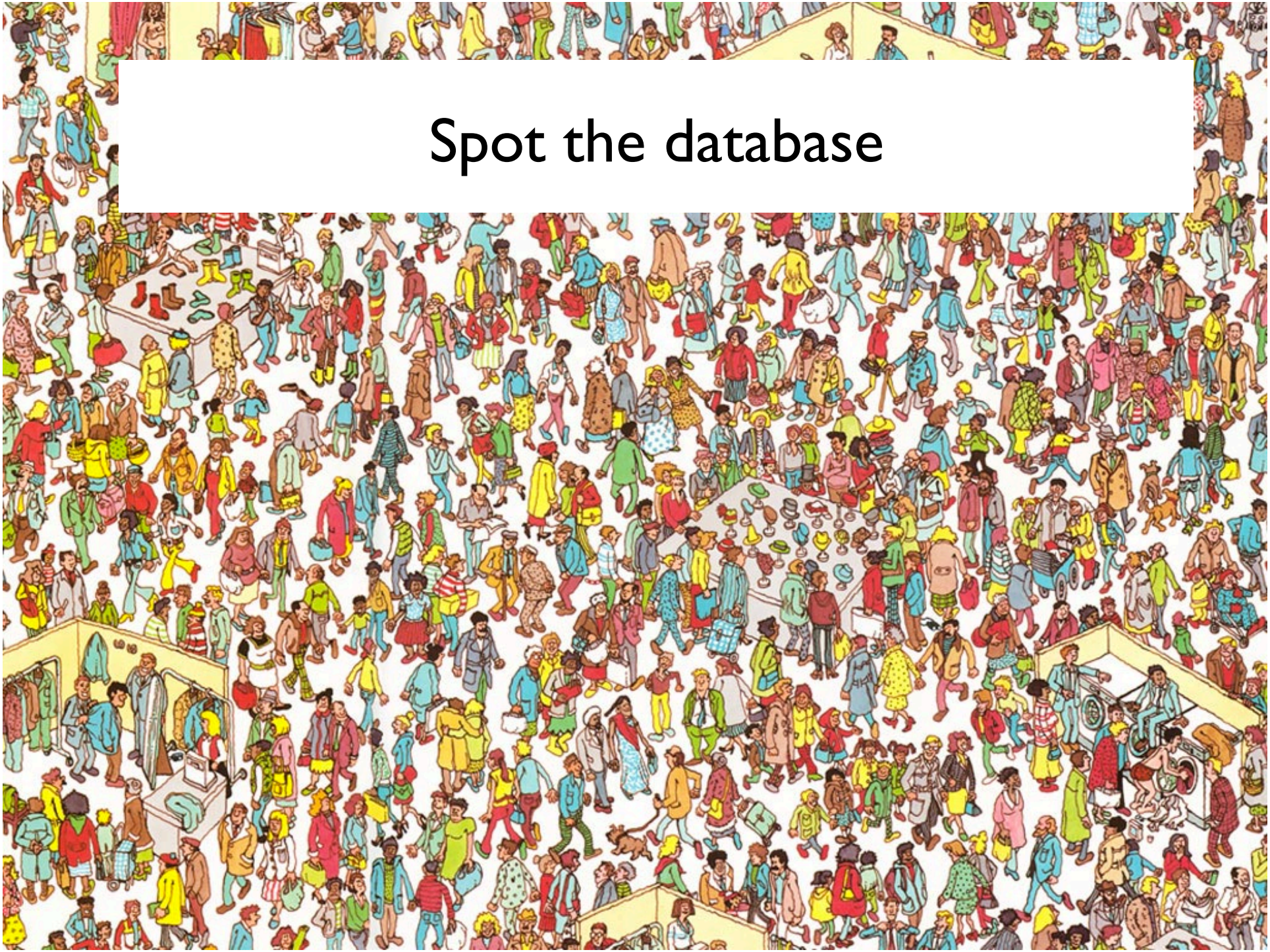


Data will be crucial to

how we live

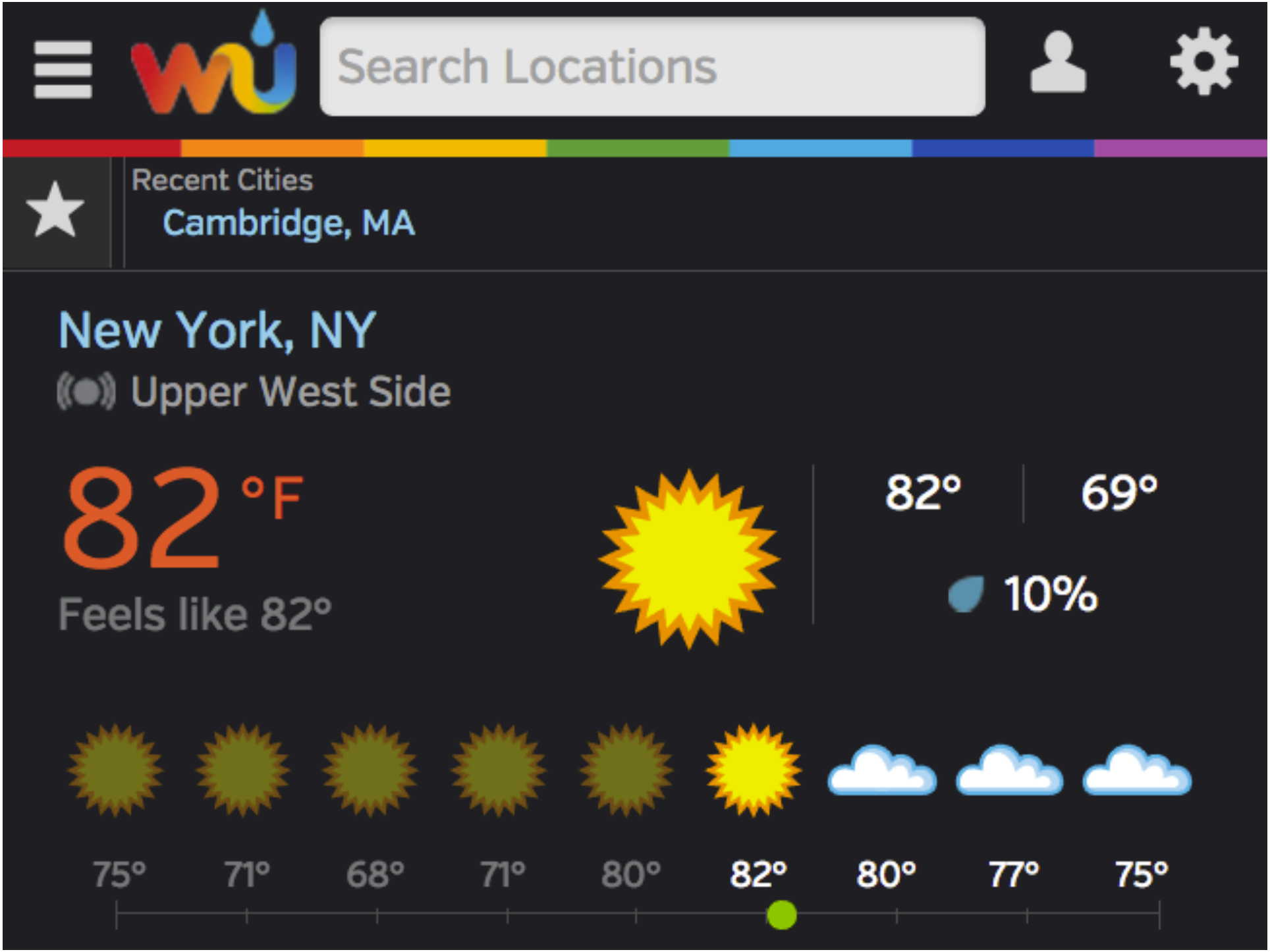
as individuals and as a society

Spot the database

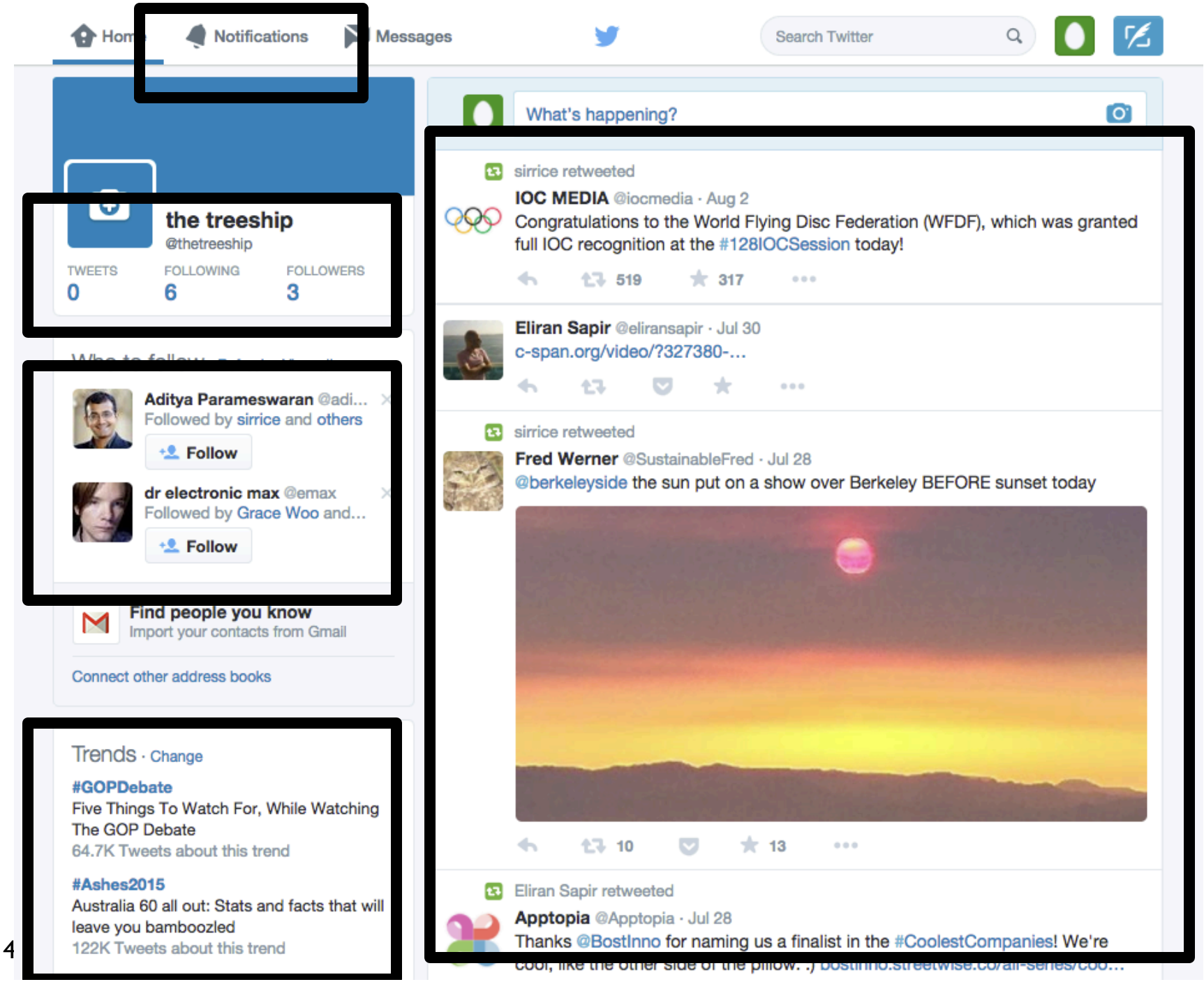



M
N
Litho Supply
100 Brand View Str
Wheeling, IL 90060
(911) 667-4545
L
K
J
H


ATM
24 HOUR BANKING


Search Locations
Recent Cities
Cambridge, MA
New York, NY
Upper West Side
82 °F
Feels like 82°
82°
69°
10%
75°
71°
68°
71°
80°
82°
80°
77°
75°


Google
Search Google or type URL

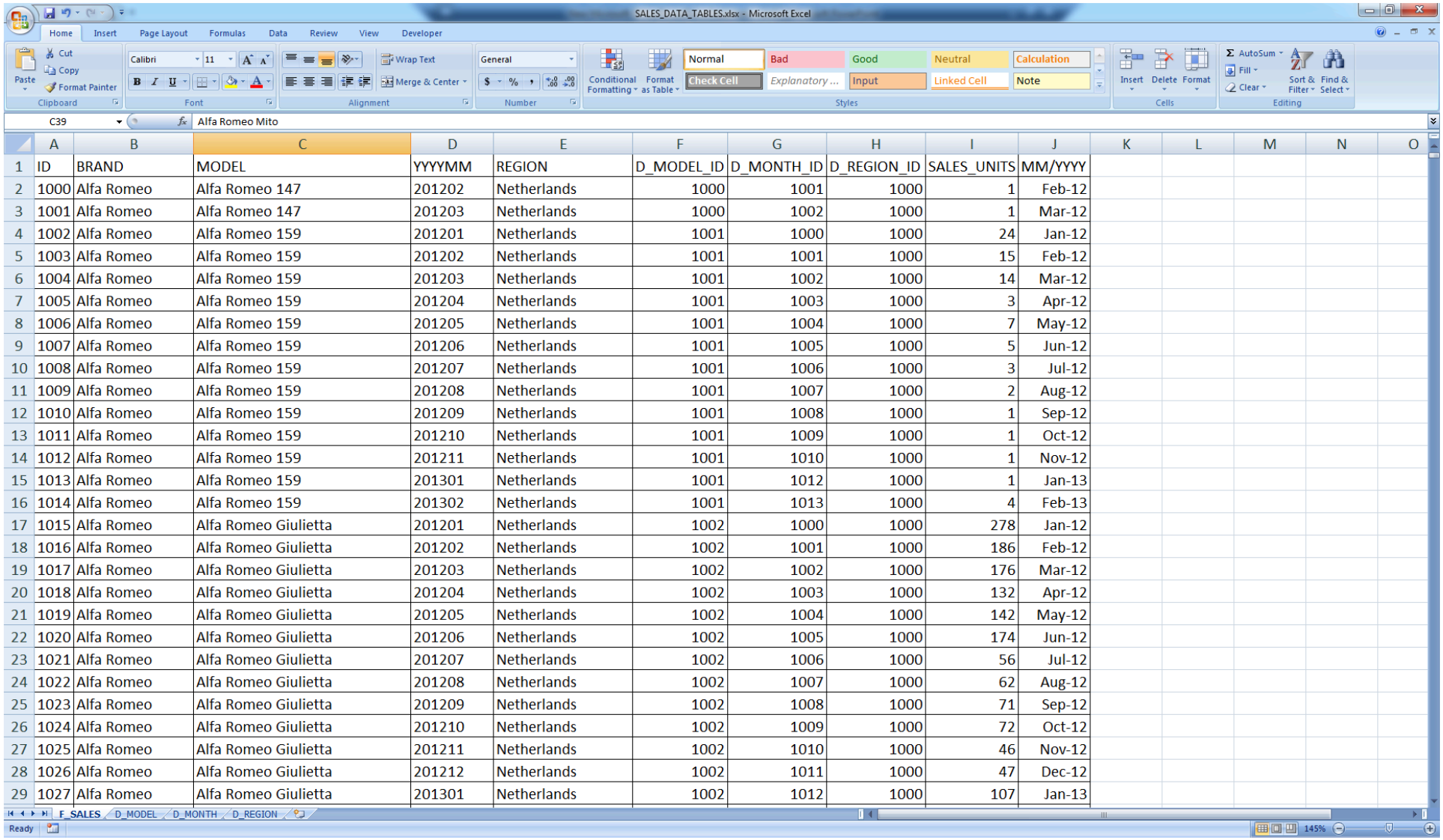

Notifications
Messages
Search Twitter
What's happening?
the treeship
@thetreeship
TWEETS 0
FOLLOWING 6
FOLLOWERS 3
Aditya Parameswaran @adi...
Followed by sirrice and others
Follow
dr electronic max @emax
Followed by Grace Woo and...
Follow
Find people you know
Import your contacts from Gmail
Connect other address books
Trends · Change
#GOPDebate
Five Things To Watch For, While Watching The GOP Debate
64.7K Tweets about this trend
#Ashes2015
Australia 60 all out: Stats and facts that will leave you bamboozled
122K Tweets about this trend
sirrice retweeted
IOC MEDIA @iocmedia · Aug 2
Congratulations to the World Flying Disc Federation (WFDF), which was granted full IOC recognition at the #128IOCSession today!
519
317
Eliran Sapir @eliransapir · Jul 30
c-span.org/video/?327380-...
sirrice retweeted
Fred Werner @SustainableFred · Jul 28
@berkeleyside the sun put on a show over Berkeley BEFORE sunset today
10
13
Eliran Sapir retweeted
Apptopia @Apptopia · Jul 28
Thanks @BostInno for naming us a finalist in the #CoolestCompanies! We're

```
2012-01-04 00:01:23,180 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Receiving
010
2012-01-04 00:01:23,184 INFO org.apache.hadoop.hdfs.server.datanode.DataNode.clienttrace
 cliID: DFSClient_-603743753, offset: 0, srvID: DS-292194659-127.0.1.1-50010-13247633001
2012-01-04 00:01:23,185 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: PacketResp
2012-01-04 00:01:23,291 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Receiving
10
2012-01-04 00:01:23,293 INFO org.apache.hadoop.hdfs.server.datanode.DataNode.clienttrace
cliID: DFSClient_-603743753, offset: 0, srvID: DS-292194659-127.0.1.1-50010-132476330017
2012-01-04 00:01:23,293 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: PacketResp
2012-01-04 00:01:23,324 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Receiving
010
2012-01-04 00:01:23,326 INFO org.apache.hadoop.hdfs.server.datanode.DataNode.clienttrace
liID: DFSClient_-603743753, offset: 0, srvID: DS-292194659-127.0.1.1-50010-1324763300176
2012-01-04 00:01:23,327 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: PacketResp
2012-01-04 00:01:23,409 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Receiving
10
2012-01-04 00:01:23,411 INFO org.apache.hadoop.hdfs.server.datanode.DataNode.clienttrace
, cliID: DFSClient_-603743753, offset: 0, srvID: DS-292194659-127.0.1.1-50010-1324763300
2012-01-04 00:01:23,411 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: PacketResp
2012-01-04 00:01:23,433 INFO org.apache.hadoop.hdfs.server.datanode.DataNode.clienttrace
 cliID: DFSClient_-2054881890, offset: 0, srvID: DS-292194659-127.0.1.1-50010-1324763300
2012-01-04 00:01:23,494 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Receiving
10
2012-01-04 00:01:23,498 INFO org.apache.hadoop.hdfs.server.datanode.DataNode.clienttrace
, cliID: DFSClient_-2054881890, offset: 0, srvID: DS-292194659-127.0.1.1-50010-132476330
2012-01-04 00:01:23,498 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: PacketResp
2012-01-04 00:01:23,523 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Receiving
010
```


Canon
EOS


4K-30
00:01
HERO4
GoPro


To the Reader.
SHAKESPEARES
COMEDIES,
HISTORIES, &
TRAGEDIES.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	ID	BRAND	MODEL	YYYYMM	REGION	D_MODEL_ID	D_MONTH_ID	D_REGION_ID	SALES_UNITS	MM/YYYY					
2	1000	Alfa Romeo	Alfa Romeo 147	201202	Netherlands	1000	1001	1000	1	Feb-12					
3	1001	Alfa Romeo	Alfa Romeo 147	201203	Netherlands	1000	1002	1000	1	Mar-12					
4	1002	Alfa Romeo	Alfa Romeo 159	201201	Netherlands	1001	1000	1000	24	Jan-12					
5	1003	Alfa Romeo	Alfa Romeo 159	201202	Netherlands	1001	1001	1000	15	Feb-12					
6	1004	Alfa Romeo	Alfa Romeo 159	201203	Netherlands	1001	1002	1000	14	Mar-12					
7	1005	Alfa Romeo	Alfa Romeo 159	201204	Netherlands	1001	1003	1000	3	Apr-12					
8	1006	Alfa Romeo	Alfa Romeo 159	201205	Netherlands	1001	1004	1000	7	May-12					
9	1007	Alfa Romeo	Alfa Romeo 159	201206	Netherlands	1001	1005	1000	5	Jun-12					
10	1008	Alfa Romeo	Alfa Romeo 159	201207	Netherlands	1001	1006	1000	3	Jul-12					
11	1009	Alfa Romeo	Alfa Romeo 159	201208	Netherlands	1001	1007	1000	2	Aug-12					
12	1010	Alfa Romeo	Alfa Romeo 159	201209	Netherlands	1001	1008	1000	1	Sep-12					
13	1011	Alfa Romeo	Alfa Romeo 159	201210	Netherlands	1001	1009	1000	1	Oct-12					
14	1012	Alfa Romeo	Alfa Romeo 159	201211	Netherlands	1001	1010	1000	1	Nov-12					
15	1013	Alfa Romeo	Alfa Romeo 159	201301	Netherlands	1001	1012	1000	1	Jan-13					
16	1014	Alfa Romeo	Alfa Romeo 159	201302	Netherlands	1001	1013	1000	4	Feb-13					
17	1015	Alfa Romeo	Alfa Romeo Giulietta	201201	Netherlands	1002	1000	1000	278	Jan-12					
18	1016	Alfa Romeo	Alfa Romeo Giulietta	201202	Netherlands	1002	1001	1000	186	Feb-12					
19	1017	Alfa Romeo	Alfa Romeo Giulietta	201203	Netherlands	1002	1002	1000	176	Mar-12					
20	1018	Alfa Romeo	Alfa Romeo Giulietta	201204	Netherlands	1002	1003	1000	132	Apr-12					
21	1019	Alfa Romeo	Alfa Romeo Giulietta	201205	Netherlands	1002	1004	1000	142	May-12					
22	1020	Alfa Romeo	Alfa Romeo Giulietta	201206	Netherlands	1002	1005	1000	174	Jun-12					
23	1021	Alfa Romeo	Alfa Romeo Giulietta	201207	Netherlands	1002	1006	1000	56	Jul-12					
24	1022	Alfa Romeo	Alfa Romeo Giulietta	201208	Netherlands	1002	1007	1000	62	Aug-12					
25	1023	Alfa Romeo	Alfa Romeo Giulietta	201209	Netherlands	1002	1008	1000	71	Sep-12					
26	1024	Alfa Romeo	Alfa Romeo Giulietta	201210	Netherlands	1002	1009	1000	72	Oct-12					
27	1025	Alfa Romeo	Alfa Romeo Giulietta	201211	Netherlands	1002	1010	1000	46	Nov-12					
28	1026	Alfa Romeo	Alfa Romeo Giulietta	201212	Netherlands	1002	1011	1000	47	Dec-12					
29	1027	Alfa Romeo	Alfa Romeo Giulietta	201301	Netherlands	1002	1012	1000	107	Jan-13					

Dewey #	10 Main Classes	Kinds of Books
000-099	General Works	encyclopedias, almanacs, record books, such as Guinness
100-199	Philosophy and Psychology	paranormal phenomena,such as ghosts, ethics, how we think
200-299	Religion	mythology, religions
300-399	Social Science	government, holidays, folklore, fairy tales, education, community
400-499	Language	English and foreign languages, sign language, dictionaries
500-599	Natural Science	math, chemistry, biology, weather, rocks, plants, animals in nature
600-699	Applied Science	inventions, health, drugs, transportation, cooking, pets
700-799	Fine Arts and Recreation	crafts, art, drawing, painting, music, games, TV, movies, sports
800-899	Literature	short stories, poetry, plays, jokes, riddles (fiction could be here)
900-999	History and Geography	countries, flags, historical events, biographies (92 or 920)

What is a Database?

Structured data

What is a Database?

Lots of

Structured data

Database Management System (DBMS)

A system to store, manage and access databases

Database Management System (DBMS)

System to safely and reliably store lots of persistent structured data and is convenient for multiple users to efficiently access and modify.

Is a program a DBMS?

Java/Python/Javascript etc

- Data stored in variables (RAM)
- Very fast access

Is a program a DBMS?

Java/Python/Javascript etc

- Data stored in variables (RAM)
- Very fast access

What about crashes? Restarts?

Is Excel a DBMS?

Visually access/modify/compute over data cells

Click save to store persistently

Is Excel a DBMS?

Visually access/modify/compute over data cells

Click save to store persistently

What about sharing? Huge tables? Multiple tables?

Is the file system a DBMS?

Manages files that are persistently stored on disk

Open/read/seek/write access to files

Access via file names

Access control via permissions

Is the file system a DBMS?

You and a friend edit the same text file

Save at the same time

What happens?

1. Your changes survive
2. Friend's changes survive
3. Both changes survive
4. No changes survive
5. ¯_(ツ)_/¯

Is the file system a DBMS?

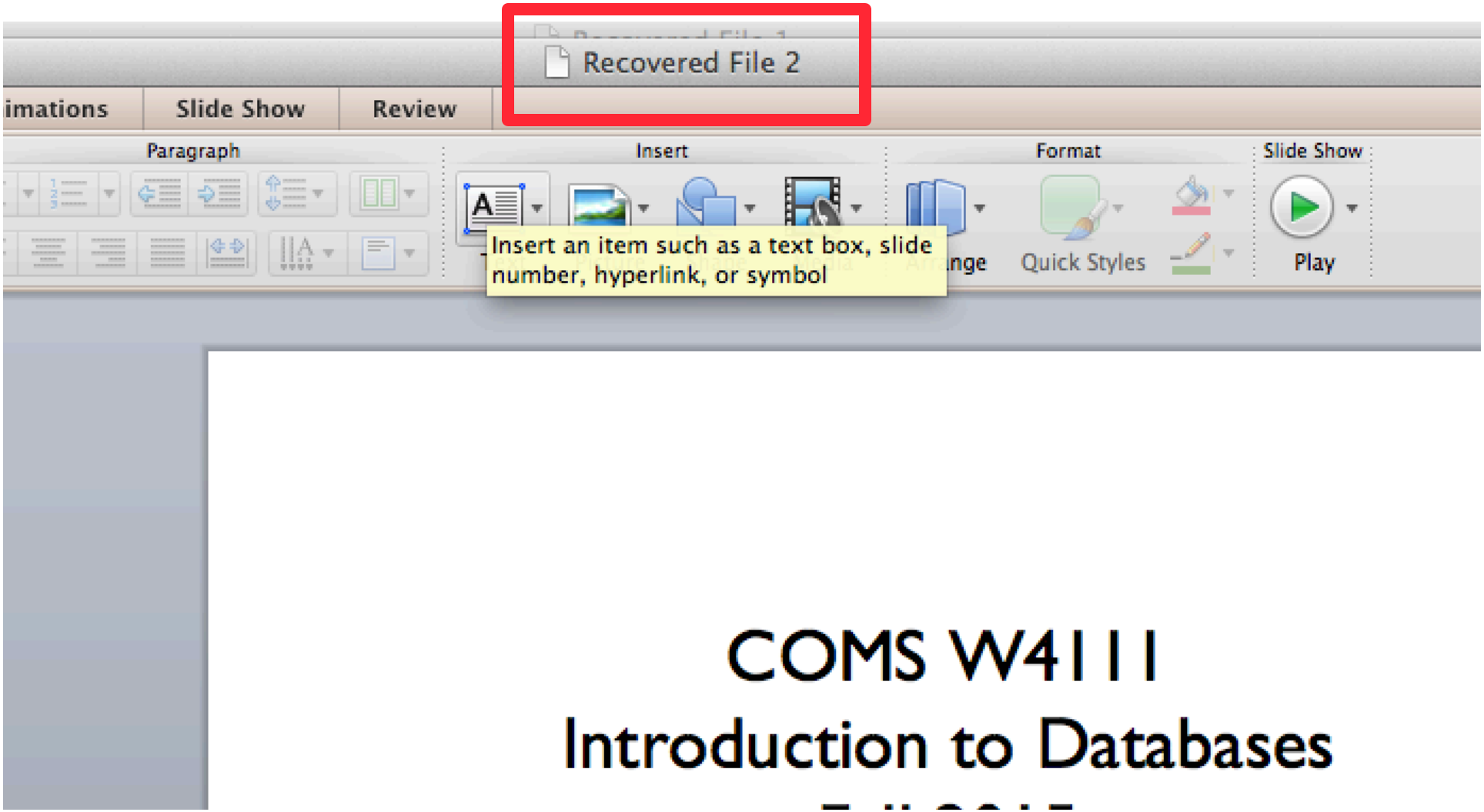
You edit a text file

Computer crashes

What happens?

1. All changes survive
2. No changes survive
3. Changes from last save survive
4. ¯_(ツ)_/¯

Is the file system a DBMS?



Want *Guarantees* from DBMS

You want to write a hot new app on a DBMS.
What do you *not* want to worry about?

Failures disk, machine, human, corruption

Lots of users

Ad-hoc data access

Data formats csv? tsv? custom format?

Database Management System (DBMS)

System to safely and reliably store lots of persistent structured data that is convenient for multiple users to efficiently access and modify.

Database Management System (DBMS)

Safe	Consistent and correct data after failures
Reliable	99.99+% Uptime
Lots	>>RAM (terabytes)
Persistent	Lives longer than one program/process
Convenient	Physical Independence. Declarative.
Multiple Users	Concurrent access. Access control.
Efficient	*Fast*: 100k+ queries / sec

DBMSes in the Wild

Classic Relational

- $$: Oracle, IBM, Microsoft, Teradata, EMC, etc
- Free: MySQL, PostgreSQL

New Relational

- In-Memory, Column-store, Streaming

Non-traditional

- Search (Google, Bing, Lucene), Scientific, Geographic

NoSQL

- Big Data: Hadoop, Spark, etc
- Key-value: Mongo, BerkeleyDB, Cassandra, etc

DBMS-as-a-Service

- Microsoft Azure, Amazon Redshift/RDS, etc…

Encompasses most of CS

OS	DBMS directly manages hardware
Languages	SQL is a domain specific language
Theory	Algorithms, models, NP-complete
AI/ML	Knowledge Discovery
Logic	Relational Algebra = 1st order logic

Good time to learn!

Cloud programmer

Data science

Data engineer

Machine learning engineer

DATA

Data Scientist: The Sexiest Job of the 21st Century

by Thomas H. Davenport and D.J. Patil

FROM THE OCTOBER 2012 ISSUE

2 Key Concepts

Data Independence

Declarative Languages

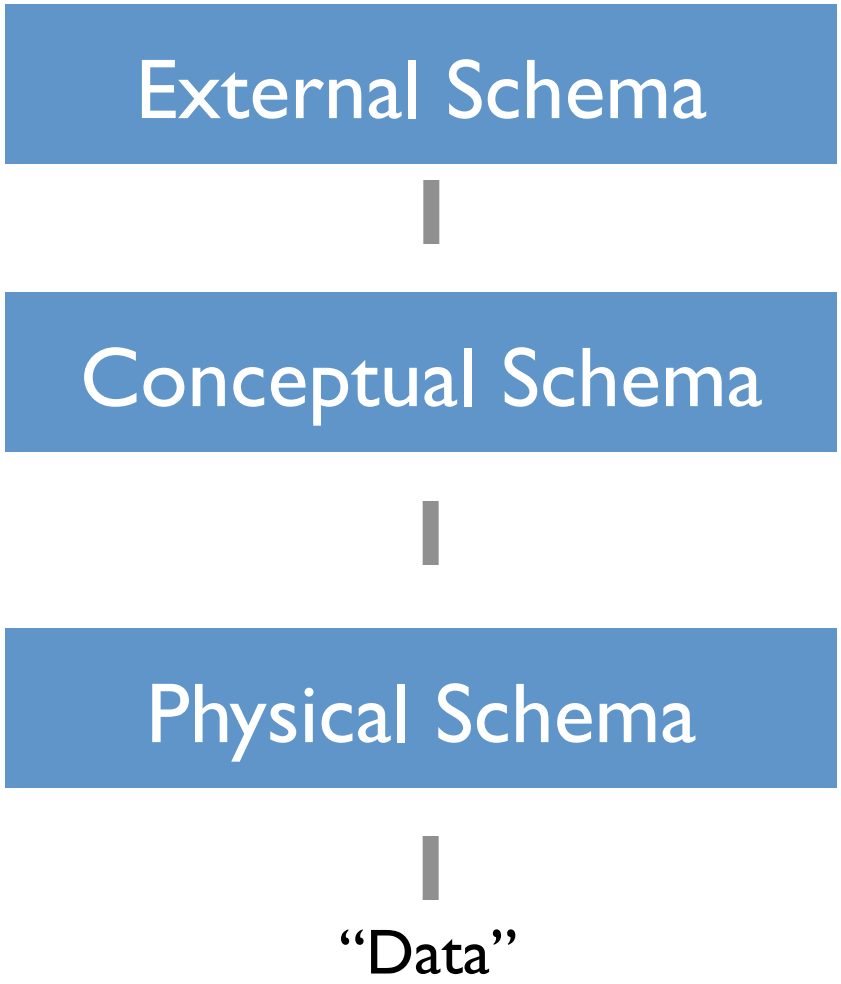
Serve to insulate application programmers from the system implementation

Data Independence

External Schema
Describe how users see data

Conceptual Schema
Describes logical structure

Physical Schema
Describes files and indexes



Example App: Guuber

Users(uid int, name str, age int)

Drivers(did int, name str)

Rides(uid int, did int, distance float, drive_time float)

Data Independence

UID	Name	Age
0	Eugene Wu	17
1	Luis Gravano	20
2	Ken Ross	30

0,Eugene Wu,17
1,Luis Gravano,20
2,Ken Ross,30

CSV File

What is the number of adults?

Data Independence

UID	Name	Age
0	Eugene Wu	17
1	Luis Gravano	20
2	Ken Ross	30

0,Eugene Wu,17
1,Luis Gravano,20
2,Ken Ross,30

CSV File

```
n = 0
for line in csv_file:
  attributes = line.split(“,”)
  if attributes[2] >= 18:
    n += 1
```

Data Independence

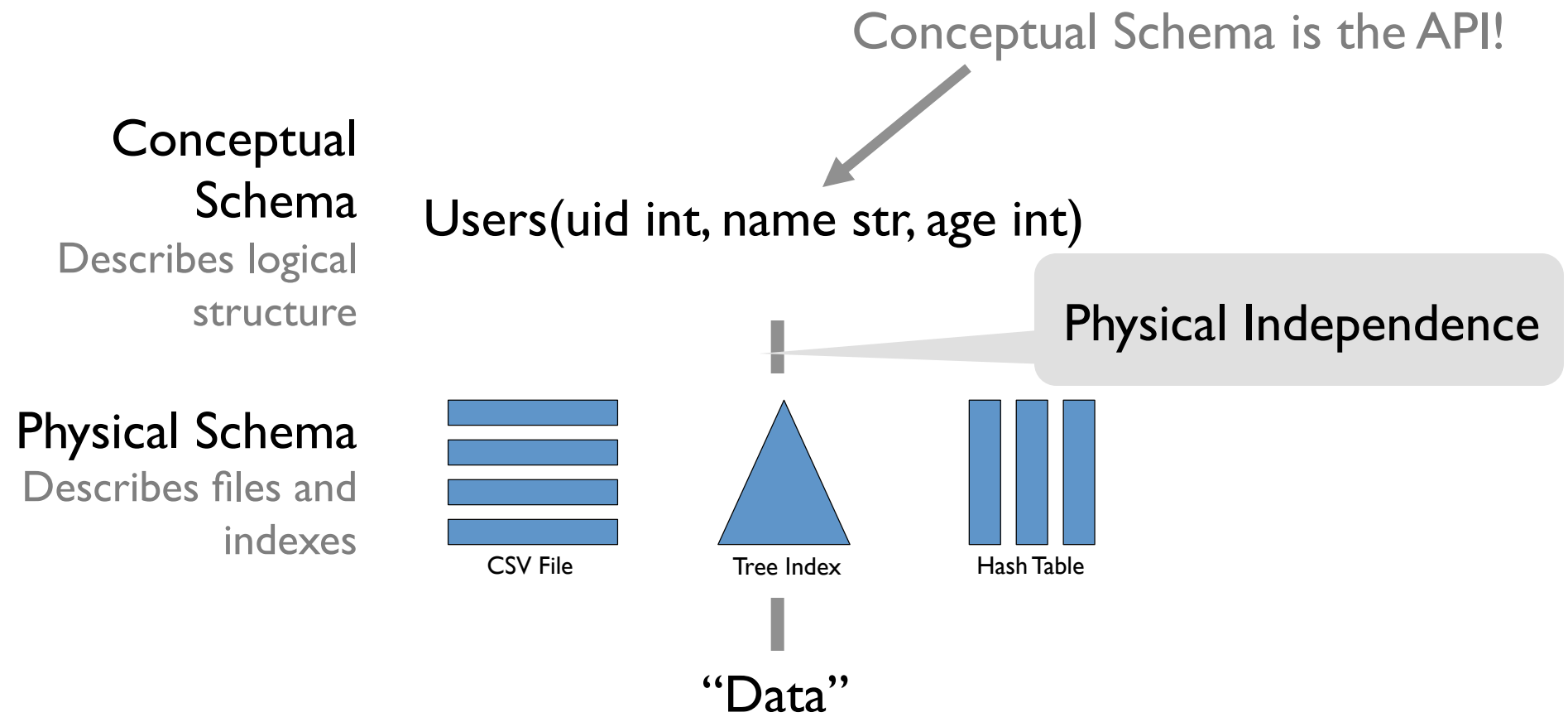
UID	Name	Age
0	Eugene Wu	17
1	Luis Gravano	20
2	Ken Ross	30

0,1,2
Eugene Wu,Luis …
17,20,30

CSV File

```
n = 0
For line in csv_file:
  attributes = line.split(“,”)
  if attributes[2] >= 18:
    n += 1
```

Data Independence



Data Independence

Users(uid int, name str, age int)

"Welcome back Mr. Wu"

Data Independence

Users(uid int, fname str, lname str, age int)

"Welcome back Mr. Wu"

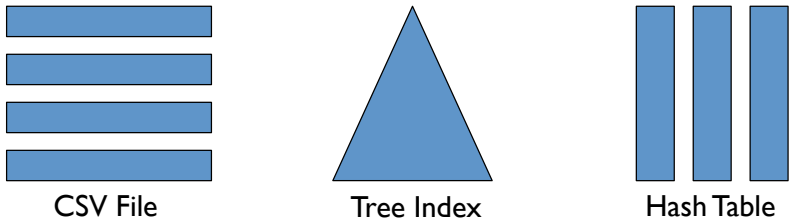
Data Independence

Conceptual Schema
Describes logical structure

Users(uid int, name str, age int)

Physical Independence

Physical Schema
Describes files and indexes



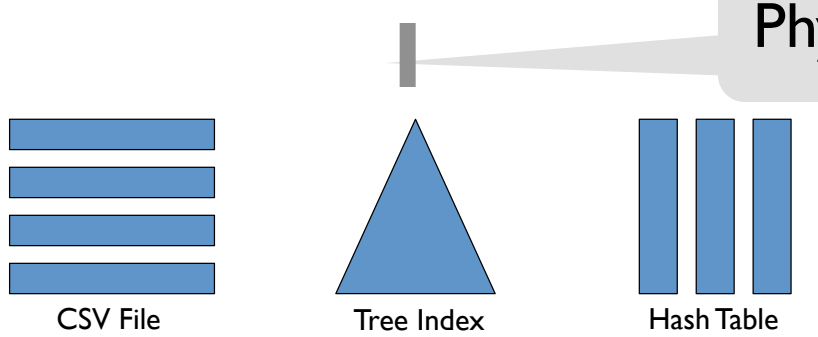
"Data"

Data Independence

Conceptual Schema
Describes logical structure

Users(uid int, fname str, lname str, age int)

Physical Schema
Describes files and indexes



Physical Independence

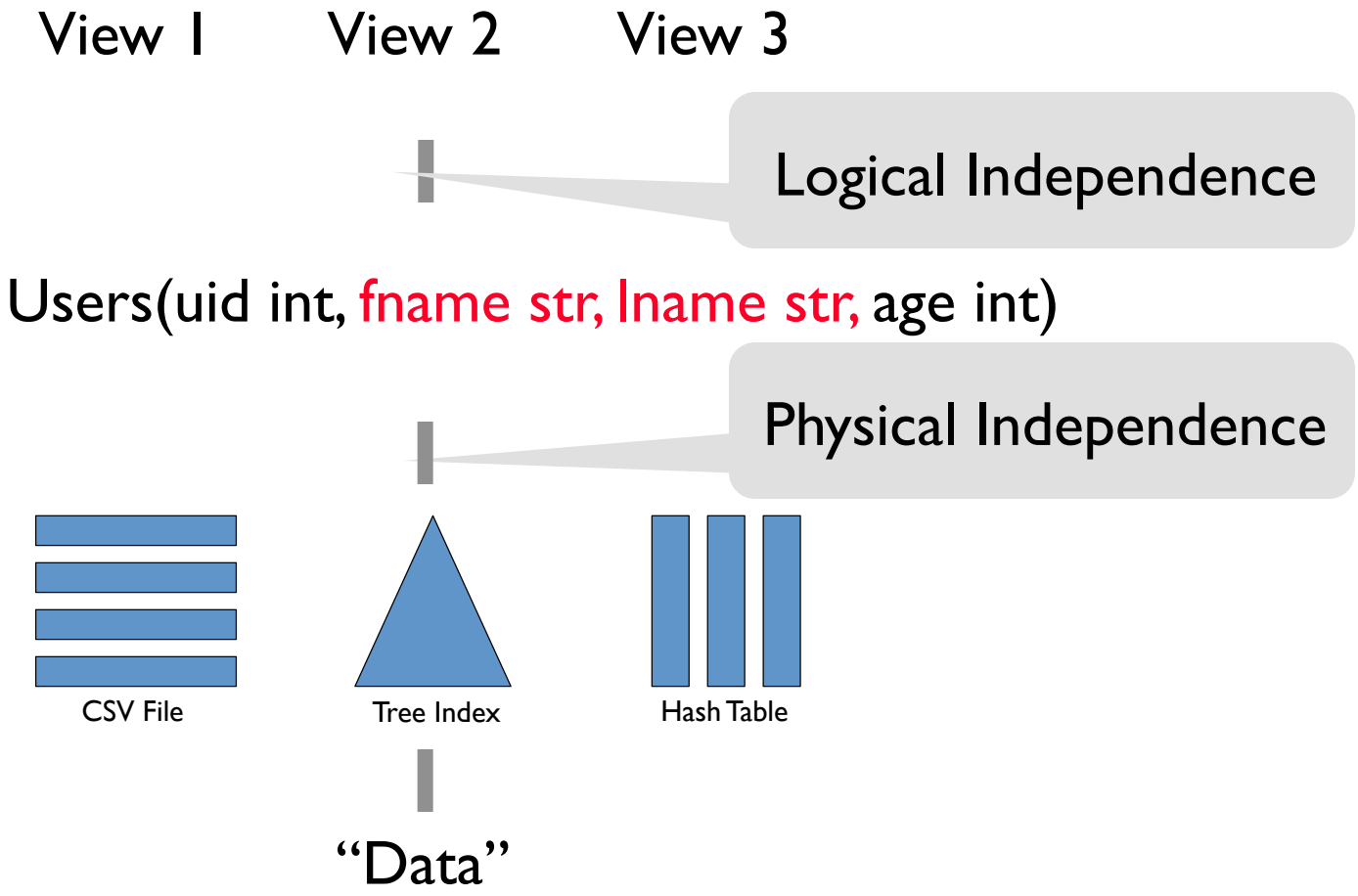
"Data"

Data Independence

External Schema
Describe how users see data

Conceptual Schema
Describes logical structure

Physical Schema
Describes files and indexes



Data Independence

Physical Independence

Protection from changes in physical structure of data

Logical Independence

Protection from changes in logical structure of data

One of most important properties of a DBMS

Declarative

What you want, **not how to do it.**

"Get me a sandwich"

"Take two slices of wheat bread out of the 2nd shelf, put them next to each other…"

Go to store and buy BLT
Make PB&J
Get Falafel delivered

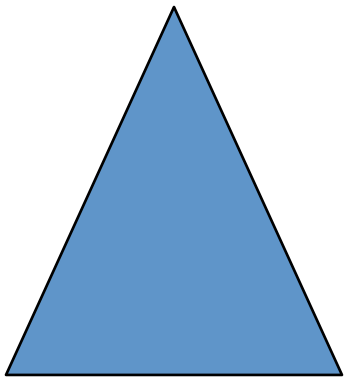
What if on 1st shelf?
Out of wheat bread?
No counter space?

Declarative

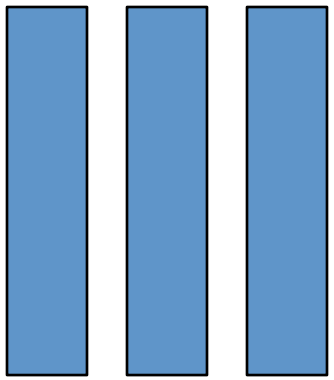
"I want W4111 instructors for Spring 2016"

DBMS

CSV File



Tree Index

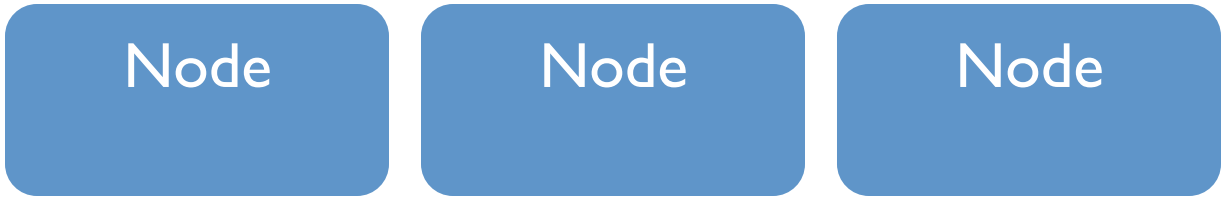


Hash Table

Declarative

"I want W4111 instructors for Spring 2016"

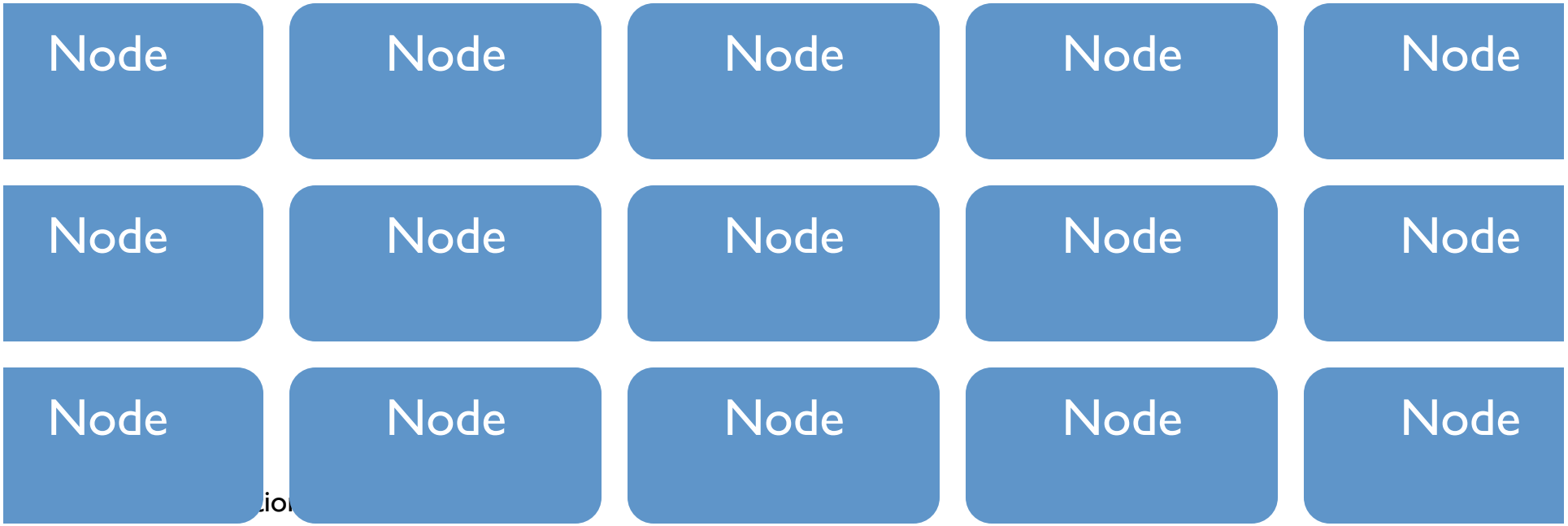
DBMS



Declarative

"I want all highly rated fast drivers"

DBMS



Classic Components in Databases

Durability (Recovery and Logs)

Transactions

Concurrency Control

Atomicity

Durability (Write-Ahead Log)

After crash: Want data to be accessible

Idea: Write modifications to a *log*, then apply to data structures

Each record allows redo/undo entire action

Transaction: Execution of a DB Program

Def: *atomic* (indivisible) sequence of DBMS actions

```
Begin;
<read Evan's account>
<deduct from Evan's account>
<increase Eugene's account>
Commit;  (or Abort;)
```

Transaction: Execution of a DB Program

Def: *atomic* (indivisible) sequence of DBMS actions

Each fully executed transaction must leave DB in *consistent state* if DB is consistent before transaction

Concurrency Control

Concurrently running multiple user programs needed for good performance

- Disks and networks are slow. Keep CPU working
- Use multiple CPUs

Concurrency can cause inconsistencies

- e.g., check cleared while account balance being computed.
- *Really* hard to ensure correctness

DBMS hides concurrency: write "single user" program

Scheduling Concurrent Transactions

Transactions T1, …, Tn are run concurrently

Equivalent to a *serial* ordering (as if no concurrency)

Locks: T_i requests and waits for lock before read/write.

e.g., T_i locks the database, updates, then releases

e.g., T_i locks the table, updates, then releases

e.g., T_i locks rows, updates, then releases

Will talk about how this works later in course.

Atomicity

Def: Xact fully completes, or never happened

even after failures e.g., crashes

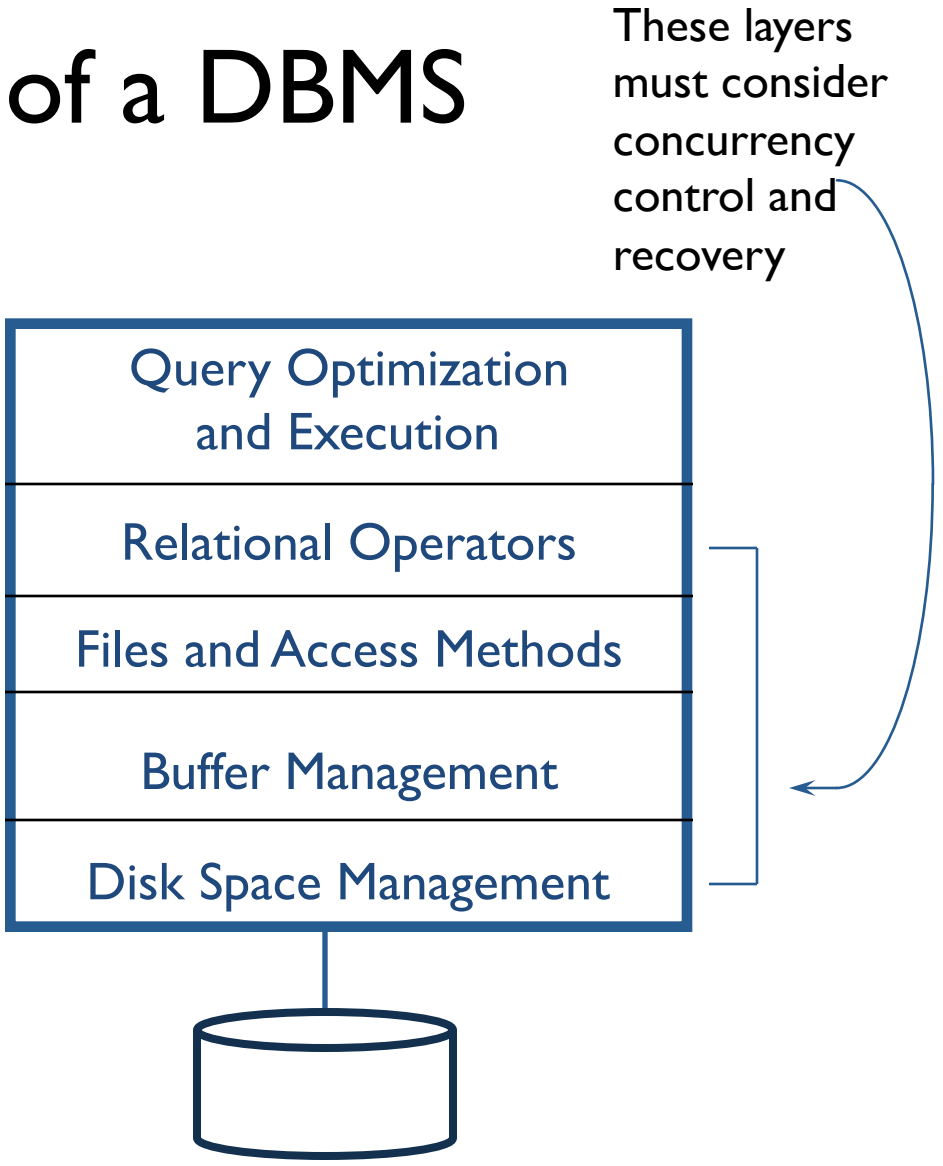
Record all actions Xact did during execution in a log

1. Write ahead logging: before making any change, ensure the change is safely recorded in log
2. After failure, read log and undo any incomplete Xacts

Classic Structure of a DBMS

Typical layered architecture
- DBMS, not OS, manages memory and disk

Doesn't show concurrency control & recovery components



Database Courses at Columbia

COMS W4111 - Intro to Databases

Prerequisites: CS3137 or CS3134; fluency in Python

Intro to DBMSes

Data Models Entity-relation, Relational, …

Relational Algebra

SQL

Applications + SQL cursors, APIs, embedded …

Normalization

Peek at DBMS internals:

- Storage and indexing
- Query optimization
- Transaction Processing

COMS W4112-Database Sys. Impl.

Storage Methods and Indexing

Query Processing and Optimization for 1NF Relations, including external sorting

Materialized Views and Use in Query Optimization

Query Processing and Optimization for ORDBMSs

Transaction Processing and Recovery

Parallel & Distributed DBMSes: Query Proc. and Optimization

Parallel and Distributed Databases: Transaction Processing

Performance Considerations Beyond I/Os

COMS E6111-Advanced Databases

Prerequisites: CS4111; fluency in Java or Python

Information Retrieval
Web Search
Distributed Information Retrieval and Web Search
Data Mining
Data Warehousing, OLAP, Decision Support
Information Extraction
Scalable Visualization and Interaction
Supporting data analysis
Exploration, explanation and exhibition techniques

Administrivia

Next Up

Set up your environment on the cloud

HW0 is out.

Due by Thursday 8:40 AM

Your Instructor: **Eugene Wu**

B.S. @U.C. Berkeley

Ph.D. @MIT

PostDoc @U.C. Berkeley

Assistant Professor since Fall 2015

Databases, visualization, data analysis

data cleaning, crowdsourcing.

Your Instructor: **Eugene Wu**

Contact

http://www.cs.columbia.edu/~ewu
ewu@cs.columbia.edu
7LW1A Schapiro CEPSR
1-212-939-7088

Office hours

Weds 4-5PM
By appointment by email

Class Resources

Class web page

http://www.cs.columbia.edu/~coms4111

Discussion board

piazza (linked from courseworks/website)

Announcements from class staff:

CourseWorks → @columbia.edu email

Your TAs

Anthony Dubis
Mengqing Wang
Jincheng Li
Sania Arif

All TA office hours in CS TA Room (see class web page)
TA office hours will be posted on class web page

Class Information: Prerequisites

COMS W3134 - *Data Structures in Java* or
COMS W3137 - *Data Structures and Algorithms*

(equivalent courses taken elsewhere are acceptable as well)

Fluency in Python

You need permission from the instructor if you don't have the prerequisites.

Class Information: Lectures

Mondays and Wednesdays

2:30 - 4 p.m.

833 Mudd

(here)

Grading Information

Midterm:	15%
Midterm 2:	40% (last day of class)
HW:	15% (4 HWs equally weighed)
Project 1:	15%
Project 2:	5%

Median grade: B+ or slightly higher.
Alternative or make-up exams will not be given.
All homework assignments are equally weighted.
Project 1 has higher weight than Project 2.

Exam Dates

Midterm 1: in class

Midterm 2: last day of class, in class

If you cannot make the midterms,

do not take this course

Homework

Homeworks usually due at 8:40AM of due date (before class)

No extensions or exceptions.

Three grace late days for hws throughout the semester.

After using all grace days, 25% grade deduction per late day.

Check full details on web site.

Projects (more details soon)

Two projects.

- Teams of two
- Run on Microsoft Azure cloud infrastructure
- Get CS account if your team doesn't have a computer
- Language is Python

Project 1

- Model and build your own database web application
- Explore "traditional" relational database features.

Project 2

- TBD

Projects (cont.)

No extensions or exceptions for project submission.

3 grace late days total for project.

After using all grace days, 25% grade deduction per late day.

Check full details on web site.

Collaboration Policy

Read Syllabus on course site for allowed conduct

CS Dept academic honesty policies
http://www.cs.columbia.edu/education/honesty

We will not tolerate *any* cheating

Collaboration Policy

Discussing lectures and course material strongly encouraged

Homework and exams are *individual*. No exceptions
Any libraries or code however minor must be disclosed.

Projects are done in teams; no collaboration between teams.

Contact the instructor right away if you have any questions or are falling behind.

Textbook

Raghu Ramakrishnan, Johannes Gehrke: *Database Management Systems*, 3rd edition, McGraw-Hill, 2002

Available from

Bookculture bookstore 536 W. 112th St.

Online retailers

Upperclass-persons

On reserve in Engineering Library

Contests and Rewards

"A+ letter-and-lunch" reward

- Students who earn an A+ will get
- Personal letter of congratulations from the instructor.
- Lunch at Faculty House with instructor and A+ students
- Top candidates for cs4111 TA positions in the future.

Project 1 contest

- Four best projects chosen as contest winners.
- Winners get:
  - Option to discuss and demo your project in class.
  - 10% boost in your Project 1 grade.

On-going Feedback

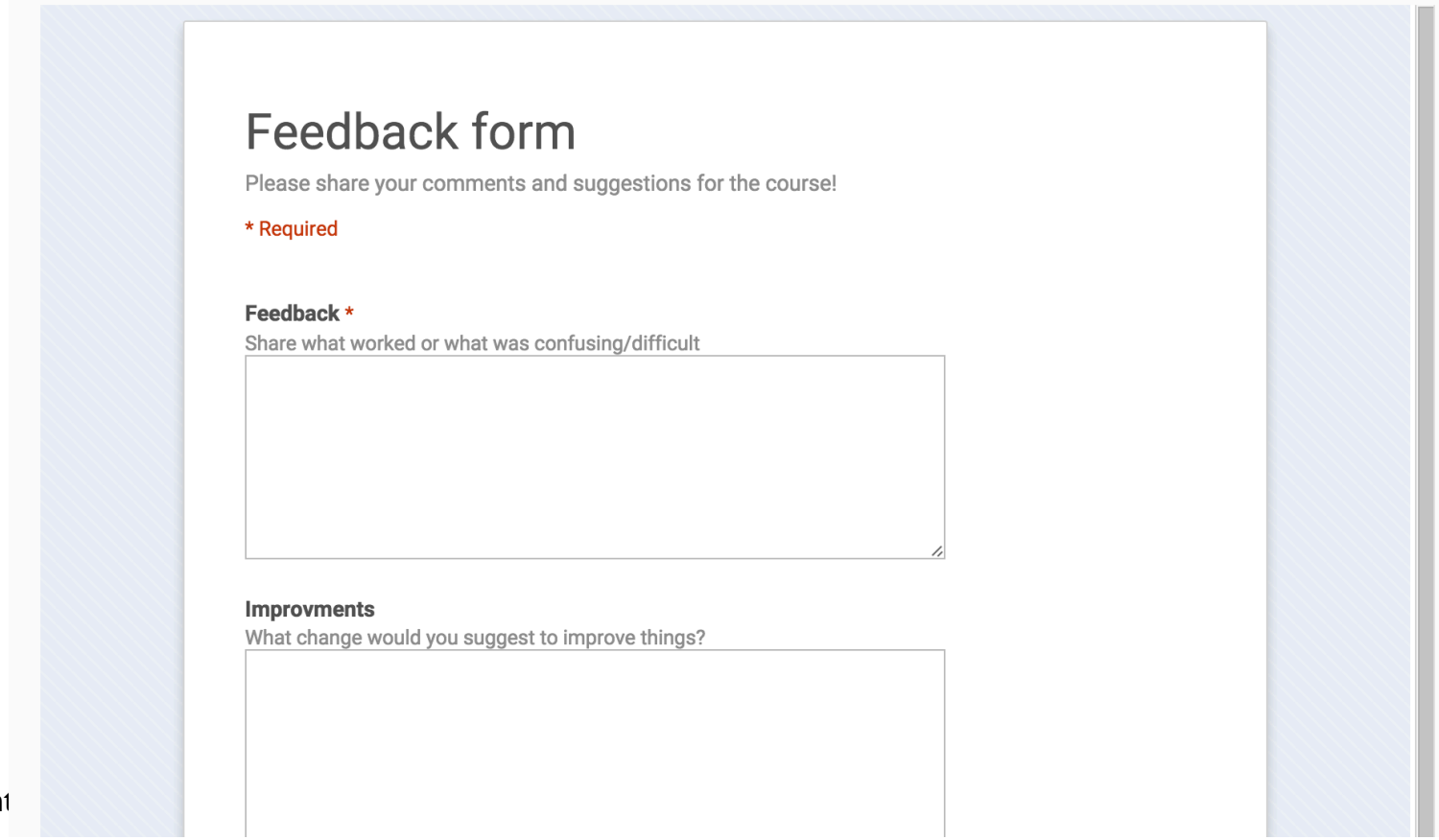
Please provide feedback throughout the course.

- What is useful or confusing in lecture
- Thoughts about software stack
- Thoughts about assignments

Email me, come to office hours, talk to staff or:

On-going Feedback

Use form on website



Slides borrow material from
Prof. Gravano
Prof. Hellerstein & Franklin@Cal
Prof. Madden & Stonebraker@MIT

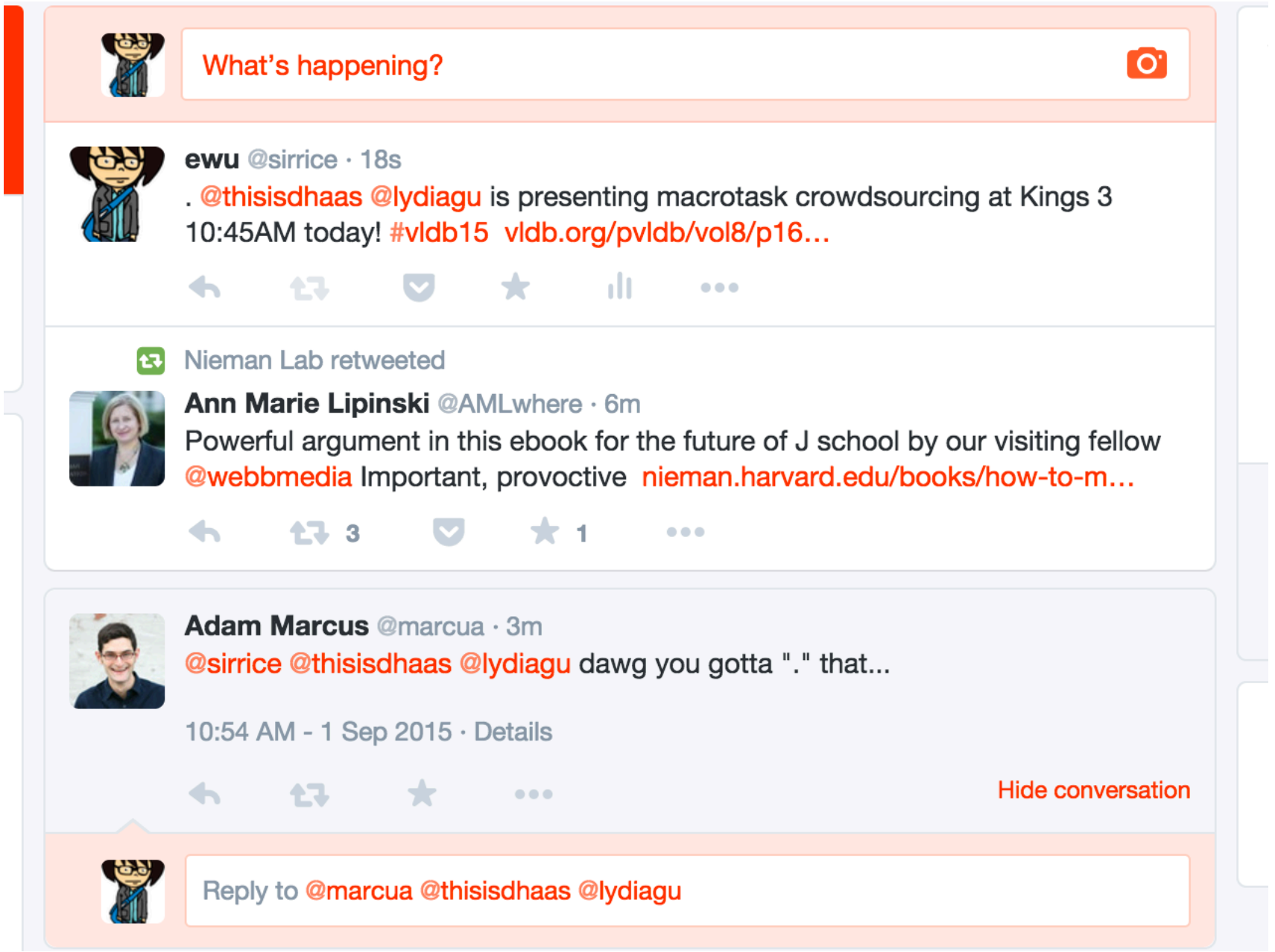
(and by transitivity Raghu Ramakrishnan and Johannes Gehrke)

Useful info

http://www.cs.columbia.edu/~coms4111

ej@evanjones.ca

DO HOMEWORK 0!


What's happening?
ewu @sirrice · 18s
. @thisisdhaas @lydiagu is presenting macrotask crowdsourcing at Kings 3 10:45AM today! #vldb15 vldb.org/pvldb/vol8/p16…
Nieman Lab retweeted
Ann Marie Lipinski @AMLwhere · 6m
Powerful argument in this ebook for the future of J school by our visiting fellow @webbmedia Important, provoctive nieman.harvard.edu/books/how-to-m…
3
1
Adam Marcus @marcua · 3m
@sirrice @thisisdhaas @lydiagu dawg you gotta "." that...
10:54 AM - 1 Sep 2015 · Details
Hide conversation
Reply to @marcua @thisisdhaas @lydiagu