

W4111

# Introduction to Databases

## Fall 2022

Computer Science Department  
Columbia University

# Welcome!

## Eugene Wu

B.S.                  U.C. Berkeley

Ph.D.                MIT

PostDoc              U.C. Berkeley

Professor           Columbia since Fall 2015

Database systems, vis, data analysis, cleaning, ML systems, crowdsourcing.

[www.eugenewu.net](http://www.eugenewu.net)

ewu@cs.columbia.edu

421 Mudd

Office hours

Thurs 12-1PM In-person/zoom.  
or by appointment

# Agenda

Overview

Course Info

Entity-Relationship Modeling

# The Future of AI: How Artificial Intelligence Will Change the World

AI is constantly changing our world. Here are just a few ways AI will influence our lives.



Written by [Mike Thomas](#)



# Artificial Intelligence and the Future of Humans

*Experts say the rise of artificial intelligence will make most people better off over the next decade, but many have concerns about how advances in AI will affect what it means to be human, to be productive and to exercise free will*

BY JANNA ANDERSON AND LEE RAINIE



OPINION

# Andrew Yang: Yes, Robots Are Stealing Your Job

Self-driving trucks will be great for the G.D.P. They'll be terrible for millions of truck drivers.

Nov. 14, 2019



# EU Parliament

## Draft Report on AI in the Digital Age

Argues that artificial intelligence (AI) is the key emerging technology within the fourth industrial revolution; notes that AI is the control centre of the new data layer that surrounds us and which can be thought of as the fifth element after air, earth, water and fire; states that by 2030, AI is expected to contribute more than EUR 11 billion to the global economy, an amount that almost matches China's GDP in 2020;

# Conventional View of AI/Data Science

Lone data scientist uses a static, clean table,  
applies statistics or fits an ML model  
to increase a well-defined score

See popular ML articles, Kaggle competitions, etc

# Conventional View of AI/Data Science

Lone data scientist uses a static, clean table,  
applies statistics or fits an ML model  
to increase a well-defined score

See popular ML articles, Kaggle competitions, etc

# Conventional View of AI/Data Science

Team

Lone data scientist uses a static, clean table,

applies statistics or fits an ML model

to increase a well-defined score

unclear, ill-defined

Huge amount of “unseen labor” (data engineering)  
in order to support real-world data science & ML

# In Reality...

An on-call engineer's biggest nightmare

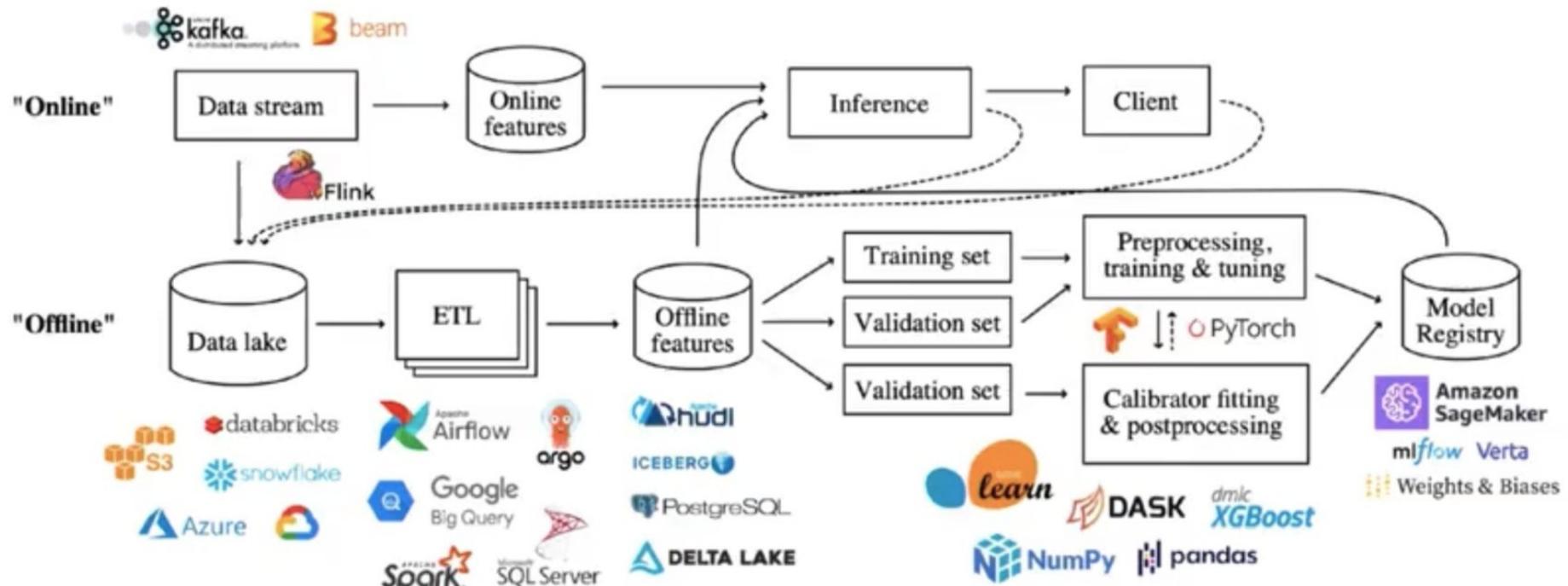


Figure 1: High-level architecture of a generic end-to-end machine learning pipeline. Logos represent a sample of tools used to construct components of the pipeline, illustrating heterogeneity in the tool stack. *Shankar et al. 2021*

<https://www.facebook.com/Engineering/videos/1578607659138164/>

# In Reality...

Data engineering dominates data science projects

Data engineer work >> data scientist work

Data engineering key to ML/AI/data science

# Data Eng Dominates Data Science Projects



**Big Data Borat**

@BigDataBorat

 Follow

In Data Science, 80% of time spent prepare data, 20% of time spent complain about need for prepare data.

Most time spent on data engineering

Often not viewed as sexy, but critical

# Data Eng Dominates Data Science Projects

---

## Hidden Technical Debt in Machine Learning Systems

---

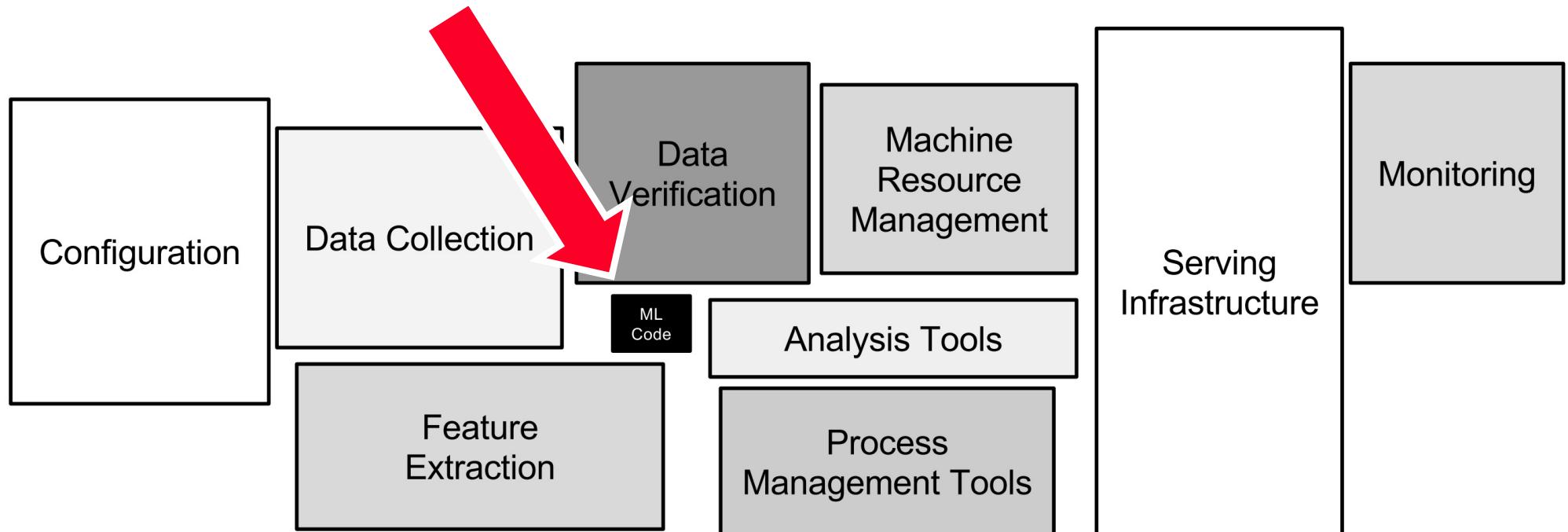
**D. Sculley, Gary Holt, Daniel Golovin, Eugene Davydov, Todd Phillips**  
{dsculley, gholt, dg, edavydov, toddphillips}@google.com  
Google, Inc.

**Dietmar Ebner, Vinay Chaudhary, Michael Young, Jean-François Crespo, Dan Dennison**  
{ebner, vchaudhary, mwyoung, jfcrespo, dennison}@google.com  
Google, Inc.

# Data Eng Dominates Data Science Projects

---

## Hidden Technical Debt in Machine Learning Systems



# Data Eng Dominates Data Science Projects



Andrej Karpathy

@karpathy

...

But as of approx. last two years, even the neural net architectures across all areas are starting to look identical - a Transformer (definable in ~200 lines of PyTorch [github.com/karpathy/minGPT...](https://github.com/karpathy/minGPT)), with very minor differences. Either as a strong baseline or (often) state of the art.

7:03 PM · Dec 7, 2021 · Twitter Web App

# THE DATA SCIENCE HIERARCHY OF NEEDS

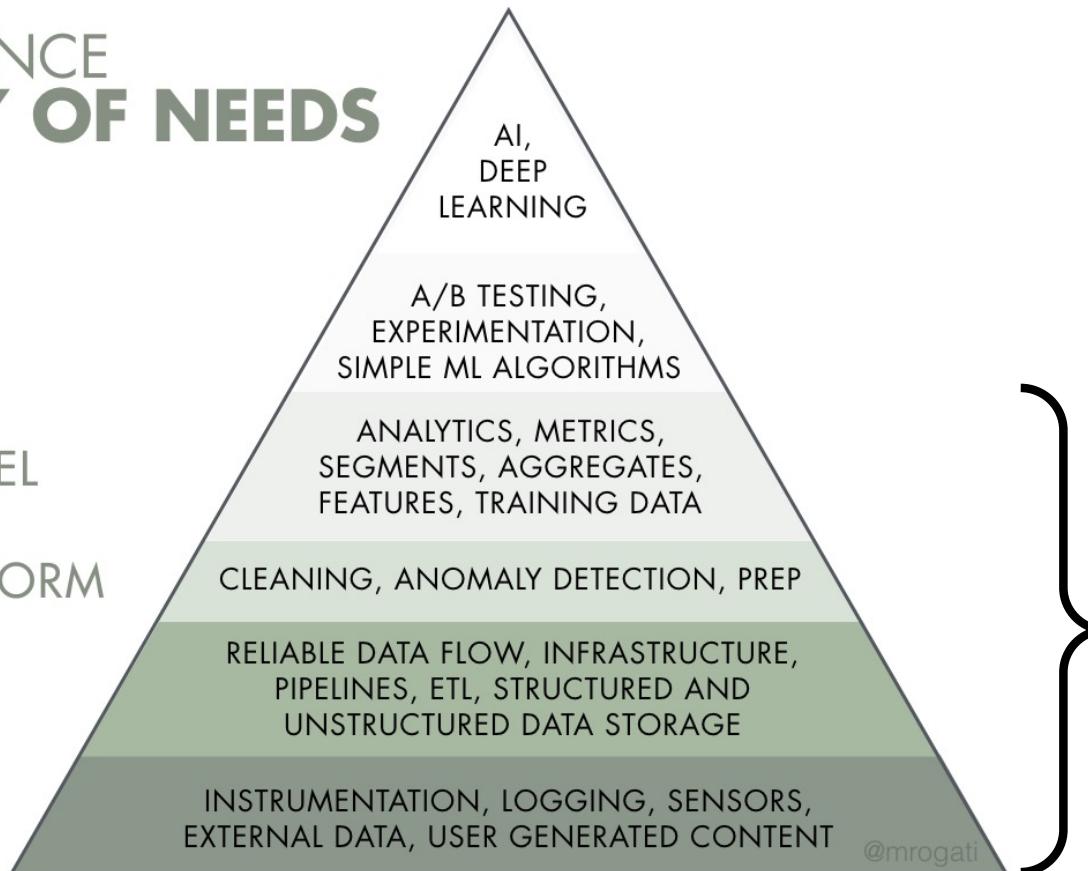
LEARN/OPTIMIZE

AGGREGATE/LABEL

EXPLORE/TRANSFORM

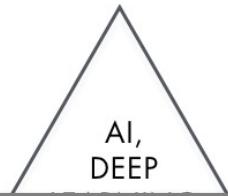
MOVE/STORE

COLLECT



Data eng. work  
that must  
happen first

## THE DATA SCIENCE **HIERARCHY OF NEEDS**



*“However, under the strong influence of the current AI hype, people try to plug in data that’s dirty & full of gaps, that spans years while changing in format and meaning, that’s not understood yet, that’s structured in ways that don’t make sense, and expect those tools to magically handle it.”*

COLLECT

INSTRUMENTATION, LOGGING, SENSORS,  
EXTERNAL DATA, USER GENERATED CONTENT

@mrogati

# Data Engineering as a Job Category

Feb 4, 2019, 08:15am EST | 171,732 views

## Why There Will Be No Data Science Job Titles By 2029



Noah Gift  
Forbes Council  
Forbes Technology Council  
Innovation

We Don't Need Data Scientists,  
We Need Data Engineers

January 2021

*“..70% more open roles at companies in data engineering as compared to data science....”*

# Data Engineering as a Job Category

## Job Opening Estimates from Zippia

Data scientist: 79K 16% growth rate

Data engineer: 170K 21% growth rate

# Overview of Data Engineering Concerns

ETL and Data Warehouses

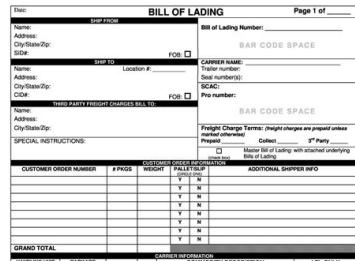
Data lakes

Data Quality

Metadata

# Preparation

PDF



```
tail -f /var/log/apache2/access.log
127.0.0.1 - - [31/Oct/2017:11:11:37 +0530] "GET / HTTP/1.1" 200 729 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/61.0.3163.100 Safari/537.36"
127.0.0.1 - - [31/Oct/2017:11:11:37 +0530] "GET /icons/blank.gif HTTP/1.1" 200 56
127.0.0.1 - - [31/Oct/2017:11:11:37 +0530] "GET /icons/folder.gif HTTP/1.1" 200 56
127.0.0.1 - - [31/Oct/2017:11:11:37 +0530] "GET /icons/text.gif HTTP/1.1" 200 56
127.0.0.1 - - [31/Oct/2017:11:11:38 +0530] "GET /favicon.ico HTTP/1.1" 404 500
127.0.0.1 - - [31/Oct/2017:11:12:05 +0530] "GET /tecmint/ HTTP/1.1" 200 56
127.0.0.1 - - [31/Oct/2017:11:12:05 +0530] "GET /tecmint/favicon.ico HTTP/1.1" 200 56
127.0.0.1 - - [31/Oct/2017:11:12:05 +0530] "GET /icons/back.gif HTTP/1.1" 200 401
127.0.0.1 - - [31/Oct/2017:11:13:58 +0530] "GET /tecmint/Videos/ HTTP/1.1" 200 401
127.0.0.1 - - [31/Oct/2017:11:13:58 +0530] "GET /icons/compressed.gif HTTP/1.1" 200 401
127.0.0.1 - - [31/Oct/2017:11:13:58 +0530] "GET /icons/movie.gif HTTP/1.1" 200 401
127.0.0.1 - - [31/Oct/2017:11:13:58 +0530] "GET /icons/moview.gif HTTP/1.1" 200 401

```

```
client.createAnnotationTask({
  callback_url: 'http://www.example.com/callback',
  instruction: 'Draw a box around each rooftop and pool.',
  attachment: 'http://i.imgur.com/X0JbalC.jpg',
  objects_to_annotate: ['pool', 'rooftop'],
  with_labels: true,
  min_width: 30,
  min_height: 30
}, (err, task) => {
  // do something with task
});
```



Python Scripts,  
Spark/Databricks, etc

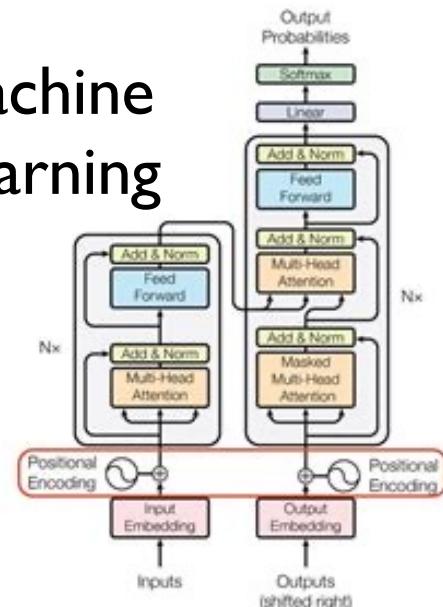
Data Preparation

Transform/clean data into useful form

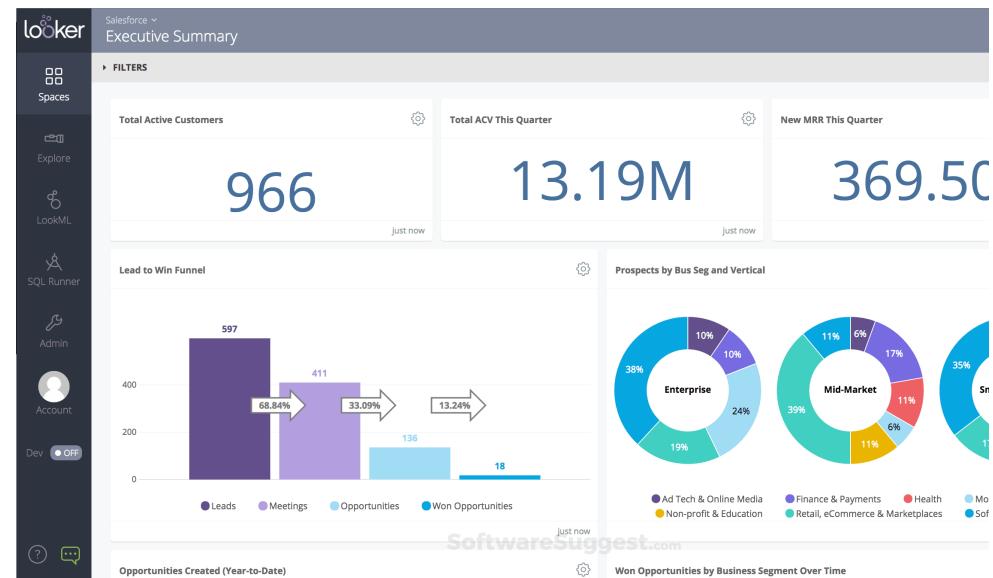
Logs

JSON

Machine  
Learning



Visualization



# Sources



**\$999.00**

Simple Mobile Apple iPhone 11 Pro Max  
Prepaid with 64GB, Space Gray (Locked  
to Carrier- Simple Mobile)

8

Save with W+

2-day shipping



+ Add

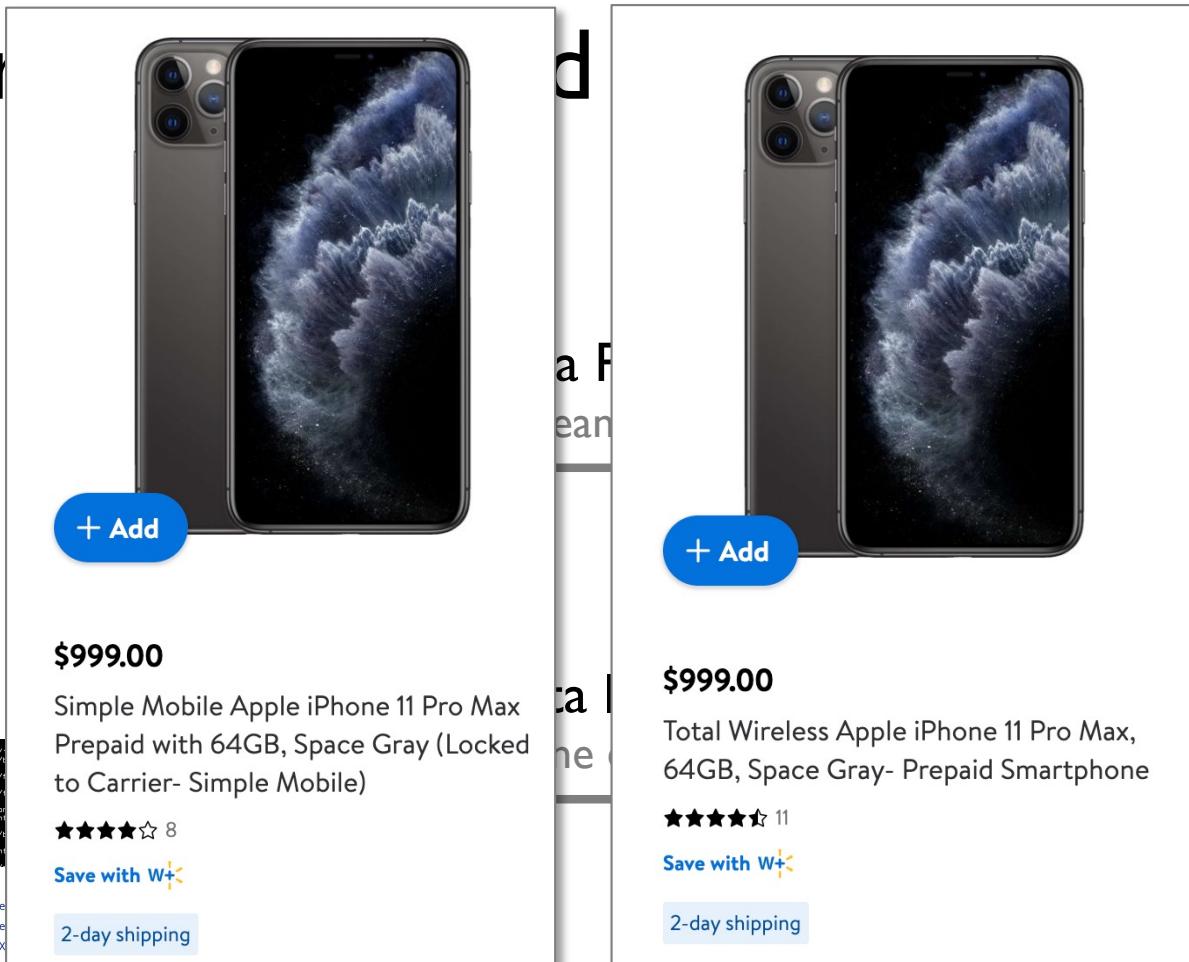
**\$999.00**

Total Wireless Apple iPhone 11 Pro Max,  
64GB, Space Gray- Prepaid Smartphone

11

**Save with W+**

2-day shipping



# Use Cases

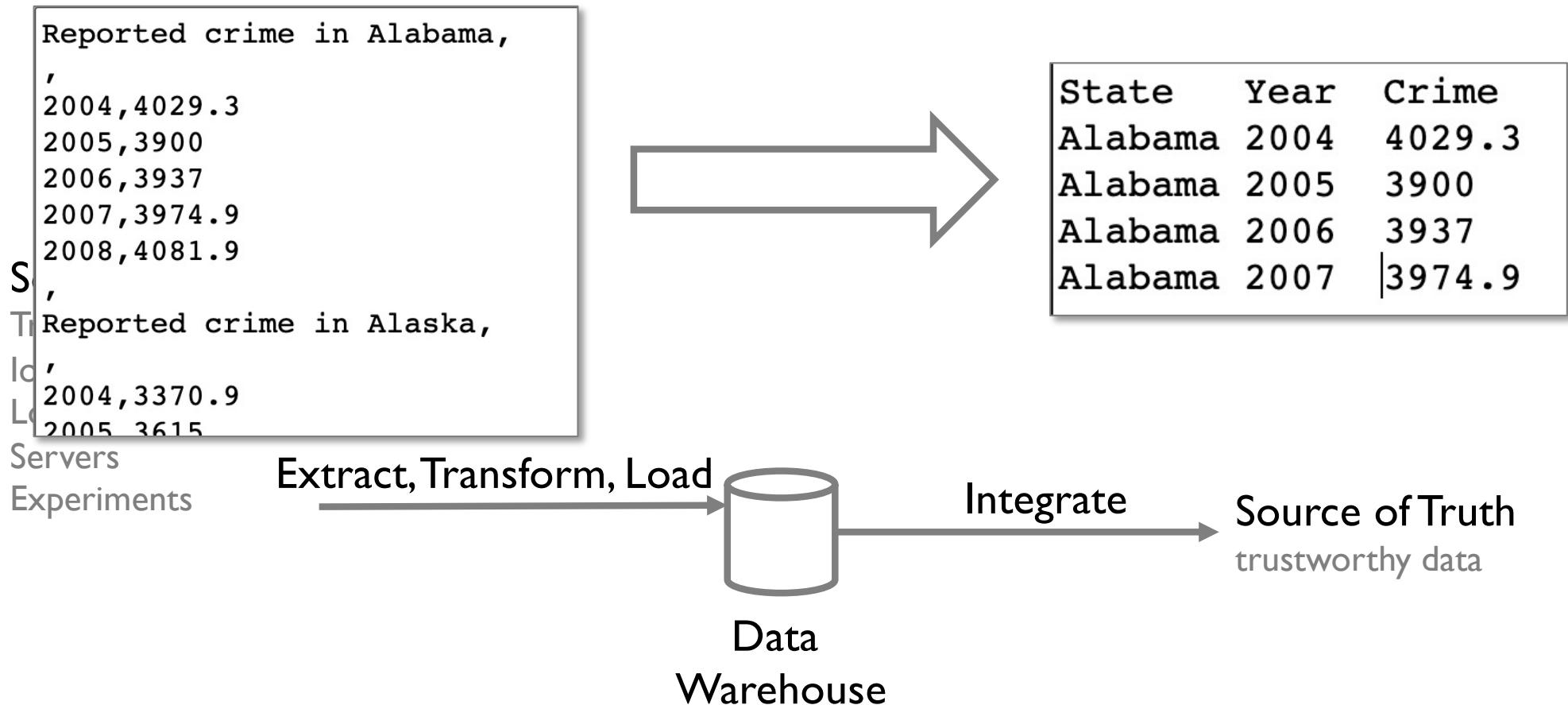
# AI, ML, Data science, Apps, Webservices

# Source of Truth

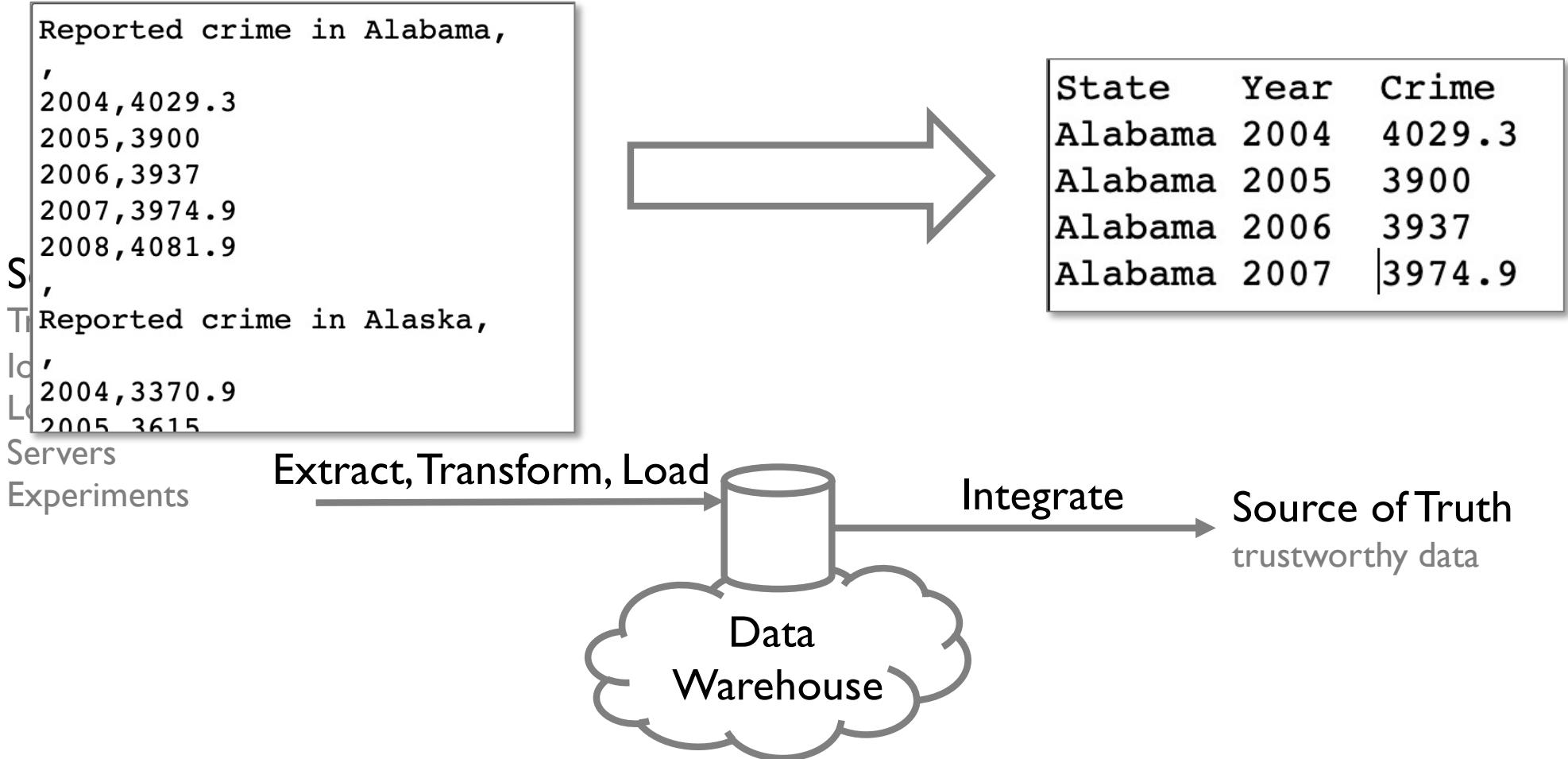
## trustworthy data

	2004	#	2005
	4829.3		3988
	3378.9		3615
2 Arizona	5873.3		4827
3 Arkansas	4833.1		4668
4 California	3423.9		3321
5 Colorado	3918.5		4841
6 Connecticut	2684.9		2579
7 Delaware	3283.6		3118
8 District of Columbia	4852.8		4490
9 Florida	4182.5		4613

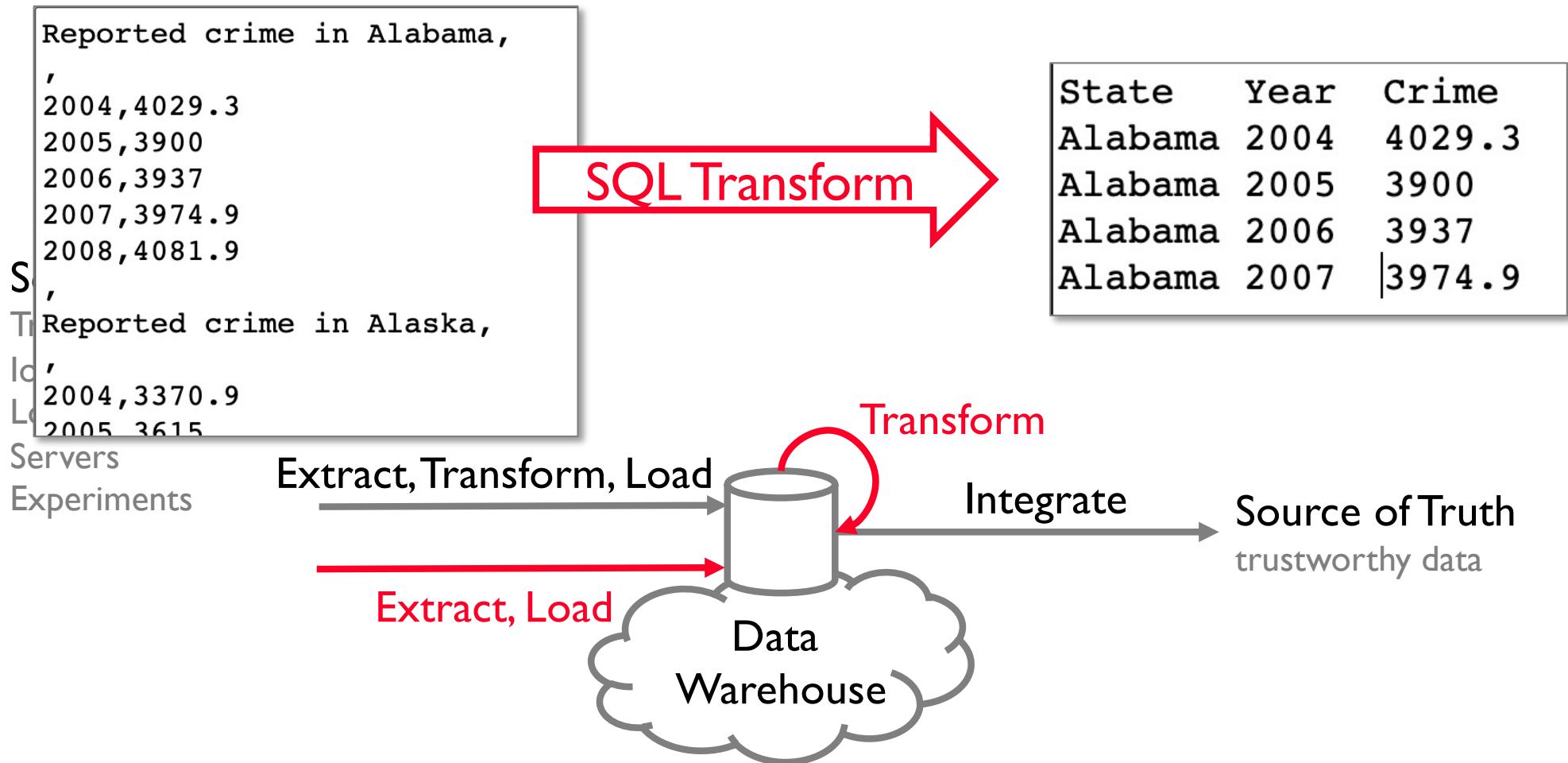
# ETL: Extract Transform Load (1980s)



# ELT: Extract Load Transform (2010s)

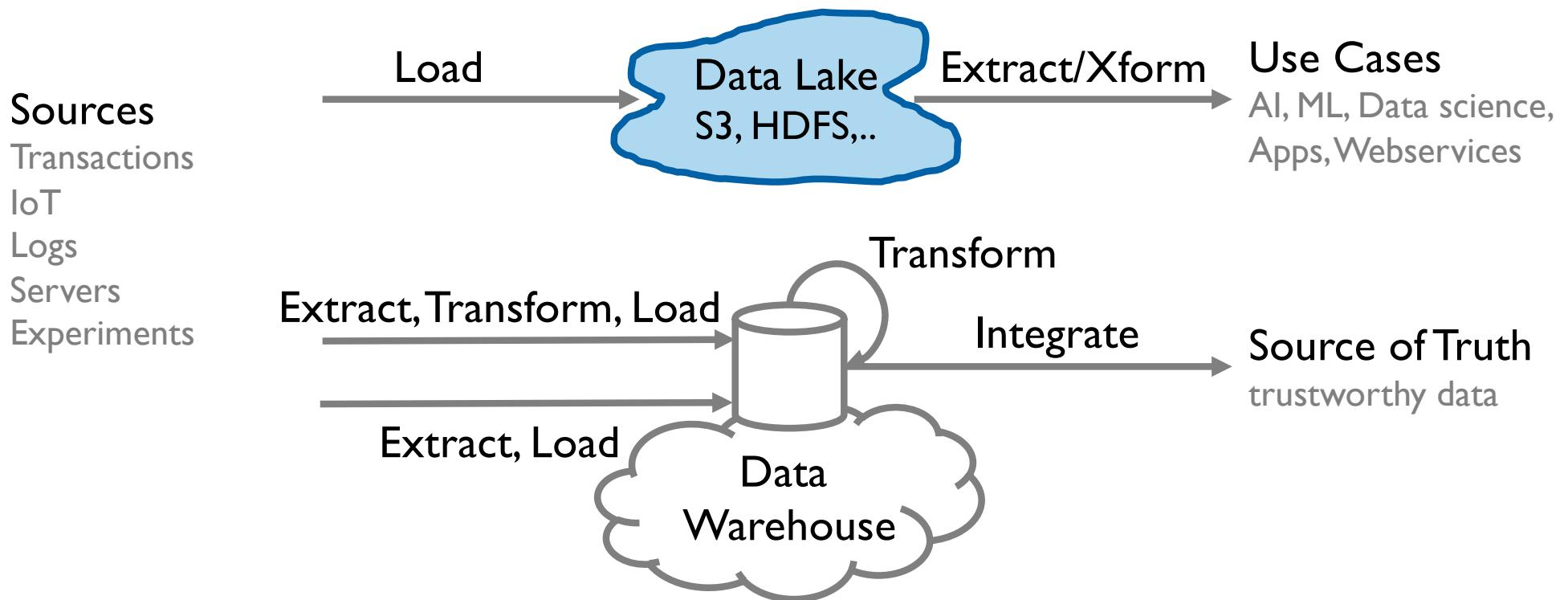


# ELT: Extract Load Transform (2010s)

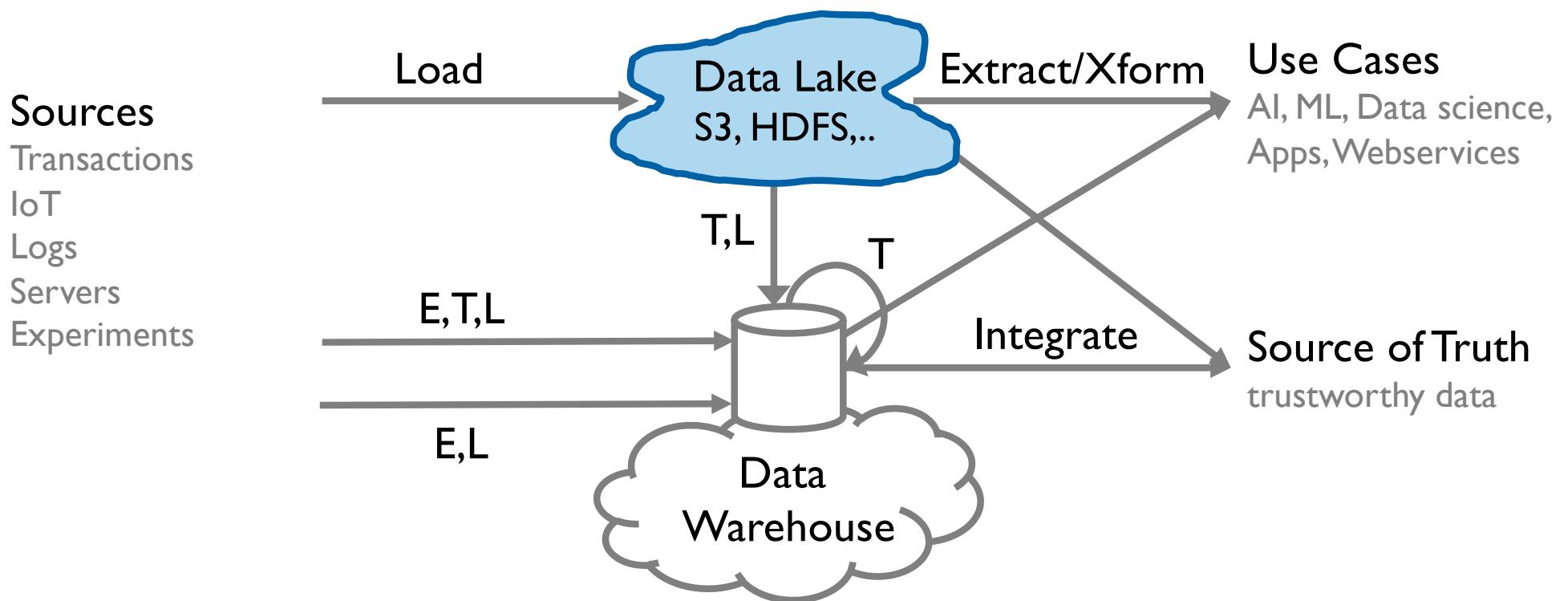


# Data Lakes (2000s)

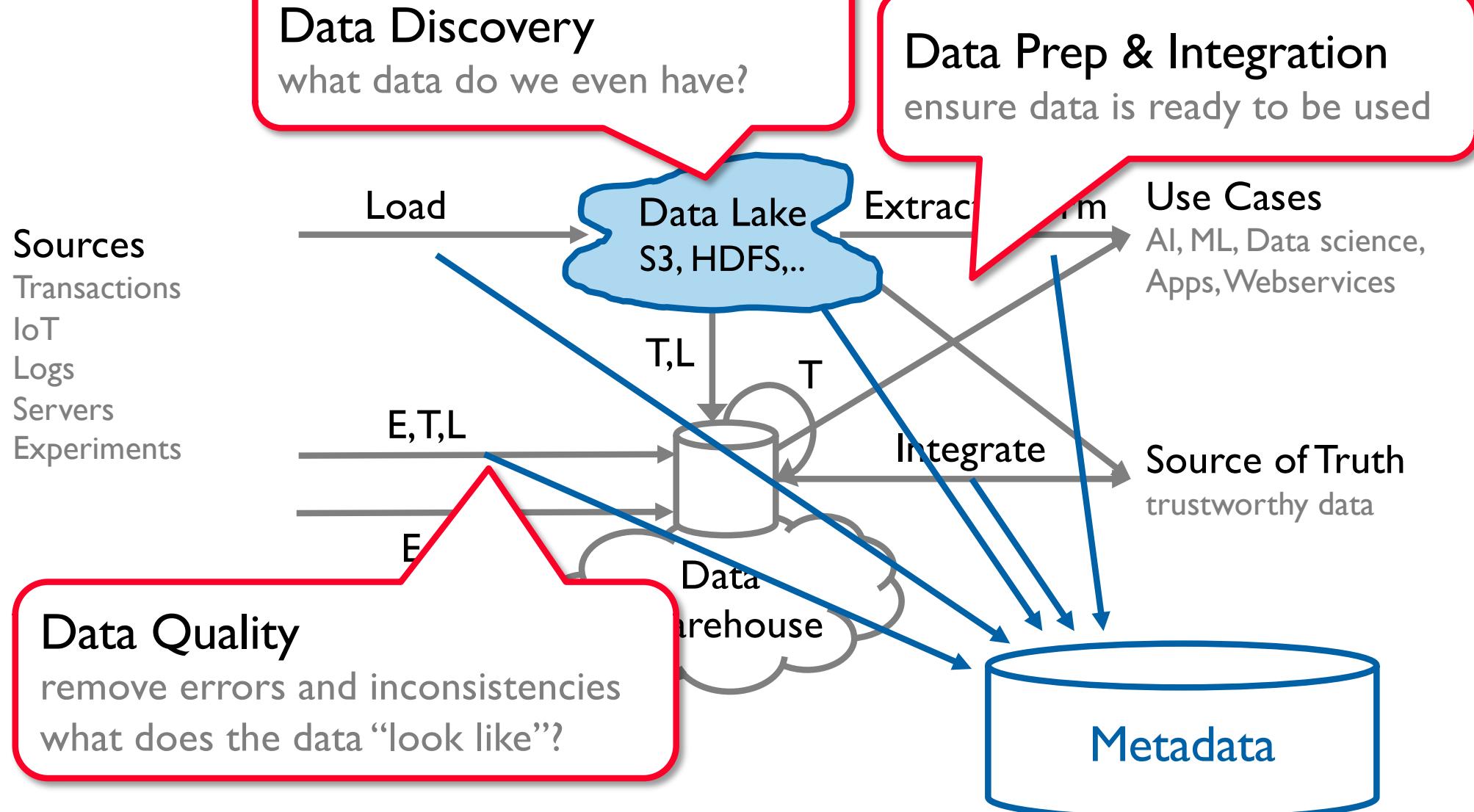
Store as files, transform when needed

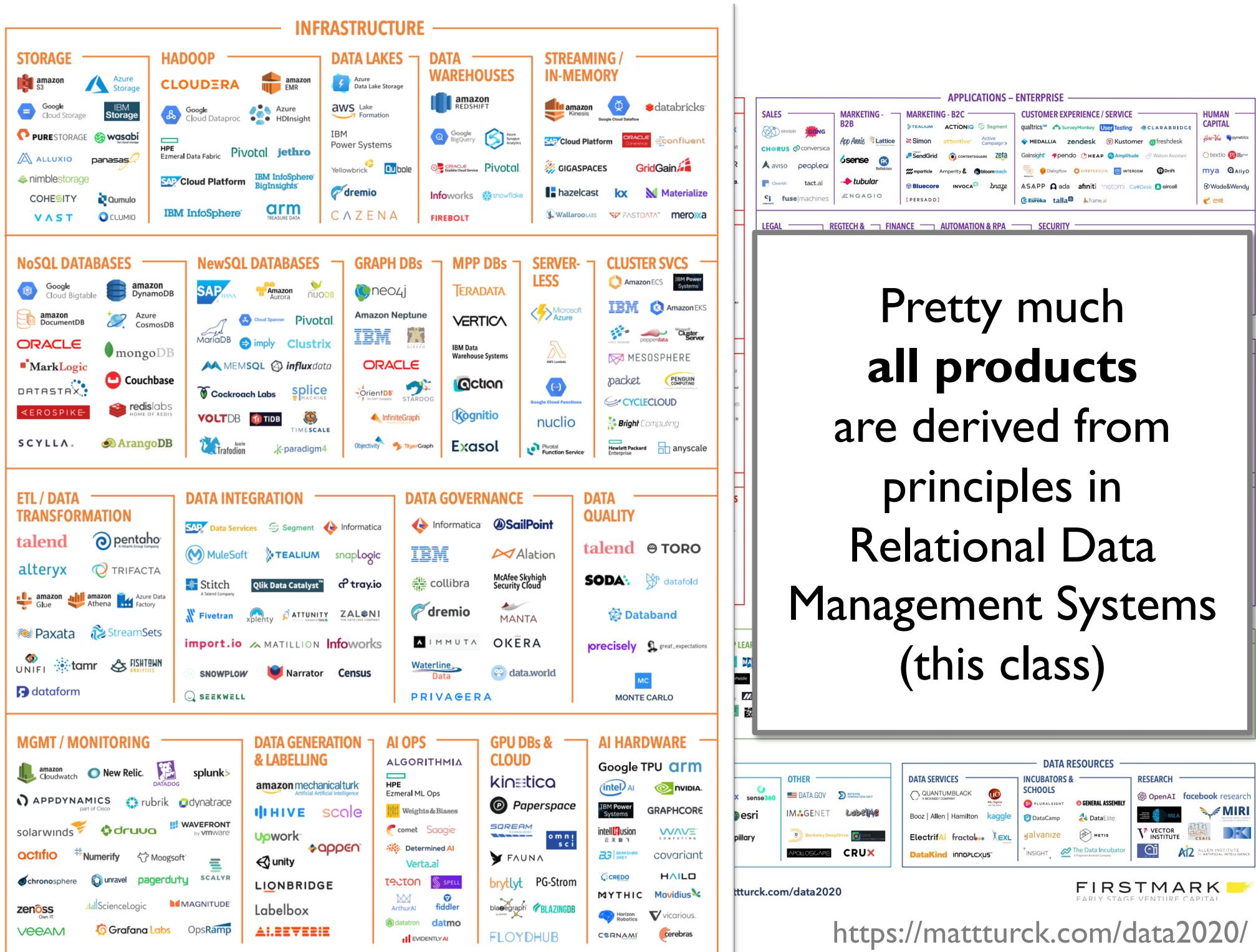


# Everything Everywhere All At Once (2010s)



# Understanding the Data





# 4111: Intro to Relational Data Management Systems

What's a database?

What's a database management system (DBMS)?

What are the core ideas?

# What is a Database?

	A	B	C	D	E	F	G	H	I	J	K
1	color	date	slug	title	lshow	link	readings	optional	assigned	ashow	due
2	white	21-Jan	Intro + ER Models			https://github.com/w4111/hw0	Ch 1, 2		< a href="https://github.com/w4111/hw0" > HW 0 </a >		
3	#e7f8ff	28-Jan	ER Models				Ch 2		< a href="https://github.com/w4111/hw1-s22" > HW 1 </a >	0	HW 0
4	#e7f8ff	4-Feb	Data Models				Ch 3	optional: <a href="https://github.com/w4111/hw1-s22" > HW 1 </a >		0	HW1 Part1
5	#e7f8ff	11-Feb	Data Models + ER->Relational				Ch 3	optional: <a href="https://github.com/w4111/hw1-s22" > HW 1 </a >		0	Project 1 Part 1 approval phase
6	#f2f9ed	18-Feb	Relational Algebra				Ch 4		< a href="https://github.com/w4111/project1" > Project 1 </a >	0	Project 1 Part 1 approval phase
7	#f2f9ed	25-Feb	SQL: Basics				Ch 5			0	HW1 Part 2
8	#f2f9ed	4-Mar	SQL: Advanced				Ch 5			0	HW2
9	white	11-Mar	Midterm	one 8x11 page cheat sheet both sides					< a href="https://github.com/w4111/project1/blob/main/midterm.pdf" > Midterm </a >	0	
10	white	18-Mar	HOLIDAY							0	Project 1 Part 2
11	#edf3f9	25-Mar	APIs				Ch 6			0	
12	#edf3f9	1-Apr	Data Quality	Normalization and data errors			Ch 19		< a href="https://github.com/w4111/hw4-s22" > HW 4 </a >	0	HW3
13	#ddf9ff	8-Apr	Physical Design				Ch 8		< a href="https://github.com/w4111/project2_s22" > Project 2 </a >	0	
14	#ddf9ff	15-Apr	Query Processing				Ch 12			0	Project 1 Part 3
15	#ddf9ff	22-Apr	Transactions				Ch 16, 18			0	
16	white	29-Apr	Data Pipelines							0	HW 4
17	white	13-May	Exam 2 (Cumulative)	one 8x11 page cheat sheet both sides						0	Project 2
18											
19											
20											

# What is a Database?



••••• AT&T ⌂ 3:00 PM 1 3G

Contacts +

Search

A

Apple Inc.

C

Call Recorder

F

Julia Fillory

Mike Fillory me

G

Justin Gilmore

Thomas Gilmore

Willa Good

H

Barry T. Hubbard

M

Favorites Recents Contacts Keypad Voicemail

A screenshot of a smartphone contacts application. The screen shows a list of contacts sorted by initial. At the top right is a red '+' button. Below the search bar, sections are labeled with letters (A, C, F, G, H, M). Each section contains one or more contact names. The 'Contacts' icon at the bottom is highlighted in blue, indicating it's the active screen. The status bar at the very top shows signal strength, battery level, and the time (3:00 PM).

# What is a Database?

```
2012-01-04 00:01:23,180 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Receiving block blk_-2281137920769  
010  
2012-01-04 00:01:23,184 INFO org.apache.hadoop.hdfs.server.datanode.DataNode.clienttrace: src: /127.0.0.1:32981,  
cliID: DFSClient_-603743753, offset: 0, srvID: DS-292194659-127.0.1.1-50010-1324763300176, blockid: blk_-228113  
2012-01-04 00:01:23,185 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: PacketResponder 0 for block blk_-2  
2012-01-04 00:01:23,291 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Receiving block blk_37660314352523  
10  
2012-01-04 00:01:23,293 INFO org.apache.hadoop.hdfs.server.datanode.DataNode.clienttrace: src: /127.0.0.1:32982,  
cliID: DFSClient_-603743753, offset: 0, srvID: DS-292194659-127.0.1.1-50010-1324763300176, blockid: blk_37660314  
2012-01-04 00:01:23,293 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: PacketResponder 0 for block blk_37  
2012-01-04 00:01:23,324 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Receiving block blk_-8044922265890  
010  
2012-01-04 00:01:23,326 INFO org.apache.hadoop.hdfs.server.datanode.DataNode.clienttrace: src: /127.0.0.1:32983,  
cliID: DFSClient_-603743753, offset: 0, srvID: DS-292194659-127.0.1.1-50010-1324763300176, blockid: blk_-80449222  
2012-01-04 00:01:23,327 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: PacketResponder 0 for block blk_-8  
2012-01-04 00:01:23,409 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Receiving block blk_-9657937572621  
10  
2012-01-04 00:01:23,411 INFO org.apache.hadoop.hdfs.server.datanode.DataNode.clienttrace: src: /127.0.0.1:32984,  
, cliID: DFSClient_-603743753, offset: 0, srvID: DS-292194659-127.0.1.1-50010-1324763300176, blockid: blk_-96579  
2012-01-04 00:01:23,411 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: PacketResponder 0 for block blk_-9  
2012-01-04 00:01:23,433 INFO org.apache.hadoop.hdfs.server.datanode.DataNode.clienttrace: src: /127.0.0.1:50010,  
cliID: DFSClient_-2054881890, offset: 0, srvID: DS-292194659-127.0.1.1-50010-1324763300176, blockid: blk_-96579  
2012-01-04 00:01:23,494 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Receiving block blk_54159109576590  
10  
2012-01-04 00:01:23,498 INFO org.apache.hadoop.hdfs.server.datanode.DataNode.clienttrace: src: /127.0.0.1:32987,  
, cliID: DFSClient_-2054881890, offset: 0, srvID: DS-292194659-127.0.1.1-50010-1324763300176, blockid: blk_54159  
2012-01-04 00:01:23,498 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: PacketResponder 0 for block blk_54  
2012-01-04 00:01:23,523 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Receiving block blk_-5517241460358
```

# What is a Database?



# 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 script a DBMS?

## Javascript/Python Script

Data stored in variables (RAM)

Very fast access

Data structures (lists, dicts, tuples)

# Is Excel a DBMS?

Microsoft office security

Visually access/modify/compute over data cells

Click save to store persistently

# 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.  $\neg \backslash (\exists) \neg$

# Is the file system a DBMS?

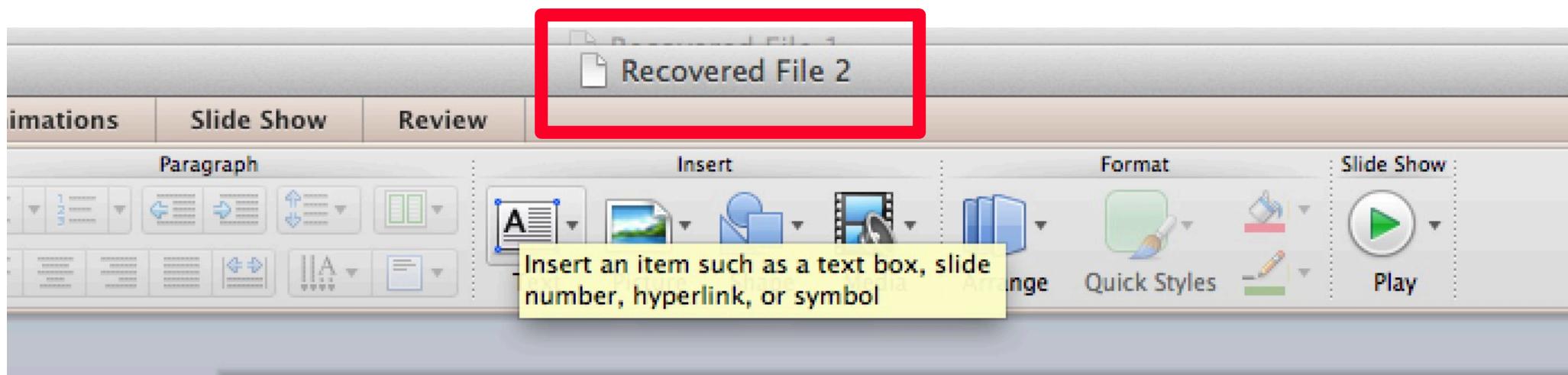
You edit a text file

Computer crashes

What happens?

1. All changes survive
2. No changes survive
3. Changes from last save survive
4.  $\neg \backslash (\exists) \backslash$

# Is the file system a DBMS?



The screenshot shows the Microsoft Word ribbon. The 'Insert' tab is highlighted with a red box. A tooltip below the 'Insert' tab says: "Insert an item such as a text box, slide number, hyperlink, or symbol". The rest of the ribbon tabs (Animations, Slide Show, Review) are visible but not highlighted.

Below the ribbon, the slide content area displays the following text:

**COMS W4111**  
**Introduction to Databases**

- . . . -

# Who... would ever do this?

Real \$IB+ Companies...

Store extracted data in a file

Every change → rewrite the file

EUGENE WU

BIO

Eugene Wu is broadly interested in technologies that help users play well at all technical levels to effectively and quickly make sense of their informatics that ultimately improve the user interface between people and data, and uses data mining and machine learning techniques to do so. Eugene Wu received a PhD from MIT, B.S. from Cal, and was a postdoc in the AMPLab. A profile, an interview, and a video of him speaking at VLDB 2018 can be found here.

Eugene Wu has received the VLDB 2018 10-year test of time award, best paper award at VLDB 2016, the SIGMOD 2016 best demo award, the NSF CAREER, and the G

The WuLab Website & Blog

We are recruiting PhDs + Postdocs, and Interns + UGrad + Mast

Overview of My Research and Teaching

SELECTED PUBLICATIONS (SHOW ALL)

Private Federated Exploration of Inference Queries  
Young Wu, Yiqia Lu, Lampros Flekas, Jianan Wang, Eugene Wu  
VLDB 2022

Explaining SQL-M: Queries with Bayesian Optimization  
Brandon Lockhart, Jianan Wang, Eugene Wu

From Debugging Before ML to Cleaning For ML  
Felix Nezura, Binger Chen, Ziaessah Abdine, Eugene Wu  
Invited, IEEE Data Engineering Bulletin 2021

Continuous Preference for Interactive Data Applications  
Haneen Mohammed, Ziyun Wei, Ravi Netravali, Eugene Wu  
VLDB 2020 Talk Video Bioghost

Complaint-driven Training Data Debugging for Query 2.0  
Young Wu, Lampros Flekas, Jianan Wang, Eugene Wu  
SIGMOD 2020 Talk Video Bioghost

Monte Carlo Tree Search for Generating Interactive Data Analysis Interfaces  
Yihe Chen, Eugene Wu

NEWS

Jun-2022: Looking forward to giving one of the keynotes at SEA DATA at VLDB 2022 this summer!

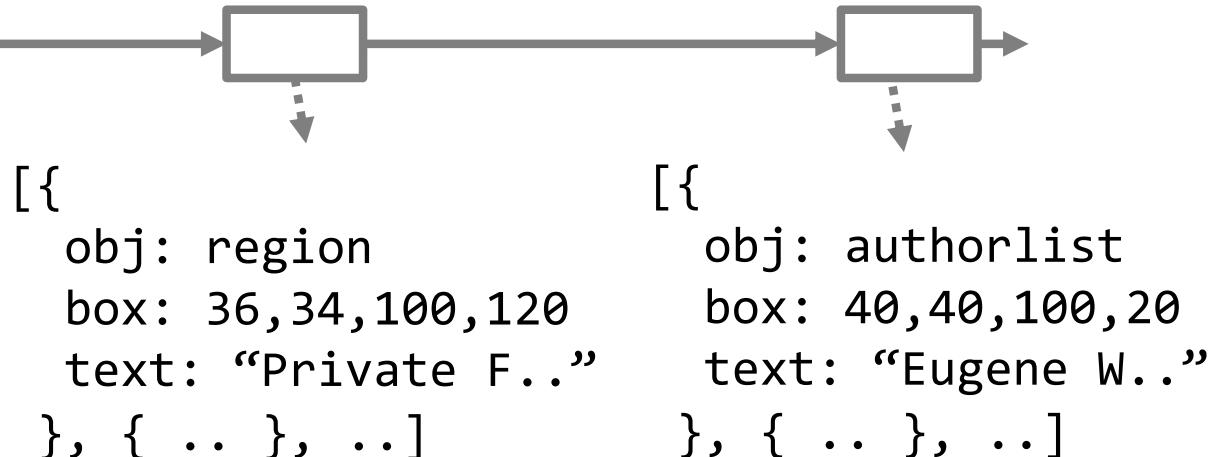
Aug-2020: For Highly Interactive Apps, Prediction is Not Enough! is a blog post to introduce our Khamelone paper. Hansen also recorded a short YouTube video about it.

Jul-2020: FLAME EBS VCT-DIST Khamelone, our rethink of client-server communication for interactive applications will be presented at VLDB 2020! With Haneen Mohammed, Tracy Wei, and Ravi Netravali. This work is based on our khamelone system and has links to previous work.

Jun-2020: Haneen participated in, and won, first place at the 2020 SIGMOD student research competition for her work on Khamelone!

Mar-2020: FATALITY! A new mortal kombat-themed system has been beaten into submission. Our full paper about it is available on Arxiv.

## Text extraction and transform tasks



# Browser

# New Tab

**EUGENE WU**



**BIO**

Eugene Wu is broadly interested in technologies that help users play with their data. His goal is for all technical levels to effectively and quickly make sense of their information. He is interested in solutions that ultimately improve the interface between users and data, and uses techniques borrowed from fields such as data management, systems, crowd sourcing, visualization, and HCI. Eugene Wu received his B.S. from MIT, and was a postdoc in the AMPLab. A profile, an obit.

Eugene Wu has received the VLDB 2018 10-year test of time award, best-of-conference citations at VLDB, the SIGMOD 2016 best demo award, the NSF CAREER, and the Google and Amazon faculty awards.

The WuLab Website & Blog  
We are recruiting PhDs + Postdocs, and Interns + UGrad + Masters!

[Overview of My Research and Teaching](#)

**NEWS**

Jun-2021: Looking forward to giving one of the keynotes at SEA DATA at VLDB 2021 this summer!

Aug-2020: For Highly Interactive Apps, Prediction is Not Enough is a blog post to introduce our Kameleon paper. Haneen also recorded a short YouTube video summarizing our work.

Jul-2020: FLAWLESS VICTORY! Kameleon, our

**SELECTED PUBLICATIONS (SHOW ALL)**

Private Federated Explanation of Inference Queries  
Young Wu, Yiqia Lu, Lampros Flekas, Jianan Wang, Eugene Wu  
VLDB 2022

Explaining SQL-ML Queries with Bayesian Optimization  
Brandon Lockhard, Jianan Wang, Eugene Wu

From Debugging Before ML to Cleaning For ML  
Felix Nezura, Binger Chen, Ziaessab Abdess, Eugene Wu  
Invited, IEEE Data Engineering Bulletin 2021

Continuous Preference for Interactive Data Applications  
Haneen Mohammad, Ziyun Wei, Rav Nettivalli, Eugene Wu  
VLDB 2020 Talk Video Biogpost

Complaint-driven Training Data Debugging for Query 2.0  
Young Wu, Lampros Flekas, Jianan Wang, Eugene Wu  
SIGMOD 2020 Talk Video Biogpost

Monte Carlo Tree Search for Generating Interactive Data Analysis Interfaces  
Yiqia Chen, Eugene Wu

**EUGENE WU**



**BIO**

Eugene Wu is broadly interested in technologies that help users play with their data. His goal is for all technical levels to effectively and quickly make sense of their information. He is interested in solutions that ultimately improve the interface between users and data, and uses techniques borrowed from fields such as data management, systems, crowd sourcing, visualization, and HCI. Eugene Wu received his B.S. from MIT, and was a postdoc in the AMPLab. A profile, an obit.

Eugene Wu has received the VLDB 2018 10-year test of time award, best-of-conference citations at VLDB, the SIGMOD 2016 best demo award, the NSF CAREER, and the Google and Amazon faculty awards.

The WuLab Website & Blog  
We are recruiting PhDs + Postdocs, and Interns + UGrad + Masters!

[Overview of My Research and Teaching](#)

**NEWS**

Jun-2021: Looking forward to giving one of the keynotes at SEA DATA at VLDB 2021 this summer!

Aug-2020: For Highly Interactive Apps, Prediction is Not Enough is a blog post to introduce our Kameleon paper. Haneen also recorded a short YouTube video summarizing our work.

Jul-2020: FLAWLESS VICTORY! Kameleon, our

**SELECTED PUBLICATIONS (SHOW ALL)**

Private Federated Explanation of Inference Queries  
Young Wu, Yiqia Lu, Lampros Flekas, Jianan Wang, Eugene Wu  
VLDB 2022

Explaining SQL-ML Queries with Bayesian Optimization  
Brandon Lockhard, Jianan Wang, Eugene Wu

From Debugging Before ML to Cleaning For ML  
Felix Nezura, Binger Chen, Ziaessab Abdess, Eugene Wu  
Invited, IEEE Data Engineering Bulletin 2021

Continuous Preference for Interactive Data Applications  
Haneen Mohammad, Ziyun Wei, Rav Nettivalli, Eugene Wu  
VLDB 2020 Talk Video Biogpost

Complaint-driven Training Data Debugging for Query 2.0  
Young Wu, Lampros Flekas, Jianan Wang, Eugene Wu  
SIGMOD 2020 Talk Video Biogpost

Monte Carlo Tree Search for Generating Interactive Data Analysis Interfaces  
Yiqia Chen, Eugene Wu

**EUGENE WU**



**BIO**

Eugene Wu is broadly interested in technologies that help users play with their data. His goal is for all technical levels to effectively and quickly make sense of their information. He is interested in solutions that ultimately improve the interface between users and data, and uses techniques borrowed from fields such as data management, systems, crowd sourcing, visualization, and HCI. Eugene Wu received his B.S. from MIT, and was a postdoc in the AMPLab. A profile, an obit.

Eugene Wu has received the VLDB 2018 10-year test of time award, best-of-conference citations at VLDB, the SIGMOD 2016 best demo award, the NSF CAREER, and the Google and Amazon faculty awards.

The WuLab Website & Blog  
We are recruiting PhDs + Postdocs, and Interns + UGrad + Masters!

[Overview of My Research and Teaching](#)

**NEWS**

Jun-2021: Looking forward to giving one of the keynotes at SEA DATA at VLDB 2021 this summer!

Aug-2020: For Highly Interactive Apps, Prediction is Not Enough is a blog post to introduce our Kameleon paper. Haneen also recorded a short YouTube video summarizing our work.

Jul-2020: FLAWLESS VICTORY! Kameleon, our

**SELECTED PUBLICATIONS (SHOW ALL)**

Private Federated Explanation of Inference Queries  
Young Wu, Yiqia Lu, Lampros Flekas, Jianan Wang, Eugene Wu  
VLDB 2022

Explaining SQL-ML Queries with Bayesian Optimization  
Brandon Lockhard, Jianan Wang, Eugene Wu

From Debugging Before ML to Cleaning For ML  
Felix Nezura, Binger Chen, Ziaessab Abdess, Eugene Wu  
Invited, IEEE Data Engineering Bulletin 2021

Continuous Preference for Interactive Data Applications  
Haneen Mohammad, Ziyun Wei, Rav Nettivalli, Eugene Wu  
VLDB 2020 Talk Video Biogpost

Complaint-driven Training Data Debugging for Query 2.0  
Young Wu, Lampros Flekas, Jianan Wang, Eugene Wu  
SIGMOD 2020 Talk Video Biogpost

Monte Carlo Tree Search for Generating Interactive Data Analysis Interfaces  
Yiqia Chen, Eugene Wu

**EUGENE WU**



**BIO**

Eugene Wu is broadly interested in technologies that help users play with their data. His goal is for all technical levels to effectively and quickly make sense of their information. He is interested in solutions that ultimately improve the interface between users and data, and uses techniques borrowed from fields such as data management, systems, crowd sourcing, visualization, and HCI. Eugene Wu received his B.S. from MIT, and was a postdoc in the AMPLab. A profile, an obit.

Eugene Wu has received the VLDB 2018 10-year test of time award, best-of-conference citations at VLDB, the SIGMOD 2016 best demo award, the NSF CAREER, and the Google and Amazon faculty awards.

The WuLab Website & Blog  
We are recruiting PhDs + Postdocs, and Interns + UGrad + Masters!

[Overview of My Research and Teaching](#)

**NEWS**

Jun-2021: Looking forward to giving one of the keynotes at SEA DATA at VLDB 2021 this summer!

Aug-2020: For Highly Interactive Apps, Prediction is Not Enough is a blog post to introduce our Kameleon paper. Haneen also recorded a short YouTube video summarizing our work.

Jul-2020: FLAWLESS VICTORY! Kameleon, our

**SELECTED PUBLICATIONS (SHOW ALL)**

Private Federated Explanation of Inference Queries  
Young Wu, Yiqia Lu, Lampros Flekas, Jianan Wang, Eugene Wu  
VLDB 2022

Explaining SQL-ML Queries with Bayesian Optimization  
Brandon Lockhard, Jianan Wang, Eugene Wu

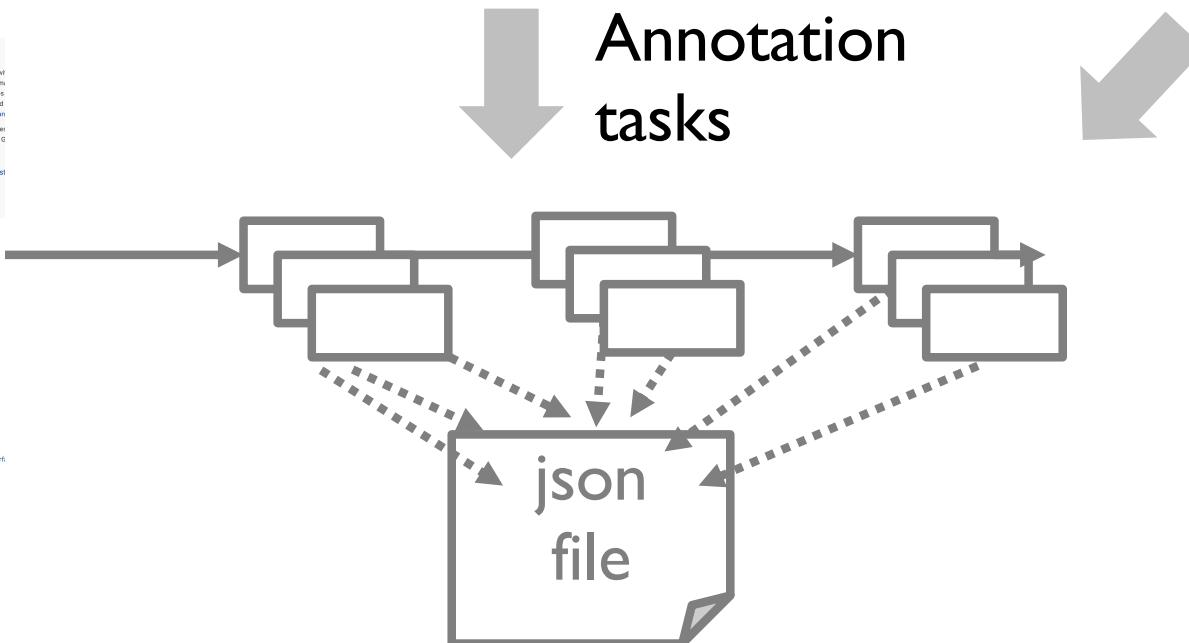
From Debugging Before ML to Cleaning For ML  
Felix Nezura, Binger Chen, Ziaessab Abdess, Eugene Wu  
Invited, IEEE Data Engineering Bulletin 2021

Continuous Preference for Interactive Data Applications  
Haneen Mohammad, Ziyun Wei, Rav Nettivalli, Eugene Wu  
VLDB 2020 Talk Video Biogpost

Complaint-driven Training Data Debugging for Query 2.0  
Young Wu, Lampros Flekas, Jianan Wang, Eugene Wu  
SIGMOD 2020 Talk Video Biogpost

Monte Carlo Tree Search for Generating Interactive Data Analysis Interfaces  
Yiqia Chen, Eugene Wu

## Annotation tasks



# 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, deity  
Lots of users concurrency, scaling, responsiveness  
Ad-hoc data access arbitrary queries  
Data formats csv? tsv? custom format?

# 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.

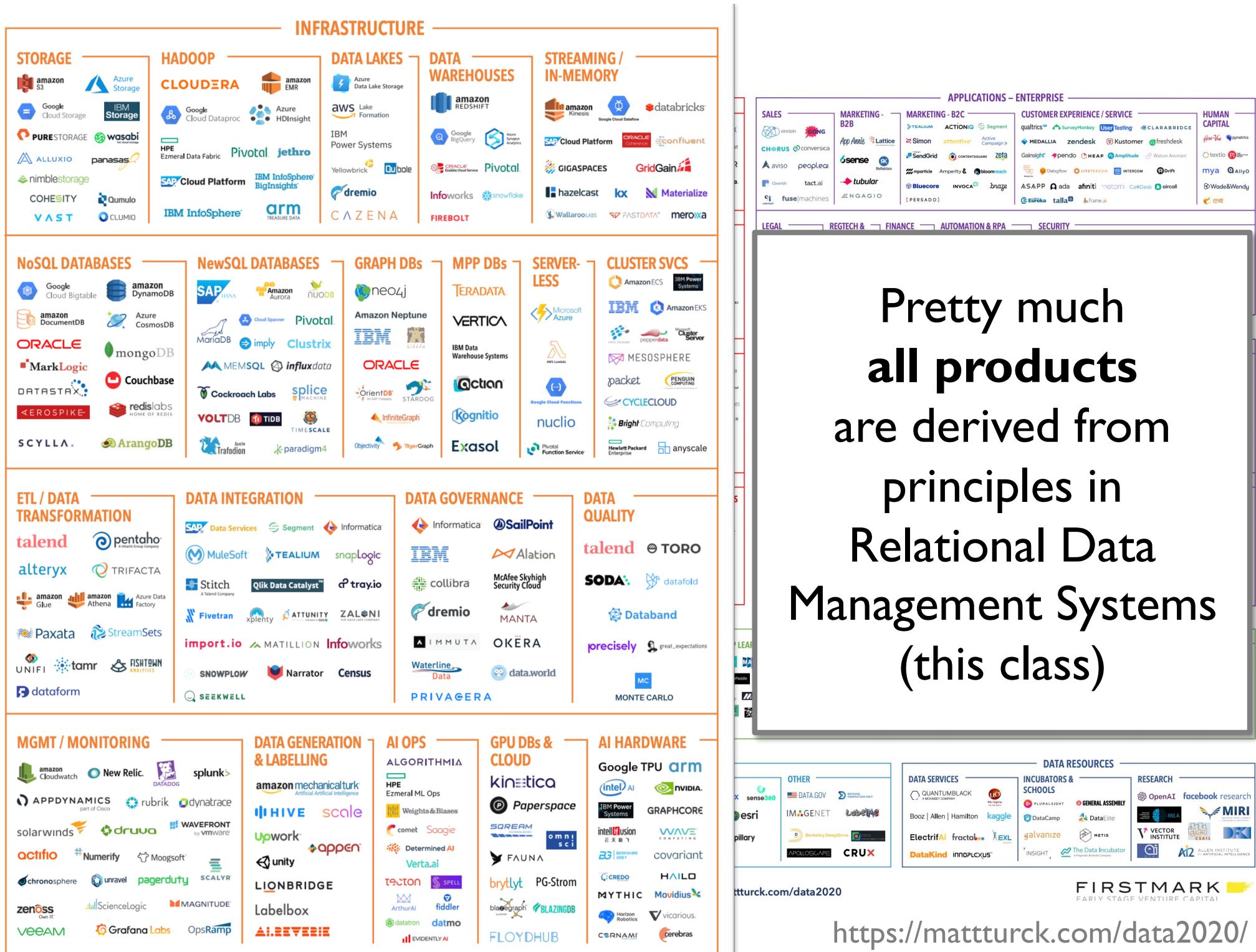
# Database Management System (DBMS)

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

# 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, KDD
Logic	Relational Algebra = 1 <sup>st</sup> order logic

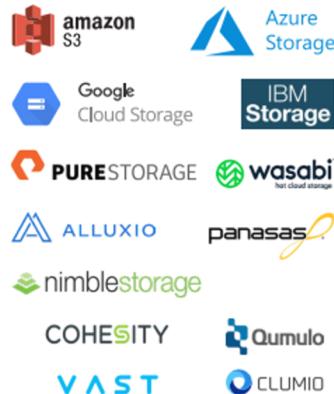
## Scalable Computer Science



# Golden Era of Data Systems!

## INFRASTRUCTURE

### STORAGE



### HADOOP



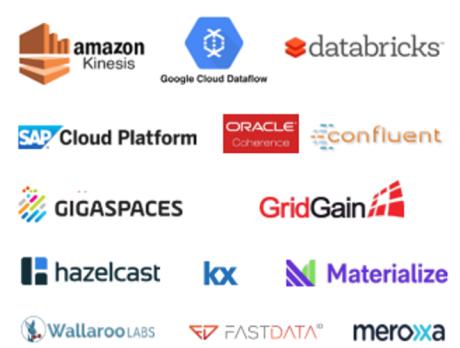
### DATA LAKES



### DATA WAREHOUSES



### STREAMING / IN-MEMORY



### NoSQL DATABASES



### NewSQL DATABASES



### GRAPH DBs



### MPP DBs



### SERVER-LESS



### CLUSTER SVCS



# 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

External Schema

**Conceptual  
Schema**

Describes logical  
structure

Conceptual Schema

**Physical Schema**

Describes files,  
formats, indexes

Physical Schema

“Data”

# 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	17
1	Luis	20
2	Ken	30

0,Eugene,17  
1,Luis,20  
2,Ken,30  
CSV File

What is the number of adults?

# Data Independence

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

0,Eugene,17  
1,Luis,20  
2,Ken,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	17
1	Luis	20
2	Ken	30

0 Eugene 17  
1 Luis 20  
2 Ken 30  
**TSV File**

~~n = 0  
for line in csv\_file:  
 attributes = line.split(",")  
 if attributes[2] >= 18:  
 n += 1~~

# Data Independence

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

0,1,2  
Eugene,Luis,Ken  
17,20,30  
**Columnar File**

~~n = 0~~  
For line in csv\_file:  
    attributes = line.split(",")  
    if attributes[2] >= 18:  
        n += 1

# Data Independence

**Conceptual Schema**

Describes logical structure

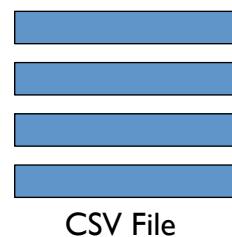
**Physical Schema**

Describes files and indexes

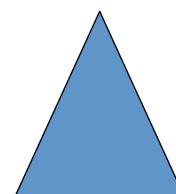
Conceptual Schema is the API!

Users(uid int, name str, age int)

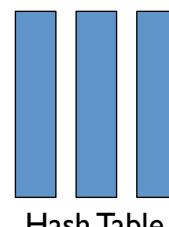
Physical Independence



CSV File



Tree Index



Hash Table

“Data”

# 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**

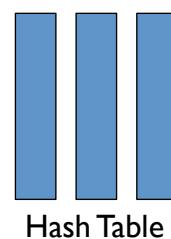
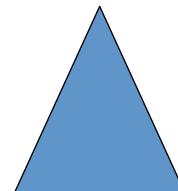
Describes logical structure

**Physical Schema**

Describes files and indexes

`Users(uid int, name str, age int)`

**Physical Independence**



“Data”

# Data Independence

**Conceptual Schema**

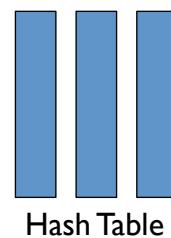
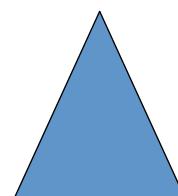
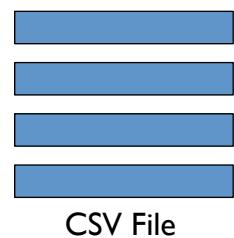
Describes logical structure

**Physical Schema**

Describes files and indexes

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

**Physical Independence**



**“Data”**

# Data Independence

## External Schema

Describe how users see data

## Conceptual Schema

Describes logical structure

## Physical Schema

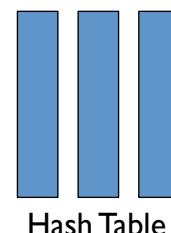
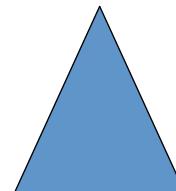
Describes files and indexes

Users(uid int, **name str**, age int)

Logical Independence

Users(uid int, **fname str**, **Iname str**, age int)

Physical Independence



“Data”

# 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 Interface

Mechanism that enables data independence  
Insulates programmer from physical schema

Rather than a list of functions,  
the API is a *query language*

# Declarative Interface

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

“Make me a sandwich”

Buy from pb&j store

Make BLT

½ Tuna

Veggie

“Take two slices of wheat bread out of the 2<sup>nd</sup> shelf, put them next to each other...”

What if on 1<sup>st</sup> shelf?  
Out of wheat bread?  
No counter space?

# Declarative Interface

“I want all highly rated fast drivers”

# Declarative Interface

`SELECT name FROM users WHERE rating > 8`

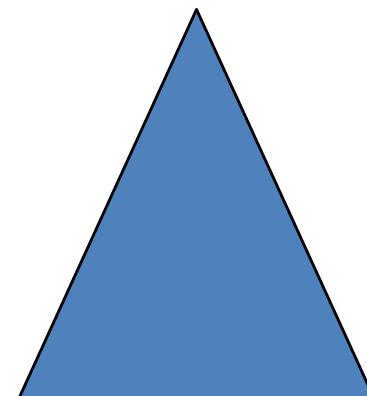
---

DBMS

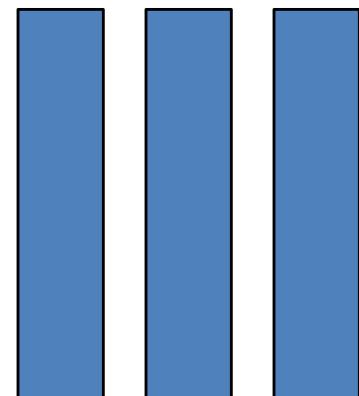
---



CSV File



Tree Index



Hash Table

# Declarative Interface

SELECT name FROM users WHERE rating > 8

---

DBMS

---

Node

Node

Node

# Declarative Interface

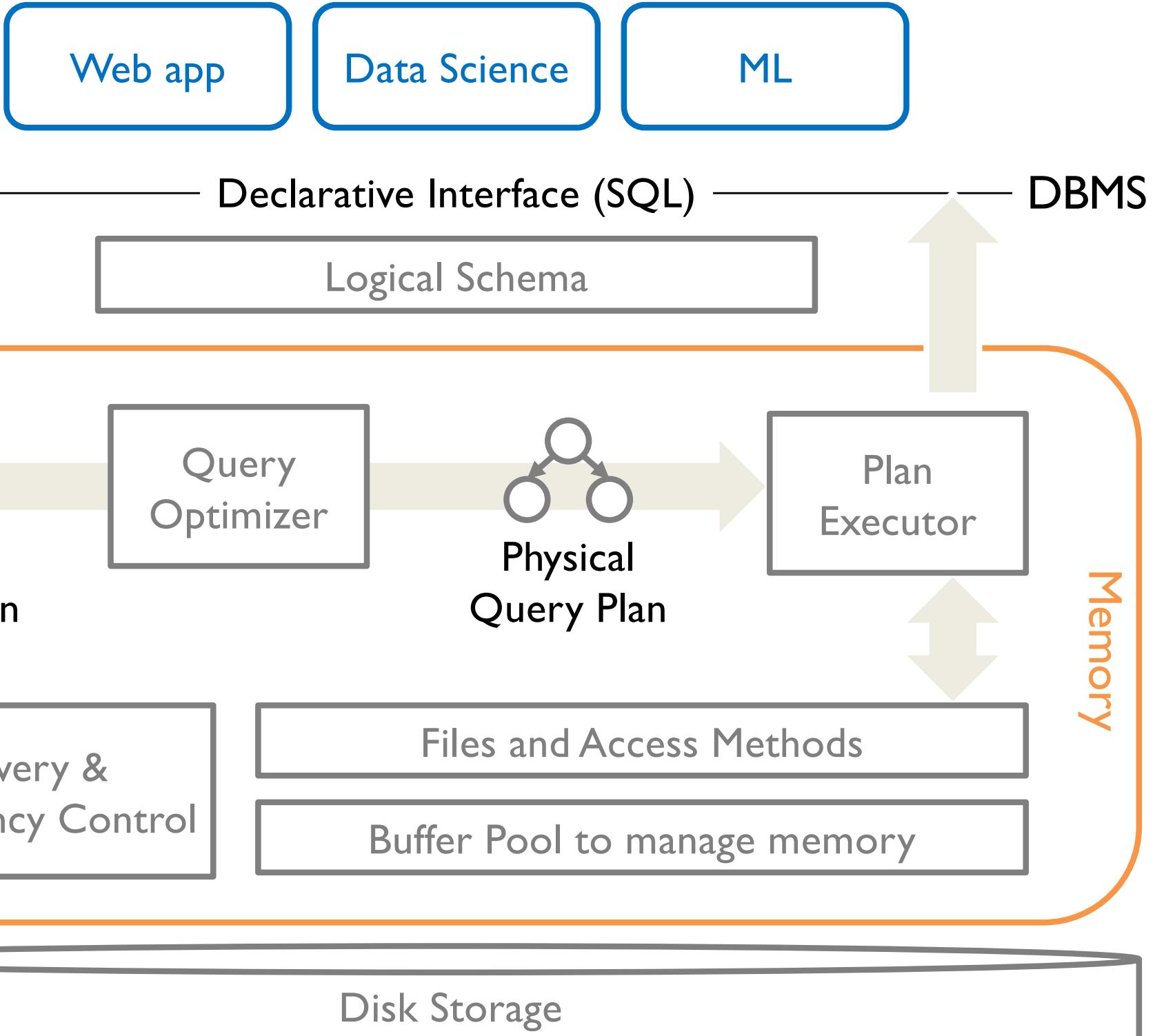
`SELECT name FROM users WHERE rating > 8`

---

DBMS

---

Node



Web app  
L13

Data Science  
L13

ML  
L13

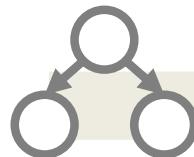
Declarative Interface (SQL L8-10)

DBMS

Logical Schema L1-5, 15-16

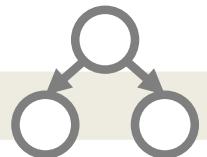
L13

L13



Logical  
Query Plan L6,7

Query  
Optimizer  
L19-20



Physical  
Query Plan

Plan  
Executor  
L19-20

Memory

Recovery &  
Concurrency Control  
L21-23

Files and Access Methods L18  
Buffer Pool to manage memory L17

Disk Storage L17

Web app  
L13

Data Science  
L13

ML  
L13

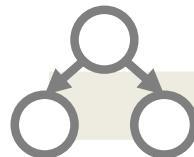
Declarative Interface (SQL L8-10)

DBMS

Logical Schema L1-5, 15-16

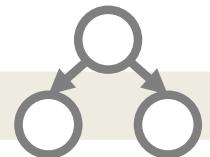
L13

L13



Logical  
Query Plan L6,7

Query  
Optimizer  
L19-20



Physical  
Query Plan

Plan  
Executor  
L19-20

Memory

Recovery &  
Concurrency Control  
L21-23

Files and Access Methods L18

Buffer Pool to manage memory L17

Disk Storage L17

# Concurrency Control

Want to let many users use database concurrently.  
How to ensure they run correctly?

# Concurrency Control

Want to let many users **use** database concurrently.  
How to ensure they run correctly?

What does "**use**" mean?

Run transaction, which groups all of the user's DBMS actions together. They either all run, or none run.

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

# Concurrency Control

Want to let many users use database concurrently.

How to ensure they run **correctly**?

What does "**Correctly**" mean?

Data in DBMS is correct if

1. Maintains all integrity constraints (types, references, checks)
2. Transactions run as if only one DBMS user at a time.

# Recovery

If the DBMS crashes, can recover to a **Correct** state.

What does “**Correct**” mean?

All committed transactions are preserved in the database.

Undo all incomplete (uncommitted) transactions.

How?

DBMS keeps a **log** of all actions each transaction performs.

Track whether an action is committed or uncommitted.

# A bit about the class

# Next Up

HW0 is out.

<https://github.com/w4111/hw0>

Due by 9/11 11:59PM.

No late submissions accepted

# 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**

# Class Information: Lectures

Tu/Th

10-11:30AM

451 CSB

# Your TAs

Office hours will be updated on course website  
later today.

Zoom links for office hours will be shared on the  
discussion board

# w4111.github.io

C O L U M B I A   U N I V E R S I T Y   C O M S   W 4 1 1 1

## INTRODUCTION TO DATABASES

### Information

- Tues/Thurs 10-11:30  
451 CSB  
3 units
- [Syllabus](#)
- [Ed Discussion](#)
- [Provide Feedback](#)
- [Course Github](#)

### Staff

- [Eugene Wu](#) Instructor  
Thurs 12-1PM
- [Zachary Huang](#)  
tba
- [Andrew Zheng](#)  
tba
- [Jennifer Wang](#)  
tba
- tba
- tba

### Office Hours

- [OH Links](#)
- [OH Calendar](#)

### Prereqs

- Required: Students are expected to be comfortable with data

### Overview

The goal of this class is two-fold. First, to introduce you to core database concepts (e.g., data modeling, logical design, SQL) so that you too can build a billion dollar application. Second, to teach enough about database engine internals (e.g., physical database design, query optimization, transaction processing) so you have a good sense of why queries may be running slowly/incorrectly. We will also discuss their relevance to systems used in industry.

The Data Management Seminar invites interesting database researchers and practitioners to speak. Students are invited to join in person or on zoom (if available). We will announce these periodically throughout the semester.

### Announcements

- [Sign up for Project 1 Part 1 staff meetings!](#) One meeting per team.
- Updated lecture 2 slides to clarify constraints over N-way relationships.
- [HW0](#) released. No Late Days! Failure to submit on time is a -5% penalty on your final grade.

### Schedule

Date	Topic	Assigned	Due
6-Sep	<a href="#">Intro and Overview</a>	<a href="#">HW 0</a> Look for teammates	
8-Sep	<a href="#">ER Models</a> optional: Textbook Chapter 6 except for Sections 6.7, 6.10, and 6.11.	HW1 Part 1 Project 1 Part 1	HW0 (9/11 11:59PM EST. NO LATE DAYS)
13-Sep	<a href="#">ER Models</a> optional: Textbook Chapter 6 except for Sections 6.7, 6.10, and 6.11.		HW 1 Part 1 (9/16 11:59PM EST) Formed Project 1 Team (no submission)
15-Sep	<a href="#">Relational Model</a> optional: Textbook Ch 2.1-2.3, 2.5, 6.7, 6.8, except 6.7.2		Project 1 Part 1 approval phase
20-Sep	<a href="#">Relational Model</a> optional: Textbook Ch 2.1-2.3, 2.5, 6.7, 6.8, except 6.7.2	HW1 Part 2 Schema	Project 1 Part 1 approval phase
22-Sep	<a href="#">Relational Algebra</a>		Project 1 Part 1 approval phase

# Discussion Board

ed COMS W4111 002 - Ed Discussion

New Thread

Search

Filter

Welcome!

Eugene Wu STAFF 6 min. ago in General

UNPIN STAR WATCHING VIEWS

Welcome! #1

Hi everyone,

We're using Ed Discussion for class Q&A.

<https://edstem.org/us/courses/28081/discussion/>

Prioritize using Ed Discussion for class and administrative questions. You will get faster answers here from staff and peers than through email.

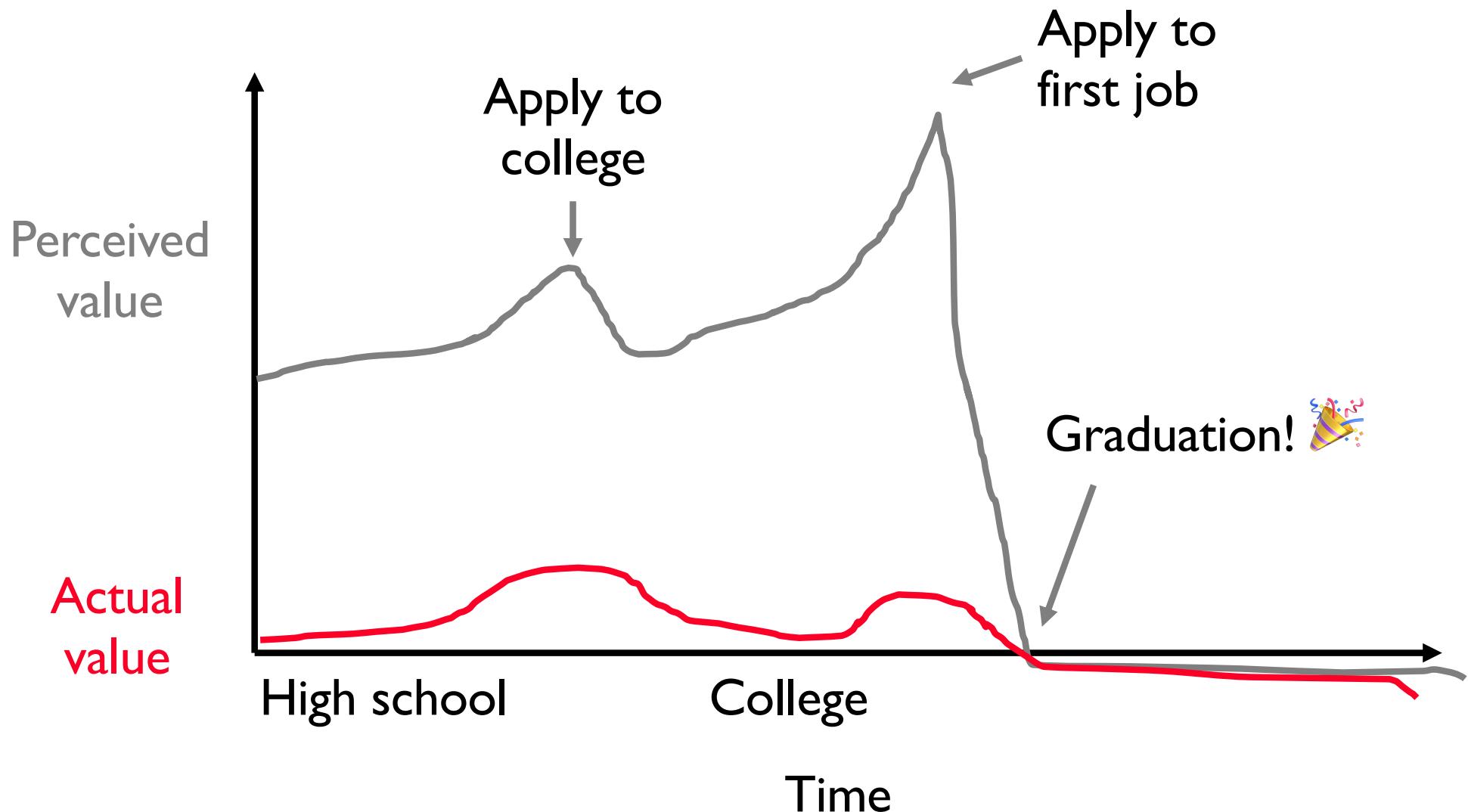
Here are some tips:

- Search before you post
- Heart questions and answers you find useful
- Answer questions you feel confident answering
- Share interesting course related content with staff and peers

For more information on Ed Discussion, you can refer to the [Quick Start Guide](#).

All the best this semester!

# Grades. How do they work?



# Grading Information

Midterm I      25%

Midterm 2      40%

HW                15% (4 HWs equally weighed)

Project I      15%

Project 2      5%

Extra credit    variable

Median grade: B or slightly higher.

# Exam Dates

Midterm I    10/13

on gradescope

Midterm 2    12/8

on gradescope, cumulative

Makeup exams are not scheduled

# Homework

Assignment will specify submission instructions.

No extensions or exceptions.

5 grace days for hws throughout the semester.

Can be applied to any assignment *unless otherwise specified*

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

Don't need to tell us, staff will assign grace days in your favor

Check full details on web site under syllabus.

# Projects (more details soon)

Two projects.

Teams of two

Run on cloud infrastructure

Python & SQL

## Project 1

Model and build your own database web application

Explore “traditional” relational database features.

Non-programming option

## Project 2

Do cool things with DBMSes

# Sports Community Mobile App

The image displays two screenshots of a mobile application interface for a sports community group named "UNYSport".

**Screenshot 1 (Top Left):** Shows the group profile page. At the top, there are two status bars: one for AT&T signal and battery at 5:30 PM, and another for Camera signal and battery at 7:21 PM. The group name "UNYSport" is displayed with a blue circular icon containing a person symbol. Below the name, there is a small thumbnail image of a group of people. The group details listed are:

- Name: Columbia Bouldering
- Sport: Bouldering
- Capacity: 100

**Screenshot 2 (Bottom Right):** Shows a messaging screen between a user and a group member named TG. The timestamp is APR 15, 2019. The messages are:

- TG: Training with Coach P tonight!  
Dont be late!  
11:21 PM
- User: Hi everyone, do you want to having a climbing practice this Thursday?  
11:21 PM
- User: Sounds, great! Lets do it!  
11:21 PM
- TG: Can someone please share their chalk with me today!  
11:21 PM

**W4111** Introduction to databases**Department:** Computer science**Description:**

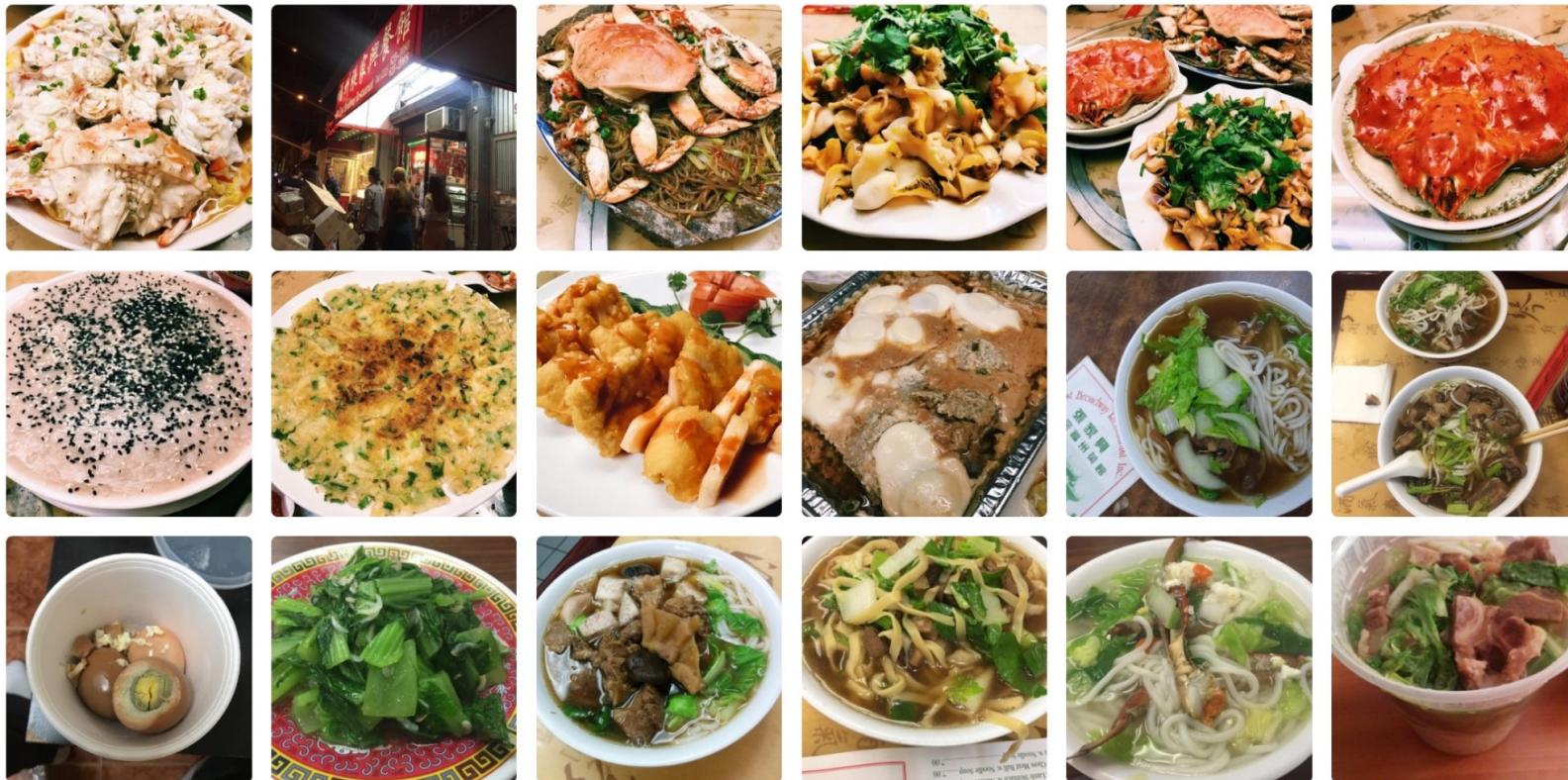
Prerequisites: (COMS W3134) or (COMS W3137) or (COMS W3136) and fluency in Java); or the instructor's permission. The fundamentals of database design and application development using databases: entity-relationship modeling, logical design of relational databases, relational data definition and manipulation languages, SQL, XML, query processing, physical database tuning, transaction processing, security. Programming projects are required.

[Sections](#)[Reviews](#)

Instructor	Time	Day	Location	Year
Alexandros Biliris	13:10-15:40	Fri	To be announced	2019
Donald F. Ferguson	10:10-12:40	Fri	To be announced	2018
Eugene Wu	16:10-17:25	Tue, Thu	501 Northwest Corner	2018
Alexandros Biliris	16:10-18:40	Mon	750 Schapiro	2017
Eugene Wu	16:10-18:40	Mon	833 Seeley W. Mudd	2016
Alexandros Biliris	16:10-18:40	Mon	752 Schapiro	2016
Luis Gravano	16:10-18:40	Mon	753 Schapiro	2016

# C-Food: Your guide to clean NYC Restaurants

## East Broadway Restaurant



Borough: manhattan

Address: 94 East Broadway

Health Investigation Score: 41/50

Average User Rating: 3.20

[Domino's](#)



# Projects (cont.)

3 grace days total for project parts 1 and 2.

No extensions or exceptions for project part 3 submission.

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

Check full details on web site.

# Extra Credit

From Midterms, Projects, standalone, ...

*all added after the curve*

Does NOT affect those that don't do extra credit

# 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.

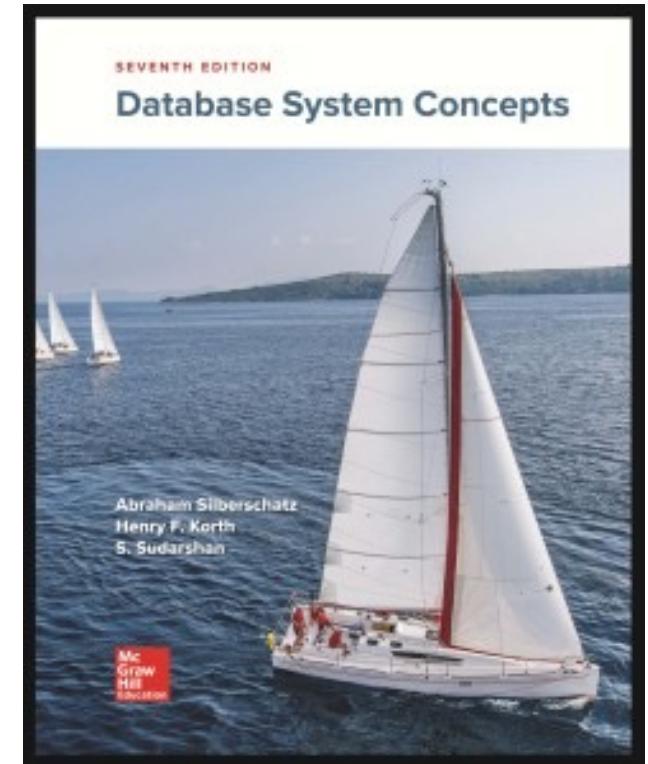
Contact the Professor Wu  
*right away if you have any questions or are falling behind.*

# Optional Textbook

Silberschatz et al.

Database System Concepts

7<sup>th</sup> ed



# On-going Feedback

COLLEGE OF COMPUTER SCIENCE UNIVERSITY OF NEW BRUNSWICK W4111

## INTRODUCTION TO DATABASES

### Information

- Tues/Thurs 10-11:30
- 451 CSB
- 3 units
- Syllabus
- Ed Discussion
- Provide Feedback
- Course Github

### Overview

The goal of this class is two-fold. First, to introduce you to core database concepts (e.g., data modeling, indexing, query optimization) that you will need to build a billion dollar application. Second, to teach enough about database engine internals (e.g., physical transaction processing) so you have a good sense of why queries may be running slowly/incorrectly. We will also cover various database systems used in industry.

### Announcements

- HW0 released. No Late Days! Failure to submit on time is a -5% penalty on your final grade.

### Staff

- Eugene Wu Instructor  
Thurs 12-1PM
- Zachary Huang  
tba
- Lia Chen  
tba
- Jessica Shi  
tba
- Jennifer Wang  
tba

### Schedule

	Date	Topic	Assigned
L1	6-Sep	Intro and Overview	HW 0 Look for teammates
L2	8-Sep	ER Models	HW1 Part 1

# Database Courses at Columbia

# **COMS W4111 - Intro to Databases**

Prerequisites: CS3137 or CS3134; fluency in Python

Intro to DBMSes

Data Models

Relational Algebra

SQL

Applications + SQL

Normalization

Peek at DBMS internals:

- Storage and indexing

- Query optimization

- Transaction Processing

# **COMS W4112-Database Sys. Impl.**

Prerequisites: CS3137 or CS3134; fluency in Python

## Components of a Database System in Detail

Storage Methods and Indexing

Query Processing and Optimization

Materialized Views

Transaction Processing and Recovery

Parallel & Distributed DBMSes

Performance Considerations Beyond Disk I/Os

# **COMS E6111-Advanced Databases**

Prerequisites: CS4111; fluency in Java or Python

Information Retrieval

Information Extraction

Web Search

Data Mining

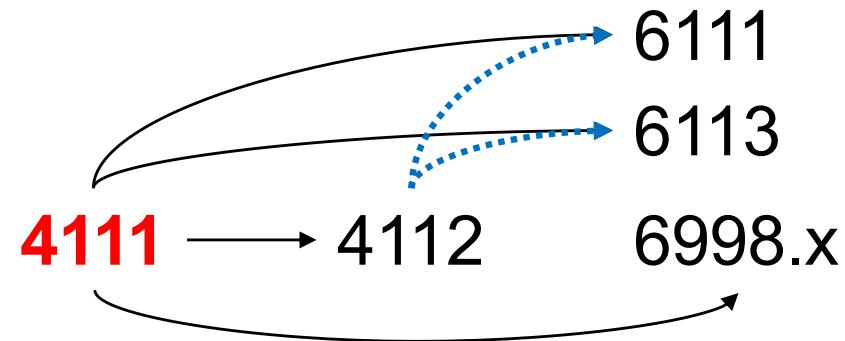
Data Warehousing, OLAP, Decision Support

# **COMS E6xxx-DB Research Seminars**

Prerequisites: CS4111; fluency in Java or Python

**6113 Database Research Topics**  
`w6113.github.io`

**6998.002 Systems for  
Human Data Interaction**  
`columbiaviz.github.io`



# Data Management at Columbia



Luis Gravano



Kenneth Ross



Eugene Wu



Mihalis Yannakakis

<http://cudbg.github.io/>

Borrowed material from  
Prof. Gravano  
Prof. Hellerstein (Cal)  
Prof. Madden & Stonebraker (MIT)

w4111.github.io

**DO HOMEWORK 0!**