

# CC5212-1

PROCESAMIENTO MASIVO DE DATOS

OTOÑO 2019

## Lecture 12

### Conclusion

Aidan Hogan

[aidhog@gmail.com](mailto:aidhog@gmail.com)

WHAT WE'VE LEARNED

# Distributed Systems

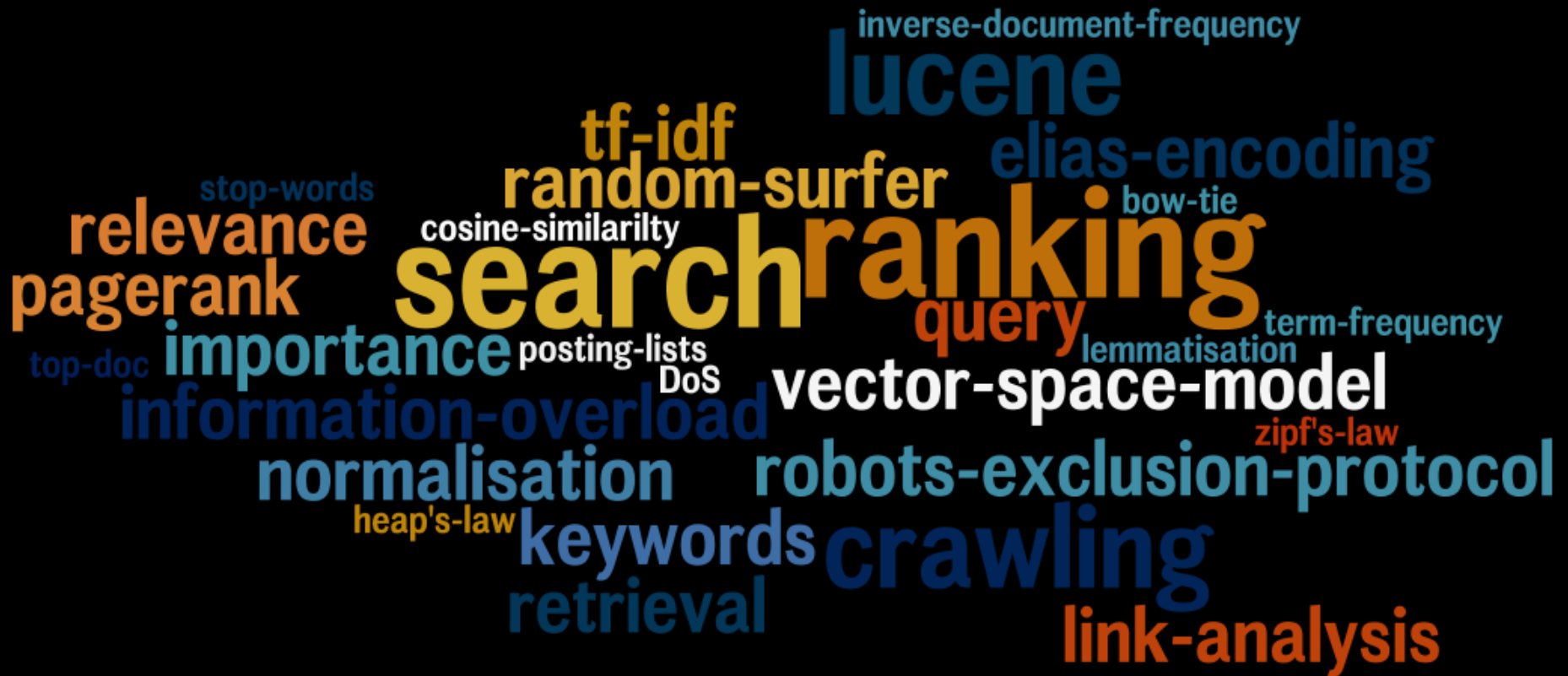
A word cloud of distributed systems concepts. The words are arranged in a roughly rectangular shape, with 'distributed systems' being the largest and most central. Other prominent words include 'availability', 'client server', 'peer to peer', 'three tier architecture', 'consistency', 'replication', 'two phase commit', 'fault tolerance', 'distributed hash table', 'partitions', 'synchronous', 'paxos', 'java rmi', 'fallacies', 'three phase commit', 'transparency', 'external sorts', 'consensus protocols', 'cap theorem', 'byzantine failure', 'cloud', 'grid', 'scalability', and 'cluster'. The words are in various colors including blue, green, yellow, orange, red, and purple.

external sorts replication consistency  
consensus protocols cap theorem  
availability two phase commit  
fault tolerance  
distributed hash table partitions  
client server synchronous  
paxos java rmi  
distributed systems  
peer to peer asynchronous fallacies  
three phase commit  
transparency three tier architecture

# Hadoop/MapReduce/Pig/Spark: Processing Un/Structured Information



# Information Retrieval: Storing Unstructured Information



NoSQL:

# Storing (Semi-)Structured Information



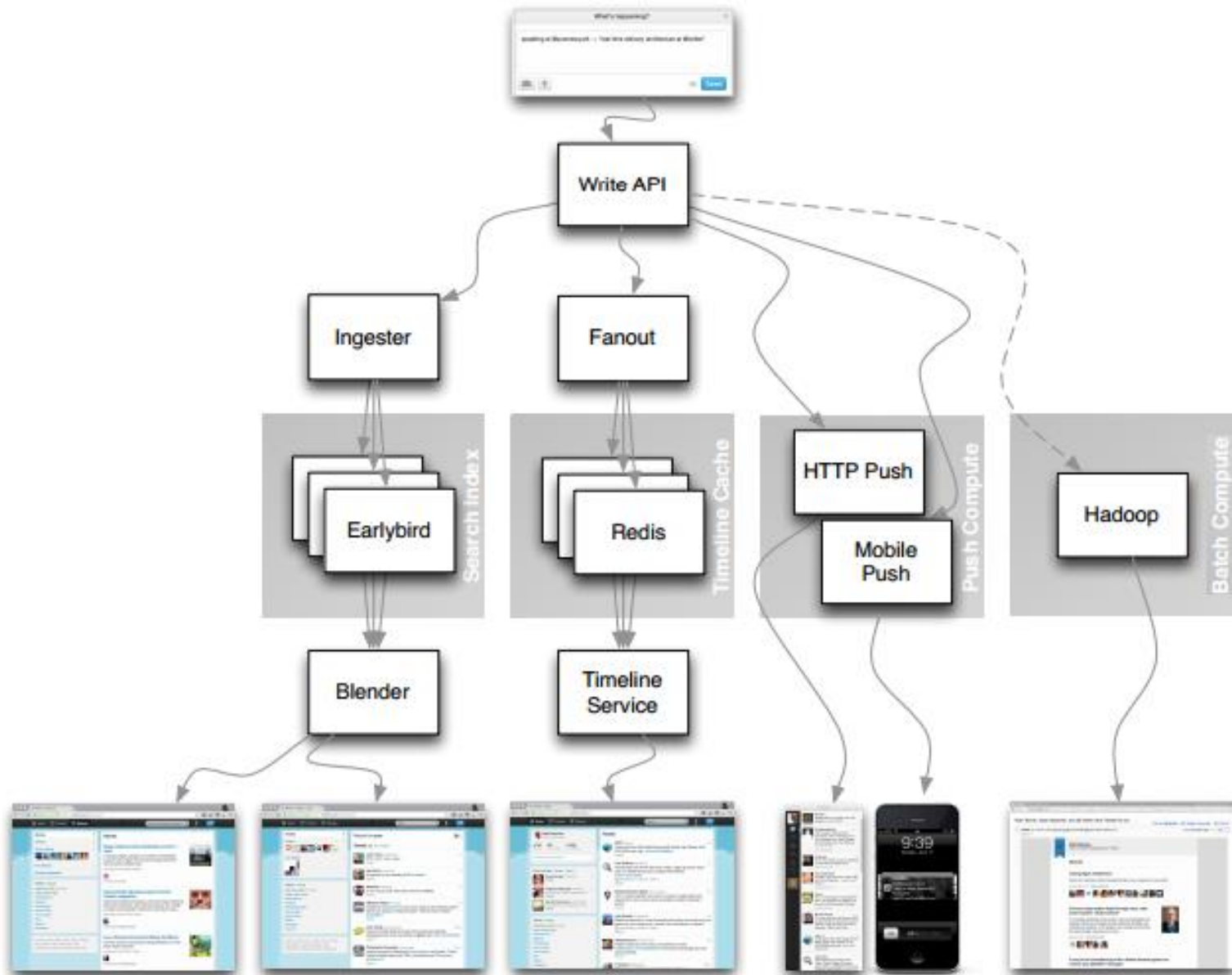
FULL-CIRCLE

# The value of data ...

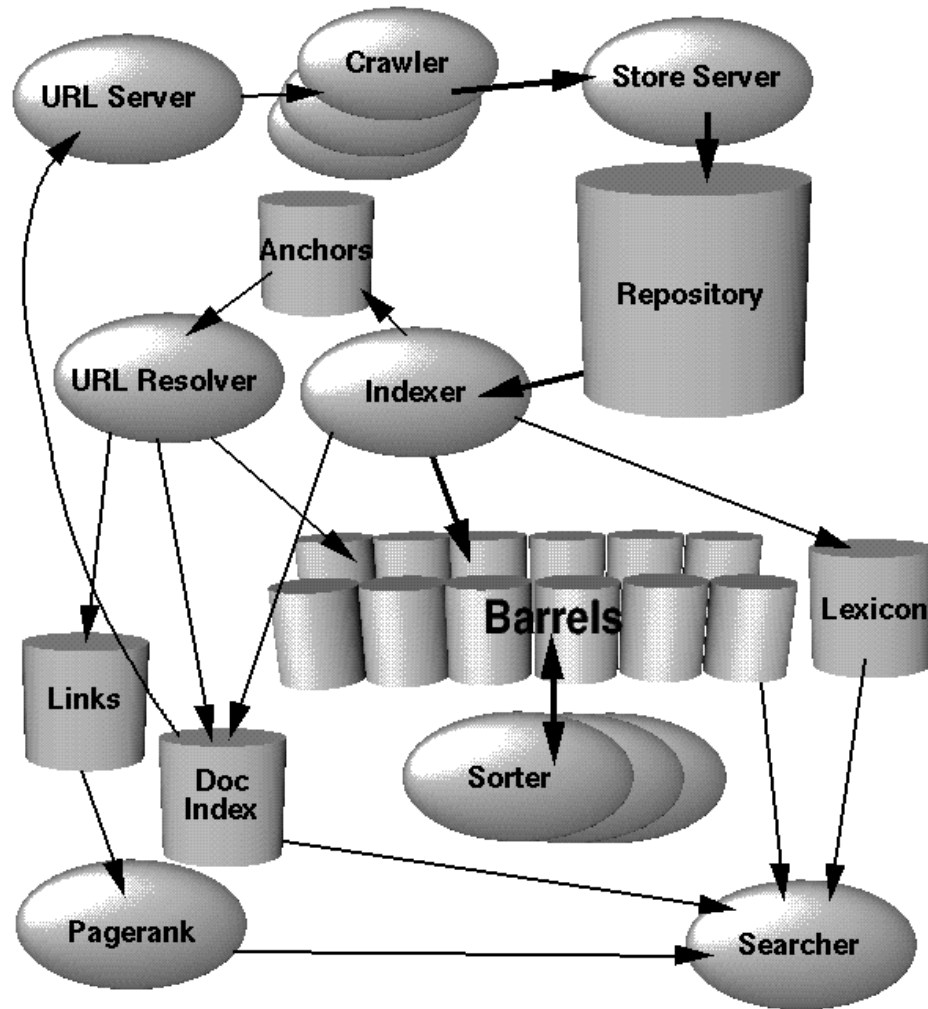




# Twitter architecture



# Google architecture



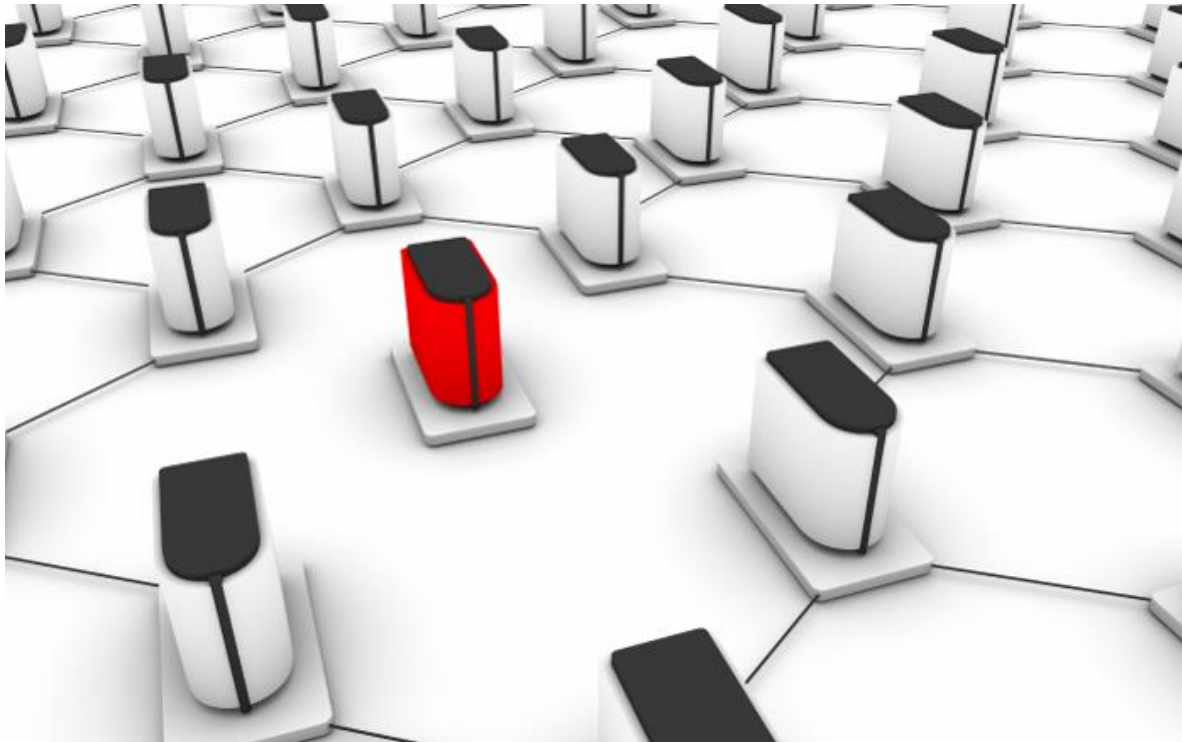
Generalise concepts to ...



# Working with large datasets

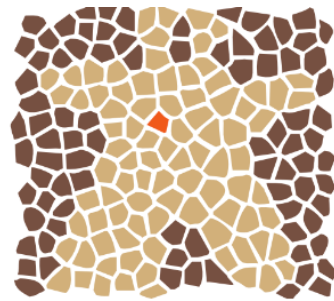


# Value/danger of distribution



# Frameworks

- For Distrib. Processing



A P A C H E  
G I R A P H

- For Distrib. Storage



cassandra



# The Big Data Buzz-word



# "Data Science"

Harvard  
Business  
Review



The shortage of data scientists is becoming a serious constraint in some sectors.

DATA

## Data Scientist: The Sexiest Job of the 21st Century

by Thomas H. Davenport and D.J. Patil

FROM THE OCTOBER 2012 ISSUE



# “Data Scientist” Job Postings (2016)

Here are the top 10 in-demand skills for data scientists:

Skills	Job skill appears in	% of jobs with skill
SQL	1987	56%
Hadoop	1713	49%
Python	1367	39%
Java	1287	36%
R	1120	32%
Hive	1099	31%
Mapreduce	768	22%
NoSQL	657	18%
Pig	561	16%
SAS	560	16%

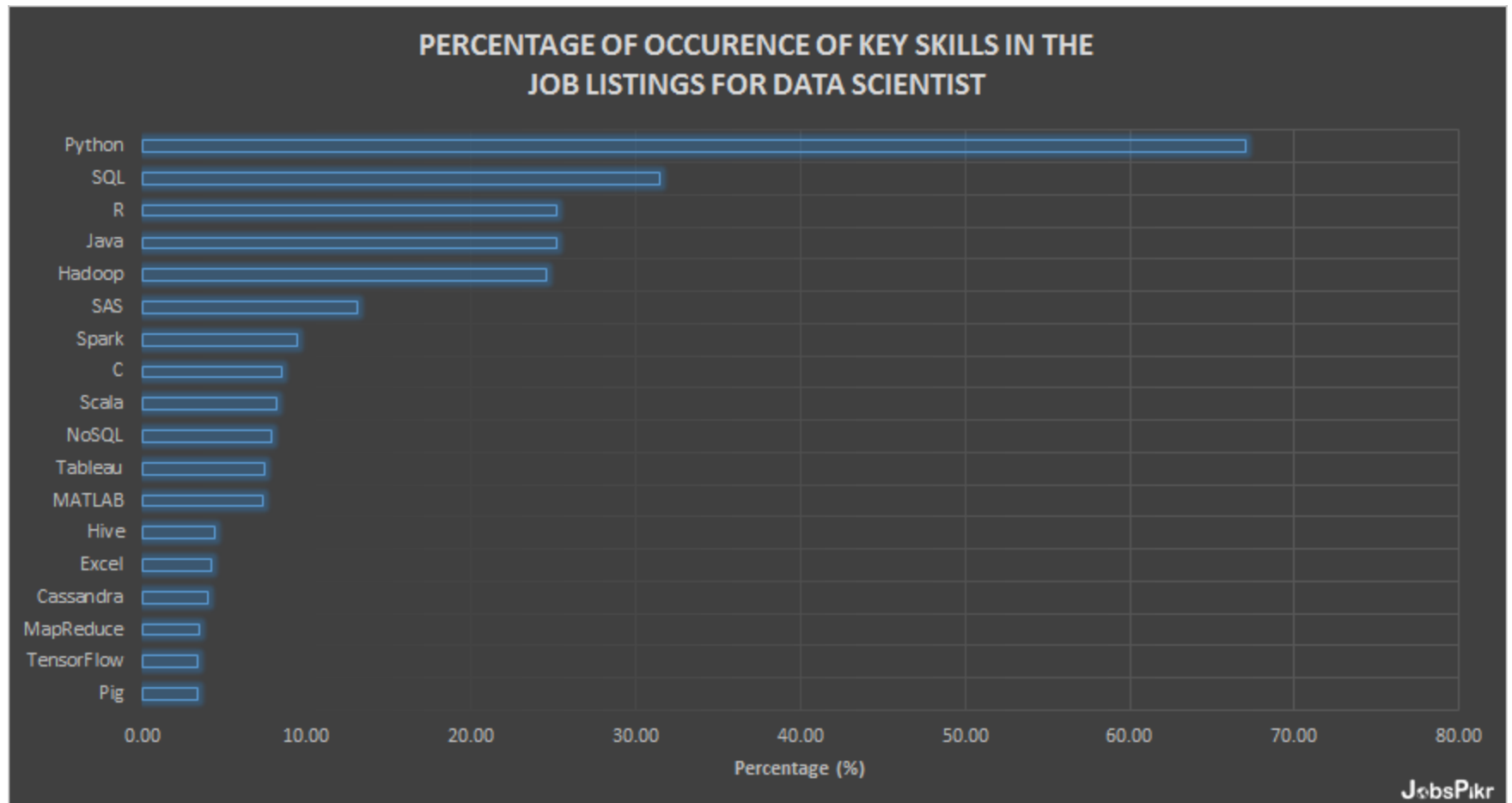
# “Data Scientist” Job Postings (2017)

## Becoming A Data Scientist: The Skills That Can Make You The Most Money

To pinpoint the most common skills, Glassdoor took 10,000 data scientist job listings that appeared on its job search platform between January and July of this year. The skills required were noted, as were the salaries offered. The data coding skills were extrapolated and analysts searched for those that came up the most within listings. The ten skills that appeared most often as prerequisites for the job, and the percentage of job listings in which they appeared, were:

1. **Python** (72%)
2. **R** (64%)
3. **SQL** (51%)
4. **Hadoop** (39%)
5. **Java** (33%)
6. **SAS** (30%)
7. **Spark** (27%)
8. **Matlab** (20%)
9. **Hive** (17%)
10. **Tableau** (14%)

# “Data Scientist” Job Postings (2018)





IMPORTANT GOAL ...

[Jobs](#) [Companies](#) [Degrees](#)

[United States](#) / [Job](#) / Big Data Consultant

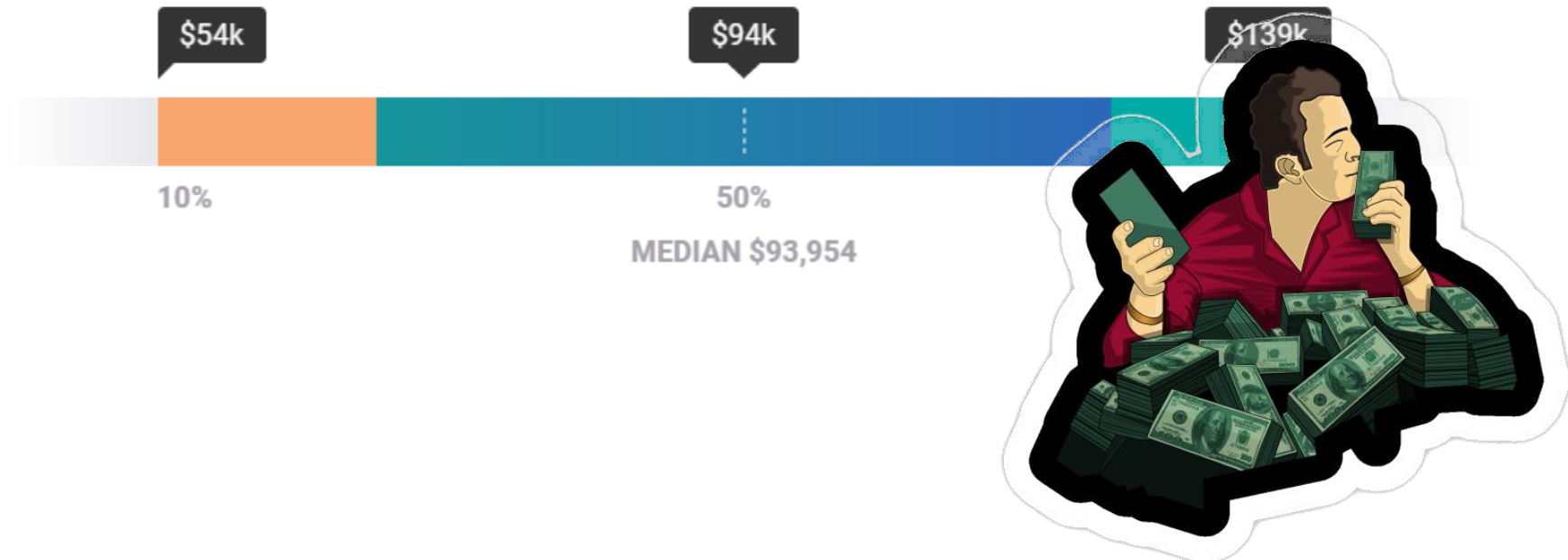
# Average Big Data Consultant Salary

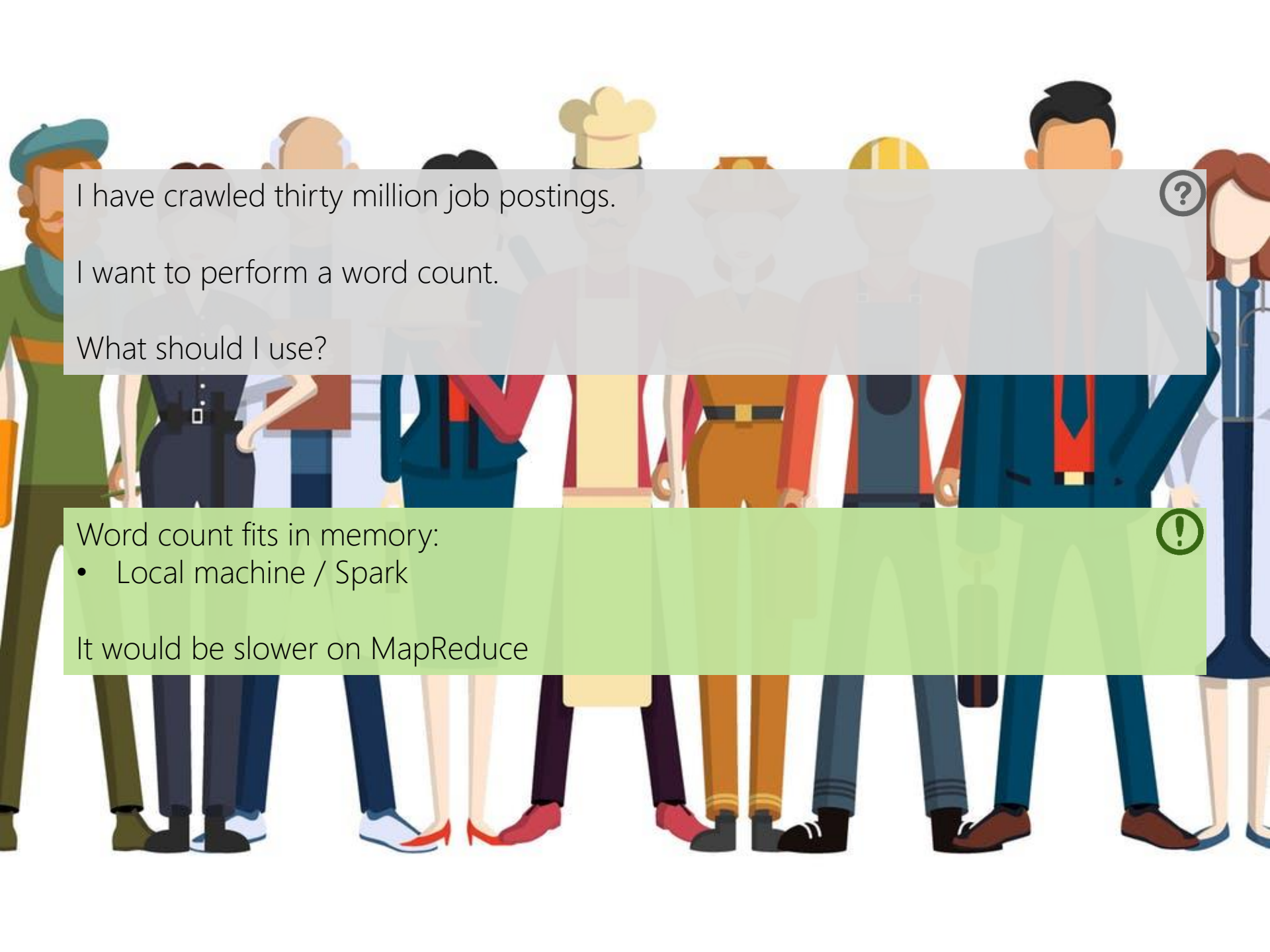
## \$93,954

Avg. Salary

**\$9,826**  
BONUS

The average salary for a Big Data Consultant is \$93,954.





I have crawled thirty million job postings.


I want to perform a word count.

What should I use?

Word count fits in memory:

- Local machine / Spark

It would be slower on MapReduce

The background of the slide features a light-colored, textured wall. At the top, there is a horizontal line of icons: a red speech bubble, a green globe, a yellow shopping cart, and a red telephone handset. Below this, a network of faint, semi-transparent icons is visible, including a question mark, a thumbs up, a camera, a Wi-Fi signal, and a mail envelope. In the lower half of the image, the lower legs and feet of four people are visible, standing against the wall and looking at their smartphones. The person on the far left is wearing dark pants and shoes. The second person from the left is wearing light blue jeans and black boots. The third person is wearing blue jeans with a tear and tan shoes. The person on the far right is wearing dark pants and shoes.

My website has 120 million users.

Each user has personal profile data, photos, friends and games.

I have 12 machines available.

What should I use?

NoSQL (something like Cassandra, MongoDB)

Replication important! (e.g., replication factor 3, 40 million users on each machine)



My company has 30,000 employees.

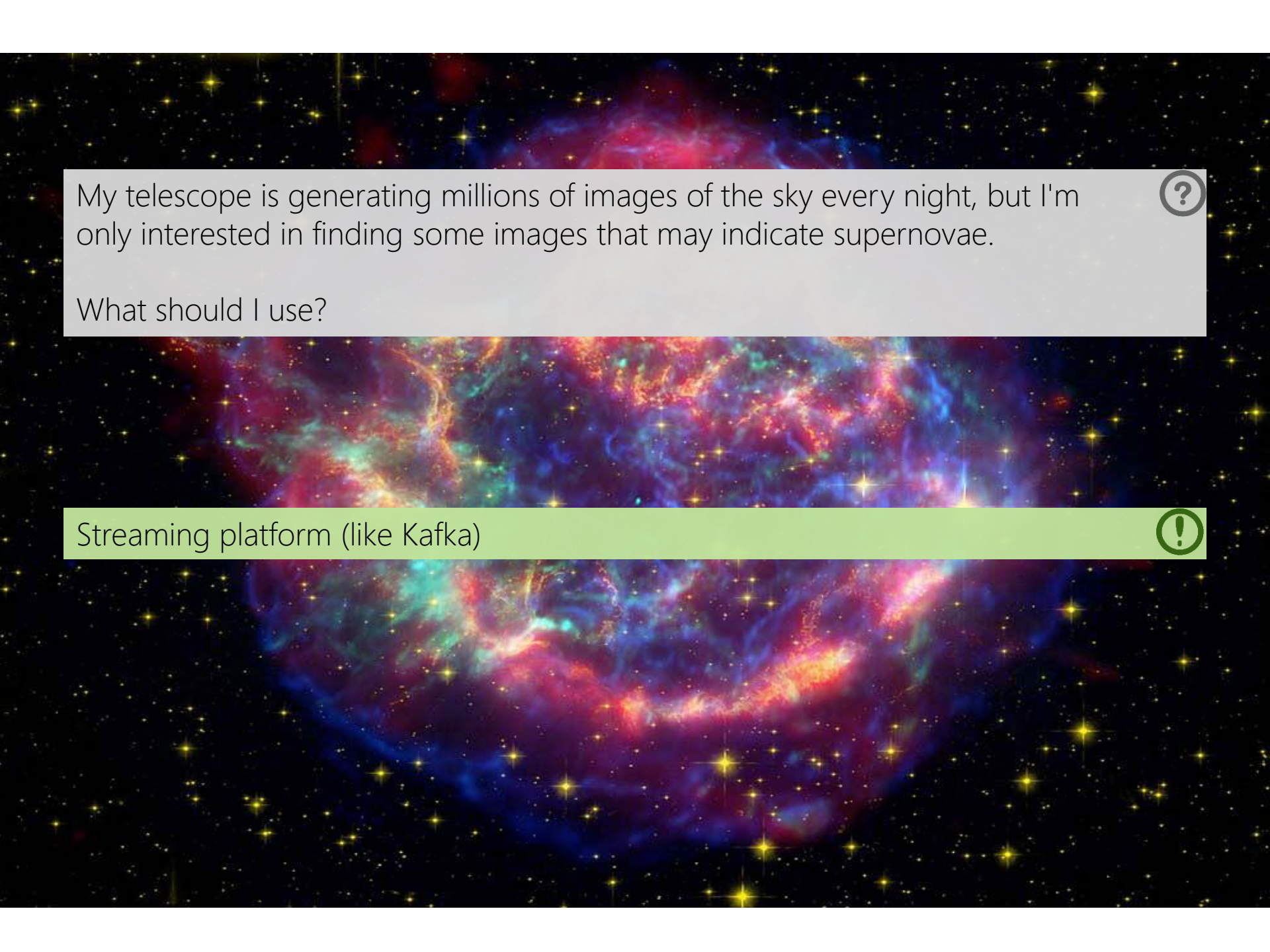


We need to store and query info about bank accounts, insurance, etc.

What should I use?

Relational Database Management System





My telescope is generating millions of images of the sky every night, but I'm only interested in finding some images that may indicate supernovae.



What should I use?

Streaming platform (like Kafka)



I am scraping data about video games and their characters from various wikis. 

In total I have scraped information from about one million pages and now I want to be able to search over what I have, for example to find all non-human characters in a particular video game, or platforming games featuring plumbers.

What should I use?

(Need flexible schema but also expressive query language) 

Document store (e.g., MongoDB)

Graph database (e.g., Neo4j)

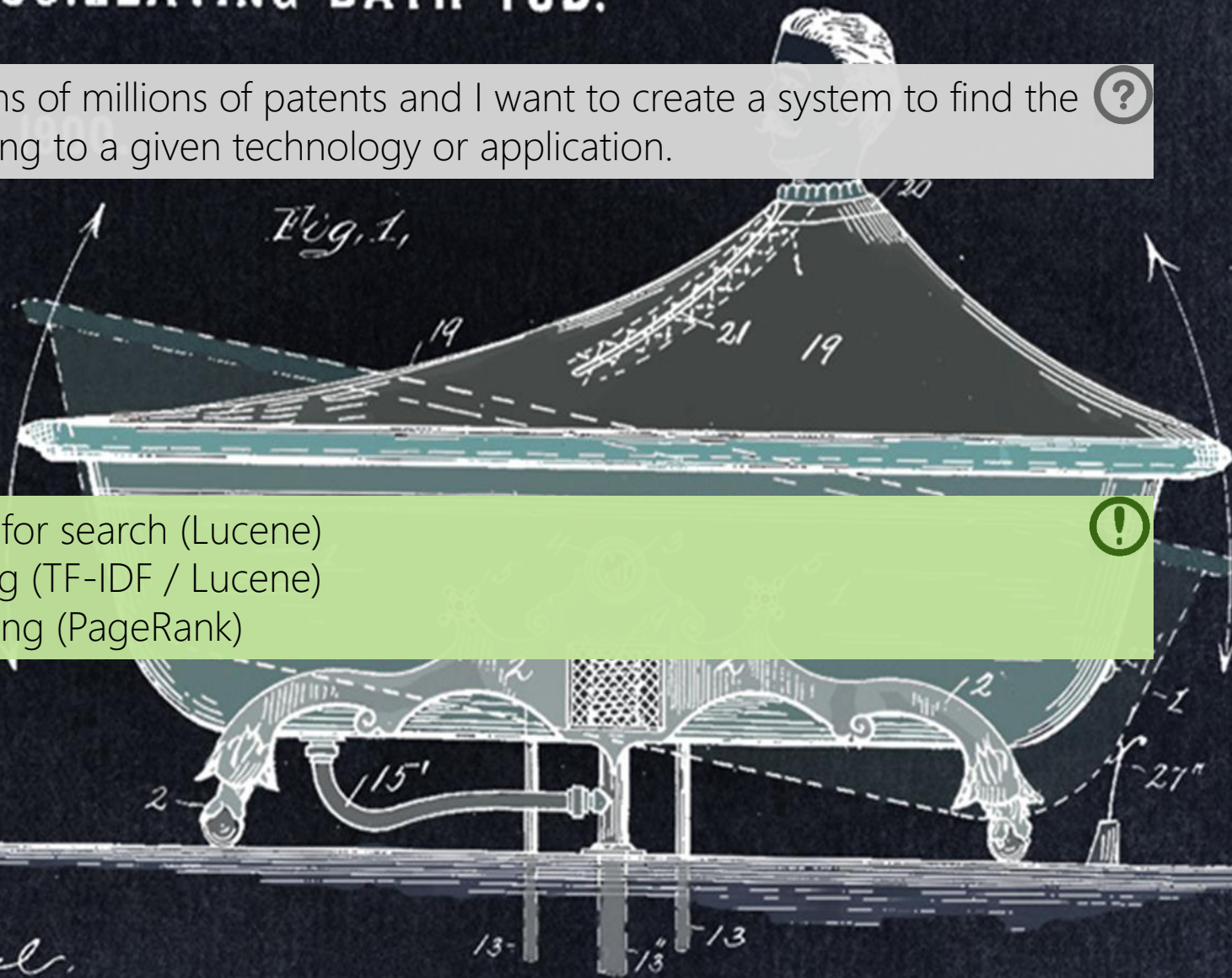
# ROCKING OR OSCILLATING BATH TUB.

O. A. HENSEL

Pat

No. 643,094.


I have descriptions of millions of patents and I want to create a system to find the key patents relating to a given technology or application.



Inverted indexes for search (Lucene)  
Relevance ranking (TF-IDF / Lucene)  
Importance ranking (PageRank)



Inventor:  
Otto A. Hensel.



I have information about tens of millions of papers and millions of movies and I want to compute all the people with a finite Erdős–Bacon number.



What should I use?

MapReduce or Spark (better Spark as the process is recursive)  
(Or even better a graph processing framework: GraphX; Giraph)





I am collecting information about research networks in Latin America.

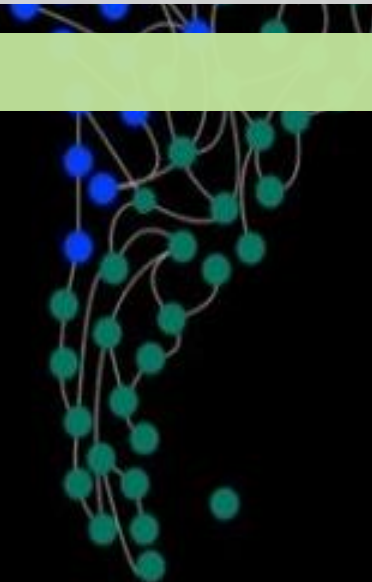


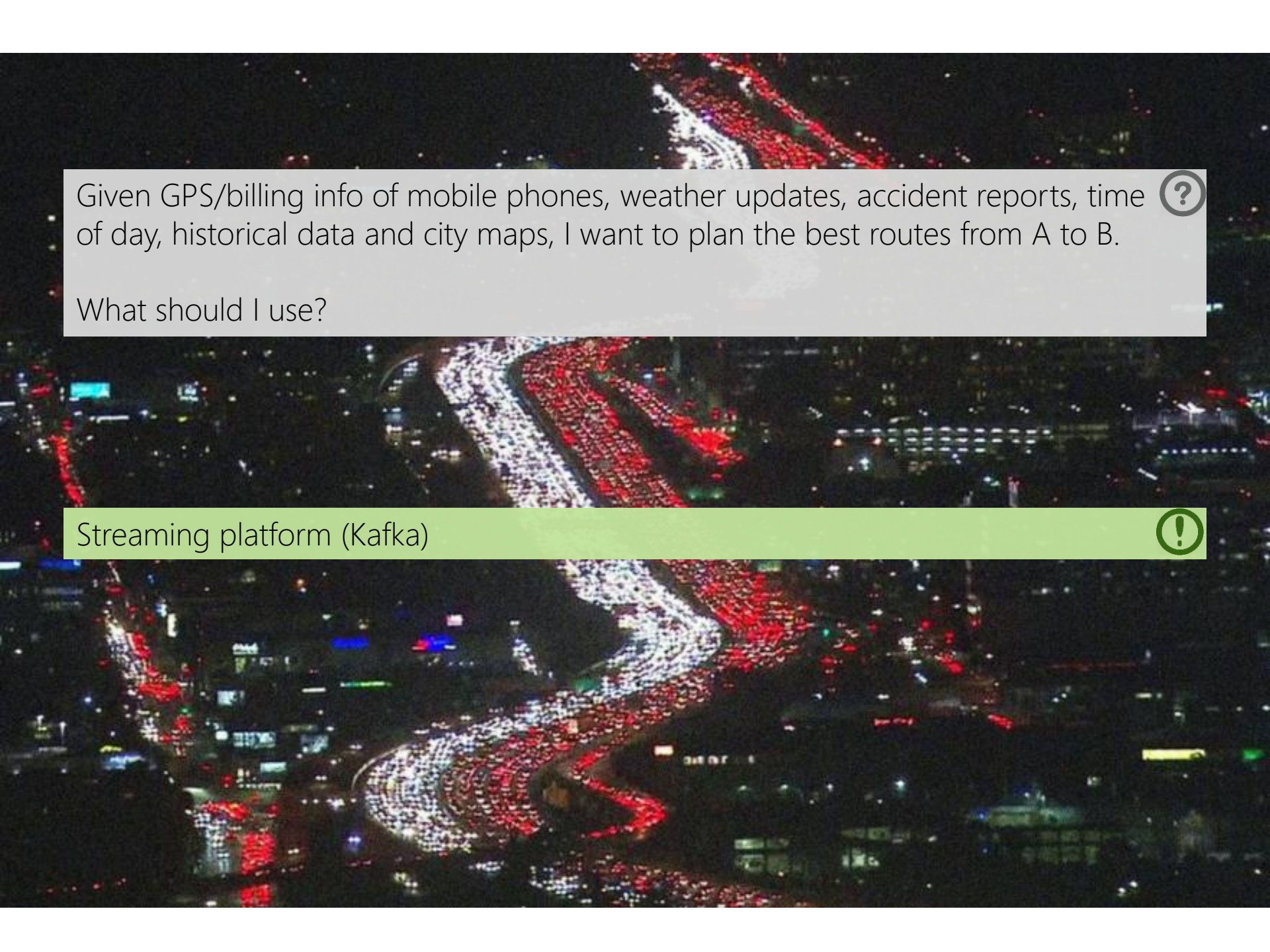
I have information about author affiliations, publications, topics, etc.


Given a particular user, I want to recommend collaborators in the region based on the coauthor network of that user.

What should I use?

Graph database (Neo4j)



An aerial night photograph of a city, showing a dense network of roads. The lights from cars create long, colorful trails of red, white, and blue, indicating traffic flow and congestion. The city lights are visible in the background, creating a vibrant urban scene.

Given GPS/billing info of mobile phones, weather updates, accident reports, time of day, historical data and city maps, I want to plan the best routes from A to B. 

What should I use?

Streaming platform (Kafka) 

I'm working at a cinema.



Given a large collection of movie data (like IMDb), I want to compute profiles for people who work in movies (actors, directors, etc.), including how many movies they have directed or starred in, what are the average ratings of the movies, their most frequent collaborators, awards won, and so forth.

Afterwards when a user visits the cinema webpage, they can hover their mouse over any person to view that person's profile.

What should I use?

MapReduce/Spark to compute profiles  
NoSQL (something like Cassandra, MongoDB)





WRAP-UP ...

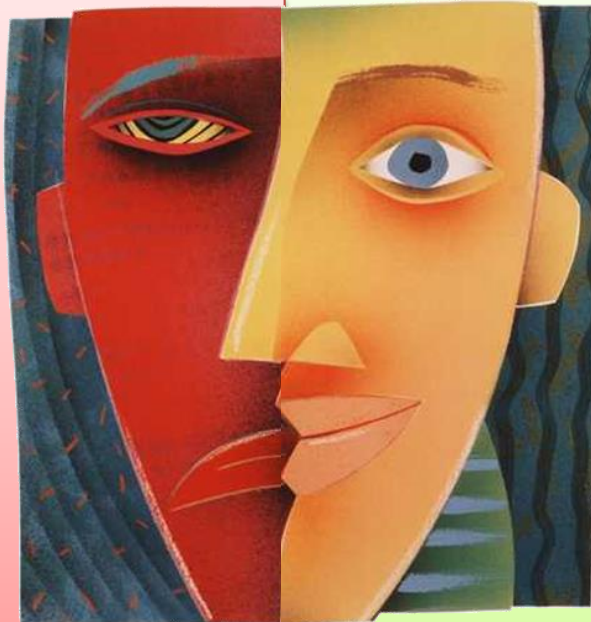
# Course Marking

- 55% for Weekly Labs (~5% a lab!)
- 15% for Class Project
- 30% for 2x Controls

Assignments each week

Controls

Working in groups



Only need to pass overall!

No final exam!

Working in groups!

# Final Exam

# Spoink



Big Data

Pokemon

FINAL BOSS



Eso.