CS196

#10

Announcements

No Homework!!

Midterm

Project Midterm Presentation

Hackathon

Midterm

The midterm will be a take-home exam and will take the place of HW7. You will get one week to work on the midterm, and will be able to work with your peers.

Project Midterm Presentation

CS196

The project midterm presentation will take place on November 14th, during lecture.

Each group will have 5 minutes to present their project progress. Groups will be allowed to have up to 3 Powerpoint slides.

Hackathon

CS196

There will be a hackathon next Saturday, November 11th, from 12-4pm in 0216.

Teams should plan to attend the hackathon at the same time so they can prepare for midterm presentations.

Food will be provided.

Today

Stacks + Queues

Hackerspaces

Mobile Dev: Siebel 1105

Data Science: Siebel 0216

Web Dev: Siebel 1304

Office Hours

There will be no Office Hours this week since there is no homework :)

Attendance

https://goo.gl/iYigpc

Keyword given at end of lecture

CS196

Stacks and Queues

Questions?

Clouds

wat dis

tyler kim

Your ~~second~~ favorite course staff

_

@tyler_thetyrant

Cloud?


Not an actual cloud

We love buzz words :)

Cloud Computing

Why and How?

Big Data

Big Computers

How big is Big Data... really?

Does size matter?

It’s not about the size

But they tend to be big...

5.2 Billion
Google Search

5.75 Billion
Facebook Likes

656 Million
Tweets

67 Million
New Instagram
Posts

4.3 Billion
FB messages
sent

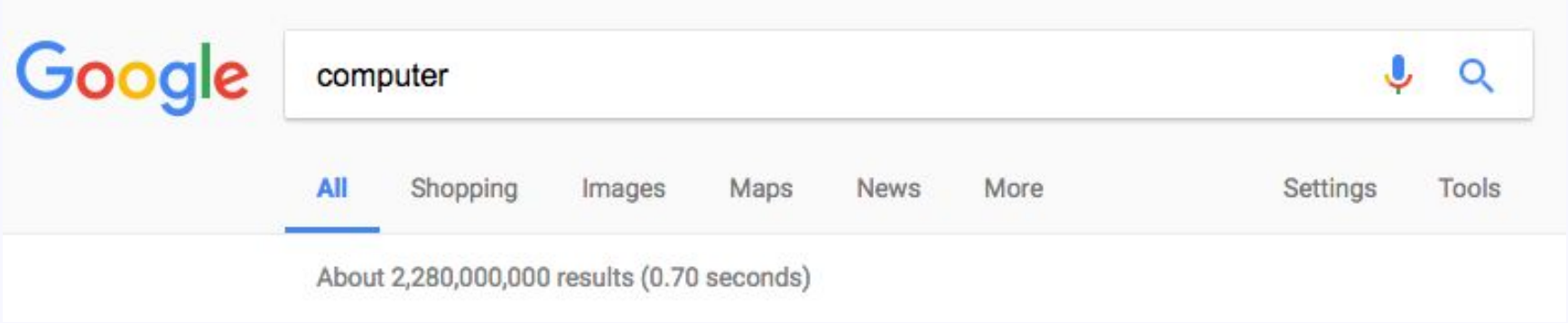
4 Million Hours
New YouTube
content

2.5 Quintillion bytes of new data Everyday

≈

50,000 GB/second

Google Search?



More CPUs?
Better Multithreading?

Still not fast enough...

distributed systems

Just add more computers


USE THE RIGHT TOOL FOR THE RIGHT JOB
CARTOONSTOCK .com
Search ID: pknn805

BIG DATA LANDSCAPE 2017

© Matt Turck (@mattturck), Jim Hao (@jimrhao), & FirstMark (@firstmarkcap) mattturck.com/bigdata2017

FIRSTMARK
EARLY STAGE VENTURE CAPITAL

Big Data or Pokemon?

Spoink

Spoink is Pokemon!

Spoink is a bouncy psychic-type Pokémon that should not be confused with Splunk, a software suite for analyzing log data.

Flink

Flink is Big Data!

Apache Flink will help us with batch and stream processing, but the logo is cute enough to become Pokémon.

Feebas

Feebas is Pokemon!

Feebas requires Swift Swim to get marvel scale.

Impala

Impala is Big Data!

Querying big data is too slow? Impala has a solution for it. But reconsider taming that one, it is built on C++.

Arvados

Arvados is Big Data!

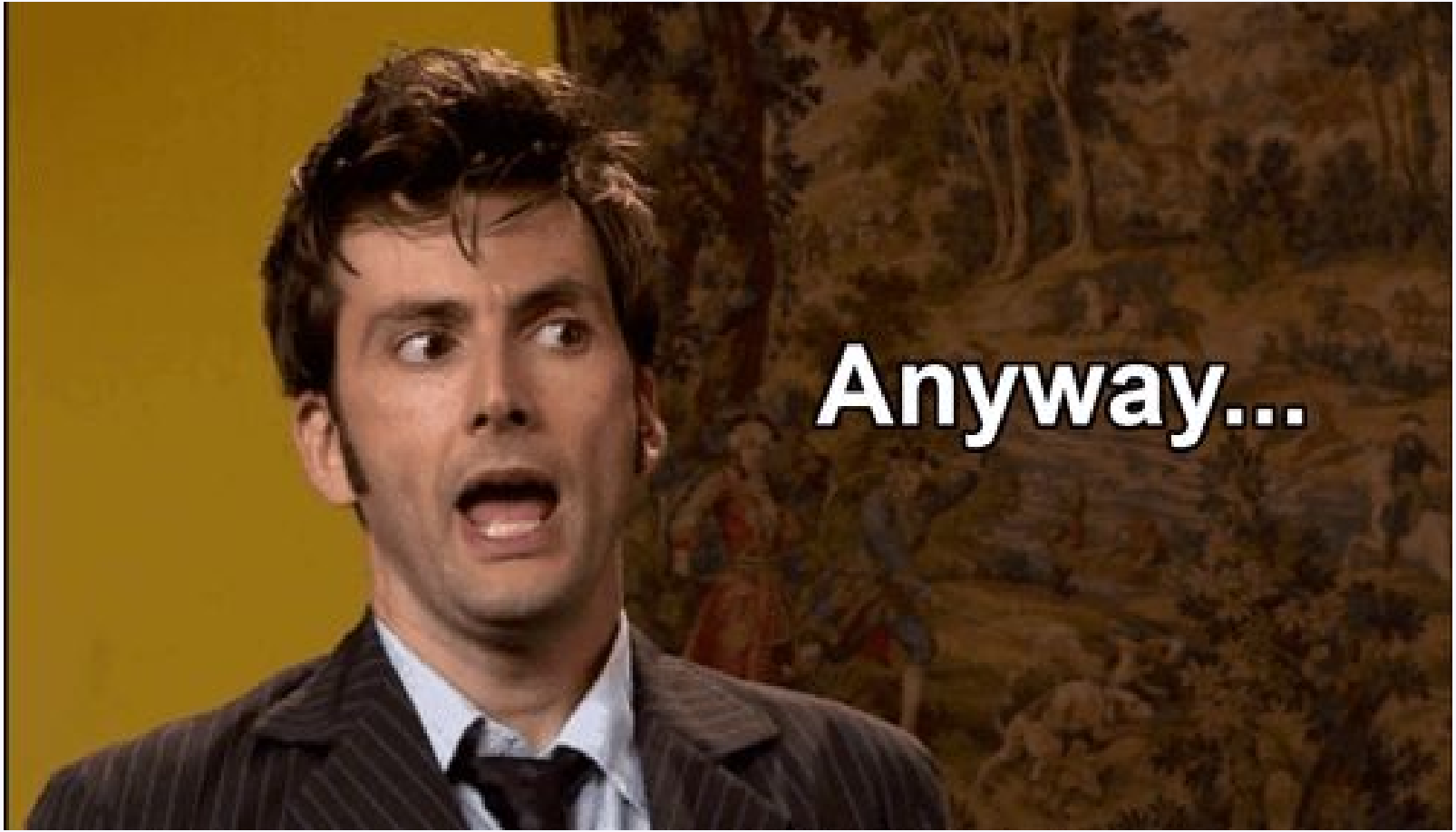
Not to be confused with Ariados. Spins a web of microservices around unsuspecting sysadmins.

Azurill

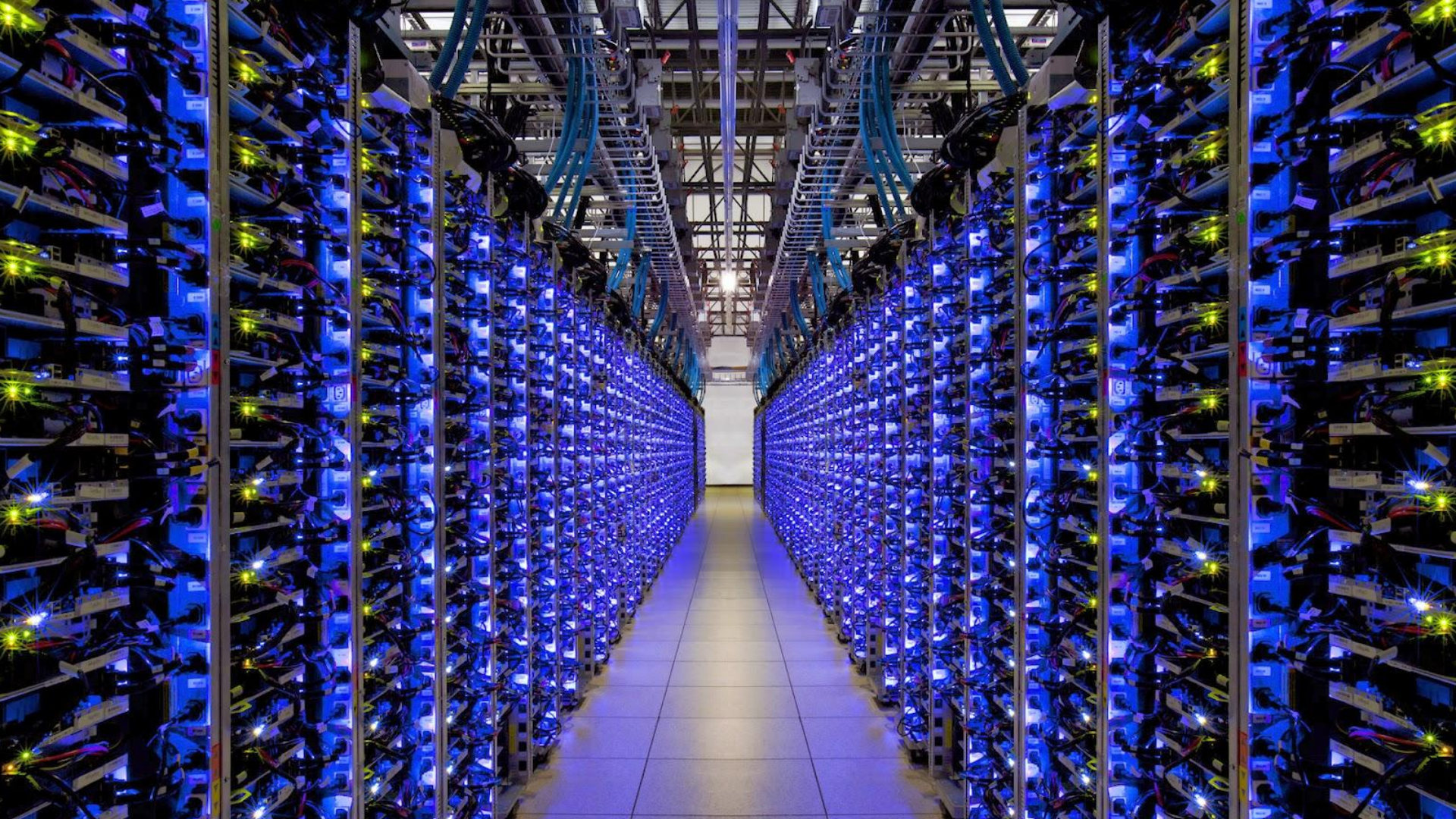
Azurill is Pokemon!

Azurill's tail is large and bouncy. Azurill can be seen bouncing and playing on its big, rubbery tail. On sunny days they gather at the edge of water and splash about for fun.

Why do they all sounds like Pokemons?


Anyway...

What else do we need?



Why use a cloud?

- **Reliability**
  - It's someone else's responsibility to fix broken machines
- Cheap and On-Demand **Scalability**
  - Pricing is per hour or second instead of sunk hardware cost
  - Can create and destroy nodes on a *per second* basis
- **Hardware** Abstraction
  - Don't have to care about underlying hardware, just the specs of your VM
- "Special Sauce"
  - Proprietary features (i.e. AWS DynamoDB or Google BigQuery)


Microsoft
Azure


Google Cloud Platform

Microsoft Azure

amazon web services

Google Cloud Platform

The largest by far of the public clouds

- You use it every day and don't even know it
- Netflix, Reddit, Spotify, and millions others

When it goes down, the half of the internet goes down

- Example: The infamous S3 outage in February 2017


Service is operating normally
S3
S3

CS196

Compute
EC2
EC2 Container Service
Lightsail
Elastic Beanstalk
Lambda
Batch

Storage
S3
EFS
Glacier
Storage Gateway

Database
RDS
DynamoDB
ElastiCache
Redshift

Networking & Content Delivery
VPC
CloudFront
Direct Connect
Route 53

Migration
Application Discovery Service
DMS
Server Migration
Snowball

Developer Tools
CodeCommit
CodeBuild
CodeDeploy
CodePipeline
X-Ray

Management Tools
CloudWatch
CloudFormation
CloudTrail
Config
OpsWorks
Service Catalog
Trusted Advisor
Managed Services

Security, Identity & Compliance
IAM
Inspector
Certificate Manager
Directory Service
WAF & Shield
Compliance Reports

Analytics
Athena
EMR
CloudSearch
Elasticsearch Service
Kinesis
Data Pipeline
QuickSight

Artificial Intelligence
Lex
Polly
Rekognition
Machine Learning

Internet Of Things
AWS IoT

Contact Center
Amazon Connect

Game Development
Amazon GameLift

Mobile Service
Mobile Hub
Cognito
Device Farm
Mobile Analytics
Pinpoint

Application Services
Step Functions
SWF
API Gateway
Elastic Transcoder

Messaging
Simple Queue Service
Simple Notification Service
SES

Business Productivity
WorkDocs
WorkMail
Amazon Chime

Desktop & App Streaming
WorkSpaces
AppStream 2.0

amazon web services

AWS Elastic Compute Cloud (EC2)

- The basic one which all of these clouds provide are Virtual Machines
- AWS has everything from the tiny to gigantic monsters
  - T2.Nano: 1 VCPU 512 MB Ram
  - X1.32xlarge: 128 VCPU 2000 GB Ram (One of these is more powerful than our cluster)
- They have GPUS!
  - Can do deep learning
- Most are fixed price per hour but there is a price auction for unused machines
  - Lets you do stuff super cheap as long as your program can handle getting a shutdown notice within 30 seconds

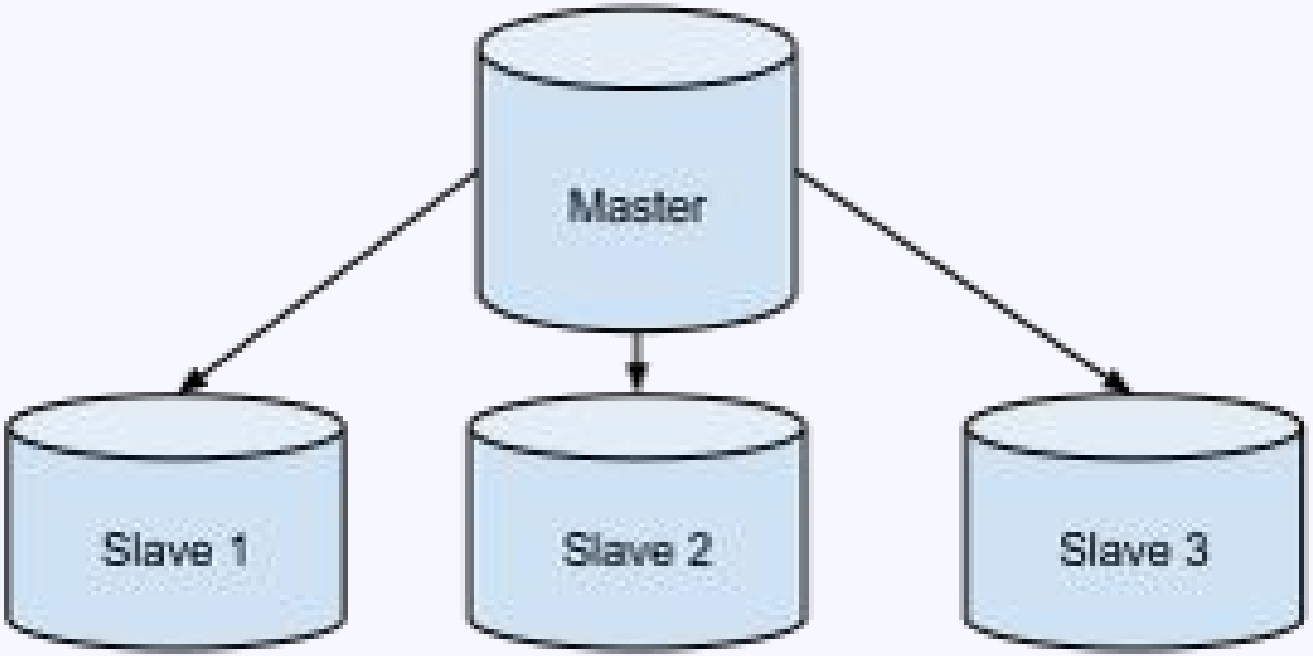
Azure Virtual Machines

- Similar to AWS
- GPUs
- Not as many CPUs (Max is 32 currently)
- Not as much ram (Max 800 GB currently)
- But you probably will not hit these limits

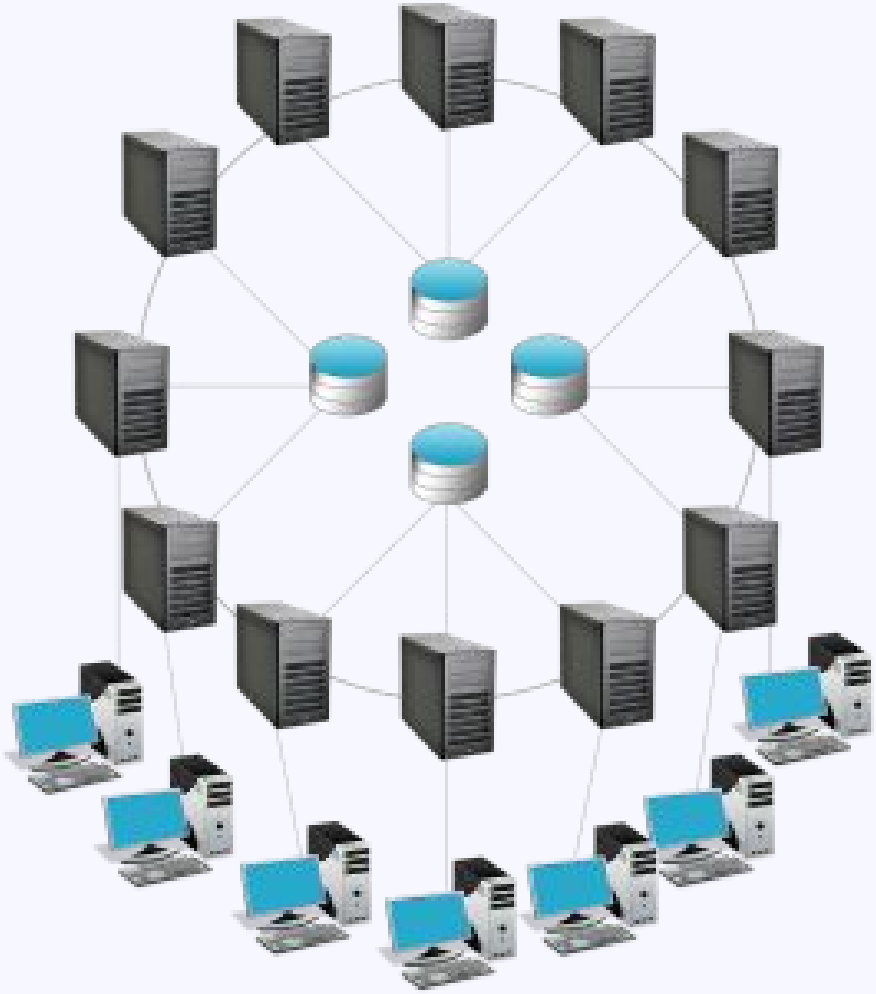
Google Compute Engine

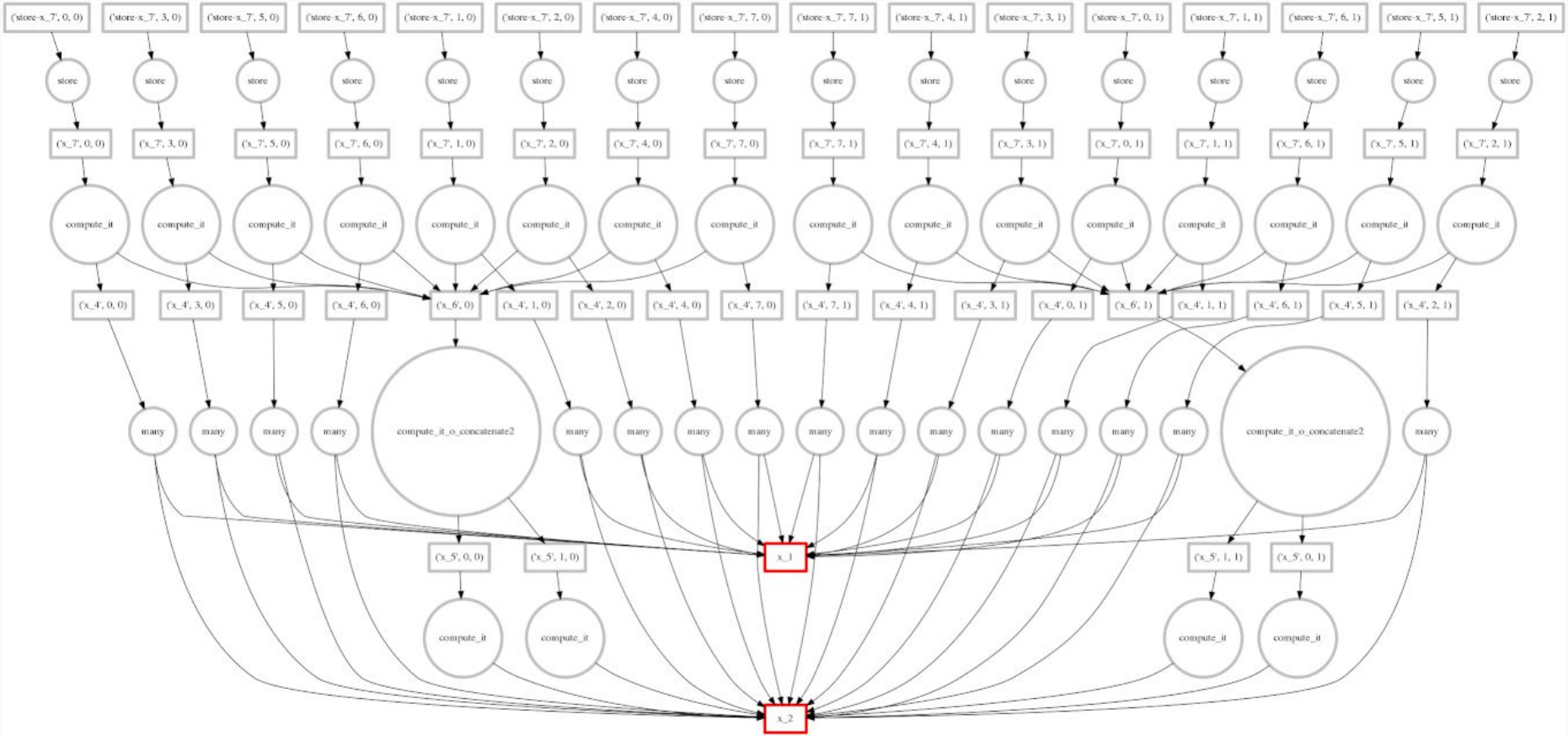
- Provides VMs
- Largest server is 96 VCPU, 624 GB Ram
- Provides custom sized machines
- Cost is per minute!!

Just one big computer…
Isn’t enough.


Master
Slave 1
Slave 2
Slave 3

Let’s look at GCP for a bit





Matthew Rocklin, Dask, 2017

This changes how we write codes

We can no longer consider our code to only run sequentially on one computer

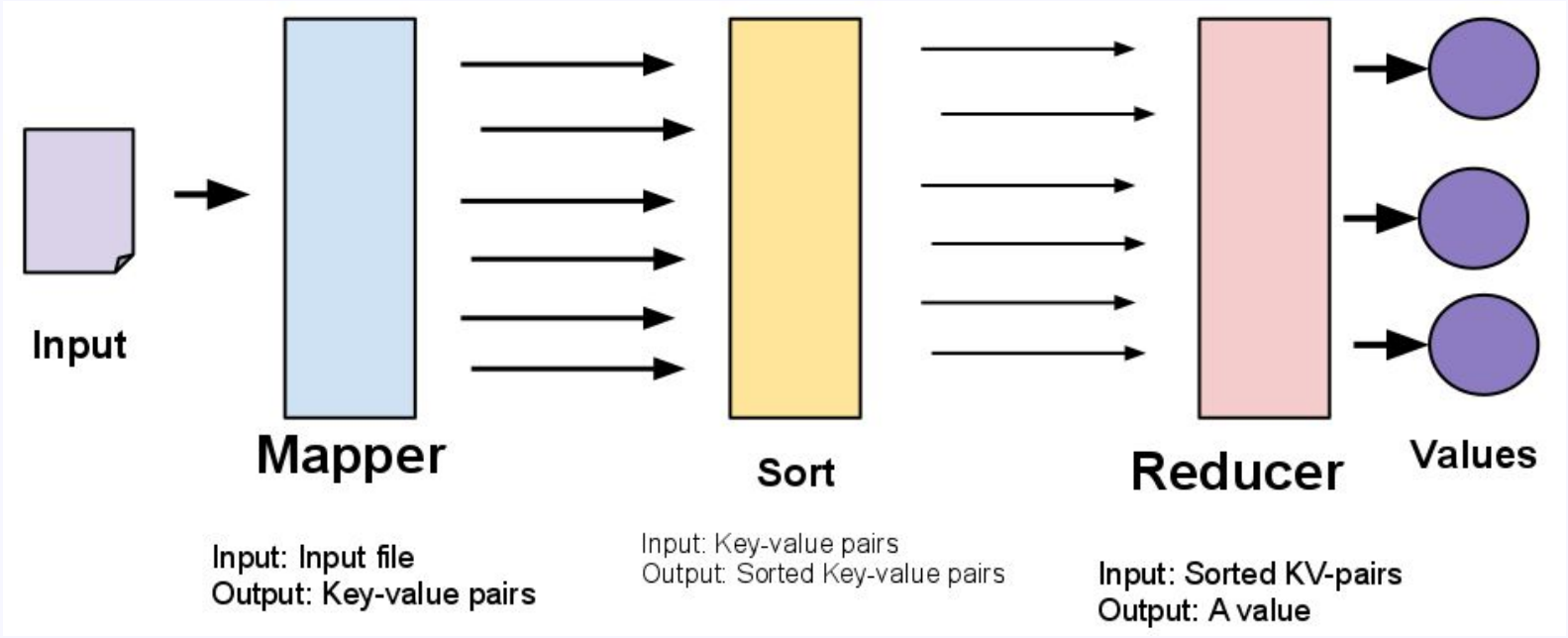
MapReduce

A MapReduce job usually splits the input data-set into independent chunks which are processed by the *map* tasks in a completely parallel manner.

The framework sorts the outputs of the maps, which are then input to the *reduce* tasks.

Map -- A function to process input key/value pairs to generate a set of intermediate key/value pairs. All the values corresponding to each intermediate key are grouped together and sent over to the Reduce function.

Reduce -- A function that merges all the intermediate values associated with the same intermediate key.


Input
Mapper
Input: Input file
Output: Key-value pairs
Sort
Input: Key-value pairs
Output: Sorted Key-value pairs
Reducer
Values
Input: Sorted KV-pairs
Output: A value

How can we rewrite this code on multiple computers?

```
arr = range(10000000)
evens = [ ]
for i in arr:
    if i % 2 == 0:
        evens.append(i)
```

```
arr = chunks(range(10000000)) # Break arr into chunks
evens = []
index = 0
for chunk in arr:         # run each chunk on a different
computer
    for i in chunk:
        if i % 2 == 0:
            evens[index].append(i)
    index += 1
# more code to recombine the lists of even numbers
```

Mapping

```
map(isEven, [0,1,2,3,4])
>   [True, False, True, False, True]
map(addOne, [0,1,2,3,4])
>   [1,2,3,4,5]
```

- By default map only runs on one processor like normal code so there is no speedup.
- But map can be rewritten to run on multiprocessors at the same time or even multiple computers
- Every map function is equivalent to a for loop

Reducing

```
reduce(lambda x, y: x+y, [1, 2, 3, 4, 5])
```

> Calculates `((((1+2)+3)+4)+5)`

Reducing

Map returns an array of results, but a lot of the time you only want one final result

```
reduce(
    function(accumulator, currentElement),
    array
)
results = map(isEven, [0,1,2,3,4])    ## [True, False, True, False, True]
F = lambda total, curEle: total + 1 if curEle == true else total
numEven = reduce(F, results)
numEven == 3
```

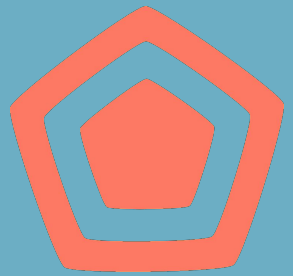
CS196

Cloud Orchestration

Gotta train your clouds :)

Where to learn Big Data and cloud computing on campus?

Find out tomorrow at my Hackerspace



Keyword

Trickortreat

https://goo.gl/iYigpc