CS15-319 / 15-619
Cloud Computing

Recitation 9

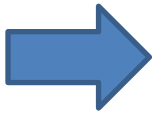
October 21st and 23rd, 2014

Announcements

- Encounter a general bug:
  - Post on Piazza
- Encounter a grading bug:
  - Post Privately on Piazza
- Don’t ask if my answer is correct
- Search before posting
- Post feedback on OLI
- **OLI does not show timer for Quiz! You have to maintain your own timer!**

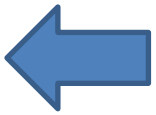
Modules to Read

- UNIT 4: Cloud Storage
  - Module 12: Cloud Storage
  - Module 13: Case Studies: Distributed File Systems
  - Module 14: Case Studies: NoSQL Databases
  - Module 15: Case Studies: Cloud Object Storage



Project 3

- Files vs. Databases
  - File vs. Database
- Vertical Scaling in Databases
  - Vertical Scaling
- Horizontal Scaling in Databases
  - Horizontal Scaling
- Sharding Databases
- Provisioned Databases



P3.2 Reflections

- Comparison? (Flat files, MySQL and Hbase)
  - Magnetic:
    - A read/write head on an arm accesses data stored in a spinning metal platter with magnetic coating;
    - Uses more electricity and produces more heat and noise;
    - Higher latency, longer r/w times and fewer IOPS than SSD.
  - SSD:
    - No moving parts. Interconnected flash memory chips;
    - Uses less electricity, no noise;
    - Lower latency, faster r/w times and more IOPS than Magnetic.

 - Caching improves performance between runs.

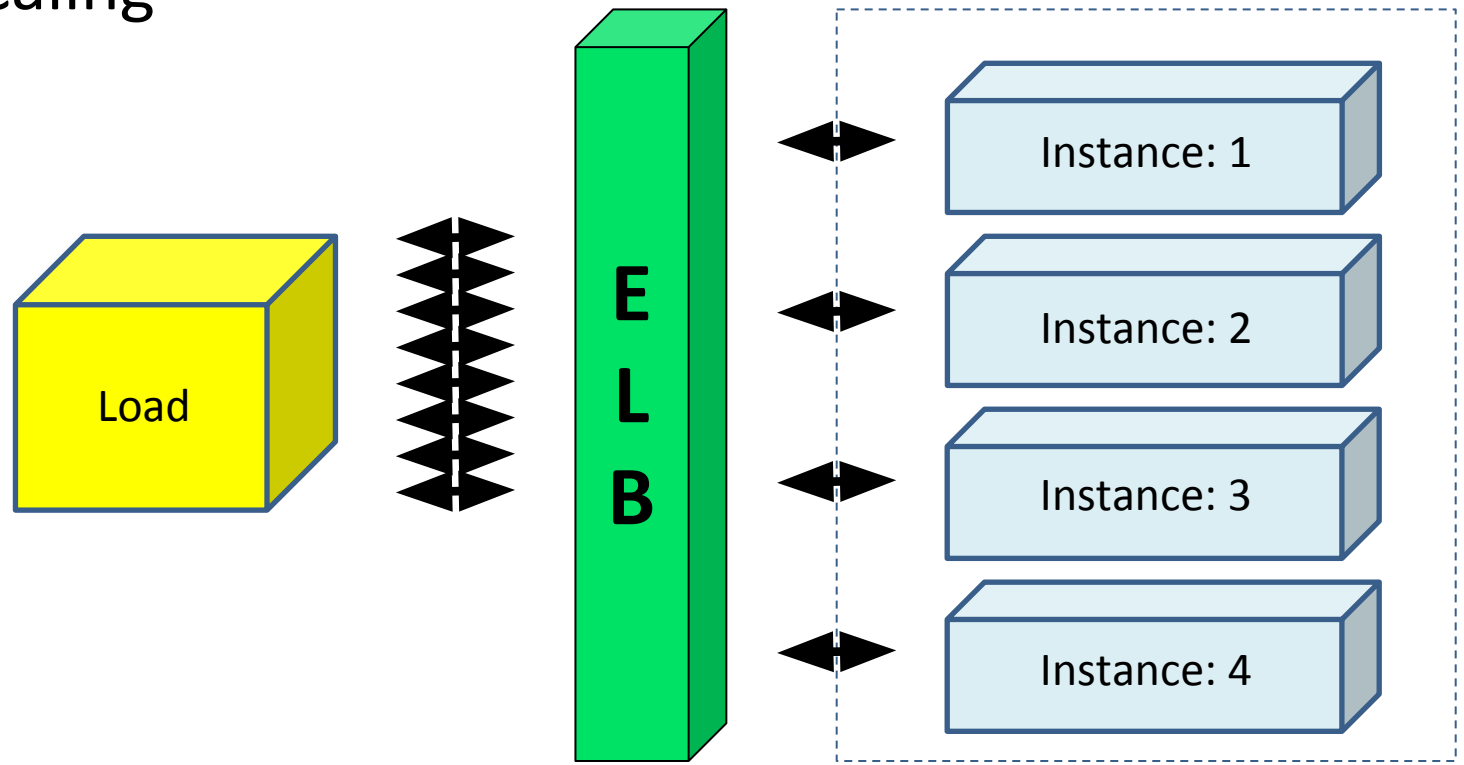
Piazza Questions

MasterNotRunningException:

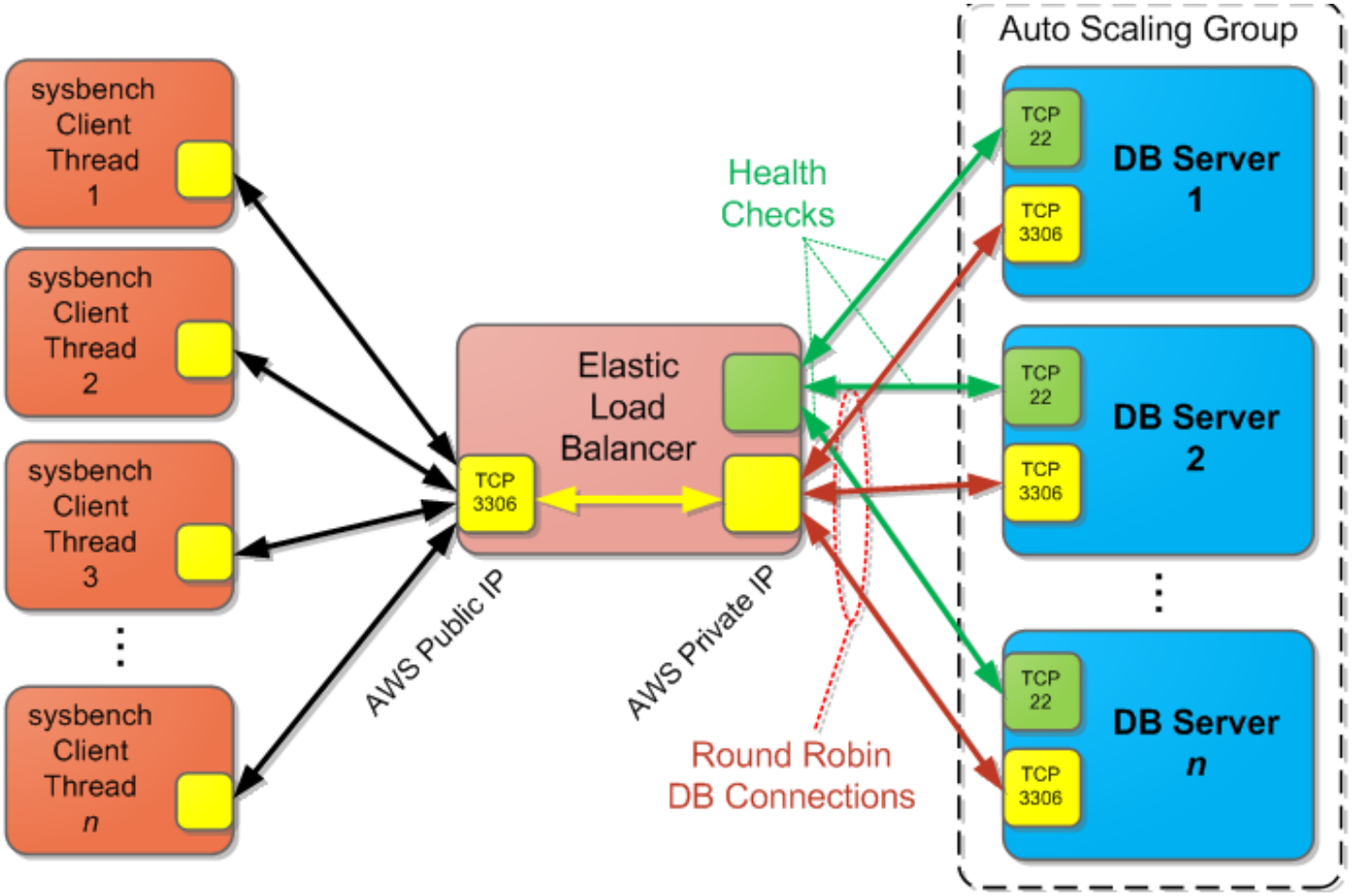
- When data directory of a node is changed, Hbase tries to recover by copying the metadata from other data nodes. But in our case some students changed the directory in all data nodes, leaving Hbase no time to recover. This corrupts metadata in Hbase master.
- Solution 1 – Reconfigure on data node at a time
- Solution 2 – Make Hbase root directory point to new directory, so master starts a fresh Hbase setup.

Project 3 Module 3

- This week, you will explore the database performance by increasing the number of servers i.e. Horizontal Scaling



Horizontal Scaling



Horizontal Scaling in Databases

- Scenario

  Your company has a popular wiki-based website, which has a strong database requirement. You are a member of the database team, and your task is to create a group of read-only clones of the master database.

To Do

- Task1 :
  - Create the TPS projection for the next year
  - Weekly data:
    - s3://15619public/proj3/week0.csv
  - Each week's traffic should be ((12%/52)x week0) higher than the previous week.

To Do (Continued)

- Task2 :
  - Determine the theoretical maximum TPS for a single m1.small. Use m1.small instance types in an Auto-Scaling Group (ASG) behind an Elastic Load Balancer (ELB)

To Do (Continued)

- Task3:
  - Number of servers = Projected_TPS/75% ×max_TPS_per_server.
  - Keep scaling the number of servers behind ELB to achieve the projected TPS.
  - How to calculate TPS every time you scale a server?
    - Sysbench and Custom CloudWatch Metric.
  - Verify that the TPS, which the CloudWatch metric reported, is the same TPS that sysbench is reporting on the client side.

Custom CloudWatch Metric

- Build and use a custom metric that will report Transactions-per-second (TPS) of the MySQL server.
- NOTE: You do not have to turn on detailed metrics to use custom metrics.
- Write a script to calculate TPS utilization (details are in the hints on OLI).
- In order to send a custom metric value to CloudWatch, you need some way of authenticating:
  - Two ways to do that:
    - Using credentials file. Danger! If system has security breach.
    - Use IAM (Identity and Access Management) roles to gain temporary credentials (Optional).

Identity and Access Management Role

- Create an IAM role.
- Define which accounts or AWS services can assume the role.
- Define which API actions and resources the application can use after assuming the role.
- Specify the role when you launch your instances.
- Have the application retrieve a set of temporary credentials and use them.

To Do (Continued)

- Task 4
  - Calculate **maximum Transactions-per-dollar** for each instance type.
    - The AWS Monthly Calculator might be helpful for getting an idea of total costs.
- Task 5:
  - Using the instance type with the highest Transactions-per-dollar you found above, calculate the total projected cost of your databases for a year.
    - Include ELB and Cloudwatch cost.

Potential Roadblocks

- Research on Linux tools that allow for automation on instance startup, for the AMI you are going to create for the ASG;
- Understanding MySQL status values for Uptime and Queries and how they change.

Upcoming Deadlines

- Project 3 Module 3:

Project 3			
	Horizontal Scaling in Databases		
	Horizontal Scaling	Checkpoint	Sunday, 26th October

TWITTER ANALYTICS:
THE 619 PROJECT

Twitter Analytics Web Service
A 619 Competition

Congratulations Team FDU

- Building a performant web server is non-trivial
- Building a performant web server that gets huge amounts of data from MySQL is hard
- Building a performant web server that gets huge amounts of data from HBase is harder

- **START NOW!!!**

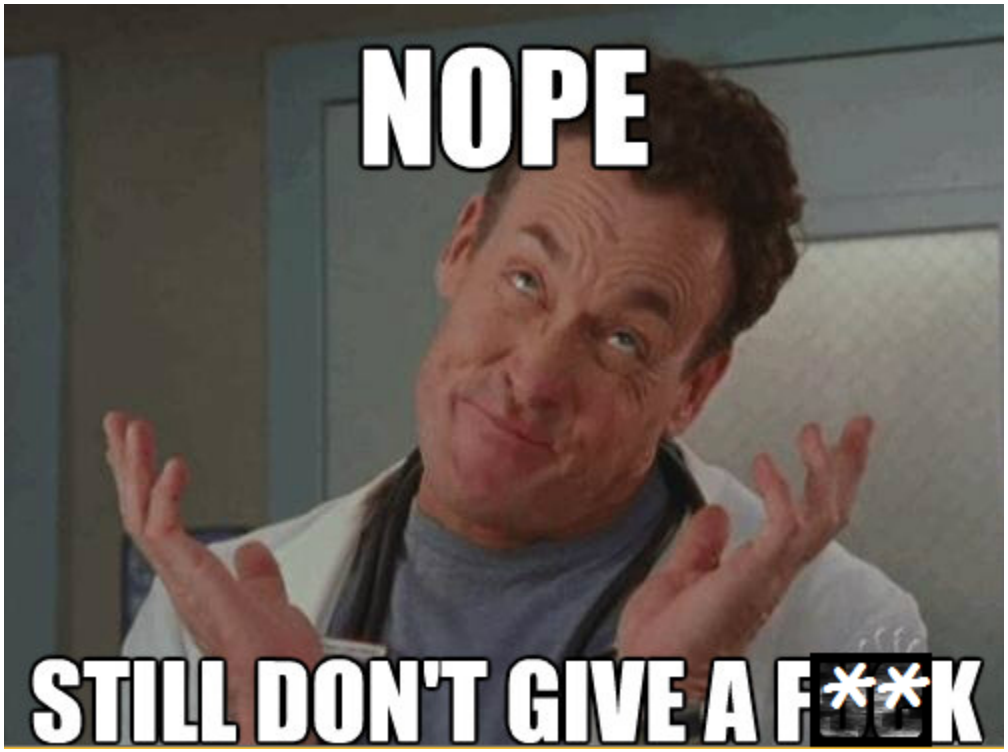
Congratulations Team FDU

- Building a performant web server is non-trivial
- Building a performant web server that gets huge amounts of data from MySQL is hard
- Building a performant web server that gets huge amounts of data from HBase is harder

- **TRAVEL BACK IN TIME AND START !!!**

It’s only 1.5%

Good for you



Updated Sample File

- Small bug in the output
- If you need it, download it again
- Thanks Zichang Feng for pointing this out

Students' Favourite Queries

- Are there any duplicates in the dataset? How do we deal with them?
  - Yes
  - Unique-ify them

- Some AFINN/banned words are non-alphanumeric
  - Ignore them
  - They will not match any tweet text post-split

More questions

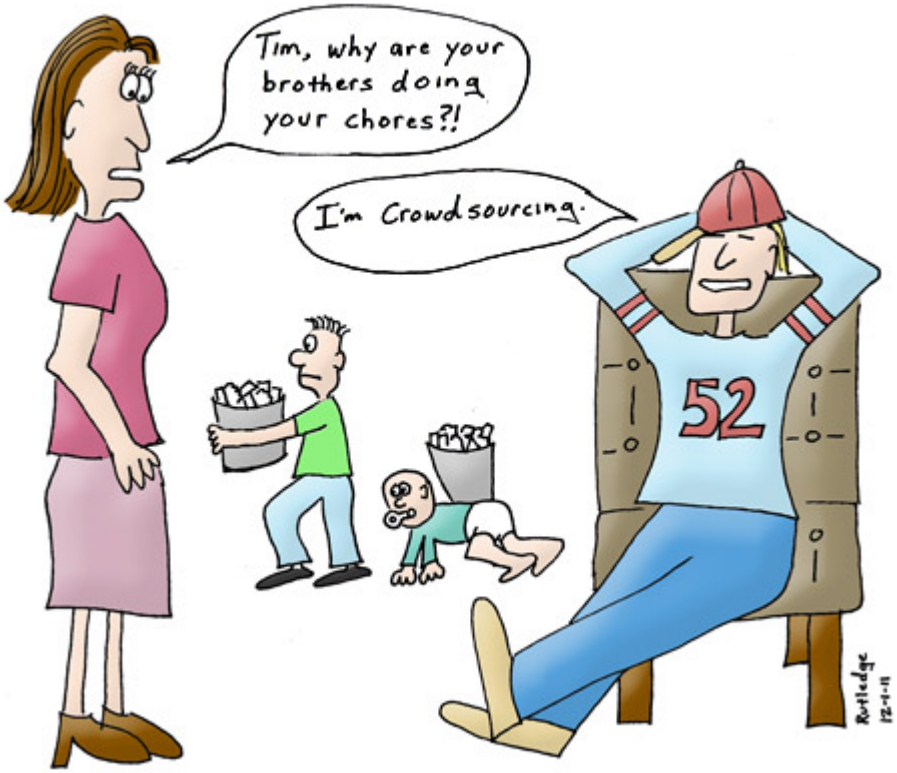
- How do I parse JSON? Error a.b.c?
- How do I configure server/framework XYZ?
- How do I store so much data on MySQL?
- How do I insert data into HBase?

RTFM

Before you ask those kinds of questions.

Piazza

- Please help each other
- Discuss ideas. Not answers.
- Share resources.



Credits

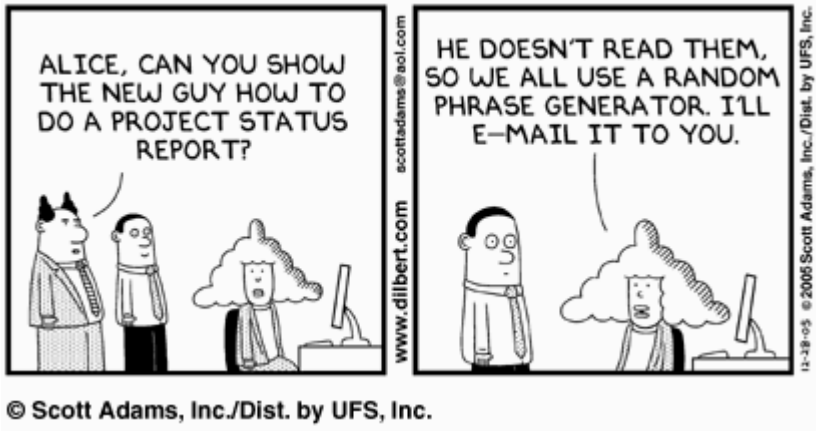
- Thanks Behrouz Rabiee [@1494]

```
$('#table tr').prepend('<td class="cnt">1</td>').each(function(idx){$(this).children(":eq(0)").html
((idx == 0 ? '#' : idx));});
```

- Thanks Eric Adlam (for the metrics dashboard in last week's recitation)

Phase 1 Report

- Start early
- Document your steps



- Identify and isolate the performance impact of each change you make
- Document your ideas and experiments

Phase 2

- Two new queries [Q3 and Q4] on 1 TB data
- Tougher to meet Q1 and Q2 requirements
- Live Test
- Mixed Queries

Suggestions / Improvements

- UI
- Favicon
- Design
- Features
- Bugs

http://bit.ly/1roJsvU

Any questions?