# CS15-319 / 15-619 Cloud Computing
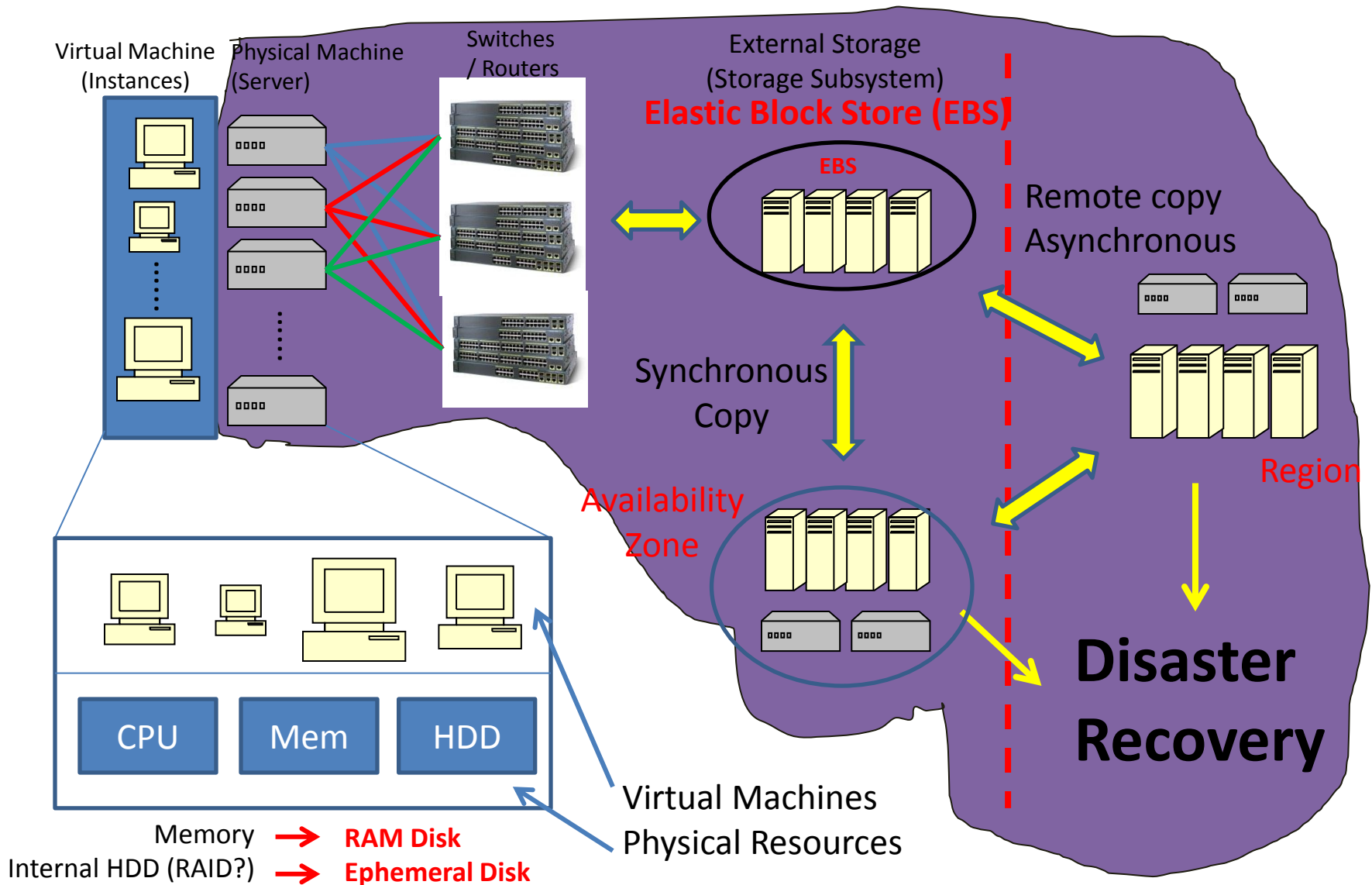
Recitation 9

March 18th and 20th, 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
- Don't post code on Piazza
- Search before posting
- Post feedback on OLI

# Piazza Questions

- What's the difference between storage types ephemeral disk and EBS?
  - Ephemeral (a.k.a instance store) drives are local disks on the nodes where your instance is running and is automatically available to the instance. Any data on the ephemeral disk is discarded when the instance is terminated.
  - EBS is block storage that can be attached to an instance. The device name depends on what you choose when you assign a volume to an instance, and the size can be whatever size you choose (8GB for example). You can either create an EBS volume during the Instance launch wizard, or you can create them afterwards under volumes and attach the volume to your running instance.

# Different Types of Storage



Virtual Machine (Instances)

Physical Machine (Server)

Switches / Routers

External Storage (Storage Subsystem)

**Elastic Block Store (EBS)**

EBS

Remote copy Asynchronous

Synchronous Copy

Availability Zone

Region

**Disaster Recovery**

CPU | Mem | HDD

Virtual Machines Physical Resources

Memory → **RAM Disk**

Internal HDD (RAID?) → **Ephemeral Disk**

# Piazza Questions

- mkfs returned error: /dev/xvdb1 is apparently in use by the system; will not make a filesystem here?
  - /dev/xvdb is mounted to /mnt by default.
  - Before creating file system, you need to umount /dev/xvdb partition from /mnt

# Project 3, Module 2 Reflections

- Local tests (from the server) are always faster than remote tests (from an m1.small client).

- EBS can be faster than Ephemeral disk (Instance Store) and RAID0 Ephemeral disk.

- EBS (improved bandwidth) is almost as fast as ramdisk. EBS is more expensive.

- Optimized EBS does not necessarily provide better performance for the application tested.

# Module 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
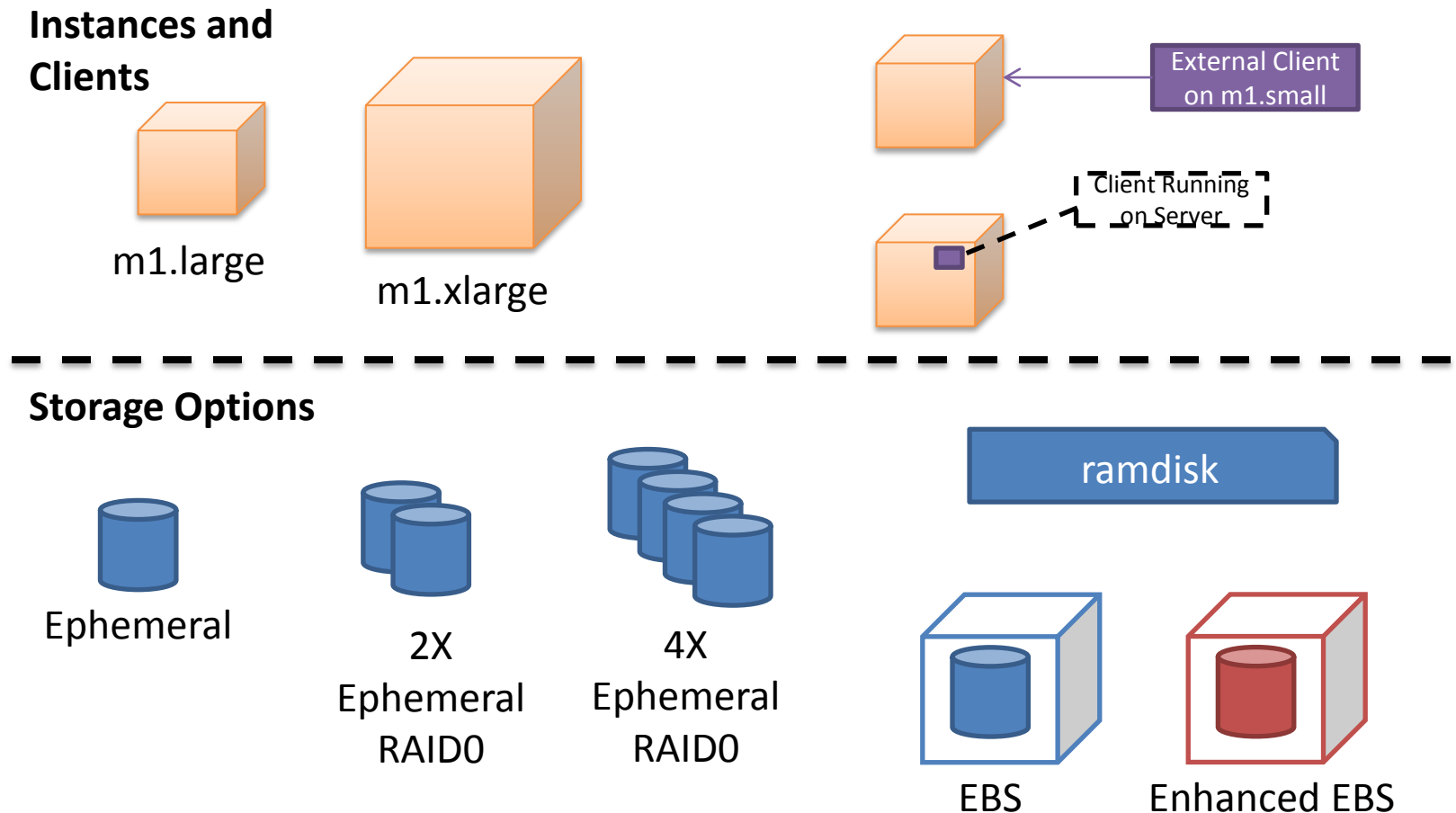  - Quiz 4: Cloud Storage

# Project 3

- Files vs. Databases
  - File vs. Database

- Vertical Scaling in Databases
  - Vertical Scaling

- Horizontal Scaling in Databases
  - Horizontal Scaling

- Working with NoSQL: DynamoDB / Hbase
  - Amazon DynamoDB
  - DynamoDB vs. HBase

# Project 3 Module 2

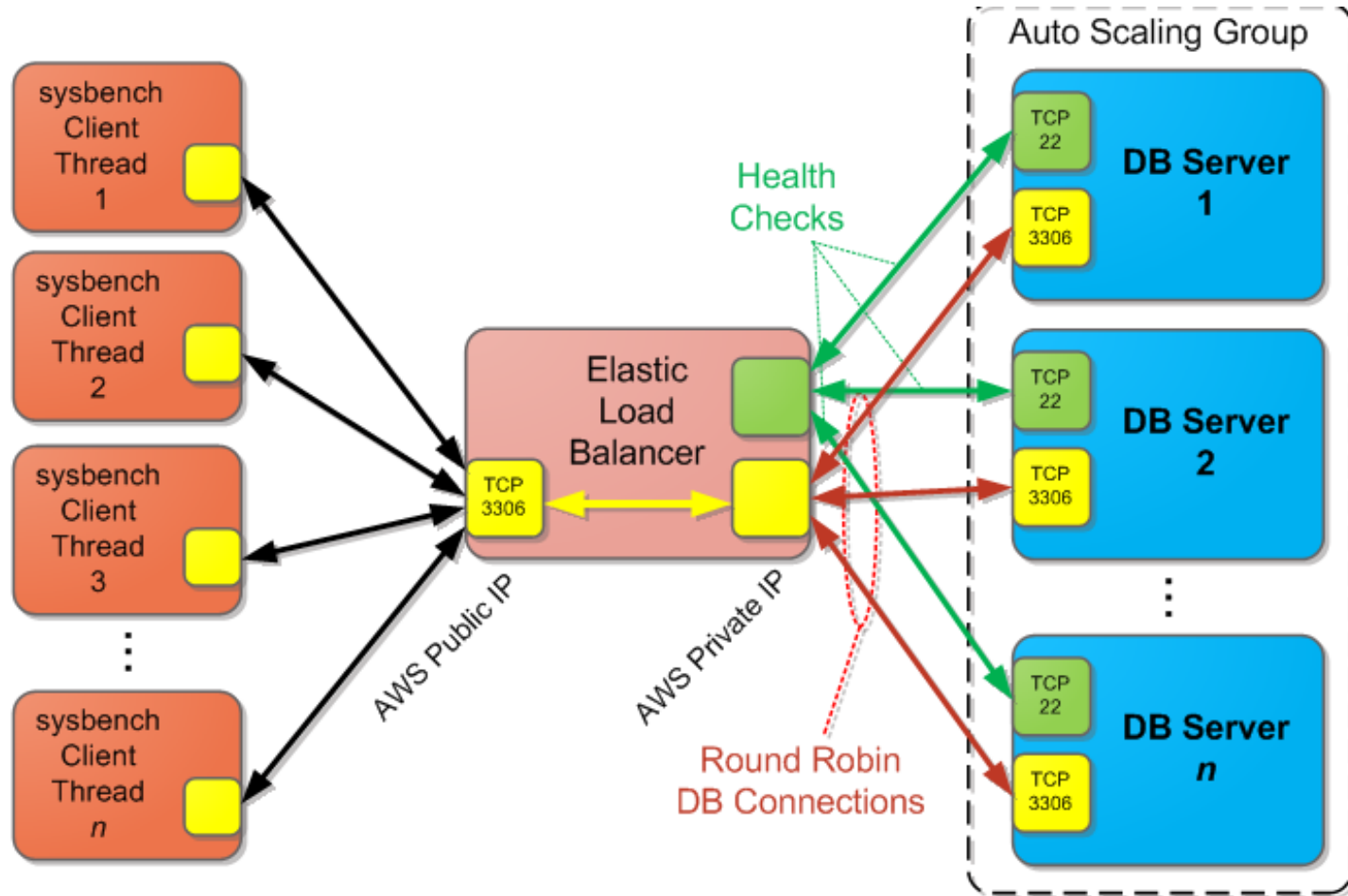- Last week we evaluated the TPS of different storage systems (vertical scaling)

**Instances and Clients**



m1.large

m1.xlarge

External Client on m1.small

Client Running on Server

**Storage Options**



Ephemeral

2X Ephemeral RAID0

4X Ephemeral RAID0

ramdisk

EBS

Enhanced EBS

# Project 3 Module 3

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

**Load**

**E L B**

**Elastic Load Balancer**

Instance: 1

Instance: 2

Instance: 3

Instance: 4
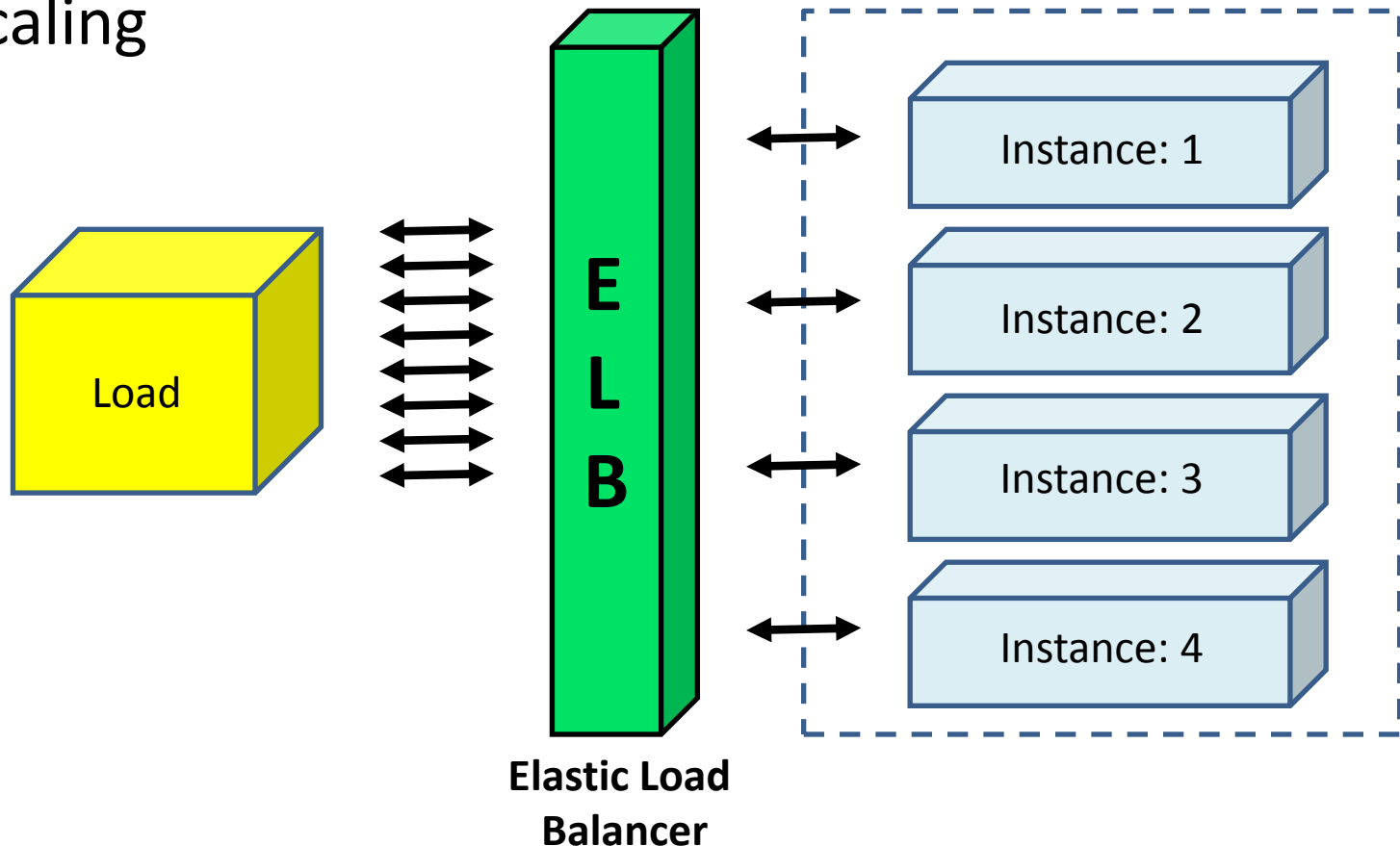
# Horizontal Scaling

# 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://15-319-s13/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 (like you did in previous the checkpoint for m1.large and m1.xlarge). Use m1.small instance types in an Auto-Scaling Group (ASG) behind an Elastic Load Balancer (ELB)

# To Do (Continued)

- Task3:
  - Number of servers (theoretical) = 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.

# Upcoming Deadlines

- Project 3:

| Project 3 | | |
|---|---|---|
| Horizontal Scaling in Databases | | |
| Horizontal Scaling | Checkpoint | Sunday, 23rd March |

- Unit 4:

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

# Demo Outline

- Defining a custom CloudWatch metric.
- Assigning IAM role to an instance.