15-319 / 15-619
Cloud Computing

Recitation 7

October 13th & 15th, 2015

Overview

- **Administrative issues**
  Office Hours, Piazza guidelines
- **Last week's reflection**
  Project 2.3, OLI unit 3 module 10, 11, 12, Quiz 5
- **This week's schedule**
  - Quiz 6 - October 16th (Module 13)
  - Project 3.1 - October 18th
- **Demo**
- **Twitter Analytics: The 15619 Project**

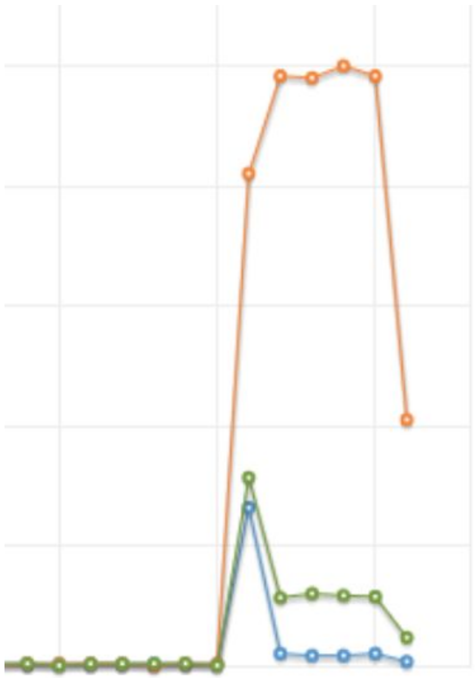
Announcements

- Monitor AWS expenses regularly
  - Check your bill (Cost Explorer > filter by tags).
- Terminate your resources when not in use
  - Stop still costs EBS money ($0.1/GB/Month)
- Use spot instances
  - And **tag them** at launch time
- Use the team AWS account and tag the 15619Project resources carefully. Otherwise, you might risk having them charged to your weekly projects.

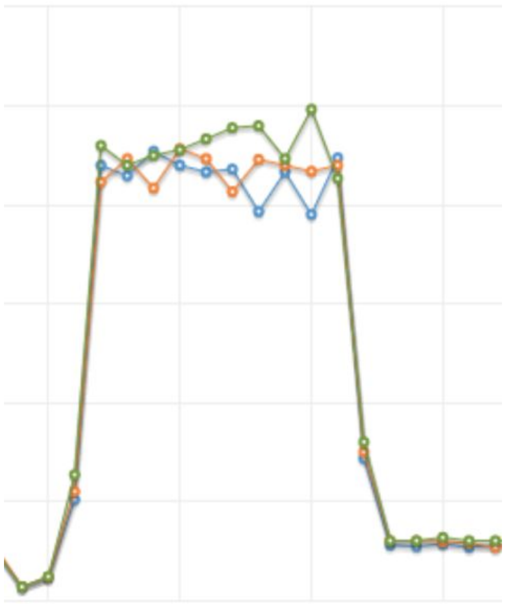
Last Week : A Reflection

- Content
  - Unit 3 - Modules 10, 11 and 12:
    Virtualizing Resources on the Cloud
  - Quiz 5 completed
- You wrote your own load balancer!
  - Round Robin
  - Custom Scheduling
  - Health check
  - Got promoted to Senior Systems Architect

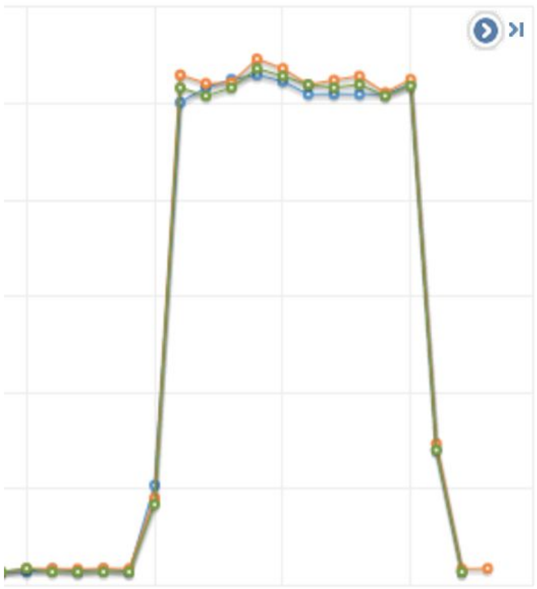
Last Week : Load Balancing



Score = 36



Score = 41



Score = 53

CPU Utilization for DCI1, DCI2 and DCI3

Project 2.3 Grading

Reminder!

- Manual Grading:
  - 20 Points are for the code, we will evaluate
    - Solution
    - Style
    - Formatting
    - Comments

Project 2 Reflection

- AWS APIs
- AutoScaling
- Trade-off between cost and performance
- Mitigating failure
- Load balancing strategies
- Multi-tiered applications

This Week: Content

UNIT 3: Virtualizing Resources for the Cloud

- Module 10: Resource virtualization (memory)
- Module 11: Resource virtualization (I/O)
- Module 12: Case Study
- **Module 13: Storage and network virtualization**
  - Software Defined Data Center (SDDC)
    - Software Defined Networking (SDN)
      - Device virtualization (Router and NIC virtualization)
      - Link virtualization (Bandwidth/datapath virtualization)
    - Software Defined Storage (SDS)
      - IOFlow
- **Quiz 6, October 16th**

Project 3 - Storage

- Storage in the cloud (It's Hot!!!)

H·BASE
amazon DynamoDB
levelDB
Amazon Redshift
Amazon RDS
VOLTDB
Cassandra
mongoDB
memsql
VERTICA
HDFS
orangeFS

Project 3 Weekly Modules

- P3.1: Files, SQL and NoSQL
- P3.2: Replication and sharding
- P3.3: Consistency
- P3.4: Social network and heterogeneous back end storage
- P3.5: Data warehousing and OLAP

This Week: Project 3.1

- P3.1: Files vs Databases
  - Data Analysis (Files, MySQL)
    - using bash scripts
    - using MySQL
      - Indexing
      - Joins
  - Vertical Scaling
    - Instance size
    - Disk type / IOPS
  - Data Analysis (HBase)

Project 3.1 Overview

- Run basic Unix commands like grep, awk etc to extract certain data from given datasets
- Use relational databases (MySQL)
- Vertical scaling in storage technologies
  - Magnetic vs SSD
  - Instance types
- Use a NoSQL database (HBase)

Flat Files

- Computer-based flat files.
  - Ex: A comma-separated ‘csv’ file.

    ```
    Mrigesh, 15619, A
    Rohit, 15319, A
    ```
- Lightweight
- Flexible
- Accessing specific data is inconvenient
- Lacking knowledge of file-layout
- …

Databases

- Organized collection of data supporting data structures

- Database management system (DBMS)
  - Interface between user and databases
  - Capture and analyze data

- Relational databases
  - Organized as fixed-length fields in tables: MySQL

- NoSQL Databases
  - Organized as Key-Value pairs:
    - DynamoDB, Cassandra, HBase

Databases

- Advantages
  - Logical and physical data independence
  - Concurrent access and transaction support

- Disadvantages
  - Cost
  - Additional expertise
  - Complex, difficult and time consuming to design

Files vs. Databases

- Compare flat files vs. MySQL

- Answer:
  - What are the advantages and disadvantages of using flat files or databases?
  - In what situation would you use a flat file or a database?
  - How to build your own databases? How to manipulate it?

MySQL Introduction

- Most popular open-source relational database
- Structured data format
- SQL - Data Manipulation Language
  - select, from, where, set operation, ordering, join

NoSQL (HBase) Introduction

- A popular NoSQL database on HDFS
- No SQL interface: Get, Scan, Put and Delete
- MySQL or HBase?

MySQL Demo

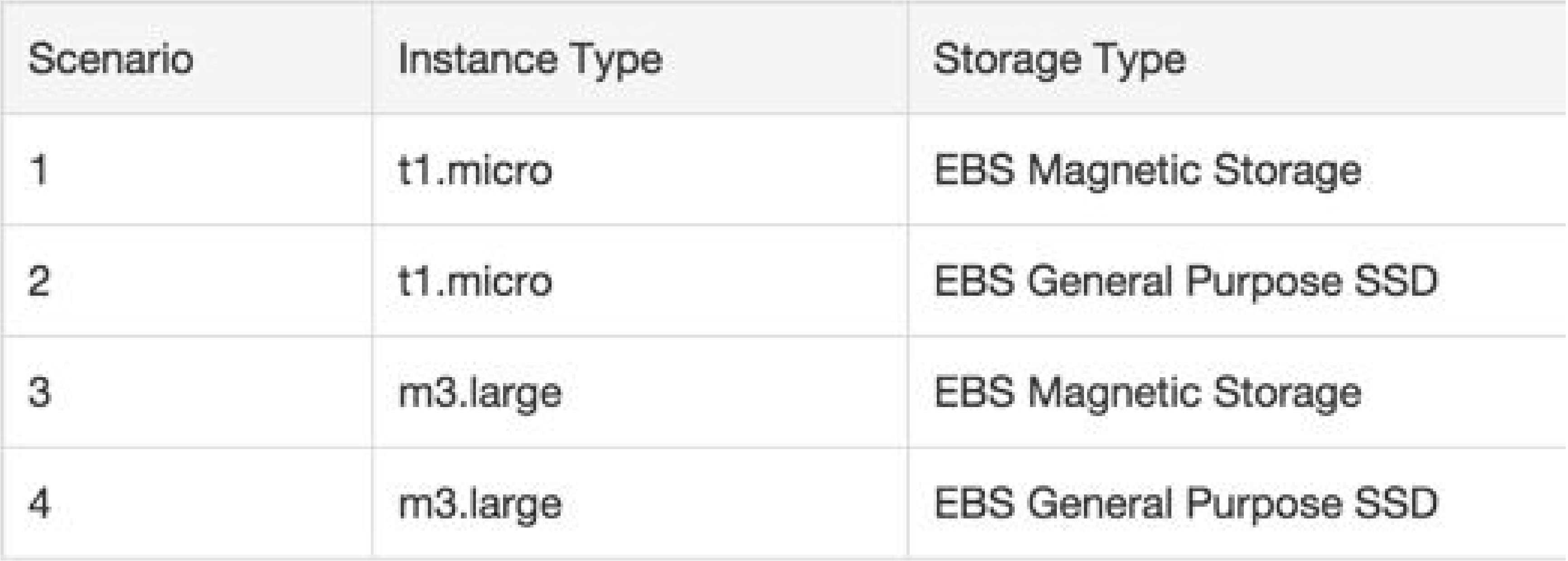
- Create a table
  - e.g. CREATE TABLE students (ID int, Name varchar(255), email varchar(255));
  - create table script is already provided for you

- Find a way to load the data properly into MySQL

- Use MySQL query to answer questions in runner.sh
  - Aggregate functions, inner join

Storage Vertical Scaling

Use the sysbench to benchmark for the following 4 scenarios:



Scenario	Instance Type	Storage Type
1	t1.micro	EBS Magnetic Storage
2	t1.micro	EBS General Purpose SSD
3	m3.large	EBS Magnetic Storage
4	m3.large	EBS General Purpose SSD

Performance Benchmarks

- Run sysbench - prepare data
  - change to mounted directory
  - use prepare option to generate the data

- Experiments
  - run sysbench with different storage systems and instance types
  - run sysbench multiple times

HBase

- Launch an EMR cluster with HBase installed.
- Follow the write-up to download and load the data into HBase.
- Use HBase querying commands in the HBase shell.

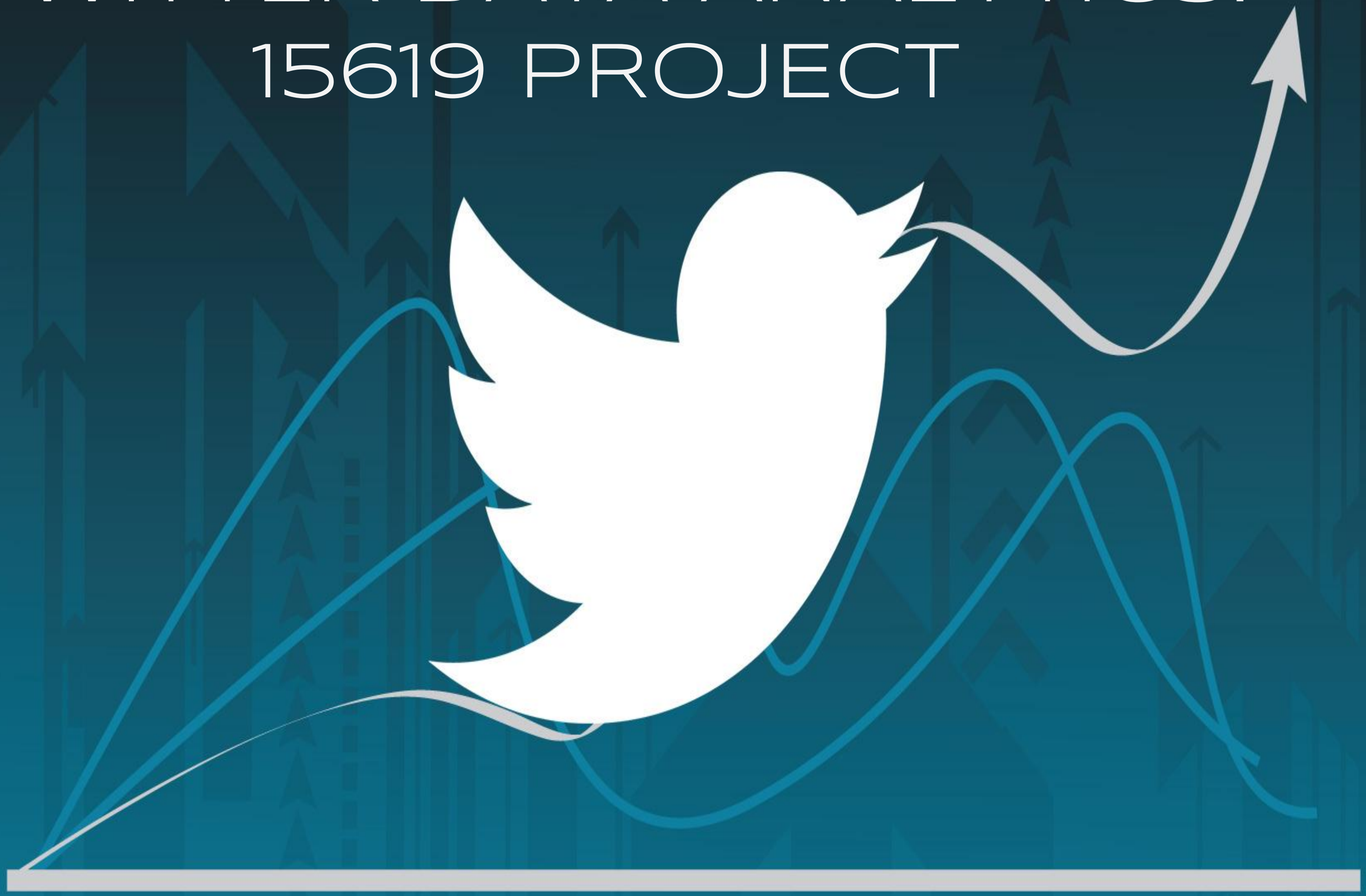
P3.1 Reminders

- Tag your resources with: Key: Project, Value: 3.1
  - manually tag your spot instances

- Be sure not to terminate the instance before answering all questions in runner.sh. Make sure to terminate the instance after answering questions in the runner.sh and submitting your answers.

- You can also save a copy of your runner.sh if you want to work on it later.

TWITTER DATA ANALYTICS:
15619 PROJECT



15619 Project Time Table

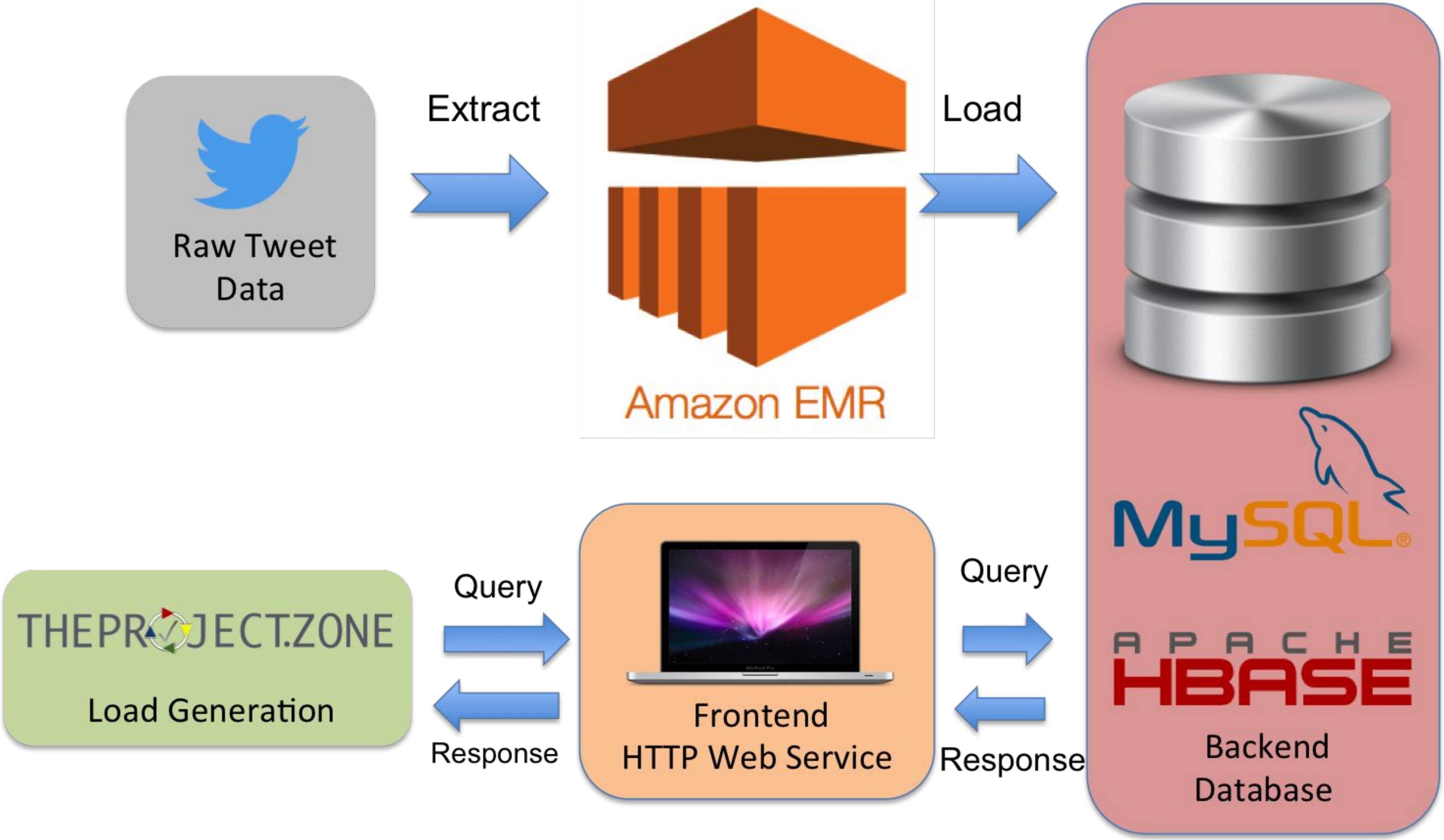
Phase (and query due)	**Start**	**Deadline**	**Code and Report Due**
Phase 1 Task 1 • Q1 **(due)**, Q2 (not yet due)	Thursday 10/15/2015 00:00:01 EDT	Wednesday 10/21/2015 23:59:59 EDT	
Phase 1 Task 2 • Q1, Q2 **(due)**	Thursday 10/22/2015 00:00:01 EDT	Wednesday 10/28/2015 23:59:59 EDT	Thursday 10/29/2015 23:59:59 EDT
Phase 2 • Q1, Q2, Q3, Q4	Thursday 10/29/2015 00:00:01 EDT	Wednesday 11/11/2015 16:59:59 E**S**T	
Phase 2 Live Test • Q1, Q2, Q3, Q4	Wednesday 11/11/2015 18:00:01 E**S**T	Wednesday 11/11/2015 23:59:59 E**S**T	Thursday 11/12/2015 23:59:59 E**S**T
Phase 3 • Q1, Q2, Q3, Q4, Q5, Q6	Thursday 11/12/2015 00:00:01 E**S**T	Wednesday 12/2/2015 18:59:59 E**S**T	
Phase 3 Live Test • Q1, Q2, Q3, Q4, Q5, Q6	Wednesday 12/2/2015 20:00:01 E**S**T	Wednesday 12/2/2015 23:59:59 E**S**T	Thursday 12/3/2015 23:59:59 E**S**T

There will also be a report due at the end of each phase, where you are expected to discuss optimizations you used to improve your performance

Busy Weeks Coming Up!

Wednesday	Thursday	Friday	Sunday
Wednesday 10/21/2015 23:59:59 EDT ● Phase 1 Task 1 (Q1 due)	Thursday 10/22/2015 23:59:59 EDT ● Quiz 7		Sunday 10/25/2015 23:59:59 EDT ● P3.2 Due
Wednesday 10/28/2015 23:59:59 EDT ● Phase 1 Task 2 (Q2 due)	Thursday 10/29/2015 23:59:59 EDT ● Phase 1 Code & Report Due	Friday 10/30/2015 23:59:59 EDT ● Quiz 8	Sunday 11/01/2015 23:59:59 E**S**T ● P3.3 Due
Wednesday 11/11/2015 18:00:01 E**S**T ● Phase 2 Live Test	Thursday 11/12/2015 23:59:59 E**S**T ● Phase 2 Code & Report Due	Friday 11/13/2015 23:59:59 E**S**T ● Quiz 10	Sunday 11/15/2015 23:59:59 E**S**T ● P3.5 Due
Wednesday 12/2/2015 20:00:01 E**S**T ● Phase 3 Live Test	Thursday 12/3/2015 23:59:59 E**S**T ● Phase 3 Code & Report Due	Friday 12/4/2015 23:59:59 E**S**T ● Quiz 12	Sunday 12/6/2015 23:59:59 E**S**T ● P4.2 Due

15619 Project System Architecture



- Web server architectures
- Dealing with Tweet Replications
- HBase and MySQL optimization

15619 Project Phase 1?

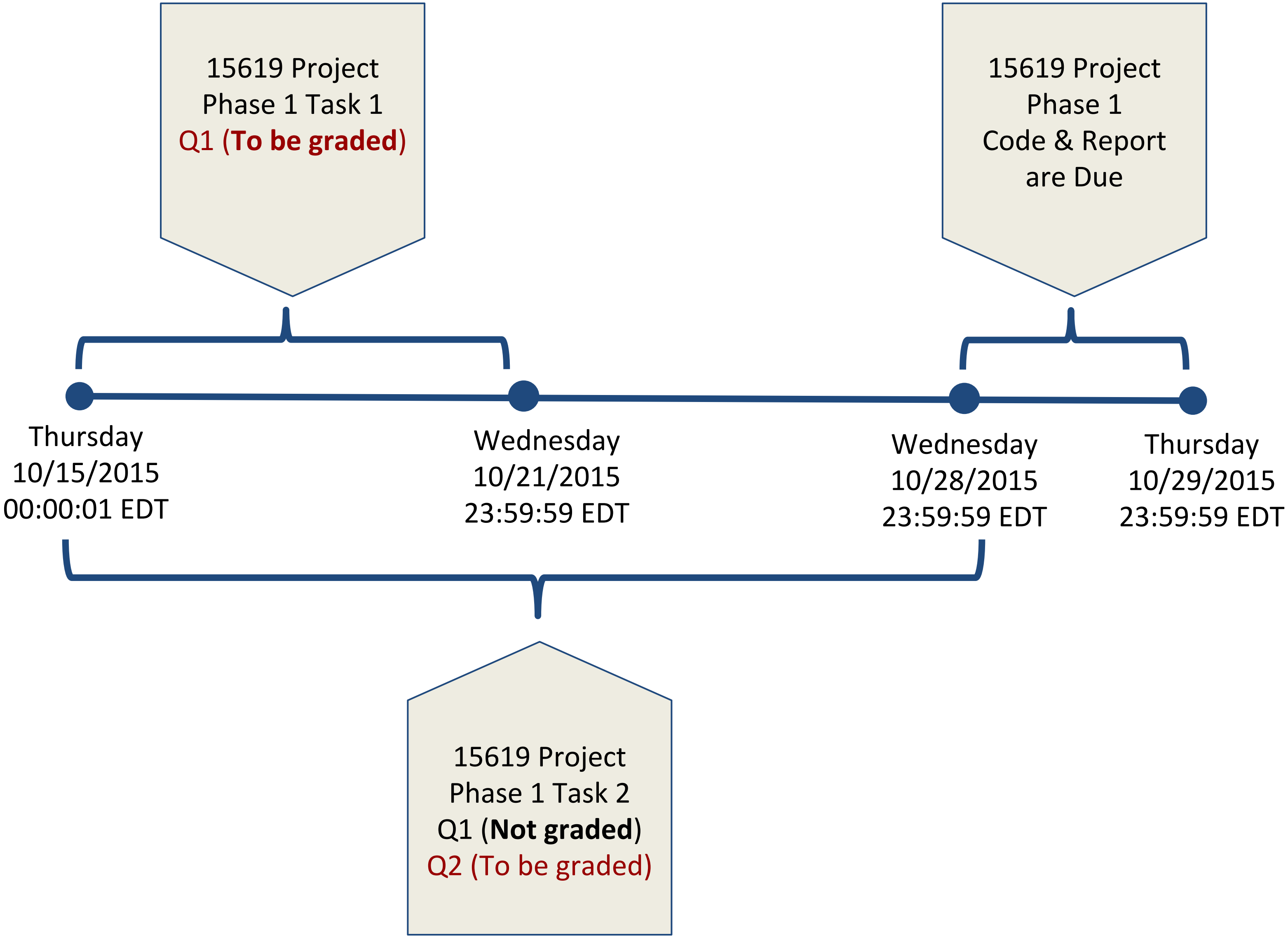
- **Step 1:** Extract tabular data from raw tweets
  - Input file: JSON Tweets (approx. 1 TB)
  - Consider using a MapReduce Job for ETL
    - ETL is expensive and there’s the potential for errors, so plan carefully, test on smaller data sets
- **Step 2:** Load the data into HBase **<u>and</u>** MySQL (both!)
- **Step 3:** Design and deploy
  - a web service for handling HTTP requests responds with data from the backend
  - an optimized backend (MySQL and HBase)

Higher throughput = More points

Winner gets grades, fame (?), job (?)

15619 Project Phase 1 Deadlines

- 1 week for Q1
- 2 weeks for Q2



15619 Project Phase 1 Task 1 Q1 (**To be graded**)

15619 Project Phase 1 Code & Report are Due

Thursday
10/15/2015
00:00:01 EDT

Wednesday
10/21/2015
23:59:59 EDT

Wednesday
10/28/2015
23:59:59 EDT

Thursday
10/29/2015
23:59:59 EDT

15619 Project Phase 1 Task 2
Q1 (**Not graded**)
Q2 (To be graded)

Upcoming Deadlines

- Quiz 6: Unit 3 - Storage and network virtualization

  Due: 10/16/2015 11:59PM Pittsburgh

- Project 3.1: Files vs Databases

  Due: 10/18/2015 11:59PM Pittsburgh

- Project 15619: Phase 1, Task 1

  Due: 10/21/2015 11:59PM Pittsburgh