15-319 / 15-619 Cloud Computing

Recitation 2

September 5 & 7, 2017

Accessing the Course

- Open Learning Initiative (OLI) Course
  - Access via canvas.cmu.edu
- http://theproject.zone
  - AWS Account Setup
  - Azure Account Setup
  - GCP Account Setup
  - Update your TPZ profile with AWS, Azure & GCP info
  - Complete the Primers on AWS, Azure and GCP
- Piazza

Amazon Web Services (AWS) Account

- === ONLY IF YOU HAVEN'T DONE SO ALREADY ===
- Log on to https://theproject.zone and make sure you follow the instructions in the Account Setup Primer
- Wait to receive Consolidated Billing Request email from Amazon
  - Manual process, waiting time varies
- When you receive the linking email, click the link to verify the linked billing
  - Many students have not clicked on the link yet!
    - Check your **SPAM** folder
  - You won't be able to complete the projects.

Azure Account

- === ONLY IF YOU HAVEN'T DONE SO ALREADY ===
- Do **not** use your @andrew.cmu.edu or other CMU issued email address.
- Update your TPZ Profile

Google Cloud Platform (GCP) Account

- === ONLY IF YOU HAVEN'T DONE SO ALREADY ===
- Please contact us if you have trouble creating your GCP account.
- Follow the instructions in the primer.
- Receive a $50 coupon on https://theproject.zone
- Redeem the coupon as per instructions on https://theproject.zone
- If you cannot view your GCP coupon in your TPZ profile (https://theproject.zone/profile) post on Piazza privately and share your Andrew ID so we can make that available for you.

Piazza

- Suggestions for using Piazza
  - Discussion forum, contribute questions and answers
  - Read the Piazza Post Guidelines (@7) before asking
- When you have a (project-specific) problem, follow this order!
  - Try to solve the problem by yourself (Search, Stack Overflow)
  - Read Piazza questions & answers carefully to avoid duplicates
    - Visit TA OHs: TA office hours are posted on Piazza and Google calendar
    - Create a piazza post
- Please note:
  - **Show the effort you have done first**
  - **Give us context** and as much information as possible!
  - Don't ask a public question about a quiz question
  - Try to ask a public question if possible
  - Provide your andrewID privately if you think we need it to help

Reflecting on Last Week

- AWS
  - Launch, connect to and terminate EC2 instances
  - Install and run software on an EC2 instance
  - Use spot instances, use S3 to store and retrieve files
  - Vertical scaling
- Azure and GCP
  - Launch, connect to and terminate VMs
  - Install & run software on a VM
  - Vertical scaling
- In P0, run a web server, test to access the server over a browser
  - Set up the web server software.
  - Open up the required ports.
- Use AWS, Azure and GCP APIs to launch instances
- Basic Linux/SSH skills

Skill Building in This Course

- Important skill to develop
  - willingness and courage to
    - recognize, explore and solve problems **on your own**
    - learn the basics of new tools quickly and make use of them in a limited time (e.g. 1 week)

Make Sure to Complete the Primers!

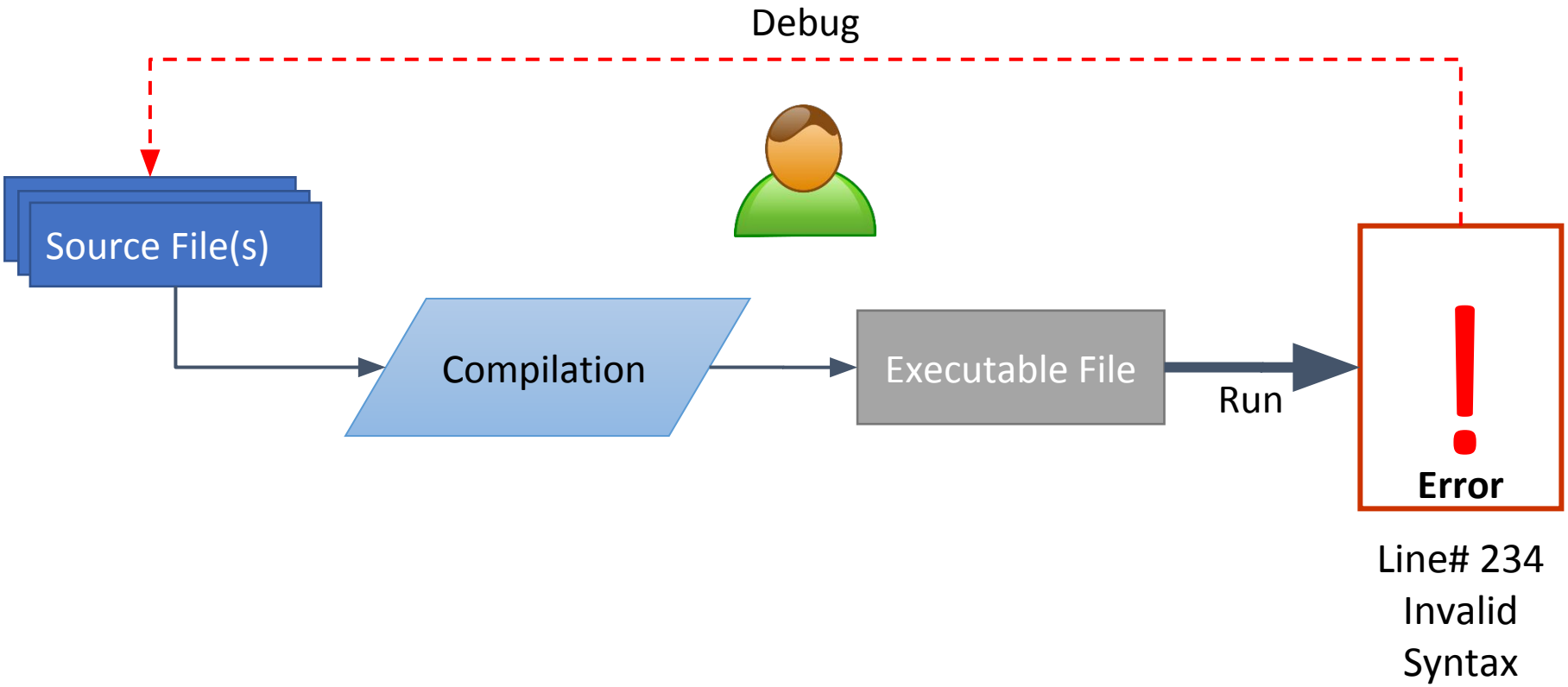
- Complete the Primers
  - Understanding AWS/Azure/GCP
    - provisioning resources, connecting to VMs, playing around, …
  - Practice working in the Linux shell

Programming Experience Expected

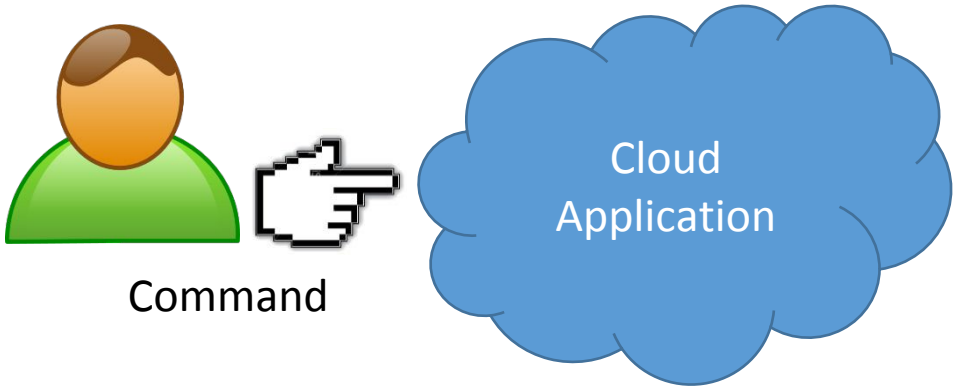
- **Strong proficiency** in at least one of the following, with some fair comprehension of the others:
  - **Java 8**
  - **Python 2/3**
  - **Bash**
-  **Java is required** to complete parts of Projects 2, 3 and 4.
- Use the time now to brush up
- Please read Maven primer!
- Do not fear bash/python scripting, it will make your life easier!

Typical Programming Workflow:

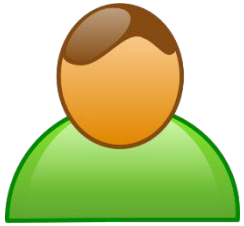
- For most courses:



In the Cloud



In the Cloud



In the Cloud

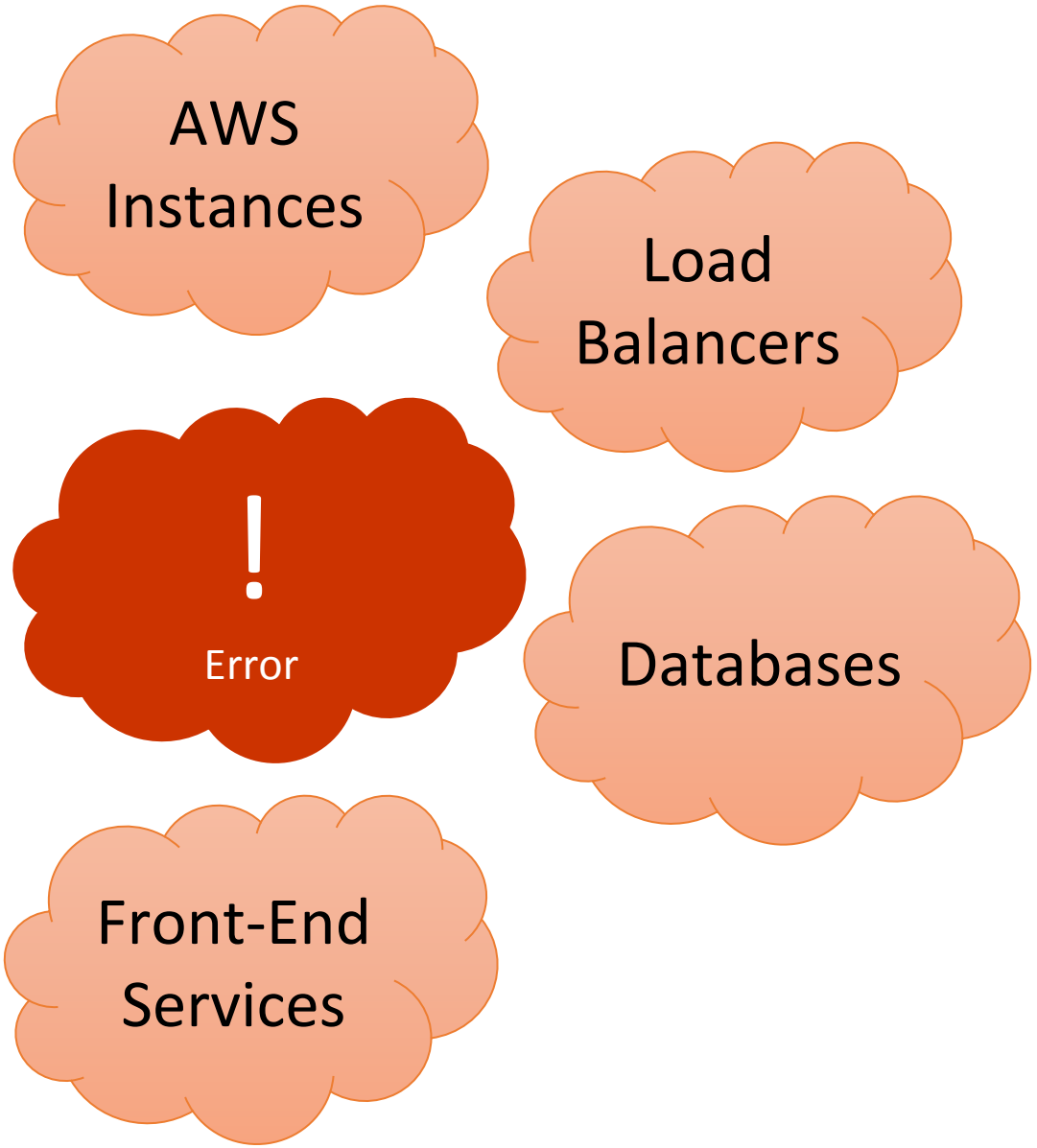
Y U NO WORK?!?

How do I even
begin to fix this?

In the Cloud

Y U NO WORK?!?

How do I even begin to fix this?



Suggested Error Debugging Workflow

What information can I get about the error?

- Read error messages, Look through logs, other information

How can I isolate the source of the problem?

- What component seems to have the problem?

What remedial action can I take?

- The error messages and other information should have clues.
- Configuration changes, command parameters

Am I Still Stuck?

- Search/StackOverflow, and then TA Office Hours/ Piazza (In this order!)

Completing Projects in this Course

- Provision AWS, Azure or GCP Resources
  - Use the AMIs/VHDs/OS Images we provide for the project
  - Tag all instances!
- Monitor your cost
  - Calculate costs before you provision!
- Complete tasks for each project module
  - Each project module has several sections unlocked by AssessMe
- Submit your work
  - Pledge of integrity
  - Results in scoreboard
- Terminate all resources when you have verified your score and kept a copy of your work (e.g. git **private** repo)

Tagging

- Tag *all* tag-able resources on AWS
  - Before you make a resource request, read the docs/specifications to find out if tagging is supported
  - We will specify which resources are required to tag in each project
  - Apply tags during resource provisioning
  - We need tags to track usage, a grade penalty will be applied automatically if you do not tag!
  - Spot instances on AWS do **not** get tagged automatically
- Tagging Format
  - `Key: Project`
  - Value: 0, 1.1, 1.2….etc.
  - Information will always be present in the project instructions

Budgets and Penalties

- No proper tags ➔ 10% grade penalty
- Budget
  - For P1.1, each student's budget is **$1**
  - Exceeding Budget ➔ 10% project penalty
  - Exceeding Budget x 2 ➔ 100% project penalty (no score)
  - You can see Cost and Penalties in TPZ.
- No exceptions.

- We will enforce these penalties automatically starting from Project 1.1

How to Work on a Budget

- P1.1 Budget → $1
- You are only allowed to use t2.micro
  - $0.012 per hour (on demand)
- Other costs to consider:
  - EBS is $0.1 per GB/month
  - Instances using our AMI gets 30 GB EBS by default.
  - Data transfer costs (minimal)

Total time: $1 / ($0.012 + 30 * $0.1 / 30 / 24) = 62 hours

- **Note**: Free Tier does not apply to linked accounts!

Stop the instance

- Good strategy to stop your instance to save the budget and restart it to continue your work.
- **Stopping** an instance is different from **terminating** it. Remember to terminate all the resources!
- For a stopped instance, you will not get charged by EC2 hourly usage or data transfer fees, but you will still pay for the storage for EBS volumes.
- When you restart a stopped instance, note that the **DNS name will most likely change**. You need to update your commands or configuration when you use SSH or SFTP.

Academic Integrity Violation

- Cheating ➔ the lowest penalty is a 200% penalty & potential dismissal
  - Other students, previous students, Internet (e.g. Stackoverflow)
  - Do not work on code together
  - This is about you struggling with something and learning
  - Penalty for cheating is SEVERE – don't do it!
  - Ask us if you are unsure

Academic Integrity Violation (cont.)

- You **can** read the code and learn ideas from Stack Overflow for general problems with citation.
- You **CANNOT** **copy** code from Stack Overflow.
- You **can** read the code from official documentation of languages or libraries with citation.
- You **can** learn ideas from Github, Bitbucket, Blog, etc. by search engines for general problems with citation.
- You **CANNOT** read the solutions from previous students on Github, Bitbucket, Blog, etc.
- You **CANNOT** look at the code of other students.

Compromised Accounts

- If you put any of your credentials in files on
  - Github, Dropbox, Google Drive, Box, etc.
  - You are vulnerable to getting your account compromised.
  - Going over 2x the project budget ⇒ 100% penalty!
- People are scanning publicly available files for cloud credentials.
  - They compromise your account and launch resources in other regions.

DO NOT SAVE YOUR CLOUD CREDENTIALS IN FILES!

Deadlines!

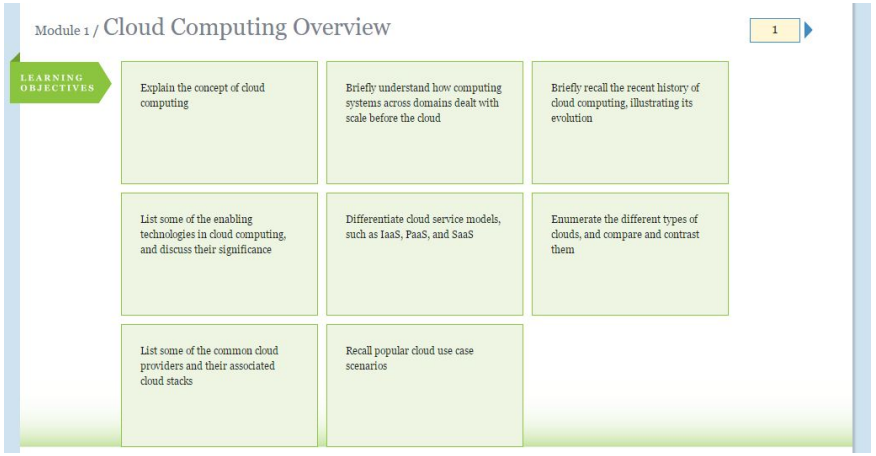
- **Hard** Deadlines
  - No late days, no extensions
  - Start early!
  - Plan your activities, interviews and other commitments around the deadlines.
  - **No exceptions!**
- Project modules are due on Sundays at 23:59 ET
- Quizzes are typically due on Fridays
  - There are two exceptions this semester

Deadlines!

- **Project deadlines**
  - **On TheProject.Zone**
- **Quiz deadlines**
  - **On OLI**

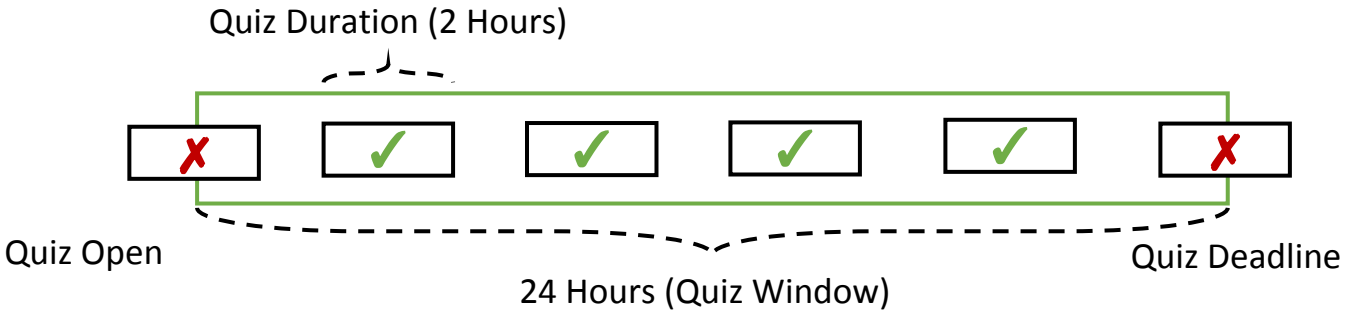
Quiz 1 Preparation

- Tests your understanding in Modules 1 and 2
  - Cloud computing fundamentals, service models, economics, SLAs, security
  - Use the activities in each page for practice.
  - You will be tested on you ability to perform the stated learning objectives on OLI:



Quiz 1 Logistics

- Quiz 1 will be open for 24 hours, Friday, Sep 8
  - Quiz 1 becomes available on **Sep 8, 00:01 AM ET.**
  - Deadline for submission is **Sep 8, 11:59 PM ET.**
  - Once open, you have **120 min** to complete the quiz.
  - You may not start the quiz after the deadline has passed.
  - Every 15 minutes you will be prompted to save.
  - **Maintain your own timer from when you start the quiz.**
  - **Click submit before deadline passes. No Exceptions!**



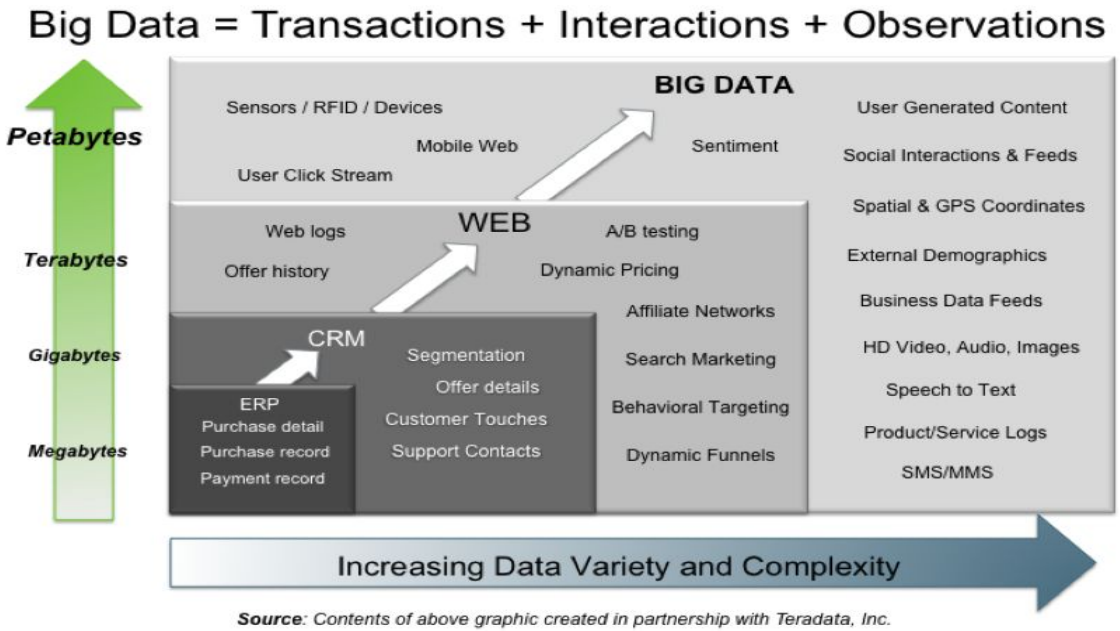
Submit Before Deadline

- When you start the Quiz, you cannot stop the clock.
  - You have 120 minutes to click on submit.
  - You have to keep track of the time yourself.
  - If you don’t click on submit you will not receive a grade.

YOU MUST SUBMIT
WITHIN 120 MINUTES
AND
BEFORE THE DEADLINE

Project 1 Motivation: Big Data

- What is Big Data?
  - It is high volume, high velocity, and/or high variety information assets.
  - There is a lot of value in the analysis of big data for organizations



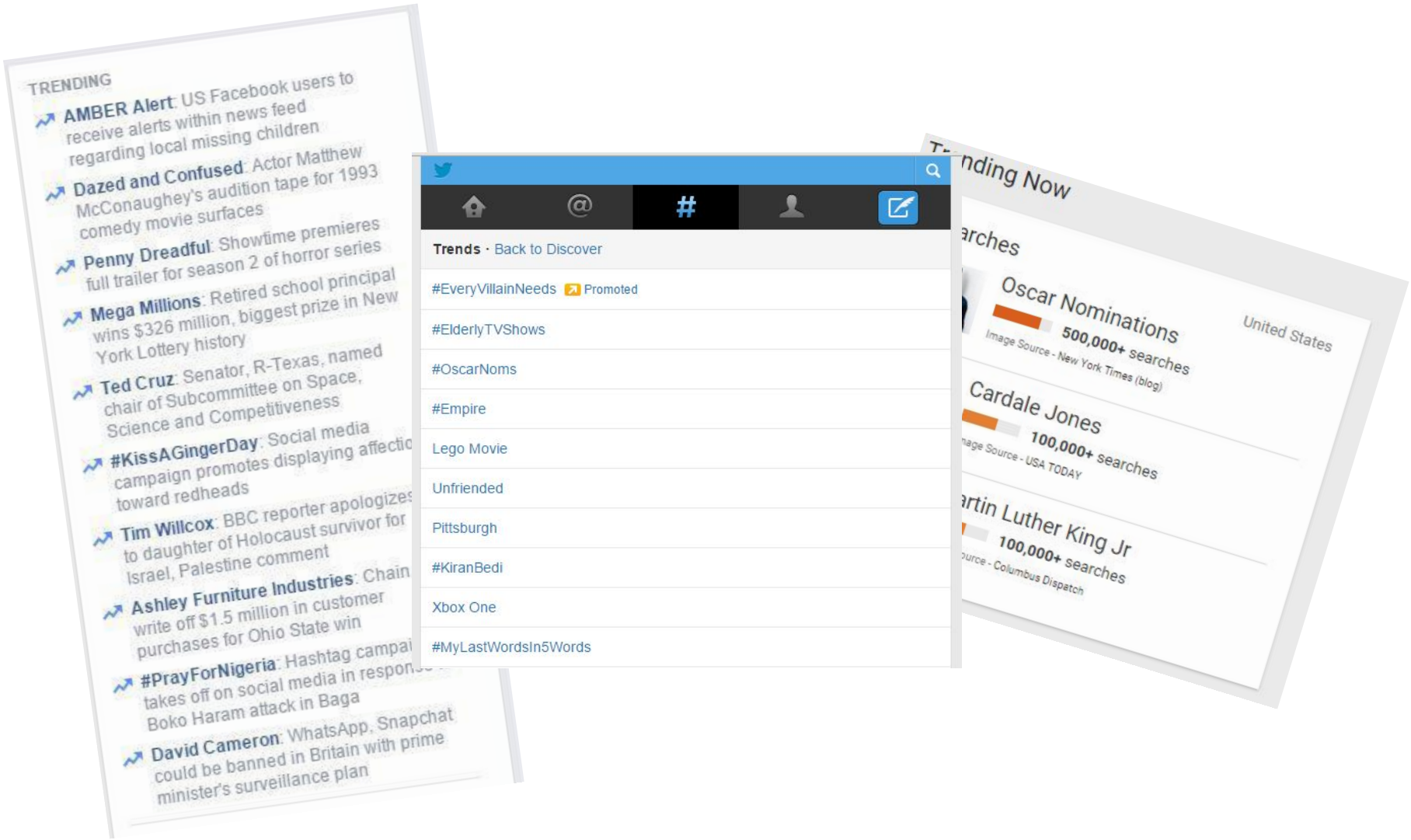
***Source**: Contents of above graphic created in partnership with Teradata, Inc.*

Use Cases: Big Data Analysis

- Online retailers are analyzing consumer spending habits to learn trends and offer personalized recommendations and offers to individual customers.

- Companies such as Time Warner, Comcast etc. are using big data to track media consumption habits of their subscribers and trends to provide value-added information to advertisers and customers.

Trending Topics are Everywhere!



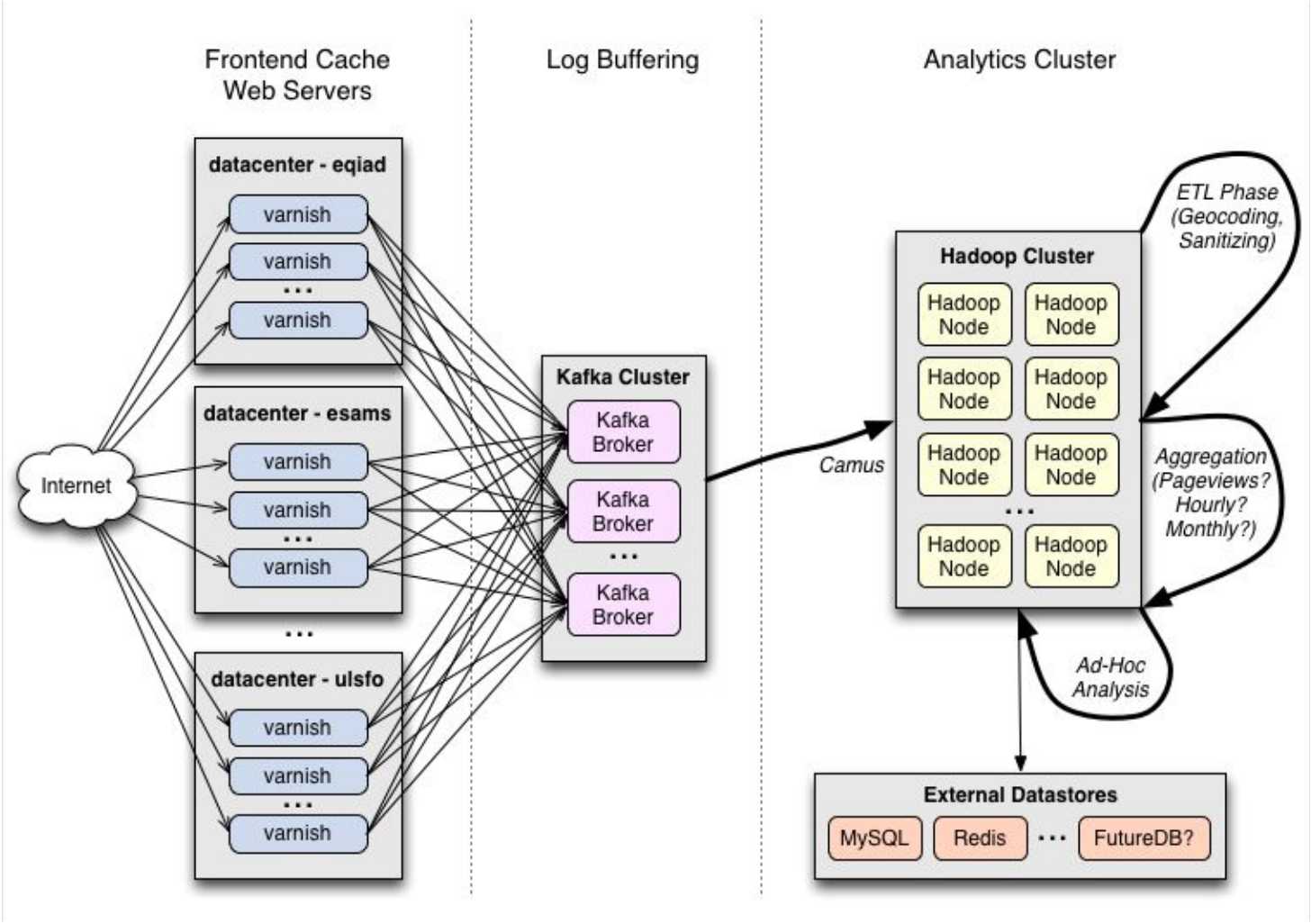
Why Trending Topics?

- Identify trends and viral content
- Maximize advertisement placement opportunities
- Search Engine Optimization (SEO)
- And more....

Project 1

- Identify Trending Topics on Wikipedia
  - Use the hourly pageviews dataset.
- Project 1.1: (This Week)
  - Find trends from a single hour of data.
- Project 1.2: (Next Week)
  - Find trends for an entire month.
    - Data from November 2016

Wikipedia Page Requests



The Dataset

- Data set
  - Wikimedia page views dataset
  - One File Per Hour
- Format:

```
<domain code> <page title> <number of accesses> <total data returned>
```

<Language>.<ProjectName>
en = English Wikipedia (Desktop)
en.b = English Wikibooks
fr.v = French Wikiversity

Data Pre-processing is Important

- Impossible: Raw Dataset →Data analysis
- Raw Dataset →Data pre-processing →Data analysis

raw data:

after data pre-processing

reference: Nishant Neeraj

Parse and Filter

- We are only interested in English Wikipedia desktop/mobile pages
- This dataset is raw, real-world
  - Never assume that the dataset is perfectly clean and well formed.
- Filter out the rest
  - Use the filtering rules specified in the writeup
- Sort the pages by number of pageviews, break ties by ascending lexicographical order
- Output: `<page title> <number of accesses>`

Develop Robust Code that Can Execute Correctly on Multiple Environments

- "My submitted code does not produce the same results as the one on my EC2 instance..."
- If your code behaves well in your development environment, it does **not** guarantee that your code will work perfectly in other environments.
- If you run into this, read the writeup carefully, check and adopt best practices before you create posts on Piazza:
- Be cautious about implicit reliance on your environment
  - Locale
  - Encoding-aware I/O
  - Newline(EOL)
  - Versions & Compatibility
  - Absolute/Relative Paths

Project 1.1 Workflow

- Launch EC2 instance with a special AMI
  - We recommend using the APIs to launch instances so you practice before Project 2.
- Download the required dataset
- Write the code to parse, filter and sort
- Complete and run the script
  - `/home/<andrew_id>/Project1_1/runner.sh`
  - Answer a set of questions by providing the commands/code inside runner.sh
- Submit your code for grading
  - Complete the `references` file in JSON format
  - Execute `submitter` to submit your code
- Finish Project Reflection (graded) before the deadline
- Finish Project Reflection Feedback for 3 students
  - Within 7 days **after** the project deadline

Grading of Your Projects

- Code submissions are auto-graded
- Scores will be made available on http://theproject.zone
  - it may take several minutes for your score to show
  - the submissions table is updated with every submission
- We will grade all the code (both auto and manually)
- **Hard to read code** of poor quality will lead to a loss of points during manual grading.
- Lack of **comments**, especially in complicated code, will lead to a loss of points during manual grading.
- Poor **indentation** will lead to a loss of points during manual grading
  - Preface each function with a header that describes what it does
    - Use descriptive variable and function names
    - Use Checkstyle, PEP8, or other tools to check your coding style
- The idea is also NOT to comment every line of code

Reminder: Deadlines

- This Friday at 23:59 ET
  - Quiz 1
- This Sunday at 23:59 ET
  - Project 1.1 (including Project Reflection)
- Next Sunday at 23:59 ET
  - Project 1.1 Reflection Feedback
- ASAP, at the latest 9/11/2017 at 23:59 ET
  - Academic Integrity Course Quiz