University of California, Berkeley
Master of Information and Data Science (MIDS)
W205 – Fundamentals of Data Engineering

Week 5 – Pipelines and Clusters of Containers

Agenda for Today's Class

- Attendance and Participation
- Announcements
- Schedule and Due Dates
- Work / Life / School Balance
- Asynch High Level Review in a Nutshell
- Breakouts
- Summary

Attendance and Participation

Please record your attendance and participation for today's class:

GitHub => ucb_mids_w205_repo => README.md => Attendance and Participation

Announcements

- Upcoming holidays and/or breaks
- Makeup classes for holidays
- Upcoming events
- Student evaluations
- Etc.

Schedule and Due Dates

Take a quick look at the next couple of weeks' due dates:

GitHub => ucb_mids_w205_repo => README.md => Schedule and Due Dates

Work / Life / School Balance
Open Discussion

Student feedback

- About 5 minutes
- How are things going related to work / life / school balance?
- How is w205 going? Difficulty? Time?
- Impact of any natural and/or man-made disasters
- Etc.

Asynch High Level Review in a Nutshell

Each week we will spend about 15 minutes reviewing the most important high level concepts from the asynch

Factory Assembly Line

- Manufacturing processes
- Parts from one or more manufacturing processes go into the next manufacturing process
- Some of the manufacturing processes can be done at the same time, that is, in parallel
- End product is a finished good

Pipelines

- Aka data pipelines
- Analogous to a factory assembly line
  - Processes
  - Output from one or more processes goes into one or more processes
  - Some of the processes can be done in parallel
  - End product is data that is ready to go for analytics

Pipeline Goals

- Fully automated—no manual steps
- Handle all data possibilities gracefully
- No crashes—automatically recover from all errors and continue
- Data should be cleansed and validated and ready to go for analytics
- Efficient
  - Data used for analytics often goes stale quickly
  - Need to make data available for analytics as soon as possible
  - Run as much in parallel as we can

Building Pipelines

- Comprehensive list of all input data
  - Encoding, formats, where from, how often, etc.
- Design and build processes
  - Acquire the data
  - Staging load of data
  - Cleanse, validate, transform data
  - Combine with other datasets, especially secondary datasets
  - At the end, data is ready for analytics
    - Not always a load

Clusters of Containers

- Why do we need them?
  - If we put everything in a single container, we have to load and configure all vendor software in that container
  - Eliminate conflicts
    - Example: one vendor needs this value for a kernel parameter and another vendors needs this value
  - Cannot scale up a single container
  - Leverage vendor container images

Clusters of Containers

- Specify:
  - Images to use
  - How many containers from an image (scale up)
  - Dependency order (A needs to be running before we start B)
  - Storage mounts
  - Networking (hostnames, ports, connections, etc.)
- One command convenience
  - One simple command to startup cluster
  - One simple command to shutdown cluster

Container Orchestration

- Think in terms of containers rather than in terms of VMs
  - Specify containers: images, CPU, memory, etc.
  - Let the container orchestration figure out the VMs
- Scale Up
  - Start with a minimum number of containers
  - Add more containers as demand increases
  - Remove containers as demand decreases
- Load Balance
  - Distribute workload evenly among containers

Breakouts

GitHub => ucb_mids_w205_repo => breakouts

(time permitting, we may not get to all of them)

Summary

Instructor will give a brief (about 2 minute) summary of today’s class.