Cloud Computing

ECPE 276

Project 1

MapReduce is Dead?

Research Timeline

- 2004 – Google "MapReduce" paper
  - Batch processing … parallelized
  - Challenges
    - Real-time processing
    - Long sequences of data transformation (i.e. "pipelines" of multiple steps)
- 2010 – Google "FlumeJava" paper
  - Data pipeline … parallelized
- 2013 – Google "MillWheel" paper
  - Streaming data engine

Commercial (Cloud) Products

- Google *FlumeJava* (API) and *MillWheel* (engine) became Google ***Dataflow*** (mid-2014 beta, spring 2015 launch)
  - Combines batch and stream processing into one product
  - Shared API to simplify work for programmers
- Competitors
  - Amazon ***Kinesis***
  - Microsoft ***Azure Stream Analytics***
- Proprietary / closed-source
  - Google has open-sourced the dataflow API layer, not its implementation….

Open Source Products

- Apache Spark
  - http://spark.apache.org/
  - Data processing engine
  - Runs on top of Apache Hadoop (MapReduce) *or* independently

- Apache Crunch
  - http://crunch.apache.org/
  - API for data pipelines
  - Runs on top of Apache Hadoop (MapReduce) and Spark

(Although many of the MR jobs are triggered by higher-level abstractions, not by programmers writing MR jobs directly…)

Common Crawl Project

CommonCrawl

Common Crawl

- http://www.commoncrawl.org/
- Free crawl of the web
  - Downloaded web pages and documents
- Dataset:
  - **1,417+ TB** *in 2015 alone!*
  - **16+ billion URLs** (documents) *in 2015 alone*
  - Stored in Amazon S3

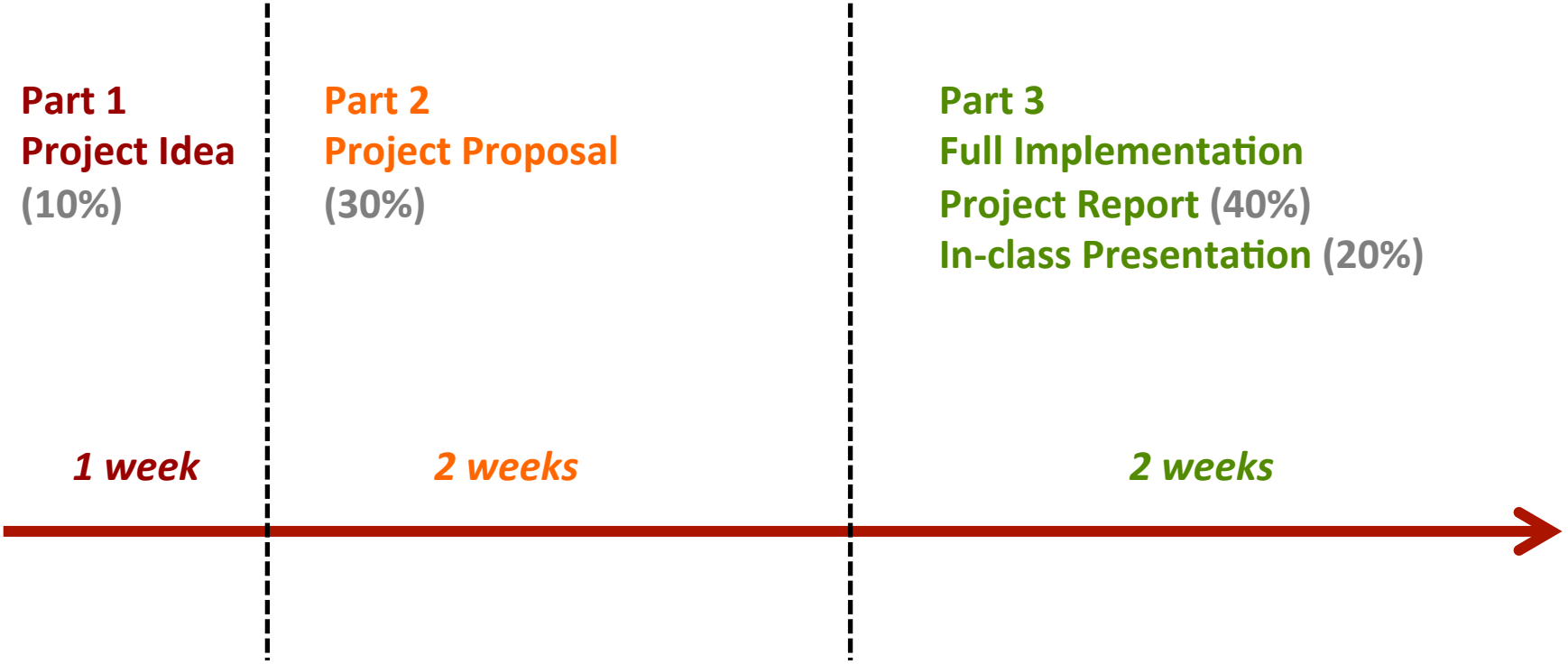
Project Specifications

- Group size: 1 or 2 people
- Data source: CommonCrawl
- Data processing method: MapReduce
  - Presumably Java, but it is (technically) possible to use other languages…
- Computation resources: Amazon Elastic MapReduce
- Project objective: Up to you!
  - Must answer a *specific question* about the dataset

Project Objective

- A few ideas I thought of
  - What are the top 100 keywords used in a website title? (Or link, description, etc..)
  - What percentage of pages uses AJAX, dynamic HTML, or <insert new web tech trend here>?
  - What languages are present in the crawl? (across all HTML pages, only in PDF documents, etc..)
  - What percentage of documents are labeled with the incorrect content type? (Web servers return a header field ("Content-Type") specifying what type of document is being provided, such as text/html, application/pdf, etc. But, this information could be wrong.)
  - What pages/documents are duplicated the most times? (i.e. at least 90% of the content on page A appears on 1500 other pages in the crawl)
  - Of the items in the crawl, are they mostly large files or small files? (i.e. a histogram of the document sizes)
  - What are the most common viruses / spyware / malware that were captured in the crawl? (Note that the crawl is deliberately unfiltered for this!)

Timeline


Part 1
Project Idea
(10%)
Part 2
Project Proposal
(30%)
Part 3
Full Implementation
Project Report (40%)
In-class Presentation (20%)
1 week
2 weeks
2 weeks

Part 1 – Project Idea

- 1 page PDF
- Contents
  - **What is your idea?**
  - Project timeline (Gantt chart)
  - Division of labor (if 2 person group)

Part 2 – Project Proposal

- **Writing a good proposal will require you to do some preliminary implementation first!**
  - Hence, 2 weeks in the schedule…
- **Introduction**
  - What question are you answering about the data?
- **Algorithm Details**
  - How are you going to find your answer?
  - What open-source tools do you intend to use to accelerate project development? (This is encouraged!)

Part 2 – Project Proposal

- **Infrastructure**
  - How much of the CommonCrawl dataset do you intend to process?
  - How many EC2 nodes will be needed to process it in parallel?
  - How many hours do you estimate it take to run the analysis to completion?
  - How much $$$ will this project cost to execute trial runs on a small data subset and do a final "production" run?
- **Analysis**
  - After running your final project on the data set, what results will you produce and how will they be presented?

Part 3 – Project Implementation

- Finish doing all the work you proposed
- Project report
  - Some sections come straight from the proposal (with minor editing)
  - New section: Analysis & Results
- Short in-class presentation on results – 8 minutes

Billing Notes

- You're charged by the hour per compute node
  - Even if you only use a minute before terminating!
- Don't be greedy and fire off 100 parallel nodes, just to make your job only take 5 minutes total
  - Note that the CommonCrawl dataset is located in Amazon's US Standard region. You should run your analysis in the same region (US) in order to avoid data transfer charges.
- Discussion on spot instances