MediaCrossing™
BRIDGING THE MARKET™

Going Live: Preparing your first Spark production deployment

Gary Malouf
Architect, MediaCrossing
@GaryMalouf

Spark Summit 2014

Overview

- Spark at MediaCrossing
- Choosing, maintaining and possibly compiling the right combination of packages to work with Spark
- Data serialization/deserialization formats
- Performance issues with small data
- Configuration/Deployment automation
- Monitoring
- Exposing Spark for non-developer usage

Spark at MediaCrossing

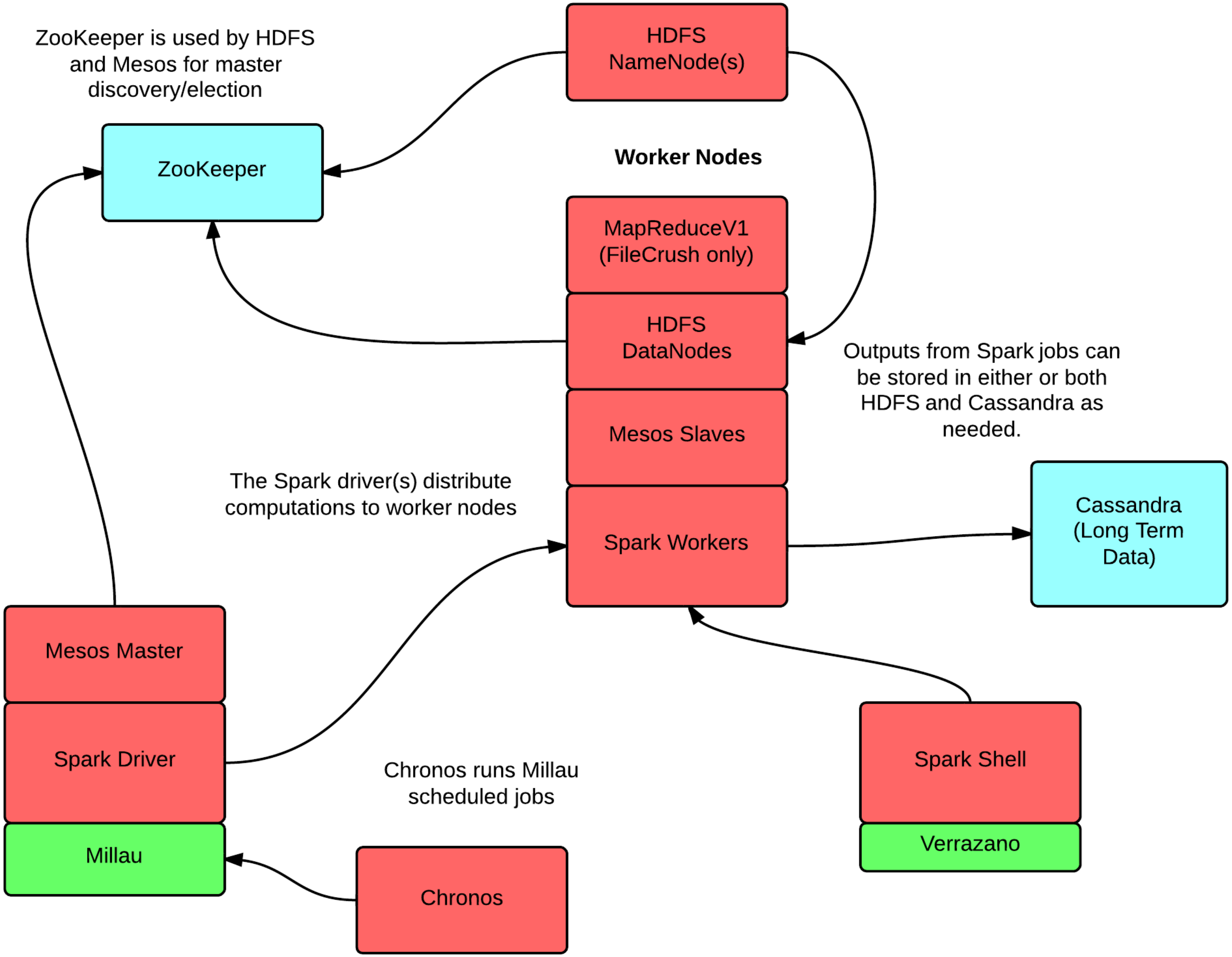
- We make it easier and more efficient for advertisers and publishers to trade digital media in milliseconds
- Since we started trading in 2013, we were able to avoid ever having to use Map/Reduce for our big data thanks to Spark
- Inspired by Nathan Marz's "Lambda Architecture" principles, our team leverages a unified view of realtime and historic user behavior to constantly adjust our buying and selling models

Target Audience

- You are building a new data/analytics system and have determined that you have big data (> 5TB)
- ‘New’ in this case means no prior Hadoop/HDFS installation
- This is your first rodeo with an analytics system based on Spark

Building the Stack

- Where will data be stored?
  - HDFS/Cassandra are two major options
  - If not already published, may need to compile Spark against your HDFS distribution version
- How to choose a cluster manager?
  - Standalone/Yarn/Mesos
  - We went with Mesos – Berkeley stack preference
- How will data get persisted into your system?
  - Continuous streaming data - Apache Flume, Spark Streaming
  - Large batches – Spark jobs!
- Reliable Job Scheduling?
  - Chronos (runs on Mesos) – fault-tolerant job scheduler



ZooKeeper is used by HDFS and Mesos for master discovery/election

ZooKeeper

HDFS NameNode(s)

Worker Nodes

MapReduceV1 (FileCrush only)

HDFS DataNodes

Mesos Slaves

Spark Workers

Outputs from Spark jobs can be stored in either or both HDFS and Cassandra as needed.

Cassandra (Long Term Data)

The Spark driver(s) distribute computations to worker nodes

Mesos Master

Spark Driver

Millau

Chronos runs Millau scheduled jobs

Chronos

Spark Shell

Verrazano

Storing your data

- Text Files
  - –Human Readable
  - –Not splittable when compressed out of box (negatively affects parallelization)
- Sequence Files (container for binary key-value pairs)
  - –Not human readable
  - –Great for storing key-value pairs without parsing
  - –Splittable (helps parallelize jobs)
  - –Storage efficient
  - –Protobuf friendly

Small File Woes

- HDFS Default Block Size – 64MB
- Data partitioning in Spark is done based on the number of blocks each source file takes up.
- Spark/HDFS performs orders of magnitude better with 100’s of files on the order of 64/128 MB in size vs 1000’s/10’s of 1000’s of much smaller files.
- Options:
  - –Batch your data into large files at write time
  - –Write your own job to aggregate small files into large ones
  - –Use an existing library like https://github.com/edwardcapriolo/filecrush

Configuration/Deployment

- Automate from day 1 – Ansible, Puppet, Chef, etc
- Like you would with source code, version control your configuration and deployment scripts
- Co-locate Spark workers (Mesos slaves) with HDFS data nodes – noticeably better performance
- Ideal World: Separate research cluster from ‘production’

Monitoring

- Spark covers this extensively in their docs: https://spark.apache.org/docs/latest/monitoring.html
- In practice, we rely on Nagios to tell us when a server is struggling and Munin for diagnosing less urgent, long running problems
- Running on Mesos, we rely on their user interface for visually seeing what work is taking place across the cluster, and when we need to increase our cluster size.

Non-developer Tooling

- Built-in PySpark for interactive querying
- SparkR - http://amplab-extras.github.io/SparkR-pkg/
- Shark (Hive on top of Spark) – SQL query language

Thank you for your time!

- We continue to increase our usage of Spark for research and other critical business functions
- If what we are doing sounds interesting…
  - For more info, feel free to reach out :
    - @GaryMalouf
    - gary.malouf@mediacrossing.com
  - We hire remote engineers:
    - careers@mediacrossing.com