IRIS: USArray Short Course in Bloomington, Indian
Special focus: Oklahoma Wavefields

An exceedingly high-level overview of ambient noise processing with Spark and Hadoop

Presented by Rob Mellors but based on work by Steven Magana-Zook, Douglas Knapp and Eric Matzel

August 9, 2016

Previously presented at the 2017 Fall AGU

LLNL-PRES-694105

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344. Lawrence Livermore National Security, LLC

Lawrence Livermore National Laboratory

Goals

- **Can ”big data” technologies reduce the time needed for extensive seismological time series computations? (**yes**)**
- **How difficult is it to implement (**not simple**)**
- **Is it worthwhile? (**I think so…**)**

Test data and setup
14 days of 500 sps continuous data from 500 stations
16 node Hadoop cluster
Available java code for ambient noise processing designed and tested on a single workstation (e.g. data prep, cross-correlation, stacking)

[we are not using GPU’s]

What is big data (and how does it differ from HPC)

	HPC	Big Data
Typical use-case	CPU-bound problems	I/O-bound problems
Cost	$0.5 M and up	~8K per node
Input vs. output size	Input << output	Input >> output
Programming Language	Fortran, C, C++ (at LLNL)	Java, Scala, R, Python, and more through Streaming/Piping
I/O	Transferred to compute node	Local to compute node
Network Backbone	Infiniband (up to 300 Gb/s with enhanced data rate & 12x port width)	Ethernet (typically 1 Gb/s or 10 Gb/s)

From Magana-Zook, 2016

Big data

- Move computation, not data
- Fundamental tools
  - Hadoop distributed filesystem (HDFS)
    - Filesystem for distributed data across nodes
    - Not like a 'normal' filesystem (some unix commands but no directories)
    - Write-once, read-many
    - Resilient to failures (mostly…)
  - YARN (Yet Another Resource Negotiator)
    - Handles resources (CPU, memory)
    - Scheduling
  - Spark, MapReduce
    - Interface to YARN (and HDFS) that you can connect programs to

We are using Spark

Adapted from Magana-Zook, 2016

Hadoop: HDFS and YARN

- HDFS
  - HDFS
  - Replicates data across nodes
  - Assumes hardware failures occur
  - Written mostly in Java

- YARN
  - Resource manager
  - Handles system resources (CPU, memory) and node managers
  - Based on MapReduce but allows more complex applications

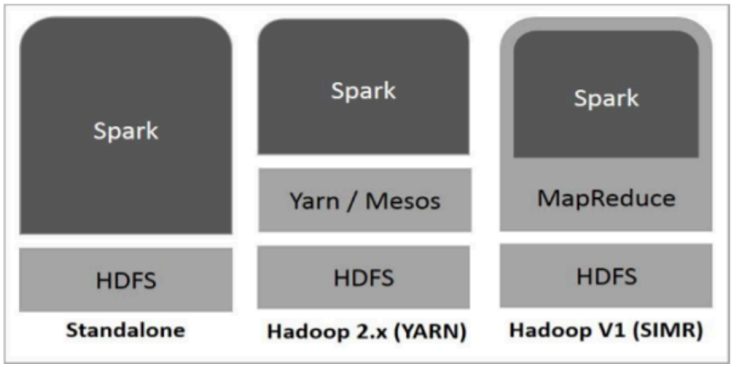
Average users should not have to worry about this too much

Spark

- General purpose distributed computation platform
- Scales from small to big
- Handles Scala, Java, and Python
- Data structure is the resilient distributed dataset (RDD)
- Handles iterative algorithms and keeps results in memory
- Needs cluster manager and distributed storage system.
- Converts tasks to a Java Virtual Machine (JVM)
- And has a machine learning library

Spark

- Define distributed dataset (user)
- Write driver program (user)
- Connect to data nodes (in YARN)
- Assigns tasks (JVM's)



Vendors and source (open-source): Hortenworks, Cloudera

https://www.tutorialspoint.com/apache_spark/apache_spark_introduction.htm

Example: Spark shell in Python

```
pyspark --master local[*]
```

Load data

```
datasetRDD = sc.textFile("file:///home/IRIS/OKWavefields/dataset.tsv")
```

```
# count records are in dataset
datasetRDD.cache().count()
```

```
# Create tuples of (key, average)
avgByKey = sumByKey.join(countByKey).map(lambda j: (j[0], j[1][0]/j[1][1]))
```

```
# View the results
avgByKey.take(2)
```

Adapted from Magana-Zook, 2016

Example: Submitting a job to YARN

```
spark-submit \
--master yarn \
--num-executors 2 \
--executor-cores 2 \
--class IU.IRIS.okalahoma.SparkBatchExercise \
OK_wavefields.jar \
dataset.tsv \
SparkResults
```

Adapted from Magana-Zook, 2016

Application to ambient noise correlation (ANC)

- Overview:
  - Need to calculate possible pairs, cross-correlate, and stack
  - These will be stages in Spark
  - Time required for each stage varies
  - All parts of one stage must complete before going to the next stage

- Write driver
  - Read in all data information
  - Calculate pairs
  - Divide into tasks to be assigned
    - Use 1 hour all station pairs as a basic unit
    - Handles missing data
  - When complete stack
  - Includes call to jar files for the computations
  - On the order of 200 lines for driver

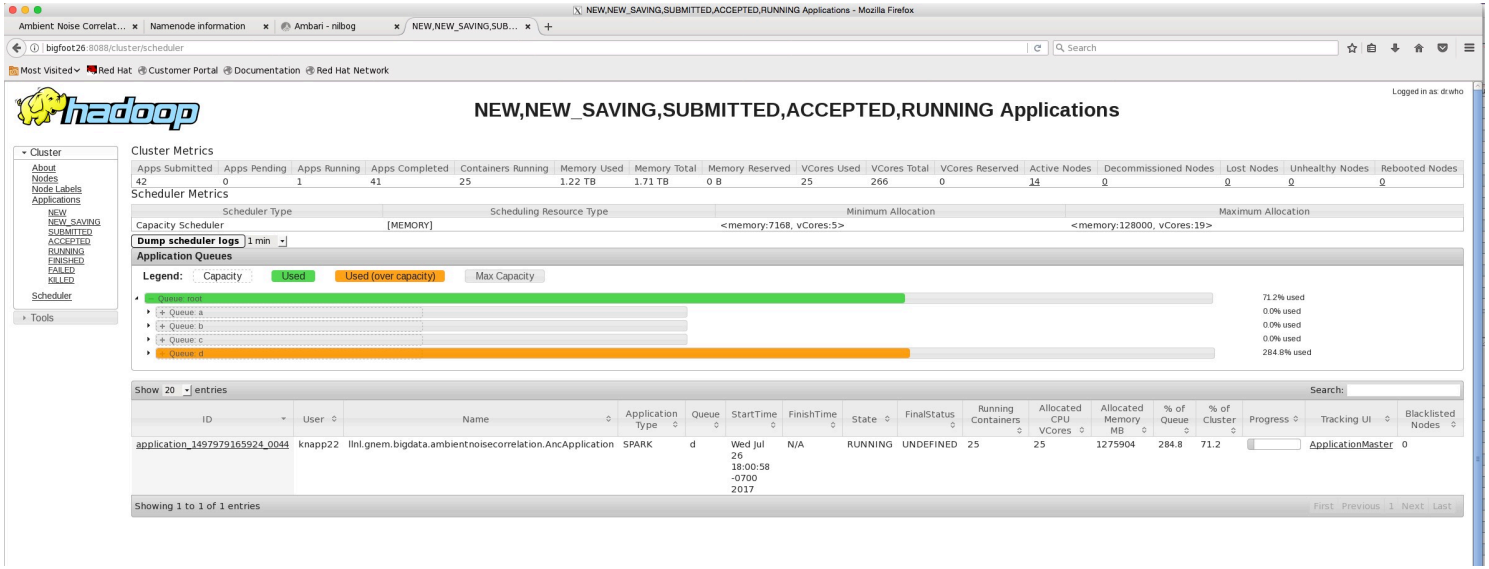
- Script passes driver to YARN (less than 20 line)

- Then wait…..

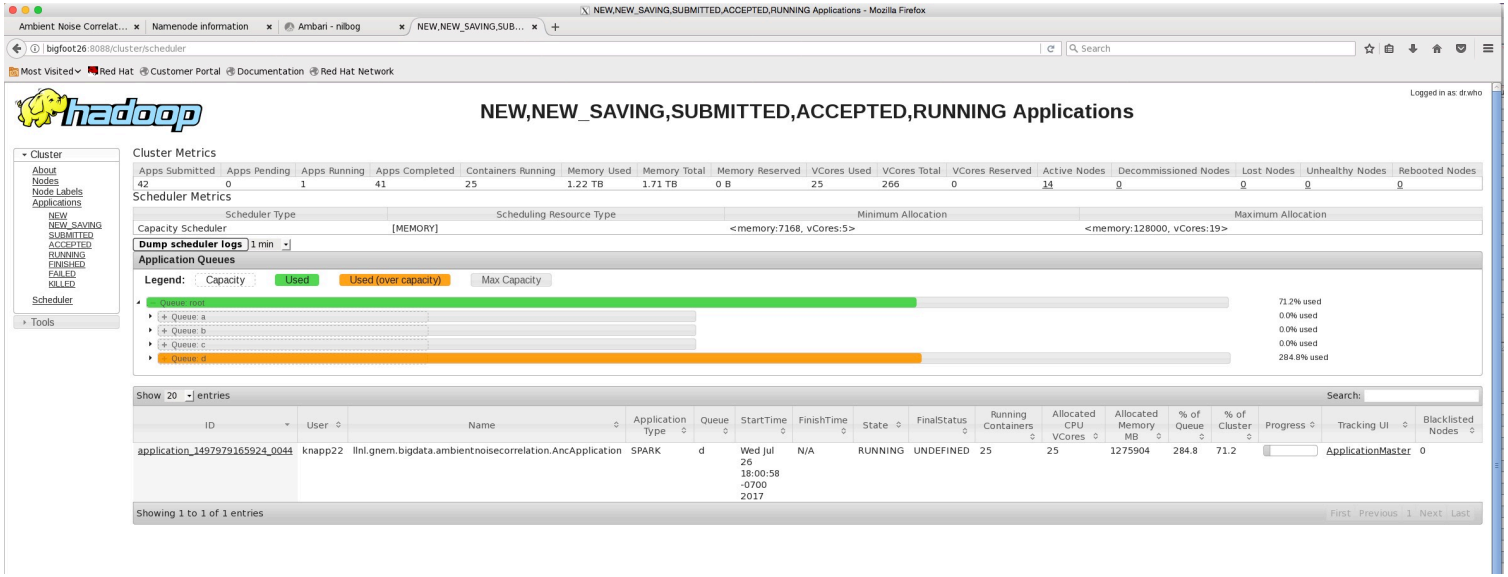
Results so far

- 500 stations, 14 days, about a week to 10 days (~16 node cluster)
- Have had some problems – handled two drive drive failures but not three.
- Be careful with driver – easy to accidentally skip pair

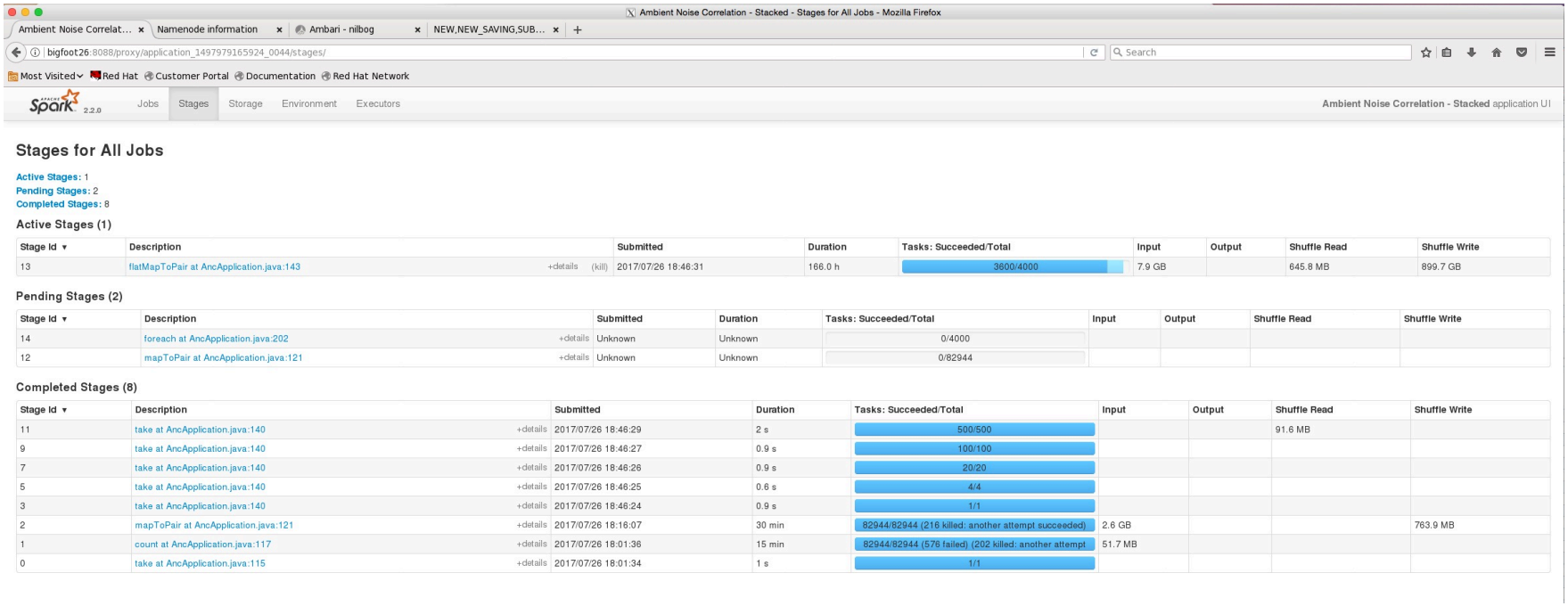
Monitoring Hadoop cluster



YARN


NEW,NEW_SAVING,SUBMITTED,ACCEPTED,RUNNING Applications - Mozilla Firefox
bigfoot26:8088/cluster/scheduler
Most Visited Red Hat Customer Portal Documentation Red Hat Network
Logged in as: dr.who
NEW,NEW_SAVING,SUBMITTED,ACCEPTED,RUNNING Applications
Cluster
About
Nodes
Node Labels
Applications
NEW
NEW_SAVING
SUBMITTED
ACCEPTED
RUNNING
FINISHED
FAILED
KILLED
Scheduler
Tools
Cluster Metrics
Apps Submitted Apps Pending Apps Running Apps Completed Containers Running Memory Used Memory Total Memory Reserved VCores Used VCores Total VCores Reserved Active Nodes Decommissioned Nodes Lost Nodes Unhealthy Nodes Rebooted Nodes
42 0 1 41 25 1.22 TB 1.71 TB 0 B 25 266 0 14 0 0 0 0
Scheduler Metrics
Scheduler Type Scheduling Resource Type Minimum Allocation Maximum Allocation
Capacity Scheduler [MEMORY] <memory:7168, vCores:5> <memory:128000, vCores:19>
Dump scheduler logs 1 min
Application Queues
Legend: Capacity Used Used (over capacity) Max Capacity
Queue: root
Queue: a
Queue: b
Queue: c
Queue: d
71.2% used
0.0% used
0.0% used
0.0% used
284.8% used
Show 20 entries
Search:
ID User Name Application Type Queue StartTime FinishTime State FinalStatus Running Containers Allocated CPU VCores Allocated Memory MB % of Queue % of Cluster Progress Tracking UI Blacklisted Nodes
application_1497979165924_0044 knapp22 llnl.gnem.bigdata.ambientnoisecorrelation.AncApplication SPARK d Wed Jul 26 18:00:58 -0700 2017 N/A RUNNING UNDEFINED 25 25 1275904 284.8 71.2 ApplicationMaster 0
Showing 1 to 1 of 1 entries
First Previous 1 Next Last

Spark - stages



Conclusions

- If it is Dec. 1 and you want to process a bunch of data for AGU, probably not the best choice
- If you have an interest in big data or have a truly large dataset (1000's of stations), it is worthwhile.
- Easy to start
  - Cluster not required to learn
  - Can run Hadoop, Spark, etc on a laptop for testing
- Might be able to do it on Amazon cloud (but $ to transfer)
- Technology advancing quickly
- Or you can apply to be a summer student at LLNL