Deploying Hadoop

CMPT 732, Fall 2017

Moving Parts

Central/master/core stuff:

- HDFS NameNode and SecondaryNameNode
- YARN ResourceManager

These can be on one computer or separated. Separating (and hardware choices) will depend on number of nodes in cluster.

Important: NameNode is a single point of failure, unless configured with [high availablility](#).

Worker nodes:

- HDFS DataNode
- YARN NodeManager
- HBase RegionServer, Cassandra node, …

Our Cluster

As an example, for our cluster:

- `gateway`: you login and start jobs.
- `master`: YARN ResourceManager, HDFS NameNode, ….
- 6 worker nodes: HDFS DataNode, YARN NodeManager, HBase RegionServer, ….

Example Configurations

- Our cluster: see files in `/etc/hadoop/conf/`, `/etc/spark/conf/`.
- Raspberry Pi cluster: [in GitHub](#).
- VM cluster from A5a: [in GitHub](#).

Are they perfect examples of Hadoop configuration? Probably not.

Hardware

In theory: any collection of computers.

In practice: there's no point having slow processors, minimal memory, or a slow network.

In reality: likely Amazon [EMR](#). Or [EC2](#) if you want to configure yourself.

Capacity Planning

Depends on the tasks you're doing, obviously.

Lesson from A5a: it's easy to add worker nodes (or replace with faster) if necessary.

- [Hadoop Operations: Planning a Hadoop Cluster](#).
- [Cloudera Hadoop/HBase Capacity Planning](#).

Processor

How much work do you need to get done, and how fast? How well does it parallelize?

Memory

How big is the working set × how many threads/processes will you be running?

Disk

Spinning is bigger, SSD is faster; more disks multiply throughput. Divide capacity by replication factor.

Disk Location

High-speed, low-latency networks make centralized storage feasable.

In general: bigger nodes are better, until cost becomes prohibitive.

Fewer large nodes means less network usage, better able to run less-parallel tasks, more threads per process (sharing working sets).

e.g. on our cluster, if you want to start workers that need 100GB of memory and 17 cores, you're out of luck.

Decisions to Make

- Hardware: how much?
- Nodes for central (NameNode, ResourceManager): one or several? HDFS high availability?
- Configure yourself, or a distribution? YARN or Mesos or Spark Standalone or …?
- HDFS block size and default replication. Or some centralized storage and fast network?
- YARN scheduler.

Hadoop Distributions

There's a lot of stuff to install: why not let somebody else worry about some of the details?

- [Cloudera](#): what is running our cluster.
- [Hortonworks HDP](#).
- [MapR](#).
- [Amazon EMR](#): EC2 + Hadoop set up automatically.