AWS EMR

Elastic MapReduce. Managed Hadoop framework on EC2 instances. Includes, Spark, HBase, Hive, etc.

EMR cluster is made up of collection of EC2 nodes.

- Master node – tracks status of tasks, monitors cluster
- Core node: hosts hdfs data and runs tasks, multin node cluster have at least one, can be scaled up and down
- Tas node: Run tasks, does not host data, optional

Ways to use EMR:

- Transient cluster – runs the cluster once to complete the job and shuts down
- Long running – basically a data warehouse with periodic processing on large dataset. Is more persistent

Hadoop – comprised of MapReduce, Yarn and HDFS. HDFS stores data across the cluster. Yarn (Yet another resource negotiater) manages cluster resources for multiple data processing frameworks. MapReduce – framework for data processing. Maps data to key/value pairs. Reduces intermediate results to final output

Apache Spark – mostly preferred over Hadoop because of in memory caching. Also has query optimization execution.

How spark works –

At the heart of it is the SparkContext or the driver program. Coordinates all other processes using Cluster Manager (Spark, YARN).

Can work with Kinesis Streaming. Also works with RedShift because it is just like another SQL data source for Spark.

Hive – Sits on top of MapReduce. Familiar SQL syntax. Can use it to query data quickly.

Apache Zeppelin – like a iPython notebook on the cluster. Allows you to run the spark code interactively.

EMR Notebook – similar to zeppelin but with more AWS integration. Backed up to S3. Provisions Clusters from the notebook.