

PROCESSING DATA WITH AWS EMR

LEARNING OBJECTIVES

At the end of this unit, you should be able to:

- Understand the Overview of Amazon EMR
- Understand the Benefits of using Amazon EMR
- Understand the Amazon EMR Architecture
- Understand the types of data sources that can be integrated with Amazon EMR for workload flexibility
- Work with Data Processing Frameworks on Amazon EMR
 - Apache Hadoop
 - Apache HIVE
 - Apache Spark
- Understanding the Use of HUE and Presto.



OVERVIEW OF AMAZON EMR

- Amazon EMR is a managed cluster platform that simplifies running big data frameworks, such as Apache Hadoop and Apache Spark, on AWS to process and analyze vast amounts of data.
- Using these frameworks and related open-source projects, such as Apache Hive and Apache Pig, you can process data for analytics purposes and business intelligence workloads.
- We can also use Amazon EMR to transform and move large amounts of data into and out of other AWS data stores and databases, such as Amazon Simple Storage Service (Amazon S3) and Amazon DynamoDB.

CLUSTERS AND NODES

The central component of Amazon EMR is the cluster.

A cluster is a collection of Amazon Elastic Compute Cloud (Amazon EC2) instances.

Each instance in the cluster is called a node.

Each node has a role within the cluster, referred to as the node type.

Amazon EMR also installs different software components on each node type, giving each node a role in a distributed application like Apache Hadoop.

TYPES OF NODES IN AMAZON EMR CLUSTER

Master node:

A node that manages the cluster by running software components to coordinate the distribution of data and tasks among other nodes for processing. The master node tracks the status of tasks and monitors the health of the cluster. Every cluster has a master node, and it's possible to create a single-node cluster with only the master node.

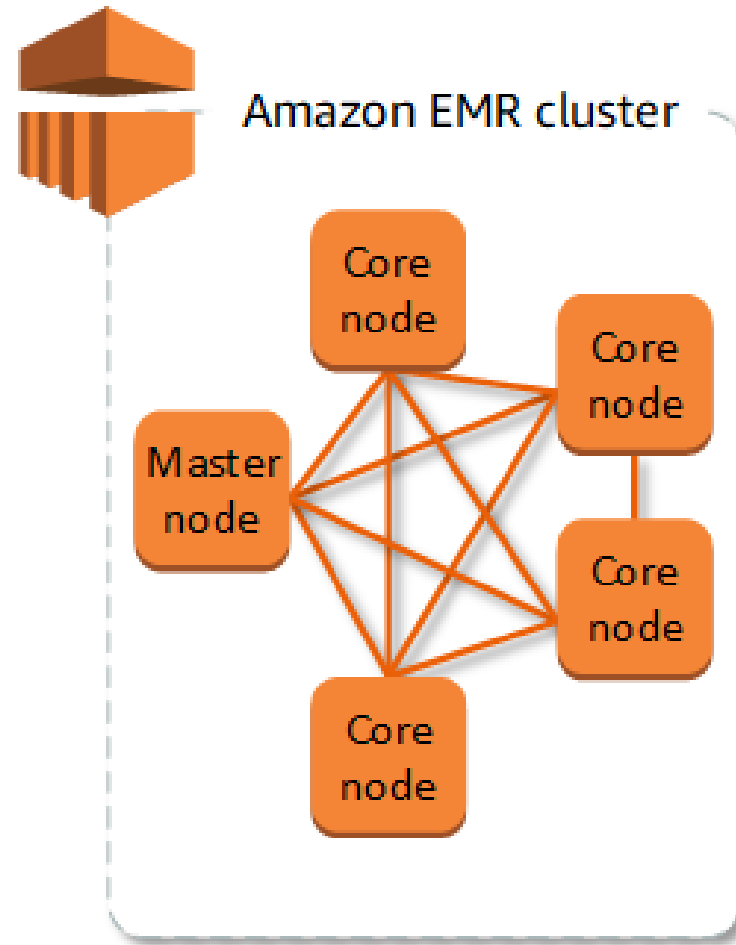
Core node:

A node with software components that run tasks and store data in the Hadoop Distributed File System (HDFS) on your cluster. Multi-node clusters have at least one core node.

Task node:

A node with software components that only runs tasks and does not store data in HDFS. Task nodes are optional.

EMR CLUSTER WITH ONE MASTER NODE AND FOUR CORE NODES



BENEFITS OF AMAZON EMR

1. Cost Savings
2. AWS Integration
3. Deployment
4. Scalability and Flexibility
5. Reliability
6. Security
7. Monitoring

COST SAVINGS

Amazon EMR pricing depends on the instance type and number of EC2 instances that you deploy and the region in which you launch your cluster.

On-demand pricing offers low rates, but you can reduce the cost even further by purchasing Reserved Instances or Spot Instances.

Spot Instances can offer significant savings—as low as a tenth of on-demand pricing in some cases

AWS INTEGRATION

Amazon EMR integrates with other AWS services to provide capabilities and functionality related to

- networking
- storage
- security

and so on, for your cluster based on your requirements.

DEPLOYMENT

The EMR cluster consists of EC2 instances, which perform the work that you submit to your cluster.

When we launch the cluster, Amazon EMR configures the instances with the applications that you choose, such as

- Apache Hadoop or
- Spark.

Choose the instance size and type that best suits the processing needs for your cluster:

- batch processing,
- low-latency queries,
- streaming data, or
- large data storage

SCALABILITY AND FLEXIBILITY

Amazon EMR provides flexibility to scale your cluster up or down as your computing needs change.

We can resize your cluster to add instances for peak workloads and remove instances to control costs when peak workloads subside.

RELIABILITY

Amazon EMR monitors nodes in your cluster and automatically terminates and replaces an instance in case of failure.

Amazon EMR provides configuration options that control how your cluster is terminated

- Automatically or
- Manually.

If we configure the cluster to be automatically terminated, it is terminated after all the steps complete. This is referred to as a transient cluster.

However, we can also configure the cluster to continue running after processing completes so that we can choose to terminate it manually when we no longer need it.

SECURITY

Amazon EMR leverages other AWS security services, such as

- IAM
- Amazon VPC,
- Amazon EC2 key pairs
- Security Groups
- Encryption

To help you secure your clusters and data.

MONITORING

We can use the Amazon EMR management interfaces and log files to troubleshoot cluster issues, such as failures or errors.

Amazon EMR provides the ability to archive log files in Amazon S3 so you can store logs and troubleshoot issues even after your cluster terminates.

Amazon EMR also provides an optional debugging tool in the Amazon EMR console to browse the log files based on steps, jobs, and tasks

MANAGEMENT INTERFACES

We can interact with Amazon EMR in several ways :

Console — A graphical user interface that you can use to launch and manage clusters.

AWS Command Line Interface (AWS CLI) — A client application you run on your local machine to connect to Amazon EMR and create and manage clusters.

Software Development Kit (SDK) — SDKs provide functions that call Amazon EMR to create and manage clusters.

Service API — A low-level interface that you can use to call the web service directly, using JSON.

AMAZON EMR - ARCHITECTURE

Amazon EMR service architecture consists of several layers, each of which provides certain capabilities and functionality to the cluster. As mentioned below.

Data Source

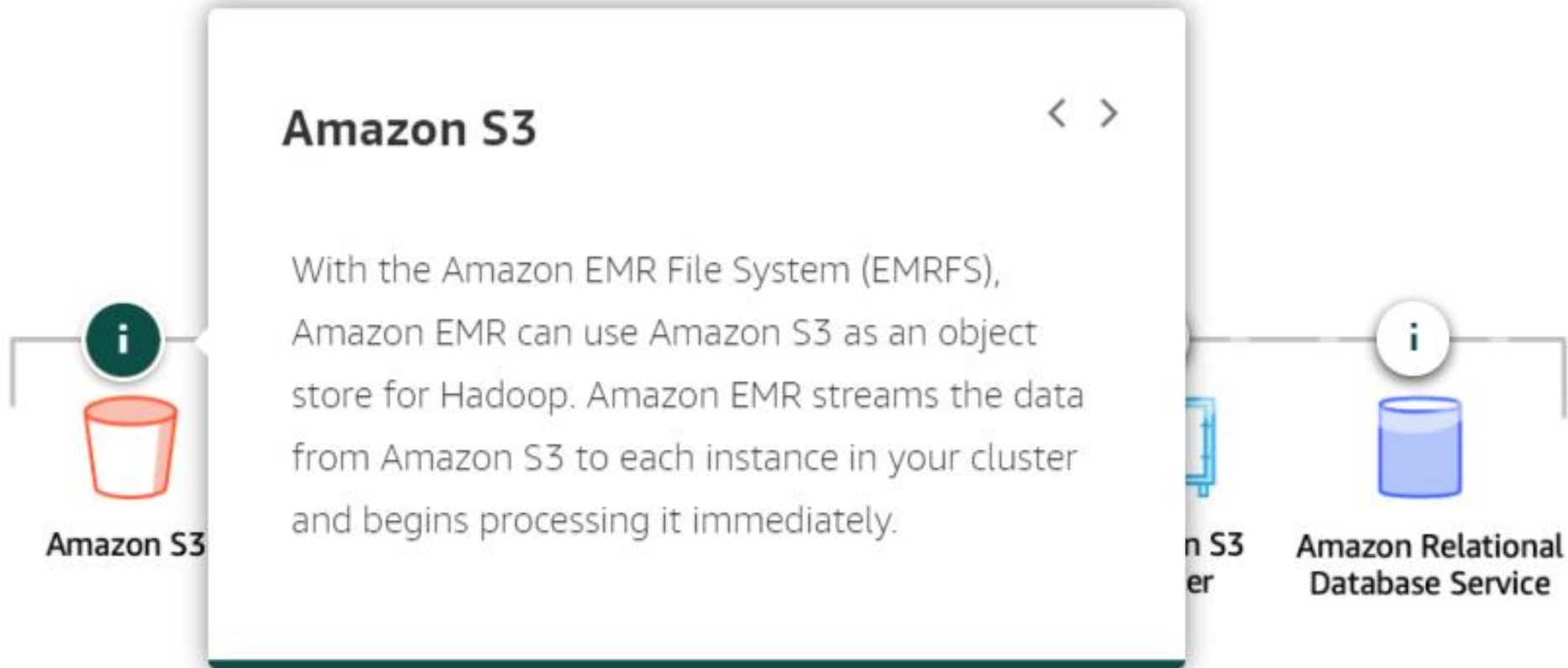
Cluster Resource Management

Data Processing Frameworks

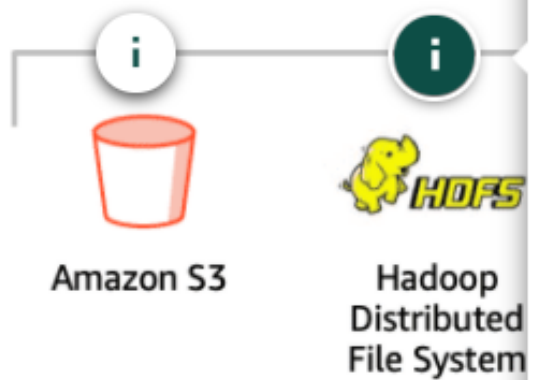
DATA SOURCES THAT CAN BE INTEGRATED WITH AMAZON EMR FOR WORKLOAD FLEXIBILITY



AMAZON S3



HDFS



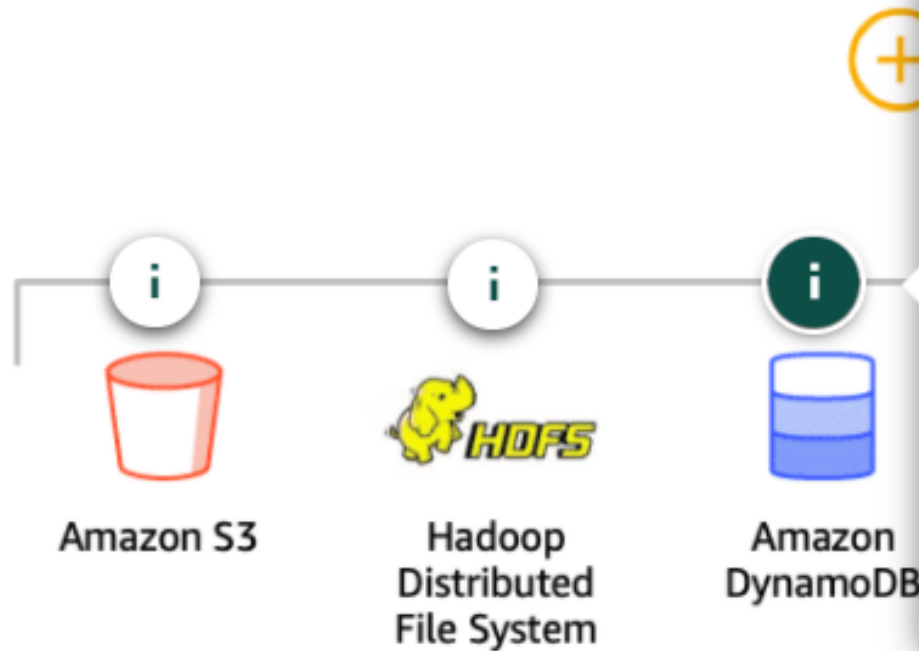
HDFS



Amazon EMR's current topology groups its instances into three logical instance groups: Master Group, which runs the YARN Resource Manager and the HDFS Name Node Service; Core Group, which runs the HDFS DataNode Daemon and the YARN Node Manager service; and Task Group, which runs the YARN Node Manager service. Amazon EMR installs HDFS on the storage associated with the instances in the Core Group.

ational
Service

AMAZON DYNAMO DB



Amazon DynamoDB



Amazon EMR has direct integration with Amazon DynamoDB so you can process data stored in DynamoDB and transfer data between Amazon DynamoDB, Amazon S3, and HDFS.

AMAZON REDSHIFT

Amazon Redshift

You can use the COPY command to load data in parallel from an Amazon EMR cluster. See the full process of how to load data from Amazon EMR into Amazon Redshift [here](#).



Amazon EMR



Amazon
Redshift

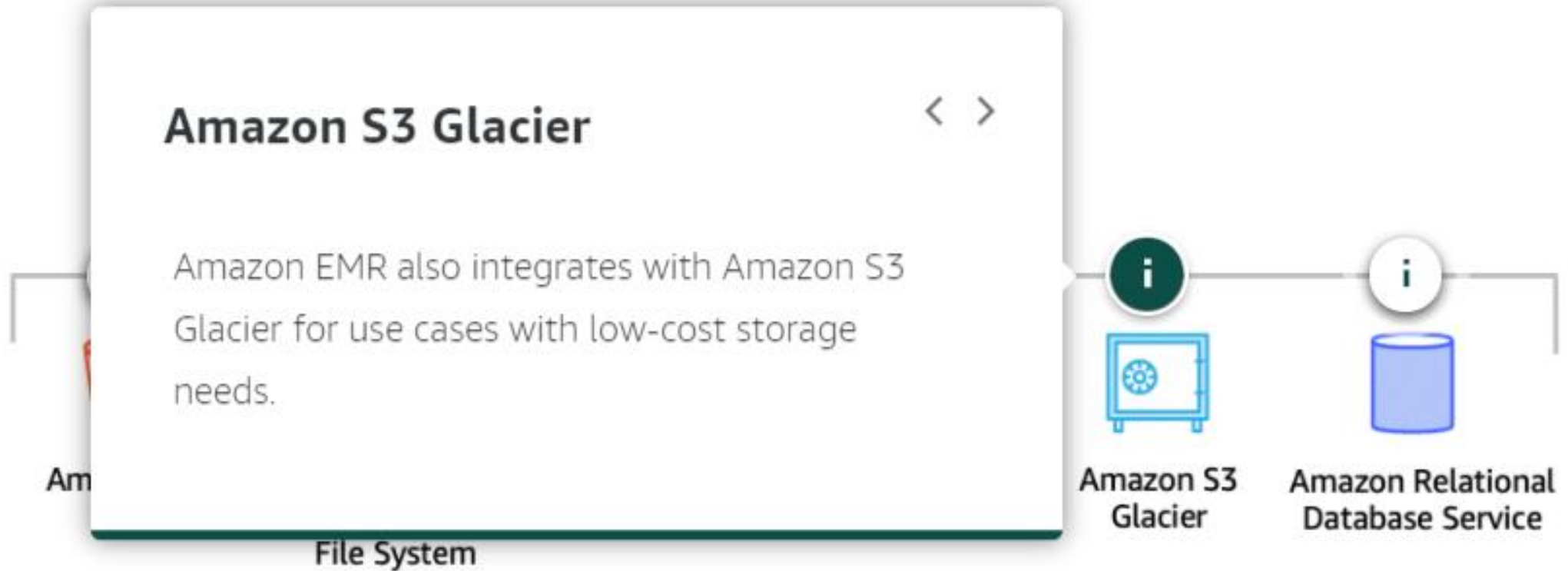


Amazon S3
Glacier

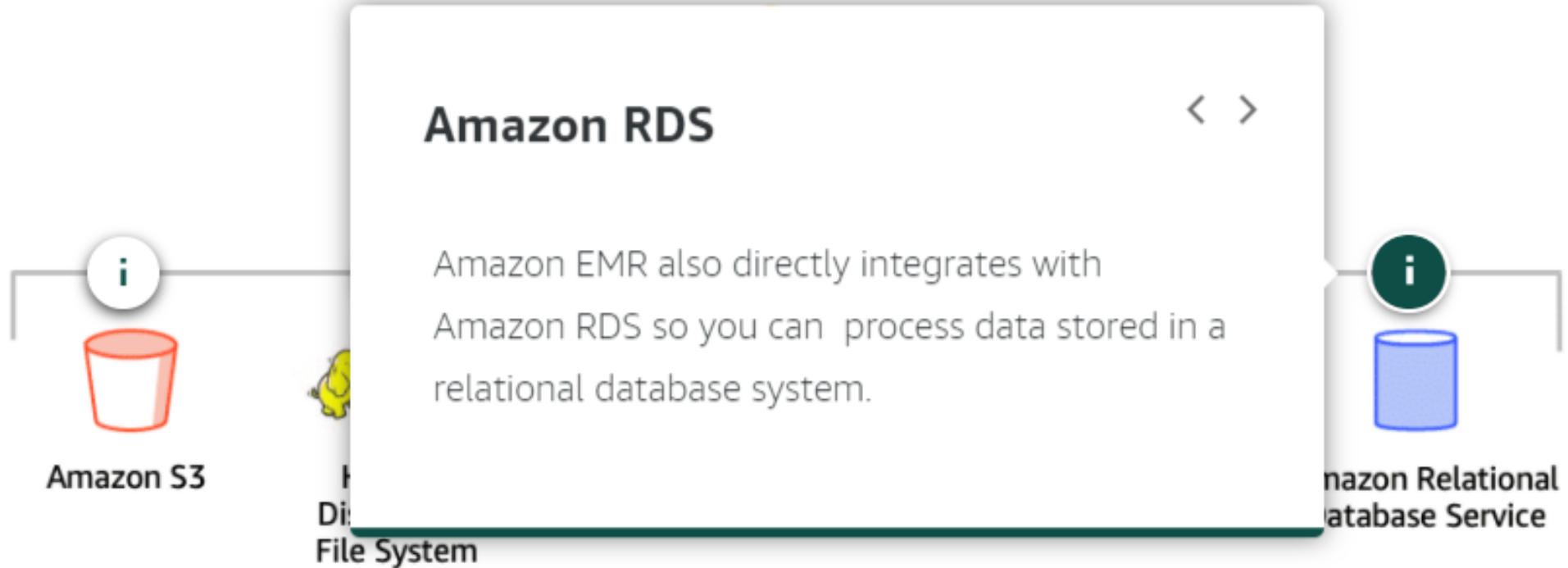


Amazon Relational
Database Service

AMAZON S3 GLACIER



AMAZON RDS



CLUSTER RESOURCE MANAGEMENT

The resource management layer is responsible for managing cluster resources and scheduling the jobs for processing data.

By default, Amazon EMR uses **YARN (Yet Another Resource Negotiator)**, which is a component introduced in Apache Hadoop 2.0 to centrally manage cluster resources for multiple data-processing frameworks.

However, there are other frameworks and applications that are offered in Amazon EMR that do not use YARN as a resource manager.

DATA PROCESSING FRAMEWORKS

The data processing framework layer is the engine used to process and analyze data.

There are many frameworks available that run on YARN or have their own resource management.

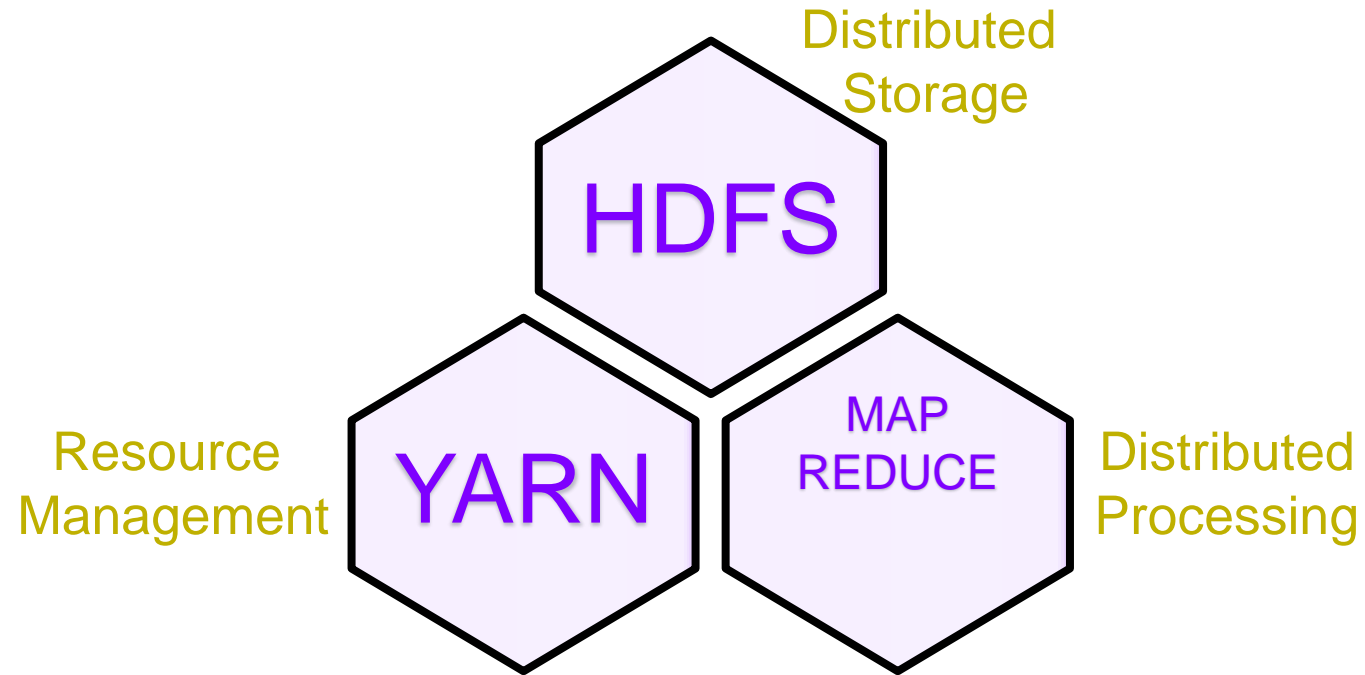
Different frameworks are available for different kinds of processing needs, such as

- Batch – Apache Hadoop , Apache Hive
- in-memory – Apache Spark (Core & SQL)
- Streaming – Apache Spark (Streaming)

APACHE HADOOP

- Framework that allows distributed storage & distributed processing of large data sets across cluster of computers.
- Uses simple programming models.
- Cluster can be built using commodity hardware.

CORE COMPONENTS OF HADOOP



HDFS – HADOOP DISTRIBUTED FILE SYSTEM

The 3 core components of Hadoop framework are :

1. **MapReduce** – A software programming model for processing large sets of data in parallel
2. **HDFS** – The Java-based distributed file system that can store all kinds of data without prior organization.
3. **YARN** – A resource management framework for scheduling and handling resource requests from distributed applications.

APACHE HIVE

- An open-source **distributed data warehousing** solution built on top of Hadoop
- Not a RDBMS

Facilitates querying and managing large datasets residing in distributed storage using SQL like language

Built on top of **Apache Hadoop**, it provides

Tools to enable easy data extract/transform/load (ETL)

A mechanism to impose structure on a variety of data formats

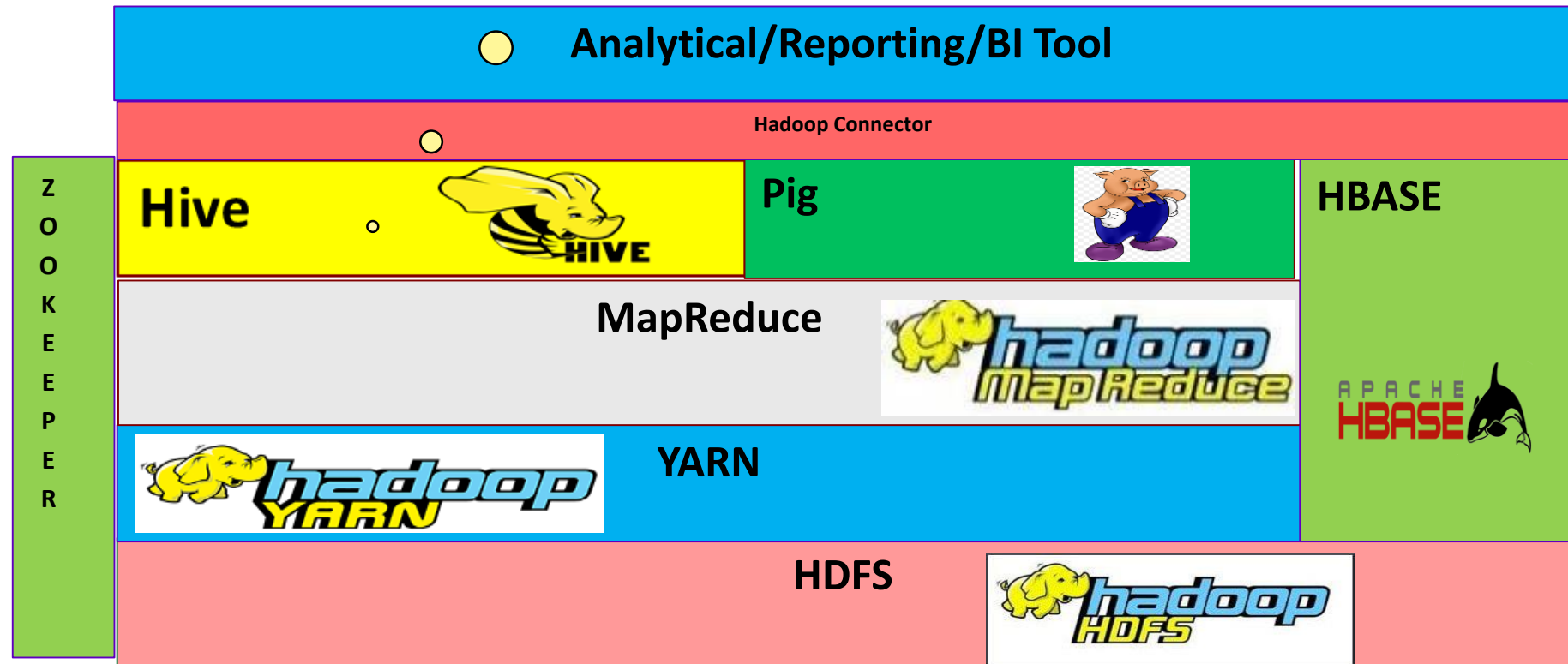
Access to files stored either directly in **Apache HDFS** or in other data storage systems such as **Apache HBase**

Query execution via **MapReduce**

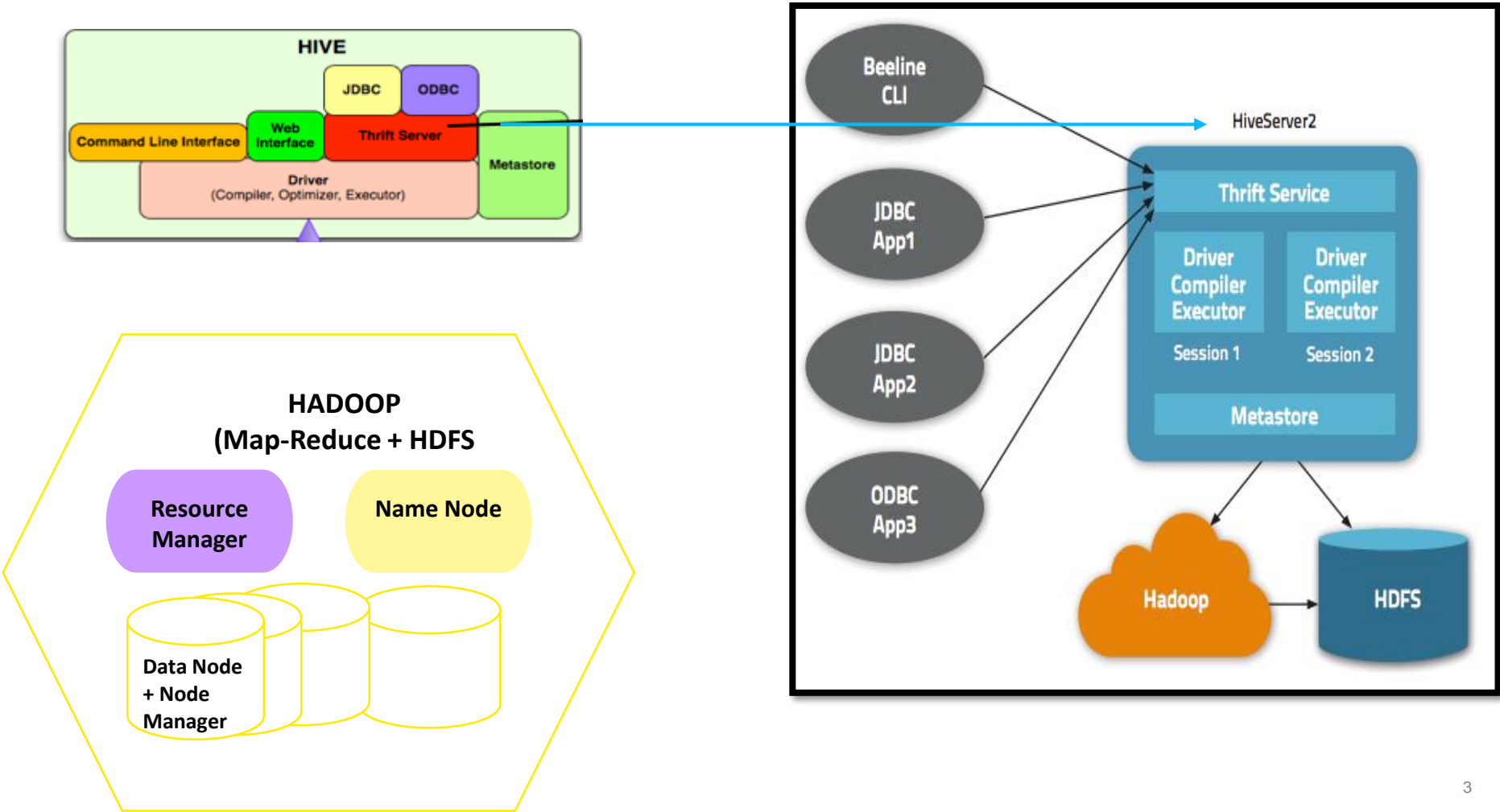
- provides an SQL dialect, called **Hive Query Language** (abbreviated **HiveQL** or just **HQL**) for querying data stored in a Hadoop cluster.
- *Hive is a way to allow non-Java programmers access to the data stored in Hadoop clusters using HQL*

APACHE HIVE

Hive – the SQL tool in
Hadoop



HIVE ARCHITECTURE



HIVE ARCHITECTURE

Metastore:

stores all the structure information of the various tables and partitions in the warehouse including column and column type information,

Stores the serializers and deserializers necessary to read and write data and the corresponding HDFS files where the data is stored.

Separate RDBMS is used for Metastore

Driver:

manages the life cycle of HiveQL query as it moves thru' HIVE,
also manages session handle and session statistics.

Query compiler:

Compiles HiveQL into a directed acyclic graph of map/reduce tasks.

Execution engines:

The component executes the tasks in proper dependency order; **interacts with Hadoop.**

Hive command-line interface (CLI) [hive> prompt] is used to connect to Hive.

HOW DOES HIVE WORKS ?

All commands and queries go to the **Driver**,
it compiles the input,
optimizes the computation required, and
executes the required steps, usually with MapReduce jobs.

When MapReduce jobs are required, Hive doesn't generate Java MapReduce programs.

It uses built-in, **generic Mapper and Reducer modules** that are driven by an **XML file representing the “job plan.”**

Generic modules function like **mini language interpreters** and the “language” to drive the computation is encoded in XML.

INTRODUCTION TO HUE

What is HUE ?

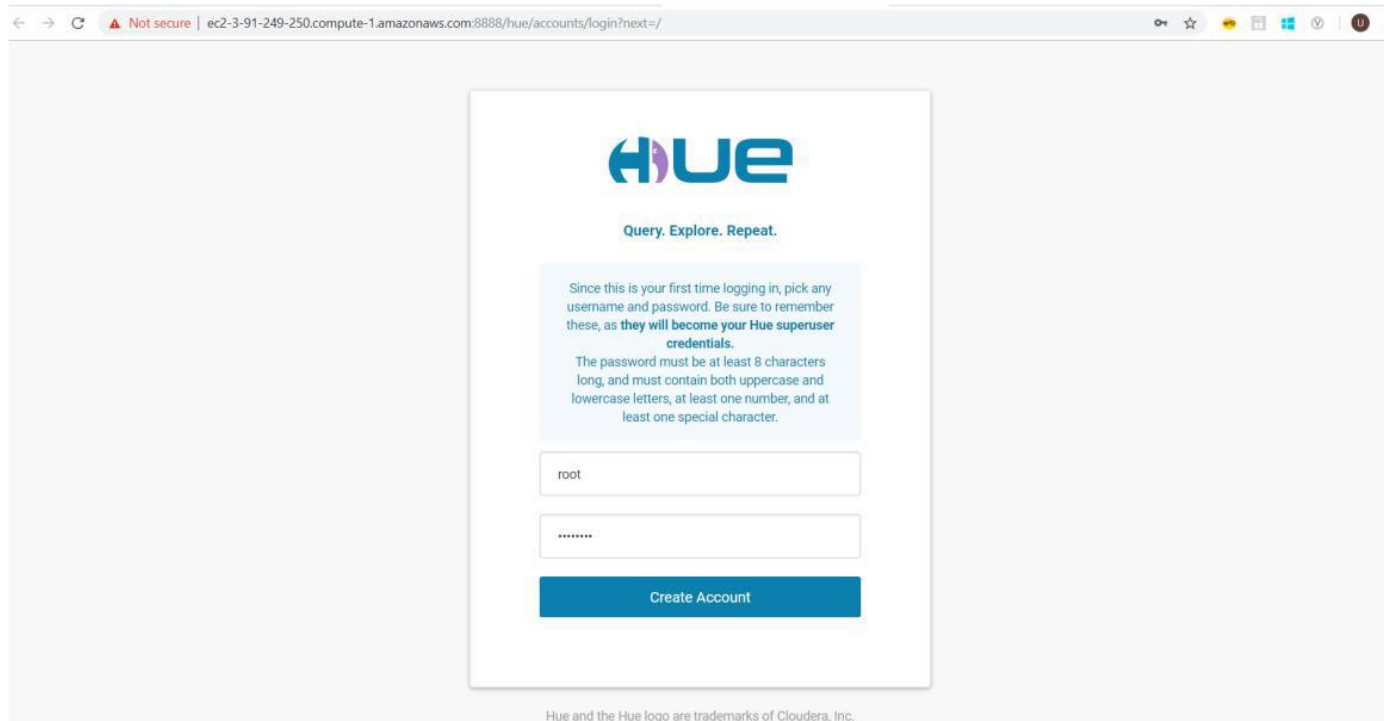
- Hadoop User Experience is a Open source , web bases graphical user interface.
- Which can be used to work with Amazon EMR and Apache Hadoop Application
- HUE groups several Hadoop ecosystem applications into a Single console
- It is browser based and helps us to Manage the EMR cluster easily.
- HUE acts a front end for the applications that run on your Cluster.
- The applications in Hue, such as the Hive and Pig editors, replace the need to log in to the cluster to run scripts interactively using each application's respective shell.

INTERACTING WITH HUE

How to work with HUE ?

- After creating a EMR Cluster with HUE, Add Custom TCP with port **8888** in the in-bound rules for Master Node Security Group
- Access HUE using a Browser as --- : **http://< Master public DNS>:8888**
- Every first time when we open HUE , the application would prompt for Username and Password.
- Once we set a Username and Password that will become your Hue superuser credentials.

HUE LOGIN SCREEN



The screenshot shows a web browser window with the URL `ec2-3-91-249-250.compute-1.amazonaws.com:8888/hue/accounts/login?next=`. The page displays the Hue logo and the tagline "Query. Explore. Repeat." Below this, a light blue box contains instructions for first-time users: "Since this is your first time logging in, pick any username and password. Be sure to remember these, as **they will become your Hue superuser credentials.** The password must be at least 8 characters long, and must contain both uppercase and lowercase letters, at least one number, and at least one special character." The form includes a username field with the text "root", a password field with masked characters "*****", and a blue "Create Account" button. At the bottom, a small footer states "Hue and the Hue logo are trademarks of Cloudera, Inc."

← → ↻ ⚠ Not secure | ec2-3-91-249-250.compute-1.amazonaws.com:8888/hue/accounts/login?next=

HUE

Query. Explore. Repeat.

Since this is your first time logging in, pick any username and password. Be sure to remember these, as **they will become your Hue superuser credentials.**

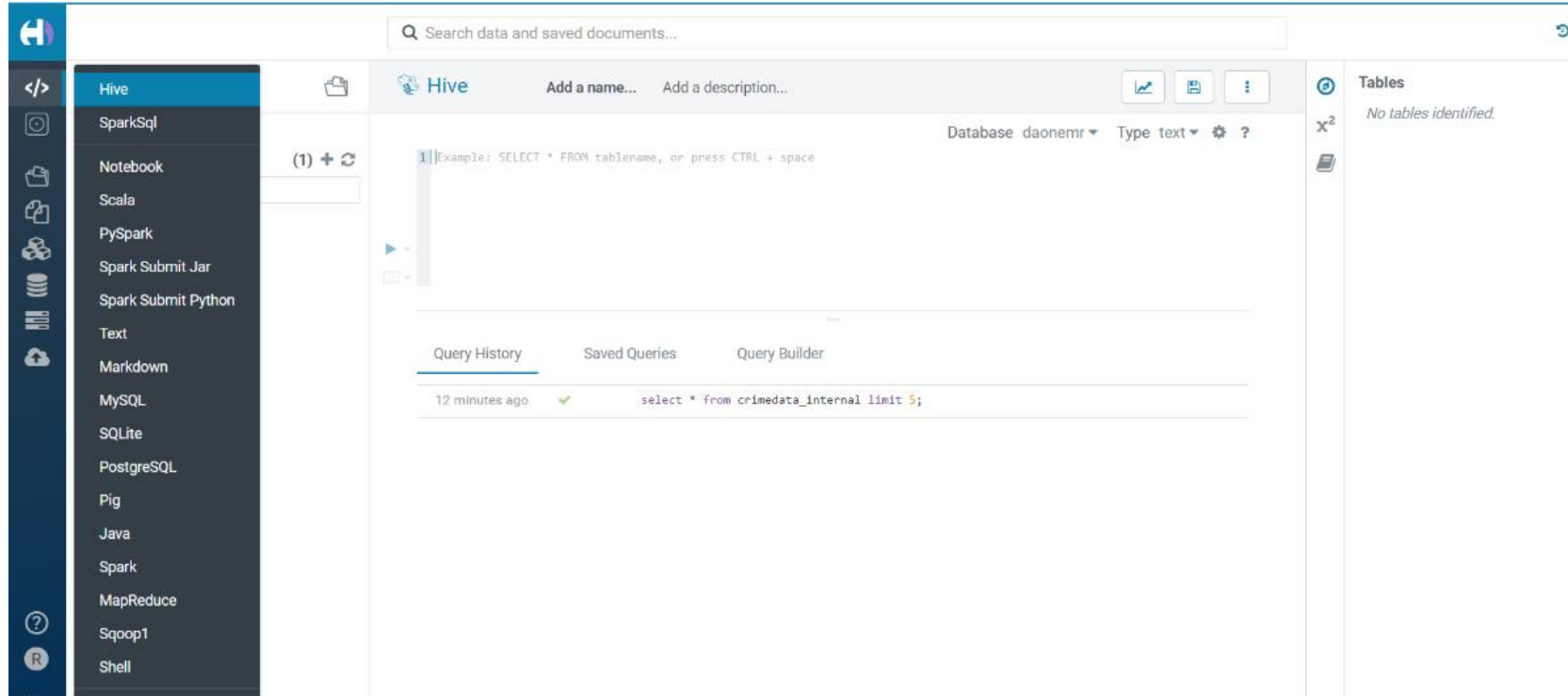
The password must be at least 8 characters long, and must contain both uppercase and lowercase letters, at least one number, and at least one special character.

root

Create Account

Hue and the Hue logo are trademarks of Cloudera, Inc.

HUE HOME PAGE



PRESTO

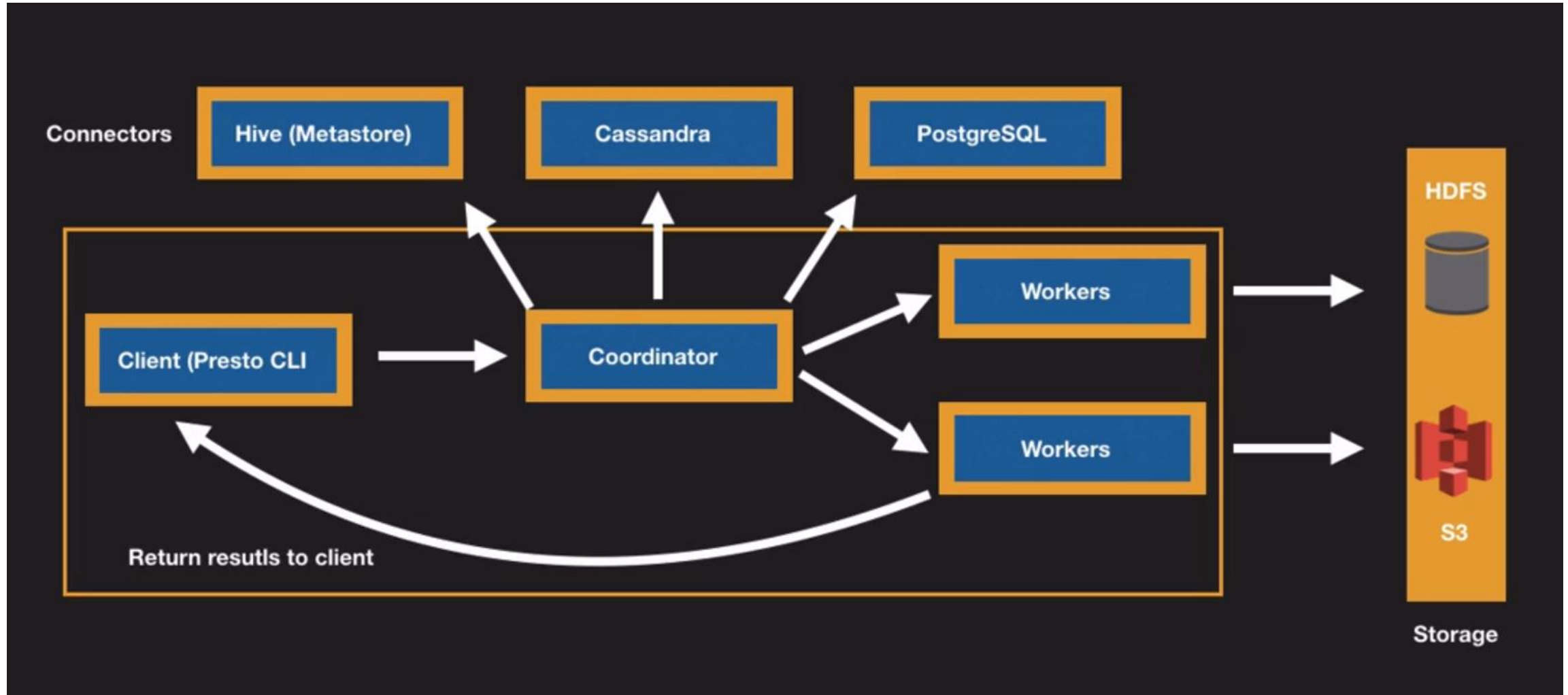
What is Presto ?

- It is an Open Source In Memory Distributed SQL Query Engine.
- It was developed by facebook in 2012 and was open sourced in 2013.

What is Presto Used for ?

- It is used for running Analytica queries against variety of Data sources holding large volume of data.
- Presto Query Engine is faster than HIVE.

PRESTO ARCHITECTURE



ADVANTAGES OF PRESTO

Presto can be used run query against Variety of Data source as listed below :

1. Relational Databases
2. NoSQL Databases
3. DW frameworks like HIVE
4. Streaming frameworks like Kafka

It is highly concurrent and can be used run thousands of queries every Sub-Seconds to Minutes.

As It is an In-Memory Processing Engine, it avoids un-necessary disc I/O operations there by reducing the Latency.

It does not need an Interpreter like HIVE need it.

WHEN NOT TO USE PRESTO

- Presto is not a Database , hence it is not meant for OLTP type of transactions.
- When there is a need to Join very large Datasets which need optimization, then better not to use Presto , instead we can use HIVE.
- Whenever there is a need to run large Batch jobs on very large volume of Data, Presto can be avoided.

APACHE SPARK

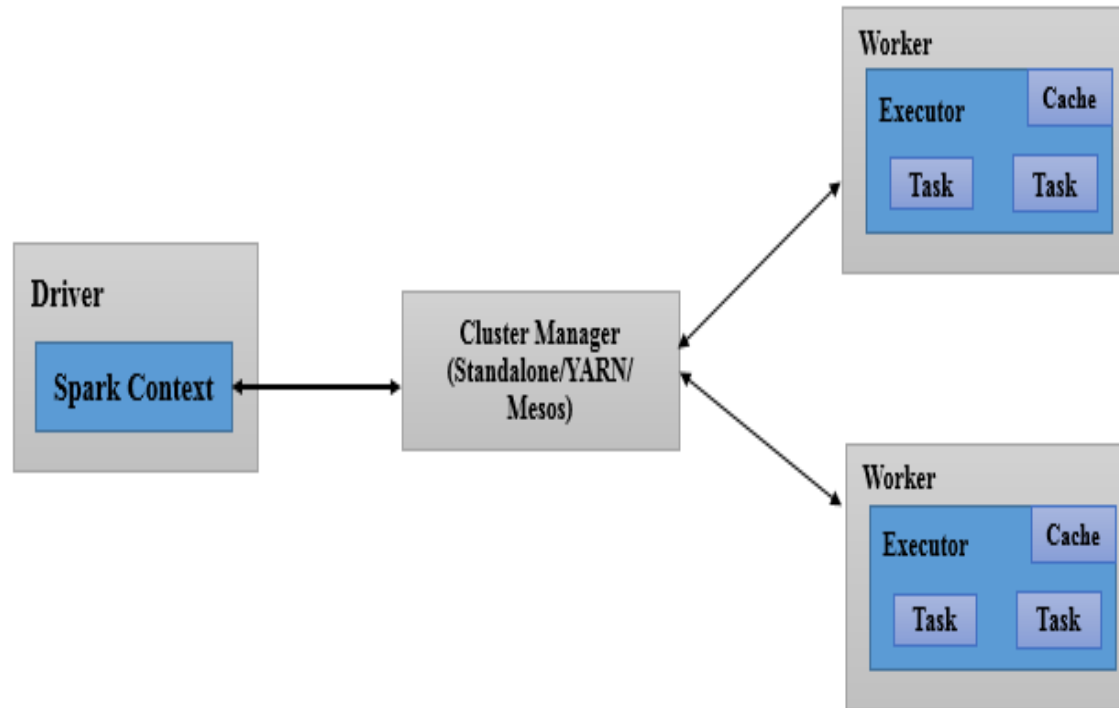
- Open source fast and general engine for large data processing.
- Developed in 2009 in UC Berkeley's AMP Lab, and open sourced in 2010 as an Apache project.
- Written in Scala.
- In memory processing framework.
- Provides high-level APIs in Java, Scala, Python and R.

WHY SPARK ?

- Unified framework to manage big data requirements.
- Provides high level operators such as filter, map, etc..
- 100x faster in memory, 10x faster on disk.
- Spark Shell
 - Interactive - for data exploration and testing.
 - Scala or Python
- Spark Applications
 - For large scale and data processing.
 - Scala, Python or Java

SPARK ARCHITECTURE

- Master/Slave architecture with cluster manager and two daemons.
- Daemons are:
 - .Master – Master/ Driver Process
 - .Worker – Slave Process



SPARK ARCHITECTURE

- A spark cluster has single coordinator called driver and many distributed workers.
- Driver communicate with large number of distributed workers called executors.
- Cluster Manager is responsible for scheduling and allocating resources to a Spark Job.

SPARK ARCHITECTURE – ROLE OF DRIVE

- The driver program
 - Entry point of the Spark Shell (Scala, Python, and R).
 - Runs application main () function
 - Is the place where Spark Context is created.
- Responsible for:
 - Scheduling job, negotiating with the cluster manager.
 - Converting a user application into smaller execution units i.e. tasks.
- You can access running spark application information through default Web UI at port 4040

SPARK ARCHITECTURE – ROLE OF EXECUTOR

- Distributed agent.
- Responsible for:
 - Executing tasks.
 - Performing data processing.
 - Reading from and Writing data to external sources.
 - Storing computation results data in-memory, cache or disk.
 - Interacting with the storage systems.

SPARK ARCHITECTURE – ROLE OF CLUSTER MANAGER

- It is an external service.
- Responsible for:
 - acquiring resources and allocating them to a spark application.
 - allocation and deallocation of various physical resources such as CPU, memory, etc..,

QUIZ : QUESTION 1

If you have to start using resources in AWS to build a big data processing system. Which one of the following services would you ideally use for this requirement?

- A.** AWS DynamoDB
- B.** AWS EMR
- C.** AWS ECS
- D.** AWS ECR

QUIZ : QUESTION 2

You have a set of Internet Information Services Servers running on EC2 Instances. You want to collect and process the log files generated from these Servers.

Which of the below services is ideal to run in this scenario?

- A.** Amazon S3 for storing the log files and Amazon EMR for processing the log files.
- B.** Amazon S3 for storing the log files and EC2 Instances for processing the log files.
- C.** Amazon EC2 for storing and processing the log files.
- D.** Amazon DynamoDB to store the logs and EC2 for running custom log analysis scripts.

QUIZ : QUESTION 3

Amazon EMR also allows you to run multiple versions concurrently, allowing you to control the version upgrade of which of the following tools ?

- A. Pig
- B. Windows Server
- C. Hive
- D. Ubuntu

QUIZ : QUESTION 4

Which of the following are the components of Spark ?

- A. Spark Streaming
- B. GraphX
- C. YARN
- D. Mesos

QUIZ : QUESTION 5

Which of the following characteristics differentiates a Task Node from a Core Mode ?

- A. Node Manager Daemon runs on the Task Node.
- B. Task Nodes are optional.
- C. Task Nodes are used for extra capacity when additional CPU and RAM are needed.
- D. Resource Manager runs on the Task Node.

QUIZ : QUESTION 6

Which of the following scenarios Spark should be avoided ?

- A. For interactive analytics
- B. In multi-user environments with high concurrency
- C. For ETL workloads
- D. For batch processing

QUIZ : QUESTION 7

Hive running on a EMR Cluster can read data from which of the below sources ?

- A. HDFS
- B. Local EC2 File System
- C. S3
- D. Kinesis Stream

QUIZ : QUESTION 8

Which of the below Opensource Web interface provides you with a easy way to run scripts, manage the Hive metastore, and view HDFS?

- A. Zeppelin
- B. Ganglia
- C. YARN Resource Manager
- D. Hue

QUIZ : QUESTION 9

Presto is a Relational Database Management System with a SQL Engine.
State TRUE or FALSE ?

A. TRUE

B. FALSE

QUIZ : QUESTION 10

The Data stored on the EBS Volume will persist even after the EMR Cluster is Terminated.
State TRUE or FALSE ?

A. TRUE

B. FALSE

Questions

