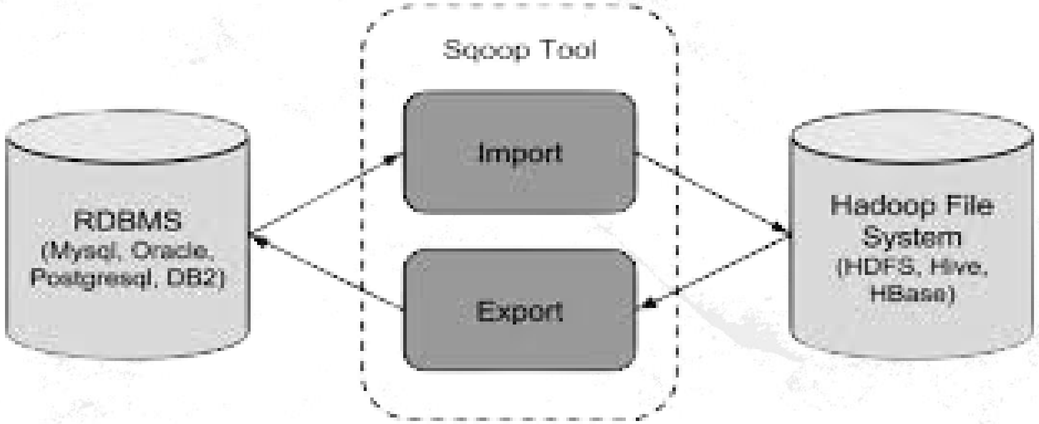


Apache Sqoop

NIELIT Chandigarh/Ropar

"Big Data is at the foundation of all the megatrends that are happening today."

Apache Sqoop Overview

- **Apache Sqoop**:
  - A tool for **transferring data** between **relational databases** (MySQL, Oracle, etc.) and **Hadoop**.
- **Features**:
  - Imports data into HDFS or Hive from relational databases.
  - Exports data back to relational databases.
  - Highly fault-tolerant and scalable.
- **Use Case**:
  - Data Importing from MySQL to HDFS Using Sqoop
  - Migrate data from a MySQL database to Hive for analysis.

Apache Sqoop

Introduction:

- Sqoop is designed to efficiently transfer data between relational databases and Hadoop.
- Useful for importing and exporting structured data to/from HDFS and relational databases like MySQL, Oracle, etc.

Key Features:

- Supports bulk data transfer with optimized performance. Incremental data load functionality.
- Ability to generate Java code for map-reduce jobs.

Apache Sqoop

- **Goal**: Import data from MySQL into HDFS using Sqoop.

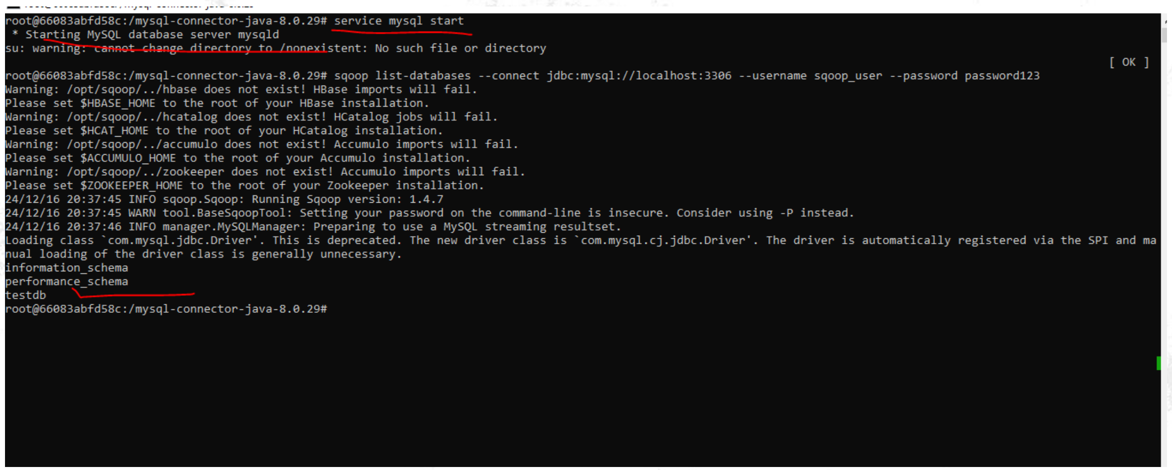
1. **Set up**:
    1. Start Hadoop and Sqoop services.
    2. Create a MySQL database with a sample table (employees).
2. **Commands**:

```
sqoop import --connect
jdbc:mysql://localhost:3306/testdb --username
sqoop_user --password password123 --table employees -
-target-dir /user/hdfs/employees_data
```

Test Sqoop with MySQL

- Use Sqoop to List MySQL Databases (make sure service mysql start is running):

```
sqoop list-databases \
  --connect jdbc:mysql://localhost:3306 \
  --username sqoop_user --password password123
```



```
root@66083abfd58c:/mysql-connector-java-8.0.29# service mysql start
 * Starting MySQL database server mysqld
su: warning: cannot change directory to /nonexistent: No such file or directory
                                                                                                    [ OK ]
root@66083abfd58c:/mysql-connector-java-8.0.29# sqoop list-databases --connect jdbc:mysql://localhost:3306 --username sqoop_user --password password123
Warning: /opt/sqoop/../hbase does not exist! HBase imports will fail.
Please set $HBASE_HOME to the root of your HBase installation.
Warning: /opt/sqoop/../hcatalog does not exist! HCatalog jobs will fail.
Please set $HCAT_HOME to the root of your HCatalog installation.
Warning: /opt/sqoop/../accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
Warning: /opt/sqoop/../zookeeper does not exist! Accumulo imports will fail.
Please set $ZOOKEEPER_HOME to the root of your Zookeeper installation.
24/12/16 20:37:45 INFO sqoop.Sqoop: Running Sqoop version: 1.4.7
24/12/16 20:37:45 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.
24/12/16 20:37:46 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
Loading class `com.mysql.jdbc.Driver'. This is deprecated. The new driver class is `com.mysql.cj.jdbc.Driver'. The driver is automatically registered via the SPI and ma
nual loading of the driver class is generally unnecessary.
information_schema
performance_schema
testdb
root@66083abfd58c:/mysql-connector-java-8.0.29#
```

Apache Sqoop

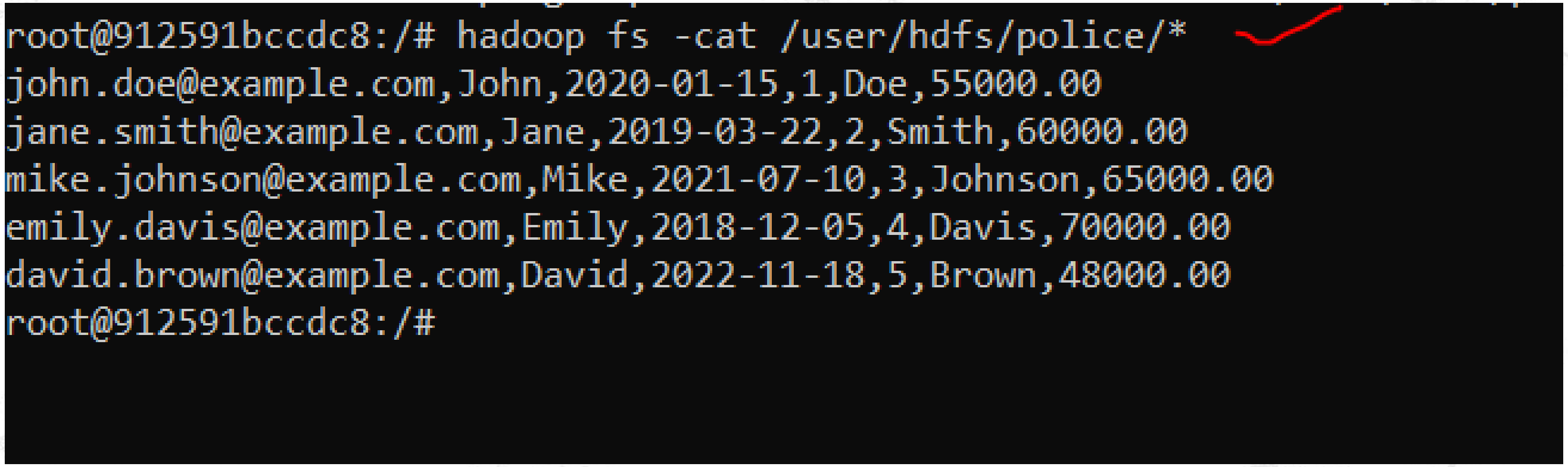
- Export to HDFS

```
sqoop export --connect
jdbc:mysql://localhost:3306/testdb \

--username sqoop_user --password password123 \

--table employees --export-dir
/user/hdfs/employees_data
```

View in HDFS

- This will show the files created in HDFS with the data from the employees table.

```
hadoop fs -cat /user/hdfs/employees_data/*
```



```
root@912591bccdc8:/# hadoop fs -cat /user/hdfs/police/*
john.doe@example.com,John,2020-01-15,1,Doe,55000.00
jane.smith@example.com,Jane,2019-03-22,2,Smith,60000.00
mike.johnson@example.com,Mike,2021-07-10,3,Johnson,65000.00
emily.davis@example.com,Emily,2018-12-05,4,Davis,70000.00
david.brown@example.com,David,2022-11-18,5,Brown,48000.00
root@912591bccdc8:/#
```

Import Data from MySQL to Hive

- Command:

```
sqoop import --connect jdbc:mysql://localhost:3306/testdb \ --
username sqoop_user --password password123 \ --table employees
--hive-import \ --hive-database default \ --hive-table
employees \ --create-hive-table \ --delete-target-dir
```

- **Explanation:**

1. **--hive-import:** Directly imports data into Hive.
2. **--hive-database:** Specifies the Hive database (default is used here).
3. **--hive-table:** Defines the table name in Hive (employees).
4. **--create-hive-table:** Automatically creates the Hive table if it doesn't exist.
5. **--delete-target-dir:** Deletes any existing data in the target directory before import.

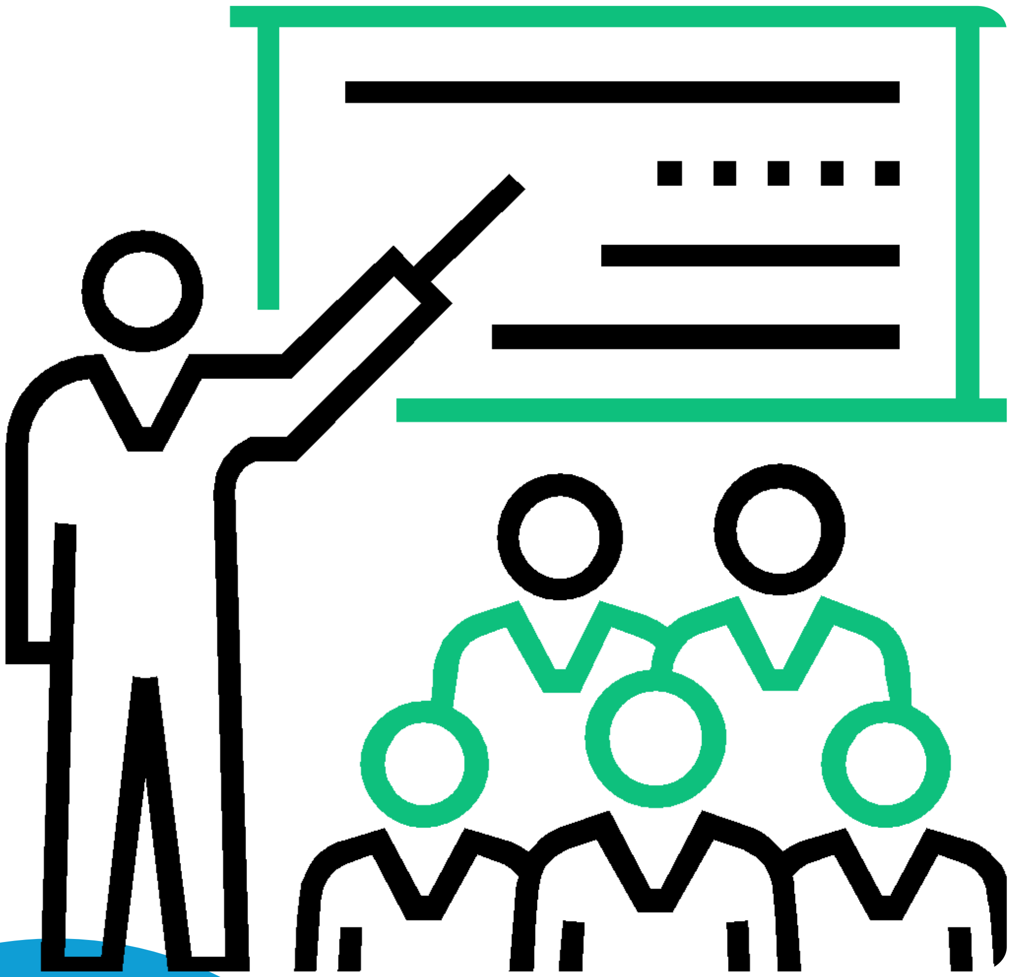
Validate Data in Hive

- **Steps to Verify:**

1. Open the Hive shell:bash
2. hive
3. Use the database and query the table:sql
4. USE default; SELECT * FROM employees;
5. Confirm the imported data matches the MySQL tab

Output:

- The employees table in Hive should now contain the data imported from MySQL.

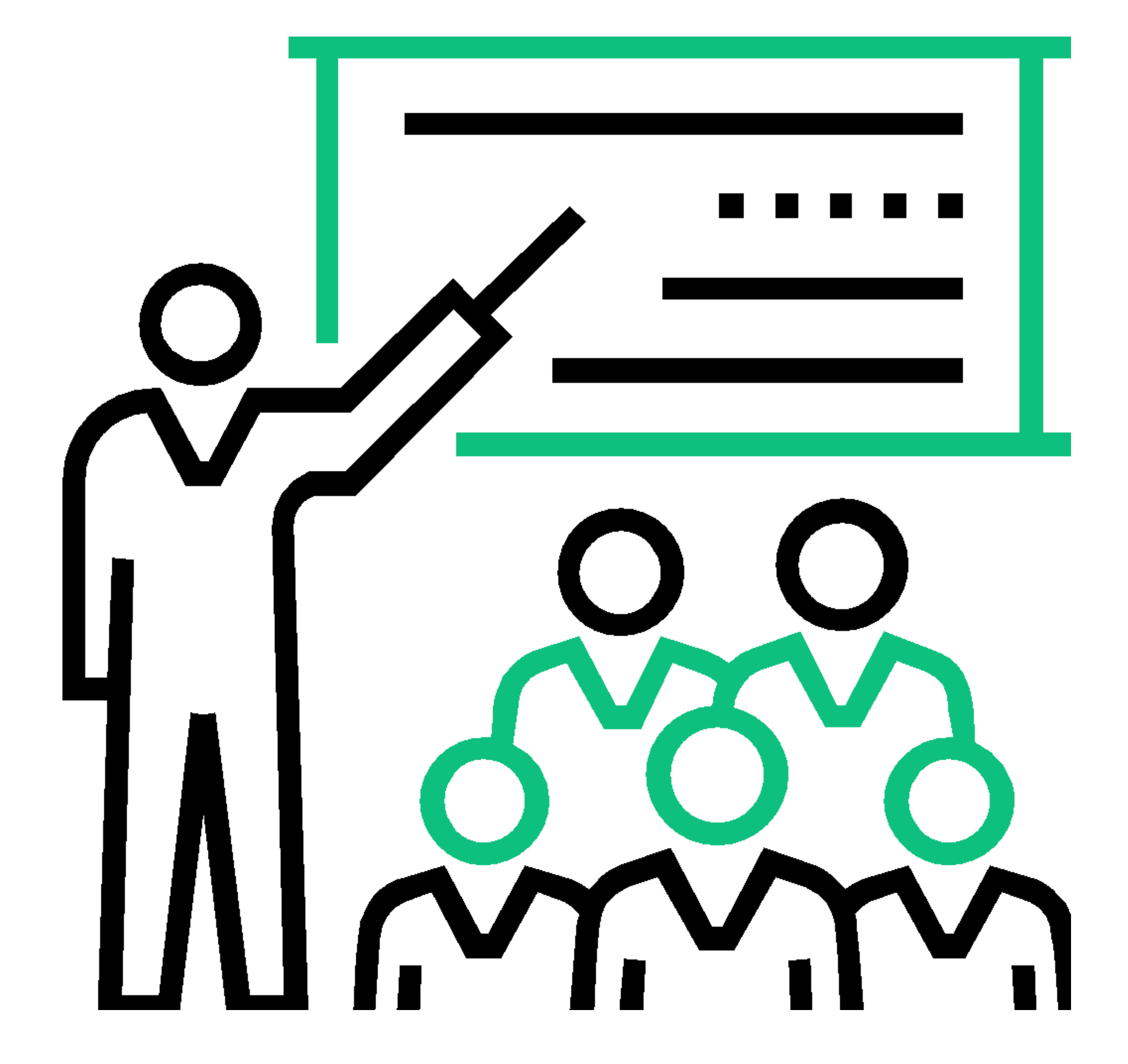


Module Objectives

- At the end of this module, you will be able to:
  - Provide an overview of Apache Sqoop and the use cases where Sqoop-based ingestion is applicable
  - Explain the important features of Sqoop
  - Define the process of installing and running Sqoop
  - Describe Sqoop architecture and its components
  - Explain the process of Sqoop import and export
  - Discuss the process of Sqoop job creation and execution
  - Define file-based ingestion
  - Describe the process of Ingesting and processing batch and streaming data

Module Topics

- Let us take a quick look at the topics that we will cover in this module:
  - Sqoop-based Ingestion – Use Cases
  - Sqoop Architecture
  - Sqoop Import and Export
  - Sqoop Job Creation
  - File Based Ingestion
  - Ingesting Batch vs Streaming Data



An Introduction to Apache Sqoop

Following are the key details of Apache Sqoop:

- ⇥ Sqoop is an open-source product of the Apache software foundation.
- ⇥ Sqoop is used to extract data from a structured data store into Hadoop for further processing, which is done using MapReduce or other high-level tools like Hive.
- ⇥ Sqoop can import and export data efficiently from relational data sources like MySQL and data stores like HDFS and vice versa.
- ⇥ Sqoop was initially developed to transfer data from RDBMS to Hadoop, i.e., SQL to Hadoop, the project release is referred to as Sqoop 1.*. Sqoop 2 is under development and is intended to enable data transfer across any two sources.
- ⇥ Sqoop 1 is the current stable release series.

Use Cases of Sqoop-Based Ingestion

Sqoop supports many RDBMS and is not limited to MySQL. Some of the use cases of Sqoop-based ingestion are as follows:

- ELT and ETL
- Data analysis
- Data archival
- Data consolidation
- Reporting

Major Features of Sqoop

The key features of Sqoop are as follows:

- Parallel import/export
- Connectors for all major RDBMS
- Importing SQL query results
- Incremental load
- Complete load
- Kerberos integration
- Direct data load to Hive/HBase
- Compression

What did You Grasp?

- State True or False.
  Sqoop enables importing only the whole database, but not in parts.
  - **True**
  - **False**

The statement is **False**.

Sqoop allows you to import specific tables or parts of a database, not just the whole database. You can specify individual tables, filter the data with SQL queries, or use partitioning to import subsets of data from a database.

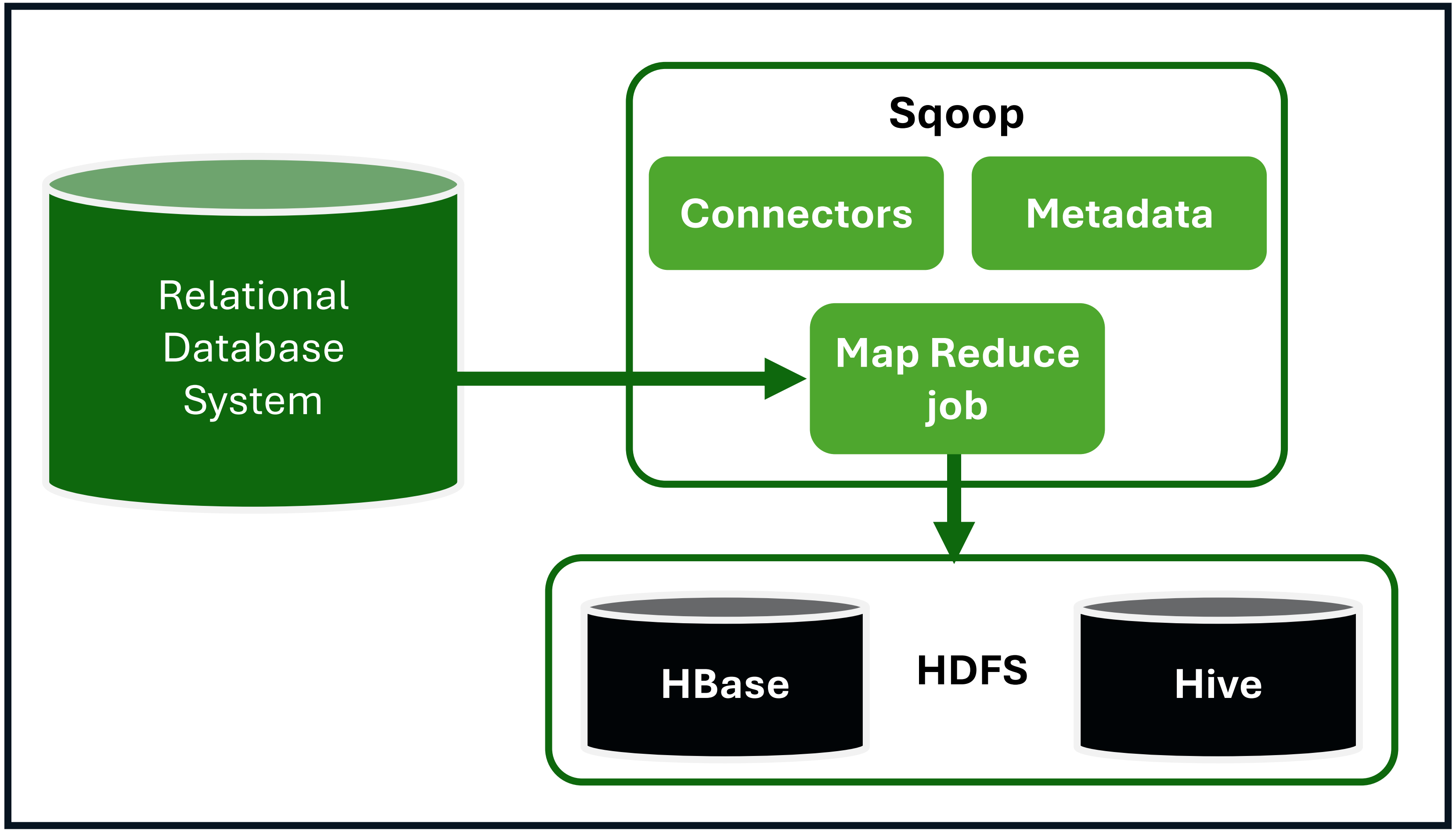
Sqoop Installation

Following are the details for the installation of Sqoop:

- Sqoop is available for download from the Apache Software Foundation's website.
- Sqoop can be downloaded from the following link - http://mirrors.wuchna.com/apachemirror/sqoop/1.4.7.
- The repository contains the complete instructions for compiling the project.
- From the `SQOOP_HOME`, the directory in which Sqoop will get installed, Sqoop is run using the executable script `$SQOOP_HOME/bin/sqoop`.
- If Sqoop is installed from any vendor, it will be placed in a standard location such as /usr/bin/sqoop. Sqoop can then be run from the command line using the command `sqoop`.

Sqoop Architecture

The following picture shows the architecture of Hadoop and the data flow between databases and Sqoop.



What did You Grasp?

- Which of the following databases for which Sqoop doesn’t have a built-in connector?
  - **MySQL**
  - **Oracle**
  - **MongoDB**
  - **Netezza**

The correct answer is MongoDB.Sqoop has built-in connectors for databases like MySQL, Oracle, and Netezza. However, MongoDB is a NoSQL database, and while Sqoop has support for exporting data to MongoDB, it does not have a built-in connector for MongoDB in the same way it does for relational databases like MySQL, Oracle, or Netezza.To work with MongoDB, Sqoop uses a separate connector or third-party tools, but it doesn't have a direct built-in connector as it does for traditional relational databases.

Sqoop Import

Let’s discuss about the Sqoop import process:

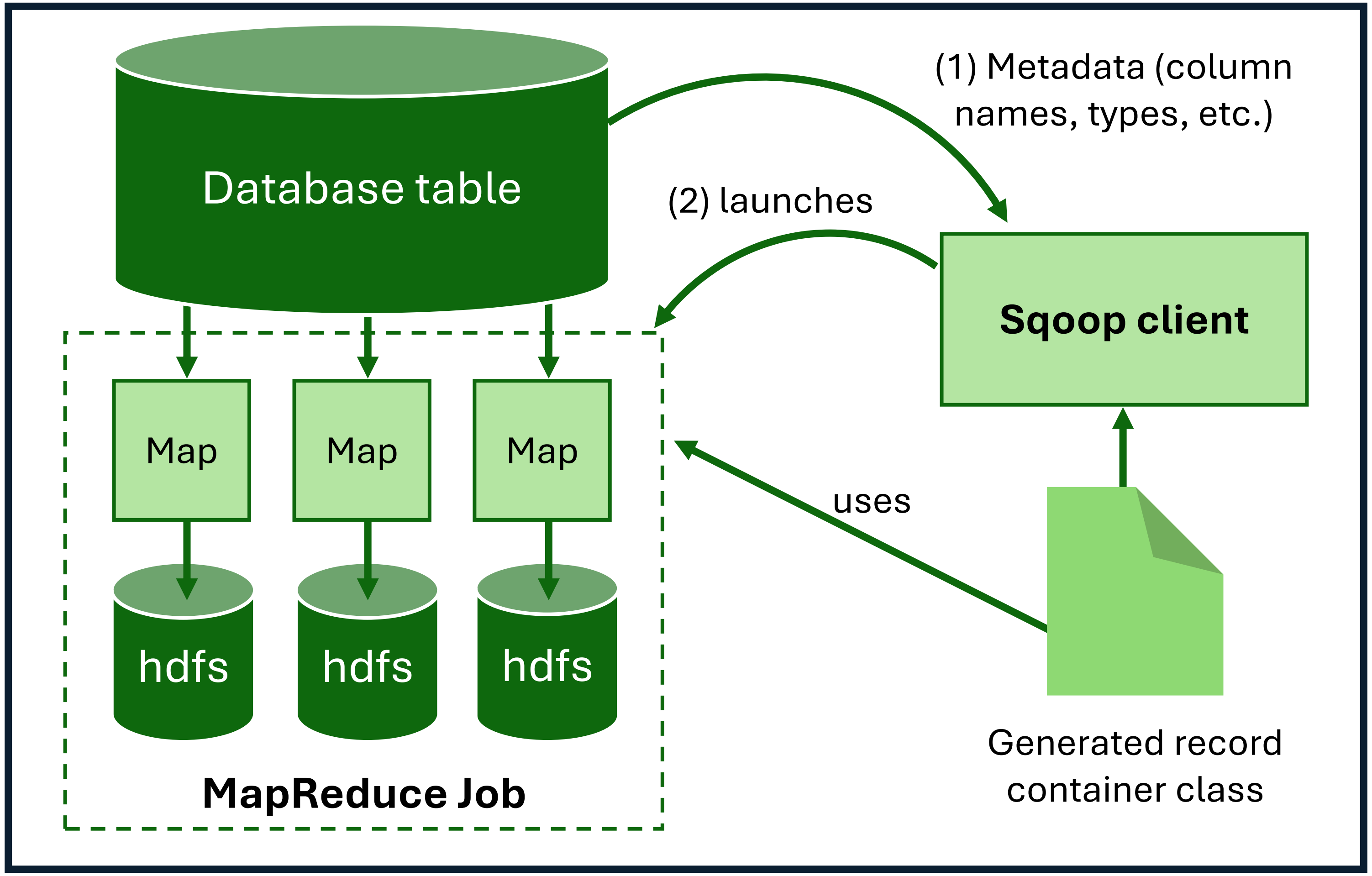
- Sqoop Import tool is used to import data into Hadoop. This tool imports individual tables from RDBMS to HDFS.
- The import tool imports tables from RDBMS to HDFS, where individual rows are considered as records in HDFS. Records are stored as text in text files and binary in Avro or Sequence files.
- A single Sqoop import command can import data from different data sources.

There are different ways in which data can be imported to HDFS, such as follows:

- Bulk data import
- Incremental import
- Free-form query import

Sqoop Import Process

The picture shows the Sqoop import process.



Sqoop Import Syntax

Some of the scenarios of Sqoop import are as follows:

- Importing a table from RDBMS into HDFS
- Importing all the tables in a database into HDFS
- Importing table data into a specific directory in HDFS
- Importing specific table data into HDFS
- Importing table data as a Sequence file into HDFS
- Importing table data as an Avro file into HDFS
- Incremental imports in Sqoop

Sqoop Incremental Imports

The salient features of UDFs are as follows:

Incremental imports are done to make sure that the data imported into HDFS is in sync with the data stored in the database and is constantly updated.

- There are two common modes of incremental imports in Sqoop, which are as follows:
  - *append*
  - *lastmodified*
- The argument `-incremental` can be used to specify the mode of incremental import.
- When the append mode is specified while importing the table, new rows will get added continuously with increasing row ids.
- The lastmodified mode is used when rows of the source table are updated, and each update will set the value of the column last modified to the current timestamp.

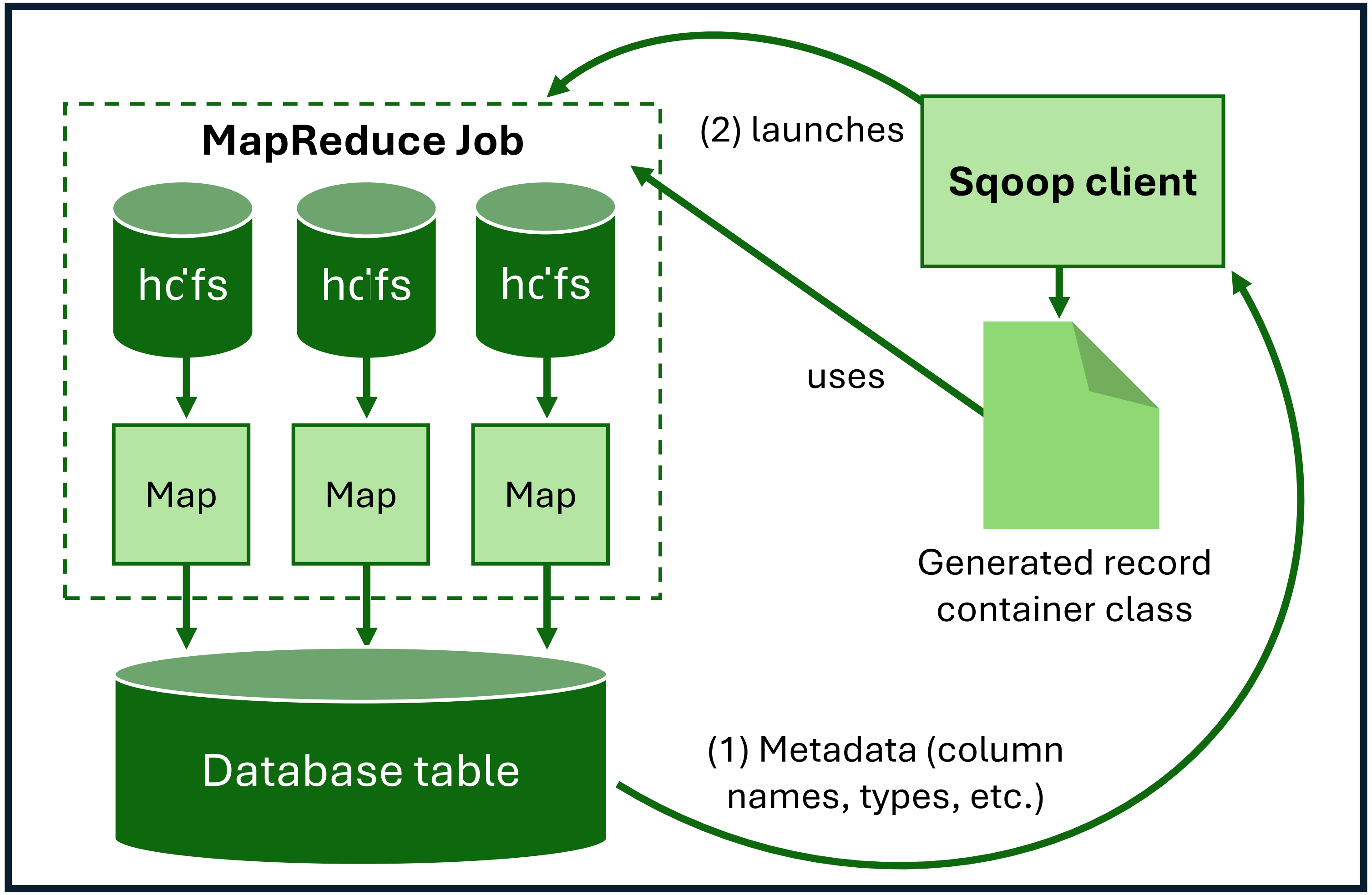
Sqoop Export

Let’s discuss about the Sqoop export process:

- The export option in Sqoop takes data from HDFS and export it to the remote database.
- Data imported into HDFS is analyzed using tools like Hive. The analyzed data is then exported back to the database for use by other tools.
- A target table has to be created in the database to which the data from HDFS has to be exported.
- Once the `export` command is run, the MySQL database can be checked if it has received the data from the export.
- Exports may fail due to a number of reasons.

Sqoop Export Process

The picture shows the Sqoop export process.



Sqoop Export Syntax

Following is the process of Sqoop export with the syntax:

- In general, input files from HDFS are transformed into a set of INSERT statements that injects data into the database.
- In update mode, Sqoop will generate UPDATE statements for replacing existing records in the database, with the ‘call mode’ Sqoop will make a stored procedure call for each record.
- The general syntax for Sqoop export is:

```
$ sqoop export (generic-args) (export-args)
```

- Hadoop generic arguments must precede the export arguments, and the export arguments can be specified in any order.
- The database needs to have the target table created in order for the data in HDFS to be exported.

What did You Grasp?

- Which of the following statements is true about Sqoop export?
  - **During export data will be placed in the database where target table is created automatically**
  - **Data will be exported to the databases using INSERT statements**
  - **Memory capacity doesn’t directly impact Sqoop export**
  - **Sqoop directly parses data in text files after the export command is run**

The correct statement is:

"Data will be exported to the databases using INSERT statements."

Explanation of each option:

"During export, data will be placed in the database where target table is created automatically": This is false. The target table must already exist in the database before performing an export operation in Sqoop. Sqoop does not automatically create the target table.

"Data will be exported to the databases using INSERT statements": This is true. When exporting data, Sqoop uses **INSERT** statements to insert data into the target table in the database.

"Memory capacity doesn’t directly impact Sqoop export": This is false. Memory capacity does affect Sqoop export, especially when dealing with large datasets, as sufficient memory is required for the job to execute efficiently.

"Sqoop directly parses data in text files after the export command is run": This is false. Sqoop typically exports data from Hadoop to a database. If the data is in text files, it first needs to be loaded into Hadoop (e.g., HDFS) before being exported. Sqoop doesn’t directly parse text files during the export process.

Therefore, the true statement is that **data will be exported to the databases using INSERT statements**.

Sqoop Job Creation

Let's discuss about the Sqoop job creation:

- In order to understand Sqoop jobs, we need to understand saved jobs. Saved jobs are used to perform imports and exports repeatedly. With Sqoop we can save jobs, which make imports and exports easier.
- Sqoop job tool is used to create and work with the saved jobs. The job specifies the parameters used to identify and recall the saved job.
- Recall and re-execution is used in incremental imports. There are four common tasks associated with Sqoop job tool. They are as follows:
  - Create
  - Verify
  - Inspect
  - Execute

Sqoop Job Operations

Some of the job operations and the commands are given in the table below:

Argument	Description
-create <job-id>	Define a new saved job with the specified job-id (name). A second Sqoop command-line, separated by a -- should be specified; this defines the saved job.
--delete <job-id>	Delete a saved job
--exec <job-id>	Given a job defined with --create, run the saved job
--show <job-id>	Show the parameters for a saved job
--list	List and verify all saved jobs

What did You Grasp?

- Which of the following arguments is used to verify the Sqoop jobs?
  - **--show**
  - **--verify**
  - **--list**
  - **--check**

The correct argument to verify Sqoop jobs is --check.This argument is used to verify the integrity of a Sqoop job before executing it, ensuring that the job will run correctly.Here’s a brief explanation of the other options:--show: Displays the details of a specific Sqoop job.--verify: This option does not exist in Sqoop.--list: Lists all the available Sqoop jobs, but doesn't verify them.So, to verify a Sqoop job, use --check

File-based Ingestion

Following are the key details of file-based ingestion:

- Hadoop data ingestion is the first step of the data pipeline in the data lake. Hadoop uses a distributed file system for reading and writing files.
- When data is written to HDFS, it is sliced into pieces and replicated across the servers in a Hadoop cluster.
- Multiple slices can be processed in parallel for enabling faster computation and when files are moved out of HDFS, the slices are integrated and written as one file on the host file system.
- Ingesting a file is a simple process, where all the file is imported to Hadoop or Data lake, then loaded into a landing server and then used Hadoop CLI to ingest the data.

Ingestion of Batch and Streaming Data

Let’s discuss about batch and streaming data:

- Data ingestion is the process in which data from external sources are moved to a data lake.
- Data can be ingested into the big data system in batches at regular intervals or real-time data can be ingested directly.
- This distinction is based on whether the new data is ingested for processing as it arrives or stored for some time and ingested at a later time for processing.
- Batch data is ingested using tools like Sqoop, whereas streaming data can be ingested real-time using tools like Kafka.

Batch Processing

Following are the key details of batch processing:

- In batch processing, newly arriving data is collected and stored into groups. Batch processing is ideal in cases where the up-to-date data is not important.
- The complete group is then processed as a batch in a future time. The time at which it is processed can be determined by factors like scheduled time or based on some conditions.
- The majority of traditional data processing technologies are designed to do batch processing. Data warehouses and Sqoop are a common examples of batch processing system.
- In cases where the batches are small or processed at small batches, the process is termed as micro-batch processing.

Stream Processing

Following are the key details of stream processing:

- In stream processing, data is ingested and processed as it arrives. Stream processing is ideal in cases where data is critical and requires real-time response.
- In stream processing, data is processed as individual pieces rather than being processed in groups.
- Apache Kafka, Storm, Samza and Heron are excellent examples of stream processing tools real-time or near real-time.
- Stream processing is useful for tasks like fraud detection, analysing health data from a critically ill patient, analysing traffic monitoring data, etc.

What did You Grasp?

- Which of the following tools is used for batch processing?
  - **Kafka**
  - **Storm**
  - **Heron**
  - **Sqoop**

The correct answer is **Sqoop**.

Sqoop is primarily used for **batch processing** of large amounts of data between relational databases and Hadoop ecosystems (like HDFS, Hive, etc.).

Here's a breakdown of the other tools:

Kafka: A distributed streaming platform, used for real-time data streaming and messaging, not batch processing.

Storm: A real-time computation system, used for processing streams of data in real time.

Heron: A real-time stream processing engine, a successor of Storm.

So, **Sqoop** is the one used for batch processing.

In a nutshell, we learnt:

- Introduction to Apache Sqoop and the use cases where Sqoop-based ingestion is applicable
- Important features of Sqoop
- Process of installing and running Sqoop
- Sqoop architecture and the components
- Sqoop import and export
- Sqoop job creation and execution
- File-based ingestion
- Ingesting and processing batch and streaming data

Apache Sqoop Setup and Project Creation (Step-by-Step Guide First need to Install MySql or Import From Hive)