hadoop

sqoop

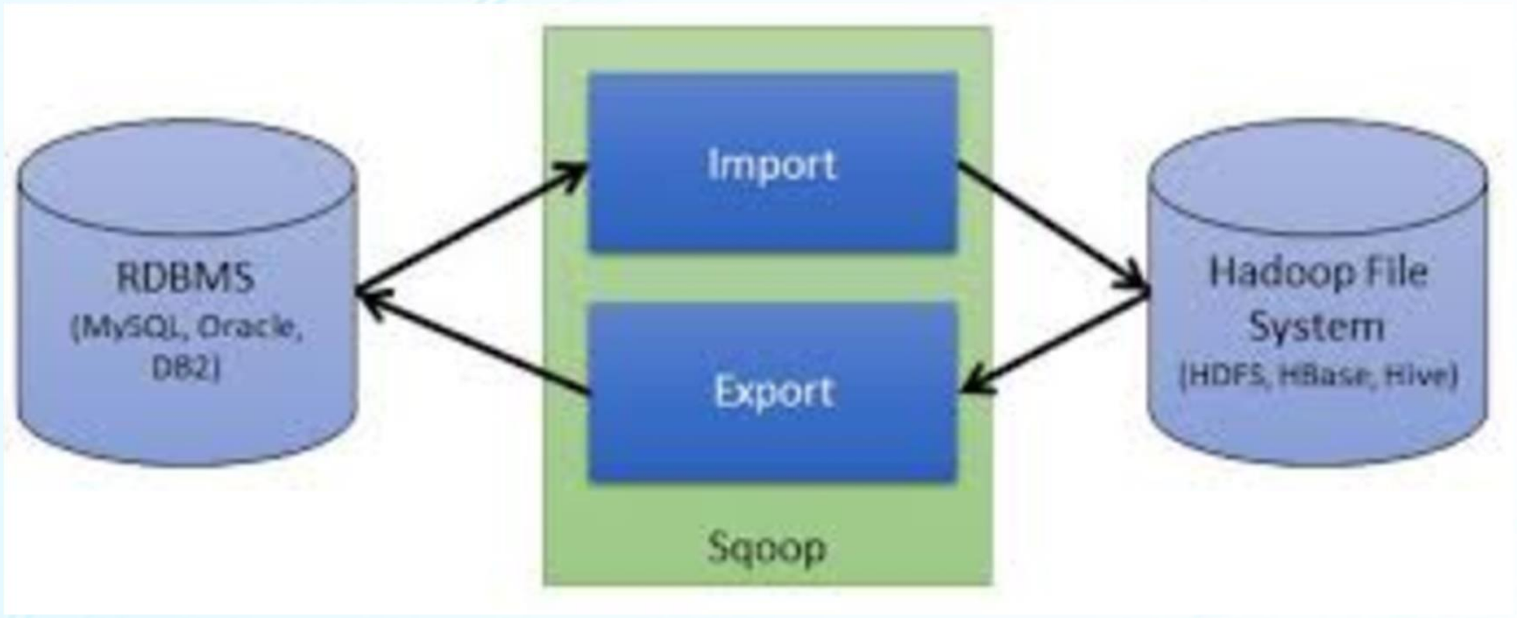
Apache SQOOP

Agenda

- ✓ What is Sqoop
- ✓ Sqoop Architecture
- ✓ Sqoop Commands & Help
- ✓ Sqoop Import
- ✓ Sqoop Export
- ✓ Other Tools in Sqoop

What is Sqoop ?

Apache Sqoop is a tool designed for efficiently transferring bulk data between Hadoop and structured data stores such as relational databases.



What is Sqoop ?

- Sqoop is a Data Ingestion tool and it deals with structured data

- Sqoop allows easy import and export of data from structured data stores such as relational databases, enterprise data warehouses, and NoSQL systems.

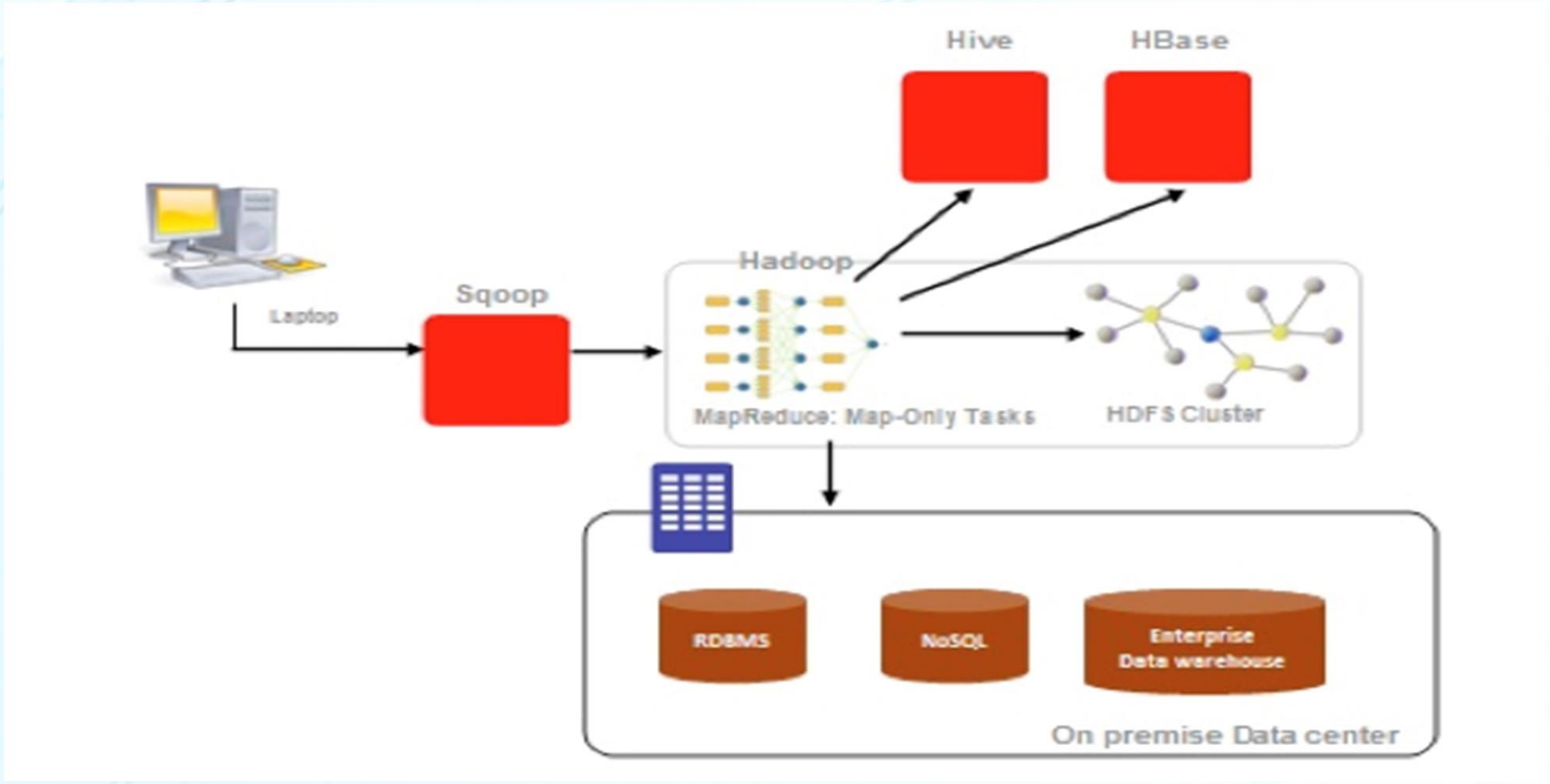
- Imports individual tables or entire databases from external database to HDFS.

- Exports data from HDFS to external data sources.

- Sqoop internally generates MapReduce code to transfer the data.

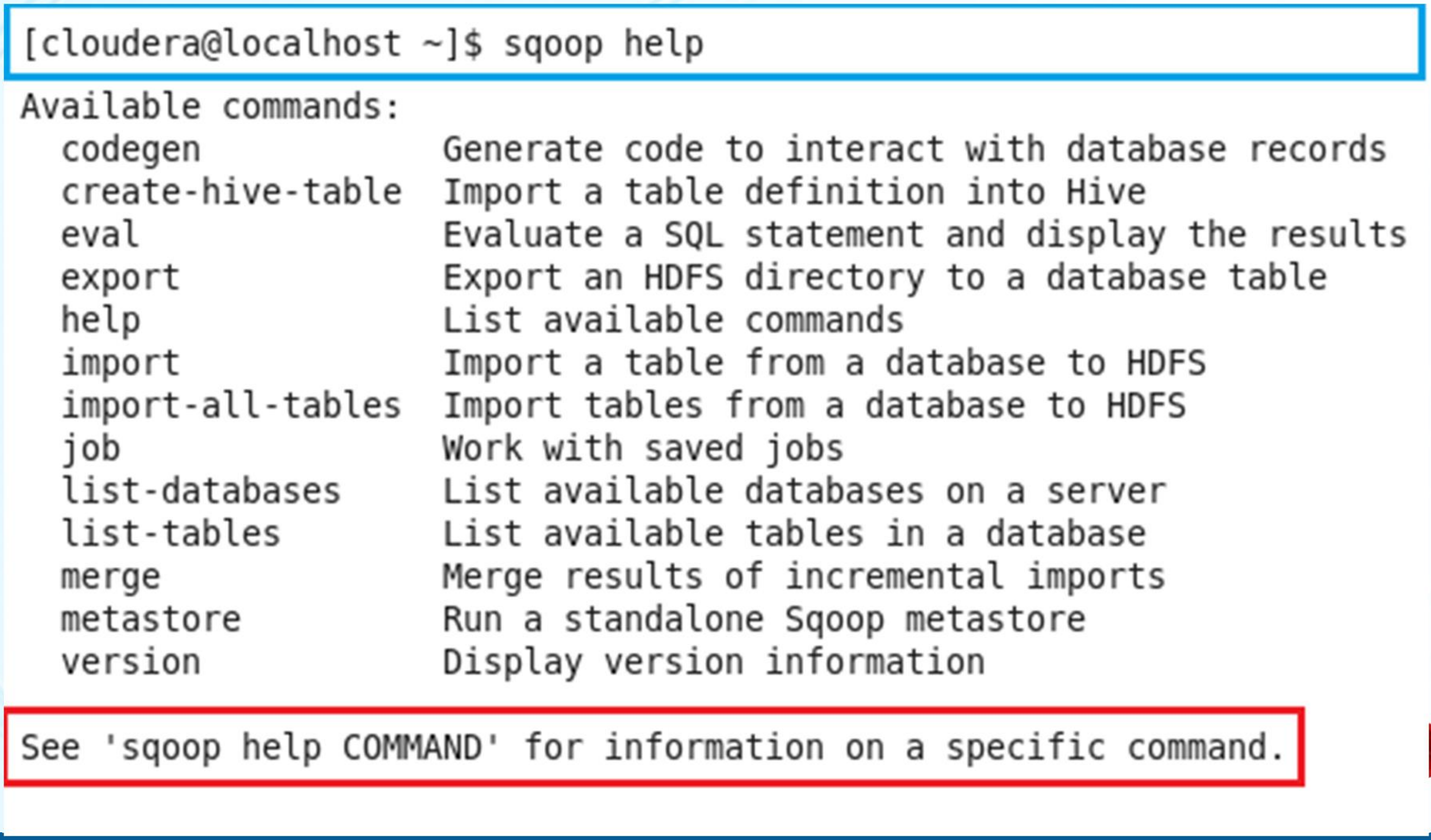
- The dataset being transferred is sliced up into different partitions and a map-only job is launched with individual mappers responsible for transferring a slice of this dataset.

Understanding Sqoop



Sqoop Tools

Apache Sqoop is organized as a set of tools/commands.



```
[cloudera@localhost ~]$ sqoop help
Available commands:
  codegen            Generate code to interact with database records
  create-hive-table  Import a table definition into Hive
  eval               Evaluate a SQL statement and display the results
  export             Export an HDFS directory to a database table
  help               List available commands
  import             Import a table from a database to HDFS
  import-all-tables  Import tables from a database to HDFS
  job                Work with saved jobs
  list-databases     List available databases on a server
  list-tables        List available tables in a database
  merge              Merge results of incremental imports
  metastore          Run a standalone Sqoop metastore
  version            Display version information

See 'sqoop help COMMAND' for information on a specific command.
```

Sqoop Commands

Commands	Description
codegen	Generate code to interact with database records
create-hive-table	Import a table definition into Hive
eval	Evaluate a SQL statement and display the results
export	Export an HDFS directory to a database table
help	List available commands
import	Import a table from a database to HDFS
import-all-tables	Import tables from a database to HDFS
list-databases	List available databases on a server
list-tables	List available tables in a database
merge	Merge results of incremental imports
metastore	Run a standalone Sqoop metastore
version	Display version information

Sqoop help command

```
$sqoop help <command-name>

Example:  $sqoop help import
```

The help tool provides specific usage instructions on a particular tool, by providing that tool’s name as an argument

List Databases & Tables

```
sqoop list-databases --connect jdbc:mysql://localhost:3306
```

```
sqoop list-tables --connect jdbc:mysql://localhost:3306/empdb
```

Sqoop Import

Usage: `sqoop import [GENERIC-ARGS] [TOOL-ARGS]`

Common Arguments

--connect <jdbc-url>	Specify JDBC connect string
--driver <class-name>	Manually specify JDBC driver class to use
--hadoop-home <dir>	Override $HADOOP_HOME
--help	Print usage instructions
--P	Read password from console
--password <password>	Set authentication password
--username <username>	Set authentication username
--verbose	Print more information while working

Sqoop Import

```
sqoop import
      --connect jdbc:mysql://localhost:3306/empdb
      --username xxx
      --password xxx
      --table emp
      -m 3
      --target-dir /user/cloudera/empdb/emp
```

Note: Before import, grant privileges to the database in MySQL.

```
mysql> GRANT ALL ON <dbname>.* TO '%'@'localhost';
mysql> GRANT ALL ON <dbname>.* to ‘’@’localhost’;
```

Sqoop Import

```
sqoop import
        --connect jdbc:mysql://localhost:3306/empdb
        --username xxx
        --password xxx
        --table emp
        -m 3
        --target-dir /user/cloudera/empdb/emp
        --fields-terminated-by '\t'
```

- Default field terminator is comma (,)

Sqoop Import

```
sqoop import
      --connect jdbc:mysql://localhost:3306/empdb
      --username xxx
      --password xxx
      --table emp
      --columns "id,name,age"
      -m 3
      --target-dir /user/cloudera/empdb/emp
      --fields-terminated-by '\t'
```

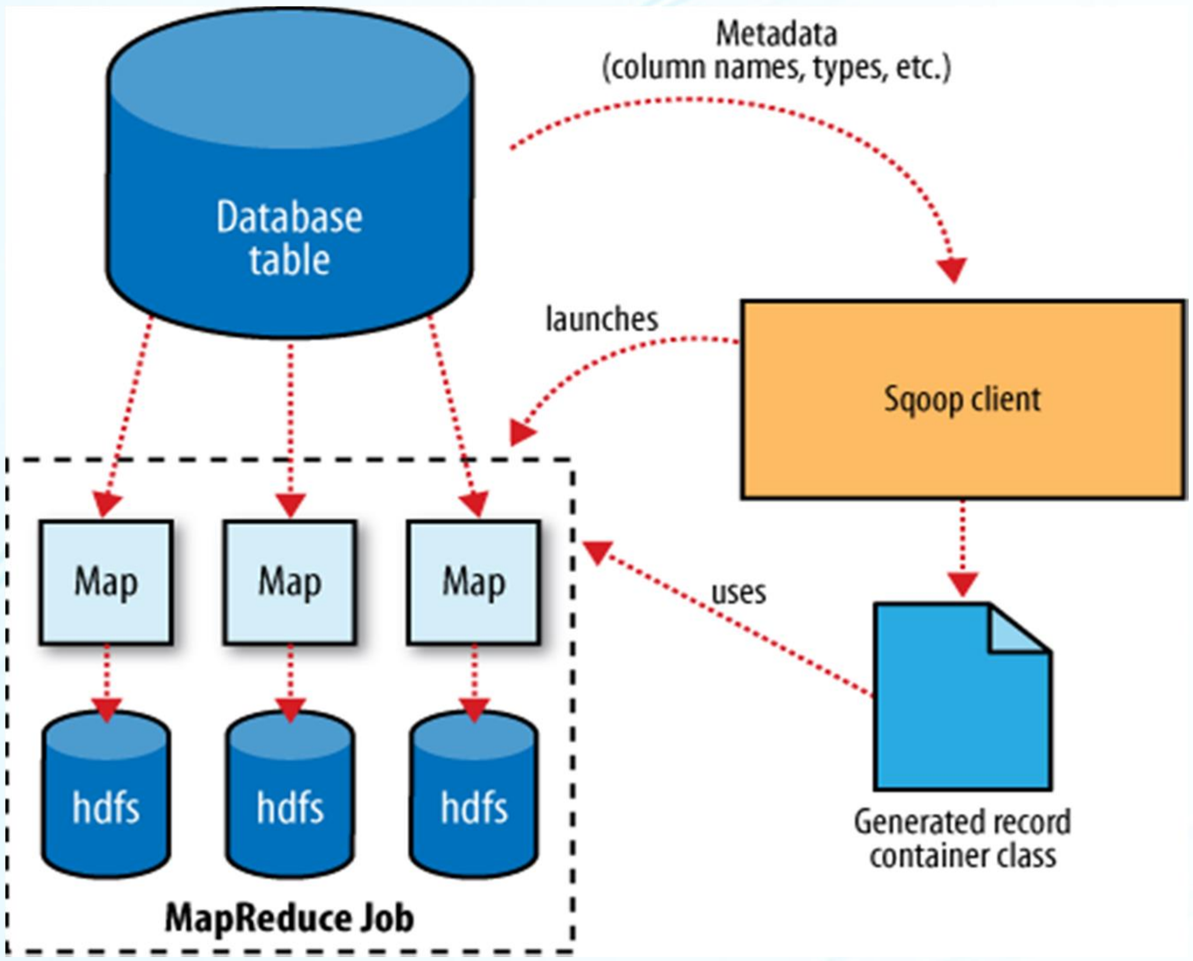
- Import specific columns only

Sqoop Import

```
  sqoop import
        --connect jdbc:mysql://localhost:3306/empdb
        --username xxx
        --password xxx
        --query “select id, name from emp where age < 50
AND \$CONDITIONS"
        -m 3
        --target-dir /user/cloudera/empdb/emp
        --fields-terminated-by ‘\t’
```

- `AND \$CONDITIONS` is required if you have enclosed your query in double quotes. Otherwise use `AND $CONDITIONS`

Sqoop Import Process



Sqoop Export

- In Sqoop, an *import* refers to the movement of data from a database system into HDFS. By contrast, an *export* uses HDFS as the source of data and a remote database as the destination.

- Before exporting a table from HDFS to a database, we must prepare the database to receive the data by creating the target table.

- While Sqoop can infer which Java types are appropriate to hold SQL data types, this translation does not work in both directions (for example, there are several possible SQL column definitions that can hold data in a Java String; this could be CHAR(64), VARCHAR(200), or something else entirely). Consequently, you must determine which types are most appropriate.

Sqoop Export

```
sqoop export
      --connect jdbc:mysql://localhost:3306/empdb
      -m 1
      --username root
      --password  xxxx
      --table emp
      --export-dir /user/cloudera/empdb/emp
      --input-fields-terminated-by ',`
```

The export tool exports a set of files from HDFS back to an RDBMS. The target table must already exist in the database. The input files are read and parsed into a set of records according to the user-specified delimiters.

eval

```
sqoop eval
      --connect jdbc:mysql://localhost:3306/empdb
      --query "select * from emp"
```

```
sqoop eval
      --connect jdbc:mysql://localhost:3306/empdb
      --query “update emp set name=‘raju’ where id=1"
```

```
sqoop eval
      --connect jdbc:mysql://localhost:3306/empdb
      --query “create table dept (did int, dname varchar(30))"
```

codegen

Generates Java code for a specified table import

```
sqoop codegen
     --connect jdbc:mysql://localhost:3306/empdb
     --table emp
     --class-name Employee
```

Loook for the file (Employee.java) in your home directory (Linux) from where you executed sqoop codegen

Importing to Hive

Import a MySQL table into Hive

```
sqoop import
      --connect jdbc:mysql://localhost:3306/empdb
      --username cloudera
      --table emp
      --hive-table hivedb.hiveemp
      -m 1
      --fields-terminated-by ','
      --hive-import
```

Importing to Hive

Import a MySQL table definition (i.e with out any data) into Hive

```
sqoop create-hive-table
      --connect jdbc:mysql://localhost:3306/empdb
      --username root
      --table emp
      --fields-terminated-by ','
```

THANK YOU