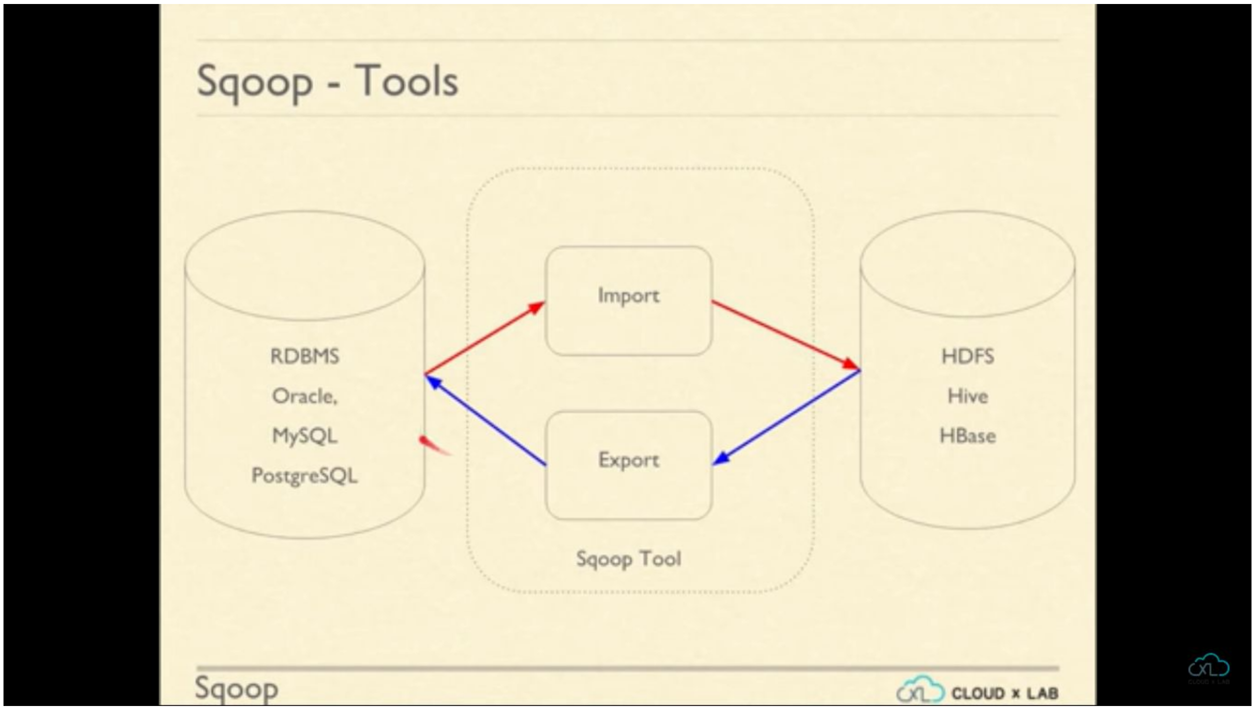
Introduction

Open source tool to efficiently transfer bulk data between Relational databases and hadoop.



Sqoop - Tools

RDBMS
Oracle,
MySQL
PostgreSQL

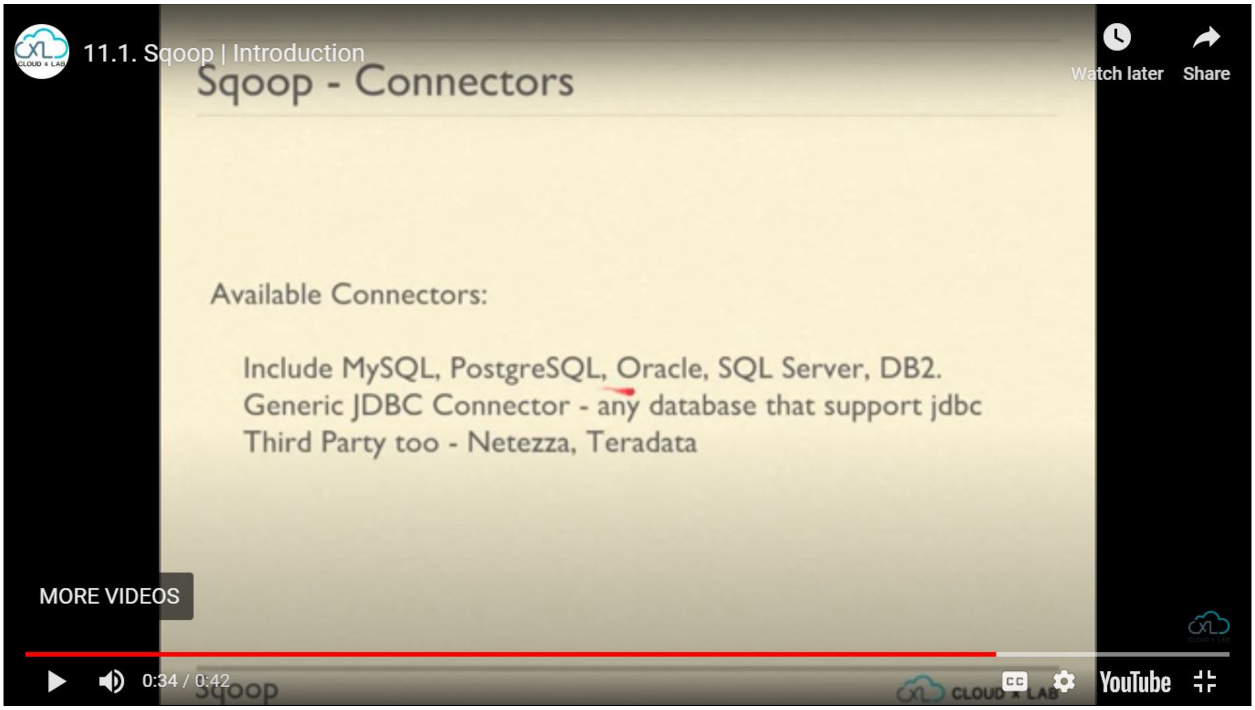
Import

Export

Sqoop Tool

HDFS
Hive
HBase

Sqoop



Sqoop - Connectors

Available Connectors:

Include MySQL, PostgreSQL, Oracle, SQL Server, DB2.
Generic JDBC Connector - any database that support jdbc
Third Party too - Netezza, Teradata

Sqoop

Login to terminal

Type **sqoop help** - see all available commands

Mysql to sqoop

Start MySQL prompt: (you will have to specify password) `mysql -h 10.142.1.2 -u sqoopuser -pNHkkP876rp`

On mysql prompt, take a look at data: `use sqoopex` `select * from widgets`

Delete the folder if exists in HDFS: `hadoop fs -rmr widgets`

Run the following to import: `sqoop import --connect jdbc:mysql://10.142.1.2/sqoopex --table widgets -m 2 --username sqoopuser --password NHkkP876rp --split-by id`

Check the content of the imported File: `hadoop fs -cat widgets/part-m-*`

NOTE: The mysql host used used in video is old. Please use the most recent host listed in "My Lab".

Mysql to hive

```
#Get the detail of MySQL server using "My Lab" tab

# Check the MySQL: Connect
mysql -u sqoopuser -p -h cxln2.c.thelab-240901.internal sqoopex

#Check the MySQL: Explore The table in MySQL using
select * from widgets;

# Import - It might ask for password. Keep the password
sqoop import --connect jdbc:mysql://cxln2.c.thelab-240901.internal/sqoopex --table
widgets -m 2 --hive-import --username sqoopuser -P --hive-database sqoop_testing
--split-by id


#Start hive
use sqoop_testing;
select * from widgets;
```

MySQL to HBase

```
sqoop import --connect jdbc:mysql://10.142.1.2/sqoopex --table widgets
--hbase-table 'sgiri:widgets' --column-family cf2 --username sqoopuser
--hbase-create-table --columns id,widget_name --hbase-row-key id -m 1
--password NHkkP876rp
```

`Type` **`hbase shell`**

`Scan ‘widgets’`

Sqoop Export - Hive to MySQL

Prepare Source

```
# Copy sales.log locally
hadoop fs -cp /data/hive/sales.log .

#Launch hive using command: hive
# Create Hive Table:
use sg;
CREATE TABLE sales_test(widget_id INT, qty INT,
street STRING, city STRING, state STRING,
zip INT, sale_date STRING)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ',';

# Find the location of your table using:
describe formatted sales_test
# Load Data:
LOAD DATA INPATH "sales.log" INTO TABLE sales_test;
# Select rows to see data:
select * from sales_test;
```

Prepare MySQL Table

```
#TO launch mysql:
    mysql -h ip-172-31-20-247 -u sqoopuser -p

#Create MYSQL Table:
use sqoopex;
CREATE TABLE sales_sgiri(widget_id INT, qty INT, street varchar(100), city
varchar(100), state varchar(100),
zip INT, sale_date varchar(100))
```

Sqoop Export - Hive To MySQL

```
# Sqoop Export:
sqoop export --connect jdbc:mysql://ip-172-31-20-247/sqoopex -m 1 --table sales_sgiri
--export-dir /apps/hive/warehouse/sg.db/sales_test --input-fields-terminated-by ','
--username sqoopuser --password NHkkP876rp

# Go back to the mysql prompt and check
```

```
use sqoopex;
select * form sales_sgiri
```