

# Big Data

## Data Loading Tools

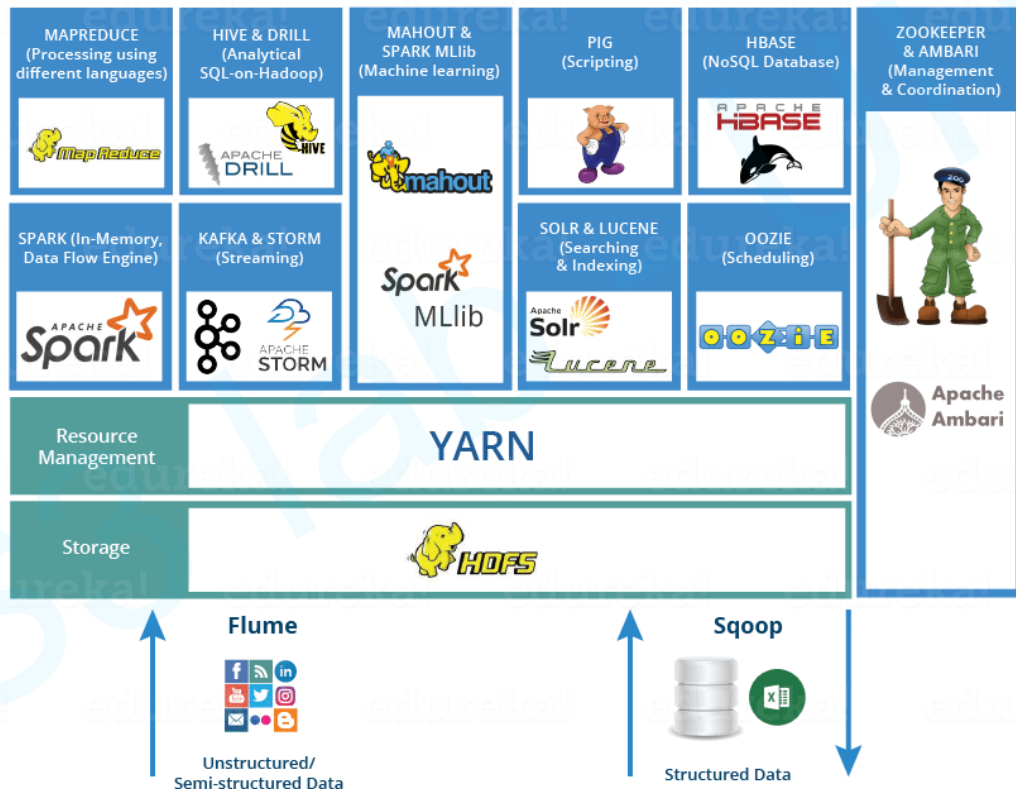
Trong-Hop Do



“Without big data, you are blind and deaf  
and in the middle of a freeway.”

– Geoffrey Moore

# Hadoop Ecosystem



# Apache Flume Tutorial

---

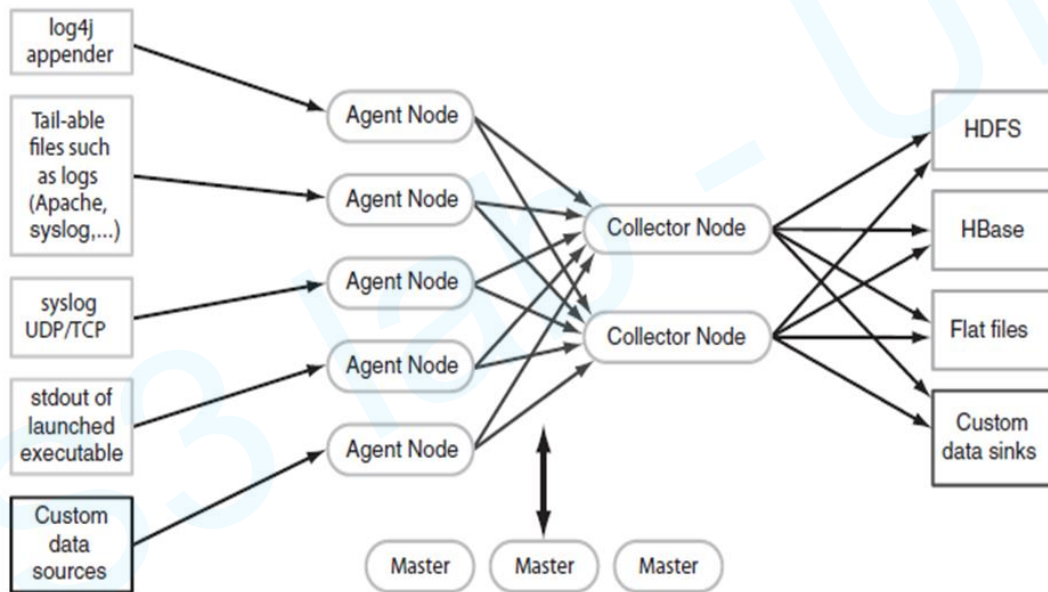
# Introduction to Apache Flume

- Apache Flume is a tool for data ingestion in HDFS. It collects, aggregates and transports large amount of streaming data such as log files, events from various sources like network traffic, social media, email messages etc. to HDFS. Flume is a highly reliable & distributed.
- The main idea behind the Flume's design is to capture streaming data from various web servers to HDFS. It has simple and flexible architecture based on streaming data flows. It is fault-tolerant and provides reliability mechanism for Fault tolerance & failure recovery.



# Data transfer components

## Flume - How it works



**Figure 2.2** Flume architecture for collecting streaming data

# Data transfer components



## *Flume - How it works*

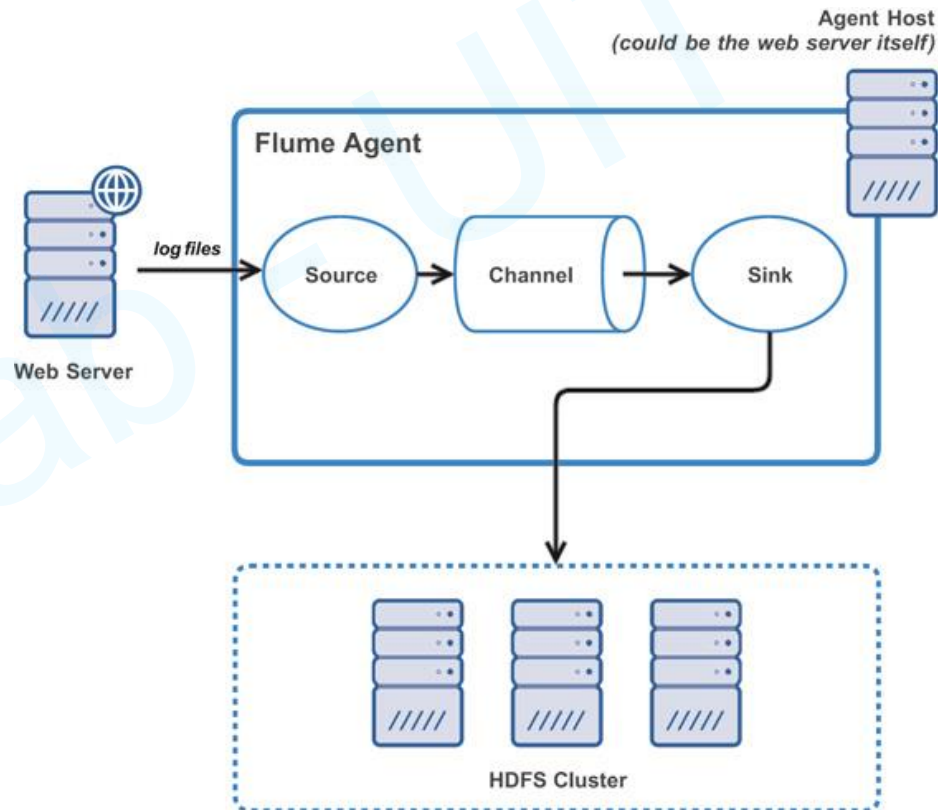
- Data flows like
- Agent tier -> Collector tier -> Storage tier
- **Agent** nodes are typically installed on the machines that generate the logs and are data's initial point of contact with Flume. They forward data to the next tier of **collector** nodes, which aggregate the separate data flows and forward them to the final **storage** tier.



# Data transfer components

## Flume - Agent architecture

- Sources:
  - HTTP, Syslog, JMS, Kafka, Avro, Twitter - stream api for tweets download, ...
- Sink:
  - HDFS, Hive, HBase, Kafka, Solr, ...
- Channel:
  - File, JDBC, Kafka, ...





# Start Flume on Cloudera Quickstart VM

- To add Flume to Cloudera Quickstart VM, you need to launch Cloudera Manager
- Configure the VM.
  - Allocate a minimum of 10023 MB memory.
  - Allocate 2 CPUs.
  - Allocate 20 MB video memory.
  - Consider setting the clipboard to bidirectional.

# Start Flume on Cloudera Quickstart VM

- Launch Cloudera Express

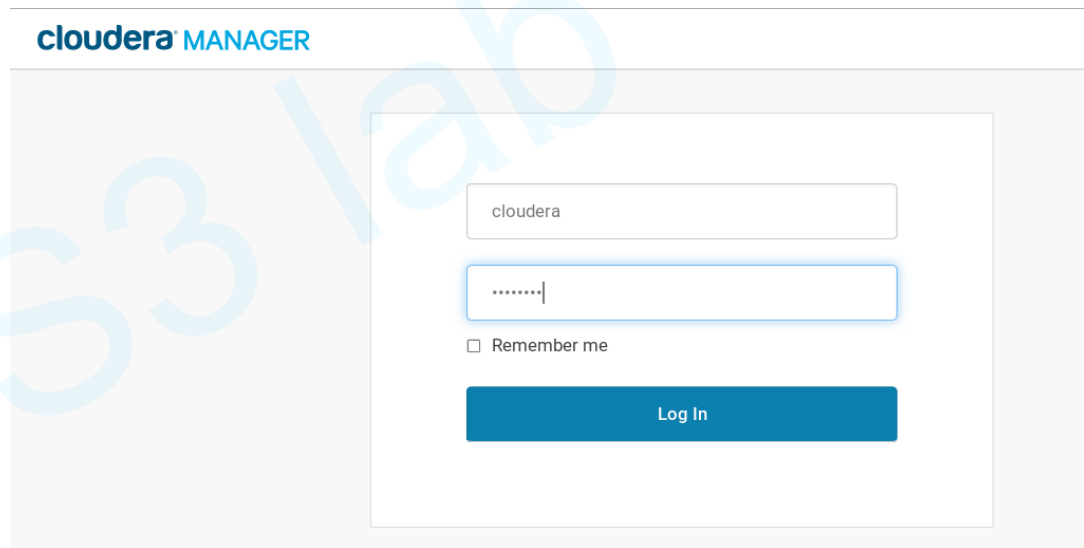


# Start Flume on Cloudera Quickstart VM

- Check the status of Namenode services
  - Command: **sudo service hadoop-hdfs-namenode status**
  - If namenode is **not** running, then start namenode service
  - Command: **sudo service hadoop-hdfs-namenode start**
- Check the status of Namenode services
  - Command: **sudo service hadoop-hdfs-datanode status**
  - If namenode is not running, then start namenode service
  - Command: **sudo service hadoop-hdfs-datanode start**

# Start Flume on Cloudera Quickstart VM

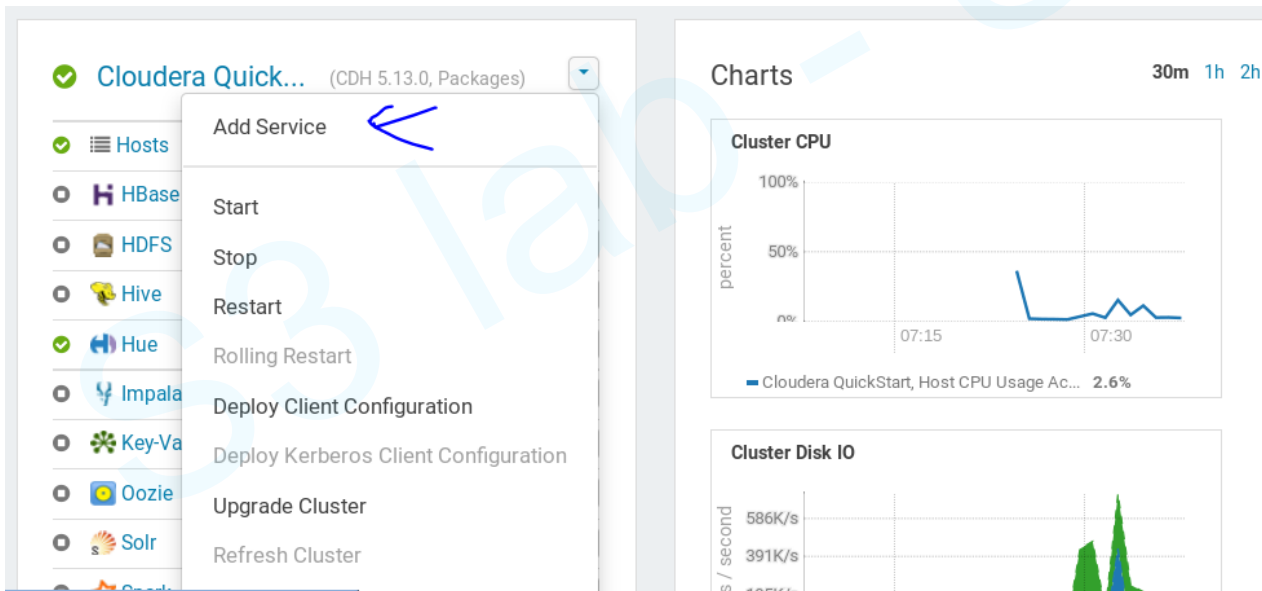
- Open Cloudera Manager in web browser
- Username: **cloudera**
- Password: **cloudera**



The screenshot shows the Cloudera Manager login interface. At the top, the text "cloudera MANAGER" is displayed. Below this, there is a login form with two input fields: the first contains the username "cloudera", and the second contains a masked password ".....". Below the password field is a checkbox labeled "Remember me". At the bottom of the form is a blue button labeled "Log In".

# Start Flume on Cloudera Quickstart VM

- After logging in to Cloudera Manager, click **Add Service**







# Start Flume on Cloudera Quickstart VM

- Select Flume

## Add Service to Cloudera QuickStart

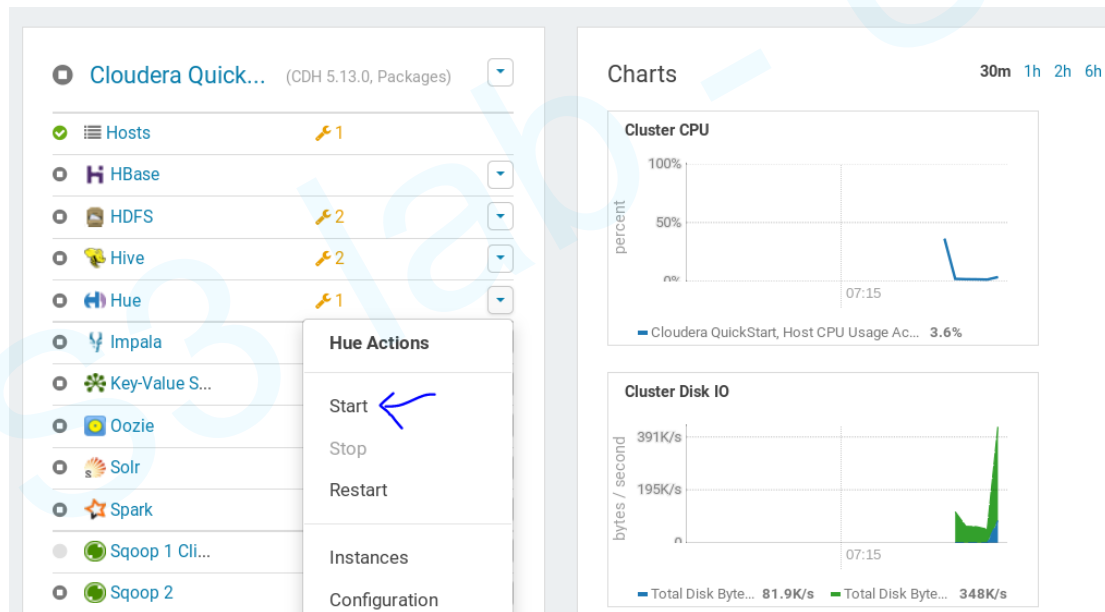
Select the type of service you want to add.

Service Type	Description
<input type="radio"/>  Accumulo	The Apache Accumulo sorted, distributed key/value store is a robust, scalable, high performance data storage and retrieval system. This service only works with releases based on Apache Accumulo 1.6 or later.
<input checked="" type="radio"/>  Flume	Flume collects and aggregates data from almost any source into a persistent store such as HDFS.
<input type="radio"/>  HBase	Apache HBase provides random, real-time, read/write access to large data sets (requires HDFS and ZooKeeper).
<input type="radio"/>  HDFS	Apache Hadoop Distributed File System (HDFS) is the primary storage system used by Hadoop applications. HDFS creates multiple replicas of data blocks and distributes them on compute hosts throughout a cluster to enable reliable, extremely rapid computations.

[Back](#)[Continue](#)

# Start Flume on Cloudera Quickstart VM

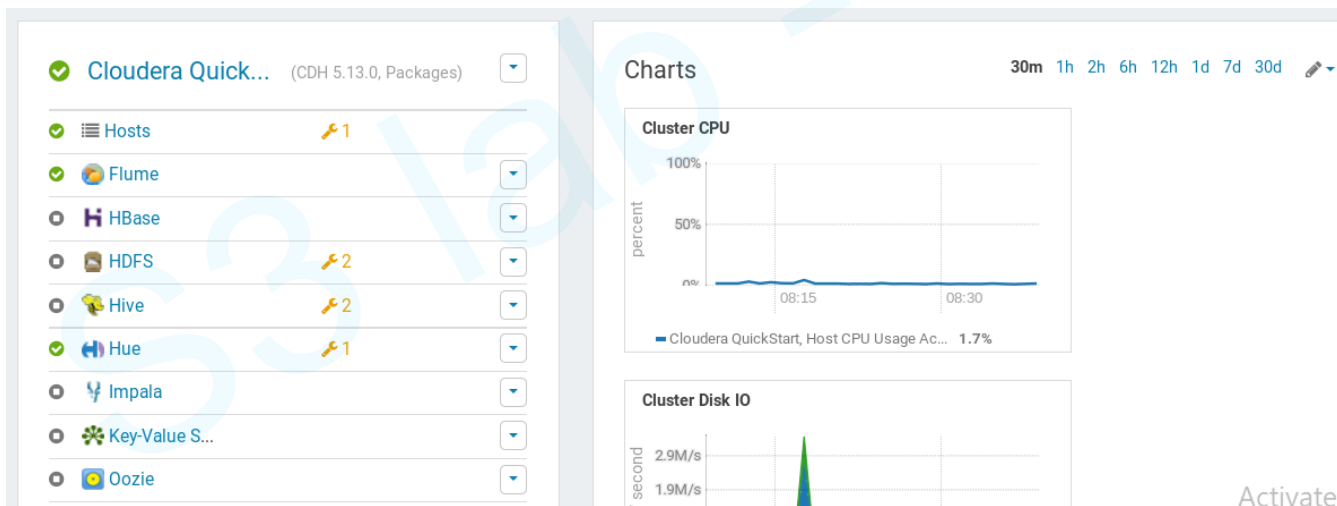
- Start Hue





# Start Flume on Cloudera Quickstart VM

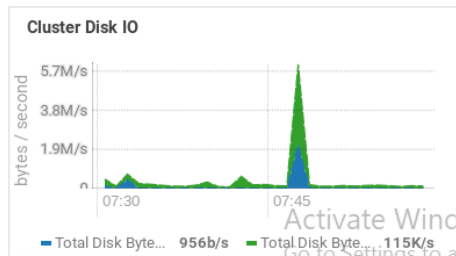
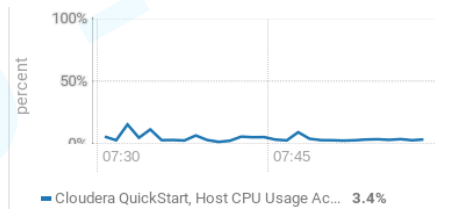
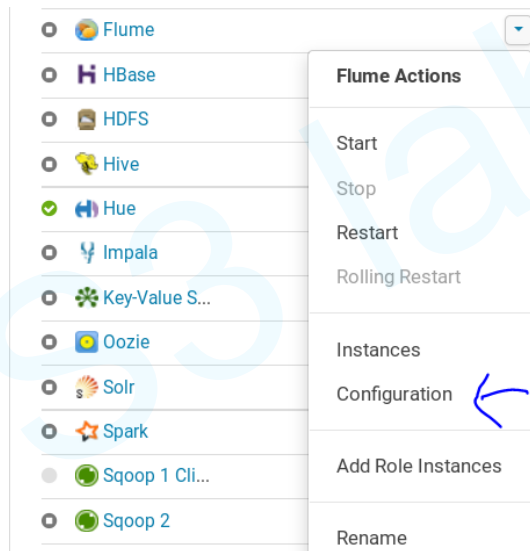
- Start Flume



Activate

# Start Flume on Cloudera Quickstart VM

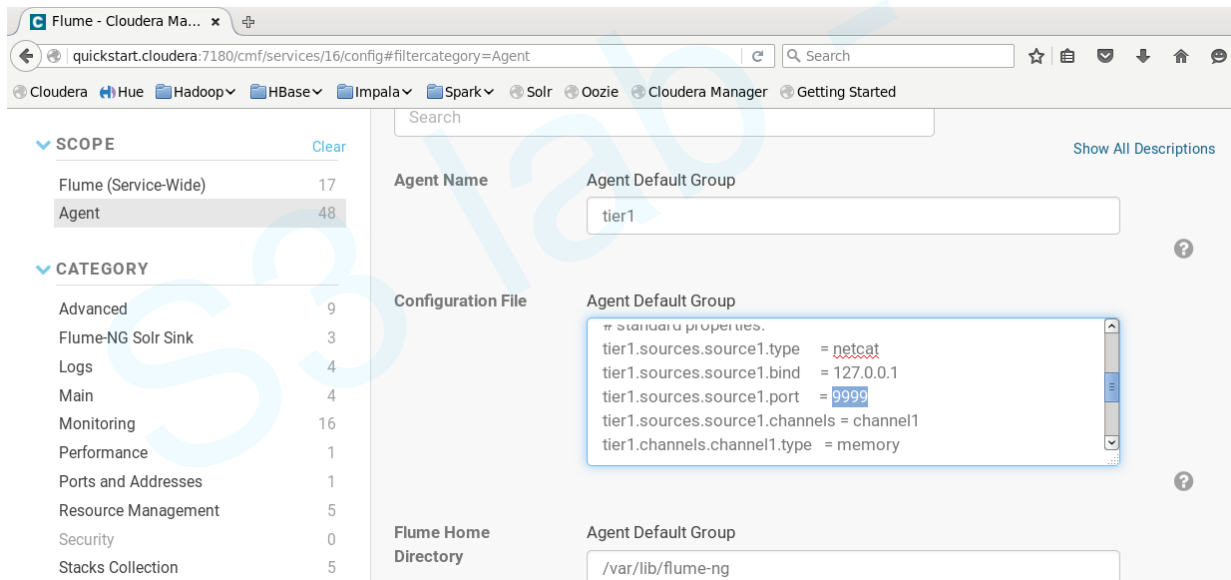
- Check the configuration of Flume



Cluster Network IO

# Start Flume on Cloudera Quickstart VM

- Check the port (**9999** in this VM)



The screenshot shows the Cloudera Manager web interface. The browser address bar displays `quickstart.cloudera:7180/cmf/services/16/config#filtercategory=Agent`. The left sidebar contains a navigation menu with the following items:

- SCOPE
  - Flume (Service-Wide) 17
  - Agent 48
- CATEGORY
  - Advanced 9
  - Flume-NG Solr Sink 3
  - Logs 4
  - Main 4
  - Monitoring 16
  - Performance 1
  - Ports and Addresses 1
  - Resource Management 5
  - Security 0
  - Stacks Collection 5

The main content area shows the configuration for the Agent Default Group. The "Agent Name" field is set to "tier1". The "Configuration File" field is highlighted with a blue border, showing the following configuration:

```
# standard properties.  
tier1.sources.source1.type = netcat  
tier1.sources.source1.bind = 127.0.0.1  
tier1.sources.source1.port = 9999  
tier1.sources.source1.channels = channel1  
tier1.channels.channel1.type = memory
```

The "Flume Home Directory" field is set to `/var/lib/flume-ng`.

# Start Flume on Cloudera Quickstart VM

Use Telnet to test the default Flume implementation

- Firstly, let's install telnet
- Command: **sudo yum install telnet**

```
[cloudera@quickstart ~]$ sudo yum install telnet
Loaded plugins: fastestmirror, security
Setting up Install Process
Determining fastest mirrors
epel/metalink                               | 5.1 kB    00:00
... ..

Installed:
  telnet.x86_64 1:0.17-49.el6_10

Complete!
[cloudera@quickstart ~]$
```

# Start Flume on Cloudera Quickstart VM

Use Telnet to test the default Flume implementation

- Launch Telnet with the command: **telnet localhost 9999**
- At the prompt, enter **Hello world ^.^**
- Press **Ctrl+]** to escape
- Type **quit** to close telnet

```
[cloudera@quickstart ~]$ telnet localhost 9999
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
Hello world ^.^
OK
^]

telnet> quit
Connection closed.
[cloudera@quickstart ~]$
```

# Start Flume on Cloudera Quickstart VM

Use Telnet to test the default Flume implementation

- Check the log
- Command: **cat /var/log/flume-ng/flume-cmf-flume-AGENT-quickstart.cloudera.log**

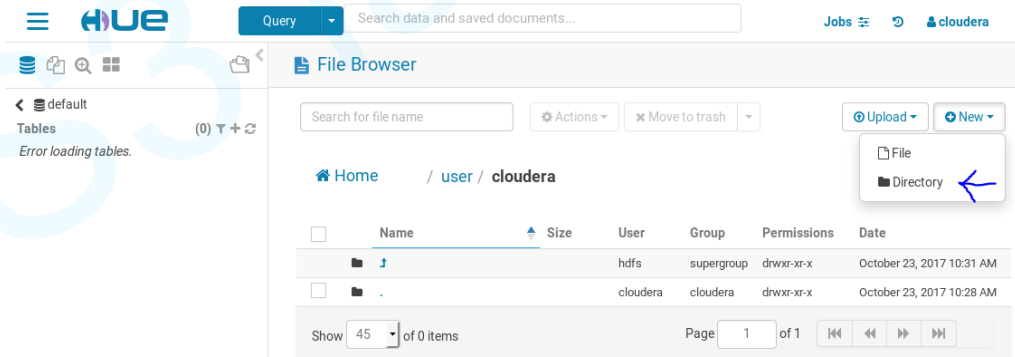
```
[cloudera@quickstart ~]$ cat /var/log/flume-ng/flume-cmf-flume-AGENT-quickstart.cloudera.log
2020-10-01 08:04:33,945 INFO org.apache.flume.node.PollingPropertiesFileConfigurationProvider: Configuration provider starting
2020-10-01 08:04:33,964 INFO org.apache.flume.node.PollingPropertiesFileConfigurationProvider: Reloading configuration file:/var/run/cloudera-scm-agent/process/8-flume-AGENT/flume.conf
2020-10-01 08:04:33,967 INFO org.apache.flume.conf.FlumeConfiguration: Processing:sink1

2020-10-01 08:16:43,778 INFO org.apache.flume.sink.LoggerSink: Event: { headers:{} body: 48 65 6C 6C 6F 20 77 6F 72 6C 64 20 5E 2E 5E 0D Hello world ^.^. }
[cloudera@quickstart ~]$
```

# Writing from Flume to HDFS

Create the /flume/events directory

- In the VM web browser, open Hue
- Click File Browser
- In the **/user/cloudera** directory, click New->Directory
- Create a directory named **flume**

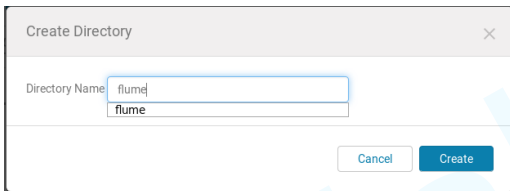




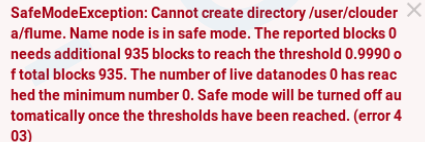
# Writing from Flume to HDFS

Create the /flume/events directory

- If you get this error when creating new directory

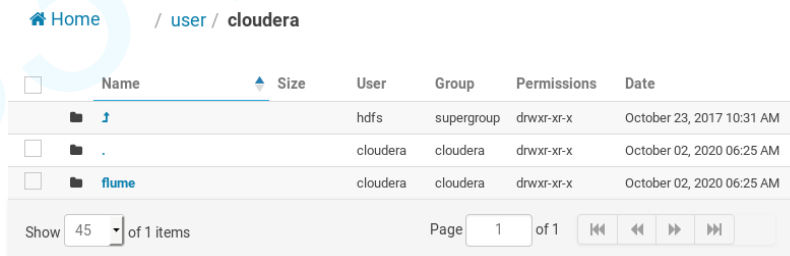


A dialog box titled "Create Directory" with a close button (X) in the top right corner. It contains a text input field labeled "Directory Name" with the text "flume" entered. Below the input field is a "Create" button and a "Cancel" button.



SafeModeException: Cannot create directory /user/cloudera/flume. Name node is in safe mode. The reported blocks 0 needs additional 935 blocks to reach the threshold 0.9990 of total blocks 935. The number of live datanodes 0 has reached the minimum number 0. Safe mode will be turned off automatically once the thresholds have been reached. (error 403)

- Then run command: `sudo -u hdfs hdfs dfsadmin -safemode leave`



A screenshot of a web-based file browser showing the directory structure of a HDFS cluster. The breadcrumb path is "/ user / cloudera". The table lists the following directories:

	Name	Size	User	Group	Permissions	Date
<input type="checkbox"/>	↑		hdfs	supergroup	drwxr-xr-x	October 23, 2017 10:31 AM
<input type="checkbox"/>	.		cloudera	cloudera	drwxr-xr-x	October 02, 2020 06:25 AM
<input type="checkbox"/>	flume		cloudera	cloudera	drwxr-xr-x	October 02, 2020 06:25 AM




At the bottom, it shows "Show 45 of 1 items" and "Page 1 of 1" with navigation buttons.

# Writing from Flume to HDFS

Create the /flume/events directory





- In the flume directory, create a directory named **events**

[Home](#) / [user](#) / [cloudera](#) / **flume**

<input type="checkbox"/>	Name	Size	User	Group	Permissions	Date
<input type="checkbox"/>	 <a href="#">↑</a>		cloudera	cloudera	drwxr-xr-x	October 02, 2020 06:25 AM
<input type="checkbox"/>	 <a href="#">.</a>		cloudera	cloudera	drwxr-xr-x	October 02, 2020 06:26 AM
<input type="checkbox"/>	 <a href="#">events</a>		cloudera	cloudera	drwxr-xr-x	October 02, 2020 06:26 AM

Show  of 1 items

Page  of 1




   

# Writing from Flume to HDFS

Create the /flume/events directory

- Check the box to the left of the events directory, then click the Permissions setting

[Home](#) / [user](#) / [cloudera](#) / **flume**

<input type="checkbox"/>	Name	Size	User	Group	Permissions	Date
<input type="checkbox"/>	 <a href="#">↑</a>		cloudera	cloudera	drwxr-xr-x	October 02, 2020 06:25 AM
<input type="checkbox"/>	 <a href="#">.</a>		cloudera	cloudera	drwxr-xr-x	October 02, 2020 06:26 AM
<input checked="" type="checkbox"/>	 <a href="#">events</a>		cloudera	cloudera	<b>drwxr-xr-x</b>	October 02, 2020 06:26 AM

Show  of 1 items

Page  of 1

[⏪](#) [⏴](#) [⏵](#) [⏩](#)

# Writing from Flume to HDFS

Create the /flume/events directory

- Enable Write access for Group and Other users
- Then click Submit

Change Permissions

	User	Group	Other
Read	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Write	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Execute	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Sticky			<input type="checkbox"/>
Recursive			<input type="checkbox"/>

Cancel

Submit

# Writing from Flume to HDFS

## Change the Flume configuration

- Open Cloudera Manager -> click Flume -> Click the Configuration tab
- Scroll or search for the **Configuration File** item.
- Append the following lines to the Configuration File settings
- Click **Save Changes**

```
tier1.sinks.sink1.type = HDFS  
tier1.sinks.sink1.filetype = DataStream  
tier1.sinks.sink1.channel = channel1  
tier1.sinks.sink1.hdfs.path =  
hdfs://localhost:8020/user/cloudera/flume/events
```

Configuration File

Agent Default Group

```
tier1.sinks.sink1.type= HDFS  
tier1.sinks.sink1.fileType=DataStream  
tier1.sinks.sink1.channel = channel1  
tier1.sinks.sink1.hdfs.path = hdfs://localhost:8020/user/cloudera  
/flume/events
```

Flume Home Directory

Agent Default Group

/var/lib/flume-ng

Plugin directories

Agent Default Group

/usr/lib/flume-ng/plugins.d

1 Edited Value Reason for change...

Save Changes

# Writing from Flume to HDFS

Change the Flume configuration

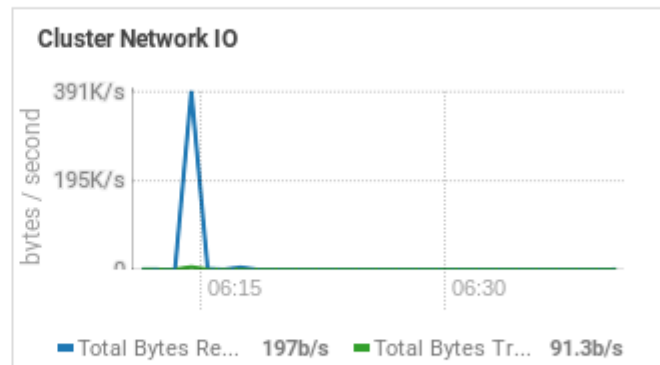
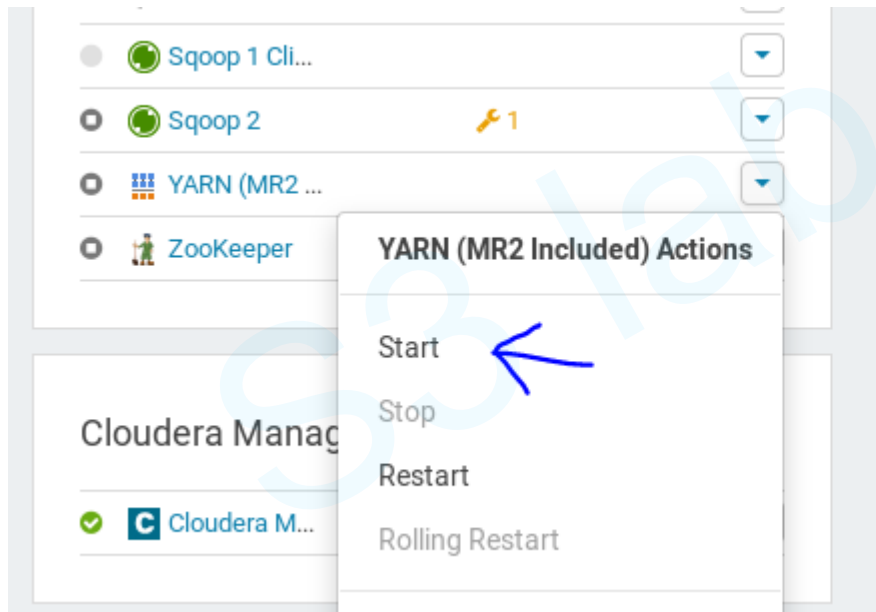
- Restart Flume

The screenshot shows the Cloudera Manager interface for the Flume service. The top navigation bar includes 'Clusters', 'Hosts', 'Diagnostics', 'Audits', 'Charts', and 'Administration'. The main header shows 'Flume (Cloudera QuickStart)' with a green checkmark and a timestamp 'Oct 2, 6:33 AM PDT'. Below the header, there are tabs for 'Status', 'Instances', and 'Configuration'. The 'Configuration' tab is active, and the 'Actions' dropdown menu is open, showing options: 'Start', 'Stop', 'Restart' (highlighted with an orange arrow), 'Rolling Restart', 'Add Role Instances', 'Rename', 'Enter Maintenance Mode', and 'Update Config'. The main content area shows the 'Flume (Service-Wide)' configuration with a radio button selected for 'HDFS' and another for 'none'. A 'Save Changes' button is visible at the bottom right.

# Writing from Flume to HDFS

Change the Flume configuration

- Start YARN



HDFS IO



# Writing from Flume to HDFS

## Writing to HDFS

- In a terminal window, launch Telnet with the command **telnet localhost 9999**
- At the prompt, enter some text

```
[cloudera@quickstart ~]$ telnet localhost 9999
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
Hello HDFS writing from Flume ^.^
OK
^]

telnet> quit
Connection closed.
[cloudera@quickstart ~]$
```

# Writing from Flume to HDFS

## Writing to HDFS

- Hue File Browser, open the /user/cloudera/flume/events directory
- Click the file name **FlumeData.xxxxxx** link to view the data sent by Flume to HDFS

The screenshot shows the Hue File Browser interface. The top navigation bar includes the Hue logo, a 'Query' dropdown, a search bar, and links for 'Jobs', a refresh icon, and the user 'cloudera'. The left sidebar shows a file tree with 'default' selected and a message 'Error loading tables.' The main panel displays the file browser for the path '/ user / cloudera / flume / events / FlumeData.1601646911241'. It includes options to 'View as text', 'Edit file', 'Download', 'View file location', and 'Refresh'. The file content is displayed in a hex dump format, with a yellow arrow pointing to the text 'Hello HDFS writing from Flume'.

```
0000000: 53 45 51 06 21 6f 72 67 2e 61 70 61 63 68 65 2e SEQ.!org.apache.
0000010: 68 61 64 6f 6f 70 2e 69 6f 2e 4c 6f 6e 67 57 72 hadoop.io.LongWr
0000020: 69 74 61 62 6c 65 22 6f 72 67 2e 61 70 61 63 68 itable"org.apach
0000030: 65 2e 68 61 64 6f 6f 70 2e 69 6f 2e 42 79 74 65 e.hadoop.io.Byte
0000040: 73 57 72 69 74 61 62 6c 65 00 00 00 00 00 2d sWritable.....-
0000050: 01 4d 2d 5e e1 65 f3 f7 1e c0 3d 4d ec 01 2e 00 .M-^..e...=M...
0000060: 00 00 2e 00 00 00 08 00 00 01 74 e9 98 5f 36 00 .....t..._6.
0000070: 00 00 22 48 65 6c 6c 6f 20 48 44 46 53 20 77 72 .."Hello HDFS wr
0000080: 69 74 69 6e 67 20 66 72 6f 6d 20 46 6c 75 6d 65 iting from Flume
0000090: 20 5e 2e 5e 0d ff ff ff ff 2d 01 4d 2d 5e e1 65 ^..^.....-..M-^..e
00000a0: f3 f7 1e c0 3d 4d ec 01 2e ....=M...
```

# Apache Sqoop Tutorial

---

# Apache Sqoop Tutorial



## Sqoop (Sql-to-hadoop)

- Command-line interface for **transforming** data between **RDBMS & Hadoop**
- Parallelized data transfer with **MapReduce**
- Support **incremental** imports
- **Imports** use to populate tables in Hadoop
- **Exports** use to put data from Hadoop into relational database
- **Sqoop2** -> Sqoop-as-a-Service: server-based implementation of Sqoop



# Apache Sqoop Tutorial

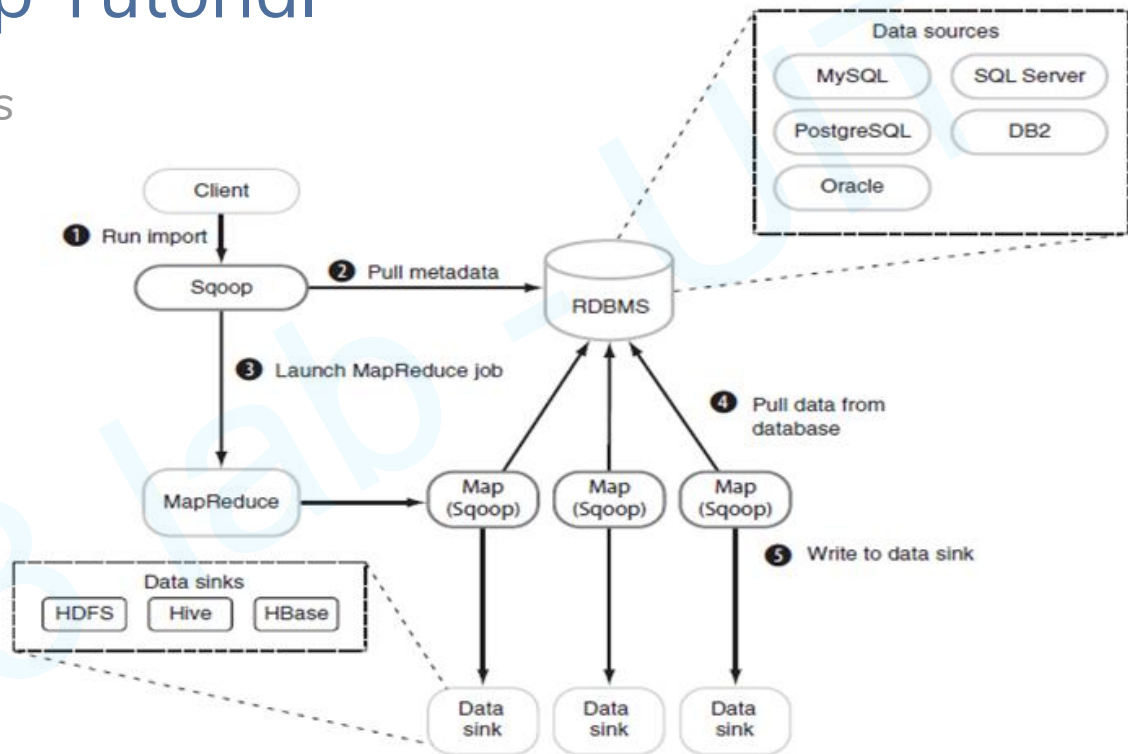


## *Sqoop - How it works*

- The dataset being transferred is broken into small blocks.
- **Map** only job is launched.
- Individual mapper is responsible for transferring a block of the dataset.

# Apache Sqoop Tutorial

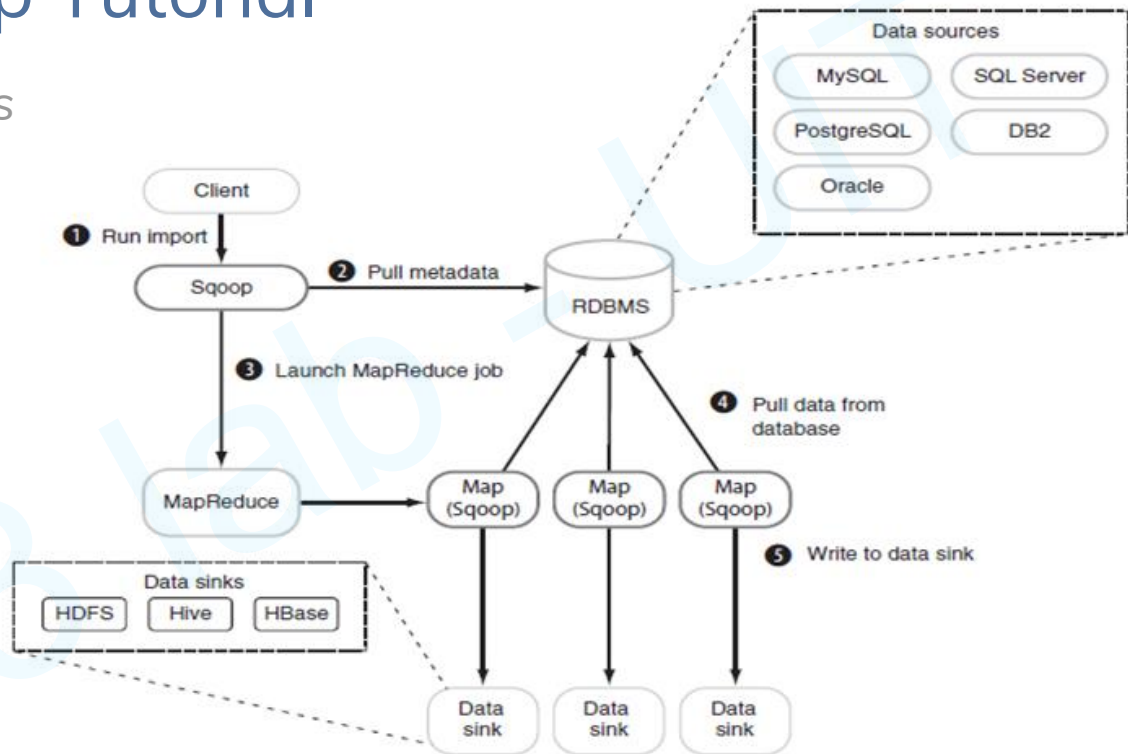
## Sqoop - How it works



**Figure 2.20** Five-stage Sqoop import overview: connecting to the data source and using MapReduce to write to a data sink

# Apache Sqoop Tutorial

## Sqoop - How it works

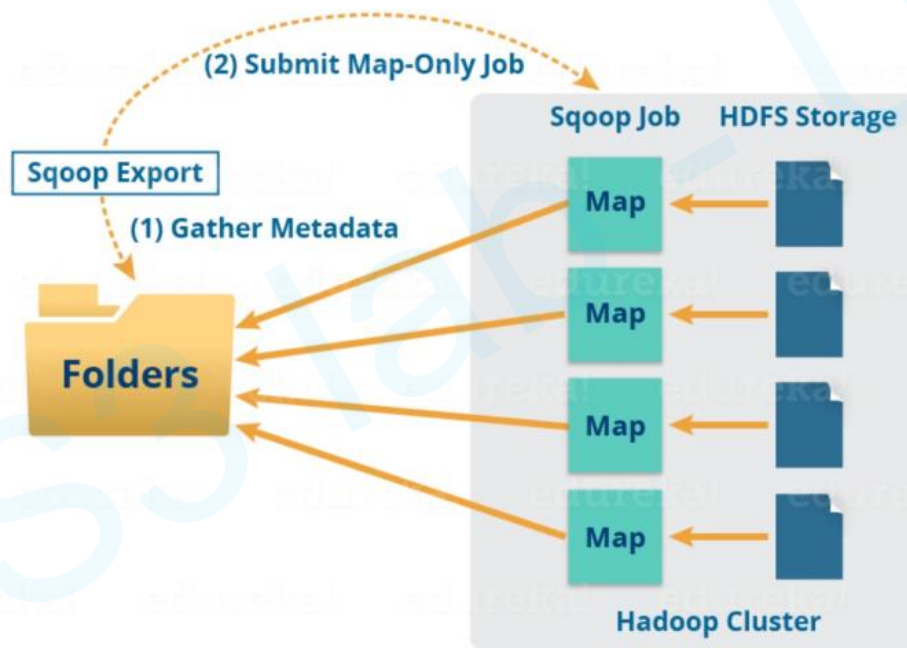


**Figure 2.20** Five-stage Sqoop import overview: connecting to the data source and using MapReduce to write to a data sink



# Apache Sqoop Tutorial

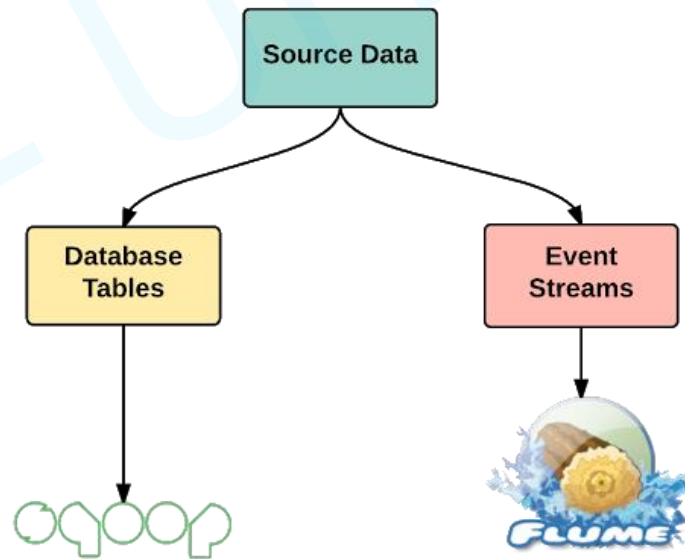
## Sqoop - How it works



# Apache Sqoop Tutorial

## Flume vs Sqoop

- Flume only ingests unstructured data or semi-structured data into HDFS.
- Sqoop can import as well as export structured data from RDBMS or Enterprise data warehouses to HDFS or vice versa.



# Apache Sqoop Tutorial

- Display a list of all available tools
- Command: **sqoop help**

```
[cloudera@quickstart ~]$ 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
import-mainframe	Import datasets from a mainframe server 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

# Sqoop connecting to a Database Server

```
[cloudera@quickstart ~]$ sqoop help import
```

Argument	Description
<code>--connect &lt;jdbc-uri&gt;</code>	Specify JDBC connect string
<code>--connection-manager &lt;class-name&gt;</code>	Specify connection manager class to use
<code>--driver &lt;class-name&gt;</code>	Manually specify JDBC driver class to use
<code>--hadoop-mapred-home &lt;dir&gt;</code>	Override \$HADOOP_MAPRED_HOME
<code>--help</code>	Print usage instructions
<code>--password-file</code>	Set path for a file containing the authentication password
<code>-P</code>	Read password from console
<code>--password &lt;password&gt;</code>	Set authentication password
<code>--username &lt;username&gt;</code>	Set authentication username
<code>--verbose</code>	Print more information while working
<code>--connection-param-file &lt;filename&gt;</code>	Optional properties file that provides connection parameters
<code>--relaxed-isolation</code>	Set connection transaction isolation to read uncommitted for the mappers.

# Sqoop import RDBMS table into HDFS

- Login to mysql

```
[cloudera@quickstart ~]$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 19
```

# Sqoop import RDBMS table into HDFS

- Create database StudentInfo

```
mysql> create database StudentInfo;  
Query OK, 1 row affected (0.00 sec)
```

```
mysql> show databases;
```

Database
information_schema
StudentInfo
cm
firehose
hue
metastore
mysql
nav
navms
oozie
retail_db
rman
sentry

```
13 rows in set (0.00 sec)
```

```
mysql> █
```

- Use the newly created database

```
mysql> use StudentInfo;  
Database changed
```

# Sqoop import RDBMS table into HDFS

- Create table student and insert some data into this table

```
mysql> create table student(std_id integer, std_name varchar(43));  
Query OK, 0 rows affected (0.01 sec)  
  
mysql> insert into student values (101,'le'), (102,'pham'), (103,'tran'), (104,'  
ngo'), (105,'vu'), (106,'dao');  
Query OK, 6 rows affected (0.00 sec)  
Records: 6  Duplicates: 0  Warnings: 0
```

# Sqoop import RDBMS table into HDFS

- Check the newly created table

```
mysql> select * from student;
```

std_id	std_name
101	le
102	pham
103	tran
104	ngo
105	vu
106	dao

```
6 rows in set (0.00 sec)
```

```
mysql> █
```



# Sqoop import RDBMS table into HDFS

- Use Sqoop to import table student into HDFS
- Command: `[cloudera@quickstart ~]$ sqoop import --connect jdbc:mysql://localhost/StudentInfo --table student --username root --password cloudera --split-by std_id --m 1 --target-dir /user/cloudera/studentInfo/student;`

```
[cloudera@quickstart ~]$ sqoop import --connect jdbc:mysql://localhost/StudentInfo --table student --username root
--password cloudera --split-by std_id --m 1 --target-dir '/user/cloudera/studentInfo/student';
Warning: /usr/lib/sqoop/./accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
20/09/29 21:39:27 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5.13.0
20/09/29 21:39:27 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P
instead.
20/09/29 21:39:28 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
20/09/29 21:39:28 INFO tool.CodeGenTool: Beginning code generation
20/09/29 21:39:28 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `student` AS t LIMIT 1
20/09/29 21:39:28 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `student` AS t LIMIT 1
20/09/29 21:39:28 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/lib/hadoop-mapreduce
...
File Input Format Counters
  Bytes Read=0
File Output Format Counters
  Bytes Written=48
20/09/29 21:39:52 INFO mapreduce.ImportJobBase: Transferred 48 bytes in 20.273 seconds (2.3677 bytes/sec)
20/09/29 21:39:52 INFO mapreduce.ImportJobBase: Retrieved 6 records.
```

# Sqoop import RDBMS table into HDFS

```
[cloudera@quickstart ~]$ hdfs dfs -ls
Found 7 items
drwxr-xr-x  - cloudera cloudera      0 2020-09-29 00:30 HSOutput
drwxr-xr-x  - cloudera cloudera      0 2020-09-28 23:13 HadoopStreaming
drwxr-xr-x  - cloudera cloudera      0 2020-09-28 05:17 ReduceJoin
drwxr-xr-x  - cloudera cloudera      0 2020-09-29 03:20 dataset
drwxr-xr-x  - cloudera cloudera      0 2020-09-27 07:07 inputWC
drwxr-xr-x  - cloudera cloudera      0 2020-09-27 07:29 outputWC
drwxr-xr-x  - cloudera cloudera      0 2020-09-29 21:39 studentInfo
[cloudera@quickstart ~]$ hdfs dfs -ls studentInfo
Found 1 items
drwxr-xr-x  - cloudera cloudera      0 2020-09-29 21:39 studentInfo/student
[cloudera@quickstart ~]$ hdfs dfs -ls studentInfo/student
Found 2 items
-rw-r--r--  1 cloudera cloudera      0 2020-09-29 21:39 studentInfo/student/_SUCCESS
-rw-r--r--  1 cloudera cloudera    48 2020-09-29 21:39 studentInfo/student/part-m-00000
[cloudera@quickstart ~]$
```

# Sqoop import RDBMS table into HDFS

- Now let us see the output on our Command shell:

```
[cloudera@quickstart ~]$ hdfs dfs -cat studentInfo/student/part-m-00000  
101,le  
102,pham  
103,tran  
104,ngo  
105,vu  
106,dao  
[cloudera@quickstart ~]$
```

---

## Sqoop import RDBMS table into HDFS without target directory

- Import RDBMS table into HDFS without specifying target directory
- Command: `[cloudera@quickstart ~]$ sqoop import --connect jdbc:mysql://localhost/StudentInfo --table student --username root --password cloudera --split-by std_id --m 1`

```
[cloudera@quickstart ~]$ sqoop import --connect jdbc:mysql://localhost/StudentInfo --table student --username root --password cloudera --split-by std_id --m 1
Warning: /usr/lib/sqoop/./accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
20/09/29 22:35:55 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5.13.0
20/09/29 22:35:55 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.
20/09/29 22:35:55 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
20/09/29 22:35:55 INFO tool.CodeGenTool: Beginning code generation
20/09/29 22:35:56 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `student` AS t LIMIT 1
20/09/29 22:35:56 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `student` AS t LIMIT 1
20/09/29 22:35:56 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `student` AS t LIMIT 1
20/09/29 22:35:56 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `student` AS t LIMIT 1
20/09/29 22:35:56 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `student` AS t LIMIT 1
20/09/29 22:36:15 INFO mapreduce.ImportJobBase: Transferred 48 bytes in 16.5944 seconds (2.8926 bytes/sec)
20/09/29 22:36:15 INFO mapreduce.ImportJobBase: Retrieved 6 records.
```

## Sqoop import RDBMS table into HDFS without target directory

- Check the newly created directory

```
[cloudera@quickstart ~]$ hdfs dfs -ls
Found 8 items
drwxr-xr-x  - cloudera cloudera      0 2020-09-29 00:30 HSOutput
drwxr-xr-x  - cloudera cloudera      0 2020-09-28 23:13 HadoopStreaming
drwxr-xr-x  - cloudera cloudera      0 2020-09-28 05:17 ReduceJoin
drwxr-xr-x  - cloudera cloudera      0 2020-09-29 03:20 dataset
drwxr-xr-x  - cloudera cloudera      0 2020-09-27 07:07 inputWC
drwxr-xr-x  - cloudera cloudera      0 2020-09-27 07:29 outputWC
drwxr-xr-x  - cloudera cloudera      0 2020-09-29 22:36 student
drwxr-xr-x  - cloudera cloudera      0 2020-09-29 21:39 studentInfo
[cloudera@quickstart ~]$ hdfs dfs -ls student
Found 2 items
-rw-r--r--  1 cloudera cloudera      0 2020-09-29 22:36 student/_SUCCESS
-rw-r--r--  1 cloudera cloudera    48 2020-09-29 22:36 student/part-m-00000
[cloudera@quickstart ~]$
```

# Sqoop – IMPORT Command with Where Clause

- Command: `sqoop import --connect jdbc:mysql://localhost/StudentInfo --username root --password cloudera --table student --m 1 --where 'std_id > 103' --target-dir /user/cloudera/studentInfo/studentAfter103`

```
[cloudera@quickstart ~]$ sqoop import --connect jdbc:mysql://localhost/StudentInfo --username root --password cloudera --table student --m 1 --where 'std_id > 103' --target-dir /user/cloudera/studentInfo/studentAfter103
Warning: /usr/lib/sqoop/./accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
20/09/30 05:04:07 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5.13.0
20/09/30 05:04:07 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.
20/09/30 05:04:07 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
20/09/30 05:04:07 INFO tool.CodeGenTool: Beginning code generation
20/09/30 05:04:08 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `student` AS t LIMIT 1
20/09/30 05:04:08 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `student` AS t LIMIT 1
```

# Sqoop – IMPORT Command with Where Clause

- Check the result in HDFS

```
[cloudera@quickstart ~]$ hdfs dfs -ls studentInfo
Found 2 items
drwxr-xr-x  - cloudera cloudera      0 2020-09-29 21:39 studentInfo/student
drwxr-xr-x  - cloudera cloudera      0 2020-09-30 05:04 studentInfo/studentAfter103
[cloudera@quickstart ~]$ hdfs dfs -ls studentInfo/studentAfter103
Found 2 items
-rw-r--r--   1 cloudera cloudera      0 2020-09-30 05:04 studentInfo/studentAfter103/_SUCCESS
-rw-r--r--   1 cloudera cloudera    23 2020-09-30 05:04 studentInfo/studentAfter103/part-m-000000
[cloudera@quickstart ~]$ hdfs dfs -cat studentInfo/studentAfter103/part*
104,ngo
105,vu
106,dao
[cloudera@quickstart ~]$
```

# Free-form Query Imports

- Sqoop can also import the result set of an arbitrary SQL query.
- When importing a free-form query, you must specify `--target-dir`, `--split-by`, and include the token `$CONDITIONS`.
- Command: `sqoop import --connect jdbc:mysql://localhost/StudentInfo --username root --password cloudera --query 'select std_name from student where std_id=103 and $CONDITIONS' --split-by std_name --target-dir /user/cloudera/studentInfo/studentName103`

```
[cloudera@quickstart ~]$ sqoop import --connect jdbc:mysql://localhost/StudentInfo --username root --password cloudera --query 'select std_name from student where std_id=103 and $CONDITIONS' --split-by std_name --target-dir /user/cloudera/studentInfo/studentName103
Warning: /usr/lib/sqoop/./accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
20/09/30 05:22:53 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5.13.0
20/09/30 05:22:53 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.

File Input Format Counters
  Bytes Read=0
File Output Format Counters
  Bytes Written=5
20/09/30 05:23:13 INFO mapreduce.ImportJobBase: Transferred 5 bytes in 16.4991 seconds (0.303 bytes/sec)
20/09/30 05:23:13 INFO mapreduce.ImportJobBase: Retrieved 1 records.
[cloudera@quickstart ~]$
```



# Free-form Query Imports

- Check the result in HDFS

```
[cloudera@quickstart ~]$ hdfs dfs -ls studentInfo
Found 3 items
drwxr-xr-x  - cloudera cloudera      0 2020-09-29 21:39 studentInfo/student
drwxr-xr-x  - cloudera cloudera      0 2020-09-30 05:04 studentInfo/studentAfter103
drwxr-xr-x  - cloudera cloudera      0 2020-09-30 05:23 studentInfo/studentName103
[cloudera@quickstart ~]$ hdfs dfs -ls studentInfo/studentName103
Found 2 items
-rw-r--r--  1 cloudera cloudera      0 2020-09-30 05:23 studentInfo/studentName103/_SUCCESS
-rw-r--r--  1 cloudera cloudera      5 2020-09-30 05:23 studentInfo/studentName103/part-m-00000
[cloudera@quickstart ~]$ hdfs dfs -cat studentInfo/studentName103/part*
tran
[cloudera@quickstart ~]$
```

# Sqoop - list all databases

- List all databases
- Command: **sqoop list-databases --connect jdbc:mysql://localhost/ --username root --password cloudera**

```
[cloudera@quickstart ~]$ sqoop list-databases --connect jdbc:mysql://localhost/ --username root --password cloudera
Warning: /usr/lib/sqoop/./accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
20/09/30 06:07:11 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5.13.0
20/09/30 06:07:11 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.
20/09/30 06:07:12 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
information_schema
StudentInfo
cm
firehose
hue
metastore
mysql
nav
navms
oozie
retail_db
rman
sentry
[cloudera@quickstart ~]$
```

# Sqoop - list all tables

- Let's create another table in the StudentInfo database

```
mysql> use StudentInfo;  
Reading table information for completion of table and column names  
You can turn off this feature to get a quicker startup with -A
```

```
Database changed
```

```
mysql> create table course (course_id integer, course_name varchar(43), num_credit integer);  
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> insert into course values (11,'big data',3), (12,'deep learning',4), (13,'data structure',4), (14,'database', 3);  
Query OK, 4 rows affected (0.00 sec)  
Records: 4 Duplicates: 0 Warnings: 0
```

```
mysql> █
```

# Sqoop - list all tables

- List all tables of a particular database
- Command: **sqoop list-tables --connect jdbc:mysql://localhost/StudentInfo --username root --password cloudera**

```
[cloudera@quickstart ~]$ sqoop list-tables --connect jdbc:mysql://localhost/StudentInfo --username root --password cloudera
Warning: /usr/lib/sqoop/./accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
20/09/30 06:11:31 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5.13.0
20/09/30 06:11:31 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.
20/09/30 06:11:31 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
course
student
[cloudera@quickstart ~]$
```

# Sqoop Imports All Tables

- Let's import all these tables into HDFS
- Command: `sqoop import-all-tables --connect jdbc:mysql://localhost/StudentInfo --username root --password cloudera --m 1`

```
[cloudera@quickstart ~]$ sqoop import-all-tables --connect jdbc:mysql://localhost/StudentInfo --username root --password cloudera --m 1
Warning: /usr/lib/sqoop/./accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
20/09/30 06:00:07 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5.13.0
20/09/30 06:00:07 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.
20/09/30 06:00:08 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
20/09/30 06:00:08 INFO tool.CodeGenTool: Beginning code generation
```

# Sqoop Imports All Tables

- Check if two folders appeared in HDFS

```
[cloudera@quickstart ~]$ hdfs dfs -ls
Found 9 items
drwxr-xr-x  - cloudera cloudera      0 2020-09-29 00:30 HSOoutput
drwxr-xr-x  - cloudera cloudera      0 2020-09-28 23:13 HadoopStreaming
drwxr-xr-x  - cloudera cloudera      0 2020-09-28 05:17 ReduceJoin
drwxr-xr-x  - cloudera cloudera      0 2020-09-30 06:00 course ←
drwxr-xr-x  - cloudera cloudera      0 2020-09-29 03:20 dataset
drwxr-xr-x  - cloudera cloudera      0 2020-09-27 07:07 inputWC
drwxr-xr-x  - cloudera cloudera      0 2020-09-27 07:29 outputWC
drwxr-xr-x  - cloudera cloudera      0 2020-09-30 06:00 student ←
drwxr-xr-x  - cloudera cloudera      0 2020-09-30 05:23 studentInfo
[cloudera@quickstart ~]$
```

# Sqoop Export data from HDFS to the RDBMS

- Create a new file in local file system
- Command: **cat > newstudent**
- Press ctr-d to save file
- Then put this file to HDFS

```
[cloudera@quickstart ~]$ cat > newstudent
107,nam
108,viet
109,quoc
[cloudera@quickstart ~]$ hdfs dfs -put newstudent
[cloudera@quickstart ~]$ hdfs dfs -ls
Found 10 items
drwxr-xr-x - cloudera cloudera          0 2020-09-29 00:30 HSOutput
drwxr-xr-x - cloudera cloudera          0 2020-09-28 23:13 HadoopStreaming
drwxr-xr-x - cloudera cloudera          0 2020-09-28 05:17 ReduceJoin
drwxr-xr-x - cloudera cloudera          0 2020-09-30 06:00 course
drwxr-xr-x - cloudera cloudera          0 2020-09-29 03:20 dataset
drwxr-xr-x - cloudera cloudera          0 2020-09-27 07:07 inputWC
-rw-r--r-- 1 cloudera cloudera        26 2020-09-30 08:25 newstudent ←
drwxr-xr-x - cloudera cloudera          0 2020-09-27 07:29 outputWC
drwxr-xr-x - cloudera cloudera          0 2020-09-30 06:00 student
drwxr-xr-x - cloudera cloudera          0 2020-09-30 05:23 studentInfo
[cloudera@quickstart ~]$
```

- ```

student in HDFS to table student in mysql
e target database
--connect jdbc:mysql://localhost/StudentIn
a --table student --export-dir /user/clouder

ct jdbc:mysql://localhost/StudentInfo --username root --password cloud
exist! Accumulo imports will fail.
Accumulo installation.
goop version: 1.4.6-cdh5.13.0
ation you received on the command line to determine. Consider using a

se: Transferred 708 bytes in 27.1772 seconds (26.0513 bytes/sec)
se: Exported 3 records.

```

```
cloudera@quickstart ~]$ sqoop export --connect jdbc:mysql://localhost/StudentInfo --username root --password cloudera --table student --export-dir /user/cloudera/newstudent
Warning: /usr/lib/sqoop/./accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
20/09/30 08:34:00 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5.13.0
20/09/30 08:34:00 INFO mapreduce.ExportJobBase: Starting export job. Consider using --direct for better performance.
20/09/30 08:34:31 INFO mapreduce.ExportJobBase: Transferred 708 bytes in 27.1772 seconds (26.0513 bytes/sec)
20/09/30 08:34:31 INFO mapreduce.ExportJobBase: Exported 3 records.
```



# Sqoop Export data from HDFS to the RDBMS

- Query table **student** in sql to check new rows are inserted

```
[cloudera@quickstart ~]$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
```

```
mysql> use StudentInfo;
```

```
mysql> select * from student;
```

| std_id | std_name |
|--------|----------|
| 101    | le       |
| 102    | pham     |
| 103    | tran     |
| 104    | ngo      |
| 105    | vu       |
| 106    | dao      |
| 107    | nam      |
| 108    | viet     |
| 109    | quoc     |

```
9 rows in set (0.00 sec)
```



## Cảm ơn đã theo dõi

Chúng tôi hy vọng cùng nhau đi đến thành công.