

# DATA INGESTION INTO HADOOP USING APCHE NIFI

## Activity 2

Apache NiFi is a dataflow system based on the concepts of flow-based programming. It supports powerful and scalable directed graphs of data routing, transformation, and system mediation logic. It has a web-based user interface for design, control, feedback, and monitoring of data flows. Put simply NiFi was built to automate the flow of data between systems. While the term 'dataflow' is used in a variety of contexts, we use it here to mean the automated and managed flow of information between systems.

In this exercise, students will learn how to move data from MySQL database into a hive table all while capturing new record inserts in real time.

**MySQL Database:** Linux

**Hive:** Hortonworks Sandbox

### Creating the Database in MySQL

Log in to MySQL and create a source database from which records will be fetched.

```
CREATE database falconpro;

CREATE TABLE `users` (
  `id` int(11) NOT NULL,
  `age` int(11) DEFAULT NULL,
  `gender` char(1) DEFAULT NULL,
  `occupation_id` int(11) DEFAULT NULL,
  `zip_code` varchar(255) DEFAULT NULL,
  PRIMARY KEY (`id`)
);
```

Insert the following records into the table user.

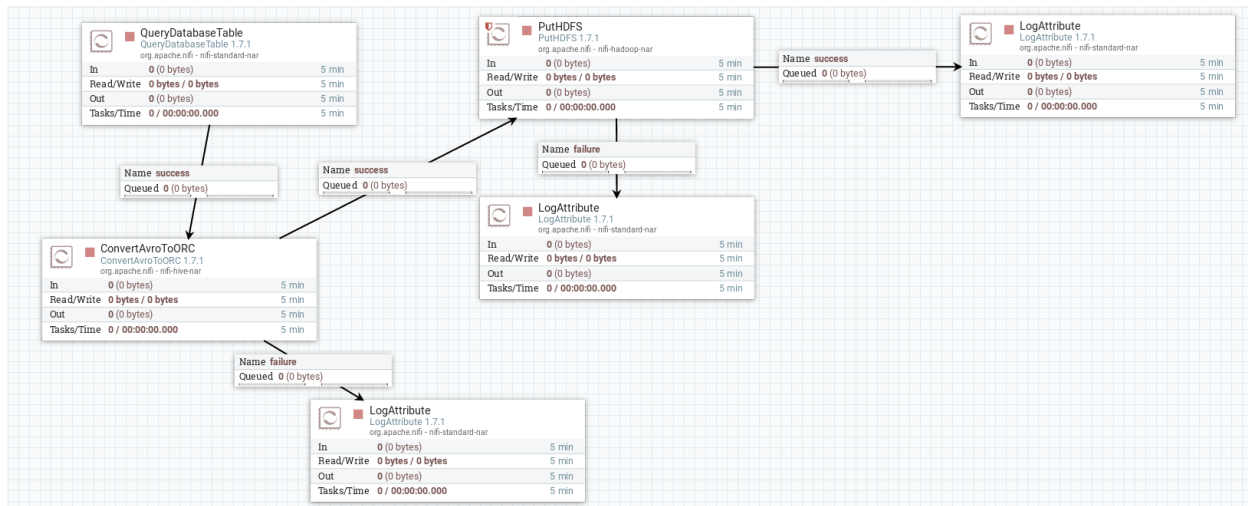
```
INSERT INTO `users` VALUES (1,24,'M',20,'85711'),(2,53,'F',14,'94043'),  
(3,23,'M',21,'32067'),(4,24,'M',20,'43537'),(5,33,'F',14,'15213'),  
(6,42,'M',7,'98101'),(7,57,'M',1,'91344'),(8,36,'M',1,'05201');
```

### **Creating the Table in Hive**

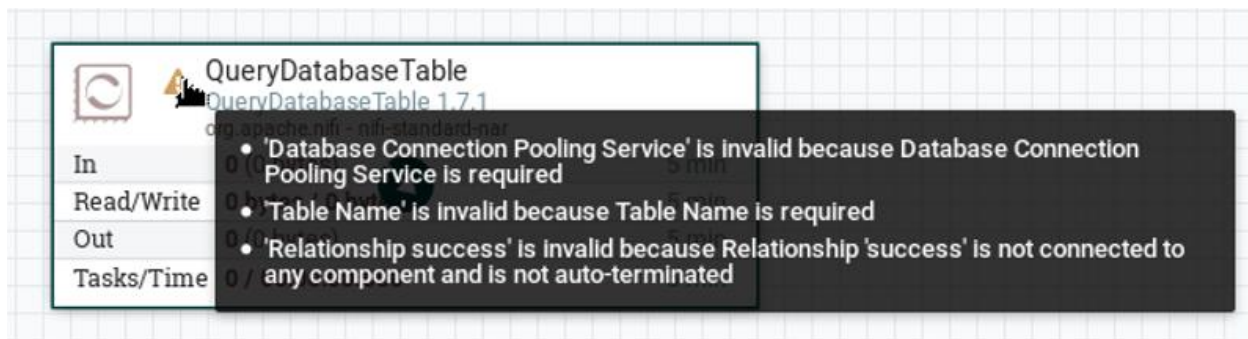
Create a table by running the following script below.

```
CREATE external TABLE falconUsers (  
    id int,  
    age int,  
    gender string,  
    occupation_id int,  
    zip_code string)  
stored as orc  
location '/user/maria_dev/falcon_db_copy';
```

## Creating the Flow in Nifi



Since the purpose of the exercise is to fetch records from a database, specifically from a table in that database, the processor of choice would be the QueryDatabaseTable component.



Of the three warning messages listed, we will focus on the 'Database Connection Pooling Service' warning as the 'Table Name' warning is self-explanatory and the 'Relationship success' warning was addressed in Exercise 1.

### Inside QueryDatabaseTable

This processor generates and executes a SQL select query to fetch all rows whose values in the specified Maximum Value column(s) are larger than the previously-seen maxima. Query result will be converted to Avro format. As seen in the image below, the Database Connection Pooling Service requires values in its field. But in this case it requires a connection to a Controller Service. This component will be used to establish connection to our database.

## Configure Processor

SETTINGS

SCHEDULING

PROPERTIES

COMMENTS

Required field

+

The Controller Service that is used to obtain a connection to the database.

Expression language scope: Not Supported

Requires Controller Service: DBCPService 1.7.1 from org.apache.nifi - nifi-standard-services-api-nar

Property		Value
Database Connection Pooling Service		No value set
Database Type		Generic
Table Name		No value set
Columns to Return		No value set
Additional WHERE clause		No value set
Custom Query		No value set
Maximum-value Columns		No value set
Max Wait Time		0 seconds
Fetch Size		0
Max Rows Per Flow File		0
Output Batch Size		0
Maximum Number of Fragments		0
Normalize Table/Column Names		false
Use Avro Logical Types		false

CANCEL

APPLY

Possible connection are listed in a dropdown by simply inside the value field.

Select DBCPConnectionPool.

## Configure Processor

SETTINGS

SCHEDULING

PROPERTIES

COMMENTS

Required field

+

Property		Value
Database Connection Pooling Service		<div> <div>DBCPConnectionPool</div> <div> <div>DBCPConnectionPool</div> <div>DBCPConnectionPool</div> <div>MySQL CDC Backup</div> <div>Create new service...</div> </div> </div>
Database Type		
Table Name		
Columns to Return		
Additional WHERE clause		
Custom Query		
Maximum-value Columns		
Max Wait Time		0 seconds
Fetch Size		0
Max Rows Per Flow File		0
Output Batch Size		0
Maximum Number of Fragments		0
Normalize Table/Column Names		false
Use Avro Logical Types		false

CANCEL

APPLY

Note: For first time users the “Create new service” field would appear instead of the list of options in the dropdown in the image above.

**Configure Processor**

SETTINGS SCHEDULING **PROPERTIES** COMMENTS

Required field +

Property	
Database Connection Pooling Service	No value
Database Type	No value
Table Name	Create new service...
Columns to Return	
Additional WHERE clause	No value set
Custom Query	No value set
Maximum-value Columns	No value set
Max Wait Time	0 seconds
Fetch Size	0
Max Rows Per Flow File	0
Output Batch Size	0
Maximum Number of Fragments	0
Normalize Table/Column Names	false
Use Avro Logical Types	false

CANCEL APPLY

By clicking on Create new service... Nifi automatically makes a suggestion for the Controller Service to use.

**Configure Processor**

SETTINGS SCHEDULING **PROPERTIES** COMMENTS

Required field +

Property	
Database Connection Pooling Service	
Database Type	
Table Name	
Columns to Return	
Additional WHERE clause	
Custom Query	
Maximum-value Columns	
Max Wait Time	
Fetch Size	
Max Rows Per Flow File	
Output Batch Size	
Maximum Number of Fragments	
Normalize Table/Column Names	
Use Avro Logical Types	false

CANCEL APPLY

**Add Controller Service**

Requires Controller Service  
DBCPService 1.7.1 from org.apache.nifi - nifi-standard-services-api-nar

Compatible Controller Services  
DBCPConnectionPool 1.7.1

Controller Service Name  
DBCPConnectionPool

Bundle  
org.apache.nifi - nifi-dbc-p-service-nar

Tags  
database, pooling, dbcp, jdbc, connection, store

Description  
Provides Database Connection Pooling Service. Connections can be...

CANCEL CREATE

### Configure Processor

SETTINGS
SCHEDULING
**PROPERTIES**
COMMENTS

Required field +

Property	Value
Database Connection Pooling Service	DBCPConnectionPool →
Database Type	Generic
Table Name	No value set
Columns to Return	No value set
Additional WHERE clause	No value set
Custom Query	No value set
Maximum-value Columns	No value set
Max Wait Time	0 seconds
Fetch Size	0
Max Rows Per Flow File	0
Output Batch Size	0
Maximum Number of Fragments	0
Normalize Table/Column Names	false
Use Auto Incremental Tunes	false

CANCEL
APPLY

Upon selecting it an arrow appears. Click on the arrow to provide properties field values to establish connection to MySQL and the intended database.

### NiFi Flow Configuration

GENERAL
**CONTROLLER SERVICES**

Name	Type	Bundle	State	Scope
DBCPConnectionPool	DBCPConnectionPool 1.7.1	org.apache.nifi-nifi-dbcop-service-nar	Invalid	NiFi Flow

Database Connection URL is invalid because Database Connection URL is required  
Database Driver Class Name is invalid because Database Driver Class Name is required

Configure here

Last updated: 23:53:19 CDT

Listed services are available to all descendant Processors and services of this Process Group.

Below are the details that go into this component. Make sure to use the proper path for your jar file, correct values for your Database user name and password.

### Configure Controller Service

SETTINGS

PROPERTIES

COMMENTS

Required field

Property	Value
Database Connection URL	jdbc:mysql://127.0.0.1:3306/falconpro
Database Driver Class Name	com.mysql.jdbc.Driver
Database Driver Location(s)	file:///usr/share/java/mysql-connector-java-5.1.37.jar
Database User	root
Password	Sensitive value set
Max Wait Time	500 millis
Max Total Connections	8
Validation query	No value set

CANCEL

APPLY

After adding the details, the component must be enabled. Going back to the QueryDatabaseTable processor, add the following details:

Table Name: users

Columns: users.columns

Maximum-value: id

As stated above, this processor generates and executes a SQL select query to fetch all rows whose values in the specified **Maximum Value column(s)** are larger than the previously-seen maxima.

For this being the case, it's best to always use fields with auto-increment unique primary keys, current timestamp or other trigger events on your table for the Maximum Value in the processor.

### Configure Processor

SETTINGS

SCHEDULING

PROPERTIES

COMMENTS

Required field

Property	Value
Database Connection Pooling Service	DBCPConnectionPool
Database Type	Generic
Table Name	users
Columns to Return	id,age,gender,occupation_id,zip_code
Additional WHERE clause	No value set
Custom Query	No value set
Maximum-value Columns	id
Max Wait Time	0 seconds
Fetch Size	0
Max Rows Per Flow File	0
Output Batch Size	0
Maximum Number of Fragments	0
Normalize Table/Column Names	false
Use Avro Logical Types	false

CANCEL

APPLY

Next up is the ConvertAvroToORC processor. In many cases, using avro type file to populate data in a hive table can result in null values in the columns. Therefore flowfiles in Avro format must be converted to ORC as the default file format of query results from the QueryDatabaseTable are converted to Avro format.

### Add Processor

Source

all groups

Displaying 10 of 274

convert

Type	Version	Tags
ConvertAvroSchema	1.7.1	convert, kite, avro
ConvertAvroToJSON	1.7.1	json, convert, avro
ConvertAvroToORC	1.7.1	hive, orc, convert, avro
ConvertCSVToAvro	1.7.1	csv, kite, avro
ConvertCharacterSet	1.7.1	character set, character set, text...
ConvertExcelToCSVProcessor	1.7.1	excel, csv, poi
ConvertJSONToAvro	1.7.1	json, kite, avro
ConvertJSONToSQL	1.7.1	database, rdbms, flat, json, inse...
ConvertRecord	1.7.1	schema, log, record, csv, freefor...
LookupRecord	1.7.1	filter, lookup, enrichment, route, ...

**ConvertAvroToORC 1.7.1** org.apache.nifi - nifi-hive-nar

Converts an Avro record into ORC file format. This processor provides a direct mapping of an Avro record to an ORC record, such that the resulting ORC file will have the same hierarchical structure as the Avro document. If an incoming FlowFile contains a stream of multiple Avro records, the resultant FlowFile will contain a ORC file containing all of the Avro records. If an incoming FlowFil...

CANCELADD

The processor is configured by adding the path to the hive-site.xml in the ORC Configuration Resources value field.

### Configure Processor

SETTINGS

SCHEDULING

PROPERTIES

COMMENTS

Required field

Expression language scope: Not Supported

ORC Configuration Resources	/root/Documents/hive-site.xml
Stripe Size	64 MB
Buffer Size	10 KB
Compression Type	NONE
Hive Table Name	No value set

CANCELAPPLY



We will then create a LogAttribute processor to keep track of failed flow file events and send all the success flow files to our PutHDFS processor.

The PutHDFS should be configured as such:

Configure Processor

SETTINGS

SCHEDULING

PROPERTIES

COMMENTS

Required field

Property	Value
Hadoop Configuration Resources	/etc/hadoop/conf/core-site.xml/etc/hadoop/conf/hdfs-sit...
Kerberos Credentials Service	No value set
Kerberos Principal	No value set
Kerberos Keytab	No value set
Kerberos Relogin Period	4 hours
Additional Classpath Resources	No value set
Directory	/user/maria_dev/falcon_db_copy
Conflict Resolution Strategy	fail
Block Size	No value set
IO Buffer Size	No value set
Replication	No value set
Permissions umask	No value set
Remote Owner	No value set
Remote Group	No value set

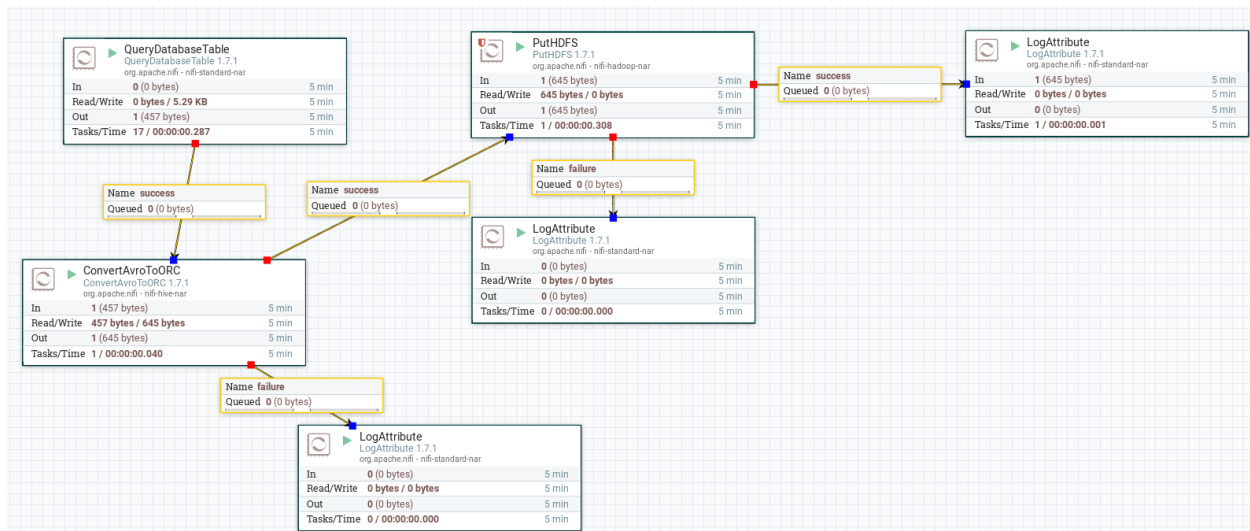
CANCEL

APPLY

Lastly we will add two additional LogAttribute processors to track our success and failures.

Select all the processor and press PLAY.

Nifi



## HDFS

Ambari Sandbox 0 ops 0 alerts Dashboard Services Hosts Alerts Admin admin

/ > user > maria\_dev > falcon\_db\_copy Total: 1 files or folders + Select All New Folder Upload

Search in current directory...

Name >	Size >	Last Modified ▼	Owner >	Group >	Permission
106922784812121.orc	0.6 kB	2018-10-19 04:14	root	hdfs	-rw-r--r--

## HIVE

```
1 SELECT * FROM falconusers LIMIT 100;
```

Execute Explain Save as... New Worksheet

Query Process Results (Status: SUCCEEDED) Save results... ▾

Logs Results

Filter columns... previous next

falconusers.id	falconusers.age	falconusers.gender	falconusers.occupation_id	falconusers.zip_code
1	24	M	20	85711
2	53	F	14	94043
3	23	M	21	32067

For incremental imports, add new records in the users table in the falcon database in MySQL all while Nifi running. Nifi will automatically fetch the new records into Hadoop.