Escuela de Datos Vivos

Curso Data Engineer: Creando un pipeline de datos

MÓDULO A - Clase 3

Ambiente Hadoop

Docker Hadoop

Bajar la imagen

```
docker pull fedepineyro/edvai_ubuntu:v6
```

Docker Hadoop

Correr la imagen

```
docker run --name edvai_hadoop -p 8081:8081 -p 8080:8080 -p 8088:8088 -p 9870:9870 -p
9868:9868 -p 9864:9864 -p 1527:1527 -p 10000:10000 -p 10002:10002 -p 50111:50111 -p
8010:8010 -p 9093:9093 -p 2181:2182 -it  --restart unless-stopped
fedepineyro/edvai_ubuntu:v6 /bin/bash -c "/home/hadoop/scripts/start-services.sh"
```

Docker Hadoop

Ingresar al bash del contenedor

```
docker exec -it edvai_hadoop bash
```

Docker Hadoop

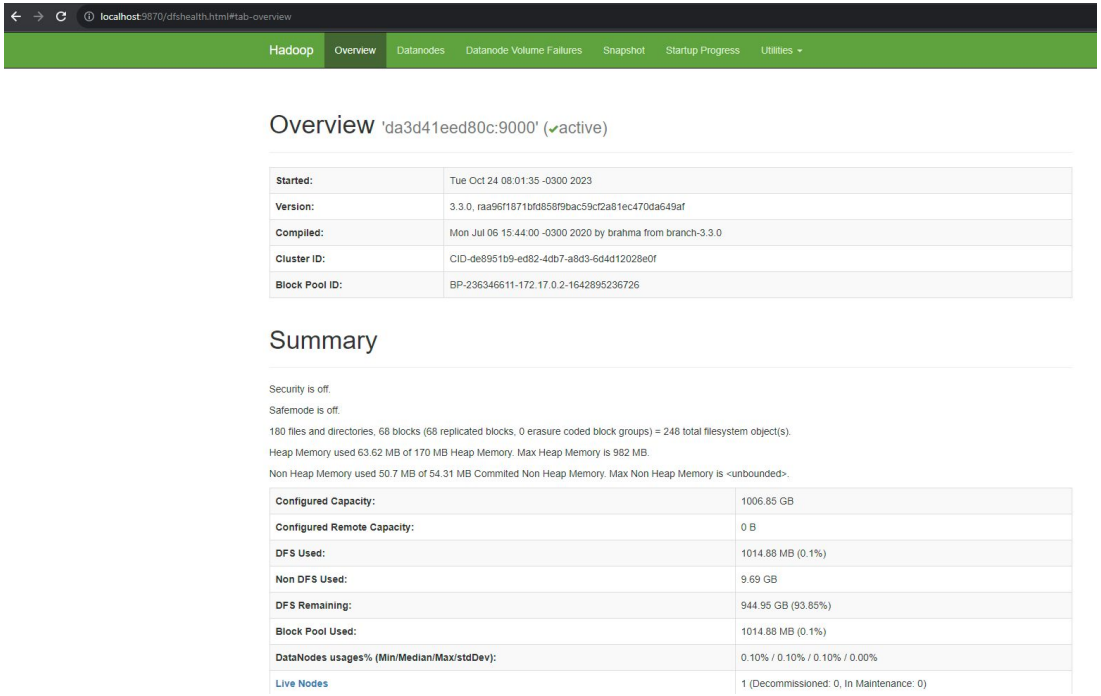
cambiar de usuario (siempre trabajar con el usr hadoop)

su hadoop

Docker Hadoop

Interfaces web: Hadoop HDFS

http://localhost:9870/



Docker Hadoop

Interfaces web: SPARK

http://localhost:8080/

localhost:8080

Spark 3.2.0

Spark Master at spark://da3d41eed80c:7077

URL: spark://da3d41eed80c:7077
Alive Workers: 1
Cores in use: 2 Total, 0 Used
Memory in use: 2.8 GiB Total, 0.0 B Used
Resources in use:
Applications: 0 Running, 0 Completed
Drivers: 0 Running, 0 Completed
Status: ALIVE

Workers (1)

Worker Id	Address	State	Cores	Memory	Resources
worker-20231024080200-172.17.0.2-43513	172.17.0.2:43513	ALIVE	2 (0 Used)	2.8 GiB (0.0 B Used)	

Running Applications (0)

Application ID	Name	Cores	Memory per Executor	Resources Per Executor	Submitted Time	User	State	Duration

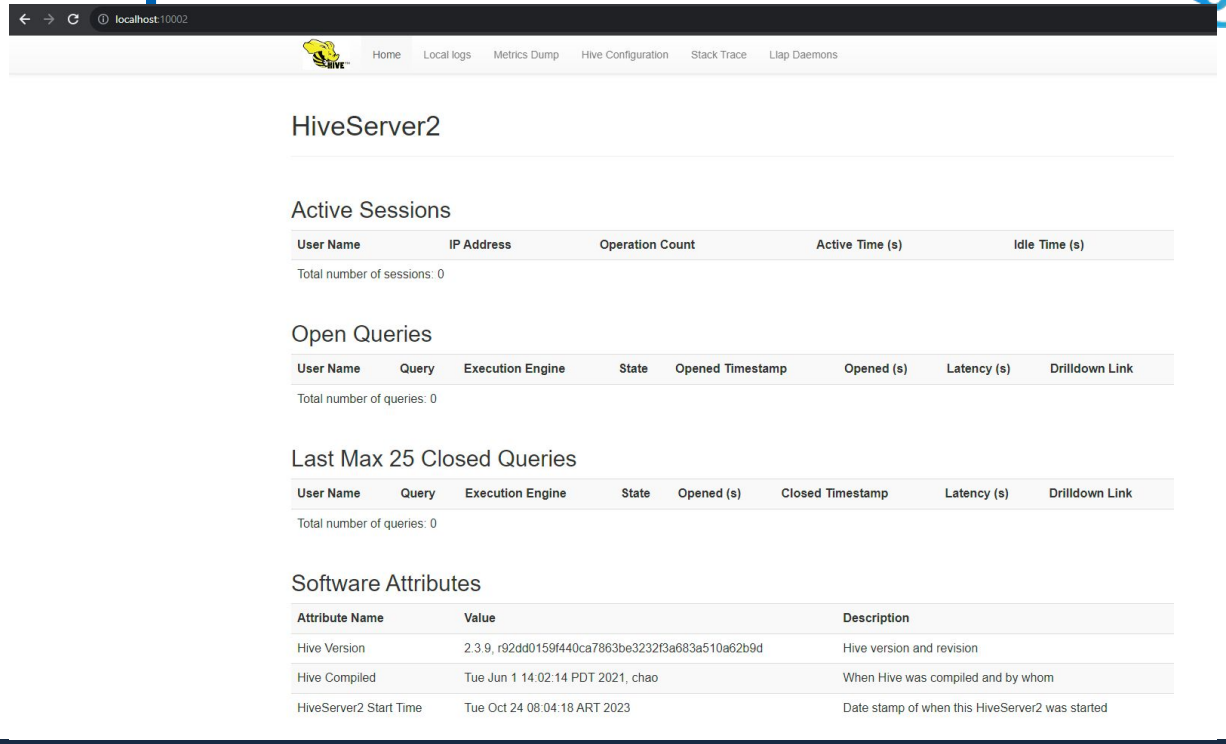
Completed Applications (0)

Application ID	Name	Cores	Memory per Executor	Resources Per Executor	Submitted Time	User	State	Duration

Docker Hadoop

Interfaces web: HIVE

http://localhost:10002/



Docker Hadoop

http://localhost:8088

Interfaces web: YARN

localhost:8088/cluster

hadoop

All Applications

- Cluster
  - About
  - Nodes
  - Node Labels
  - Applications
    - NEW
    - NEW_SAVING
    - SUBMITTED
    - ACCEPTED
    - RUNNING
    - FINISHED
    - FAILED
    - KILLED
  - Scheduler
- Tools

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Reserved	VCores Us
0	0	0	0	0	0 B	4.50 GB	0 B	0

Cluster Nodes Metrics

Active Nodes	Decommissioning Nodes	Decommissioned Nodes	Lost Nodes	Unhealthy Nodes	Reboote
1	0	0	0	0	0

Scheduler Metrics

Scheduler Type	Scheduling Resource Type	Minimum Allocation	Maximum Allocation	
Capacity Scheduler	[memory-mb (unit=Mi), vcores]	<memory:1536, vCores:1>	<memory:4608, vCores:4>	0

Show 20 entries

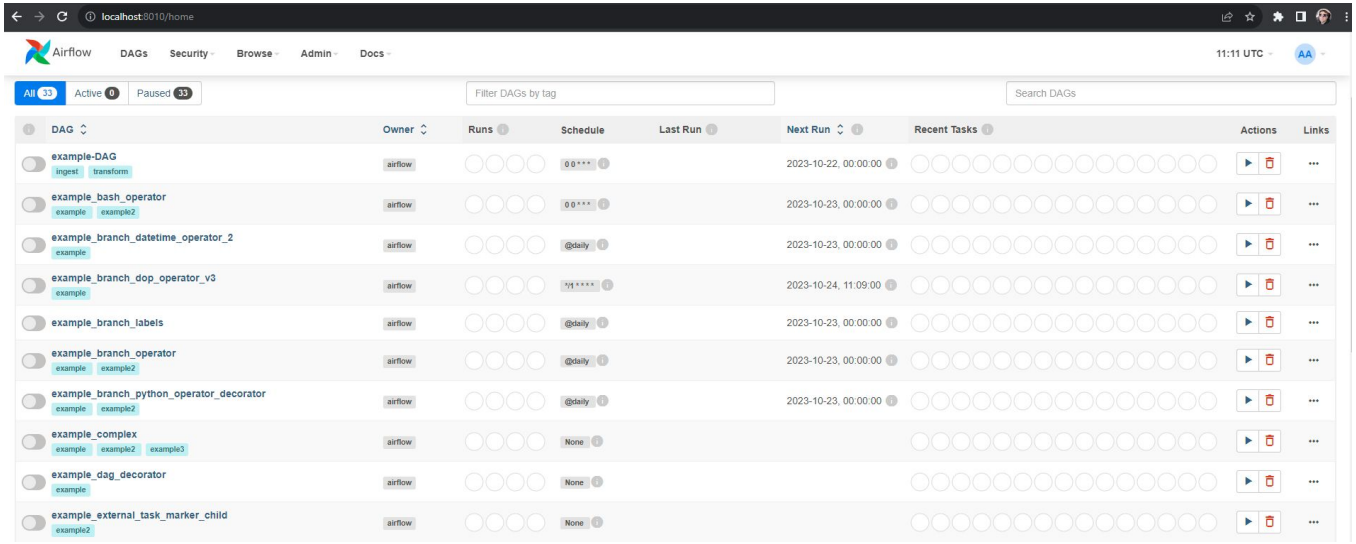
ID	User	Name	Application Type	Application Tags	Queue	Application Priority	StartTime	LaunchTime	FinishTime	State	FinalStatus	Running Containers	Allocated CPU VCores	Allocated Memory MB	Reserved CPU VCores	Reserved Memory MB	% Q
No data available in table																	

Showing 0 to 0 of 0 entries

Docker Hadoop

http://localhost:8010

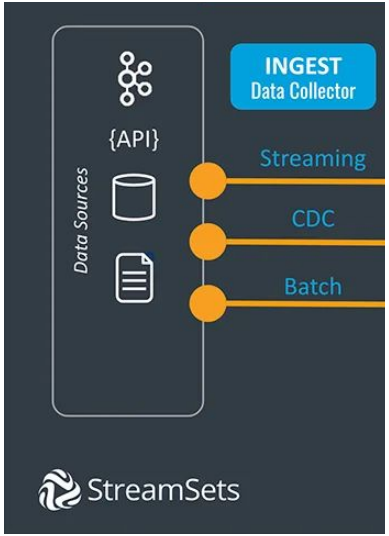
Interfaces web: AIRFLOW



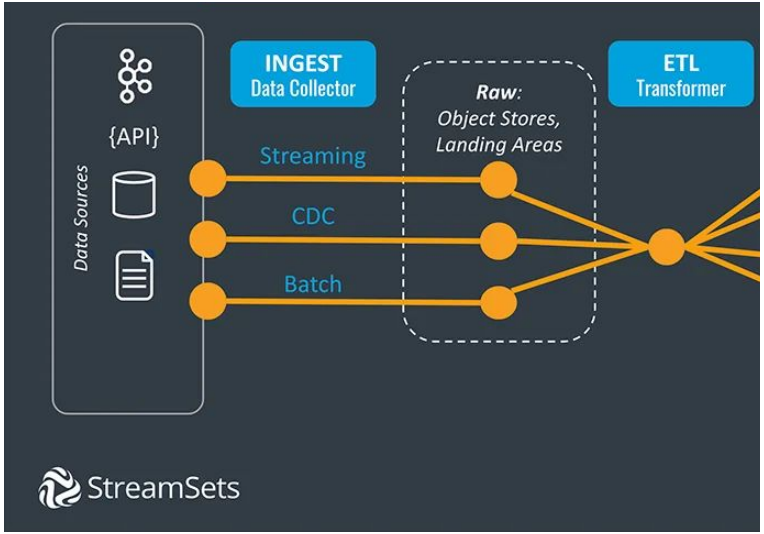
Usr: airflow
Pass: airflow

Ecosistema Hadoop

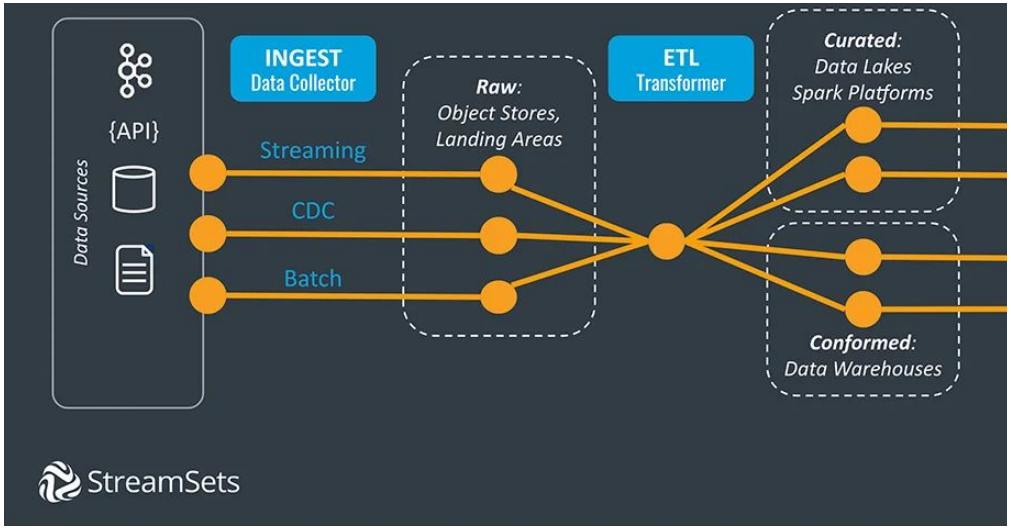
Arquitectura Big Data



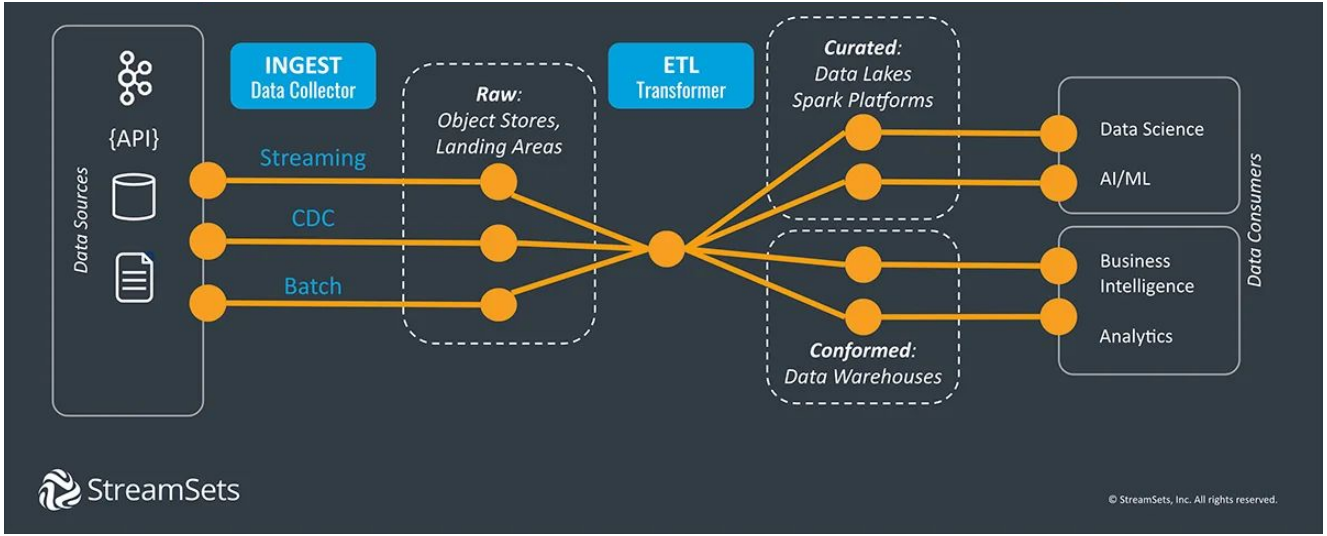
Arquitectura Big Data



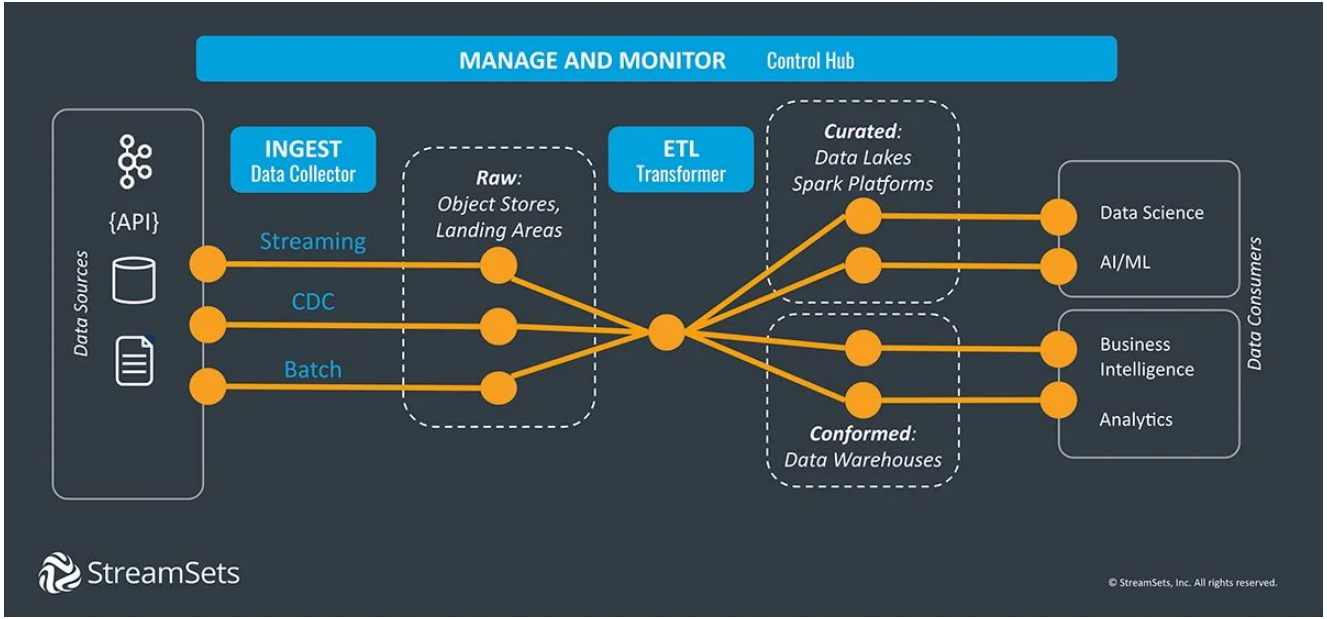
Arquitectura Big Data



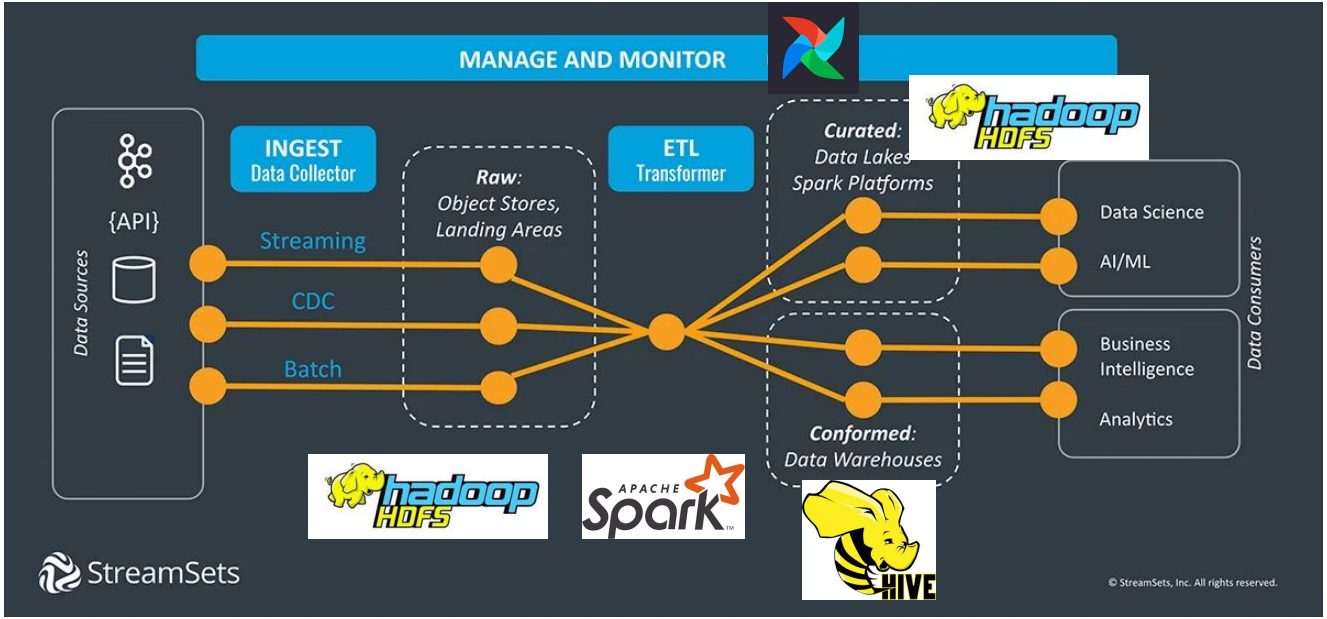
Arquitectura Big Data



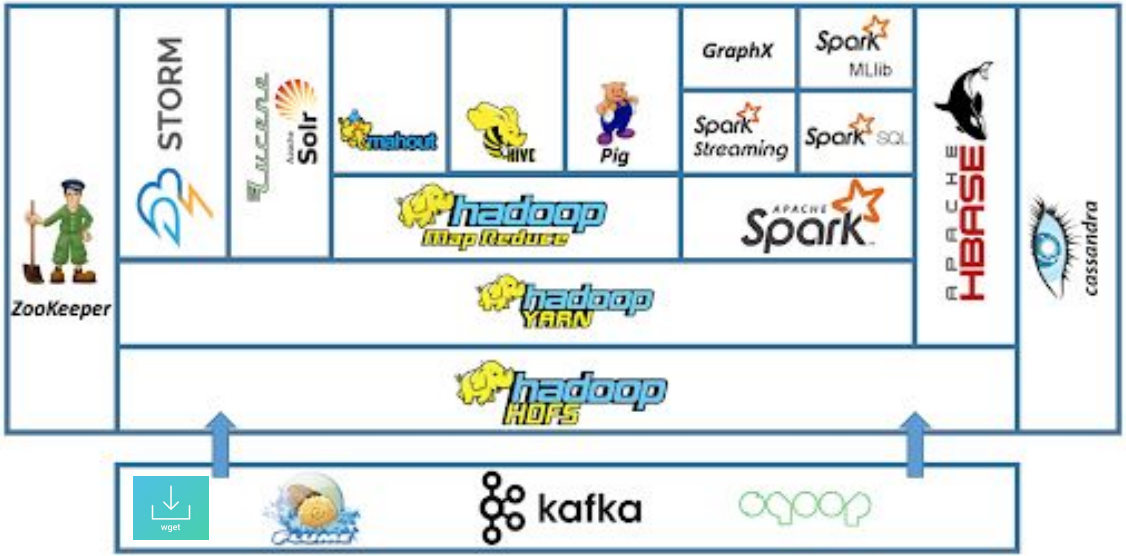
Arquitectura Big Data



Arquitectura Big Data

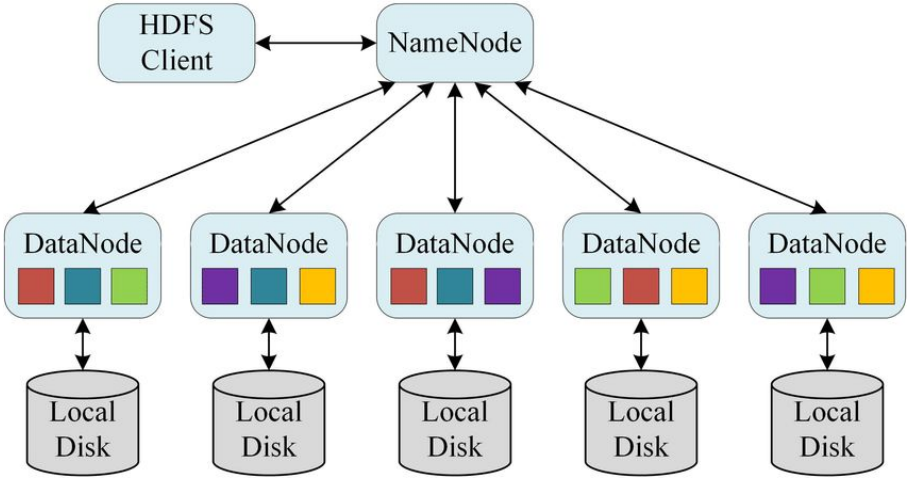


Ecosistema Hadoop



HDFS (Hadoop file system)

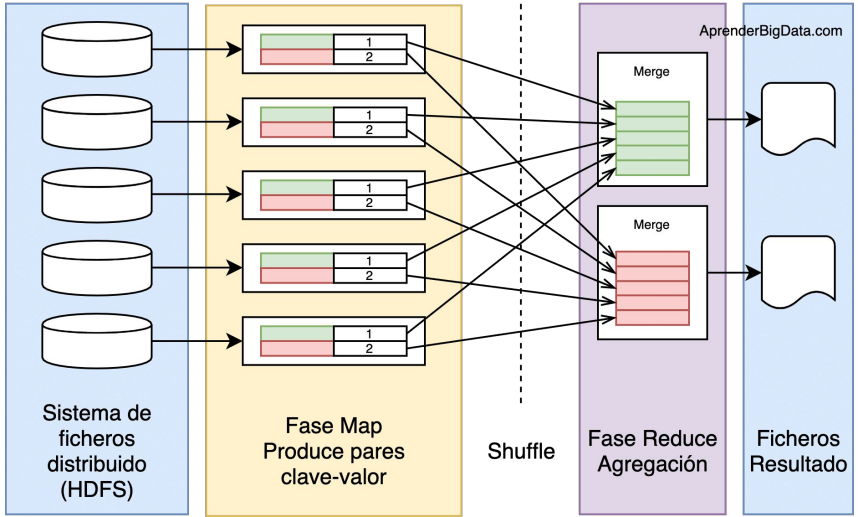
- Almacenamiento con tolerancia a fallos
- Almacena en bloques de 128 MB (configurable) en los nodos del cluster
- Escalamiento horizontal (agregar más HDDs o nodos)
- Integridad: almacena 3 copias de cada bloque de datos
- Name Node: gestiona el acceso a los datos y los metadatos, no almacena datos en sí.
- Data Node: nodos del cluster que almacenan información en sus HDDs
- Write once read many: no se pueden editar ficheros almacenados HDFS, pero sí se pueden añadir datos.

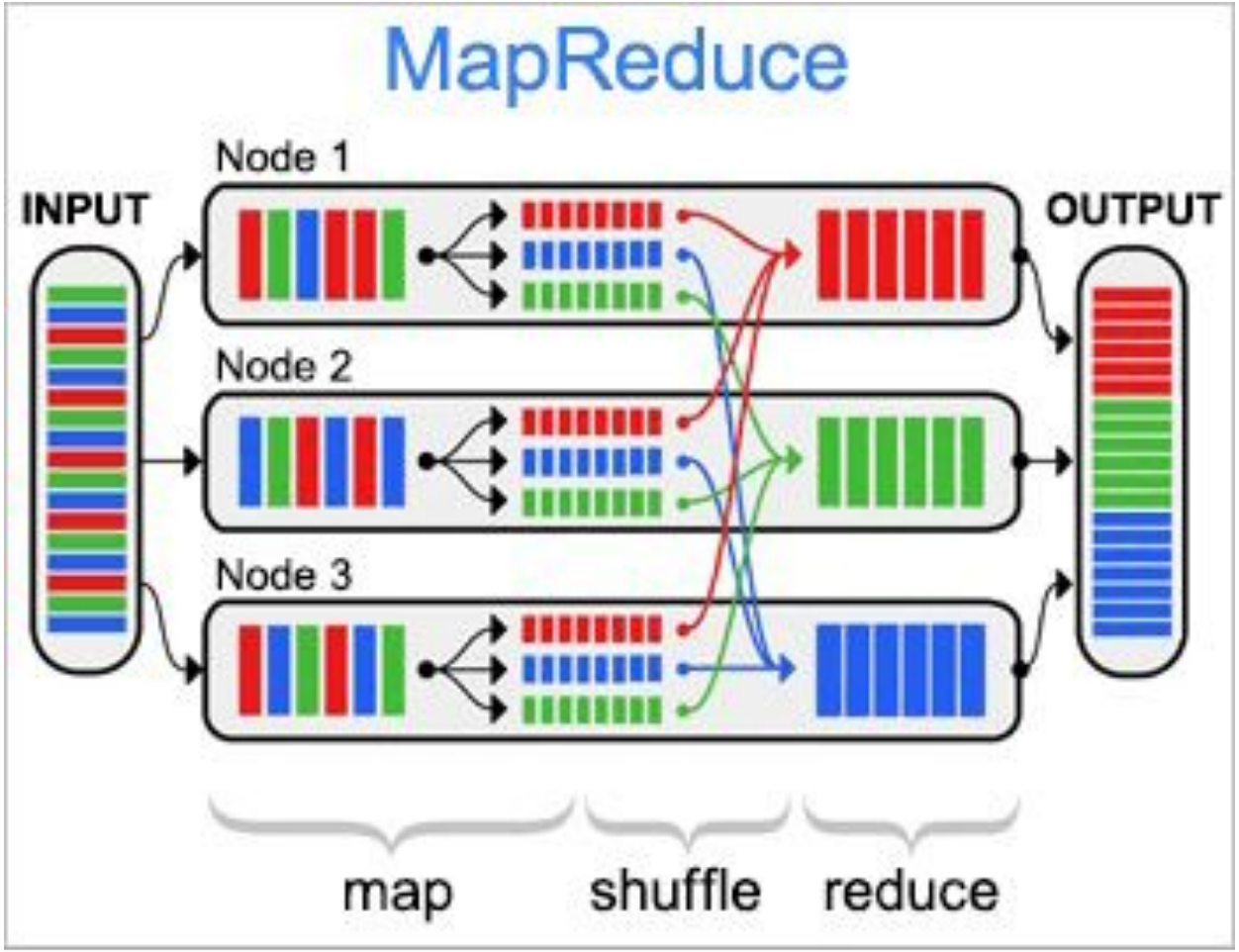


MapReduce

- **Map**: se ejecuta en subtareas llamadas mappers. Estos componentes son los responsables de **generar pares clave-valor** filtrando, agrupando, ordenando o transformando los datos originales. Los pares de datos intermedios, no se almacenan en HDFS.
- **Shuffle:** (sort) puede no ser necesaria. Es el paso intermedio entre Map y reduce que ayuda a recoger los datos y **ordenarlos** de manera conveniente para el procesamiento. Con esta fase, se pretende agregar las ocurrencias repetidas en cada uno de los mappers.
- **Reduce:** gestiona la **agregación** de los valores producidos por todos los mappers del sistema (o por shuffle) de tipo clave-valor en función de su clave. Por último, cada reducer **genera su archivo** de salida de forma independiente, generalmente **escrito en HDFS**.

Es un paradigma de procesamiento distribuido de datos caracterizado por dividirse en dos fases: Map y Reduce




MapReduce
INPUT
Node 1
Node 2
Node 3
OUTPUT
map
shuffle
reduce
ComputerHope.com

Map

custId	month	amt	ptype
123098	1	23010.70	Cred
123987	1	1320.50	Cash
123098	2	1500.00	Cash
123098	3	2450.99	Cred
123987	3	1500.00	Cred

Map

```
123098: [23010.70 1500.00 2450.99]
123987: [1320.50 1500.00]
```

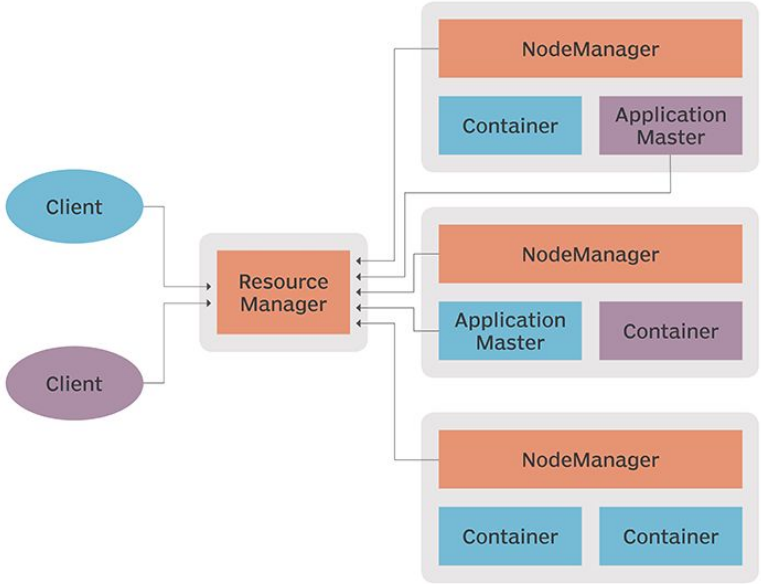
Reduce

```
123098:26961.69
123987:2820.50
```

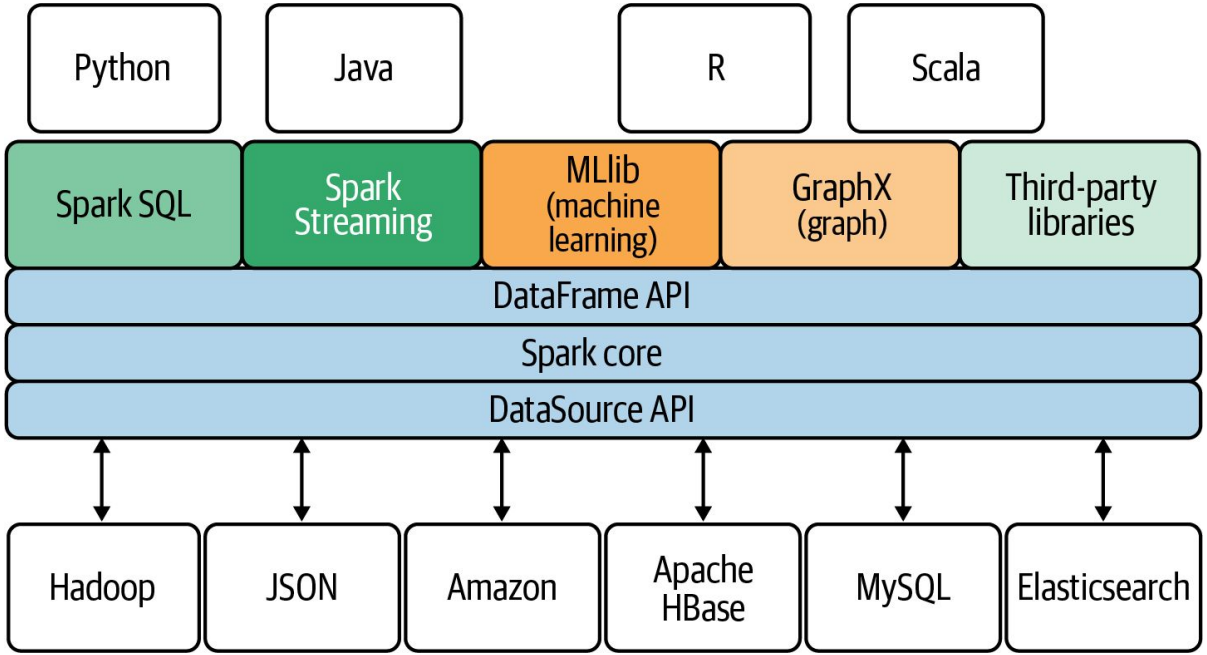
Yarn (Yet Another Resource Negotiator)

Apache Hadoop YARN descentraliza la ejecución y el monitoreo de los trabajos de procesamiento al separar las diversas responsabilidades en estos componentes:

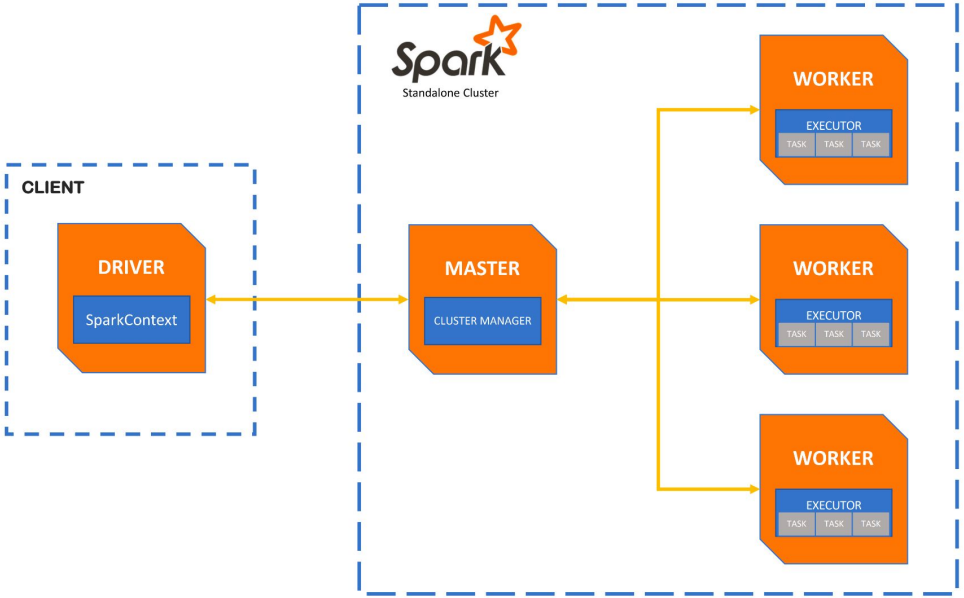
- **ResourceManager:** acepta envíos de trabajos de los usuarios, programa los trabajos y les asigna recursos.
- **NodeManager:** funciona como un agente de supervisión y presentación de informes del ResourceManager
- **ApplicationMaster:** negocia recursos y trabaja con NodeManager para ejecutar y monitorear tareas.
- **Contenedores:** controlados por NodeManagers y asigna los recursos del sistema (CPU cores, RAM, disks) a aplicaciones individuales.



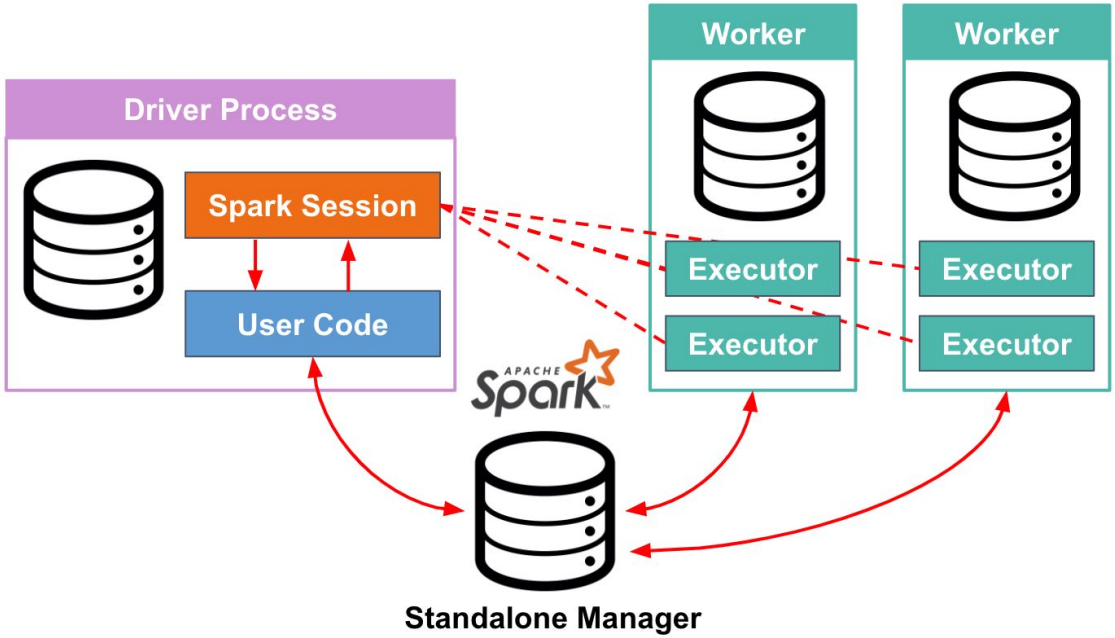
Arquitectura Spark



Spark Master & Workers

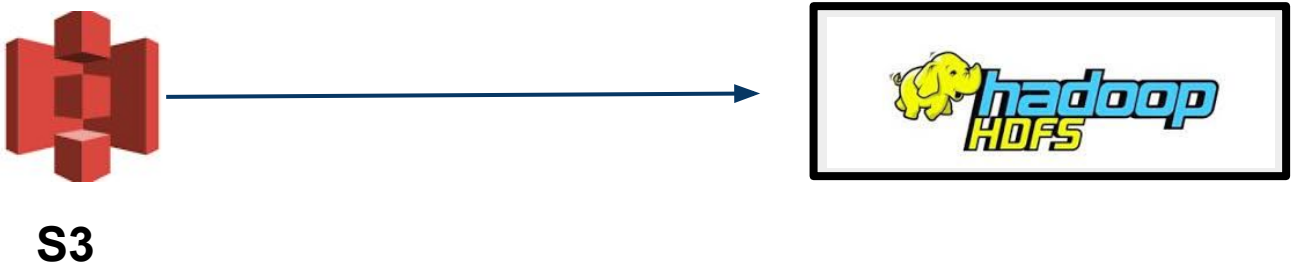


Spark Session



Ingest

Ingest con WGET



Ingest mediante scripts

Podemos utilizar algunos comandos de linux para hacer ingest de archivos.

Obtenemos los archivos con WGET:

- **wget -P /home/hadoop/landing**

  https://data-engineer-edvai.s3.amazonaws.com/yellow_tripdata_2021-01.csv

Movemos los archivos a HDFS:

- **hdfs dfs -put /home/hadoop/landing/yellow_tripdata_2021-01.csv /ingest**

Ingest mediante sqoop

Verificar funcionamiento y versión:

- **sqoop-version**

```
hadoop@5dc251dd43fb:~$ sqoop-version
Warning: /usr/lib/sqoop/../hbase does not exist! HBase imports will fail.
Please set $HBASE_HOME to the root of your HBase installation.
Warning: /usr/lib/sqoop/../hcatalog does not exist! HCatalog jobs will fail.
Please set $HCAT_HOME to the root of your HCatalog installation.
Warning: /usr/lib/sqoop/../accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
Warning: /usr/lib/sqoop/../zookeeper does not exist! Accumulo imports will fail.
Please set $ZOOKEEPER_HOME to the root of your Zookeeper installation.
2023-03-16 19:02:28,767 INFO sqoop.Sqoop: Running Sqoop version: 1.4.7
Sqoop 1.4.7
git commit id 2328971411f57f0cb683dfb79d19d4d19d185dd8
Compiled by maugli on Thu Dec 21 15:59:58 STD 2017
hadoop@5dc251dd43fb:~$
```

Ingest mediante sqoop

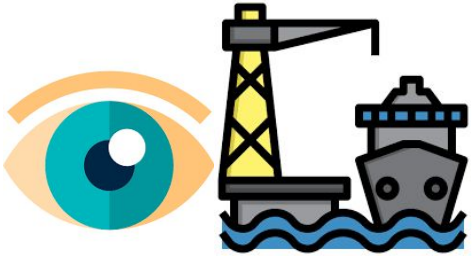
Listar databases:

**sqoop list-databases **
**—connect jdbc:postgresql://172.17.0.3:5432/northwind **
—username postgres -P

```
2023-03-16 20:36:39,489 INFO sqoop.Sqoop: Running Sqoop version: 1.4.7
Enter password:
2023-03-16 20:36:42,458 INFO manager.SqlManager: Using default fetchSize of 1000
postgres
northwind
template1
template0
hadoop@bdc251dd43fb:/$
```

Ingest mediante sqoop

Listar tablas:

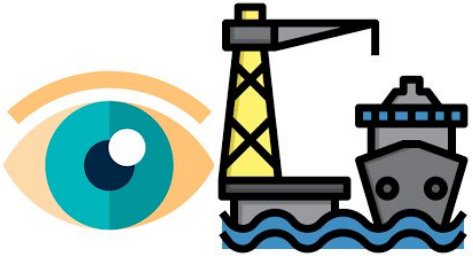


**sqoop list-tables **
**—connect jdbc:postgresql://172.17.0.3:5432/northwind **
—username postgres -P

```
Enter password:
2023-03-16 19:05:58.822 INFO manager.SqlManager: Using default fetchSize of 1000
territories
order_details
employee_territories
us_states
customers
orders
employees
shippers
products
categories
suppliers
region
customer_demographics
customer_customer_demo
hadoop@5dc251dd43fb:~$
```

Ingest mediante sqoop

Ejecutar Queries:

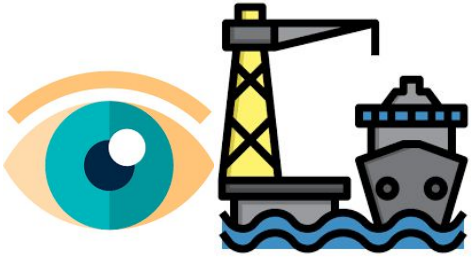


**sqoop eval **
**—connect jdbc:postgresql://172.17.0.3:5432/northwind **
**—username postgres **
**—P **
—query "select * from region limit 10"

```
Enter password:
2023-03-16 19:47:52,266 INFO manager.SqlManager: Using default fetchSize of 1000
----------------------------------
| region_id | region_description   |
----------------------------------
| 1         | Eastern              |
| 2         | Western              |
| 3         | Northern             |
| 4         | Southern             |
----------------------------------
```

Ingest mediante sqoop

Importar tablas:



```
sqoop import \
  –connect jdbc:postgresql://172.17.0.3:5432/northwind \
  –username postgres\
  –table region\
  –m 1 \
  –P \
  –target-dir /sqoop/ingest \
  –as-parquetfile \
  –delete-target-dir
```

```
                Total time spent by all maps in occupied slots (ms)=8675
                Total time spent by all reduces in occupied slots (ms)=0
                Total time spent by all map tasks (ms)=8675
                Total vcore-milliseconds taken by all map tasks=8675
                Total megabyte-milliseconds taken by all map tasks=13324800
        Map-Reduce Framework
                Map input records=4
                Map output records=4
                Input split bytes=87
                Spilled Records=0
                Failed Shuffles=0
                Merged Map outputs=0
                GC time elapsed (ms)=66
                CPU time spent (ms)=4570
                Physical memory (bytes) snapshot=275968000
                Virtual memory (bytes) snapshot=2981875712
                Total committed heap usage (bytes)=180355072
                Peak Map Physical memory (bytes)=275968000
                Peak Map Virtual memory (bytes)=2981875712
        File Input Format Counters
                Bytes Read=0
        File Output Format Counters
                Bytes Written=0
2023-03-16 20:06:32,380 INFO mapreduce.ImportJobBase: Transferred 1.8496 KB in 38.8773 seconds (48.7174 bytes/sec)
2023-03-16 20:06:32,391 INFO mapreduce.ImportJobBase: Retrieved 4 records.
```

Ingest mediante sqoop

Importar tablas con filtro:

```
sqoop import \
  –connect jdbc:postgresql://172.17.0.3:5432/northwind \
  –username postgres\
  –table region\
  –m 1 \
  –P \
  –target-dir /sqoop/ingest/southern \
  –as-parquetfile \
  –where "region_description = 'Southern' " \
  –delete-target-dir
```

```
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=10
                HDFS: Number of bytes read erasure-coded=0
        Job Counters
                Launched map tasks=1
                Other local map tasks=1
                Total time spent by all maps in occupied slots (ms)=8319
                Total time spent by all reduces in occupied slots (ms)=0
                Total time spent by all map tasks (ms)=8319
                Total vcore-milliseconds taken by all map tasks=8319
                Total megabyte-milliseconds taken by all map tasks=12777984
        Map-Reduce Framework
                Map input records=1
                Map output records=1
                Input split bytes=87
                Spilled Records=0
                Failed Shuffles=0
                Merged Map outputs=0
                GC time elapsed (ms)=74
                CPU time spent (ms)=4060
                Physical memory (bytes) snapshot=254566400
                Virtual memory (bytes) snapshot=2971721728
                Total committed heap usage (bytes)=181403648
                Peak Map Physical memory (bytes)=254566400
                Peak Map Virtual memory (bytes)=2971721728
        File Input Format Counters
                Bytes Read=0
        File Output Format Counters
                Bytes Written=0
2023-03-16 20:20:21,436 INFO mapreduce.ImportJobBase: Transferred 1.8115 KB in 30.653 seconds (60.5161 bytes/sec)
2023-03-16 20:20:21,447 INFO mapreduce.ImportJobBase: Retrieved 1 records.
```

Ingest mediante sqoop

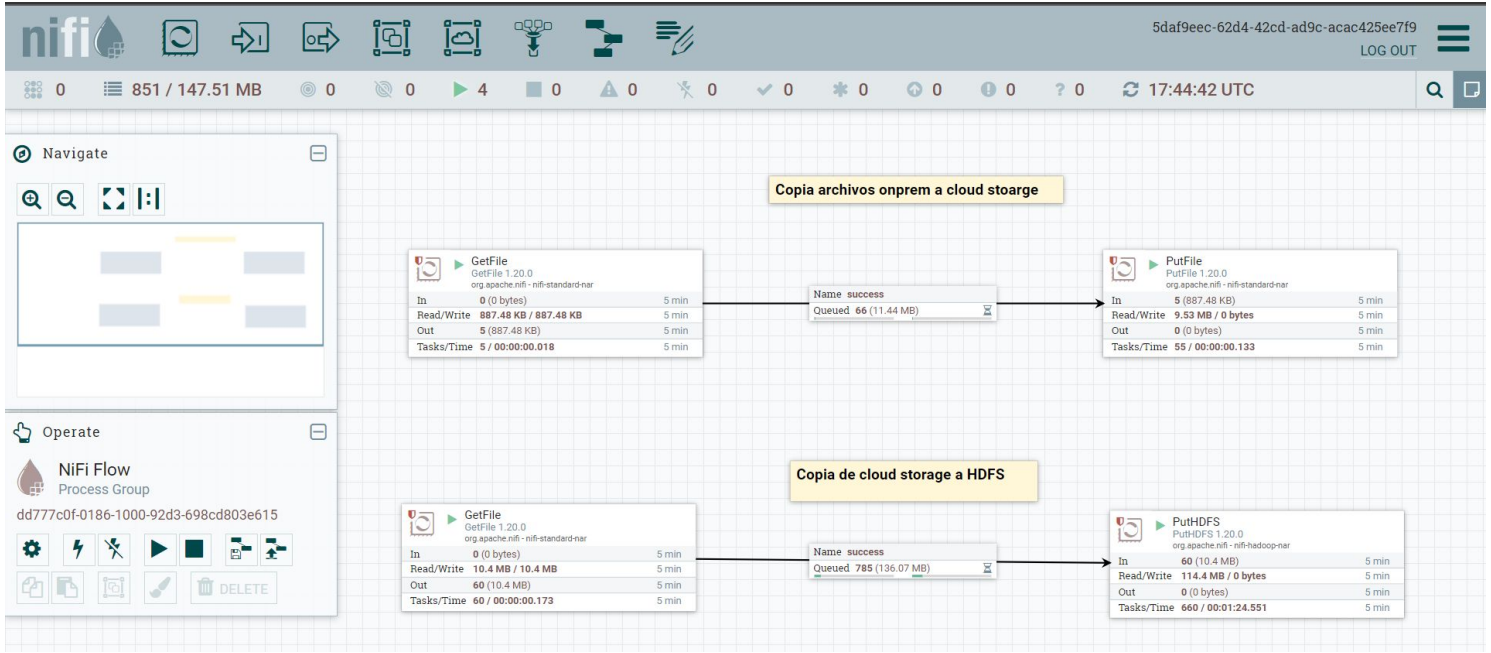
Importar tablas desde una query:

```
sqoop import \
  –connect jdbc:postgresql://172.17.0.3:5432/northwind \
  –username postgres\
  –query "select * from region where region_id = 3 AND \$CONDITIONS"\
  –m 1 \
  –P \
  –target-dir /sqoop/ingest \
  –as-parquetfile \
  –delete-target-dir
```

```
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=10
                HDFS: Number of bytes read erasure-coded=0
        Job Counters
                Launched map tasks=1
                Other local map tasks=1
                Total time spent by all maps in occupied slots (ms)=8319
                Total time spent by all reduces in occupied slots (ms)=0
                Total time spent by all map tasks (ms)=8319
                Total vcore-milliseconds taken by all map tasks=8319
                Total megabyte-milliseconds taken by all map tasks=12777984
        Map-Reduce Framework
                Map input records=1
                Map output records=1
                Input split bytes=87
                Spilled Records=0
                Failed Shuffles=0
                Merged Map outputs=0
                GC time elapsed (ms)=74
                CPU time spent (ms)=4060
                Physical memory (bytes) snapshot=254566400
                Virtual memory (bytes) snapshot=2971721728
                Total committed heap usage (bytes)=1814036648
                Peak Map Physical memory (bytes)=254566400
                Peak Map Virtual memory (bytes)=2971721728
        File Input Format Counters
                Bytes Read=0
        File Output Format Counters
                Bytes Written=0
2023-03-16 20:20:21,436 INFO mapreduce.ImportJobBase: Transferred 1.8115 KB in 30.653 seconds (60.5161 bytes/sec)
2023-03-16 20:20:21,447 INFO mapreduce.ImportJobBase: Retrieved 1 records.
```

APACHE nifi

APACHE nifi


nifi
5daf9eec-62d4-42cd-ad9c-acac425ee7f9
LOG OUT
0
851 / 147.51 MB
0
0
4
0
0
0
0
0
0
0
17:44:42 UTC
Navigate
Operate
NiFi Flow
Process Group
dd777c0f-0186-1000-92d3-698cd803e615
DELETE
Copia archivos onprem a cloud stoarge
GetFile
GetFile 1.20.0
org.apache.nifi - nifi-standard-nar
In 0 (0 bytes) 5 min
Read/Write 887.48 KB / 887.48 KB 5 min
Out 5 (887.48 KB) 5 min
Tasks/Time 5 / 00:00:00.018 5 min
Name success
Queued 66 (11.44 MB)
PutFile
PutFile 1.20.0
org.apache.nifi - nifi-standard-nar
In 5 (887.48 KB) 5 min
Read/Write 9.53 MB / 0 bytes 5 min
Out 0 (0 bytes) 5 min
Tasks/Time 55 / 00:00:00.133 5 min
Copia de cloud storage a HDFS
GetFile
GetFile 1.20.0
org.apache.nifi - nifi-standard-nar
In 0 (0 bytes) 5 min
Read/Write 10.4 MB / 10.4 MB 5 min
Out 60 (10.4 MB) 5 min
Tasks/Time 60 / 00:00:00.173 5 min
Name success
Queued 785 (136.07 MB)
PutHDFS
PutHDFS 1.20.0
org.apache.nifi - nifi-hadoop-nar
In 60 (10.4 MB) 5 min
Read/Write 114.4 MB / 0 bytes 5 min
Out 0 (0 bytes) 5 min
Tasks/Time 660 / 00:01:24.551 5 min

APACHE nifi

GetFile

Processor Details

Running — STOP & CONFIGURE

SETTINGS | SCHEDULING | PROPERTIES | RELATIONSHIPS | COMMENTS

Required field

Property	Value
Input Directory	/home/nifi/ingest
File Filter	starwars.csv
Path Filter	No value set
Batch Size	10
Keep Source File	true
Recurse Subdirectories	true
Polling Interval	0 sec
Ignore Hidden Files	true
Minimum File Age	0 sec
Maximum File Age	No value set
Minimum File Size	0 B
Maximum File Size	No value set

APACHE nifi

PutFile

Processor Details

Running

STOP & CONFIGURE

SETTINGS | SCHEDULING | PROPERTIES | RELATIONSHIPS | COMMENTS

Required field

Property	Value
Directory	/home/nifi/bucket
Conflict Resolution Strategy	replace
Create Missing Directories	true
Maximum File Count	No value set
Last Modified Time	No value set
Permissions	No value set
Owner	No value set
Group	No value set

APACHE nifi

PutHDFS

Processor Details

Running (1) STOP & CONFIGURE

SETTINGS | SCHEDULING | PROPERTIES | RELATIONSHIPS | COMMENTS

Required field

Property	Value
Hadoop Configuration Resources	/home/nifi/hadoop/core-site.xml, /home/nifi/hadoop...
Kerberos Credentials Service	No value set
Kerberos User Service	No value set
Kerberos Principal	No value set
Kerberos Keytab	No value set
Kerberos Password	No value set
Kerberos Relogin Period	4 hours
Additional Classpath Resources	No value set
Directory	**/nifi**
Conflict Resolution Strategy	**replace**
Writing Strategy	**Write and rename**
Block Size	No value set

APACHE nifi

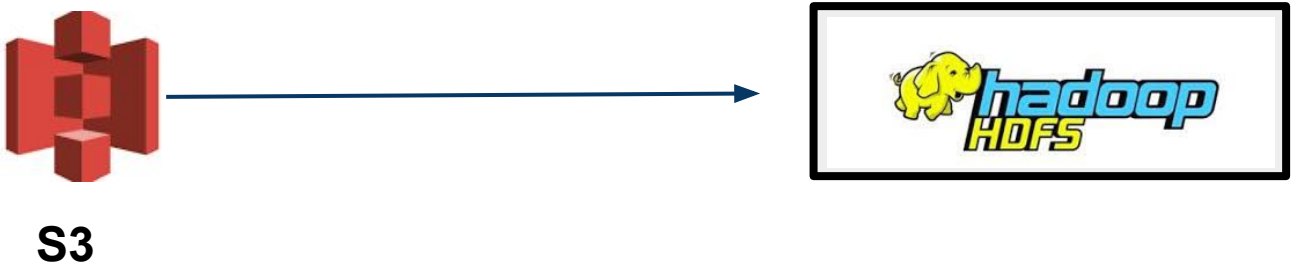
- **Instalación**:
  - Instalado en la VM
  - instalar desde docker (docker pull apache/nifi)
  - docker run —name nifi -p 8443:8443 –dns=8.8.8.8 -d apache/nifi:latest
- **Usr y contraseña:**
  - Usr: d30eb1a2-3bfe-4c85-9ea4-9562915a70e6
  - Pass: NvxFSKesWIiU1K4XL1AQJwovv9z7TW4h
  - /opt/nifi/nifi-current/bin nifi.sh set-single-user-credentials nifi <password>
  - En caso que lo instalen desde docker buscar el usr y pass en docker logs nifi (docker logs nifi | grep Generated)
- **Archivos de configuración Hadoop:**
  - core-site.xml: https://github.com/fpineyro/homework-0/blob/2767f00cf9c16774dbb10fc2d7b8d17f11114750/core-site.xml
  - hdfs-site.xml: https://github.com/fpineyro/homework-0/blob/2767f00cf9c16774dbb10fc2d7b8d17f11114750/hdfs-site.xml

Ejercicio

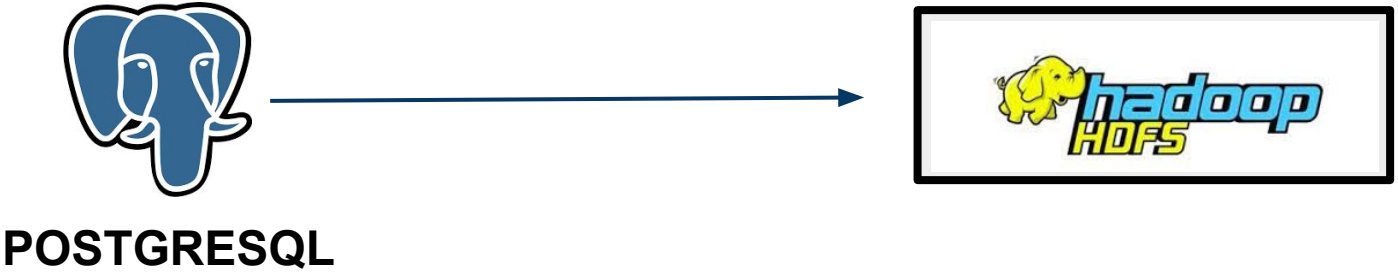
Ejercicios

- Ingest
  - WGET
  - HDFS DFS -PUT
  - SQOOP
  - NIFI

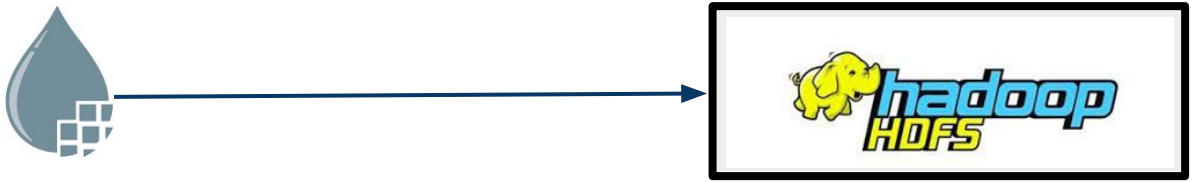
Ingest con WGET



Ingest con SQOOP



Ingest con APACHE nifi



APACHE nifi