

Manual de Instalación de Apache Hive 4.0.1

Entorno Pseudo-Distribuido

BSG Institute

Programa: Certified AI Data Engineer

Curso: Procesos ETL para Workloads de AI

Fecha: 04 de November de 2025

1. Objetivo del laboratorio

Este laboratorio tiene como objetivo que el estudiante sea capaz de:

- Eliminar completamente una instalación previa de Apache Hive.
- Realizar una instalación limpia de Apache Hive 4.0.1 sobre Hadoop 3.4.2 y Java 8.
- Inicializar y verificar el metastore embebido (Derby).
- Conectarse a Hive mediante Beeline y ejecutar comandos básicos.

2. Requisitos previos

Antes de iniciar, el estudiante debe contar con:

- Sistema operativo tipo Linux (Ubuntu recomendado).
- Java 8 instalado y configurado (JAVA_HOME apuntando a Java 8).
- Hadoop 3.4.2 instalado y funcional (start-dfs.sh y start-yarn.sh funcionando).
- Usuario con permisos de sudo para administrar /usr/local.

3. Visión general del proceso

El proceso se divide en dos grandes fases:

- Fase 1: Limpieza total de cualquier instalación previa de Hive.
- Fase 2: Instalación limpia de Apache Hive 4.0.1, inicialización del metastore Derby y verificación de la conexión mediante Beeline.

Esta práctica ayuda al estudiante a entender la relación entre:

- El sistema operativo (archivos en /usr/local).
- El sistema de archivos distribuido (HDFS).
- El motor de metadatos (metastore Derby) y HiveServer2.

4. Fase 1: Limpieza total de Apache Hive

Ejecutar los siguientes pasos en una terminal:

```
# 1. Detener cualquier servicio activo de Hive  
pkill -f hiveserver2  
pkill -f metastore  
pkill -f org.apache.hive
```

```
# 2. Eliminar carpetas antiguas en el sistema operativo  
sudo rm -rf /usr/local/hive
```

```

rm -rf ~/metastore_db
rm -rf /tmp/hive*

# 3. Limpiar rutas antiguas en HDFS (si existen)
hdfs dfs -rm -r -skipTrash /user/hive/warehouse
hdfs dfs -rm -r -skipTrash /tmp/hive

# 4. Revisar y limpiar variables de entorno antiguas de Hive
nano ~/.bashrc
# -> eliminar líneas con HIVE_HOME o PATH=$PATH:$HIVE_HOME/bin
source ~/.bashrc

```

Esta fase refuerza la idea de que los componentes de Big Data tienen configuración tanto a nivel de sistema operativo como en HDFS. Se recomienda al docente pausar y discutir con los estudiantes qué parte de Hive vive en el sistema de archivos local y qué parte vive en HDFS.

5. Fase 2: Instalación limpia de Apache Hive 4.0.1

5.1 Descarga e instalación de Hive

```

cd /usr/local
sudo wget https://downloads.apache.org/hive/hive-4.0.1/apache-hive-4.0.1-
bin.tar.gz
sudo tar -xzvf apache-hive-4.0.1-bin.tar.gz
sudo mv apache-hive-4.0.1-bin hive
sudo rm apache-hive-4.0.1-bin.tar.gz

# Ajustar permisos para el usuario actual
sudo chown -R $USER:$USER /usr/local/hive

```

5.2 Configuración de variables de entorno

```

nano ~/.bashrc

# Agregar al final del archivo:
export HIVE_HOME=/usr/local/hive
export PATH=$PATH:$HIVE_HOME/bin
export HADOOP_HOME=/usr/local/Hadoop
export PATH=$PATH:$HADOOP_HOME/bin:$HADOOP_HOME/sbin

# Aplicar cambios
source ~/.bashrc

```

5.3 Configuración de hive-env.sh

```
cd $HIVE_HOME/conf
```

```
cp hive-env.sh.template hive-env.sh
nano hive-env.sh

# Añadir al final:
export HADOOP_HOME=/usr/local/Hadoop
export HIVE_CONF_DIR=$HIVE_HOME/conf
export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
```

5.4 Inicialización del metastore Derby

```
cd $HIVE_HOME
schematool -initSchema -dbType derby
```

Si el comando anterior finaliza correctamente, el metastore embebido de Hive queda creado y listo para ser utilizado. En este punto es útil mostrar a los estudiantes la carpeta metastore_db generada y explicar que allí se almacenan los metadatos (esquemas de bases de datos y tablas).

5.5 Creación de `hive-site.xml` mínimo

```
cd $HIVE_HOME/conf

cat > hive-site.xml << 'EOF'
<?xml version="1.0"?>
<configuration>
<property>
<name>hive.server2.enable.doAs</name>
<value>false</value>
</property>
<property>
<name>hive.server2.authentication</name>
<value>NONE</value>
</property>
</configuration>
EOF
```

En este archivo solo se desactiva la impersonación (doAs) y se mantiene una autenticación simple, suficiente para un entorno de laboratorio en un solo nodo.

5.6 Preparación de directorios en HDFS

```
$HADOOP_HOME/sbin/start-dfs.sh
$HADOOP_HOME/sbin/start-yarn.sh

hdfs dfs -mkdir -p /tmp /user/hive/warehouse
hdfs dfs -chmod 1777 /tmp /user/hive/warehouse

hdfs dfs -mkdir -p /user/hive/warehouse/scratchdir
```

```
hdfs dfs -chmod g+w /user/hive/warehouse/scratchdir
```

```
hdfs dfs -chmod g+w /user  
hdfs dfs -chmod g+wx /user  
hdfs dfs -chmod g+w /tmp  
hdfs dfs -chmod g+wx /tmp
```

```
hdfs dfs -mkdir -p /cursobsg  
hdfs dfs -chmod g+w /cursobsg  
hdfs dfs -chmod g+wx /cursobsg
```

```
hdfs dfs -mkdir -p /cursobsg/database  
hdfs dfs -chmod g+w /cursobsg/database  
hdfs dfs -chmod g+wx /cursobsg/database
```

```
hdfs dfs -mkdir -p /cursobsg/tables  
hdfs dfs -chmod g+w /cursobsg/tables  
hdfs dfs -chmod g+wx /cursobsg/tables
```

5.7 Inicio de HiveServer2 y conexión con Beeline

```
# En una terminal (dejar corriendo):
```

```
cd $HIVE_HOME  
hive --service hiveserver2
```

```
# En otra terminal:
```

```
beeline -u "jdbc:hive2://localhost:10000/default" -n suusuario
```

```
# Comandos de prueba en Beeline:
```

```
show databases;  
create database demo_db;  
use demo_db;  
create table ejemplo (id int, nombre string);  
show tables;
```

6. Resultados de aprendizaje esperados

Al finalizar este laboratorio, el estudiante deberá ser capaz de:

- Explicar el rol de Apache Hive dentro de un ecosistema Hadoop.
- Diferenciar entre el sistema de archivos local y HDFS en el contexto de Hive.
- Ejecutar un procedimiento completo de reinstalación de Hive ante una falla.
- Inicializar y verificar el metastore embebido (Derby).
- Conectarse a Hive mediante Beeline y ejecutar consultas básicas.

7. Actividades sugeridas para clase

- Pedir a los estudiantes que introduzcan un error intencional (por ejemplo, borrar la carpeta metastore_db) y que repitan el proceso de inicialización.
- Solicitar que creen una base de datos y varias tablas en Hive, y luego exploren en HDFS las carpetas correspondientes en /user/hive/warehouse.
- Proponer un ejercicio donde deban documentar, paso a paso, cómo recuperar un entorno de Hive dañado utilizando esta guía.

8. Errores típicos y cómo explicarlos

1. Connection refused al conectar con Beeline:
 - Causa posible: HiveServer2 no está levantado.
 - Explicación pedagógica: diferenciar entre cliente (Beeline) y servidor (HiveServer2).
2. Problemas de permisos en HDFS (/tmp o /user/hive/warehouse):
 - Mostrar cómo los permisos en HDFS afectan la ejecución de consultas.
3. Errores de metastore (schema no inicializado):
 - Reforzar el concepto de metastore como catálogo de metadatos de Hive.

9. Resumen de arranque diario

```
# Iniciar Hadoop y HiveServer2  
$HADOOP_HOME/sbin/start-dfs.sh  
$HADOOP_HOME/sbin/start-yarn.sh  
  
cd $HIVE_HOME  
hive --service hiveserver2  
  
# En otra terminal:  
beeline -u "jdbc:hive2://localhost:10000/default" -n arojaspa
```