|  |
| --- |
| Máster en Big Data |
| Tecnologías de Almacenamiento |
| 1. Hands-On: Ejecución Apache Spark |

Índice

[1. Introducción 3](#_Toc96241474)

[2. Entorno 3](#_Toc96241475)

[3. Ejecución Spark 3](#_Toc96241476)

[4. SparkUI 4](#_Toc96241477)

# Introducción

El objetivo de este Hands-On es aprender a ejecutar un job de Spark mediante un ejemplo precompilado.

# Entorno

Para la realización de este hands-on, vamos a desplegar un sandbox de MapR en VMWare.

Una vez arrancado el nodo, las credenciales son:

Usuario: *mapr*; Password: *mapr*

Busca la IP de la máquina virtual y accede a HUE http://<ip>:8888/ y abre una conexión ssh para más comodidad (puedes añadir la ip en hosts con el nombre *maprdemo*).

# Ejecución Spark

La ubicación de Spark dentro de la distribución MapR la podemos encontrar en:

/opt/mapr/spark/spark-2.1.0/

Se pide ejecutar el programa SparkPi, contra YARN en modo cliente, con una memoria por ejecutor de 1g.

El jar ya compilado lo podemos encontrar en: /opt/mapr/spark/spark-2.1.0/examples/jars/spark-examples\_2.11-2.1.0-mapr-1710.jar

Y el package y nombre de la clase es: *org.apache.spark.examples.SparkPi*

El programa acepta por parámetro el número de paralelización que queremos darle, vamos a pasarle, por ejemplo, 4.

Engancha aquí el comando para enviar el Job y una captura de pantalla con la salida.

# SparkUI

Consulta la *SparkUI* después de ejecutar el Job y responde a las siguientes preguntas.

Requisito: No accedas directamente a la interficie del job history de spark, hazlo mediante HUE (acceso user:*mapr* / pass: *mapr*)

¿Cuántas tasks se han ejecutado?

¿Cuántos executors se han ejecutado?

¿Cuántos RDDs se han creado y que transformaciones los han desencadenado?

Ejecuta el job con 2, 4 y 10 tasks y compara el rendimiento, ¿Que sucede en relación con la performance? ¿Porque se produce este comportamiento?