


Capítulo 7: Administración de Hadoop

Capítulo 8: Big Data 2.0 – Spark




División de Alta Tecnología

# 8

## Big Data 2.0 - Spark

Big Data

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.




### Objetivos

Al finalizar el capítulo, el alumno logrará:

- Describir la tecnología Spark
- Implementar un cluster Spark
- Conocer los fundamentos del lenguaje Scala y Python para ciencia de datos.

8 - 2

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Agenda

- Introducción a Spark
- Propósitos de Spark
- Componentes
- Instalación y configuración
- Scala y Python

8 - 3

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Introducción a Spark

Apache Spark es una infraestructura de cluster de código abierto usado con frecuencia para cargas de trabajo de Big Data. Además, ofrece un desempeño rápido , ya que el almacenamiento de datos se gestiona en memoria, lo que mejora el desempeño de cargas de trabajo interactivas sin costos de E/S. Por otro lado, Apache Spark es compatible con las bases de datos de gráficos, el análisis de transmisiones, el procesamiento general por lotes, las consultas ad-hoc y el Machine Learning.

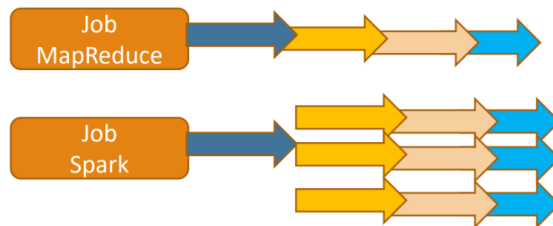
8 - 4

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Introducción a Spark

Job Spark, trabaja en paralelo vs Job Map Reduce, trabaja en secuencia.

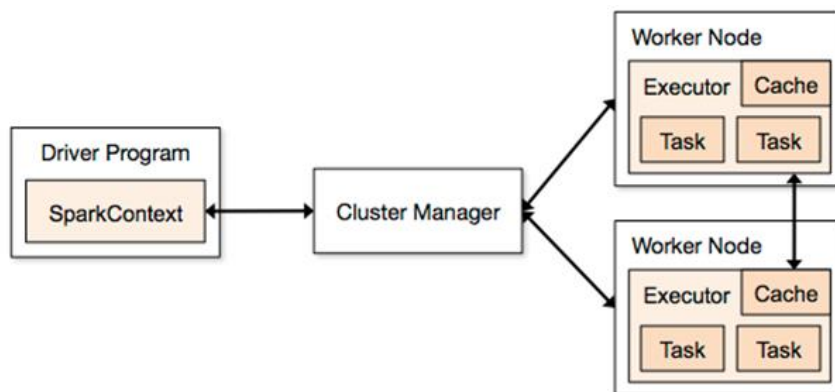


8 - 5

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Spark Arquitectura



8 - 6

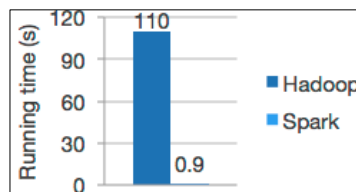
Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Propósitos de Spark

### Velocidad

- Ejecute programas hasta 100 veces más rápido que Hadoop MapReduce en la memoria, o 10 veces más rápido en el disco.
- Apache Spark tiene un motor de ejecución DAG avanzado que admite flujo de datos acíclico y cómputo en memoria.



8 - 7

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Propósitos de Spark

### Facilidad de uso

- Escribir aplicaciones rápidamente en Java, Scala, Python, R.
- Spark ofrece más de 80 operadores de alto nivel que facilitan la creación de aplicaciones paralelas. Y puede usarlo interactivamente desde las consolas de Scala, Python y R.

```
text_file = spark.textFile("hdfs://...")

text_file.flatMap(lambda line: line.split())
    .map(lambda word: (word, 1))
    .reduceByKey(lambda a, b: a+b)
```

Word count in Spark's Python API

8 - 8

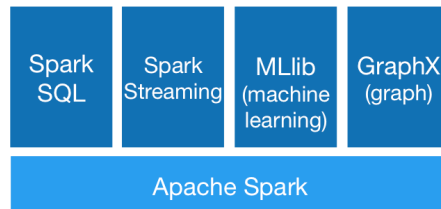
Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Propósitos de Spark

### Generalidad

- Combina SQL, transmisión y análisis complejos.
- Spark alimenta una pila de bibliotecas que incluyen SQL y DataFrames, MLlib para aprendizaje automático, GraphX y Spark Streaming. Puede combinar estas bibliotecas a la perfección en la misma aplicación.



8 - 9

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Propósitos de Spark

### Corre por todas partes

- Spark se ejecuta en Hadoop, Mesos, Kubernetes, de forma independiente o en la nube. Puede acceder a diversas fuentes de datos, incluidos HDFS, Cassandra, HBase y S3.
- Puede ejecutar Spark usando su modo de clúster independiente, en EC2, en HADOOP YARN, en Apache Mesos o en Kubernetes. Acceda a los datos en HDFS, Cassandra, HBase, Hive, Tachyon y cualquier fuente de datos de Hadoop.



8 - 10

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Propósitos de Spark

### Comunidad

- Spark se utiliza en una amplia gama de organizaciones para procesar grandes conjuntos de datos. Puede encontrar muchos ejemplos de casos de uso en la página Powered By.
- Hay muchas formas de llegar a la comunidad:
  - Use las listas de correo para hacer preguntas.
  - Los eventos en persona incluyen numerosos grupos Meetup y conferencias.
  - Usamos JIRA para el seguimiento de problemas.
  - <https://spark.apache.org/community.html>

8 - 11

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Propósitos de Spark

### Colaboradores

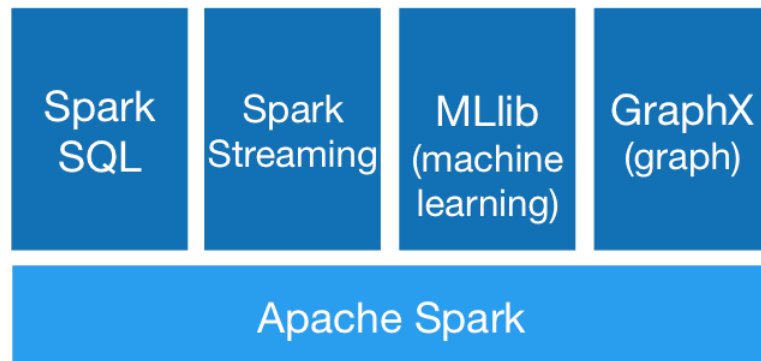
- Apache Spark está construido por un amplio conjunto de desarrolladores de más de 200 compañías. Desde 2009, ¡más de 1000 desarrolladores han contribuido a Spark!
- Los committers del proyecto provienen de más de 20 organizaciones. (Databricks, Palantir, Uber, University of Michigan, **IBM**, **Intel**, UC Berkeley, **Facebook**, Red Hat, **Cloudera**, QuestTec B.V., Nexstar Digital, **Hortonworks**, Quantifind, Webtrends, Princeton University, **Google**, Remix, NTT Data, **Netflix**, **Huawei**, **Alibaba**, entre otros)

8 - 12

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Componentes de Spark



8 - 13

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Componentes Spark Core

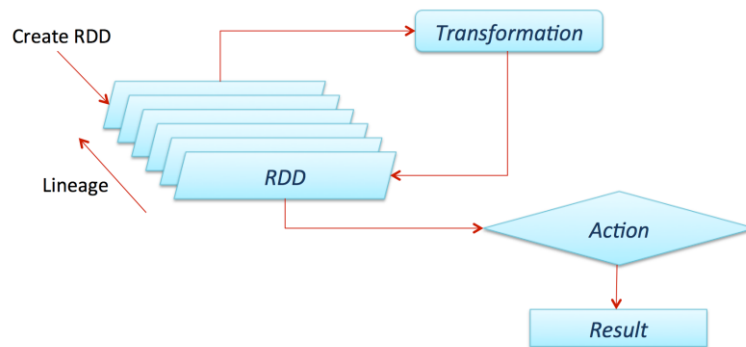
- El motor base para el procesamiento en escala y distribuido
- Aunque está construido en Scala, hay APIs para Python, Java y R.
- Se encarga entre otras cosas de:
  - Gestión de la memoria
  - Recuperación ante fallos
  - Planificación, distribución de trabajos en el cluster
  - Monitorizar trabajo
  - Accedes a los sistemas de almacenamiento

8 - 14

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Componentes Spark Core



8 - 15

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Componentes Spark Core - RDD

- La principal abstracción de datos en Spark es el RDD (Resilient Distributed Dataset)
- Un RDD representa una colección de items que pueden ser distribuidos en los nodos de computo
- Los APIs disponibles para trabajar con RDDs son Java, Python y Scala
- Operaciones en RDD:
  - Transformaciones: crea un nuevo RDD aplicando una transformación a un RDD existente
  - Acciones: retorna los resultados al driver o a un archivo de salida

8 - 16

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.





## Componentes Spark Streaming

- Se usa para procesar fuentes de datos en tiempo real (streaming data).
- Permite procesar con una alta tolerancia a fallos y un gran rendimiento las fuentes “vivas” de información que le suministremos.
- Su unidad fundamental de trabajo es el Dstream (serie de RDDs, que veremos posteriormente).



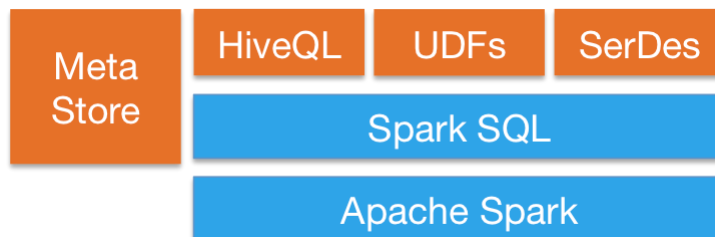
8 - 17

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Componentes Spark Sql

- Permite integrar comandos y componentes relacionales junto con la programación funcional de Spark.
- Podemos usar SQL o Hive Query Language.
- Permite el acceso a múltiples fuentes de datos.
- Permite el acceso por JDBC o ODBC.



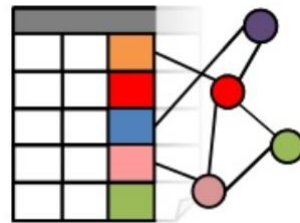
8 - 18

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Componentes Spark GraphX

- GraphX es el API para procesamientos paralelo en grafos.
- Spark GraphX implementa Resilient Distributed Graph (RDG- una abstracción de los RDD's).
- RDG's asocia registros con los vertices y bordes de un grafo. Sin embargo, se pueden seguir viendo como colecciones tradicionales de RDD.
- Se dispone de una gran cantidad de algoritmos preparados, que permiten agilizar el proceso de construcción de aplicaciones y mejora el rendimiento y velocidad.



8 - 19

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Componentes Spark MLlib

- Se dispone de una variedad de algoritmos y otros procesos como "data cleaning"
- Por ejemplo clasificación, clustering, regression, extracción etc...
- Permite su ejecución sobre HDFS, HBase, etc...



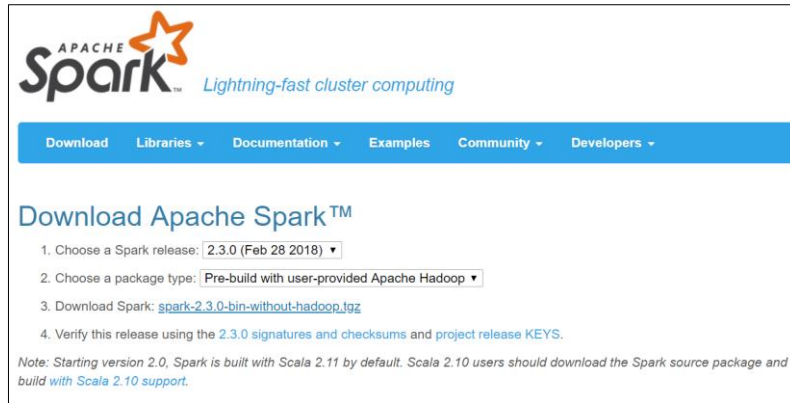
8 - 20

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Instalación y configuración Spark

<https://spark.apache.org/downloads.html>



8 - 21

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Instalación y configuración Spark

```
useradd spark
cd /opt
tar -xzf spark-2.3.0-bin-hadoop2.7.tgz
mv spark-2.3.0-bin-hadoop2.7 spark
chown -R spark:spark spark
su -l spark
cd /opt/spark
export JAVA_HOME=/usr/java/jdk1.8.0_161/jre
export SPARK_HOME=/opt/spark
export PATH=$SPARK_HOME/bin:$PATH
spark-shell
```

8 - 22

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.





## Python

```
$ pyspark
```

```
>>> textFile = spark.read.text("README.md")
```

```
>>> textFile.count() # Number of rows in this DataFrame
```

```
>>> textFile.first() # First row in this DataFrame
```

```
>>> linesWithSpark = textFile.filter(textFile.value.contains("Spark"))
```

```
>>> textFile.filter(textFile.value.contains("Spark")).count() # How  
many lines contain "Spark"?
```

8 - 25

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Ejercicio N° 8.1: Scala y Python

Al finalizar la tarea, el alumno logrará:

- Trabajar con diversos programas de Spark.

8 - 26

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Tarea N° 8: Spark

Al finalizar la tarea, el alumno logrará:

- Aprenderá los conceptos de la tecnología Spark.

8 - 27

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



## Resumen

En este capítulo, hemos aprendido como trabajar con Spark y como puede ser aplicado para el procesamiento de datos en memoria.

8 - 28

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.

