

# Ciencia de Datos: Un Enfoque Práctico en la Era del Big Data Entorno de procesamiento Hadoop

Sara Del Río García

Departamento de Ciencias de la Computación e Inteligencia Artificial,  
E.T.S. de Ingenierías Informática y de Telecomunicación,  
[srio@decsai.ugr.es](mailto:srio@decsai.ugr.es)



# Contenido

- 1 Introducción
- 2 Instalación
- 3 Referencias

# Contenido

## 1 Introducción

- Qué es Hadoop?
- Arquitectura Hadoop

## 2 Instalación

- Instalación CDH5 con YARN en un único nodo Linux
- HDFS (Hadoop Distributed File System)

## 3 Referencias

# Contenido

## 1 Introducción

- Qué es Hadoop?
- Arquitectura Hadoop

## 2 Instalación

- Instalación CDH5 con YARN en un único nodo Linux
- HDFS (Hadoop Distributed File System)

## 3 Referencias

# Qué es Hadoop?

- Es un proyecto de código abierto escrito en Java administrado por la fundación Apache
- Permite el almacenamiento y procesamiento distribuido de datos a gran escala en grandes clústeres de *comodity hardware*
- Se inspiró en:
  - *Google's MapReduce*
  - *Google's GFS (Google Distributed File system)*



# Características de Hadoop

- Consta de dos servicios principales:
  - **Almacenamiento:** HDFS
  - **Procesamiento:** MapReduce



- Aporta una serie de **ventajas**:
  - **Bajo coste:** clústeres baratos / cloud
  - **Facilidad de uso**
  - **Tolerancia a fallos**

# Contenido

## 1 Introducción

- Qué es Hadoop?
- **Arquitectura Hadoop**

## 2 Instalación

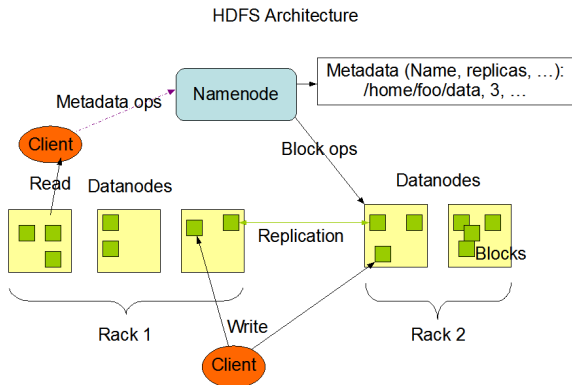
- Instalación CDH5 con YARN en un único nodo Linux
- HDFS (Hadoop Distributed File System)

## 3 Referencias

# Arquitectura HDFS

## Arquitectura Maestro Esclavo:

- **Maestro:** NameNode
- **Esclavo:** DataNode, ..., DataNode

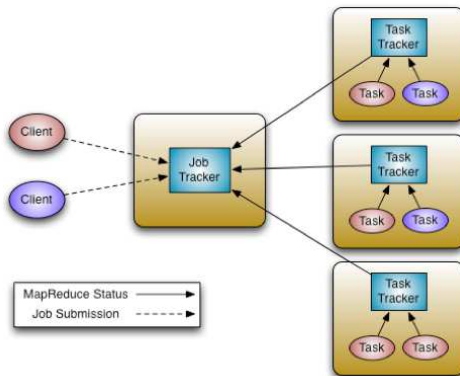




# Arquitectura MapReduce (MRv1)

Arquitectura Maestro Esclavo:

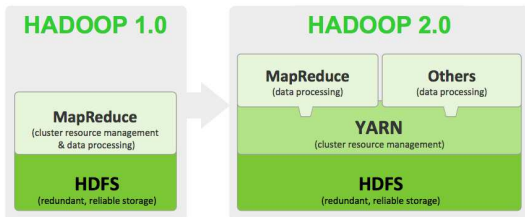
- **Maestro:** JobTracker
- **Esclavo:** TaskTraker, ..., TaskTraker



# YARN (Yet Another Resource Negotiator)

La arquitectura de Hadoop 1.0 (MRv1) se ha modificado en Hadoop 2.0 con YARN (MRv2):

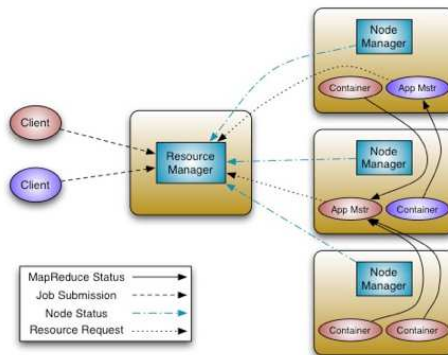
- Separa las dos funcionalidades del Jobtracker (gestión de recursos y job-scheduling/monitorización) en demonios separados
- Permite que diferentes tipos de aplicaciones (no sólo MapReduce) se ejecuten en el cluster



# Arquitectura MapReduce (YARN)

Arquitectura Maestro Esclavo:

- **Maestro:** Resource Manager
- **Esclavo:** Node Manager, ..., Node Manager



# Contenido

## 1 Introducción

- Qué es Hadoop?
- Arquitectura Hadoop

## 2 Instalación

- Instalación CDH5 con YARN en un único nodo Linux
- HDFS (Hadoop Distributed File System)

## 3 Referencias

# Instalación

- Fuentes: <http://hadoop.apache.org/releases.html>
- Sistemas pre-configurados proporcionados por empresas. Las tres distribuciones más extendidas son:
  - 1 **Cloudera** ([www.cloudera.com](http://www.cloudera.com)): contribuidor activo al proyecto que proporciona una distribución comercial y no-comercial de Hadoop (CDH)
  - 2 **MapR** ([www.mapr.com](http://www.mapr.com))
  - 3 **Hortonworks** ([www.hortonworks.com](http://www.hortonworks.com))
- Cada proveedor ofrece imágenes de VM con Linux y Hadoop ya instalado



# Instalación

## 3 modos de funcionamiento:

- En un nodo:
  - **Standalone:** de forma predeterminada, Hadoop está configurado para ejecutarse en un modo no distribuido, como un único proceso Java.
  - **Pseudo-Distribuido:** cada demonio de Hadoop se ejecuta en un proceso Java independiente
- En un clúster:
  - **Distribuido**

# Contenido

## 1 Introducción

- Qué es Hadoop?
- Arquitectura Hadoop

## 2 Instalación

- Instalación CDH5 con YARN en un único nodo Linux
- HDFS (Hadoop Distributed File System)

## 3 Referencias

# Instalación CDH5 con YARN en un único nodo Linux

- ❶ Asegurarse de tener Oracle JDK instalado:
  - `java -version`
  - `echo $JAVA_HOME`
- ❷ Instalar Oracle JDK (64 Bit) (si no está instalado):
  - Extraer e instalar el contenido del archivo binario RPM:
    - `su` (password root: hadoop)
    - `cd /home/hadoop`
    - `rpm -Uvh jdk-8u31-linux-x64.rpm`
    - `alternatives --install /usr/bin/java java /usr/java/jdk1.8.0_31/bin/java 1`
  - Configurar la variable de entorno `JAVA_HOME`:
    - `export JAVA_HOME=/usr/java/jdk1.8.0_31/`
    - `export PATH=$JAVA_HOME/bin:$PATH`
    - `env | grep JAVA_HOME`
    - `java -version`



# Instalación CDH5 con YARN en un único nodo Linux

- ③ Instalar el RPM para CDH5:
  - `sudo yum --nogpgcheck localinstall cloudera-cdh-5-0.x86_64.rpm`
- ④ Instalar Hadoop en modo Pseudo-distribuido:
  - `sudo yum install hadoop-conf-pseudo`
- ⑤ Iniciar Hadoop y verificar que funciona correctamente:
  - Para ver los archivos de configuración:
    - `rpm -ql hadoop-conf-pseudo`
    - Se obtendrá la siguiente salida:  
`/etc/hadoop/conf.pseudo /etc/hadoop/conf.pseudo/README`  
`/etc/hadoop/conf.pseudo/core-site.xml`  
`/etc/hadoop/conf.pseudo/hadoop-env.sh`  
`/etc/hadoop/conf.pseudo/hadoop-metrics.properties`  
`/etc/hadoop/conf.pseudo/hdfs-site.xml`  
`/etc/hadoop/conf.pseudo/log4j.properties`  
`/etc/hadoop/conf.pseudo/mapred-site.xml`  
`/etc/hadoop/conf.pseudo/yarn-site.xml`

# Instalación CDH5 con YARN en un único nodo Linux

- 5 Iniciar Hadoop y verificar que funciona correctamente:
  - Formatear el sistema de archivos (NameNode):
    - `sudo -u hdfs hdfs namenode -format`
  - Iniciar HDFS:
    - `for x in `cd /etc/init.d ; ls hadoop-hdfs-*` ; do sudo service $x start ; done`
  - Para verificar que los servicios se han iniciado, acceder a la consola web del NameNode (podremos consultar la capacidad de HDFS, número de DataNodes y logs):  
<http://localhost:50070/>

# Instalación CDH5 con YARN en un único nodo Linux

- 5 Iniciar Hadoop y verificar que funciona correctamente:
  - Consola web del NameNode: <http://localhost:50070/>

The screenshot shows the Hadoop web console interface. The top navigation bar includes links for Hadoop, Overview, Datanodes, Snapshot, Startup Progress, and Utilities. The main content area is divided into two sections: Overview and Summary.

**Overview 'localhost:8020' (active)**

|                       |   |
|-----------------------|---|
| <b>Started:</b>       | Wed Apr 08 14:32:12 CEST 2015                             |
| <b>Version:</b>       | 2.5.0-cdh5.3.2, r399edecc52da6b8eef1e88d8a563ede94c9cc87c |
| <b>Compiled:</b>      | 2015-02-24T20:54Z by jenkins from Unknown                 |
| <b>Cluster ID:</b>    | CID-8202c412-372f-4a8d-b89f-ae43572c3bed                  |
| <b>Block Pool ID:</b> | BP-1286068691-127.0.0.1-1428496270892                     |

**Summary**

Security is off.  
Safemode is off.

1 files and directories, 0 blocks = 1 total filesystem object(s).

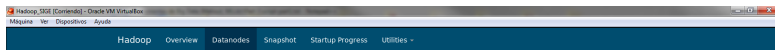
Heap Memory used 31.46 MB of 51.29 MB Heap Memory. Max Heap Memory is 966.69 MB.

Non Heap Memory used 34.01 MB of 35.13 MB Committed Non Heap Memory. Max Non Heap Memory is 1 B.

|                             |         |
|-----------------------------|---------|
| <b>Configured Capacity:</b> | 6.48 GB |
| <b>DFS Used:</b>            | 24 KB   |
| <b>Non DFS Used:</b>        | 3.28 GB |
| <b>DFS Remaining:</b>       | 3.2 GB  |

# Instalación CDH5 con YARN en un único nodo Linux

- 5 Iniciar Hadoop y verificar que funciona correctamente:
- En este caso veremos un único DataNode denominado localhost:



## Datanode Information

### In operation

| Node                                    | Last contact | Admin State | Capacity | Used  | Non DFS Used | Remaining | Blocks | Block pool used | Failed Volumes | Version        |
|---|--------------|-------------|----------|-------|--------------|-----------|--------|-----------------|----------------|----------------|
| localhost.localdomain (127.0.0.1:90010) | 1            | In Service  | 6.48 GB  | 24 KB | 3.20 GB      | 3.2 GB    | 0      | 24 KB (0%)      | 0              | 2.5.0-cdh5.3.2 |

### Decommissioning

| Node | Last contact | Under replicated blocks | Blocks with no live replicas | Under Replicated Blocks in files under construction |
|------|--------------|-------------------------|------------------------------|---|
|      |              |                         |                              |   |

Hadoop, 2014.

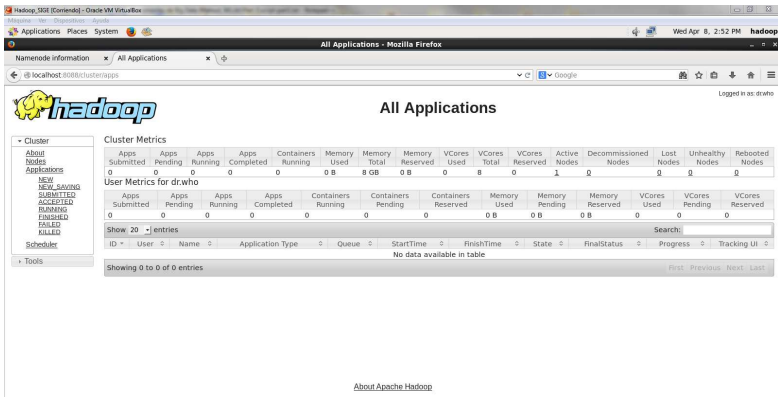
Legacy UI

# Instalación CDH5 con YARN en un único nodo Linux

- 5 Iniciar Hadoop y verificar que funciona correctamente:
  - Crear los directorios necesarios para los procesos de Hadoop con los permisos adecuados:
    - `sudo /usr/lib/hadoop/libexec/init-hdfs.sh`
  - Verificar la estructura de los ficheros para HDFS:
    - `sudo -u hdfs hadoop fs -ls -R /`
  - Iniciar YARN (ResourceManager, NodeManager):
    - `sudo service hadoop-yarn-resourcemanager start`
    - `sudo service hadoop-yarn-nodemanager start`
    - `sudo service hadoop-mapreduce-historyserver start`
  - Para verificar que los servicios se han iniciado, acceder a la consola web:  
<http://localhost:8088/cluster>

# Instalación CDH5 con YARN en un único nodo Linux

- 5 Iniciar Hadoop y verificar que funciona correctamente:
- Consola web YARN: <http://localhost:8088/cluster>



The screenshot shows the Hadoop YARN web console interface. The top navigation bar includes 'NameNode Information' and 'All Applications'. The main content area displays 'All Applications' for user 'dr.who'. On the left, a sidebar menu lists 'Cluster', 'About nodes', 'Applications', 'NEW', 'NEW SAVING', 'SUBMITTED', 'ACCEPTED', 'RUNNING', 'FINISHED', 'FAILED', 'KILLED', 'Scheduler', and 'Tools'. The 'Cluster Metrics' table shows various metrics for the cluster, including Apps Submitted, Pending, Running, Completed, Containers Running, Memory Used, Total, and Reserved, V-Cores Used, Total, and Reserved, Active Nodes, Decommissioned Nodes, Last Nodes, Unhealthy Nodes, and Rebooted Nodes. The 'User Metrics for dr.who' table shows similar metrics for the user. Below these tables, a search bar and a table of application entries are visible. The table header includes ID, User, Name, Application Type, Queue, StartTime, FinishTime, State, FinalStatus, Progress, and Tracking UI. The table body shows 'Showing 0 to 0 of 0 entries' and 'No data available in table'. The bottom of the page has a link to 'About Apache Hadoop'.

| Apps Submitted | Apps Pending | Apps Running | Apps Completed | Containers Running | Memory Used | Memory Total | Memory Reserved | V-Cores Used | V-Cores Total | V-Cores Reserved | Active Nodes | Decommissioned Nodes | Last Nodes | Unhealthy Nodes | Rebooted Nodes |
|----------------|--------------|--------------|----------------|--------------------|-------------|--------------|-----------------|--------------|---------------|------------------|--------------|----------------------|------------|-----------------|----------------|
| 0              | 0            | 0            | 0              | 0                  | 0 B         | 8 GB         | 0 B             | 0            | 8             | 0                | 1            | 0                    | 0          | 0               | 0              |

| Apps Submitted | Apps Pending | Apps Running | Apps Completed | Containers Running | Containers Pending | Containers Reserved | Memory Used | Memory Pending | Memory Reserved | V-Cores Used | V-Cores Pending | V-Cores Reserved |
|----------------|--------------|--------------|----------------|--------------------|--------------------|---------------------|-------------|----------------|-----------------|--------------|-----------------|------------------|
| 0              | 0            | 0            | 0              | 0                  | 0                  | 0                   | 0 B         | 0 B            | 0 B             | 0            | 0               | 0                |

Showing 0 to 0 of 0 entries

No data available in table

First Previous Next Last

# Instalación CDH5 con YARN en un único nodo Linux

- ⑤ Iniciar Hadoop y verificar que funciona correctamente:
  - Crear los directorios de los usuarios en HDFS:
    - `sudo -u hdfs hadoop fs -mkdir /user/hadoop`
    - `sudo -u hdfs hadoop fs -chown hadoop /user/hadoop`

# Contenido

## 1 Introducción

- Qué es Hadoop?
- Arquitectura Hadoop

## 2 Instalación

- Instalación CDH5 con YARN en un único nodo Linux
- HDFS (Hadoop Distributed File System)

## 3 Referencias



# HDFS (Hadoop Distributed File System)

- HDFS cuenta con tres interfaces:
  - **API de programación**
  - **Interfaz web**
    - Puerto 50070 del Namenode
  - **Línea de comandos:**
    - HDFS tiene su propia shell
    - Ayuda: [hadoop fs -help](#)

# HDFS (Hadoop Distributed File System)

Algunos comandos:

| Comando  | Descripción                    |
|--|--------------------------------|
| <code>hadoop fs -ls &lt;path&gt;</code>                            | Lista ficheros                 |
| <code>hadoop fs -cp &lt;src&gt; &lt;dst&gt;</code>                 | Copia ficheros de HDFS a HDFS  |
| <code>hadoop fs -mv &lt;src&gt; &lt;dst&gt;</code>                 | Mueve ficheros de HDFS a HDFS  |
| <code>hadoop fs -rm &lt;path&gt;</code>                            | Borra ficheros en HDFS         |
| <code>hadoop fs -rmr &lt;path&gt;</code>                           | Borra recursivamente en HDFS   |
| <code>hadoop fs -cat &lt;path&gt;</code>                           | Muestra fichero en HDFS        |
| <code>hadoop fs -mkdir &lt;path&gt;</code>                         | Crea directorio en HDFS        |
| <code>hadoop fs -put &lt;localsrc&gt; &lt;dst&gt;</code>           | Copia ficheros de local a HDFS |
| <code>hadoop fs -copyFromLocal &lt;localsrc&gt; &lt;dst&gt;</code> |                                |
| <code>hadoop fs -get &lt;src&gt; &lt;localdst&gt;</code>           | Copia ficheros de HDFS a local |
| <code>hadoop fs -copyToLocal &lt;src&gt; &lt;localdst&gt;</code>   |                                |

# Contenido

## 1 Introducción

- Qué es Hadoop?
- Arquitectura Hadoop

## 2 Instalación

- Instalación CDH5 con YARN en un único nodo Linux
- HDFS (Hadoop Distributed File System)

## 3 Referencias

# Referencias

- Apache Hadoop:

<http://hadoop.apache.org/>

- CDH 5 QuickStart Guide:

[http://www.cloudera.com/content/cloudera/en/documentation/core/latest/topics/cdh\\_qs.html](http://www.cloudera.com/content/cloudera/en/documentation/core/latest/topics/cdh_qs.html)

## Happy Hadooping!



# Ciencia de Datos: Un Enfoque Práctico en la Era del Big Data Entorno de procesamiento Hadoop

Sara Del Río García

Departamento de Ciencias de la Computación e Inteligencia Artificial,  
E.T.S. de Ingenierías Informática y de Telecomunicación,  
[srio@decsai.ugr.es](mailto:srio@decsai.ugr.es)

