


# Requisitos Hardware Big Data

Apache Hadoop, Apache HBase, Apache Spark

---

Document identifier:	<b>DO_SIS_Requisitos_Hardware_Big_Data_V9.odt</b>
Date:	<b>16/03/2015</b>
Document status:	<b>DRAFT</b>
Document link:	
License:	

---

Resumen: Este documento ofrece un análisis de los requisitos hardware generales para soluciones Big Data como por ejemplo Apache Hadoop, Apache Spark o Apache Hbase.

Copyright notice:

Copyright © CESGA, 2014.

See [www.cesga.es](http://www.cesga.es) for details on the copyright holder.

You are permitted to copy, modify and distribute copies of this document under the terms of the CC BY-SA 3.0 license described under <http://creativecommons.org/licenses/by-sa/3.0/>

Using this document in a way and/or for purposes not foreseen in the previous license, requires the prior written permission of the copyright holders.

The information contained in this document represents the views of the copyright holders as of the date such views are published.

THE INFORMATION CONTAINED IN THIS DOCUMENT IS PROVIDED BY THE COPYRIGHT HOLDERS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE MEMBERS OF THE EGEE-III COLLABORATION, INCLUDING THE COPYRIGHT HOLDERS, OR THE EUROPEAN COMMISSION BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THE INFORMATION CONTAINED IN THIS DOCUMENT, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Trademarks:

Hadoop, Cassandra, Lucene, Spark, and Hbase are registered trademarks held by The Apache Software Foundation.

Systap Bigdata is a registered trademark held by Systap.

The icons used in this document were obtained from:

<http://www.iconarchive.com>

<http://www.iconarchive.com/show/icloud-icons-by-ahdesign91.html>

<http://www.archlinux.org/packages/extra/any/oxygen-icons/download>

<http://www.iconarchive.com/show/vista-hardware-devices-icons-by-icons-land.html>

<http://hortonworks.com/blog/a-set-of-hadoop-related-icons/>

## Control de Cambios

<b>Versión</b>	<b>Fecha</b>	<b>Descripción</b>	<b>Autor</b>
1	07/02/2014	Estructura del documento e introducción	Javier Cacheiro
2	11/02/2014	Versión inicial	Javier Cacheiro
3	11/02/2014	Nodos maestro	Javier Cacheiro
4	11/02/2014	Lista para comentarios	Javier Cacheiro
5	12/02/2014	Revisión	Juan Villasuso
6	13/02/2014	Revisión menor	Javier Cacheiro
7	12/05/2014	Revisión menor	Javier Cacheiro
8	10/03/2015	Incluir alternativa discos SSD	Javier Cacheiro
9	16/03/2015	Completar información solución SSD	Javier Cacheiro

## Content

<b>1</b>	<b>Introducción.....</b>	<b>5</b>
1.1	Propósito del documento.....	5
1.2	Área de Aplicación.....	5
1.3	Referencias.....	5
1.4	Modificaciones al Documento.....	6
1.5	Terminología.....	6
1.6	Convenciones.....	6
<b>2</b>	<b>Estructura del Documento.....</b>	<b>8</b>
<b>3</b>	<b>Introducción.....</b>	<b>9</b>
<b>4</b>	<b>Nodos Esclavos.....</b>	<b>11</b>
4.1	Procesador.....	11
4.2	Memoria.....	12
4.3	Disco.....	13
4.4	Red.....	14
4.5	Otras consideraciones.....	14
<b>5</b>	<b>Nodos maestros.....</b>	<b>16</b>
5.1	Procesador.....	17
5.2	Memoria.....	17
5.3	Disco.....	17
5.4	Red.....	18
5.5	Otras consideraciones.....	18
<b>6</b>	<b>Arquitecturas de Referencia.....</b>	<b>19</b>
<b>7</b>	<b>Benchmarks.....</b>	<b>21</b>

# 1 Introducción

## 1.1 Propósito del documento

Este documento analiza los requisitos hardware necesarios para desplegar distintas soluciones Big Data como por ejemplo Apache Hadoop, Apache Hbase o Apache Spark. Todas estas soluciones comparten un punto en común y es que están basadas en el sistema de ficheros paralelo HDFS,

## 1.2 Área de Aplicación

Este documento está dirigido a administradores de sistemas y personas encargadas de la adquisición de hardware para plataformas Big Data.

## 1.3 Referencias

**Tabla 1: Tabla de referencias**

<b>R1</b>	Apache Hadoop, <a href="http://hadoop.apache.org/">http://hadoop.apache.org/</a>
<b>R2</b>	SciDB: <a href="http://scidb.org/">http://scidb.org/</a>
<b>R3</b>	Cassandra: <a href="http://cassandra.apache.org/">http://cassandra.apache.org/</a>
<b>R4</b>	Systap Bigdata: <a href="http://www.systap.com/bigdata.htm">http://www.systap.com/bigdata.htm</a>
<b>R5</b>	Sesame: <a href="http://www.openrdf.org/">http://www.openrdf.org/</a>
<b>R6</b>	Lucene: <a href="http://lucene.apache.org/">http://lucene.apache.org/</a>
<b>R7</b>	Hbase: <a href="http://hbase.apache.org/">http://hbase.apache.org/</a>
<b>R8</b>	Spark: <a href="http://spark.apache.org/">http://spark.apache.org/</a>
<b>R8</b>	HP AppSystem for Apache Hadoop: <a href="http://www8.hp.com/us/en/products/solutions/product-detail.html?oid=5318722#!tab=specs">http://www8.hp.com/us/en/products/solutions/product-detail.html?oid=5318722#!tab=specs</a>
<b>R9</b>	IBM PureData System for Hadoop: <a href="http://www.ibm.com/ibm/puresystems/es/es/pd_hadoop.html">http://www.ibm.com/ibm/puresystems/es/es/pd_hadoop.html</a>
<b>R10</b>	FUJITSU Software Interstage Big Data Parallel Processing Server: <a href="http://www.fujitsu.com/global/services/software/interstage/solutions/big-data/bdpp/">http://www.fujitsu.com/global/services/software/interstage/solutions/big-data/bdpp/</a>
<b>R11</b>	The Truth About MapReduce Performance on SSDs <a href="http://blog.cloudera.com/blog/2014/03/the-truth-about-mapreduce-performance-on-ssds/">http://blog.cloudera.com/blog/2014/03/the-truth-about-mapreduce-performance-on-ssds/</a>
<b>R12</b>	Heterogeneous Storages in HDFS <a href="http://hortonworks.com/blog/heterogeneous-storages-hdfs/">http://hortonworks.com/blog/heterogeneous-storages-hdfs/</a>

## 1.4 Modificaciones al Documento

Este documento está bajo la responsabilidad del CESGA. Cualquier modificación, comentario o sugerencia puede ser enviado a Javier Cacheiro (jlopez [at] cesga.es).



## 1.5 Terminología

**Table 2: Glosario**

<b>Hadoop</b>	Apache Hadoop es una plataforma Big Data para ejecución de cálculos MapReduce.
<b>SciDB</b>	Solución Big Data orientada al análisis de arrays multidimensionales de datos los cuales son empleados con gran frecuencia en distintas áreas de la investigación científica.
<b>Cassandra</b>	Apache Cassandra es una base de datos NoSQL distribuida
<b>Hbase</b>	Apache Hbase es una base de datos NoSQL que se ejecuta por encima de HDFS (el sistema de ficheros de Hadoop).
<b>Systap Bigdata</b>	Almacén RDF distribuido altamente escalable. Se puede utilizar como backend de Sesame.
<b>Sesame</b>	OpenRDF Sesame es el standard de-facto para procesar datos RDF.
<b>Lucene</b>	Apache Lucene es una API de código abierto para recuperación de información

## 1.6 Convenciones

Este documento usa varias convenciones para destacar ciertas palabras y frases y así llamar la atención del lector sobre partes determinadas del mismo.

	Este icono indica sugerencias que podrían ser útiles al lector.
	Este icono indica consideraciones importantes que podrían pasarse fácilmente por alto.



Este icono indica aspectos críticos que no deberían pasarse por alto.

## 2 Estructura del Documento

El documento está organizado como sigue:

- En la Sección 3 se muestran las similitudes entre las distintas soluciones Big Data y se establece una arquitectura genérica basada en nodos maestro y nodos esclavos.
- En la Sección 4 se hace un análisis de las características de los nodos esclavos.
- En la Sección 5 se hace un análisis de las características de los nodos maestros.
- En la Sección 6 se analizan las arquitecturas de referencia y las soluciones comerciales propuestas por distintos fabricantes como HP, IBM, Fujitsu u Oracle.
- En la Sección 7 se hace una corta revisión con respecto a los posibles benchmarks a utilizar para medir el rendimiento de una plataforma Big Data.



### 3 Introducción

Las soluciones Big Data existentes hoy en día comparten muchas características tanto desde el punto de vista de la arquitectura como desde el punto de vista del entorno de ejecución que hacen que, en general, podamos tratar de una forma conjunta los requisitos hardware de todas ellas.

En cuanto a arquitectura, podemos decir que todas las plataformas Big Data cuentan con dos tipos de nodos:

- **Nodos maestro:** son los nodos de gestión de los servicios Big Data como pueden ser la distribución de los trabajos y el servicio de metadatos del sistema de ficheros paralelo.
- **Nodos esclavo:** son el equivalente a los nodos de computación de un cluster de computación tradicional.

En cuanto a entorno de ejecución, la mayor parte de las soluciones Big Data (Hadoop, Spark, Hbase, SciDB, Cassandra, Sesame, Systap Bigdata, Lucene, etc) están programadas en Java y emplean una **máquina virtual Java** (JVM) para ejecutar de forma independiente cada uno de los servicios así como las tareas asociadas a un trabajo determinado.

Esto hace que en general todas las soluciones Big Data compartan requisitos similares en cuanto al hardware físico a emplear.



Las soluciones Big Data comparten requisitos similares en cuanto al hardware físico a emplear.

Por tanto a la hora de realizar el análisis nos centraremos en la solución más extendida en la actualidad: Apache Hadoop.

En la Figura 1 pueden verse los principales servicios de Hadoop y como se distribuyen entre los nodos maestros y esclavos. Como se puede observar en el caso de Hadoop lo habitual es contar con tres nodos maestro para configuraciones de sistemas en producción.

Distintas soluciones Big Data emplean distinto número de nodos maestros e incluso en una solución determinada se pueden juntar servicios en un sólo nodo maestro perjudicando en general el rendimiento y la fiabilidad ante un fallo eventual del servicio. Sin embargo, **los requisitos generales de los nodos maestro en cuanto al hardware a emplear son muy similares independientemente de la plataforma.**

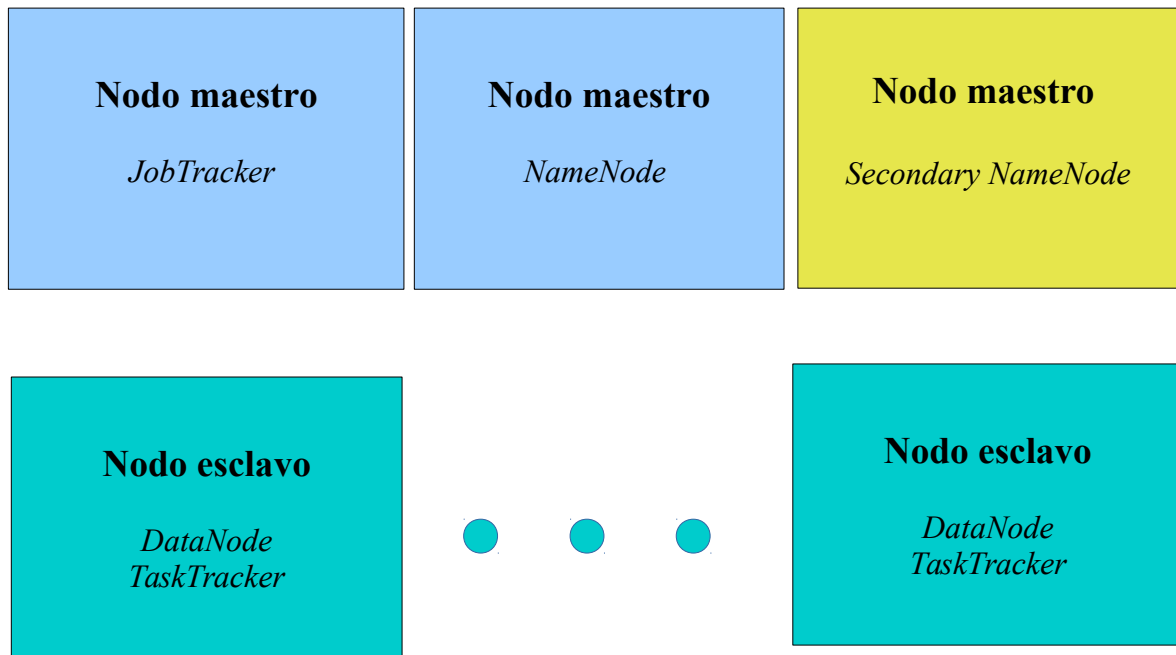


Figura 1: Arquitectura general Hadoop

## 4 Nodos Esclavos

Los nodos esclavos componen la parte más importante del cluster y en el caso de Hadoop son los encargados de ejecutar dos demonios: el DataNode y el TaskTracker (ver figura 1).

El primero de los demonios (DataNode) corresponde al sistema de ficheros distribuido HDFS y el segundo (TaskTracker) al entorno de ejecución de trabajos MapReduce.



En general todas las soluciones Big Data cuentan con un **sistema de ficheros paralelo** y un **sistema de gestión para la ejecución de trabajos** sobre esos ficheros. Por tanto, aunque con nombres diferentes, veremos dos demonios similares en otras soluciones Big Data.

La prioridad en los nodos esclavos es conseguir el mayor ancho de banda posible de entrada salida a dos niveles:

- Entrada salida a disco local para el procesamiento de los datos locales.
- Entrada salida a través de la red para transferir los resultados intermedios.

En ambos casos estamos hablando de ancho de banda y no de latencia.



Las características fundamentales que persigue un nodo esclavo son:

- Ancho de banda a disco local
- Ancho de banda a red

Una configuración típica de un nodo esclavo sería:

- Discos duros: 12x2TB SATA (sin RAID, configuración JBOD)
- Red: 4x1Gb o 1x10Gb
- Procesador: 2x6 cores
- Memoria: 64GB o 96GB

### 4.1 Procesador

Los cálculos Big Data raramente están limitados por la CPU, en general están limitados por la E/S a disco y a la red.

Por tanto no son necesarios procesadores de gama alta y se prima el número de cores a la frecuencia ya que la mayor parte del tiempo la CPU estará esperando por la E/S. Por este motivo es recomendable habilitar el HyperThreading (HT) y el Quick-Path Interconnect (QPI).

Las tareas de los trabajos MapReduce no están compiladas para aprovechar optimizaciones específicas del procesador ya que, en el mejor de los casos, están

programadas en Java y distribuidas como *bytecode*. O en otros casos, si usan *streaming* con algún otro lenguaje de alto nivel (generalmente Perl, Python o Ruby) aun ofrecen un rendimiento menor.

## 4.2 Memoria

La memoria de los nodos esclavos debe ser proporcional al número máximo de tareas que permitimos ejecutar simultáneamente en un nodo.

En general se deberían reservar entre 2GB y 4GB para cada tarea Map o Reduce.



Una buena aproximación es utilizar un número de tareas por nodo igual a **1.5 x <número de cores>**. De este modo si tenemos nodos con 16 cores permitiremos 24 tareas por nodo.

De este modo la memoria necesaria en el caso un nodo con 16 cores, en el que permitamos 24 tareas, serían entre 48GB y 96GB.

Para evitar que se produzca *swapping* por agotarse la memoria física de la máquina, tendríamos que sumar también la memoria necesaria para los dos demonios que se ejecutan en el nodo (DataNode y TaskTracker) y para el S.O. De este modo, en general, podemos añadir 4GB adicionales a los requisitos de memoria.



En general se recomienda una configuración en donde los nodos esclavos tengan al menos la siguiente memoria:

$$\text{Memoria mínima (GB)} = 1.5 \times \text{<num cores>} \times 2 + 4$$

A modo orientativo podríamos usar los valores de la Tabla 1:

Número de cores	Memoria mínima	Memoria máxima
4	16GB	28GB
8	28GB	52GB
12	40GB	76GB
16	52GB	100GB
20	64GB	124GB
24	76GB	148GB

**Tabla 1: Nodos esclavo:** Memoria RAM recomendada en función del número de cores disponibles

Es importante garantizar que todas las tareas de un trabajo dispongan de la misma cantidad de memoria ya que los trabajos MapReduce emplean el mismo código en todos los nodos. Lo mismo ocurre en las otras soluciones Big Data analizadas donde las tareas a distribuir entre todos los nodos son equivalentes.



Esto no implica que todos los nodos deban tener la misma memoria sino que lo que debe respetarse la relación memoria por tarea permitida en el nodo.

### 4.3 Disco

En cuanto a los requisitos de disco debe tenerse en cuenta que HDFS y en general los sistemas de ficheros paralelos empleados en soluciones Big Data guardan los datos replicados. El número de réplicas es configurable y en el caso de HDFS el valor por defecto es de tres.

En general, como en otros sistemas de almacenamiento paralelos, cuantos más ejes (discos) mejor. Cuantos más discos tengamos en los nodos esclavo más probable será que estemos accediendo datos en discos diferentes.

Debido a su menor precio, mejor rendimiento y mayor capacidad se recomiendan los **discos de 3,5"** frente a los de 2.5". Sin embargo no se recomiendan los discos de 10k o 15k RPM por su incremento de precio que no compensa el incremento de rendimiento obtenido.



Es preferible una configuración de los nodos esclavos con 8 discos de 1TB que con 4 discos de 2TB.

Por otro lado tampoco es recomendable configurar una cantidad muy elevada de almacenamiento por nodo esclavo, ya que en caso de fallo del nodo el resto de nodos se pondrán a replicar los datos que estaban en ese nodo para que así se mantenga el número de replicas por bloque al que tengamos establecido. **Esto provocará un gran tráfico de red mientras se hace esta re-replicación.**



Se recomienda no superar los 36TB por nodo esclavo para evitar la saturación de la red en caso de fallo de un nodo.

No se recomienda utilizar sistemas RAID ni LVM para agrupar los discos ni por tolerancia a fallos ni por rendimiento. HDFS ya se encarga de replicar los datos para garantizar la tolerancia a fallos por lo que no es necesario un RAID1, RAID5, RAID6, etc. Por otro lado si usamos una configuración en stripping (RAID 0 o LVM) estaremos limitando el rendimiento al del disco más lento del array, sin embargo en una configuración JBOD con discos independientes HDFS accederá a todos los discos de forma independiente y la velocidad media siempre será mayor que la del disco más lento.



Utilizar configuraciones con discos independientes (JBOD) y evitar utilizar configuraciones RAID o LVM con stripping.

#### 4.4 Red

Hadoop al igual que otras soluciones Big Data es muy intensivo en cuanto al uso de la red. Así por ejemplo, en el caso de Hadoop, todos los nodos esclavos intercambian datos entre ellos antes de pasar de la fase Map a la fase Reduce.

Los nodos deben disponer al menos de conectividad Gbit, siendo recomendable que dispongan de varias conexiones de 1Gbit o de una conexión 10Gbit. **Teniendo en cuenta el elevado coste de las conexiones 10Gbit puede ser más adecuado utilizar 4 conexiones 1Gbit agregadas.**



Se recomienda utilizar switches dedicados para la plataforma.

Los nodos se conectan a switches Top-of-the-Rack (ToR) y estos a su vez a switches core. En el caso de **las conexiones de los switches ToR a los switches core se recomienda utilizar conexiones 10Gbit/s.**

#### 4.5 Otras consideraciones

Si usamos **virtualización** tendremos penalizaciones en cuanto al rendimiento en dos áreas críticas: entrada salida a disco y entrada salida a red. En general, tampoco podremos garantizar que las máquinas virtuales están distribuidas de forma óptima con respecto a los recursos físicos de disco y red, de modo que es posible que todas las réplicas de un bloque acaben estando en un mismo disco físico.

Alternativas actuales a la virtualización completa usando contenedores como **Docker** pueden ser una buena opción ya que eliminan la penalización de rendimiento de las

soluciones de virtualización completa. Actualmente se está trabajando en el soporte de Hadoop para Docker por lo que esta puede ser una buena alternativa en el futuro.



Se recomienda utilizar nodos físicos y no nodos virtuales.

**Tampoco son recomendables los servidores blades o SL** porque limitan enormemente la cantidad de discos que se pueden añadir a los nodos esclavo: generalmente el máximo es 2 y utilizan discos de 2.5". En el caso de blades también limitan la interconexión de red al salir todos por la misma boca (excepto en configuraciones passthrough).

## 5 Nodos maestros

Los nodos maestro son los encargados de gestionar el cluster y en el caso de Hadoop se componen de los siguientes demonios: el NameNode, Secondary NameNode y el JobTracker (ver figura 1).

Los dos primeros demonios (NameNode y Secondary NameNode) corresponde al sistema de ficheros distribuido HDFS y el tercero (JobTracker) al entorno de ejecución de trabajos MapReduce.



A partir de la versión de Hadoop 2.0.0 se han introducido una nueva característica que permite configurar el **HDFS en alta disponibilidad (HA)**, de modo que el NameNode ya no sea un punto único de fallo (SPOF) del sistema.

Para realizar la configuración del HDFS en alta disponibilidad se configuran dos NameNodes redundantes en una configuración activo/pasivo. En la práctica esto implica añadir un nodo maestro adicional que actúe como respaldo del NameNode principal.



Si usamos Hadoop > 2.0.0 es recomendable disponer de 4 nodos físicos como nodos maestros y si usamos versiones de Hadoop anteriores 3 nodos físicos.

Hoy en día, las versiones más comunes para *cluster* de producción son las anteriores a la 2.0.0 sin embargo distribuciones como CDH integran la nueva versión de HDFS y nos permiten seguir usando MRv1.

Una configuración típica de un nodo maestro sería:

- Red: 2x1Gb (con bonding)
- Procesador: 2x4 cores
- Memoria: 24-48GB (ECC)
- Fuente de alimentación redundante
- Discos duros: 6x1TB SAS o 6x512GB SSD
- Discos en RAID



En general se recomienda utilizar el hardware más fiable posible para los nodos maestros.



## 5.1 Procesador

En el caso de los nodos maestros no es necesario que dispongan de gran cantidad de cores y se prima la fiabilidad al rendimiento.

En general con procesadores quad-core es más que suficiente, incluso se podría utilizar un sólo procesador en lugar de dos.

## 5.2 Memoria

En cuanto a la memoria necesaria, dependerá del número de nodos esclavos que tenga el cluster. Se puede usar como referencia la siguiente tabla:

Número de nodos esclavo	Memoria recomendada
20	24GB
300	48GB
>300	96GB

Tabla 2: Nodos maestros: Memoria RAM recomendada en función del número de nodos esclavo del cluster

## 5.3 Disco

El principal condicionante es la fiabilidad ya que en los nodos maestros se almacena información crítica como los metadatos del sistema de ficheros paralelo. También es importante la velocidad de acceso aleatorio ya que el patrón habitual de estos nodos serán en su lecturas aleatorias de pequeño tamaño combinadas con un pequeño porcentaje de escrituras también aleatorias y de pequeño tamaño.

Para obtener una mayor fiabilidad se recomienda utilizar **discos SAS** en vez de discos SATA ya que en general presentan mejores tasas de fiabilidad. También se recomienda una configuración de los discos en RAID redundante (RAID6, RAID10).

Dado que el rendimiento de E/S aleatorio es la parte más crítica dentro del servidor de metadatos y del jobtracker se recomienda utilizar discos SAS de **10k o 15k RPM**.

Como alternativa, también se podrían utilizar **discos SSD**, ya que se han abaratado notablemente durante los últimos años y hoy en día podemos encontrar discos SSD dentro del rango de precios de los discos SAS (aunque a menor capacidad). Esta opción no está tan probada como la alternativa con discos SAS, pero los rendimientos que se pueden obtener con un mismo número de discos son muy superiores.

La mayor ventaja de los discos SSD es que tienen un rendimiento en acceso aleatorio muy superior a los discos SAS (80.000 IOPs frente a 100-200 IOPs). Dado su elevado

rendimiento en algunos casos se conectan directamente por PCI Express en lugar de a través de los puertos SATA.

Si se escogen discos SSD es importante verificar que la controladora RAID sea capaz de soportar los altos valores de IOPS de estos discos para que no actúe de cuello de botella. Las controladoras RAID hardware tradicionales tienen un límite de rendimiento alrededor de 80.000 IOPS. **Si la controladora no es específica para discos SSD se obtendrá mejor rendimiento si se conectan los discos directamente a los puertos SATA de la placa base y se hace el RAID por software.**

Existen también nuevos modelos de controladoras que disponen de soporte específico para discos SSD a través de aceleradores software (Intel RAID Fastpath IO, LSI MegaRAID FastPath, HP SSD Smart Path) incrementando el rendimiento máximo que son capaces de suministrar hasta por encima de las 400.000 IOPS. Por tanto, **si se opta por una solución con controladora RAID hardware es importante verificar que la controladoras RAID cuente para aceleración por software y sea capaz de soportar más de 400.000 IOPS.**

Otra consideración importante si se usan discos SSD es el tipo de RAID a utilizar. Si se utiliza RAID5 el rendimiento en escritura aleatoria se degrada enormemente bajando en algunos benchmarks de 140.000 IOPS a sólo 20.000 IOPS en un RAID con 8 discos SSD Intel X25E. Además se reduce la vida útil de los discos debido a las re-escrituras de la paridad cada vez que se cambia un *stripe*. Lo recomendable en estos casos para mantener un buen rendimiento de escritura aleatoria es utilizar RAID10.

En discos SSD también es importante tener en cuenta la fiabilidad del disco con parámetros como: MTBF, **número máximo de ciclos borrado/escritura (P/E cycle)** y la garantía. En este sentido los discos SSD con tecnología SLC se consideran más fiables que los que emplean tecnología MLC ya que soportan hasta 100.000 ciclos de borrado/escritura por celda frente a los 10.000 que soporta cada celda de los discos MLC. A pesar de esto debemos tener en cuenta que los discos más comunes son MLC incluso en gama profesional.

También debemos tener en cuenta que los para mejorar la vida útil de los discos SSD estos implementan *wear leveling* y otros algoritmos para distribuir las escrituras, pero debemos tener en cuenta que llega un momento en que estos discos pueden fallar repentinamente.

Para tener unos valores de referencia el MTBF de discos de gama profesional es de 1.2 millones de horas y **2000 ciclos de borrado/escritura** (este valor equivale a más de 1 escritura completa diaria durante 5 años) y una garantía de 5 años.

Por último hay dos factores adicionales en cuanto a los discos SSD que debemos tener en cuenta:

- El **período de retención de la información**, es decir el tiempo que la información permanece legible en el disco si este está apagado. Este tiempo depende de factores como el número de ciclos P/E utilizados o la temperatura de almacenamiento, situándose en torno a los 3 meses.
- La **protección frente a pérdidas de alimentación**: los discos SSD son más sensibles ante un corte eléctrico y se pueden provocar una corrupción del

sistema de ficheros. Para evitar este problema algunos discos SSD disponen de mecanismos hardware y software que permitan detectar este tipo de situaciones y realizar el correcto vaciado de los buffers y metadata a la memoria flash NAND, contando un circuito específico basado en condensadores que permitan disponer de suficiente energía para la operación.

## 5.4 Red

En el caso de los nodos maestros dos conexiones 1Gbit son suficientes. Lo importante en este caso es configurar las distintas NIC en bonding para mejorar la tolerancia a fallos.



Utilizar configuración de red redundante con bonding.

## 5.5 Otras consideraciones

La principal prioridad en los nodos maestros es la fiabilidad de los mismos. Por ello se recomienda utilizar hardware fiable.

## 6 Arquitecturas de Referencia

Dentro de sus portfolio de productos muchos fabricantes incluyen ya soluciones específicas Big Data (en general basadas en Hadoop):

Fabricante	Nombre de la solución
HP	HP AppSystem for Hadoop
IBM	IBM PureData System for Hadoop
Fujitsu	FUJITSU Software Interstage Big Data Parallel Processing Server
Oracle	Oracle Big Data Appliance
EMC	EMC Greenplum Data Computing Appliance (DCA)
NetApp	NetApp Open Solution for Hadoop
Supermicro	Supermicro Hadoop

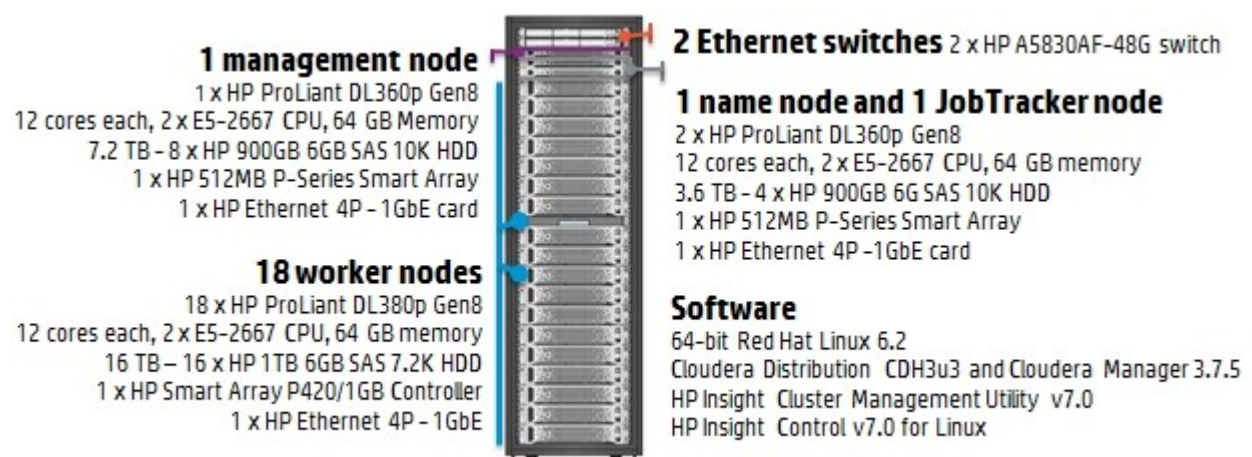
*Tabla 3:* Plataformas Big Data ofertadas por distintos fabricantes.

Cada fabricante propone una arquitectura de referencia para soluciones Big Data usando servidores de su catálogo. En general se recomiendan los siguientes modelos de servidor para nodos maestro y esclavo:

Nodos maestro	Nodos esclavo
HP ProLiant DL360 Gen8	HP ProLiant DL380e Gen8
HP ProLiant SL4540	HP ProLiant SL4540
IBM System x3550 M4	IBM – System x3630 M4
Dell PowerEdge R620	Dell – PowerEdge R520
Supermicro SYS-HNW0-15026364-HADP	Supermicro SYS-6028U-TR4 Supermicro SYS-HDD0-25126332-HADP Supermicro SYS-HDD0-25226332-HADP

Tabla 4: Modelos de servidor recomendados por fabricante

A modo de referencia a continuación se incluye el diagrama correspondiente a la solución HP AppSystem for Hadoop:



La información detallada sobre el último modelo se puede encontrar en:  
<http://h18000.www1.hp.com/products/quickspecs/productbulletin.html#spectype=worldwide&type=html&docid=14465>

Esto es lo que menciona HP sobre su arquitectura de referencia en su página web:

***HP Reference Architectures for Hadoop***

*HP offers you a choice of reference architectures for Apache Hadoop with each of the three leading distributions: Cloudera, Hortonworks and MapR. These reference architectures are available on DL380 and SL4500 platforms, offering you choice as you build the Hadoop ecosystem to best fit your needs.*

## 7 Benchmarks

En cuanto a los benchmarks a utilizar para evaluar las soluciones Big Data se está llevando a cabo una iniciativa para establecer un benchmark con el que crear una lista de los 100 clusters Big Data más potentes del mundo:

<http://www.bigdatatop100.org/>

En el caso de las soluciones basadas en Hadoop el benchmark más extendido es la utilidad **TeraSort** que se distribuye con el paquete de Apache Hadoop `org.apache.hadoop.examples.terasort`.

Adicionalmente se puede utilizar:

- **TeraGen**: para medir el rendimiento de carga de datos con replicación triple y uso intensivo de la red
- **TeraRead**: trabajo de sólo lectura (sin shuffle&sort ni tarea reduce)
- **WordCount**: trabajo típico map reduce con elevado uso de CPU que se encarga de contar las ocurrencias de cada palabra