

UD 1. Xestión de sistemas de almacenamento e ecosistemas big data.

Contidos

- Computación distribuída. Computación paralela.
- Sistemas de almacenamento distribuídos. Tolerancia a fallos.
 - Sistemas de almacenamento distribuídos
 - Tolerancia a fallos
- Ferramentas:
 - Map Reduce.
 - Pig, Hive, Flume.
 - Sqoop, Oozie.
 - Automatización de Jobs.
 - Consultas Pig e Hive.
- Outras ferramentas.

Criterios de avaliación

- Determinouse a importancia dos sistemas de almacenamento para depositar e procesar grandes cantidades de calquera tipo de datos rapidamente.
- Comprobouse o poder de procesamento do seu modelo de computación distribuída.
- Probouse a tolerancia a fallos dos sistemas.
- Determinouse que se poden almacenar tantos datos como se desexe e decidir como utilízalos máis tarde.
- Visualizouse que o sistema pode crecer doadamente engadindo módulos.

Índice

Clusterización e sistemas de almacenamento.....	5
Clusterización.....	5
Sistemas monoprocesador.....	5
Clusterización.....	6
Procesamento paralelo ou paralelizado / distribuído.....	6
Clusterización.....	7
Procesamento paralelo ou paralelizado / distribuído.....	7
Ley de Amdahl.....	7
Clusterización.....	8
Procesamento paralelo ou paralelizado / distribuído.....	8
Paradigmas de paralelismo clásicos.....	8
Clusterización.....	9
Procesamento paralelo ou paralelizado / distribuído.....	9
Novo paradigma: “commodity cluster”.....	9
Clusterización.....	10
Procesamento paralelo ou paralelizado / distribuído.....	10
Novo paradigma: puntos clave.....	10
Clusterización.....	11
Procesamento paralelo ou paralelizado / distribuído.....	11
Clusterización.....	12
Procesamento paralelo ou paralelizado / distribuído.....	12
Clusterización.....	15
Procesamento paralelo ou paralelizado / distribuído.....	15
Clusterización.....	17
Procesamento paralelo ou paralelizado / distribuído.....	17
Clusterización.....	18
Procesamento paralelo ou paralelizado / distribuído.....	18
Arquitectura clúster.....	18
Clusterización.....	19
Procesamento paralelo ou paralelizado / distribuído.....	19
Servizos ofrecidos por un clúster.....	19
Sistemas de xestión do almacenamento.....	21
Sistemas de xestión de almacenamento en big data.....	21
Sistemas de xestión do almacenamento.....	22
Sistemas de xestión de almacenamento en big data.....	22
Sistemas de xestión do almacenamento.....	23
Sistemas de xestión de almacenamento en big data.....	23
Sistemas de arquivos distribuídos.....	23
Sistemas de xestión do almacenamento.....	24
Sistemas de xestión de almacenamento en big data.....	24
Sistemas de arquivos distribuídos.....	24
Sistemas de xestión do almacenamento.....	25
Sistemas de xestión de almacenamento en big data.....	25
Sistemas de arquivos distribuídos: HDFS (Hadoop File System).....	25
Sistemas de xestión do almacenamento.....	27
Sistemas de xestión de almacenamento en big data.....	27
Sistemas de arquivos distribuídos: HDFS (Hadoop File System).....	27
Sistemas de xestión do almacenamento.....	29

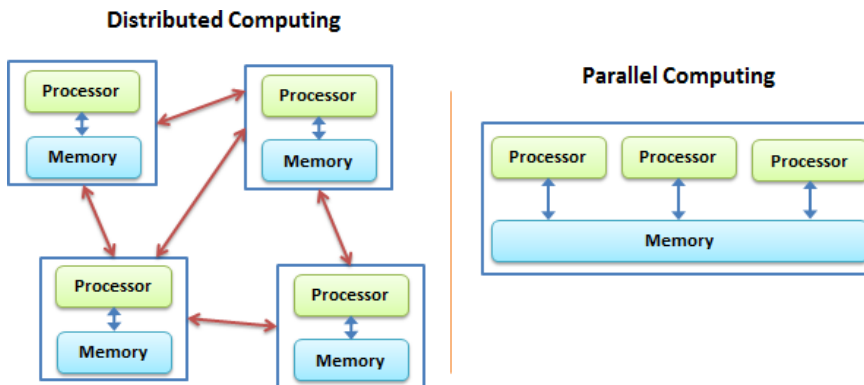
Sistemas de xestión de almacenamento en big data.....	29
Sistemas de arquivos distribuídos: HDFS (Hadoop File System).....	29
Caso práctico.....	29
Sistemas de xestión do almacenamento.....	30
Sistemas de xestión de almacenamento en big data.....	30
Sistemas de arquivos distribuídos: HDFS (Hadoop File System).....	30
Esquema de Sistema de ficheros distribuído Hadoop ou HDFS (Hadoop Distributed File System).....	30
Sistemas de xestión do almacenamento.....	31
Sistemas de xestión de almacenamento en big data.....	31
Sistemas de arquivos distribuídos: HDFS (Hadoop File System).....	31
Esquema do HDFS (Hadoop Distributed File System).....	31
Sistemas de xestión do almacenamento.....	33
Sistemas de xestión de almacenamento en big data.....	33
Sistemas de arquivos distribuídos: HDFS (Hadoop File System).....	33
Esquema de Sistema de ficheros distribuído Hadoop ou HDFS (Hadoop Distributed File System).....	33
Sistemas de xestión do almacenamento.....	35
Sistemas de xestión de almacenamento en big data.....	35
Sistemas de arquivos distribuídos: HDFS (Hadoop File System).....	35
Esquema de Sistema de ficheros distribuído Hadoop ou HDFS (Hadoop Distributed File System).....	35
Sistemas de xestión do almacenamento.....	37
Sistemas de xestión de almacenamento en big data.....	37
Sistemas de arquivos distribuídos: HDFS (Hadoop File System).....	37
Arquitectura Hadoop Cluster.....	37
Sistemas de xestión do almacenamento.....	38
Sistemas de xestión de almacenamento en big data.....	38
Principais sistemas de ficheiros utilizados en Big Data.....	38
Sistemas de xestión do almacenamento.....	40
Sistemas de xestión de almacenamento en big data.....	40
Sistema de Ficheiros Distribuído ou DFS (Distributed File System).....	40
Sistemas de xestión do almacenamento.....	41
Sistemas de xestión de almacenamento en big data.....	41
Sistema de Ficheiros Distribuído ou DFS (Distributed File System).....	41
Sistemas de xestión do almacenamento.....	42
Sistemas de xestión de almacenamento en big data.....	42
Sistema de Ficheiros Distribuído ou DFS (Distributed File System).....	42
Vantaxes dun Sistema de Ficheiros Distribuído ou DFS.....	42
Sistemas de xestión do almacenamento.....	43
Sistemas de xestión de almacenamento en big data.....	43
Sistema de Ficheiros Distribuído ou DFS (Distributed File System).....	43
Vantaxes dun Sistema de Ficheiros Distribuído ou DFS.....	43
Transparencia.....	43
Sistemas de xestión do almacenamento.....	44
Sistemas de xestión de almacenamento en big data.....	44
Sistema de Ficheiros Distribuído ou DFS (Distributed File System).....	44
Vantaxes dun Sistema de Ficheiros Distribuído ou DFS.....	44
Mobilidade do recurso.....	44
Rendemento.....	44

Simplicidade no uso e acceso.....	44
Alta dispoñibilidade.....	44
Alta escalabilidade.....	44
Sistemas de xestión do almacenamento.....	45
Sistemas de xestión de almacenamento en big data.....	45
Sistema de Ficheiros Distribuído ou DFS (Distributed File System).....	45
Vantaxes dun Sistema de Ficheiros Distribuído ou DFS.....	45
Alta fiabilidade.....	45
Integridade da información.....	45
Heteroxeneidade.....	45
Seguridade.....	45

Clusterización

Procesamento paralelo ou paralelizado / distribuído

- **Solución:** utilizar **varios procesadores** para executar distintas partes da tarefa de forma simultánea.
 - **Sistemas paralelos:** dotar ós sistemas de varios procesadores (“cores”)
 - Paradigma seguido polos procesadores modernos (multicore): un número crecente de procesadores, non necesariamente máis rápidos.
 - **Sistemas distribuídos:** múltiples procesadores distribuídos en máquinas diferentes, conectados entre si por algún tipo de rede de comunicación.
- **¡As melloras de rendemento non son automáticas!: o software debe implementarse de tal forma que poida explotarse o paralelismo.**



Fonte: https://subscription.packtpub.com/book/application_development/9781787126992/1/ch01/vl1sec10/parallel-versus-distributed-computing

PARALLEL COMPUTING VERSUS DISTRIBUTED COMPUTING	
PARALLEL COMPUTING	DISTRIBUTED COMPUTING
Type of computation in which many calculations or the execution of processes are carried out simultaneously.	A system whose components are located on different networked computers, which communicate and coordinate their actions by passing messages to one another.
Occurs in a single computer	Involves multiple computers
Multiple processors execute multiple tasks at the same time	Multiple computers perform tasks at the same time
Computer can have shared memory or distributed memory	Each computer has its own memory
Processors communicate with each other using a bus	Computers communicate with each other via the network
Increase the performance of the system	Allows scalability, sharing resources and helps to perform computation tasks efficiently
Visit www.PEDIAA.com	

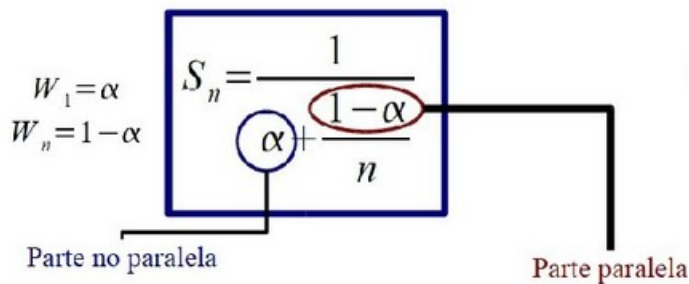
Fonte: <https://pediaa.com/what-is-the-difference-between-parallel-and-distributed-computing/>

Clusterización

Procesamento paralelo ou paralelizado / distribuído

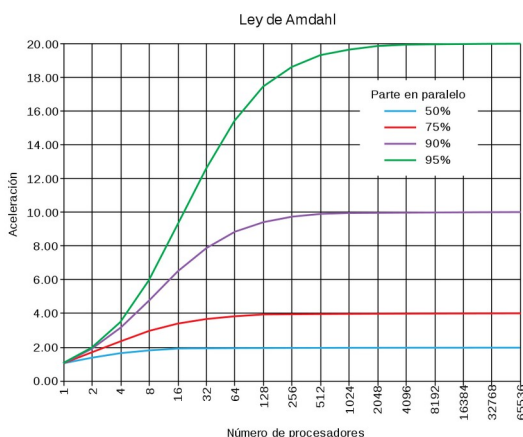
Ley de Amdahl

- As melloras no tempo de execución son totalmente dependentes do tipo de computación a realizar: **certas partes poden ser paralelizables, outras non**
- A Ley de Amdahl permite facer unha estimación da mellora de rendemento máxima teórica



Fonte: <https://slidetodoc.com/tema-3-leyes-sobre-el-aumento-de-prestaciones/>

- Na práctica, a mellora é **sempre menor á máxima teórica**:
 - a implementación práctica de algoritmos paralelos/distribuídos implica tarefas de preparación, xestión, limpeza que consumen tempo de execución e recursos:
 - consolidar resultados, acceso a recursos compartidos, distribución de carga de traballo,...



O incremento de velocidade dun programa utilizando múltiples procesadores en computación distribuída está limitado pola fracción secuencial do programa.

- Por exemplo, se a porción 0'5 do programa é secuencial, o incremento de velocidade máximo teórico con computación distribuída será de 2 ($1/(0'5 + (1 - 0'5)/n)$) cando n sexa moi grande.

Fonte: <https://slidetodoc.com/tema-3-leyes-sobre-el-aumento-de-prestaciones/>

Clusterización

Procesamento paralelo ou paralelizado / distribuído

Paradigmas de paralelismo clásicos

- **Programación baseada en Fíos (“threads”):**
 - Un procesador con múltiples núcleos é capaz de executar **múltiples tarefas á vez**.
 - As **linguaxes de programación** ofrecen funcións específicas para lanzar, sincronizar e comunicar múltiples procesos cun obxectivo común.
 - Necesita soporte explícito do **sistema operativo**.
 - Exemplo: Threads en programación Java
- **Programación baseada en Paso de Mensaxes**
 - Paradigma **distribuído**: os cálculos realízanse en memoria local, os **nodos intercambian datos** enviando e recibindo mensaxes
 - Un ou máis **nodos “Mestre”** coordinan os cálculos que realizan os diferentes nodos para alcanza-lo obxectivo común.
 - Exemplo: librarías MPI, Message Passing Interface, (API para a programación paralela baseada no paso de mensaxes para a comunicación e sincronización dos procesos).
- **Limitacións (en ambos casos):** desenvolvemento complexo (acceso a recursos compartidos, interbloqueos), dificultades para a depuración (“debugging”), ...

Fonte: Curso CAFI – S2101132 - Ano 2021 – edu.xunta.gal. MP5075: MÓDULO PROFESIONAL BIG DATA APLICADO. Impartido por: Antonio Fernández Bravo

Clusterización

Procesamento paralelo ou paralelizado / distribuído

Novo paradigma: “commodity cluster”

- “**Commodity cluster**”: nova arquitectura que permite procesar de forma escalable e predecible volumes elevados de datos (Petabytes)
- Gran cantidade de **ordenadores estándar de uso xeral**, non especialmente potentes
- **Rede de comunicación estándar** (e.g. Gigabit Ethernet)



Fonte: https://en.wikipedia.org/wiki/Computer_cluster

- Esta arquitectura inspírase nunha serie de proxectos académicos de computación distribuída da década dos 90 do século XX:
- Great Internet Mersenne Prime Search (GIMPS)
- MilkyWay@HOME
- Einstein@HOME
- SETI@HOME: <https://en.wikipedia.org/wiki/SETI@home>

Clusterización

Procesamento paralelo ou paralelizado / distribuído

Novo paradigma: puntos clave

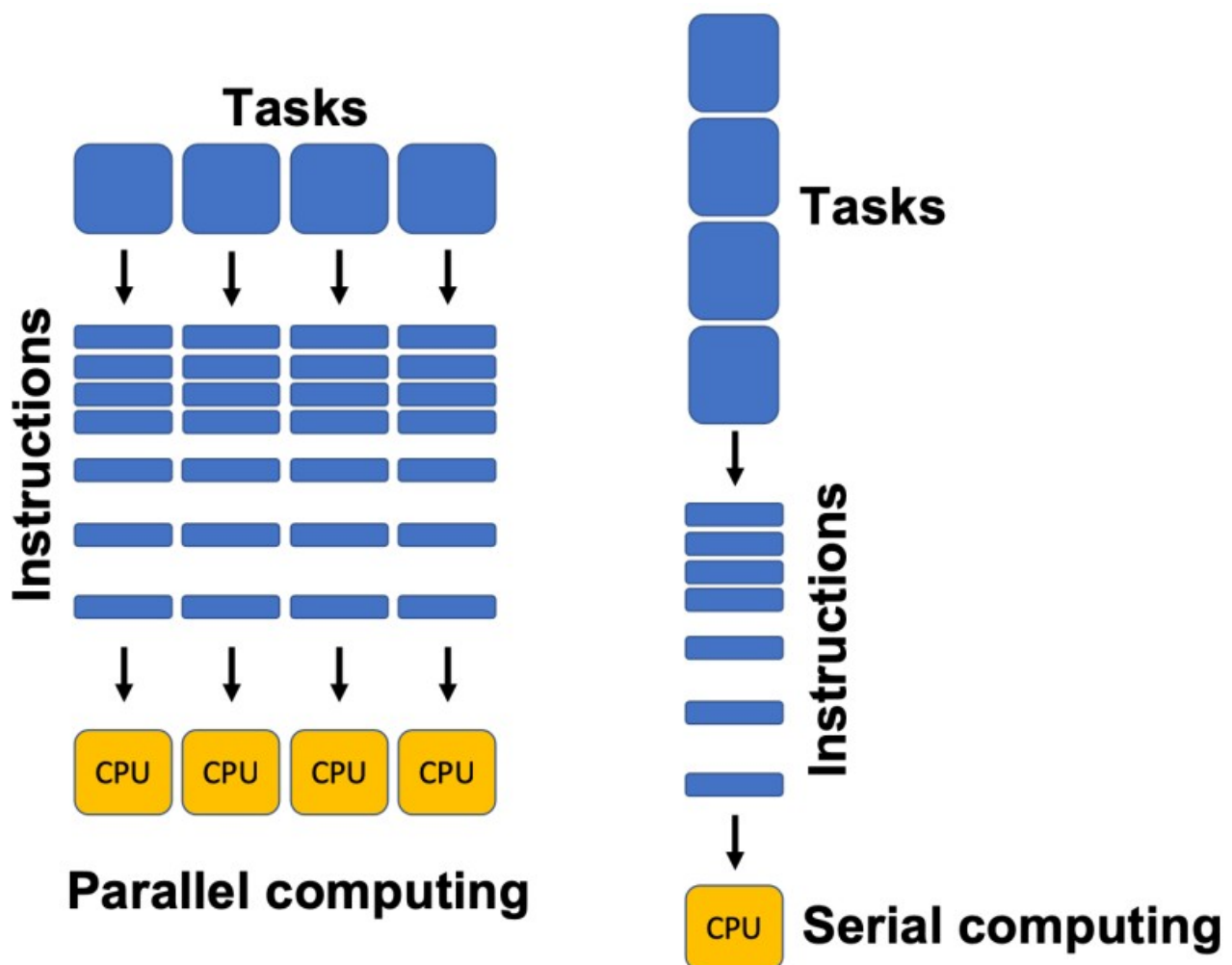
Problemas do paralelismo clásico	Novo enfoque
Primitivas de baixo nivel Depuración complexa Código difícil de manter e reutilizar	Abstraccións de alto nivel que se encarguen dos “detalles” e distribúan as tarefas entre os nodos.
Como distribuír datos entre as máquinas? Que ocorre se as maquinas fallan?	Os ficheiros almacénanse múltiples veces nun Sistema de Arquivos distribuído (DFS) para conseguir tolerancia a fallos
Como distribuí-las tarefas entre os nodos?	Non realizar transferencias de datos, organiza-lo procesamento en torno á distribución existente (“Bring computation to the data”)

Fonte: Curso CAFI – S2101132 - Ano 2021 – edu.xunta.gal. MP5075: MÓDULO PROFESIONAL BIG DATA APLICADO. Impartido por: Antonio Fernández Bravo

Clusterización

Procesamento paralelo ou paralelizado / distribuído

- A **computación en clúster** é vista frecuentemente como unha forma de computación paralela
 - na que unha tarefa computacionalmente difícil, é dividida en subtarefas máis asequibles.
 - Estas subtarefas son executadas en paralelo
 - en diferentes nodos (servidores) dun clúster.
- Importante: unha tarefa só é executable en paralelo ...
 - ... se non existen dependencias entre as subtarefas.



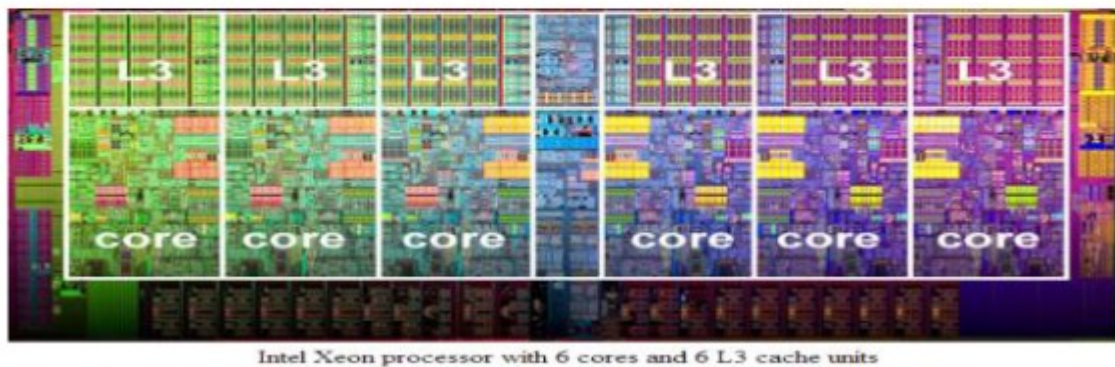
Procesamento paralelo nun computador con múltiples procesadores (ou cun procesador multicore, ou cun superordenador con múltiples nodos) vs procesamento serie nun ordenador monoprocesador.

Fonte: <https://pythonnumericalmethods.berkeley.edu/notebooks/chapter13.01-Parallel-Computing-Basics.html>

Clusterización

Procesamiento paralelo ou paralelizado / distribuído

Fig. 9 shows an example of a parallel computer with multi-core and multiple machines designed for parallelism connected/networked together to form a massively parallel architecture [2].



- Networks connect multiple stand-alone computers (nodes) to make larger parallel computer clusters.

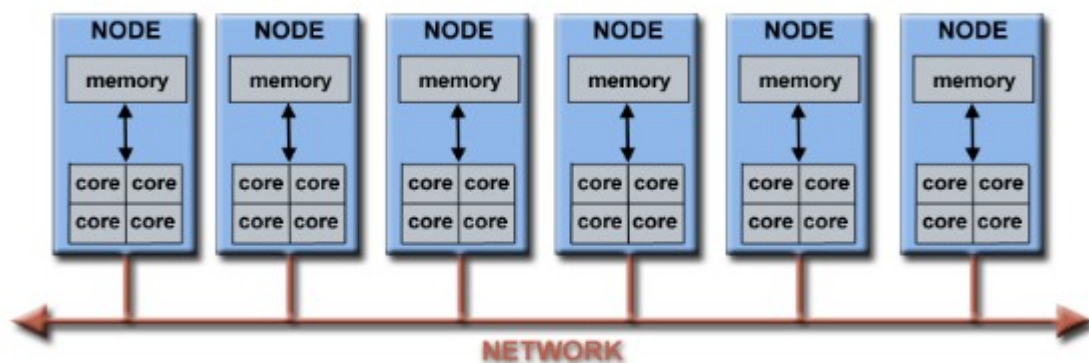
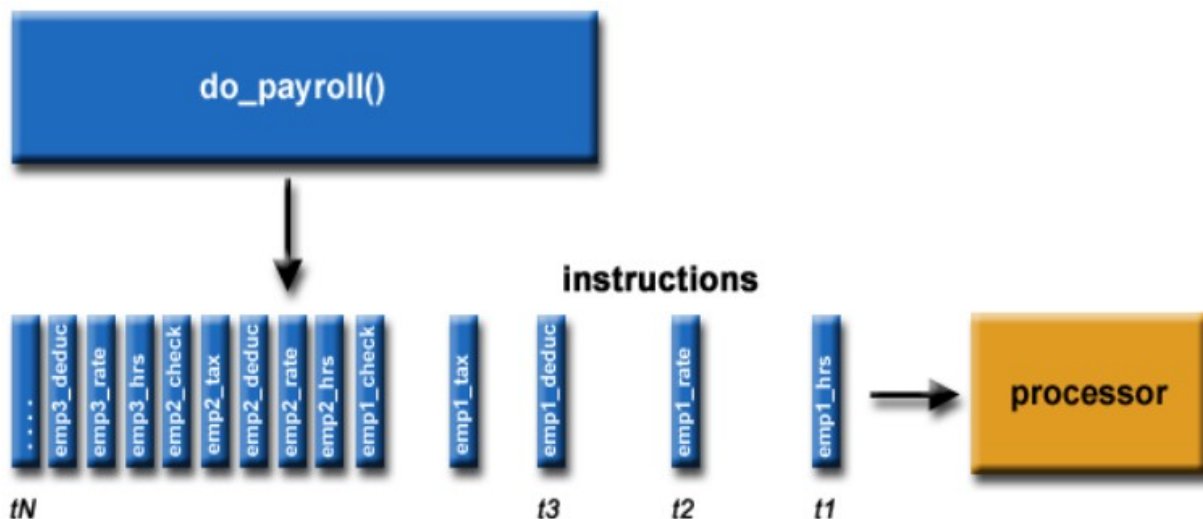


Fig. 9: Multi-core on single machine architecture and single machines with multi-core on network architecture to provide massively parallel processing

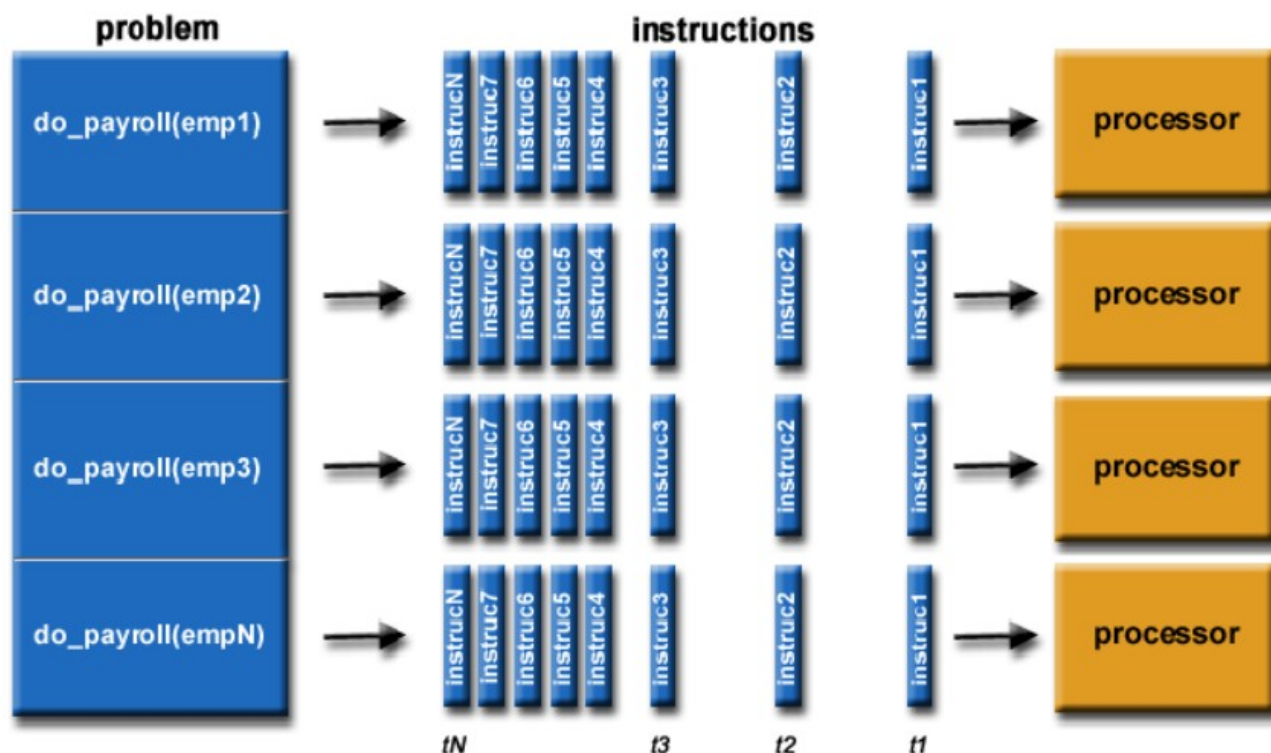
Fonte: <https://www.semanticscholar.org/paper/Parallel-Computing-Hardware-and-Software-for-High-Clevenger-Eng/a6dfbc155ee750012b73f4e62f2a4c9b63334598>

O procesamento paralelo consiste nun problema que se divide en subtarefas, asignadas a diferentes procesadores, nos que as instrucións de cada subtarefa vanse procesando en paralelo.

- Exemplos prácticos de subtarefas paralelizables:
 - Proceso de cálculo de nóminas.



Na computación serie, o proceso realízase repetindo as mesmas instrucións para cada empregada, unha a unha, sen que empece unha tarefa de cálculo dunha empregada ata que finalice a da anterior.



O mesmo proceso de nóminas escrito en estilo de computación paralelo, procesa o cálculo de varias empregadas simultaneamente, mellorando significativamente o tempo necesario.

Fonte: <https://www.semanticscholar.org/paper/Parallel-Computing-Hardware-and-Software-for-High-Clevenger-Eng/a6dfbc155ee750012b73f4e62f2a4c9b63334598>

- Proceso de cálculo de matrices.

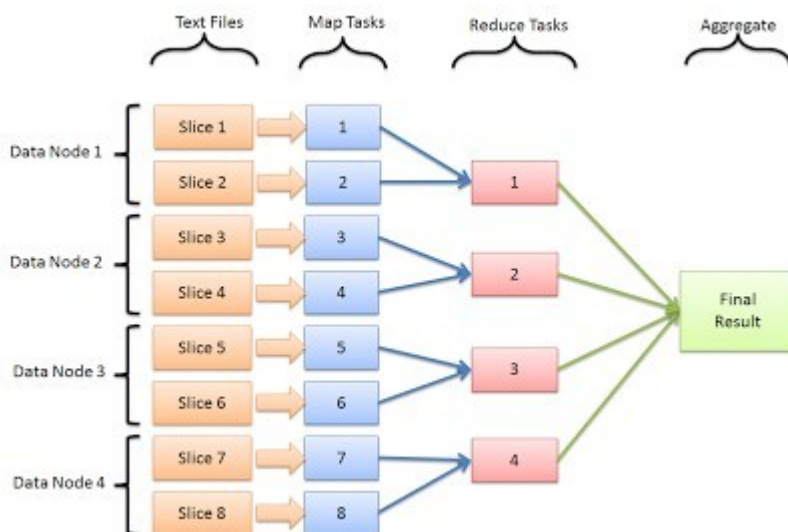
$$\begin{array}{c}
 \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 1 & 3 & 2 \end{pmatrix} \quad \begin{pmatrix} 10 & 11 \\ 7 & 5 \\ 2 & 4 \end{pmatrix} \\
 3 \times 3 \quad \quad 3 \times 2
 \end{array}
 =
 \begin{array}{c}
 \begin{pmatrix} 1*10+2*7+3*2 & 1*11+2*5+3*4 \\ 4*10+5*7+6*2 & 4*11+5*5+6*4 \\ 1*10+3*7+2*2 & 1*11+3*5+2*4 \end{pmatrix} \\
 3 \times 2
 \end{array}$$

- Google MapReduce

Este tipo de tecnoloxía (MapReduce) permitiu a Google indexar toda a web.

Bo resumo en https://www.slideshare.net/romain_jacotin/the-google-mapreduce, baseado en “MapReduce:

Simplified Data Processing on Large Clusters” <https://research.google.com/archive/mapreduce-osdi04-slides/index.html>

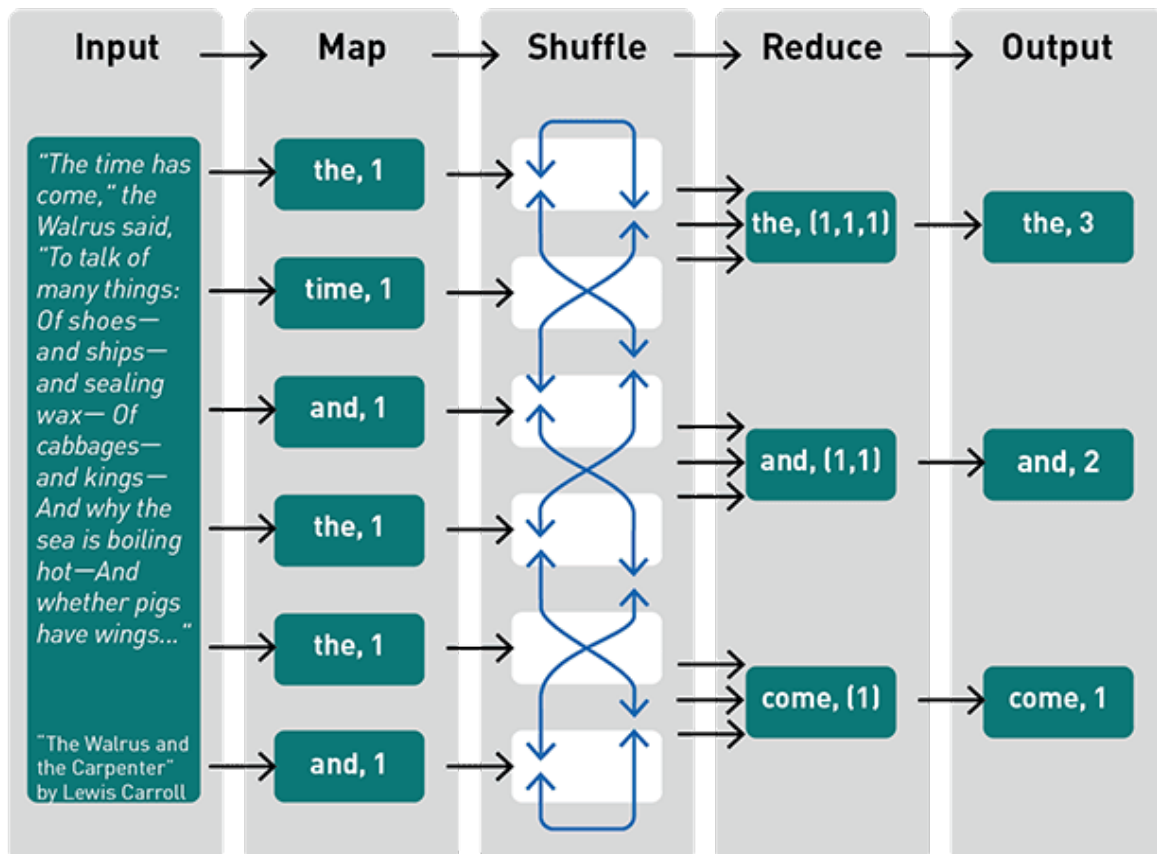


Clusterización

Procesamento paralelo ou paralelizado / distribuído

Exemplo de uso do concepto MapReduce:

Uso de MapReduce para contar palabras en logs, blogs, recomendacións, etc.



Fonte: <https://developer.hpe.com/blog/spark-101-what-is-it-what-it-does-and-why-it-matters/>

“A estratexia de Google baséase en tres conceptos fundamentais:

- **Distribuí-los datos:** cando se carga un arquivo de datos no clúster, divídese en anacos, chamados bloques de datos, e distribúense entre os nodos de datos e replícase no clúster.
- **Distribuí-lo cálculo:** os/as usuarios/as especifican unha función **map** que procesa un par chave/valor para xerar un conxunto de pares intermedios chave/valor e unha función **reduce** que fusiona tódolos valores intermedios asociados á mesma chave intermedia. Os programas escritos neste estilo funcional paralelízanse automaticamente e execútanse nun gran cluster de máquinas *commodity* (ordenadores de uso xeral, non necesariamente servidores) do seguinte xeito:
 - O proceso de mapeo execútase en cada nodo de datos asignado, traballando só no seu bloque de datos dun arquivo distribuído.

- Os resultados dos procesos de mapeo envíanse ós redutores nun proceso chamado "barallar e ordear": os pares chave/valor dos mapeadores ordénanse por chave, divídense polo número de redutores e logo envíanse a través da rede e escríbense en "arquivos de secuencia" ordeados por chave nos nodos redutores.
- O proceso redutor execútase no seu nodo asignado e traballa só no seu subconxunto de datos (seu arquivo de secuencia). A saída do proceso redutor escríbese nun arquivo de saída.
- **Tolerar fallos:** tanto os datos como o cálculo poden tolerar fallos, mediante el paso a outro nodo para os datos ou o procesamento.”

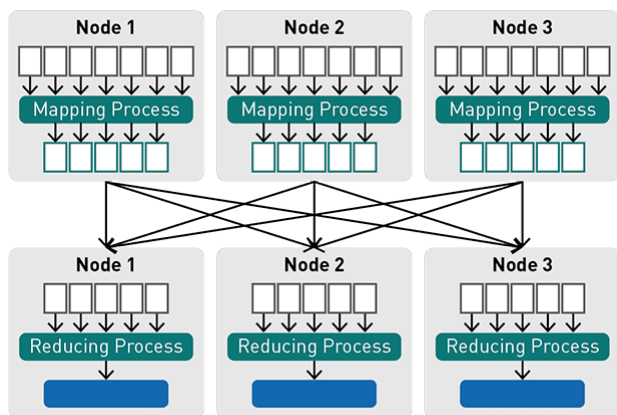
Diapositivas explicativas de Google Map Reduce (<https://research.google/pubs/pub62/>):
<https://research.google.com/archive/mapreduce-osdi04-slides/index.html>

Clusterización

Procesamento paralelo ou paralelizado / distribuído

Os tres conceptos principais tra-lo procesamento paralelo.

- **Distribución da información**
 - A información divídese en porcións cando se carga no clúster.
 - Estas porcións denomínanse bloques de datos ou data blocks
 - Cada un dos bloques de datos son distribuídos...
 - ... entre os nodos do clúster ...
 - ... e son replicados ó ancho do clúster.
- **Distribución da computación**
 - A información dividida en bloques de datos é procesada mediante diferentes funcións.
 - Cada función actúa sobre a información dispoñible no seu nodo (denomínase “mapping”)
 - Os resultados do mapping son enviados ós chamados “reducers”
 - Estes procesan os resultados nun proceso chamado "shuffle and sort" mediante key/value ou chave/valor.
- **Tolerancia a fallos**
 - Tanto os datos como a capacidade de computación.
 - poden tolerar fallos nalgún dos nodos
 - transferindo os procesos do/s nodo/s caídos
 - ó resto de nodos activos.



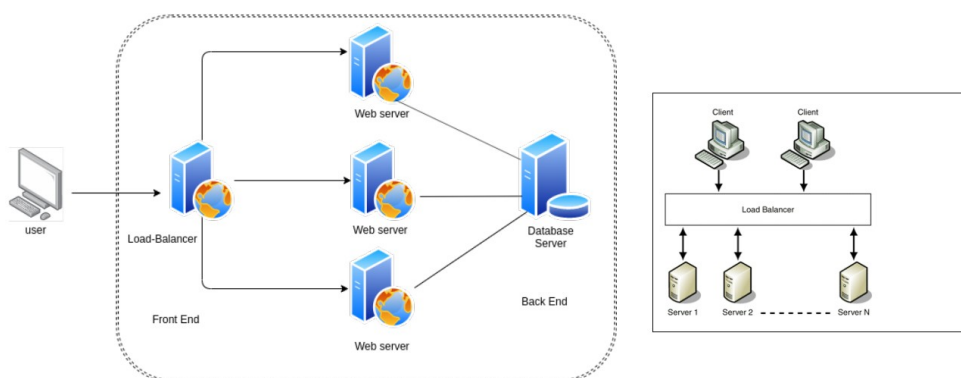
Fonte: <https://developer.hpe.com/blog/spark-101-what-is-it-what-it-does-and-why-it-matters/>

Clusterización

Procesamento paralelo ou paralelizado / distribuído

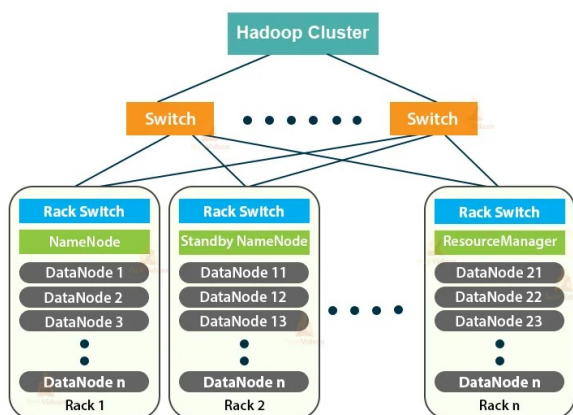
Arquitectura clúster

- Un cluster é unha agrupación de servidores/nodos.
- Por exemplo, nun cluster web, tódalas peticións de usuario/a diríxense ó balanceador de carga.
 - A función dun balanceador é simplemente despacha-las peticións ós nodos, en función do algoritmo que teña para a súa elección.
 - Os nodos, neste caso un servidor web, son os que realizan o proceso para resolve-la petición.



Fonte: <https://www.znetlive.com/blog/how-cluster-architecture-benefits-data-base-web-applications/>

- No caso dun cluster Hadoop:
 - Hai unha serie de switches para conmutar entre os ...
 - ... racks que teñen memoria, discos,... e varios nodos (servidores).
 - Os nodos son os encargados do procesamento da información.



Fonte: <https://techvidvan.com/tutorials/hadoop-cluster/>

Clusterización

Procesamento paralelo ou paralelizado / distribuído

Servizos ofrecidos por un clúster

- Os clusters clasifícanse en función do servizo que prestan
 - Clúster de almacenamento.
 - Permiten ofrecer un sistema de almacenamento fiable ós sistemas.
 - Isto permite asegura-la lectura e escrituras dos nodos e servidores
 - Este tipo de clusters permite elimina-la realización de copias de datos
 - Simplifica as copias de seguridade e disaster recovery.
 - Clusters de alta dispoñibilidade
 - Son grupos de computadoras que permiten.
 - Mante-lo funcionamento ininterrumpido de aplicacións críticas
 - Permiten reduci-la necesidade de copias de seguridade.
 - Eliminan o punto único de fallo.
 - Clusters balanceadores de carga
 - Permiten o reparto de carga entre diferentes nodos.
 - Permiten absorber picos de traballo, sen merma da calidade do servizo para o/a cliente/a.
 - Permiten un mellor aproveitamento dos recursos computacionais
 - Clusters de alto rendemento.
 - Ofrecen execución concorrente de aplicacións ...
 - ... co fin de permiti-la computación de alta esixencia de recursos
 - Permiten mellora-lo rendemento de aplicacións ...
 - ... científicas, simulacións, médicas, análise de bolsa, compañías aéreas, meteoroloxía, etc.

Cal dos seguintes sería a definición máis acertada de “clúster de computadoras”?

- a) Uso de recursos informáticos distribuídos para procesamento común.
- b) Conxunto de computadoras que traballan xuntas para que poidan verse como un só sistema.
- c) Computadora cun alto rendemento en comparación cunha computadora de uso xeral.
- d) Todas as anteriores son verdadeiras

Exercicio LAB 2.1 – HADOOP e MAPREDUCE

Obxectivos

- Primeira toma de contacto con MapReduce e Hadoop, a súa implementación máis estendida
- Instalación e configuración de Hadoop na súa configuración máis sinxela (modo Standalone)
- Executar un programa de exemplo seguindo o paradigma MapReduce
 - En MapReduce “contar palabras” é o equivalente ó “Ola Mundo”, usado como primeiro exemplo de iniciación nas linguaxes de programación convencionais.

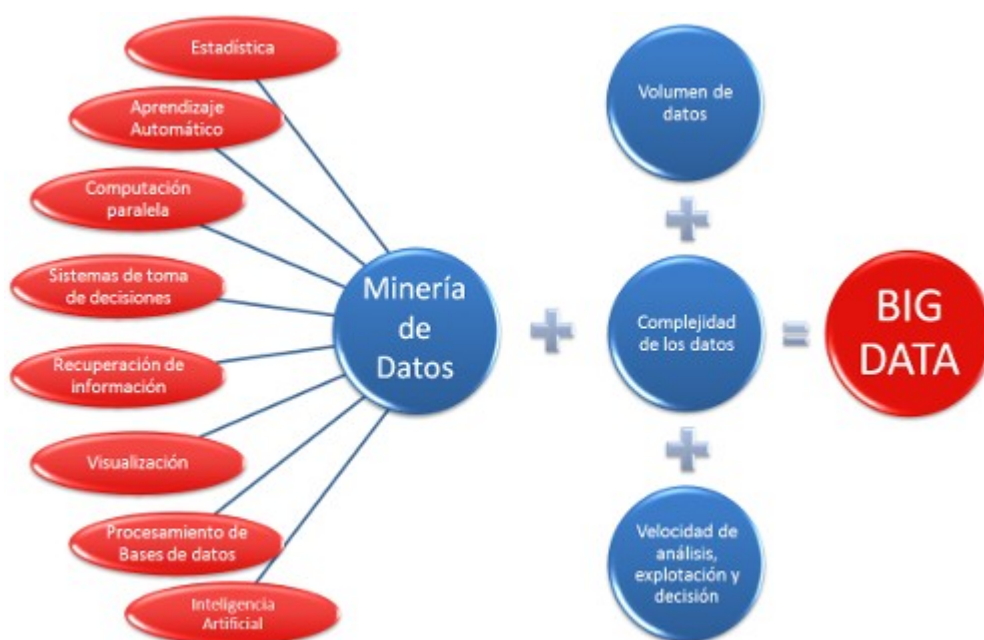
Programación con **Hadoop**:

- **MapReduce** ocúpase das fases de mapeo e redución (computación distribuída)

Sistemas de xestión do almacenamento

Sistemas de xestión de almacenamento en big data

- É unha arquitectura mixta computadora-almacenamento (SAN: Storage Area Network).
- Empregada para almacenar e xestionar **grandes cantidades de información**.
 - Para a realización de **análise de información en tempo real**.
 - Soen aparecer en forma de obxectos e **ficheiros**.
 - Os datos dentro do big data atópanse **desestruturados**.
 - Este tipo de almacenamento contén terabytes ou petabytes, ...
 - O almacenamento pode ser en HD (por custe) ou “flash” (por su rapidez)
- **Deseñáronse especificamente para estas necesidades ...**
 - **... porque os sistemas tradicionais non servían para isto.**



Fonte: <https://www.incibe.es/protege-tu-empresa/blog/mineria-datos-big-data-seguridad>

Sistemas de xestión do almacenamento

Sistemas de xestión de almacenamento en big data

- Os sistemas de ficheiros **especificamente deseñados para big data**
 - poden ser **baseados en disco**, como HDFS
 - **ou baseados en memoria** como Oracle Big Data File System (BDFS)
- Son sistemas que han de permiti-lo almacenamento e acceso a milleiros de billóns de rexistros (Terabytes, Petabytes, ...)
 - Permitir tolerancia a fallos porque ...
 - ... en miles de nodos é inevitable que constantemente fallen varios deles.
 - “Nun clúster de 10.000 nodos, segundo as estatísticas, 10 nodos estragaranse tódolos días.”
 - Cando se teñen enormes cantidades de datos, deben almacenarse nalgún lugar. Hai dúas opcións:
 - Escalado cara arriba ou **escalado vertical**: obter un nodo de gran capacidade.
 - Consegue un disco máis grande, pero nada é infinito: no 2014 Facebook tiña un almacenamento de 300 petabytes de datos; sabes onde comprar 300 petabytes disco?
 - Escalar cara a fóra ou **escalado horizontal**: almacena-los datos nunha colección de nodos.
 - Estatisticamente calcúlase que un nodo prestará servizo 3 anos de media. Polo tanto, cun grupo de 1.000 nodos teríase un fallo cada día, aproximadamente.
 - Cada enfoque de escalado ten os seus propios pros e contras.
 - Ó acceder ós datos, normalmente obtense unha latencia máis baixa co escalado vertical.
 - Co escalado horizontal hai maior latencia, pero pódese construír un almacenamento máis grande.

Hai unha gran diferenza entre ter 2 TB de disco duro por nodo ou 2 discos duros por nodo de 1 TB cada un. Por exemplo, cun disco SSD Samsung 940 Pro cunha velocidade de lectura de disco de 3'5 GBps tardaríanse en ler 10 PB de datos dun disco duro, aproximadamente, 35 días; pero se se tivesen 5000 unidades, poderíanse ler todos estes datos en paralelo en 10 minutos. Por tanto, a cantidade de unidades nun clúster ten unha relación lineal coa velocidade de procesamento de datos.

Sistemas de xestión do almacenamento

Sistemas de xestión de almacenamento en big data

Sistemas de arquivos distribuídos

- Permiten **almacenar datos de forma distribuída** nun entorno onde tanto discos como máquinas poden fallar en calquera momento.
- **Optimizados para** os patróns de **acceso habituais en tarefas de análise de datos**:
 - Grandes volumes de información (GB ou TB).
 - Fundamentalmente operacións de lectura e agregación (“append”).
 - Actualizacións (“update”) de datos pouco frecuentes.
- Teñen un espazo de nomes (“Namespace”) global.
- Algúns exemplos: GFS (Google), HDFS (Ecosistema Hadoop).

O sistema GFS liberado por Google: <https://research.google/pubs/pub51/>

Sistemas de xestión do almacenamento

Sistemas de xestión de almacenamento en big data

Sistemas de arquivos distribuídos

- Os datos almacénanse varias veces (**replícanse**) para conseguir **tolerancia a fallos**:
 - Divídense en anacos (“chunks”) de 32/64/128 MB e replícanse en varios nodos (“Data Nodes”).
 - Por exemplo, o arquivo C, dun tamaño de 360 MB, trocéase en tres cachos: C0, C1 e C2; os dous primeiros de 128 MB e o último de 104 MB.
 - O trozo C0 gárdase no nodo 1 e unha réplica no nodo N.
 - O trozo C1 gárdase no nodo 1 e unha réplica no nodo 3.
 - O trozo C2 gárdase no nodo 2 e unha réplica no nodo 3.
 - Un nodo mestre (“Master Node”) rexistra os metadatos.
 - Os nodos cliente consultan o nodo mestre, que identifica onde se atopan as porcións a ler. Con esa información, o cliente recupéraos directamente de cada nodo.



Fonte: Curso CAFI – S2101132 - Ano 2021 – edu.xunta.gal. MP5075: MÓDULO PROFESIONAL BIG DATA APLICADO. Impartido por: Antonio Fernández Bravo

Sistemas de xestión do almacenamento

Sistemas de xestión de almacenamento en big data

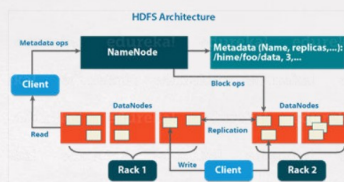
Sistemas de arquivos distribuídos: HDFS (Hadoop File System)

- HDFS é un **sistema de ficheiros distribuído** (non ten que estar a información nun único lugar, senón que se pode atopar incluso ó redor de todo o mundo) ...
 - ... que permite xestionar **grandes cantidades de datos** ...
 - ... en **hardware non especializado** (de uso xeral)
 - Permite escala-los clusters de Apache Hadoop a **centos ou milleiros de nodos**.
 - É un dos compoñentes principais de **Apache Hadoop**, xunto con ...
 - ... MapReduce (encargado de distribuír e procesar en paralelo, e logo xuntar e ordena-la información) ...
 - ... e YARN (encargado de planifica-los traballos nun cluster e xestiona-los recursos hardware dos que se dispón para executa-las aplicacións Big Data) ...

HDFS Y YARN: ALMACENAMIENTO Y PROCESAMIENTO

- **HDFS: Almacenamiento**

- Hadoop Distributed File System.
- Sistema de almacenamiento de Hadoop.
- Apariencia de un único sistema de ficheros, similar a Linux.
- Es altamente escalable y tolerante a fallos.



HDFS Y YARN: ALMACENAMIENTO Y PROCESAMIENTO

- **MapReduce: Procesamiento**

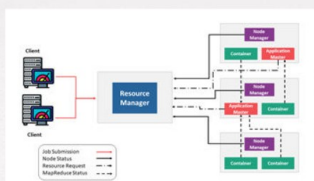
- MapReduce permite el procesamiento distribuido sobre HDFS.
- Permite obtener Data Locality (revolución tecnológica de Big Data).
- Es el primer paradigma de programación de Hadoop.
 - Con una fase de Mapeo (Map), y otra de Agrupación (Reduce).
- Su funcionamiento lo ilustra un "Contar palabras".



HDFS Y YARN: ALMACENAMIENTO Y PROCESAMIENTO

- **YARN: Gestión de recursos**

- YARN gestiona los recursos (CPU y RAM) del cluster para procesar
- Es tolerante a fallos y permite multitud de ejecuciones en paralelo.
- Gestiona cuotas de ejecución según prioridad, necesidad, disponibilidad, ...
- Apareció con MapReduce, pero puede trabajar con otros frameworks de procesamiento, como Spark.



Fonte: <https://openwebinars.net/academia/aprende/big-data-general/9131/>

- HDFS é a capa de almacenamento de Hadoop, e como tal, é un sistema de ficheiros distribuído e tolerante a fallos que pode almacenar gran cantidade de datos, escalar de forma incremental e sobrevivir a fallos de hardware sen perder datos.
- É un sistema que reparte os datos entre tódolos nodos do clúster de Hadoop, dividindo os ficheiros en bloques (cada bloque por defecto é de 128MB) e almacenando copias duplicadas a través dos nodos. Por defecto se replica en 3 nodos distintos (isto coñécese como o factor de replicación).
- HDFS asegura que se poidan engadir servidores para incrementa-lo tamaño de almacenamento de forma lineal, de maneira que ó introducir un novo nodo, incrementase tanto a redundancia como a capacidade de almacenamento.
- Está deseñado para escribi-los datos unha vez e lelos moitos veces (**WORM / Write Once, Read Many**).
 - HDFS NON ofrece bo rendemento para:
 - Accesos de baixa latencia. Realmente utilízase para almacenar datos de entrada necesarios para procesos de computación.
 - Ficheiros pequenos (a menos que se agrupen). Funciona mellor con “pequenas” cantidades de ficheiros grandes, é dicir, mellor millóns de ficheiros de 100MB que billóns de ficheiros de 1MB.
 - Múltiples escritores.
 - Modificacións arbitrarias de ficheiros (non soporta actualizacións e lecturas aleatorias).
 - Así pois, os datos, unha vez escritos en HDFS son immutable. Cada ficheiro de HDFS só permite engadir contido (*append-only*). Unha vez creado e escrito nel, só podemos engadir contido ou eliminalo. É dicir, a priori, non podemos modifica-los datos.

Sistemas de xestión do almacenamento

Sistemas de xestión de almacenamento en big data

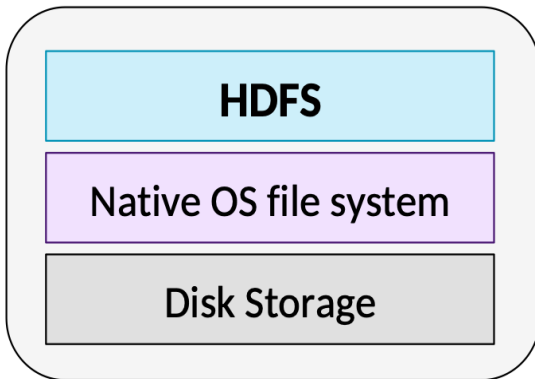
Sistemas de arquivos distribuídos: HDFS (Hadoop File System)

- Rápida recuperación en caso de fallo
 - Dado que HDFS está pensado para dar servizo a ...
 - ... centos ou milleiros de nodos ...
 - ... a caída dalgún deles é inevitable.
 - Tempo medio entre fallos para 1 nodo = 3 anos
 - Tempo medio entre fallos para 1.000 nodos = 1 día
 - “Nun cluster de 10.000 nodos, segundo as estatísticas, 10 nodos estragaranse tódolos días.”
 - HDFS está deseñado para a pronta detección de fallos ...
 - ... e recuperarse con rapidez.
 - Acceso a streaming de datos.
 - HDFS está pensado principalmente para procesos por lotes, para lecturas de ficheiros en grandes streamings, en contraposición ó uso interactivo de datos, ...
 - ... por iso, o seu deseño está optimizado para maximiza-la transferencia de datos, ...
 - ... o que lle permite **servir streaming de datos**.
 - Soporte de grandes conxuntos de datos.
 - HDFS permite xestionar volumes da orde de Gigabytes ata Terabytes, ...
 - Permite aumenta-lo ancho de banda de acceso a datos ...
 - ... e con iso, escalar ata centos ou milleiros de nodos nun único cluster.

Se engadimos nodos, auméntase o ancho de banda de lectura/escritura, dado que eses nodos van a estar accedendo en paralelo ós ficheiros/datos simultaneamente xunto co resto.

- Portabilidade
 - Co obxecto de facilita-lo seu uso ...
 - HDFS permite a súa extensión entre múltiples tipos de hardware
 - Discos SATA, SAS,...
 - e é compatible con distintos tipos de sistemas operativos como...

- Nativos de linux (ext3, ext4, xfs), HP-UX, AIX, VOS400, Windows, etc



Sistemas de xestión do almacenamento

Sistemas de xestión de almacenamento en big data

Sistemas de arquivos distribuídos: HDFS (Hadoop File System)

Caso práctico

- Unha base de datos que inclúa os datos (nome, teléfono, dirección), de tódolos habitantes dun país.
 - Os nomes que empezan pola letra A no servidor 1, os que empezan por B no 2, etc.
 - Para previr fallos, HDFS clona cada ficheiro en 2 servidores (polo menos).
 - A subdivisión de datos facilita a Hadoop a escalabilidade ...
 - ... e a aplicación de procesos de forma local (se é posible).

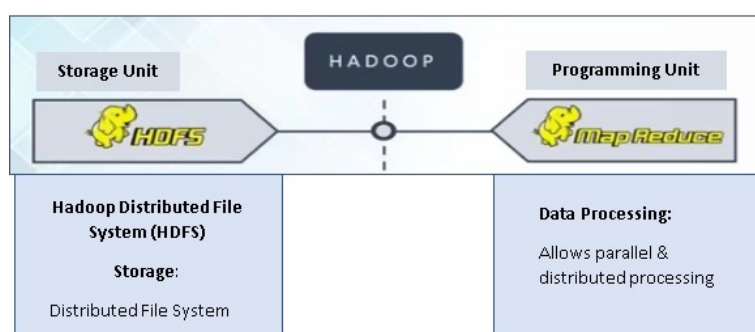
Sistemas de xestión do almacenamento

Sistemas de xestión de almacenamento en big data

Sistemas de arquivos distribuídos: HDFS (Hadoop File System)

Esquema de Sistema de ficheros distribuído Hadoop ou HDFS (Hadoop Distributed File System)

- Hadoop é un marco para procesar Big data, que
 - permite almacenar e procesar conxuntos de datos grandes
 - en paralelo e
 - de forma distribuída.
- Hai dous compoñentes principais de Hadoop: HDFS e MapReduce.
 - Hadoop Distributed File System (**HDFS**) ocúpase da parte de almacenamento da arquitectura de Hadoop.
 - **MapReduce** é un modelo de procesamento e un marco de software para escribir aplicacións que se poden executar en Hadoop. Estes programas de MapReduce son capaces de procesar Big data en paralelo en grandes grupos de nodos computacionais.
 - MapReduce utilízase para o procesamento eficiente dos datos nun sistema de arquivos distribuído.



Hadoop Core Components: hai dous compoñentes principais de Hadoop, HDFS e MapReduce. Fonte: <https://mobilemonitoringsolutions.com/hadoop-for-beginners/>

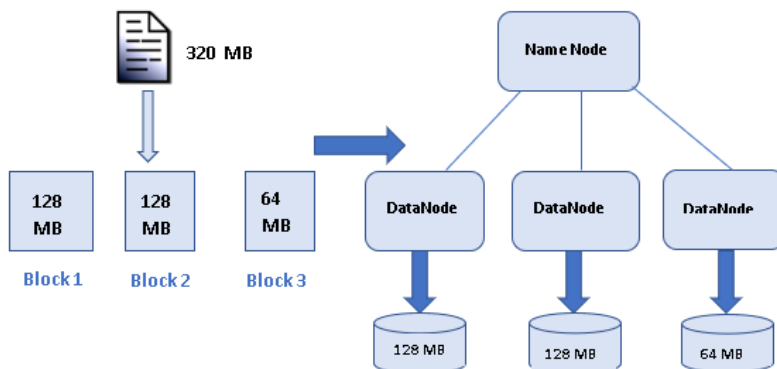
Sistemas de xestión do almacenamento

Sistemas de xestión de almacenamento en big data

Sistemas de arquivos distribuídos: HDFS (Hadoop File System)

Esquema do HDFS (Hadoop Distributed File System)

- HDFS almacena arquivos en moitos nodos dun cluster.
- Hadoop segue a arquitectura mestre-escravo e, polo tanto HDFS, que é o seu compoñente principal, tamén segue a mesma arquitectura.
- Como se almacenan os datos en DataNodes?
 - En bloques de datos HDFS.
 - Un bloque é a cantidade mínima de datos que pode ser lida ou escrita. O tamaño predeterminado de HDFS son 128 MB, xa que Hadoop está pensado para traballar con ficheiros de gran tamaño.
 - Cada arquivo almacénase en HDFS como bloques.
 - Despois de que o arquivo se divide en bloques de datos, estes bloques de datos distribuiranse a través de tódolos nodos de datos presentes no cluster Hadoop.
 - Ademais, cada bloque replicarase para ter varias copias do mesmo (habitualmente 3).



Bloques de datos HDFS. Fonte: <https://mobilemonitoringsolutions.com/hadoop-for-beginners/>

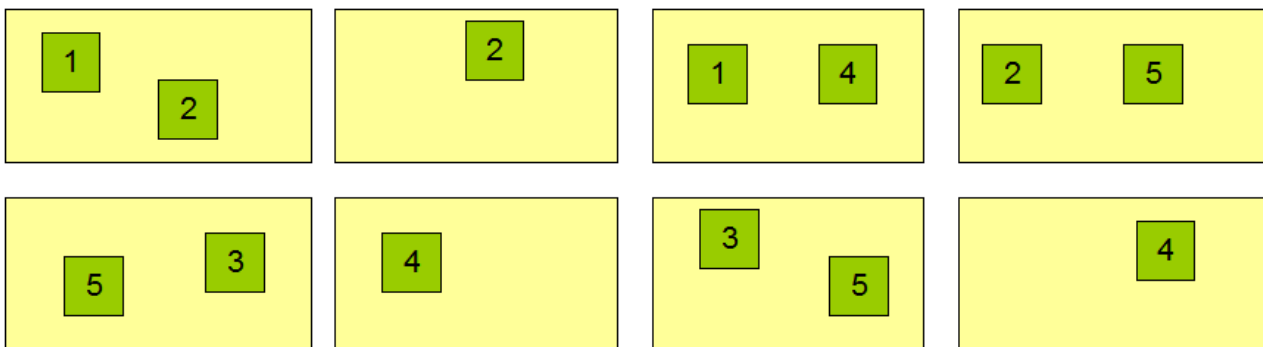
- Se, por exemplo, subimos un ficheiro de 600MB, dividirase en 5 bloques de 128MB. Estes bloques distribúense por tódolos nodos de datos do clúster de Hadoop.
 - Se un ficheiro de HDFS é menor que o tamaño dun bloque, é dicir, menor de 128MB, ocupará un bloque lóxico pero en disco unicamente ocupará o espazo necesario. É dicir, un arquivo de 1MB ocupará un bloque de 128MB, pero utilizará 1MB en disco.
 - A partir do factor de replicación, cada bloque almacénase varias veces en máquinas distintas. O valor por defecto é 3. Por tanto, o arquivo de 600MB que tiñamos

dividido en 5 bloques de 128MB, se o replicamos tres veces, terémolo repartido en 15 bloques entre todos os nodos do clúster.

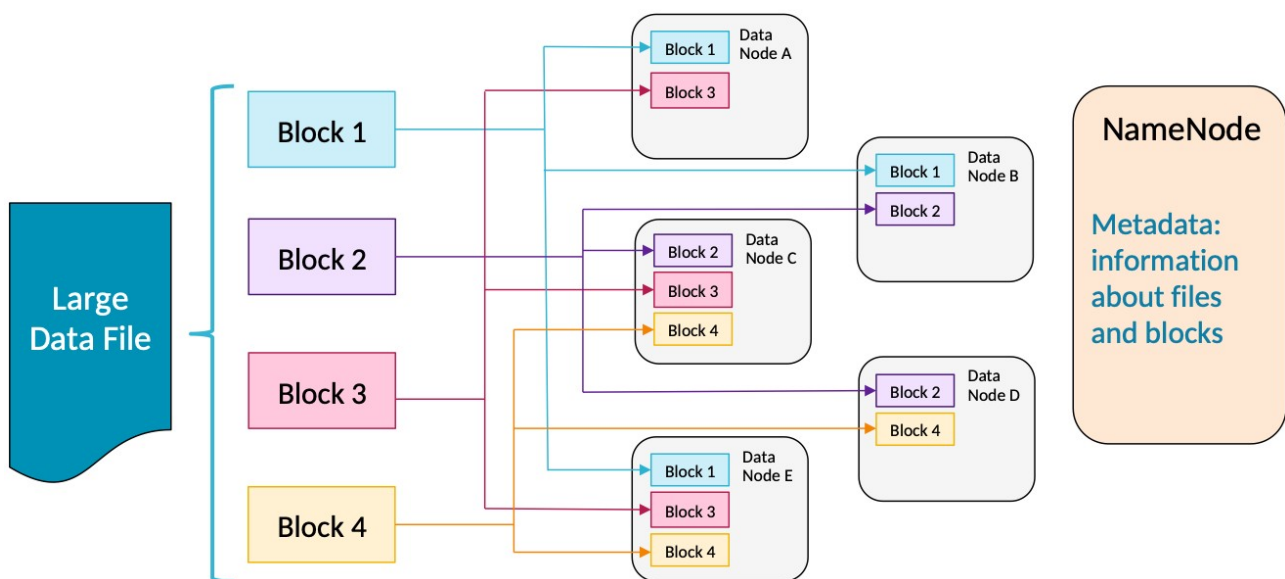
Block Replication

Namenode (Filename, numReplicas, block-ids, ...)
/users/sameerp/data/part-0, r:2, {1,3}, ...
/users/sameerp/data/part-1, r:3, {2,4,5}, ...

Datanodes



Fonte: <https://hadoop.apache.org/docs/r3.3.4/hadoop-project-dist/hadoop-hdfs/HdfsDesign.html>



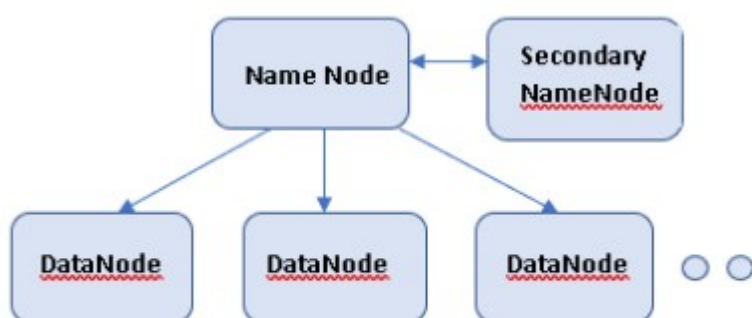
Sistemas de xestión do almacenamento

Sistemas de xestión de almacenamento en big data

Sistemas de arquivos distribuídos: HDFS (Hadoop File System)

Esquema de Sistema de ficheiros distribuído Hadoop ou HDFS (Hadoop Distributed File System)

- **NameNode** e **DataNode** son os compoñentes principais de HDFS, pero tamén existe o **SecondaryNameNode**.



Master- NameNode and Slave - DataNode in HDFS

Arquitectura Hadoop Cluster. Fonte: <https://mobilemonitoringsolutions.com/hadoop-for-beginners/>

- **NameNode:**
 - Actúa como *master* e almacena tódolos metadatos necesarios para construí-lo sistema de ficheiros a partir dos seus bloques. Ten control sobre onde se atopan tódolos bloques.
 - Garda os metadatos dos arquivos, é dicir información sobre bloques de datos, p.e. a localización dos datanodes con bloques almacenados, o nome e tamaño dos arquivos, xerarquía, atributos (tempo de creación, número de copias, permisos, ...), etc.
 - Recibe o estado e o informe de bloqueio de todos os DataNodes.
 - Almacena tódolos datos da xerarquía de carpetas e directorios na memoria RAM (o tamaño típico que ocupa a información de cada obxecto -directorio ou ficheiro- é duns 150 bytes).
 - É importante que o Namenode sexa robusto e non teña caídas; por este motivo, utilízase hardware máis resiliente que o dos Datanodes, por exemplo con fontes de alimentación dobres ou cunha disposición de discos en RAID, para ter dous discos con idénticos datos e desta forma, non estar desprotexidos ante unha rotura dun

deles. Asemade, sóense planificar copias de seguridade con bastante frecuencia para ter una salvaguarda da información.

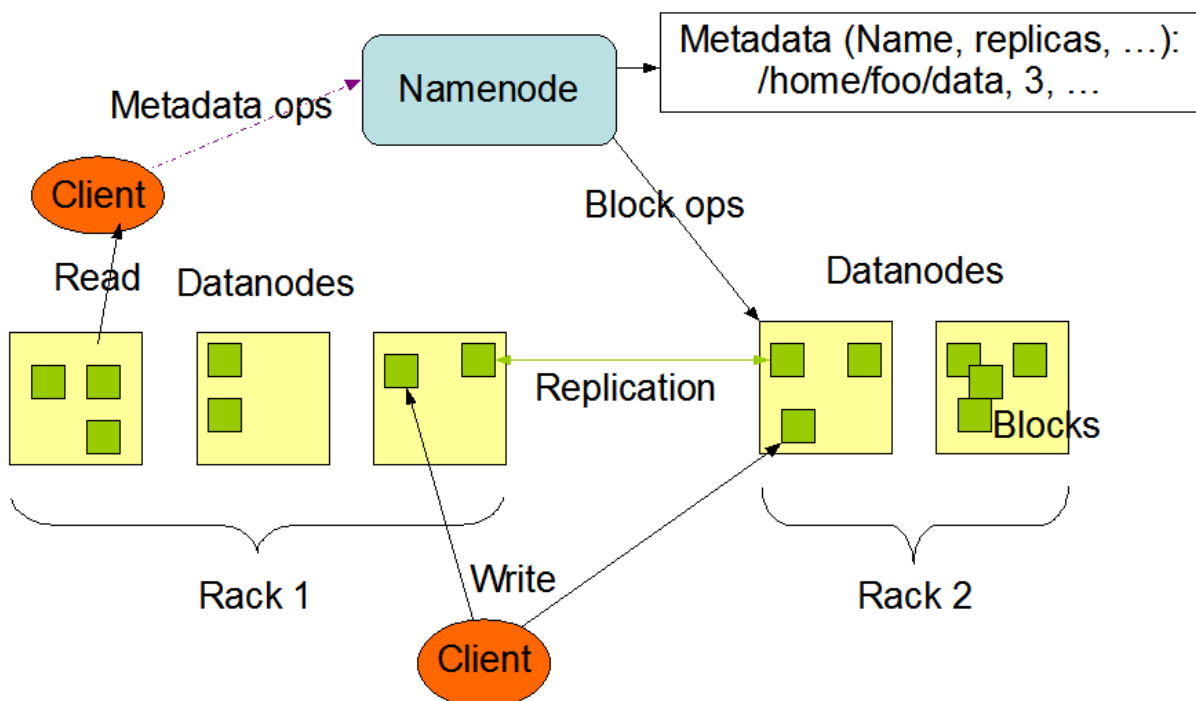
- **DataNode:**

- Demos escravos. Envía sinais a NameNode.
- Almacena os bloques (chunks) que compoñen cada arquivo, e a suma de verificación dos datos do bloque no sistema de arquivos local.
- Serve solicitudes de lectura e escritura dos clientes.

- **SecondaryNameNode:**

- Para mellora-a tolerancia a fallos, adoita existir este nodo secundario do mestre, que NON é un NameNode de respaldo, senón que monitoriza o estado de HDFS e obtén instantáneas de metadatos a intervalos regulares.
- É un servizo separado coa función principal de tomar puntos de control dos metadatos do sistema de ficheiros do Namenode.
 - Para elo, conserva unha copia dos rexistros de logs do Namenode (os de edición (edits) e os da imaxe do sistema de arquivos (fsimage)) e combínaos para mante-lo tamaño razoable do arquivo.
- De hardware, sóese utilizar servidores coas mesmas características que o Namenode.

HDFS Architecture



Arquitectura HDFS. Fonte: <https://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-hdfs/HdfsDesign.html>

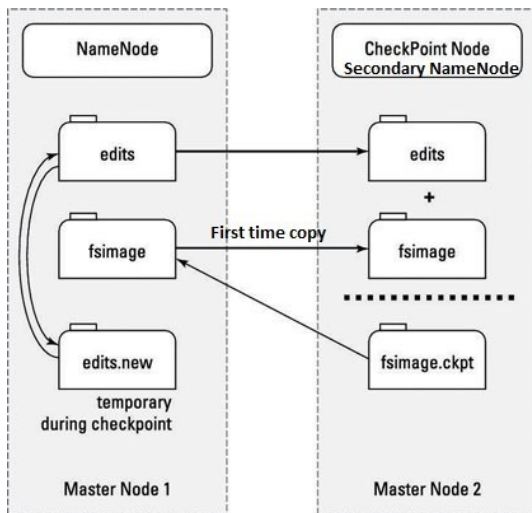
Sistemas de xestión do almacenamento

Sistemas de xestión de almacenamento en big data

Sistemas de arquivos distribuídos: HDFS (Hadoop File System)

Esquema de Sistema de ficheros distribuído Hadoop ou HDFS (Hadoop Distributed File System)

- Os **metadatos** do NameNode son manexados por 2 ficheiros: ***fsimage*** e os logs ***edits***.
 - ***fsimage***: este ficheiro é unha instantánea dos metadatos do sistema de arquivos HDFS que se realiza cada certo tempo: contén tódolos cambios realizados no HDFS cando arranxa o NameNode.
 - Almacénase no disco local da máquina NameNode.
 - Logs ***edits***: este ficheiro contén as modificacións máis recentes; é un dun tamaño pequeno, en comparación co ***fsimage***. É un rexistro dos cambios que se producen nos metadatos do sistema de arquivos.
 - Bórrase cada vez que hai unha nova instantánea en ***fsimage***.
 - Almacénase na memoria RAM da máquina NameNode.
- O Secondary NameNode realiza a función de punto de control (***checkpointing***).
 - O ***checkpointing*** é o proceso de combina-los logs ***edits*** co ***fsimage*** (logs edit + fsimage).
 - O Secondary NameNode crea copia dos log ***Edits*** e do ***fsimage*** do NameNode para crealo ficheiro final ***fsimage***.
 - O proceso de ***checkpointing*** realízase periodicamente (por defecto cada hora).
- O ***fsimage*** final no Secondary NameNode permite unha conmutación por fallo máis rápida xa que evita que os rexistros de edición no NameNode se fagan demasiado grandes.
 - O novo ficheiro de rexistro de edición no NameNode contén tódalas modificacións/cambios que se producen durante o ***checkpointing***.



```

tnp
dfs
  name
    current
      edits_00000000000000000001-00000000000000000002
      edits_00000000000000000003-00000000000000000003
      edits_00000000000000000004-00000000000000000004
      edits_00000000000000000005-00000000000000000027
      edits_00000000000000000028-00000000000000000028
      edits_00000000000000000029-00000000000000000030
      edits_00000000000000000031-00000000000000000095
      edits_00000000000000000096-00000000000000000096
      edits_00000000000000000097-00000000000000000098
      edits_00000000000000000099-00000000000000000110
      edits_00000000000000000111-00000000000000000167
      edits_00000000000000000168-00000000000000000169
      edits_00000000000000000170-00000000000000000211
      edits_00000000000000000212-00000000000000000292
      edits_00000000000000000293-00000000000000000409
      edits_00000000000000000410-00000000000000000417
      edits_00000000000000000418-00000000000000000432
      edits_inprogress_00000000000000000433
      fsimage_00000000000000000417
      fsimage_00000000000000000417.md5
      fsimage_00000000000000000432
      fsimage_00000000000000000432.md5
      seen_txld
      VERSION
    in_use.lock
  name.secondary
    current
      edits_00000000000000000001-00000000000000000002
      edits_00000000000000000005-00000000000000000027
      edits_00000000000000000028-00000000000000000028
      edits_00000000000000000029-00000000000000000030

```

Fonte: <https://mobilemonitoringsolutions.com/hadoop-for-beginners/>

Sistemas de xestión do almacenamento

Sistemas de xestión de almacenamento en big data

Sistemas de arquivos distribuídos: HDFS (Hadoop File System)

Arquitectura Hadoop Cluster

Documentación oficial de Apache: <https://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-hdfs/HdfsDesign.html>

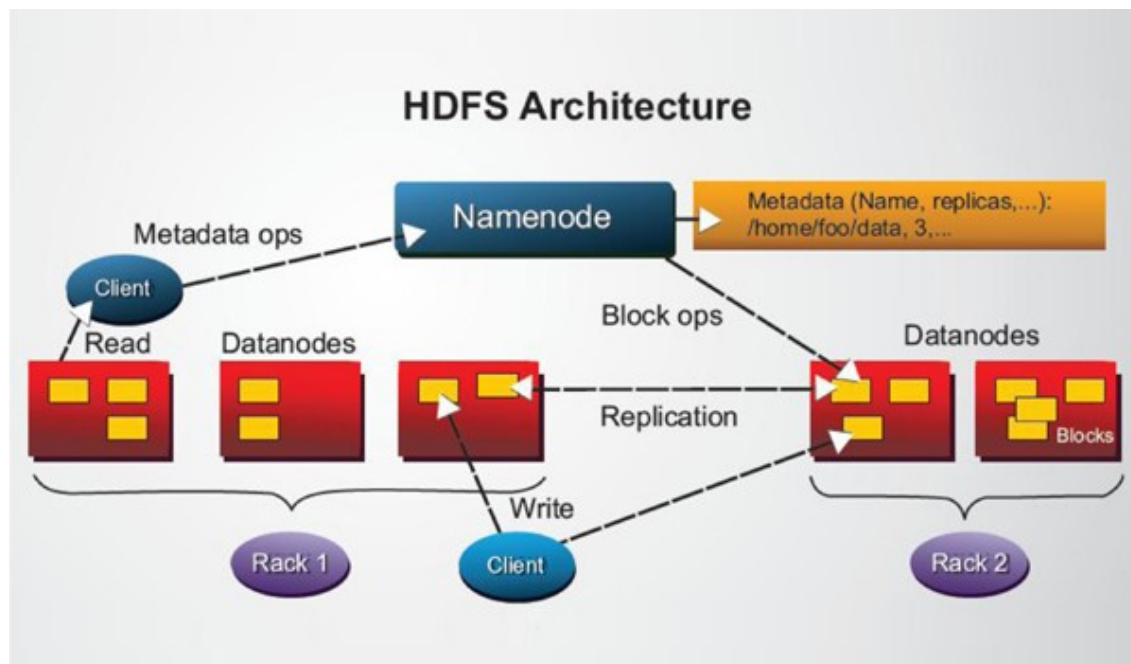
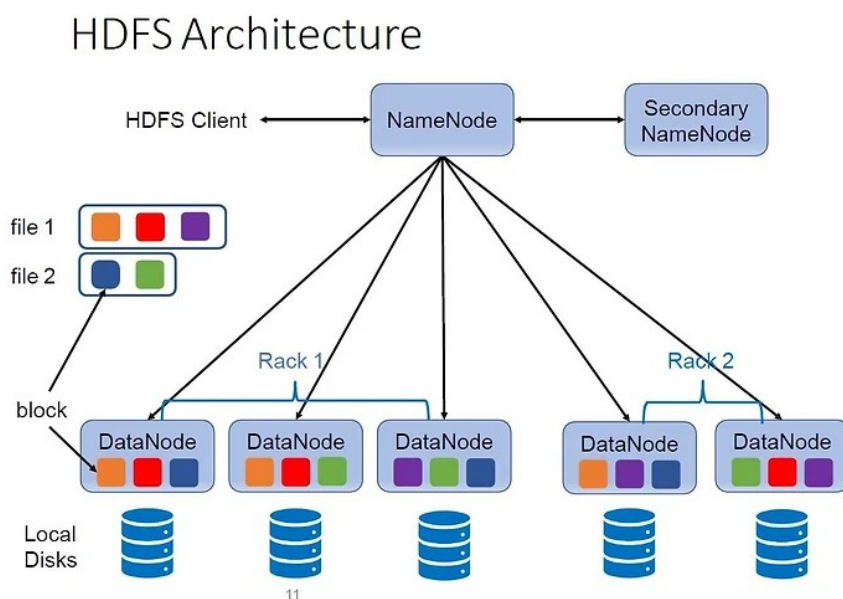


Diagrama de arquitectura HDFS. Fonte: <https://programmerclick.com/article/3665106614/>



Fonte: <https://www.codersarts.com/post/big-data-hdfs-spark-project-help>

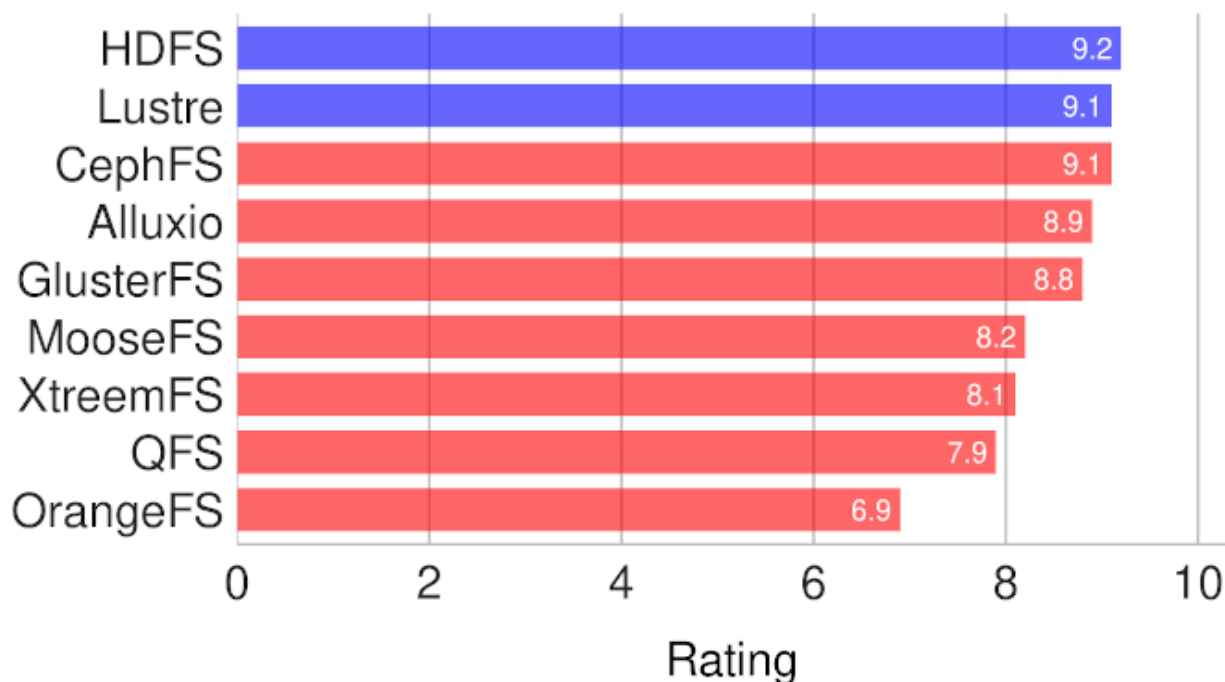
Sistemas de xestión do almacenamento

Sistemas de xestión de almacenamento en big data

Principais sistemas de ficheiros utilizados en Big Data

Best Free File Systems for Big Data

■ Recommended ■ Good

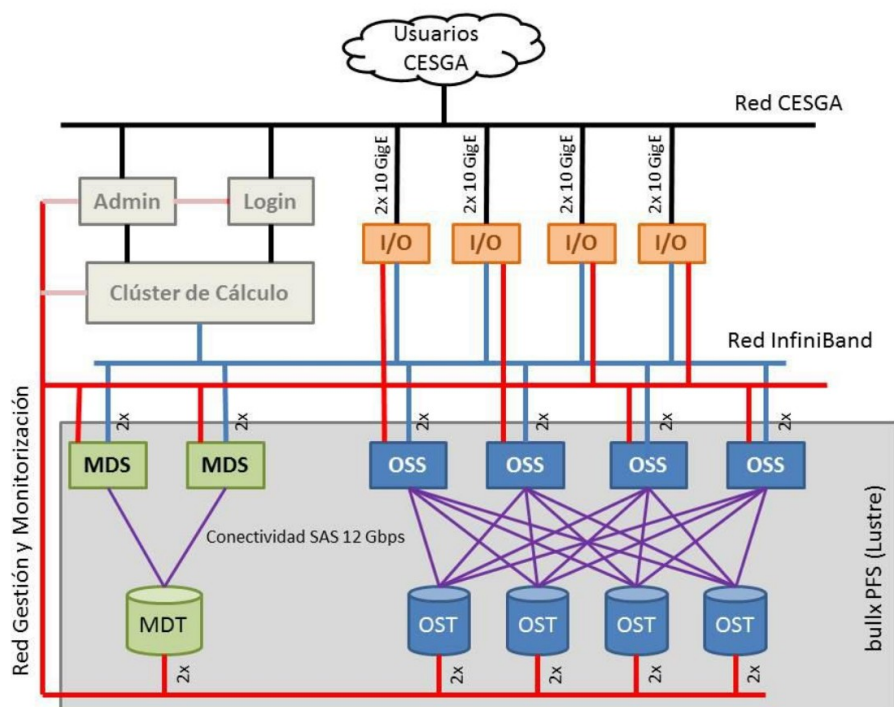
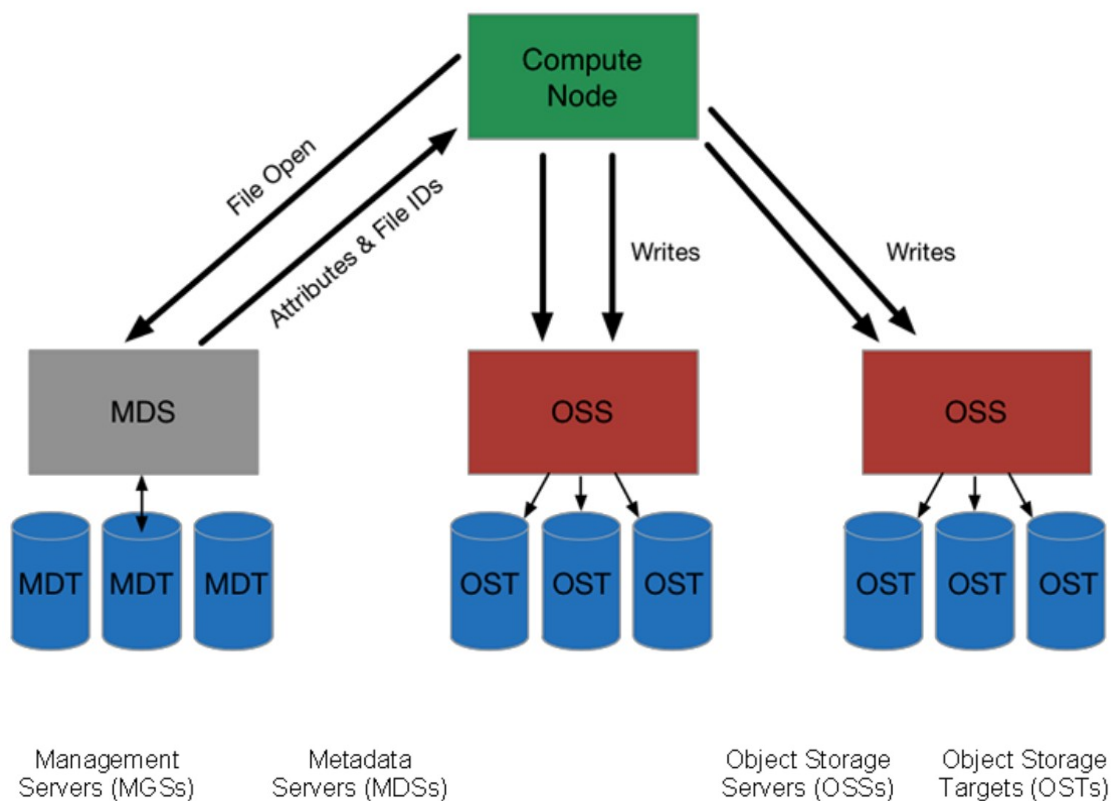


**Nota: Alluxio é un sistema de ficheiros en memoria (Oracle)*

File Systems

<u>HDFS</u>	Distributed file system providing high-throughput access
<u>Lustre</u>	File system for computer clusters
<u>CephFS</u>	Unified, distributed storage system
<u>Alluxio</u>	Virtual distributed file system
<u>GlusterFS</u>	Scale-out NAS file system
<u>MooseFS</u>	POSIX-compliant distributed file system
<u>XtreemFS</u>	Object-based, distributed file system for wide area networks
<u>Quantcast File System</u>	High-performance, fault-tolerant, distributed file system
<u>OrangeFS</u>	Multi-server scalable parallel file system

Fonte: <https://www.linuxlinks.com/filesystems/>



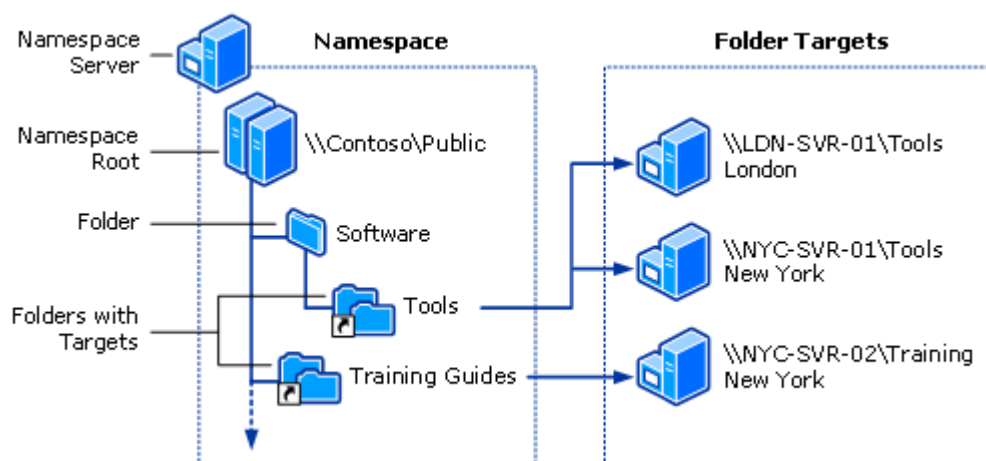
Esquema do sistema Lustre, do CESGA. Fonte: Curso CAFI – F2201002 - Ano 2022 – edu.xunta.gal. Almacenamento e procesamento de datos en contornas Hadoop e Spark. Impartido por: Javier Cacheiro López

Sistemas de xestión do almacenamento

Sistemas de xestión de almacenamento en big data

Sistema de Ficheiros Distribuído ou DFS (Distributed File System)

- É un sistema de ficheiros que se atopa **distribuído entre múltiples servidores**
 - permitindo gardar información **coma se dun disco local se tratase**
 - e permitindo o **acceso ós datos dende calquera punto da rede**,
 - compartindo a información máis facilmente.
 - Non se mapean os recursos físicos, senón que se representan como lóxicos (úsase un mesmo nome de recurso dende calquera punto da rede, en vez de unha ubicación física).
 - *Podemos pensar neles como sistemas tipo SAMBA, NFS,... pero mellorados...*



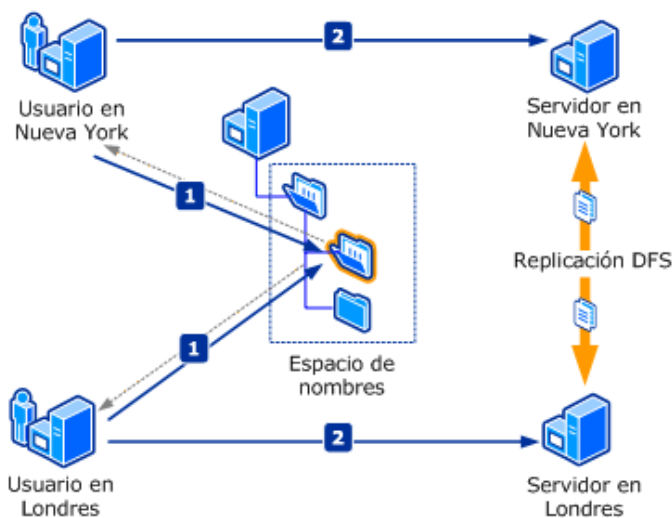
Fonte: <https://docs.microsoft.com/es-es/windows-server/storage/dfs-namespaces/dfs-overview>

Sistemas de xestión do almacenamento

Sistemas de xestión de almacenamento en big data

Sistema de Ficheiros Distribuído ou DFS (Distributed File System)

- Un Sistema de Ficheiros Distribuído incorpora dúas propiedades principais:
 - **Transparencia na localización** (o mesmo recurso vese co mesmo nome e cos mesmos permisos dende calquera punto da rede).
 - Conséguese mediante un sistema de nome de recursos de rede ou “*namespace*”.
 - **Redundancia** (os ficheiros están duplicados en varios nodos para permitir accesos máis rápidos á copia máis próxima, e para garantir que non se perda nada).
 - Conséguese mediante a **duplicidade de datos**.
- En caso de fallo ou alta carga de traballo ...
 - ... a arquitectura distribuída mellora a **dispoñibilidade da información** ...
 - ... compartindo información que se atopa en **diversos emplacements** ...
 - ... que se amosa como unha soa entidade, coñecida como “*DFS raíz*”.



Descripción del proceso

- 1** Si los usuarios tienen acceso a una carpeta con destinos en el espacio de nombres, los equipos cliente contactan con un servidor de espacio de nombres y reciben una referencia.
- 2** Los equipos cliente tienen acceso al primer servidor de las referencias respectivas.

Fonte: <https://gestion-redesdedatos.blogspot.com/2012/10/servicio-de-archivos-en-windows-dfs.html>

Sistemas de xestión do almacenamento

Sistemas de xestión de almacenamento en big data

Sistema de Ficheiros Distribuído ou DFS (Distributed File System)

Vantaxes dun Sistema de Ficheiros Distribuído ou DFS

- Transparencia
 - estrutural
 - de acceso
 - na replicación de datos
 - no nome dos recursos
- Mobilidade do recurso
- Rendemento
- Simplicidade no uso e acceso
- Alta dispoñibilidade
- Alta escalabilidade
- Alta fiabilidade
- Integridade da información
- Heteroxeneidade
- Seguridade

Sistemas de xestión do almacenamento

Sistemas de xestión de almacenamento en big data

Sistema de Ficheiros Distribuído ou DFS (Distributed File System)

Vantaxes dun Sistema de Ficheiros Distribuído ou DFS

Transparencia.

- Transparencia estrutural
 - O/A usuario/a non necesita coñecer cantos servidores, emprazamentos, etc., existen.
 - Iso permite mellora-la tolerancia a fallos.
- Transparencia de acceso.
 - Non existe diferenza entre acceder a ficheiros locais e remotos.
 - O sistema de ficheiros xestiona os accesos.
- Transparencia na replicación de datos.
 - Se un ficheiro se copia en múltiples nodos o nome real do mesmo permanece oculto ó resto de nodos, para evitar duplicidade de nomes.
 - Por exemplo, en Google Docs, ou AWS, o sistema xestiona o nome cun identificador único dentro da plataforma, para evitar que 2 ficheiros diferentes subidos por persoas distintas non poidan te-lo mesmo nome orixinal definido por cada persoa.
- Transparencia no nome dos recursos.
 - O nome do recurso determina o acceso á información.
 - O nome dos ficheiros hase de manter durante a transmisión entre os nodos.

Sistemas de xestión do almacenamento

Sistemas de xestión de almacenamento en big data

Sistema de Ficheiros Distribuído ou DFS (Distributed File System)

Vantaxes dun Sistema de Ficheiros Distribuído ou DFS

Mobilidade do recurso.

- A información estará dispoñible alí onde o cliente se atope ...
 - ... independentemente da súa localización.

Rendemento.

- O rendemento mídese en función do acceso ó recurso.
 - Segundo o tipo de infraestrutura hardware subxacente ...
 - ... o acceso pode ser incluso máis rápido que a un recurso local.

Simplicidade no uso e acceso.

- O acceso é tan sinxelo como acceder polo nome do recurso.
 - A complexidade de xestión e transferencia de datos ...
 - ... é asumida polo sistema de ficheiros

Alta dispoñibilidade.

- O sistema de ficheiros distribuído ...
 - ... é capaz de funcionar aínda con caídas de parte dos nodos.
 - Isto implica a replicación non só dos datos ...
 - ... se non tamén, da seguridade destes.

Alta escalabilidade.

- O DFS pode crecer en número de nodos de forma case infinita.
 - Ademais implica o poder engadir ou retirar nodos á infraestrutura ...
 - ... de forma dinámica (*non-stop*).

Sistemas de xestión do almacenamento

Sistemas de xestión de almacenamento en big data

Sistema de Ficheiros Distribuído ou DFS (Distributed File System)

Vantaxes dun Sistema de Ficheiros Distribuído ou DFS

Alta fiabilidade.

- Isto fai que a probabilidade de perda de información ...
 - ... sexa mínima nun DFS.
 - Isto conséguese con replica de información entre os nodos do sistema.

Integridade da información.

- A información depositada nun DFS é compartida, ...
 - ... ten que preserva-la súa integridade mentres dure unha transacción.
 - O sistema de ficheiros garante que un mesmo dato non está a ser modificado por varios nodos de forma simultánea.

Heteroxeneidade.

- Un sistema DFS permite ser construído ...
 - ... entre nodos con arquitecturas hardware e software
 - ... diferentes (samba, NFS,...), ...
 - ... sendo transparente para o cliente.

Seguridade.

- Un sistema DFS permite controla-lo acceso á información ...
 - ... de forma granular (a nivel de ficheiro non só de directorios ou estruturas).
 - Isto implica a incorporación dun sistema de ...
 - ...seguridade ou permisos ó menor nivel do dato.

Exercicio LAB 2.1 – HADOOP e MAPREDUCE

Obxectivos

- Instalación e configuración de Hadoop nun único nodo (modo pseudodistribuído).
- Configuración e uso do sistema de arquivos HDFS.
- Executar un programa de exemplo seguindo o paradigma MapReduce sobre o sistema HDFS.
 - HDFS é o sistema de arquivos distribuído de Hadoop; está deseñado para usar múltiples computadoras xenéricas (commodity hardware) agrupadas en clústeres, é tolerante a fallos e proporciona un alto rendemento no acceso ós datos.

Programación con **Hadoop**:

- **HDFS** ocúpase de partir e replica-los datos (proporciona almacenamento distribuído).