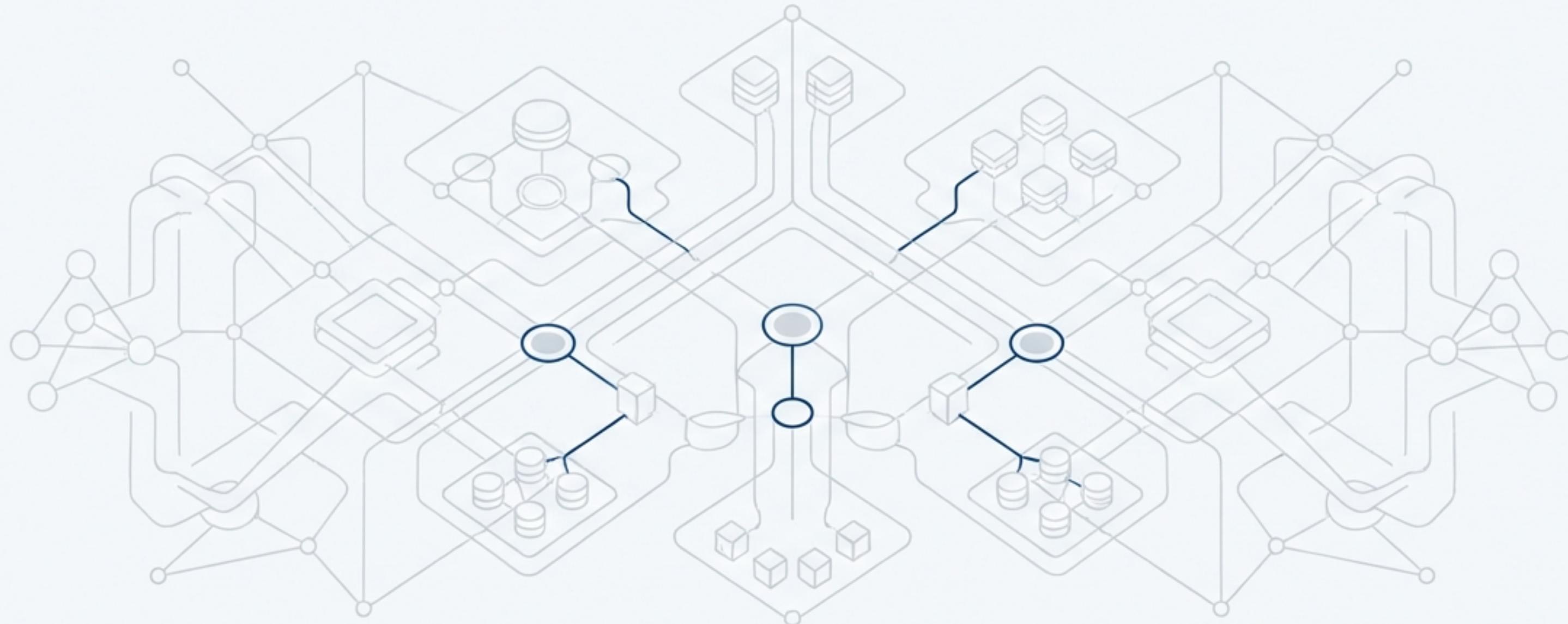


Dominando Big Data: Una Introducción Práctica a Hadoop

De los conceptos fundamentales a tu primer job MapReduce.



El Desafío: ¿Qué es Big Data?

Se define como un conjunto de grandes volúmenes de datos (terabytes, petabytes o más) cuyo objetivo es extraer valor a partir de información dispersa y compleja.



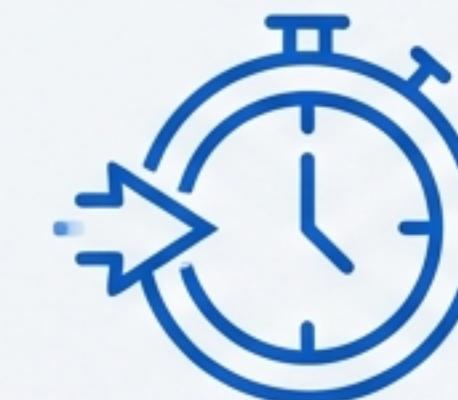
Volumen

Datos masivos que superan la capacidad de las herramientas tradicionales.



Variedad

Datos estructurados (BBDD), semiestructurados (logs) y no estructurados (imágenes, tweets).



Velocidad

La necesidad de procesar los datos de forma continua y casi en tiempo real.

La Solución: ¿Qué es Hadoop?

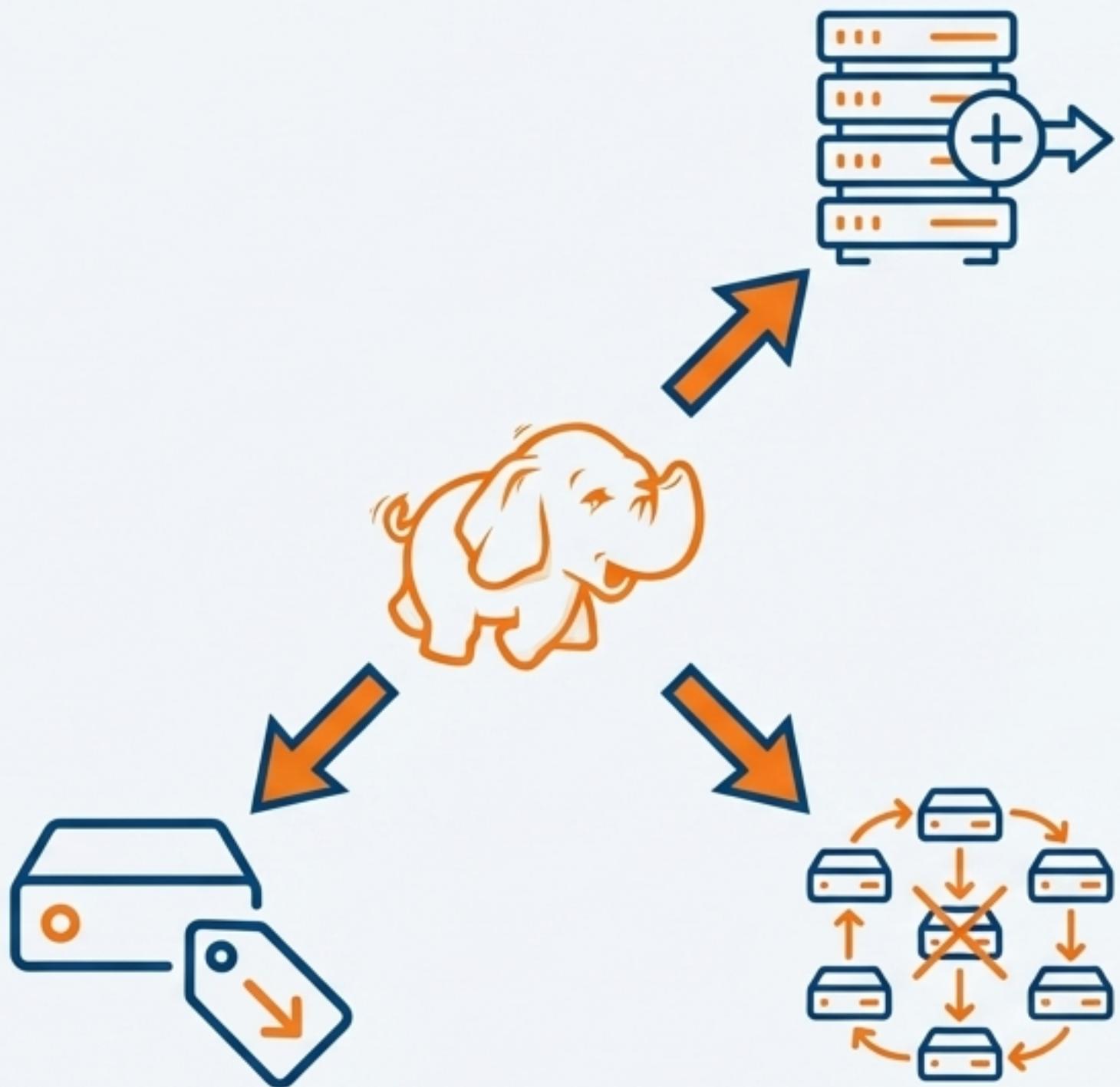
Un framework open source de Apache diseñado específicamente para almacenar y procesar Big Data de forma distribuida y paralela.

¿Cómo lo hace?

- Almacenamiento: Guarda los datos de forma distribuida en múltiples máquinas (HDFS).
- Procesamiento: Procesa los datos en paralelo aprovechando todos los nodos del clúster (MapReduce, YARN).

Ventajas Clave

- Escalabilidad horizontal: Se pueden añadir más máquinas para aumentar la capacidad.
- Tolerancia a fallos: La replicación de datos evita pérdidas si un nodo falla.
- Bajo coste: Utiliza hardware común y accesible.



El Núcleo de Hadoop: Un Resumen

HDFS

Dónde se guardan los datos.

Sistema de archivos que divide los datos en bloques, los reparte y replica entre los nodos del clúster.

"Como una biblioteca digital repartida en muchos ordenadores."

YARN

Quién gestiona los recursos.

Asigna CPU, memoria y red a los trabajos, permitiendo que múltiples aplicaciones compartan el clúster de forma ordenada.

"El gestor de tráfico del clúster, que decide qué recursos usa cada proceso."

MapReduce

Cómo se procesan los datos.

Modelo de programación que divide las tareas en dos fases: *Map* (procesar fragmentos) y *Reduce* (combinar resultados).

"Para contar palabras en millones de documentos, Map cuenta en cada nodo y Reduce suma los totales."

Más Allá del Núcleo: El Ecosistema Hadoop



Un conjunto de herramientas que complementan y potencian el núcleo de Hadoop para distintos casos de uso.

¿Cómo se Usa en el Mundo Real? Distribuciones y Cloud

¿Qué es una distribución?

Una implementación “llave en mano” de Hadoop y su ecosistema, lista para usar.

Ahorra tiempo en instalación y configuración manual.



Ejemplos Principales



- **On-Premise:** Cloudera (que integra la antigua Hortonworks) es el líder del mercado.

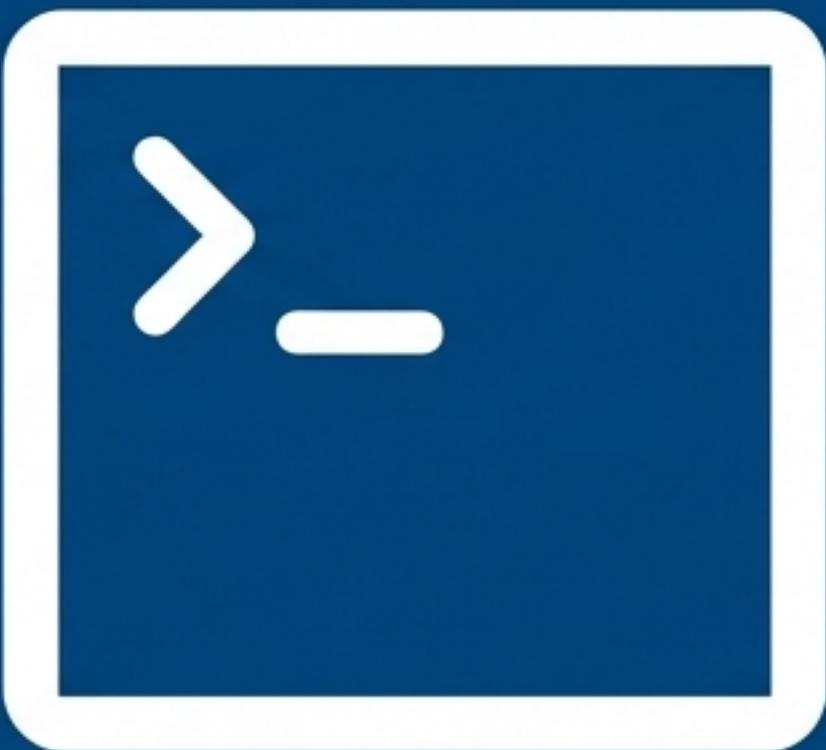


- **En la Nube:** Servicios gestionados como Amazon EMR, Microsoft Azure HDInsight y Google Cloud ofrecen Hadoop bajo demanda.



- **Para Aprender y Desarrollar:** La instalación manual (como en nuestro laboratorio) o el uso de contenedores (**Docker**) son alternativas excelentes.

Pongámoslo a Prueba: Tu Primer Job MapReduce



Ahora que entendemos el 'qué' y el 'porqué', vamos a ver el 'cómo' en acción.

Laboratorio: Objetivo y Entorno



Objetivo de la Práctica

Meta: Comprobar que Hadoop puede ejecutar un job MapReduce de ejemplo (`grep`) y que produce la salida esperada.

Pruebas:

- Realizaremos 2 ejecuciones con expresiones distintas:
 1. Buscar el texto `dfsadmin`.
 2. Contar todas las ocurrencias de la letra `'a'`.



Entorno de Trabajo

- Directorio Hadoop:
`/home/manuu/Desktop/hadoop`
- Jar de Ejemplos:
`/home/manuu/Desktop/hadoop/share/hadoop/mapreduce/hadoop-mapreduce-examples-3.3.6.jar`
- Directarios de Datos (locales):
 - Entrada: `/tmp/entrada`
(debe contener archivos XML)
 - Salida: `/tmp/salida`

Paso 1: Preparación del Entorno

Acción 1: Navegar al directorio de trabajo

Para simplificar la ruta del comando `jar`, nos situamos en la carpeta que lo contiene.

```
cd /home/manuu/Desktop/hadoop/share/hadoop/mapreduce/
```

Acción 2: Asegurar los datos de entrada

El directorio `/tmp/entrada` debe existir y contener los archivos XML que se usarán como fuente de datos.

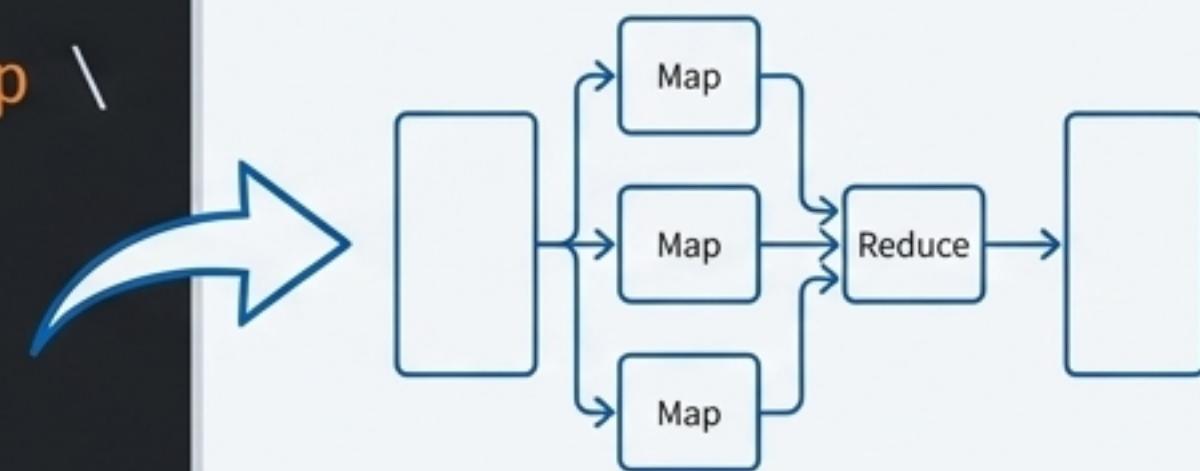
Criterios de Evaluación Iniciales

- ✓ El archivo `hadoop-mapreduce-examples-3.3.6.jar` existe en el directorio actual.
- ✓ El directorio `/tmp/entrada` contiene archivos XML.

Paso 2: Ejecución #1 - Búsqueda de "dfs"

Ejecutar el job `grep` de ejemplo para encontrar todas las líneas que contengan el patrón `dfs` seguido de letras minúsculas.

```
hadoop jar hadoop-mapreduce-examples-3.3.6.jar grep \
  /tmp/entrada \
  /tmp/salida \
  'dfs[a-z.]+'
```



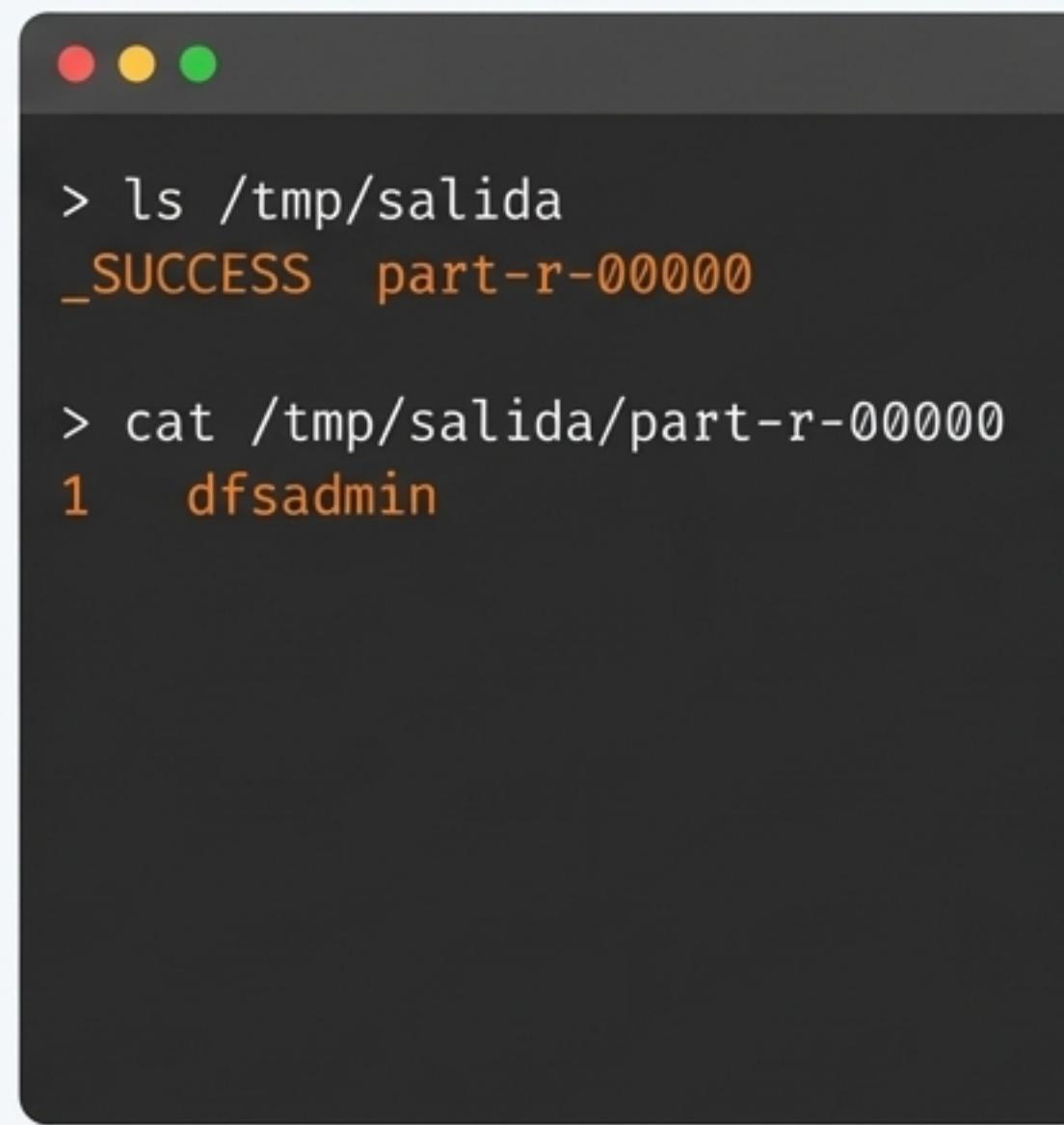
Observa los logs en la terminal. Verás el progreso de las fases Map y Reduce. El job finaliza con un mensaje de 'SUCCEEDED'.

Paso 3: Verificación #1 - El Resultado

Comprobar que el job se ejecutó correctamente y generó la salida esperada.

```
# Listar el contenido del
# directorio de salida
ls /tmp/salida

# Ver el contenido del
# fichero de resultados
cat /tmp/salida/part-r-00000
```

A screenshot of a macOS-style terminal window. The title bar has three colored dots (red, yellow, green). The window contains the following text:

```
> ls /tmp/salida
_SUCCESS  part-r-00000

> cat /tmp/salida/part-r-00000
1  dfsadmin
```

¿Qué buscar para dar el OK?

- El job finalizó con estado `SUCCEEDED`.
- El fichero `part-r-00000` existe en `/tmp/salida`.
- El contenido del fichero muestra `1 dfsadmin`, indicando una coincidencia.

Paso 4: Ejecución #2 - Conteo de la letra 'a'

Acción 1: Borrar la salida anterior

Hadoop no sobrescribe un directorio de salida existente.
Es necesario borrarlo antes de una nueva ejecución.

```
rm -rf /tmp/salida
```



Acción 2: Ejecutar el segundo job

Ahora, contamos todas las ocurrencias de la letra `a` (case-sensitive) en los mismos archivos de entrada.

```
hadoop jar hadoop-mapreduce-examples-3.3.6.jar grep \  
/tmp/entrada \  
/tmp/salida \  
'a'
```

Paso 5: Verificación #2 e Interpretación

Verificación

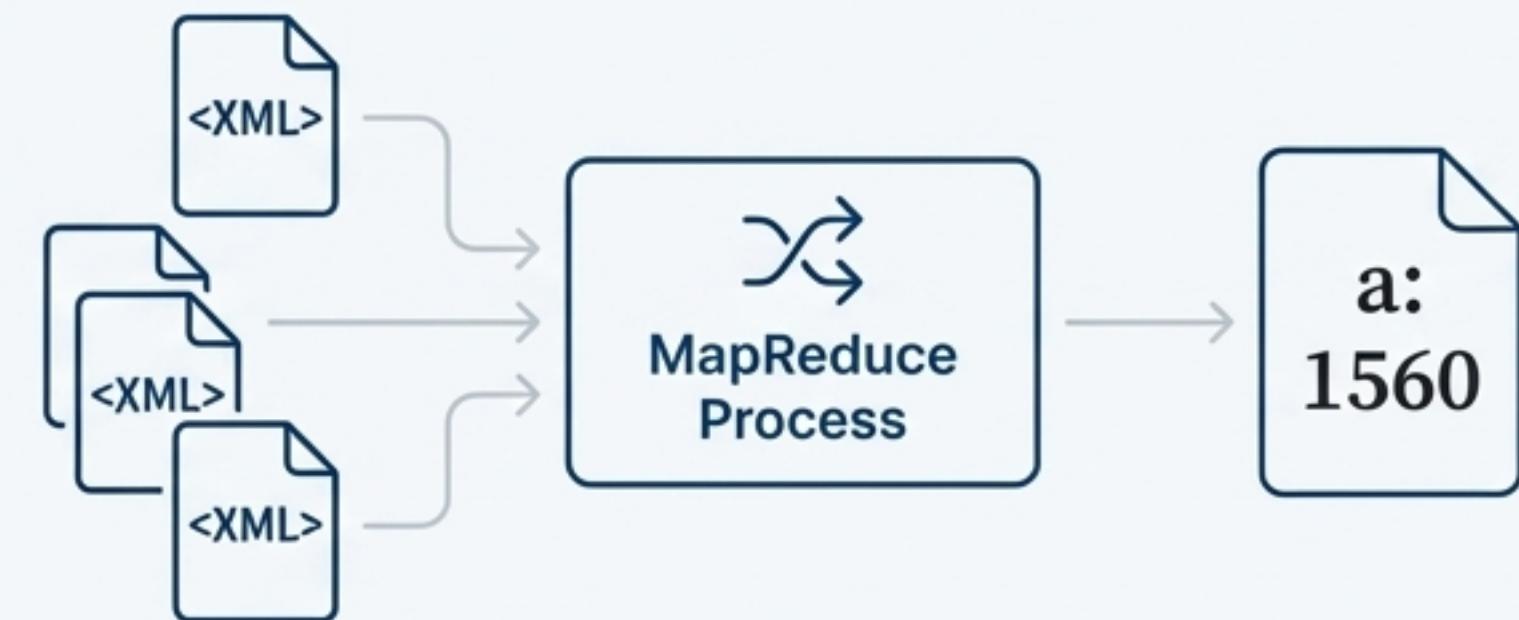
Acción: Comprobar el resultado del segundo job.

```
cat /tmp/salida/part-r-00000
```

```
1560 a
```

Interpretación del Resultado

- **Clave:** a es el patrón que buscamos.
- **Valor:** 1560 es el número total de veces que la letra "a" aparece en todos los ficheros XML de entrada.
- **Conclusión:** El job ha procesado con éxito todos los ficheros y agregado los resultados.



Notas y Puntos Clave del Laboratorio



El Directorio de Salida

El job falla si el directorio de salida (`/tmp/salida`) ya existe. Siempre bórralo antes de ejecutar (`rm -rf /tmp/salida`).

Aa

Sensibilidad a Mayúsculas

La expresión '`dfs[a-z.]+`' es *case-sensitive*, por eso encontrò `dfsadmin` en minúsculas.



Rutas de Comandos

Ejecutamos el comando desde `.../mapreduce/` para poder usar el nombre del `jar` directamente, sin la ruta completa.



Modo Local vs. HDFS

Esta práctica usa el sistema de archivos local. En un clúster real, los comandos serían análogos pero usando `hdfs dfs -cat` y `hdfs dfs -put`.

Resumen y Próximos Pasos

¡Lo que has logrado!

- ✓ Has comprendido los desafíos del Big Data y la arquitectura fundamental de Hadoop (HDFS, YARN, MapReduce).
- ✓ Has configurado un entorno y ejecutado con éxito dos jobs MapReduce reales.
- ✓ Has verificado e interpretado los resultados, validando el procesamiento distribuido en acción.



Tu Primer Job

El Camino a Seguir

Este primer job es la puerta de entrada al ecosistema Hadoop. Los próximos pasos lógicos incluyen explorar herramientas de más alto nivel que se ejecutan sobre esta base, como:

- **Hive**: para análisis con una sintaxis familiar tipo SQL.
- **Spark**: para un procesamiento en memoria aún más rápido.

