

Preguntas teoria

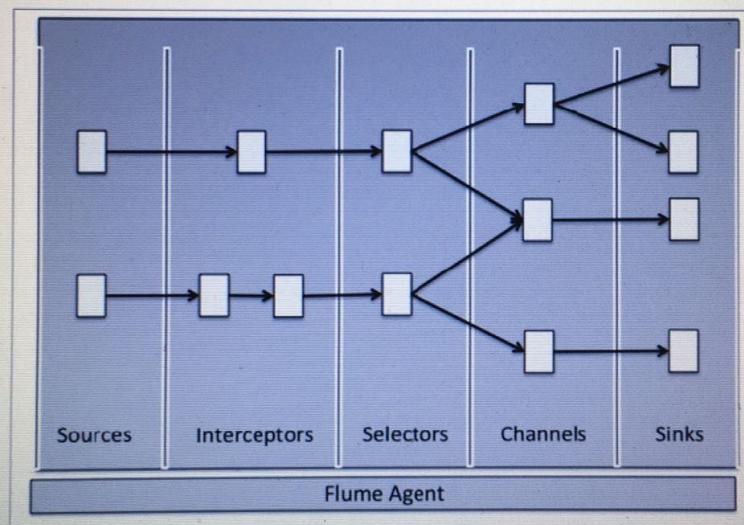
\*Obligatorio

Un agente de Flume puede tener... \*

- Un sólo source, un sólo channel y un sólo sink
- Un sólo source, Un sólo channel y uno ó más sinks
- Un sólo source, uno ó más channels y uno ó más sinks
- Uno ó más sources, uno ó más channels y uno ó más sinks

Porque?

- Un **agente Flume** es un contenedor para todos los componentes descritos. Es un proceso JVM con un conjunto de sources, sinks, channels... etc.



Esta no encontré una respuesta muy clara en los apuntes.

Hacer copias periódicas de ficheros desde el sistema de ficheros local a HDFS usando comandos como "hdfs put", es una estrategia de adquisición de datos adecuada cuando... \*

- Necesitamos realizar modificaciones al vuelo de los datos
- Necesitamos realizar analítica en tiempo real de los datos
- Podemos permitirnos una latencia media/alta, y vamos a hacer procesado "batch" de los datos, probablemente con más de una tecnología (MapReduce, Pig, Hive, Spark... etc)
- Ninguna de las anteriores

siguiente

Utilizar Flume para nuestra adquisición de datos nos asegura... \*

- Que los eventos/datos generados en origen se entregan exactamente una vez en destino
- Que los eventos/datos generados en origen se entregan al menos una vez en destino
- Que nunca vamos a tener duplicados en el destino
- No nos asegura nada

## IFF

---

## Transacciones

- Flume se asegura de que todo evento producido en la fuente alcanza el sink **al menos una vez**
- Es posible que se produzcan eventos duplicados en caso de fallos o reinicios de agentes
- La semántica de eventos “al menos una vez” es una decisión de diseño de Flume. Al ser más eficiente de implementar que “exactamente una vez” le permite ser una herramienta capaz de manejar volúmenes muy grandes de datos
- El **tratamiento de eventos duplicados** se suele hacer a posteriori, por ejemplo en jobs Mapreduce ó Hive.

Esta tampoco encontré respuesta clara

HDFS es un buen destino de datos en una arquitectura de adquisición de datos porque... \*

- Permite realizar modificaciones (UPDATE) en los ficheros del dataset
- Es un File-System distribuído, altamente escalable, y que puede ser utilizado por múltiples tecnologías big data
- Está optimizado para almacenar un gran número de ficheros pequeños
- Permite realizar queries complejas sobre los datos

○ Cómo importar datos desde una Base de

## Cuando usamos Sqoop para importar datos desde una Base de Datos relacional a HDFS... \*

- Sólo podemos importar bases de datos completas
- Sólo podemos importar tablas completas
- Podemos seleccionar qué tabla/tablas queremos importar, y podemos también seleccionar qué datos de esas tablas importar
- Ninguna de las anteriores

- Un caso común es querer mantener los datos en Hadoop sincronizados con la BBDD haciendo **imports incrementales**
- Si es una **tabla pequeña, no merece la pena**. Simplemente importamos cada vez la tabla completa
- Si es una **tabla grande**, que tarda bastante en importarse, utilizaremos **imports incrementales**
  - Se basan en verificar el valor de una columna determinada (`--check-column`) para identificar cambios.
  - Sólo se importarán los registros con un valor mayor de un valor especificado en dicha columna (`--last-value`)

Desde el punto de vista de Apache Kafka, un "topic" es... \*

- Un proceso que produce mensajes
- Un proceso que consume mensajes
- Es un log replicado de eventos, que puede estar particionado y distribuido en el clúster de Kafka
- Es un proceso que asegura la consistencia de los datos

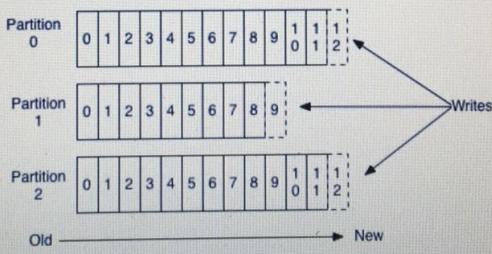
¿Cuál es la respuesta correcta? ¿Por qué no se creó un tópico con 2 particiones?

CIFF

Tópicos

- Cada **tópico** en kafka es un “**log particionado**”
- Cada partición es una **secuencia ordenada de mensajes** a la que continuamente se añaden nuevos mensajes
- Cada mensaje en una partición tiene asignado un ID secuencial llamado **offset** que identifica únicamente cada mensaje en la partición

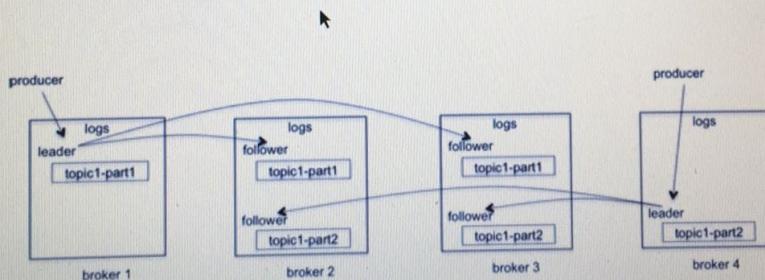
#### Anatomy of a Topic



Si tienes un cluster Kafka de 4 brokers, y tienes un tópico con 2 particiones y factor de replicación 3 (en total hay 3 copias de cada partición, incluyendo la del broker "leader"). ¿Cuántos brokers como máximo podrían caer sin que hubiera pérdida de datos? \*

- 1
- 2
- 3
- No podría caer ningún broker. (Es decir, si cae uno sólo, ya perderíamos datos)

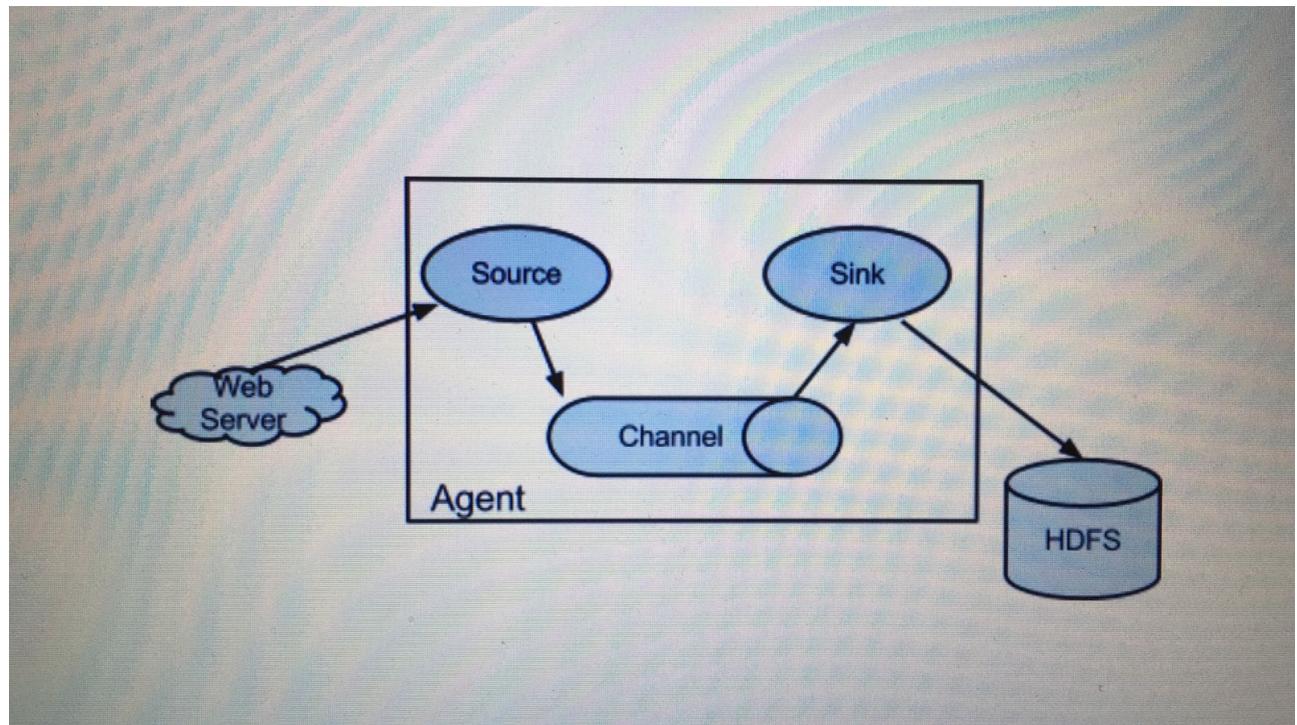
- Un topic con factor de replicación N, tolerará **N-1 brokers caídos sin perder ningún mensaje**
- Para cada partición hay **un broker que actúa como líder**, y cero ó más brokers que actúan como "followers".
- El broker líder atiende **todas las peticiones de lectura y escritura** en la partición



Como tenemos factor de replicación 3,  $N-1 = 2$

Las 3 partes principales de la arquitectura de Flume son: \*

- Routers (encaminadores), Processors (procesadores) y Almacenes (warehouses)
- Sources (fuentes), Channels (canales) y Sinks (sumideros)
- Producers (productores), Consumers (consumidores) y Topics (topics)
- Ninguna de las anteriores



Cuando Sqoop lanza un Job MapReduce para importar datos desde una Base de Datos relacional a HDFS... \*

- Sqoop realiza la importación de datos en una sola transacción contra la BBDD
- Sqoop lanza una o varias tareas Map en paralelo, cada una con una conexión a la BBDD independiente.
- Sqoop lanza las distintas tareas Map, una detrás de otra, ejecutando sólo una cuando la anterior ha terminado
- Ninguna de las anteriores

CIFF



Importar Datos

- Por defecto, Sqoop utilizará 4 mappers para hacer la importación, **cada uno con su propia conexión a la BBDD y con su propio fichero de salida en HDFS** (en el mismo directorio)
- Por defecto el formato de los ficheros importados es Text Files con formato CSV pero se puede configurar para utilizar formatos binarios (SequenceFile, Avro, Parquet...), compresión, etc.
- Normalmente la “**splitting column**” por la que se divide la query de importación entre varios mappers es la Primary Key, pero el usuario puede especificar otra
- Sqoop obtendrá el máximo y mínimo valor de la “splitting column” y dividirá el rango entre el número de mappers (también configurable)

Entiendo que las 4 son en paralelo