Como Funciona sin fallos

**Fase 1: Map**

En esta fase, los nodos **Map** de cada **MapReduce** trabajan en paralelo para procesar diferentes partes ("chunks") del archivo de entrada. El proceso es el siguiente:

1. **MapReduce1**:
   * **Nodo 0**: Procesa chunk\_0. Una vez que termina, pasa a procesar el siguiente chunk.
   * **Nodo 1**: Procesa chunk\_1. Al completarlo, toma otro chunk (en este caso, chunk\_5, luego chunk\_9, etc.).
   * **Nodo 2**: Procesa chunk\_2 y sigue con chunk\_6, chunk\_10, y así sucesivamente.
   * **Nodo 3**: Procesa chunk\_3, luego chunk\_7, chunk\_11, etc.
   * Este trabajo ocurre en **paralelo**, con cada nodo trabajando en un chunk a la vez hasta que todos los chunks de **MapReduce1** han sido procesados.
2. **MapReduce2**:
   * Funciona de manera similar a **MapReduce1**, pero con un conjunto diferente de datos (que podrían venir de un archivo diferente o una parte posterior del mismo archivo):
     + **Nodo 0**: Procesa chunk\_20, luego chunk\_24, chunk\_28, y así sucesivamente.
     + **Nodo 1**: Procesa chunk\_21, luego chunk\_25, chunk\_29, y así sucesivamente.
     + **Nodo 2**: Procesa chunk\_22, luego chunk\_26, chunk\_30, y así sucesivamente.
     + **Nodo 3**: Procesa chunk\_23, luego chunk\_27, chunk\_31, y así sucesivamente.
   * Cada nodo trabaja en **paralelo**, ejecutando las tareas de procesamiento de chunks en su propio ritmo.

**¿Qué sucede en cada nodo durante la fase Map?**

Cada nodo **Map** lee su chunk asignado (una parte del archivo), procesa el texto y genera pares **clave-valor** para cada palabra y su frecuencia dentro del chunk. Estos pares se almacenan para ser enviados a la fase **Shuffle**.

**Fase 2: Shuffle**

En la fase **Shuffle**, los resultados generados por los nodos **Map** son agrupados por claves (palabras) y distribuidos a los nodos **Reduce** correspondientes.

1. **MapReduce1**:
   * **Nodo 0**: Procesa el "subset 0" (agrupación de claves), que proviene de varios chunks procesados por los nodos **Map**.
   * **Nodo 1**: Procesa el "subset 1".
   * **Nodo 2**: Procesa el "subset 2".
   * **Nodo 3**: Procesa el "subset 3".
   * Aquí, todos los nodos trabajan en **paralelo**, organizando las palabras para que cada nodo **Reduce** reciba todas las instancias de una clave específica. Por ejemplo, si la palabra "casa" apareció en diferentes chunks, todas las instancias de "casa" se agrupan en un solo nodo.
2. **MapReduce2**:
   * **Nodo 0**: Procesa el "subset 0".
   * **Nodo 1**: Procesa el "subset 1".
   * **Nodo 2**: Procesa el "subset 2".
   * **Nodo 3**: Procesa el "subset 3".
   * Nuevamente, la fase **Shuffle** ocurre en **paralelo**, con cada nodo procesando su parte de las claves. Cada nodo **Reduce** se prepara para tomar todas las ocurrencias de una clave.

**Fase 3: Reduce**

En esta fase, los nodos **Reduce** reciben los conjuntos de pares clave-valor agrupados y los reducen para obtener el resultado final (en este caso, el conteo total de cada palabra).

1. **MapReduce1**:
   * **Nodo 1**: Procesa shuffle\_0, que contiene todas las claves-valor para las palabras en el "subset 0".
   * **Nodo 2**: Procesa shuffle\_2, que contiene el "subset 2".
   * **Nodo 3**: Procesa shuffle\_3.
   * **Nodo 1**: Procesa otro conjunto en shuffle\_1.
   * Cada nodo **Reduce** trabaja en **paralelo** y toma su conjunto de palabras agrupadas, sumando las ocurrencias de cada una. Por ejemplo, si la palabra "casa" aparece en los subsets asignados a varios nodos, el nodo correspondiente sumará todas las ocurrencias.
2. **MapReduce2**:
   * **Nodo 1**: Procesa shuffle\_0.
   * **Nodo 2**: Procesa shuffle\_2.
   * **Nodo 3**: Procesa shuffle\_3.
   * **Nodo 1**: Procesa shuffle\_1.
   * Igual que en **MapReduce1**, todos los nodos **Reduce** trabajan en **paralelo**. Cada nodo suma las ocurrencias de sus claves asignadas.

**Fase 4: Nodo Final Reduce**

Después de que todos los nodos **Reduce** terminan de procesar sus respectivos subsets, los resultados se combinan en un único archivo final.

* **Nodo Final Reduce**: Este nodo toma todos los resultados de los nodos **Reduce** de ambos **MapReduce1** y **MapReduce2**. Se encarga de combinar los archivos generados por cada nodo **Reduce** en un solo archivo de salida.
* La combinación ocurre de manera **secuencial**, ya que es el último paso y no se puede paralelizar fácilmente.

**Conclusión:**

* **Map** y **Shuffle** son fases altamente paralelizables. Los nodos procesan sus respectivos datos simultáneamente sin depender unos de otros.
* La fase **Reduce** también se paraleliza, ya que los nodos pueden procesar diferentes subconjuntos de claves al mismo tiempo.
* El **Nodo Final Reduce** combina los resultados de los nodos **Reduce**, y este paso es secuencial.

El proceso finaliza con éxito cuando el **Nodo Final Reduce** completa la combinación de los resultados, generando el archivo final. En todo momento, la operación en paralelo de los nodos permite que el sistema distribuya eficientemente la carga de trabajo y finalice más rápido.

Como funciona con Fallos

Caso: Fallo en Coordinator

En este caso, el fallo se induce en el **Coordinador**, y es una situación crítica, ya que el Coordinador es el componente encargado de orquestar y gestionar todo el proceso de MapReduce. Su función principal es asignar tareas a los nodos Map, Shuffle, y Reduce, monitorear su progreso y gestionar la recuperación de errores en los nodos.

**Explicación del Proceso:**

**Fase Inicial:**

1. **Coordinador**: Es el encargado de iniciar todo el proceso de MapReduce, asignando los chunks de datos a los nodos Map en ambos MapReducers y coordinando las fases de Map, Shuffle y Reduce. También se encarga de reasignar tareas en caso de fallos en los nodos individuales.

**Inducción de Error en el Coordinador:**

1. **Fallo inducido en el Coordinador**: Cuando induces el fallo en el Coordinador, el sistema no puede continuar porque este es el componente central que coordina todas las fases. El fallo detiene inmediatamente el programa:

java

Copy code

El coordinador ha fallado. Terminando programa...

Dado que el Coordinador es el responsable de controlar todo el flujo de ejecución de los nodos, su fallo significa que no hay supervisión ni reasignación de tareas, lo que implica que todo el proceso de MapReduce se detiene.

**Consecuencias del Fallo:**

* **Sin recuperación**: A diferencia de los nodos individuales de Map, Shuffle o Reduce, donde el Coordinador puede reasignar las tareas en caso de fallos, si el propio Coordinador falla, no hay un sistema de recuperación interno que pueda reiniciarlo o continuar el proceso.
* **Terminación del programa**: Debido a que el Coordinador es esencial para gestionar todas las fases, el programa simplemente termina al detectar el fallo, sin intentar continuar o reiniciar.

**Resumen:**

* **Importancia del Coordinador**: El Coordinador es vital para que todo el proceso de MapReduce funcione. Gestiona la distribución de tareas, controla la ejecución de los nodos, maneja la reasignación en caso de fallos y asegura que todo se ejecute correctamente.
* **Fallo fatal**: Un fallo en el Coordinador detiene todo el programa inmediatamente, ya que no hay ningún mecanismo para continuar sin él.

Caso: Fallo en Nodo Map de MapReduce1

En este caso, el proceso de MapReduce1 y MapReduce2 se ejecuta en paralelo, y tú indujiste un fallo en un nodo Map del MapReduce1. A continuación te explico el proceso en detalle, fase por fase, centrándome en el nodo MapReduce1, pero mencionando también los aspectos relevantes de MapReduce2.

**Fase de Map:**

1. **Inicio del proceso Map**: Ambos coordinadores, MapReduce1 y MapReduce2, comienzan a procesar sus respectivos chunks de datos en paralelo. MapReduce1 tiene 4 nodos Map, que comienzan a procesar los chunks del archivo de entrada:
   * Nodo Map 2 del MapReduce1 procesa el chunk\_2.
   * Nodo Map 1 del MapReduce1 procesa el chunk\_1.
   * Nodo Map 3 del MapReduce1 procesa el chunk\_3.
2. **Fallo inducido en el nodo Map 0**: Indujiste un fallo en el nodo Map 0 del MapReduce1, que estaba asignado al chunk\_0. Debido a esto, el nodo Map 0 falla, y el sistema lo detecta:
   * El sistema marca el nodo como fallido y automáticamente reasigna el chunk\_0 al nodo Map 1, que procesa exitosamente este chunk después de la reasignación.
3. **Proceso Map completado**: Después de la reasignación, todos los nodos del MapReduce1 completan exitosamente sus tareas en la fase Map. Cada nodo procesa su chunk asignado, incluyendo los chunks reasignados por fallos.

Simultáneamente, en el MapReduce2, todos sus nodos Map (nodos 0, 1, 2, y 3) procesan sus respectivos chunks sin fallos, completando la fase Map exitosamente.

**Fase de Shuffle:**

1. **Distribución de los resultados de Map**: Después de la fase Map, se procede a la fase Shuffle, donde los resultados de los nodos Map se reorganizan y distribuyen entre los nodos Shuffle.
2. **Ejecución de los nodos Shuffle**:
   * Los 4 nodos Shuffle de MapReduce1 (nodos 0, 1, 2, 3) procesan los subsets generados por los nodos Map. Dado que no hubo fallos inducidos en esta fase, todos los nodos Shuffle de MapReduce1 completan exitosamente el procesamiento de sus subsets.

De forma similar, en MapReduce2, los nodos Shuffle procesan y reorganizan los resultados de los nodos Map sin errores, completando también su fase Shuffle con éxito.

**Fase de Reduce:**

1. **Ejecución de la fase Reduce**: Una vez que los nodos Shuffle han reorganizado los datos, la fase Reduce toma estos subsets reorganizados y los combina en resultados finales.
2. **Procesamiento de los nodos Reduce**:
   * MapReduce1 cuenta con dos nodos Reduce (nodos 1 y 2), que procesan los resultados de los nodos Shuffle. Ambos nodos completan exitosamente el procesamiento de los subsets asignados sin errores.

Simultáneamente, MapReduce2 también ejecuta su fase Reduce sin errores, con ambos nodos Reduce (nodos 1 y 2) procesando sus subsets y completando la tarea exitosamente.

**Nodo Final Reduce:**

1. **Combinación final**: Después de que los nodos Reduce de ambos MapReducers completan su trabajo, se pasa a la fase final de combinación de resultados. El Nodo Final Reduce toma los archivos de salida generados por los nodos Reduce y los combina en un solo archivo de salida.
2. **Ejecución exitosa**: Dado que no induciste fallos en esta fase, el Nodo Final Reduce combina exitosamente los resultados de MapReduce1 y MapReduce2.

**Resumen del fallo en MapReduce1:**

* El fallo en el nodo Map 0 del MapReduce1 fue detectado y manejado automáticamente por el sistema, reasignando el chunk\_0 al nodo Map 1, que lo procesó exitosamente.
* El resto de los nodos Map, Shuffle, y Reduce de MapReduce1 completaron sus respectivas fases sin problemas después de la reasignación.
* MapReduce2 no experimentó fallos y completó todas sus fases sin incidentes.

Este mecanismo asegura que los errores en los nodos no detienen el procesamiento global y que el sistema reasigna las tareas fallidas a otros nodos disponibles para mantener el flujo de trabajo.

Caso: Fallo en Nodo Shuffle de MapReduce1

En este caso, indujiste un fallo en un nodo Shuffle del MapReduce1. A continuación te explico cómo fue el proceso de los dos MapReducers (MapReduce1 y MapReduce2) en cada fase y cómo se manejó el fallo inducido en el nodo Shuffle del MapReduce1.

**Fase de Map:**

1. **Inicio del proceso Map**: Ambos coordinadores, MapReduce1 y MapReduce2, comienzan a procesar sus respectivos chunks en paralelo.
   * **MapReduce1**: Sus cuatro nodos Map (nodos 0, 1, 2 y 3) procesan los chunks chunk\_0 a chunk\_3. Posteriormente, procesan más chunks hasta completar los 20 chunks asignados.
   * **MapReduce2**: Sus nodos Map (nodos 0, 1, 2 y 3) procesan también los chunks correspondientes (chunk\_20 a chunk\_23) y luego otros chunks hasta completar sus 21 chunks asignados.
2. **Finalización exitosa de la fase Map**: Todos los nodos Map de MapReduce1 y MapReduce2 completan exitosamente el procesamiento de sus chunks. No hubo fallos en esta fase para ninguno de los dos MapReducers.

**Fase de Shuffle:**

1. **Distribución de los resultados de Map**: Después de la fase Map, se procede a la fase Shuffle, donde los resultados generados por los nodos Map se reorganizan y distribuyen entre los nodos Shuffle. Los subsets se asignan a cuatro nodos Shuffle en cada MapReducer.
2. **Fallo inducido en el nodo Shuffle 0 de MapReduce1**:
   * Induces un fallo en el nodo Shuffle 0 de MapReduce1 mientras procesaba el subset\_0. El sistema detecta este fallo, marca el nodo Shuffle 0 como fallido, y automáticamente reasigna el subset\_0 a otro nodo Shuffle.
   * El subset\_0 es reasignado al nodo Shuffle 1 de MapReduce1, que procesa exitosamente el subset después de la reasignación.
3. **Procesamiento exitoso en otros nodos Shuffle**:
   * **MapReduce1**: Los otros nodos Shuffle de MapReduce1 (nodos 1, 2 y 3) procesan exitosamente sus respectivos subsets sin fallos.
   * **MapReduce2**: Todos los nodos Shuffle de MapReduce2 (nodos 0, 1, 2 y 3) procesan sus subsets sin errores, completando la fase Shuffle con éxito.

**Fase de Reduce:**

1. **Combinación de los subsets Shuffle**: Tras la fase Shuffle, los resultados reorganizados se envían a los nodos Reduce, que combinan los subsets procesados.
   * **MapReduce1**: Los nodos Reduce (nodos 1 y 2) procesan los subsets reorganizados sin fallos. Todos los subsets de Shuffle son combinados correctamente, y la fase Reduce se completa exitosamente.
   * **MapReduce2**: Los nodos Reduce de MapReduce2 también completan su fase sin incidentes, combinando correctamente los subsets.

**Nodo Final Reduce:**

1. **Combinación final de resultados**: Después de que ambos MapReducers completan sus fases Reduce, el Nodo Final Reduce combina los resultados de los nodos Reduce en un archivo final.
   * **MapReduce1 y MapReduce2**: El Nodo Final Reduce combina exitosamente los resultados de ambos MapReducers.

**Resumen del fallo en MapReduce1:**

* **Fase Map**: No hubo fallos en los nodos Map de MapReduce1 o MapReduce2.
* **Fallo en nodo Shuffle 0 del MapReduce1**: El fallo inducido fue detectado y manejado correctamente. El sistema reasignó el subset\_0 a otro nodo Shuffle (nodo 1), que lo procesó exitosamente. Los otros nodos Shuffle de MapReduce1 procesaron sin problemas sus subsets asignados.
* **Fase Reduce**: Los nodos Reduce de ambos MapReducers completaron exitosamente sus tareas sin fallos.
* **Nodo Final Reduce**: Combinó los resultados de ambos MapReducers sin incidentes.

Este proceso demuestra cómo el sistema es capaz de manejar fallos en nodos individuales (como un nodo Shuffle) y reasignar tareas automáticamente, asegurando que el procesamiento global no se vea interrumpido.

Caso: Fallo en Nodo Reduce de MapReduce2

En este caso, induciste un fallo en los nodos Reduce del MapReduce2. A continuación te explico cómo fue el proceso de ambos MapReducers (MapReduce1 y MapReduce2) en cada fase y cómo se manejó el fallo inducido en los nodos Reduce del MapReduce2.

**Fase de Map:**

1. **Inicio del proceso Map**: Ambos coordinadores, MapReduce1 y MapReduce2, comienzan a procesar sus respectivos chunks de datos en paralelo.
   * **MapReduce1**: Los nodos Map (nodos 0, 1, 2, y 3) procesan sus chunks correspondientes (chunk\_0 a chunk\_3) y continúan con los siguientes chunks sin errores, completando la fase Map exitosamente.
   * **MapReduce2**: Los nodos Map (nodos 0, 1, 2, y 3) procesan sus chunks correspondientes (chunk\_20 a chunk\_23) y otros chunks, completando también su fase Map sin problemas.
2. **Finalización exitosa de la fase Map**: Todos los nodos Map de ambos MapReducers completan exitosamente sus respectivas tareas de procesamiento de chunks.

**Fase de Shuffle:**

1. **Distribución de los resultados de Map**: Después de la fase Map, se procede a la fase Shuffle. Los resultados de los nodos Map se reorganizan y distribuyen en subsets que son procesados por los nodos Shuffle.
   * **MapReduce1**: Los nodos Shuffle (nodos 0, 1, 2, y 3) procesan exitosamente sus subsets, reorganizando los resultados de la fase Map sin fallos.
   * **MapReduce2**: Los nodos Shuffle (nodos 0, 1, 2, y 3) también procesan exitosamente sus subsets y completan la fase Shuffle sin incidentes.

**Fase de Reduce:**

1. **Fallo inducido en los nodos Reduce de MapReduce2**:
   * Los nodos Reduce (nodos 1 y 2) del MapReduce2 procesan los resultados reorganizados por la fase Shuffle. Sin embargo, induciste un fallo en ambos nodos Reduce de MapReduce2, lo que provoca que fallen durante el procesamiento de los subsets.
   * El sistema detecta los fallos en ambos nodos Reduce (nodo Reduce 1 y nodo Reduce 2), e imprime mensajes indicando el fallo inducido:

java

Copy code

Error inducido en el Nodo Reduce 2 del MapReduce2. Fallo en el procesamiento de shuffleFiles.

Error inducido en el Nodo Reduce 1 del MapReduce2. Fallo en el procesamiento de shuffleFiles.

1. **Reinicio de los nodos Reduce**:
   * Después del fallo inducido, el sistema intenta reiniciar los nodos Reduce. Los nodos Reduce 1 y 2 de MapReduce2 son reiniciados, y el sistema intenta procesar nuevamente los mismos subsets.
   * Tras el reinicio, ambos nodos Reduce de MapReduce2 completan exitosamente el procesamiento de los subsets que habían fallado:

java

Copy code

REDUCE Nodo 2 del MapReduce2 finalizó exitosamente el procesamiento de shuffle\_2.txt

REDUCE Nodo 1 del MapReduce2 finalizó exitosamente el procesamiento de shuffle\_0.txt

REDUCE Nodo 2 del MapReduce2 finalizó exitosamente el procesamiento de shuffle\_3.txt

REDUCE Nodo 1 del MapReduce2 finalizó exitosamente el procesamiento de shuffle\_1.txt

1. **MapReduce1 en la fase Reduce**:
   * En paralelo, los nodos Reduce de MapReduce1 (nodos 1 y 2) completan la fase Reduce sin fallos. Todos los subsets son procesados correctamente y la fase Reduce se completa exitosamente para MapReduce1.

**Nodo Final Reduce:**

1. **Combinación final de resultados**:
   * Después de que los nodos Reduce de ambos MapReducers completan sus tareas, el Nodo Final Reduce combina los resultados de ambos en un archivo final.
   * La combinación final se lleva a cabo exitosamente, y los resultados de los nodos Reduce de ambos MapReducers se integran sin problemas:

java

Copy code

Nodo Final Reduce: Combinando resultados de archivos de reducción...

Nodo Final Reduce: Combinación de resultados completada exitosamente.

**Resumen del fallo en los nodos Reduce del MapReduce2:**

* **Fase Map**: Los nodos Map de ambos MapReducers completaron la fase sin fallos.
* **Fase Shuffle**: Tanto en MapReduce1 como en MapReduce2, los nodos Shuffle reorganizaron los resultados de los nodos Map sin problemas.
* **Fallo en los nodos Reduce del MapReduce2**: Ambos nodos Reduce de MapReduce2 fallaron, pero el sistema detectó los errores y reinició los nodos Reduce, que finalmente completaron sus tareas exitosamente.
* **Nodo Final Reduce**: Combinó los resultados de los nodos Reduce de ambos MapReducers en un archivo final, y la combinación fue completada con éxito.

Este proceso muestra la capacidad del sistema para gestionar fallos en los nodos Reduce, reasignar y reiniciar el procesamiento fallido, asegurando que la ejecución global continúe sin interrupciones graves.

Caso: Fallo en Nodo FinalReduce (aquí no importa de qué MapReduce es, porque está fuera)

En este caso, induciste un fallo en el **Nodo Final Reduce**, que es el nodo encargado de combinar los resultados de ambos MapReducers una vez que todas las fases de Map, Shuffle, y Reduce han sido completadas.

**Explicación del Proceso:**

**Fase Map:**

1. **Procesamiento de chunks en MapReduce1 y MapReduce2**: Los nodos Map en ambos MapReducers procesan sus respectivos chunks de datos en paralelo.
   * **MapReduce1**: Nodos 0, 1, 2, y 3 procesan los chunks chunk\_0 a chunk\_19.
   * **MapReduce2**: Nodos 0, 1, 2, y 3 procesan los chunks chunk\_20 a chunk\_40.
   * Ambos MapReducers completan exitosamente la fase Map sin errores.

**Fase Shuffle:**

1. **Redistribución de datos en MapReduce1 y MapReduce2**: Luego de la fase Map, se reorganizan los datos en subsets.
   * **MapReduce1**: Los nodos Shuffle de MapReduce1 (0, 1, 2, y 3) procesan y reorganizan exitosamente los resultados de la fase Map en subsets correspondientes a sus chunks.
   * **MapReduce2**: Los nodos Shuffle de MapReduce2 hacen lo mismo con sus propios chunks reorganizando los resultados para la fase Reduce.
   * Ambos MapReducers completan la fase Shuffle exitosamente.

**Fase Reduce:**

1. **Reducción de resultados en MapReduce1 y MapReduce2**: La fase Reduce toma los datos reorganizados y los procesa en cada nodo Reduce.
   * **MapReduce1**: Nodos Reduce 1 y 2 procesan todos los subsets sin fallos.
   * **MapReduce2**: Nodos Reduce 1 y 2 completan la fase de Reduce sin errores.
   * Los resultados son listos para ser combinados en la fase final.

**Fase Final Reduce:**

1. **Fallo inducido en el Nodo Final Reduce**: Este nodo se encuentra fuera de los MapReducers y su función es combinar los resultados de los nodos Reduce de ambos MapReducers.
   * Induciste un fallo en este nodo, lo que provocó que la combinación de los archivos de reducción fallara:

java

Copy code

Error inducido en el Nodo Final Reduce. Fallo en la combinación de resultados.

**Mecanismo de Recuperación:**

1. **Reinicio del Nodo Final Reduce**: Después de detectar el fallo, el sistema reinicia el nodo:

java

Copy code

Reiniciando Nodo Final Reduce después del fallo...

Una vez reiniciado, el Nodo Final Reduce intenta nuevamente combinar los resultados y esta vez lo hace exitosamente:

java

Copy code

Nodo Final Reduce: Combinando resultados de archivos de reducción...

Nodo Final Reduce: Combinación de resultados completada exitosamente.

**Importancia del Nodo Final Reduce:**

* Como mencionas, el **Nodo Final Reduce** está fuera del flujo de los MapReducers (1 y 2) y no depende directamente de ellos una vez que han completado su fase de Reduce. Este nodo es crucial para la última etapa del proceso, ya que combina los resultados de los dos MapReducers para generar el resultado final.
* Si este nodo falla, ningún MapReducer se ve afectado, pero es imposible obtener el resultado final hasta que se complete la combinación de datos, lo cual ocurrió tras el reinicio.

En resumen, este nodo es responsable de la última parte del proceso y su correcto funcionamiento es clave para generar la salida final del trabajo de los MapReducers.