



Informe Laboratorio 2 - Sistemas Distribuidos

Claudia Hazard Valdés - Rol: 201404523-9

Samuel Marin Navarro - Rol: 201604503-1

12 de Diciembre de 2020.

1. INTRODUCCIÓN

En este informe se realiza la implementación para un sistema de biblioteca en el que se pueden subir libros, los cuales se dividen en Chunks y distribuyen entre tres DataNodes. Además, se pueden descargar dichos libros consultando por la ubicación de cada Chunk, para luego armarlo y obtenerlo como pdf nuevamente.

2. IMPLEMENTACIÓN

Se realiza un sistema distribuido en el que el Cliente-Uploader fragmenta en Chunks y envía estos a uno de los tres NameNodes. Al recibir todos los Chunk, el DataNode genera una propuesta de distribución la cual puede utilizarse de forma *Centralizada* o *Distribuida*, hasta que esta sea aceptada. Luego de que la distribución es aceptada, el DataNode envía cada Chunk según la distribución al DataNode correspondiente. Al enviar todos los Chunk, el DataNode guarda en el node nuevamente con implementación *Centralizada* o *Distribuida* las cuales se explicaran a detalle en los siguientes puntos.

2.1. Centralizada

En este método, las decisiones son tomadas por parte del NameNode, por lo que los DataNodes deben consultar a este.

2.1.1. Propuesta de Distribución

Para utilizar la distribución el DataNode envía una propuesta al NameNode hasta que este acepte la propuesta. Al recibir la propuesta el NameNode envía un mensaje a los DataNodes que estén en la propuesta para chequear que no se encuentren caídos:

- En el caso de que todos los DataNodes estén disponibles la propuesta es aceptada.
- En el caso de que un DataNode esté caído, la propuesta es rechazada.

2.1.2. Utilización del Log

El DataNode consulta por el uso del Log al NameNode, el NameNode tiene guardado si algún otro DataNode está utilizándolo:

- En el caso de que no se esté utilizando el Log, el NameNode envía un mensaje *Ok* y el DataNode utiliza el Log enviando luego un mensaje de *Release*.
- En el caso de que un DataNode esté utilizando el Log, el NameNode espera a que este lo deje de utilizar para poder enviar un mensaje *Ok*.

Lo explicado anteriormente, puede observarse en la Figura 1. En la parte *a*, dos DataNode enviaron un mensaje para utilizar el Log, pero el NameNode solo responde *Ok* a uno de ellos. En la parte *b* el DataNode1 que ha terminado de usar el Log envía un mensaje de *Release*, por lo que el NameNode responde con un mensaje de *Ok* al DataNode2 que se encontraba esperando.

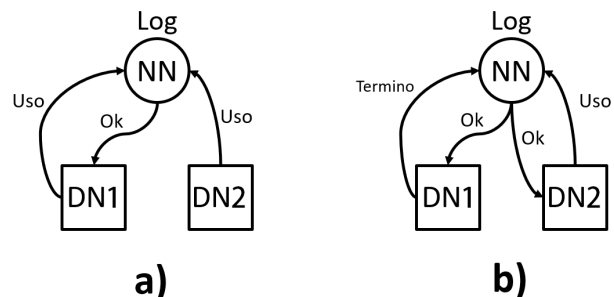


Figura 1. Envío peticiones del uso del Log para método Centralizado, donde DN1, DN2 es DataNode 1 y 2. NN corresponde al NameNode.

2.2. Distribuida

En este método, las decisiones son consultadas a los otros DataNodes, por lo que los DataNodes deben enviar mensajes a sus pares.

2.2.1. Propuesta de Distribución

Para utilizar la distribución el DataNode envía una propuesta a los otros DataNodes hasta que todos los DataNodes de la propuesta se encuentren activos (No estén caídos).

Al enviar la propuesta, existen dos casos:

- El DataNode recibe un mensaje de *Ok* por parte de los otros DataNode, en este caso la propuesta es aceptada.
- El DataNode no recibe un mensaje de *Ok* por uno o mas DataNode, por lo que se genera una nueva propuesta y se envía un mensaje a los DataNodes que estén involucrados en esta.

2.2.2. Utilización del Log

El DataNode envía un mensaje de consulta por el uso del Log a los otros DataNode con un Timestamp, en el que al intentar acceder dos nodos a la vez, utilizara el Log el DataNode que tenga el menor Timestamp.

Debido a que este método genera una espera infinita en el caso de que uno de los DataNodes este caído, pues se esperará por un mensaje de *Ok* por parte de todos los nodos para poder escribir, se decidió lidiar con este problema agregando un chequeo de conexión antes de consultar por el uso del Log. En el que:

- Si el DataNode al que se consulta está caído, se toma como que es posible utilizar el Log.
- Si el DataNode al que se consulta no está caído, se realiza el intercambio de mensajes hasta esperar por un *Ok*.

Lo explicado anteriormente se muestra en la Figura 2, que corresponde a la implementación del algoritmo de distribución de Ricart y Agrawala. En la parte *a* tanto el DataNode2 como el DataNode3 envían que utilizaran el Log, pero el DataNode2 tiene un Timestamp menor que el DataNode3 por lo que este le responde con un mensaje *Ok*. En *b*, luego de recibir mensajes *Ok* de los otros dos DataNodes, el DataNode2 escribe en el NameNode y este le responde que ya fue escrito, por lo que el DataNode2 puede responder con un mensaje de *Ok* a la petición realizada anteriormente en *a*.

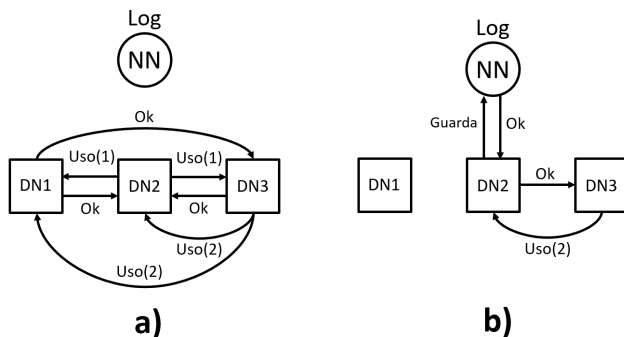


Figura 2. Envío peticiones del uso del Log para método Distribuido, donde DN1, DN2, DN3 corresponde a los DataNodes 1, 2 y 3 respectivamente. NN corresponde al NameNode.

3. RESULTADOS

Se realizaron 6 ejecuciones, las cuales se dividen en 3 grupos. Realizando 3 distintas desconexiones para cada método (Distribuido y Centralizado):

1. Centralizada, Sin nodos caídos.
2. Distribuida, Sin nodos caídos.
3. Centralizada, con el DataNode 3 caído en 2 libros.
4. Distribuida, con el DataNode 3 caído en 2 libros.
5. Centralizada, Nodo 3 caído en 2 libros, luego reconectado. Luego Nodo 2 desconectado en 2 libros.
6. Distribuida, Nodo 3 caído en 2 libros, luego reconectado. Luego Nodo 2 desconectado en 2 libros.

Los resultados obtenidos se muestran en las Tablas 1, 2 y 3. Estos resultados se analizaran en la siguiente sección.

Los mensajes calculados corresponden a los mensajes enviados a otro DataNode o al NameNode en las operaciones de Propuesta de distribución y Uso del Log. No se agregaron mensajes de envío de Chunks entre DataNodes pues estos funcionarían de misma manera en todas las ejecuciones. Estos mensajes corresponden a:

- Centralizado: Mensajes de DataNodes a NameNode por propuesta de distribución, mensajes de chequeo de conexión del NameNode a los DataNodes para aceptar propuesta, mensajes de DataNodes a NameNode por uso del Log.
- Distribuido: Mensajes de DataNodes a DataNodes por propuesta de distribución, mensajes de uso del Log de DataNodes a DataNodes, mensaje de escritura del Log de un DataNode al NameNode.

Ejecución	DN1 [msj]	DN2 [msj]	DN3 [msj]	NN [msj]	MT [msj]
1	8	8	13	43	72
2	16	16	24	0	56
3	9	11	11	47	78
4	17	21	20	0	58
5	10	10	15	55	90
6	18	18	26	0	62

Tabla 1

Mensajes obtenidos en cada ejecución. DN1, DN2, DN3 corresponden a los DataNodes 1, 2, 3. NN corresponde al NameNode y MT corresponde al total de mensajes enviados en esa ejecución.

El tiempo calculado es la suma de tiempos corresponde a:

- Centralizado: La suma del tiempo calculado cada vez que el DataNode envió la propuesta a escribir al NameNode y recibió el mensaje de que fue escrita.
- Distribuido: La suma del tiempo calculado cada vez que el DataNode envió mensajes para escribir a los DataNodes, recibió una respuesta de *Ok* y utilizó el Log.

Ejecución	DN1 [s]	DN2 [s]	DN3 [s]	TT [s]
1	0.01924355	0.0211332	0.10330499	0.14368175
2	0.0449389	0.0440409	0.09108134	0.18006113
3	0.01599456	0.03944898	0.03842678	0.09387032
4	0.05592405	0.08057064	0.09908029	0.23557498
5	0.01503952	0.01992678	0.01684957	0.05181586
6	0.05464686	0.03872429	0.05359873	0.14696989

Tabla 2

Tiempo utilizado en total al escribir en el Log, contando el tiempo de espera de respuesta según el método utilizado. DN1, DN2, DN3 corresponden a los DataNodes 1,2,3. TT corresponde al tiempo total en esa ejecución.

Ejecución	DN1 [libro]	DN2 [libro]	DN3 [libro]
1	4	4	6
2	4	4	6
3	4	5	5
4	4	5	5
5	4	4	6
6	4	4	6

Tabla 3

Libros procesados por cada DataNode. DN1, DN2, DN3 corresponden a los DataNodes 1,2,3.

4. ANÁLISIS Y DISCUSIÓN

De los resultados obtenidos, se observa que la cantidad de mensajes enviados aumenta al existir nodos caídos, lo cual tiene sentido pues se deben generar mas propuestas.

La cantidad de mensajes enviados por el NameNode corresponde a la cantidad de veces que chequeó nodos conectados para aceptar una petición, esta cantidad es bastante alta en comparación a utilizar el método Distribuido. Cabe recalcar que entre los mensajes mostrados en el método Centralizado, cada nodo envía peticiones a sus pares pero no se agregan al contador los mensajes a si mismo.

En cuanto a tiempo el sistema Centralizado obtiene resultados con menores tiempos, esto puede deberse a que puede entregar una respuesta instantánea en caso de que otros DataNodes no estén utilizando el Log, mientras que en el sistema Distribuido siempre se debe consultar a los otros DataNodes y esperar la respuesta de estos.

En la ejecución 6, el DataNode 3 envió una mayor cantidad de mensajes pero obtuvo menor tiempo que en las ejecuciones anteriores, esto se debe al envío de nuevas propuestas, junto con solo estar usando el Log con el DataNode1 en el momento que el DataNode2 estaba caído, como se mencionó anteriormente ya que al estar un nodo caído esperar por la respuesta de este para el uso del Log tomará un tiempo infinito, se realiza un chequeo de conexión previo al envío del uso del Log, por lo que el DataNode no debe esperar una respuesta si ese DataNode está caído.

La cantidad de libros fue enviada de forma parcialmente equitativa, a pesar de existir desconexiones en las pruebas realizadas, los libros se enviaron a las maquinas en una cantidad similar.

A pesar de que el tiempo en la ejecución 1 es menor que en la ejecución 2, se observa que el tiempo de espera del

DataNode 3 es mucho mayor que para los DataNodes 1 y 2 en la ejecución 1, mientras que es relativamente similar en la ejecución 2. Este resultado tiene sentido debido a la prioridad del DataNode 1 sobre los otros DataNode en el sistema centralizado, mientras que el sistema Distribuido es dependiente del Timestamp.

Finalmente, se observa que el tiempo de las primeras ejecuciones es mucho mayor que los tiempos obtenidos en las ultimas, esto se debe a que a mayor cantidad de nodos a la vez, mas tiempo de espera se debe realizar, pues en las ejecuciones 5 y 6 en aproximadamente la mitad de los libros solo habían 2 DataNode funcionando a la vez.

5. CONCLUSIONES

Se realizo la implementación de dos sistemas de distribución y acceso de datos de los cuales ambos tienen ventajas y desventajas.

En cuanto al sistema Centralizado, el tiempo obtenido para utilizar el recurso compartido fue menor, pero la cantidad de mensajes fue mayor, además en el caso de que el NameNode se caiga los DataNodes no podrán realizar ninguna operación y si existiera una mayor cantidad de DataNodes se debería generar algún sistema de prioridad para que todos reciban respuestas en un tiempo apropiado.

En cuanto al Sistema Distribuido, la cantidad de mensajes enviados fue menor pues no chequea la conexión de si mismo para aceptar una propuesta, en cuanto a la implementación de los mensajes para el uso del Log es un poco mas compleja que el sistema Centralizado y puede generarse una espera infinita en el caso de que se espere un mensaje de vuelta de un DataNode caído.

De los resultados obtenidos para esta implementación se obtuvo que se envían mas mensajes al existir un nodo caído, pero también el tiempo de espera para el uso del Log es menor. Además, el DataNode debe enviar mensajes de chequeo constantemente para aceptar las propuestas de todos los Libros en el sistema Centralizado.