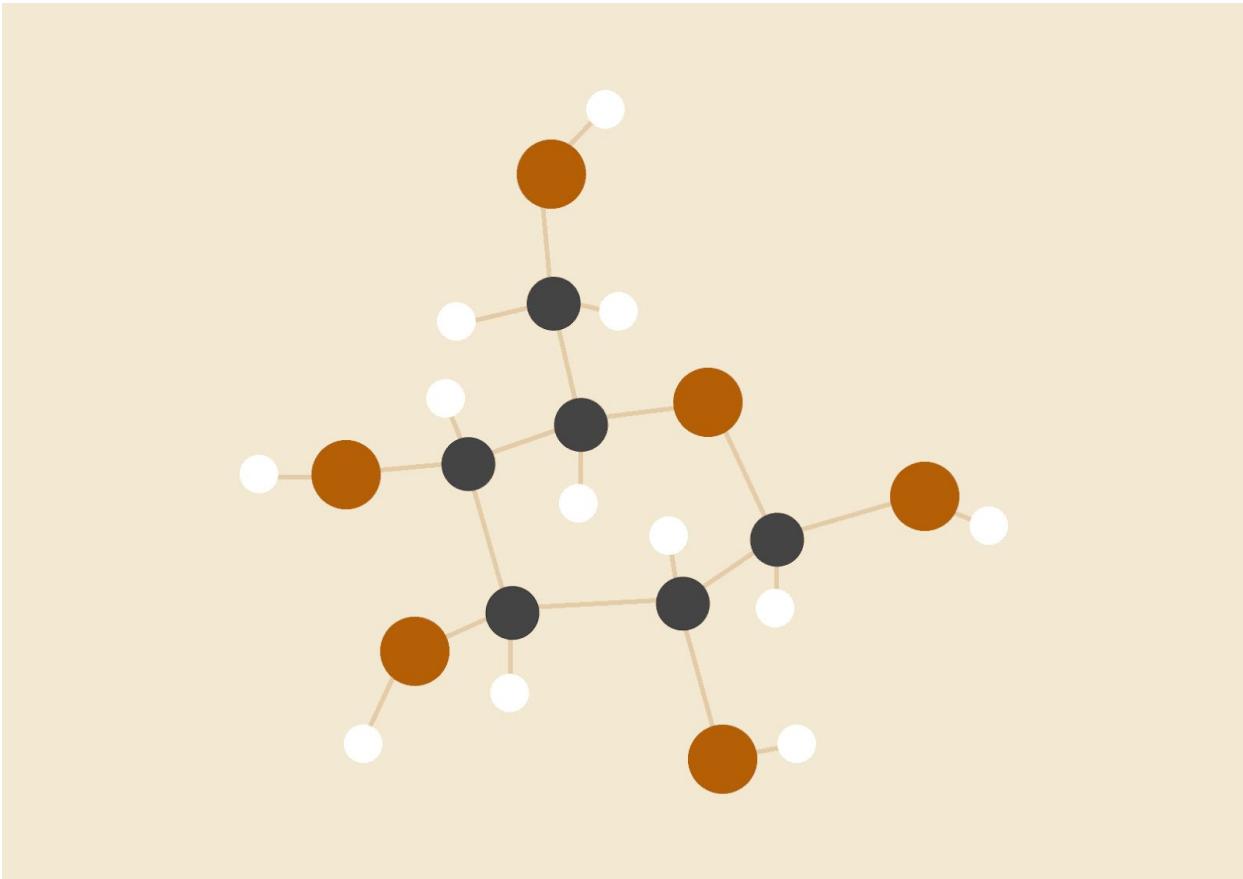


Laboratorio 2

Sistemas Distribuidos



Martín Menares Valencia - 201573536-0

Gabriel Pezoa Riutor - 201673562-3

RESÚMEN

Para el presente laboratorio se replicó un sistema de distribución de libros digitales de una biblioteca. Para esto, cada documento es fragmentado y repartido entre 3 terminales llamadas Data Nodes. Por otra parte, una cuarta terminal (Name Node) se encarga de mantener el registro de qué fragmento se encuentra en qué nodo.

Se implementaron dos métodos para resolver esta problemática, el primero funciona de manera centralizada, es decir, el Name Node toma las decisiones de dónde se guarda cada fragmento. El segundo método funciona de manera distribuida; el Data Node se comunica con sus pares para ver cuál está disponible para recibir fragmentos y luego notifica al Name node sobre la decisión que se tomó.

INTRODUCCIÓN

En la descripción del problema se identifican 4 actores principales:

- **Client Uploader:** Encargado de separar los documentos en chunks de tamaño 250kb. Luego de esto se comunica con alguno de los Data nodes y comienza la distribución de los fragmentos.
- **Client Downloader:** Encargado de recuperar los documentos y rearmar el libro completo. Para esto se comunica con el Name node para saber en qué nodos se encuentran los fragmentos y luego se comunica con ellos para recuperarlos.
- **Data node:** Encargado de almacenar los chunks de los libros. Tras recibir los chunks de parte del cliente, dependiendo del método que se utilice, hará:
 - **Centralizado:** Generará una propuesta de distribución y se la enviará al Namenode para que de su aprobación o genera una nueva. Luego de recibir la respuesta del Name node, realizará la distribución que recibió en esa respuesta.
 - **Distribuido:** Enviará la propuesta a los otros Data nodes. Se obtiene el estado de los otros nodos en base a la respuesta que se recibe y se distribuyen los chunks entre los nodos disponibles. Finalmente se notifica al Name node la desición que se tomó.
- **Name node:** Encargado de mantener el registro de la distribución de los chunks entre los diferentes nodos en un archivo llamado log.txt. Si es que el algoritmo utilizado es el centralizado, además de la tarea recién mencionada, se encargará de tomar la decisión sobre la distribución de los chunks.

RESULTADOS

A continuación se muestran los resultados obtenidos con cada algoritmo utilizado:

- Tiempo transcurrido en Name node desde inicio hasta que escribe el Log

Centralizado			
	1 nodo vivo [s]	2 nodos vivos [s]	3 nodos vivos [s]
9 chunks	5.002259	2.506333	0.001386
8 chunks	5.001884	2.505487	0.001077

Distribuido			
	1 nodo vivo [μs]	2 nodos vivos [μs]	3 nodos vivos [μs]
9 chunks	392.2	389.9	394.3
8 chunks	359.5	351.6	377.4

- Cantidad de mensajes enviados

Centralizado		
1 nodo vivo	2 nodos vivos	3 nodos vivos
2	6	10

Distribuido		
1 nodo vivo	2 nodos vivos	3 nodos vivos
2	8	14

ANÁLISIS Y DISCUSIÓN

Sobre los resultados obtenidos en los tiempos de escritura en el Log, es posible observar que los tiempos relacionados con el algoritmo centralizado son más elevados que los del algoritmo distribuido. Esto se debe a que en el algoritmo centralizado, el Name node debe revisar la propuesta que le envía en nodo, verificar la disponibilidad de los otros nodos y luego escribir en el Log, mientras que en el algoritmo distribuido, el Name node recibe la distribución hecha por el nodo, por lo que su trabajo se reduce a guardar la distribución en el Log únicamente.

Además, en el algoritmo centralizado se observa un aumento en el tiempo, dependiendo de la cantidad de nodos que no se encuentran disponibles. Esto se debe a que para establecer cada conexión, el Name node intenta conectarse por 2.5 segundos a cada nodo.

Por otro lado, si se observa el número de mensajes que se intercambia en cada algoritmo, se tiene que el algoritmo distribuido realiza una mayor cantidad, debido a que cuando el nodo que tiene los chunks envía la propuesta al resto de los nodos, estos, a su vez, verifican que los nodos propuestos se encuentren disponibles, enviando dos mensajes más por cada nodo que esté disponible.

CONCLUSIÓN

En base a los resultados obtenidos, es posible concluir que el rendimiento del algoritmo centralizado, en cuanto al tiempo de ejecución, dependerá de la tasa de fallos de los nodos, ya que aumenta de manera considerable con cada nodo que no se encuentra disponible. En cuanto al intercambio de mensajes, el algoritmo centralizado aumenta de manera menos precipitada conforme aumenta la cantidad de nodos.

Por otro lado, el algoritmo distribuido se muestra inmutable en cuanto a los tiempos, ya que el trabajo del Name node se limita a escribir el Log únicamente. En cuanto a la cantidad de mensajes enviados, el algoritmo distribuido muestra ser menos eficiente debido a la cantidad de verificaciones que se realiza en cada nodo.

En base a esto es imperante considerar la cantidad de nodos que se pretende implementar y la tasa de fallo de estos para tomar la decisión de cuál algoritmo implementar.