Desduplicación.

Cruz Martínez Raúl
Ingeniería en computación,
FI, UNAM
CDMX, México
espacio.rcruz@gmail.com

Villanueva Corona Miguel Ángel
Ingeniería en computación,
FI, UNAM
CDMX, México
miguel8.villanueva.cch.vallejo@gmail.com

I. INTRODUCCIÓN

A finales de los años ochenta, el uso de computadoras personales se volvió relativamente común, y desde entonces, con este uso, muchos archivos dentro del almacenamiento se volvían repetitivos o tenían patrones similares. Es necesario recordar que, en ese entonces, el almacenamiento era reducido, es así que se debían buscar formas de comprimir la información de la forma más eficiente posible. Una de las propuestas para esto, fue la desduplicación.

De manera general la desduplicación es una característica de los sistemas operativos el cual nos ayuda a guardar una sola copia de los archivos que se encuentren repetidos dentro de la memoria, esto para optimizar mejor el almacenamiento de memoria a través de hashes criptográficos los cuales mencionaremos más adelante.[1]

II. DESARROLLO

Como mencionamos en la introducción, la desduplicación no es un mecanismo de ahorro de almacenamiento reciente (si es que podemos referirnos a algo dentro de la computación como "reciente"), pero, no hubo mucha explotación de este mecanismo, porque no existía una implementación realmente eficiente, y porque no existía una necesidad realmente demandante para que pusieran manos a la obra con la desduplicación.

Todo cambió con la llegada del uso masivo de los servidores, ya que debía reducirse lo más posible la información almacenada en ellos. Además de que "La ventaja máxima se observa en entornos virtuales donde se utilizan varias máquinas virtuales para las implementaciones de pruebas/desarrollo y aplicaciones [2]".

La desduplicación, como veremos más adelante, se puede mostrar en diferentes granularidades; para comenzar, podemos ver a la desduplicación como la comparación de secuencias de datos, y, que en caso de que estas secuencias sean parecidas, únicamente se colocará un apuntador a la primera secuencia almacenada, evitando así la duplicación y desperdicio de almacenamiento.

Granularidad

El factor de granularidad es de suma importancia para la desduplicación, ya que de este derivan 3 diferentes tipos de niveles, los cuales son: a nivel de archivos, a nivel de bloques y por último a nivel de bytes.

A nivel de archivo: también llamado por sus siglas SIS, es el primer nivel de granularidad en hacer uso de los hashes para comparar el contenido de los archivos generando un único identificador por cada fracción de datos analizados. "La ventaja de SIS es que requiere menos potencia de procesamiento ya que los números de hash de los ficheros son más fáciles de generar, resultado ser rápido y sencillo. Su principal desventaja radica en que, si se cambia un solo byte del archivo, el número de hash también cambia, por lo que ambas versiones del archivo deberán guardarse nuevamente por separado [3]".

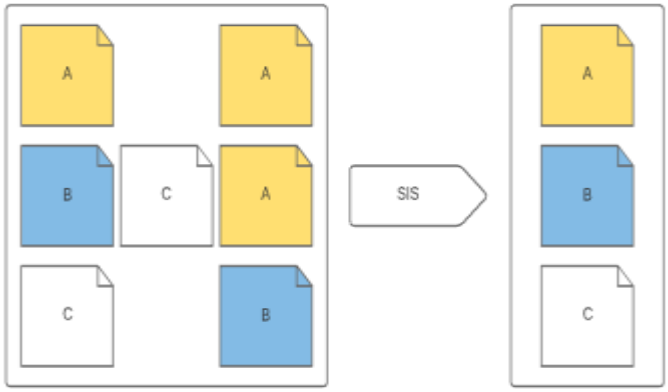

Fig.1 demostración grafica del nivel de archivo

A nivel de bloque: este tipo de nivel en lo que consiste es en separar por partes a los datos, para poder comparar de manera óptima, por lo tanto "cuanto más pequeños son los bloques, mayor es el número de fragmentos, y por consiguiente mayor es el número de comparaciones con el índice y mayor es el potencial de identificar y suprimir redundancias [4]". A diferencia del SIS es que si se llega a hacer un cambio este sólo guarda el bloque modificado, por lo que no es necesario guardar de nuevo todo el archivo. Una desventaja de este nivel es que si se llega a fragmentar en bloques muy pequeños, podemos tener un menor rendimiento.

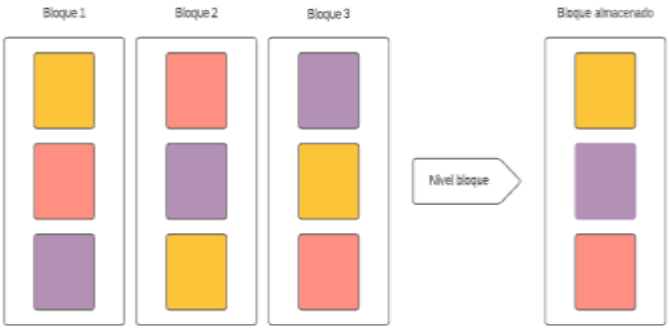

Fig.2 demostración grafica del nivel de bloque

A nivel de bytes: Este nivel es el que tiene una mayor granularidad en comparación con el SIS y a nivel de bloque, este nivel tiene una precisión más alta, ya que divide los datos byte por byte, por lo que al hacer modificaciones no se ve afectado a la hora de guardarlos. El defecto con este tipo de nivel es que se ve muy afectado en cuestión a su rendimiento.

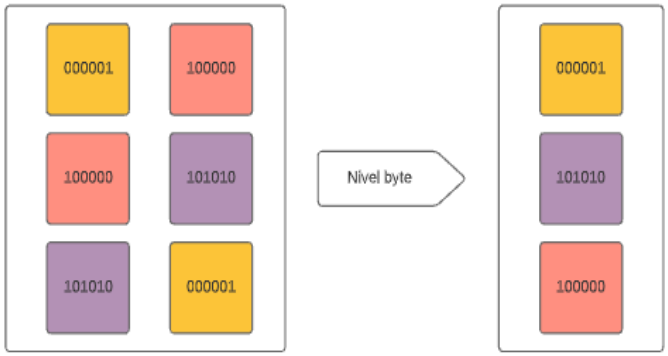

Fig.3 demostración grafica del nivel de bytes

Ya que hemos descrito lo necesario para tener una idea suficientemente buena de lo que es la desduplicación, podemos hablar de cómo es que podemos saber cómo se mide esta desduplicación. A decir verdad, el método de medición que vamos a describir a continuación es bastante intuitivo, pues se expresa a través de una relación, la cual involucra a la cantidad de datos totales sin desduplicar, contra los que ya fueron desduplicados, llevándonos así a una mejor apreciación de la cantidad de almacenamiento ahorrado con este mecanismo. [5]

Lo descrito en el párrafo anterior, lleva por nombre **Ratio de desduplicación**, el cual se calcula como:

$$Ratio = \frac{Bytes\ de\ entrada}{Bytes\ de\ salida}$$

En caso de querer expresar esta cantidad como porcentaje, bastará con aplicar lo siguiente [5]:

$$Porcentaje = \left[1 - \left(\frac{1}{Ratio}\right)\right] \times 100\%$$

Tipos de desduplicación

Desduplicación online: este tipo de desduplicación se basa en que previamente guarda un bloque, el cual va a ser de utilidad para cuando se llegue a duplicar dicho bloque, ya que cuando esto pase, no se va a volver a almacenar si no que le va a crear un puntero que apunte al bloque previamente almacenado, debido a que su estructura primero optimiza los datos y después los almacena. La desventaja de esto es que si se llegan a meter una gran cantidad de datos pude desfavorecer su rendimiento.

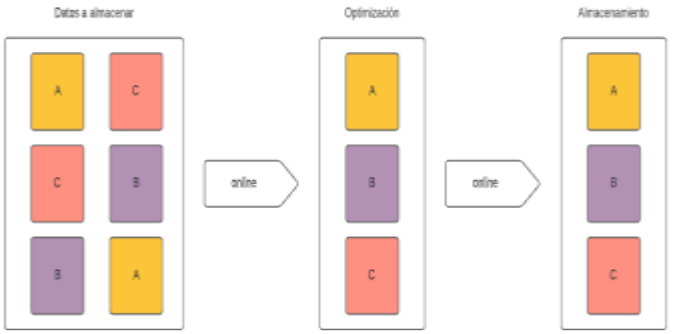

Fig.4 demostración grafica de desduplicación online

Desduplicación offline: A diferencia de la desduplicación online, este tipo de desduplicación lo primero que hace es almacenar todos los datos que lleguen para posteriormente analizar y optimizar los datos para dejar solo una copia de dichos datos. Pero su desventaja es que necesita demasiado espacio de almacenamiento para que pueda hacer la desduplicación, y esto es un problema muy grande si es que el almacenamiento se encuentra a punto de llegar a su capacidad máxima.

Fig.5 demostración grafica de desduplicación offline

Desduplicación a través del hashing criptográfico.

Hasta este momento no hemos visto cómo es que podemos verificar si alguna secuencia de bytes ya se encuentra en disco o no, para esto, es momento de presentar el hash criptográfico como uno de los métodos para la desduplicación.

La importancia de utilizar el hash criptográfico es, que este genera llaves con poca probabilidad de colisión y seguridad suficiente como para manejar grandes cantidades de información.

Los algoritmos más usados son MD5, SHA-1 Y SHA-2 [5], y la eficiencia de estos algoritmos, dependerá del tipo de secuencias de bytes a manejar, pues pueden ser secuencias cortas, lo que nos permite la resolución de llaves con operaciones aritméticas sencillas, lo que reduce el tiempo de cómputo; podemos manejar secuencias largas, que tardarán mayor tiempo en ser procesadas, pero reducen aún más las posibilidades de colisión entre las llaves [6]; y, podemos seleccionar secuencias de bytes de longitud variable, que implicará mayor tiempo de cómputo, pues se apoya de puntos clave en los archivos, pero, encontrará mayores índices de duplicación, lo que implica un mayor ahorro de almacenamiento.

Hablaremos primero del algoritmo MD5, el cual fue creado en 1991 por Ronald Rivest en el MIT. MD5 es utilizado cuando tenemos secuencias de bytes variables, pues, a pesar de que la longitud es variable, nos entregará una llave de 128 bits (16 bytes), para escribir así 32 caracteres hexadecimales [7].

Por otro lado, está el SHA-1 (Secure Hash Algorithm), fue diseñado por la Agencia de Seguridad Nacional (NSA) y por el Instituto Nacional de Estándares y Tecnología (NIST) en 1995. En este caso, la entrada puede ser de longitud variable, pues el tamaño máximo de entrada es de 2^{64} bits, y entregará una salida de 160 bits, lo que equivale a 20 bytes [5].

El SHA-1 permite segmentaciones grandes de secuencias, pero, debemos considerar que para que la secuencia sea procesada en menor tiempo, esta deberá estar en memoria, y, en caso de ser muy grande, podrían rebasar el tamaño de la memoria, lo que implica guardar en disco, lo que inevitablemente alentará mucho el procesamiento de las llaves.

En cuanto a este apartado, finalmente hemos de decir, que, en cuanto más aumente el tamaño de los servidores, más grandes deberán ser los índices.

Desduplicación y overcommit

En cuestión al sobre comprometimiento podemos decir que hay un producto de IBM el cual nos dice que la desduplicación de memoria activa reduce el overcommit al eliminar páginas de la memoria para eliminar o minimizar fragmentos idénticos de la memoria [8].

Ventajas y Desventajas.

Como vimos a lo largo de este documento, existen múltiples formas de implementar la desduplicación, y estas formas tan diversas, serán aplicables en varias circunstancias. Es así que analizaremos de forma general las ventajas y desventajas de la desduplicación.

Ventajas:

Como ventaja principal se encuentra la optimización del espacio físico es bastante buena, aunque el aprovechamiento de este dependerá totalmente de cómo esté estructurado el sistema, pues mientras mayor cantidad de desduplicación sea propenso a tener, el aprovechamiento será muy grande.

Dependiendo de si hacemos una desduplicación en línea o fuera de línea, podremos reducir el tiempo de procesamiento y escritura de la información en disco o en los servidores.

Finalmente, hemos de decir que, al tener identificadores para segmentos específicos de bytes, podremos realizar mantenimientos de forma mucho más sencilla.

Desventajas:

La primera ventaja presentada, va totalmente de la mano con una desventaja, pues, llega un punto, en el que el ahorro de almacenamiento empieza a llegar a una cota, viéndose limitado el aprovechamiento del método.

Para ver más claro este punto, debemos recordar que la forma en la que podemos medir el aprovechamiento de las desduplicación es el Ratio.

Un ejemplo de que no todo es miel sobre hojuelas, se muestra en el siguiente gráfico [9]:

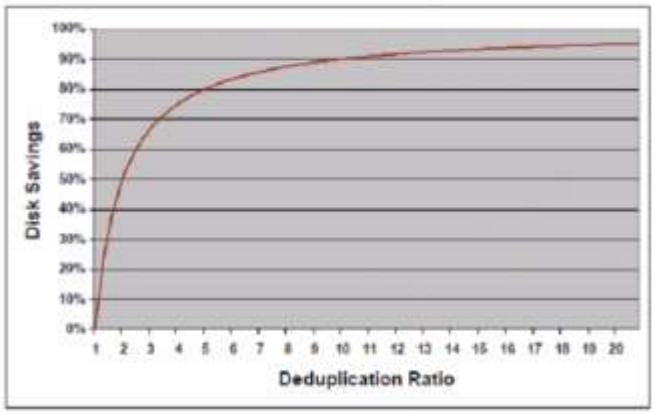

Fig. 6. Comparativa del porcentaje del aprovechamiento del almacenamiento contra los Ratios.

Con la gráfica, es mucho más fácil visualizar que a partir de los 10 Ratios, el incremento de la eficiencia no aumenta más allá del 96%.

Por otra parte, la desduplicación, será efectiva sólo para información que deba residir en almacenamiento o en los servidores durante un periodo largo de tiempo, de modo que, si nuestro sistema está compuesto de información volátil y fugaz, saldrá más caro el tiempo de cómputo y procesamiento de las llaves, en comparación del almacenamiento "ahorrado".

Otra de las situaciones en las que no es favorable la desduplicación, es cuando no existen tantos archivos con duplicación, y en su mayoría son únicos. Nuevamente nos lleva a desperdiciar tiempo en el hash, y en la verificación de la duplicación.

III. CONCLUSIÓN

Después de elaborar este documento, observamos que la desduplicación es un buen método para organizar memoria, ya que existen dos formas de hacer este proceso, aunque el problema es que a veces estos escasean en algunas cuestiones como en el caso de no haber suficiente espacio de memoria, no nos va a servir del todo, además de que podemos tener una sorpresa indeseada al momento de hacer la desduplicación, ya que a veces algunos tipos de archivos se ven afectados.

Continuando con la idea del párrafo anterior, creemos que a pesar de que la desduplicación es una gran forma de ahorrar almacenamiento, no debe usarse como único método, pues como vimos, existen múltiples desventajas en la desduplicación.

Para concluir, podemos decir que la desduplicación es un método que se puede utilizar en sistemas de producción ya que, al realizar esta investigación, encontramos varios casos particulares de algunas empresas que utilizan a la desduplicación para el ingreso de archivos. Un ejemplo de esto es Facebook, el cual utiliza la desduplicación para que las personas que ingresen archivos duplicados se les guarde el primer archivo que manda, y este se va verificando con las nuevas entradas del usuario (hacen uso de la desduplicación online). Además de otras empresas que usan la desduplicación para mantener sus servidores libres de saturaciones de archivos duplicados, liberando así espacio de almacenamiento, algunas de estas empresas son: IBM, Oracle, Wbsgo, netapp, etc.

IV. ÍNDICE DE FIGURAS

V. REFERENCIAS

[1] Wolf. G., Ruiz. E., Bergero. F., Meza E. (2015). *Fundamentos de sistemas operativos*. [Online]. Available: http://sistop.org/pdf/sistemas_operativos.pdf

[2] NetApp. (2019). *¿Qué es la deduplicación de datos?* [Online]. Available: https://www.netapp.com/es/data-management/what-is-data-deduplication/

[3] A. F. Mancheno, "*EVALUACIÓN DE LOS SISTEMAS DE DEDUPLICACIÓN OPENDEDUP (SDFS) Y ZFS PARA OPTIMIZAR EL SISTEMA DE ALMACENAMIENTO EN UN SERVIDOR DE BACKUPS*". Facultad de Informática y Electrónica. ESPC. Riobamba. Ecuador. 2015.

[4] Whitehouse. L. (2009). *Cuando y donde utilizar la tecnología de deduplicación de datos en el safeguard de disco*. [Online]. Available: https://searchdatacenter.techtarget.com/es/consejo/Cuando-y-como-utilizar-la-tecnologia-de-deduplicacion-de-datos-en-el-safeguard-de-disco

[5] Jiménez. F. (2009). *TÉCNICAS DE DEDUPLICACIÓN DE DATOS Y APLICACIÓN EN LIBRERÍAS VIRTUALES DE CINTAS*. [Online]. Available: https://oa.upm.es/1803/1/PFC_FRANCISCO_JAVIER_JIMENEZ_PATRICIO.pdf

[6] S. T. Ahmed, L. E. George. "*Lightweight hash-based de-duplication system using the self detection of most repeated patterns as chunks divisors*". Journal of King Saud University - Computer and Information Sciences. pp. 6-7. April 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1319157821000914

[7] Menezes. A., Van Oorschot. P., Vanstone S. (1996). *Hash Functions and Data Integrity*. [Online]. Available: https://cacr.uwaterloo.ca/hac/about/chap9.pdf

[8] IBM. (2016). Active Memory deduplication. [Online]. Available: https://www.ibm.com/docs/en/linux-on-systems?topic=linuxonibm/liabd/virtcon-active-memory-de-duplication.htm

[9] Dutch. M. (2008). Understanding data deduplication ratios. [Online]. Available: https://www.snia.org/sites/default/files/Understanding_Data_Deduplication_Ratios-20080718.pdf

Temas de interés relacionados a la desduplicación

- Oracle. (2004). 4.10. Memory Overcommitment. [Online]. Available: https://docs.oracle.com/en/virtualization/virtualbox/6.0/user/guestadd-memory-usage.html
- Facebook. (2021). información sobre la desduplicación de eventos offline. [Online]. Available: https://www.facebook.com/business/help/1772588746090250?id=565900110447546
- IBM. (1993). Deduplication. [Online]. Available: https://www.ibm.com/docs/en/spectrum-protect/8.1.12?topic=reference-deduplication
- Wbsgo. (2014). Deduplicacion en el backup de datos. [Online]. Available: https://www.whitebearsolutions.com/la-deduplicacion-en-el-backup/