

Optimización del Almacenamiento con Erasure Coding mediante Caché Inteligente para la Reducción de la Carga Computacional

Jhair Agila ¹ Gerardo Herrera ¹ Leonardo Orozco ¹ Kevin Sarango ¹

¹Universidad Nacional de Loja

Introducción

En el contexto actual de creciente demanda por almacenamiento eficiente y procesamiento ágil de datos, algunas tecnologías como Erasure Coding (EC) emergen pues son soluciones avanzadas frente a los esquemas tradicionales de replicación. EC permite dividir y distribuir fragmentos de datos con redundancia mínima, optimizando el uso de espacio sin comprometer la disponibilidad ni la resiliencia, lo que conlleva una carga computacional significativa, especialmente durante los procesos de codificación y recuperación de datos.

Este trabajo busca analizar cómo se da la optimización del almacenamiento a través de Erasure Coding, potenciado mediante el uso de una caché inteligente, permitiendo obtener el beneficio de la reducción de la carga computacional. A través del enfoque de Aprendizaje Basado en la Investigación (ABI), se ha recopilado literatura científica de las siguientes bases de datos: SCOPUS, IEEE Xplore y arXiv, esta información recopilada fue complementada con análisis bibliométrico en ResearchRabbit, lo que ha permitido identificar tendencias tecnológicas y evaluar el impacto de las distintas propuestas en el campo.

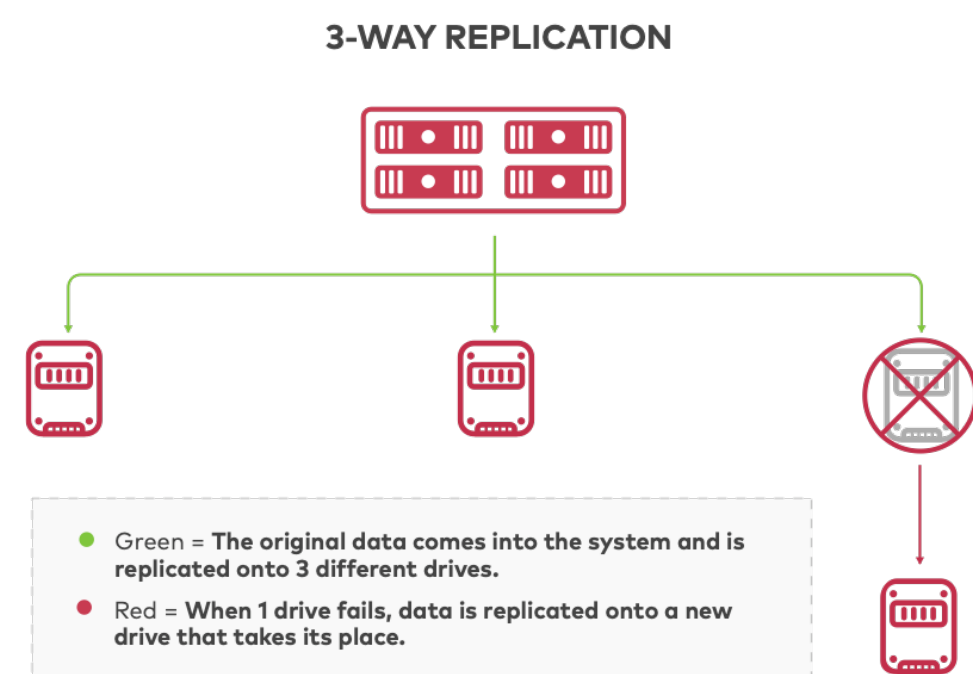


Figure 1. Proceso de codificación y recuperación de datos mediante Erasure Coding (5,3).

Objetivos

Objetivo general:

- Analizar como se da la optimización del almacenamiento con Erasure Coding mediante la caché inteligente para la reducción de carga computacional.

Objetivos específicos:

- Buscar información relacionada a Erasure Coding y a la optimización del almacenamiento usando una cache inteligente.
- Evaluar la mejora que presenta el uso de una cache inteligente para reducir la carga computacional usada por Erasure Coding.

Metodología

Para realizar este póster se aplico la metodología ABI (Aprendizaje Basado en la investigación), aquí se realizó cadenas de búsqueda refinadas en 3 bases de datos (SCOPUS, IEEE Xplore y Arxiv) para obtener información útil para el desarrollo del trabajo investigativo, esta información recolectada fue examinada cuidadosamente para determinar si su contenido aporta o no al proceso de investigación. Para lograr está búsqueda se hizo uso de palabras clave, filtros, criterios de inclusión y exclusión y tipos de documentos aceptados para su análisis en Zotero.

Ademas se realizo análisis bibliométrico en ReseachRabbit para el mapeo de documentos, donde se pudo identificar tendencias, evaluación de impacto y mapeo de la estructura intelectual del campo.

Resultados representativos

Erasure Coding (EC) se implementa en redes de entrega de contenido para mejorar el almacenamiento y disponibilidad. EC reduce la sobrecarga de almacenamiento en comparación con la replica tradicional, esto lo logra mediante el uso de fragmentos de paridad que junto con el algoritmo de Max Flow permite mejorar el balanceo de carga entre los servidores [4]. Se puede configurar EC para que aplique a objetos con contenido > 128 KB, evitando así sobrecarga, además esta técnica ayuda a:

- Reducir "miss ratio" causado por la in disponibilidad de servidores [4].
- Mejorar la durabilidad de SSDs [4].
- Bajo uso de overhead computacional, aunque se hace uso de un 14% más del uso de CPU, compensa el ancho de banda no usado y la eficiencia de almacenamiento [4].

Los resultados indican que La aplicación de Erasure Codign (EC) como una alternativa a la replicación puede resultar de gran ayuda pues ayuda a reducir el gasto por almacenmiento excesivo, de la misma forma . La referencia [2] explica que se puede mejorar el rendimiento del almacenamiento distribuido basado en EC mediante el uso de caché de matrices, esto permite reutilizar matrices en lugar de generar nuevas en los procesos de codificación y decodificación. Además para poder llevar a cabo con éxito ésta técnica se puede hacer uso de memoria caché con políticas avanzadas de remplazo, lo cual ayuda a reducir aún más la carga computacional [2].

Se han realizado varios experimentos usando la cache en EC, entre ellos está el uso de HDFS (Sistema de archivos distribuidos Hadoop) básico sin cache, HDFS con políticas de LRU, LFU, HDFS con políticas WRP y HDFS con propuesta WSCRП (Ponderación del tamaño y la política de reemplazo de costos) donde se toma como base 1000 operaciones para demostrar que usando WSCRП reduce significativamente el tiempo de escritura (10 s), lectura (23 s) y recuperación (30 s) en comparación con sistemas sin cache o uso de una cache básica como HDFS básico sin cache [2].

Otra técnica que podría usar en EC es el uso de caches multinivel, donde se explora la integración de codificación en la red en diversos dispositivos como estaciones base y usuarios cooperativos, esto ayuda a mejorar la eficacia del sistema, carga de la red y procesos computacionales [1].

En EC se puede hacer uso de diversos tipos de codificación como la Reed-Solomon vista en la figura 2, esta es usada en sistemas HDFS y Ceph, modelos formales como SLP (Programas de línea recta) para aplicar técnicas clásicas de optimización de compiladores, estas están enfocadas en reducir el almacenamiento y el número de operaciones [3]. Entre las estrategias apra la optimización de cache esta el uso de XorRePair, fusión de operaciones y el bloqueo y planificación de caché [3].

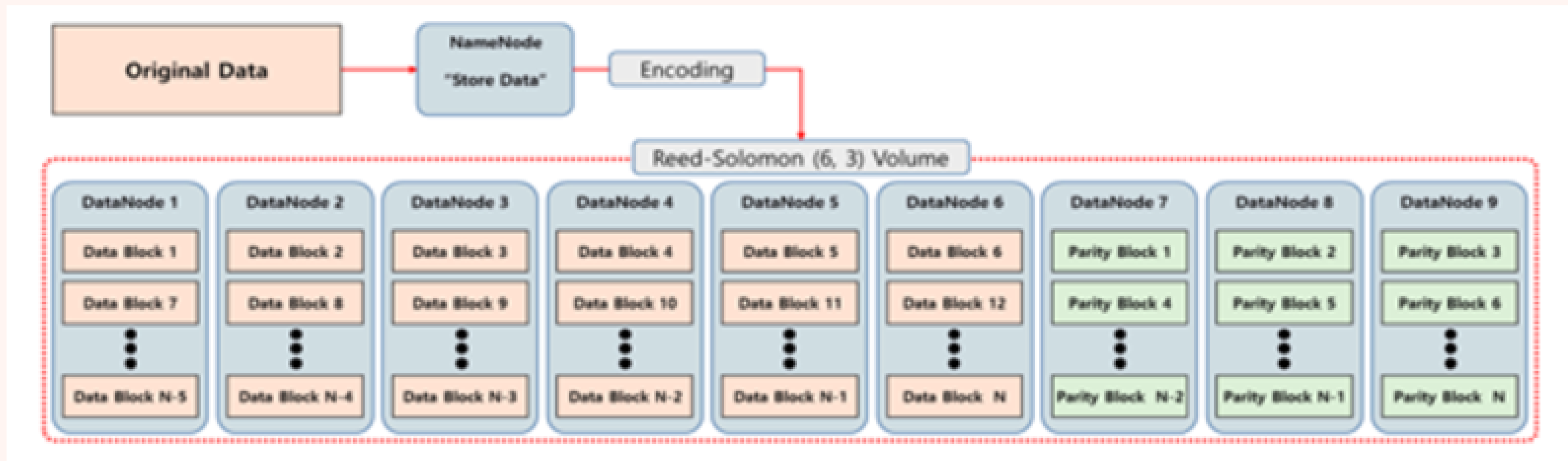


Figure 2. Ejemplo de aplicando Erasure-Coding [2].

Caso de aplicación

Para demostrar el funcionamiento de **Erasure Coding (EC)** y su impacto en la carga computacional, se utilizó **MinIO**, un sistema de almacenamiento distribuido que simula servicios similares a Amazon S3. Se trabajó con un archivo binario de 100 MB (**archivo_prueba.bin**) y cuatro directorios (**disco1, disco2, disco3, disco4**) que simulan discos físicos distribuidos.

MinIO generó automáticamente 2 bloques de datos y 2 de paridad, distribuyéndolos entre los cuatro discos. A continuación, se midió el tiempo de respuesta durante una solicitud de recuperación de datos sin uso de caché inteligente, revelando el consumo computacional inherente a EC. Posteriormente, se repitió la misma operación, aprovechando la memoria caché creada en la primera petición, lo que produjo mejoras evidentes en el rendimiento y los tiempos de acceso, esto se puede observar en la figura 3.

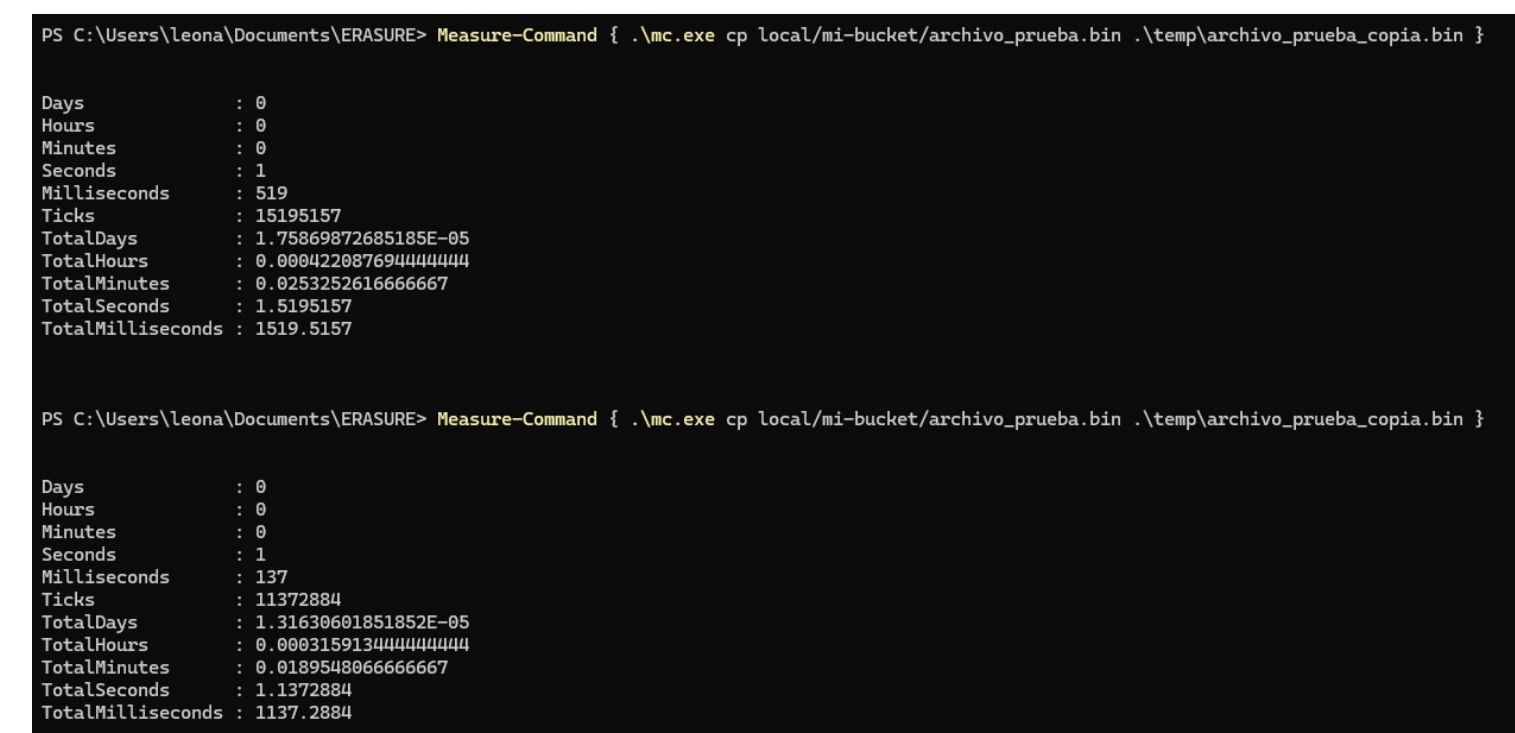


Figure 3. Simulación con MinIO de Erasure Coding sobre discos distribuidos virtuales.

Conclusiones

- Erasure Coding** se presenta como una solución efectiva frente a los altos costos que implica la replicación tradicional, reduciendo considerablemente el uso de espacio gracias a la fragmentación de datos y el uso de paridad.
- La integración de **caché inteligente** en sistemas basados en EC ofrece mejoras sustanciales en el rendimiento computacional, permitiendo reducir operaciones redundantes y tiempos de acceso mediante técnicas de reutilización y gestión avanzada de memoria.
- La aplicación de estrategias como **WSCRП** y el uso de cachés multinivel refuerza la eficacia de los sistemas distribuidos, mejorando el balance de carga, la escalabilidad y la velocidad de recuperación de datos. Estas técnicas respaldan el diseño de infraestructuras más sostenibles y ágiles.

Referencias

- Mohammed S. Al-Abiad, Md. Zoheb Hassan, and Md. Jahangir Hossain. Throughput maximization of network-coded and multi-level cache-enabled heterogeneous network. *IEEE Transactions on Vehicular Technology*, 70(10):11039–11043, 2021. doi: 10.1109/TVT.2021.3106148.
- Dong-Jin Shin and Jeong-Joon Kim. Cache-based matrix technology for efficient write and recovery in erasure coding distributed file systems. *Symmetry*, 15(4), 2023. doi: 10.3390/sym15040872. URL <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85156261849&doi=10.3390%2fsym15040872&partnerID=40&md5=6e70809d9b556eff99eabae826fa89>. Cited by: 3; All Open Access, Gold Open Access.
- Yuya Uezato. Accelerating xor-based erasure coding using program optimization techniques. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '21, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450384421. doi: 10.1145/3458817.3476204. URL <https://doi.org/10.1145/3458817.3476204>.
- Juncheng Yang, Anirudh Sabnis, Daniel S. Berger, K.V. Rashmi, and Ramesh K. Sitaraman. C2dn: How to harness erasure codes at the edge for efficient content delivery. page 1159 – 1177, 2022. URL <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85140992024&partnerID=40&md5=997dea7058f8c54a4704e3af41252ab9>. Cited by: 26.