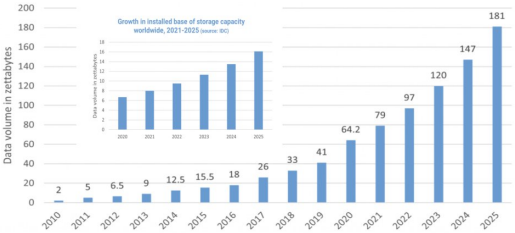
On the Energy Footprint of Erasure Codes in Ceph

Marc Tranzer and Shadi Ibrahim

Inria, Univ. Rennes, CNRS, IRISA, Rennes, France

This research is supported by the FrugalCloud collaboration between Inria and OVHCloud.

Context



Storage costs and energy consumption increase exponentially

- Account for almost 11% of the total energy consumed in a datacenter.
- 43% of the server cost alone [2].

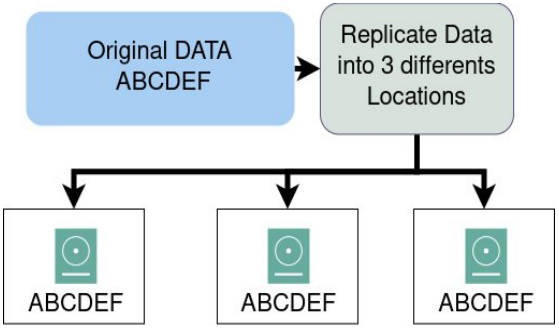
Data storage requires high availability and fault tolerance

1 Source: IDC "Global DataSphere Forecast, 2021-2025"
2 A. Shehabi, et al, "United States Data Center Energy Usage Report", LBNL, Berkeley, California. LBNL-1005775

Replication

+ Data available locally
+ Fast And Simple Repair Flow

- High Storage overhead
- Amplifies the storage and energy problem



Context: Erasure coding

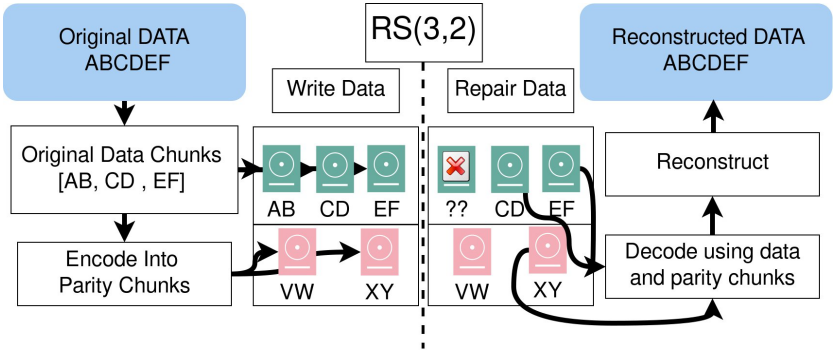


Figure: Write and Repair data under Erasure Coding

+ **Space Efficient:** 45% less space than 3-Way Replication.
  ==> Drastically reduces the data-at-rest energy cost
+ **Fast Encoding :** Thanks to ISA-L library and modern CPUs

Motivation

Understand the challenges of building a highly available and energy-efficient distributed storage system

We run a complete evaluation of replication and erasure coding inside a Ceph File System cluster with 18 storage nodes equipped with wattmeters

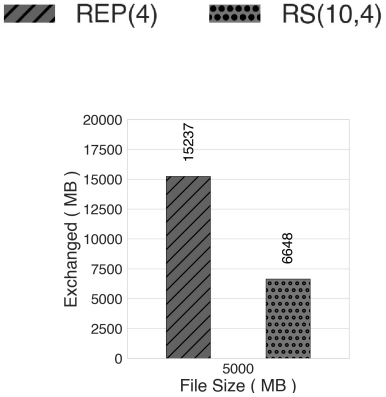
Single Write

Writing 1 file of 5GB

In CephFS:

1. Data block is transfer to one cluster node
2. The cluster node then sends
    - 3 replica of data under REP4
    - 13 chunks under RS(10,4)

==> 60% Less data exchanged under EC



Data Exchange inside the cluster

Single Write

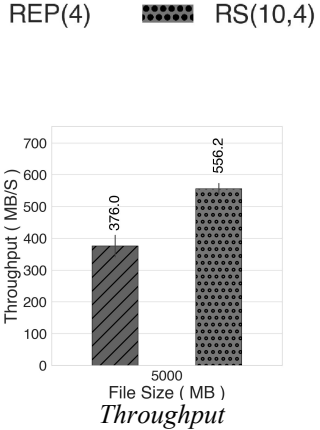
Writing 1 file of 5GB

In CephFS:

1. Data block is transfer to one cluster node
2. The cluster node then sends
    - 3 replica of data under REP4
    - 13 chunks under RS(10,4)

==> 60% Less data exchanged under EC

==> 48% Higher throughput



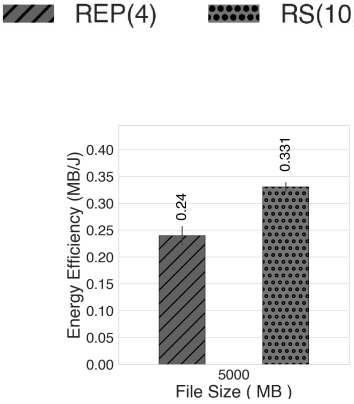
Throughput

Single Write

Writing 1 file of 5GB

Energy is not proportional to performance

- **10% Higher power under EC due to the encoding**
- **37% Higher energy efficiency**



Energy Efficiency

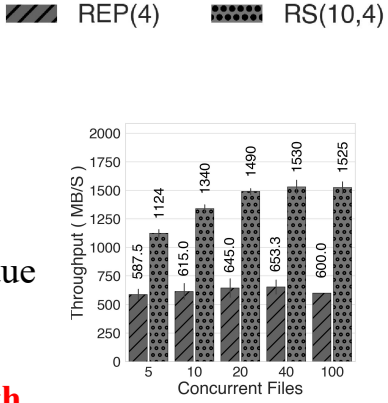
Parallel Write

Writing 5-100 files of 1GB with 5 clients

- **1 write: EC is 55% faster than REP**
- **100 writes: EC is 154% faster than REP**

Replication suffers from high disk contention due to write imbalance and high storage overhead

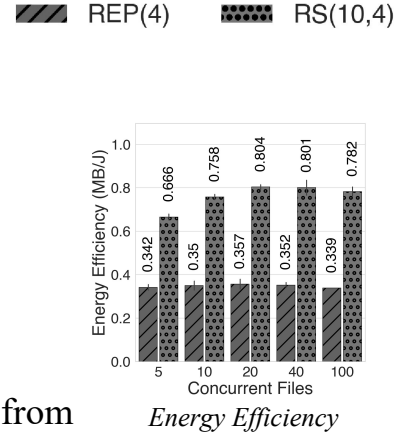
- **Affect replication performance under high concurrency**



Throughput

Parallel Write

Writing 5-100 files of 1GB with 5 clients



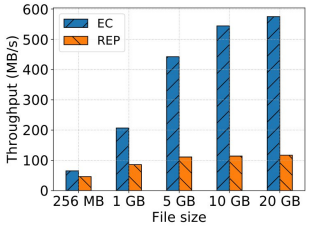
Energy Efficiency

EC energy efficiency increases from 47% to 124% compared to replication

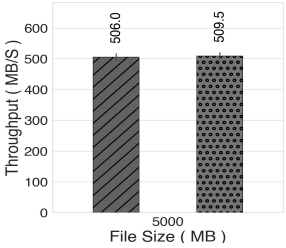
- EC energy efficiency increase with better performance
- Energy impact of network transfer increase from 4% to 7% of the total energy consumed under replication

Single Read

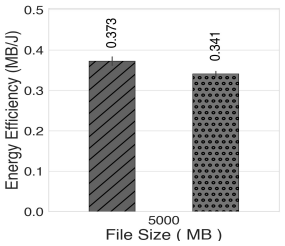
Reading 1 file of 5GB



Throughput with HDFS



Throughput with CephFS



Energy efficiency with CephFS

Observation: EC has a similar throughput but worse energy consumption than REP

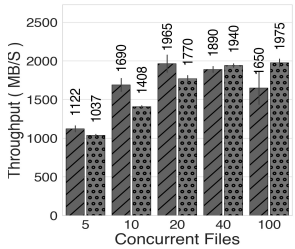
Explanation: Ceph retrieval involves two steps:

1) Retrieving the chunk from the other OSDs, **which results in high network traffic**
2) Sending the block sequentially to the client, **which results in low parallelization**

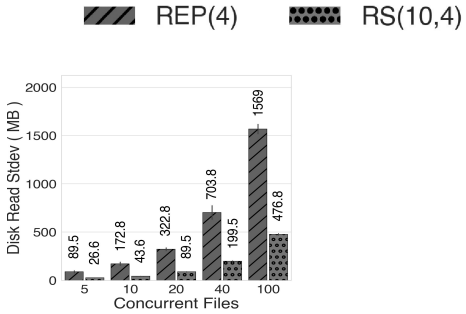
[1] Jad Darrous and Shadi Ibrahim "Understanding the performance of erasure codes in hadoop distributed file system", in CHEOPS '22

Parallel Read

Reading 5-100 files of 1GB



Throughput

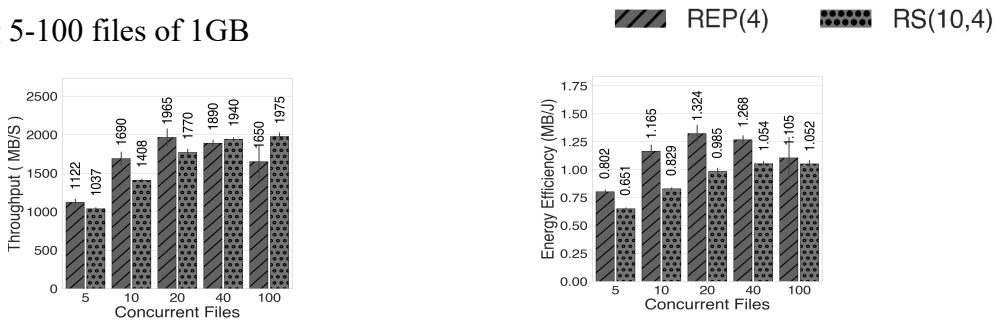


Standard Deviation of Read Amount Per OSD

Observation: EC performs **better at high concurrency** but **worse at low concurrency**
Concurrency: Replication can read from multiple disks instead of one
High concurrency: EC has good read balance which reduce the disk contention

Parallel Read

Reading 5-100 files of 1GB



Throughput

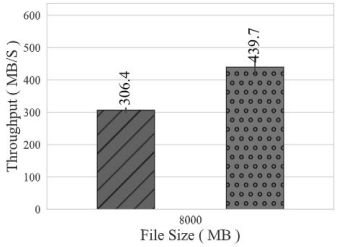
Energy Efficiency

Observation: Higher EC throughput doesn't lead to higher energy efficiency

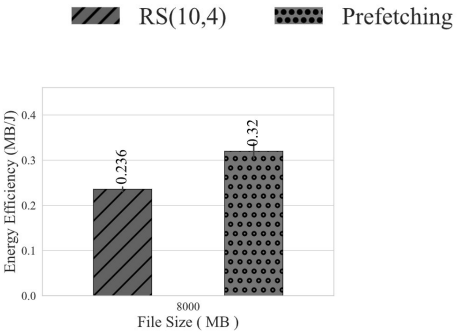
- **EC doesn't have local access to the data**
- **1 file: Network consumes 2.7% of read energy under EC**
- **100 files: Network consumes 13.5% of read energy under EC**

Prefetching Erasure Coded Blocks

Reading 1 file of 5GB (500Mbps network)



Throughput

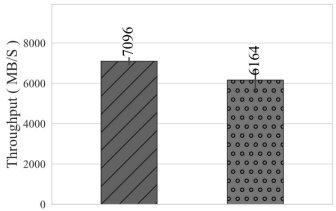


Energy Efficiency

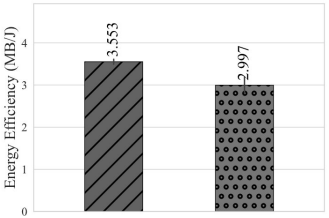
Naive Solution: Prefetching the blocks into the storage node is a common solution to accelerate the retrieval

Prefetching Erasure Coded Blocks

Reading 20 file of 5GB (10Gbps network)



Throughput (20clients)



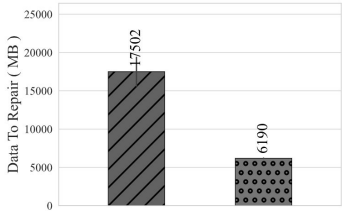
Energy Efficiency (20clients)

Problem At Scale:

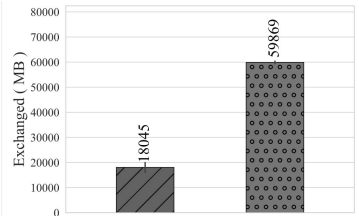
- Increases contention when the cluster is heavily used
- Consumes extra energy by storing the prefetched block in RAM

Node Repair

Removing one node after storing an 80GB dataset



Amount of data to repair



Exchanged data during the repair

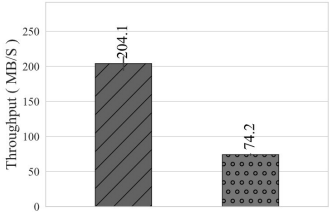
Low Storage overhead from EC: Less data to repair

BUT

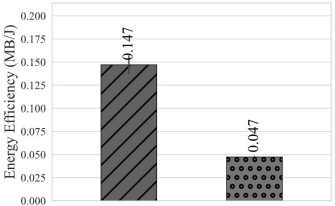
Repairing **400 KB** chunk: RS(10,4) needs to reconstruct the entire **4 MB** block

Node Repair

Removing one node after storing an 80GB dataset



Repair Throughput



Energy Efficiency

Leads to long and energy inefficient repairs

Motivate the research for repair optimized network families

Conclusion and Contributions

Conclusion:

+ Erasure coding can be used not only for archival data but also for warm and hot scenario

- However it exhibits higher energy tradeoffs and high data transfer during read and repair

Contributions and Future Works:

- A complete study with more scenario and including other EC families in Ceph
- Design a scalable and effective data prefetching for Erasure-Coded Storage
- Improve the repair workload under warm and hot data access pattern

Thank you for your attention!

Feel free to discuss with me about the whole experiment and our insights

Global challenges on building energy efficient EC codes

Contradicting Design Goals

High locality: Less network traffic **but** Poor read/write balance
Network-aware retrieval: Less network traffic **but** higher decoding cost
EC at Cluster level: Control the encoding performance **but** poor read performance

No perfect solution

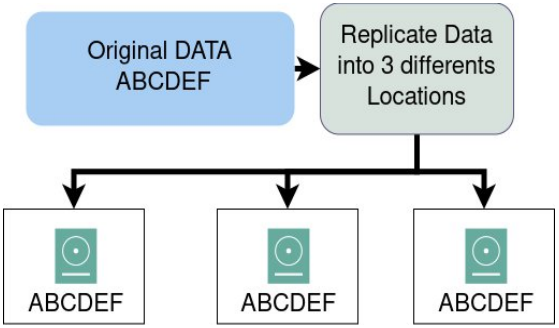
RS code: Good storage overhead and fast encoding **but** high repair overhead
Clay code: Reduced repair overhead **but** high encoding cost
LRC code: Good repair locality and fast encoding **but** reduced storage overhead

Context Replication At Scale

Facebook 2013 Report: 20 non-transient node failures every day on a 3000 nodes cluster [1]

==> Requires multiple replicas
==> High Storage overhead

Amplifies the storage and energy problem



[1] Maheswaran Sathiamoorthy, et al "XORing Elephants: Novel Erasure Codes for Big Data

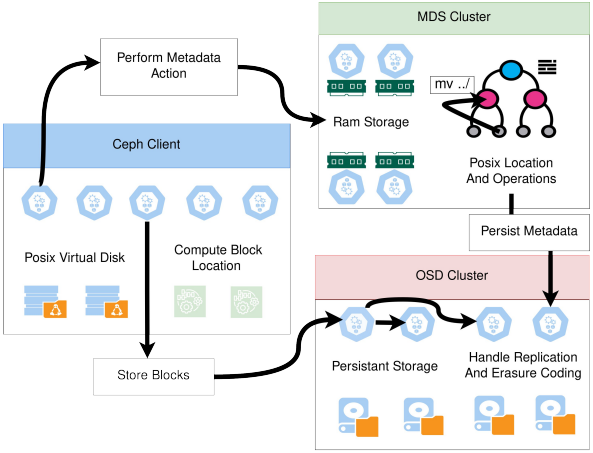
Evaluated File System

Run our evaluation in CephFS

- Highly adopted file system
- Used by IBM, OVH ...

All Erasure Coding operations are done in storage (OSD) cluster

- Differs from Hadoop
- Similar to F4 and Azure



Experimental Setup

- **Grid'5000 [1] Testbed:**
  - 18 OSDs and 1-5 clients with HDD

- **Reed Solomon (10, 4):** Production Erasure Code

- Energy of nodes measured with high frequency wattmeter
- Network traffic measured with *psutils* and corresponding energy calculated based on models from [2]

1 Daniel Balouek, et al, "Adding virtualization capabilities to the Grid'5000 testbed." In Cloud Computing and Services Science, 2013.
2 Loic Guegan and Anne-Cécile Orgerie. "Estimating the End-to-End Energy Consumption of Low-Bandwidth IoT Applications for WiFi Devices

Research focus to build energy efficient writes

Minimize amount of data written to disks and transferred across nodes

- Minimize storage overhead and data transfer during write

Improve energy efficiency of encoding

- Develop energy-efficient EC encoding algorithms
- Build energy-optimized encoding nodes and supporting infrastructure

Ensure balanced disk utilization

- Design a balance-aware data placement algorithm
- Develop EC schemes that naturally distribute write load evenly across disks

Research focus to build energy efficient reads

Minimize EC chunks network access cost

- Optimize EC chunk placement to maximize rack locality
- Select EC chunks for retrieval based on their network energy cost

Ensure balanced read across disks

- Design EC schemes that naturally distribute read operations across disks
- Build imbalance-aware chunk retrieval algorithms

Reduce computational cost

- Develop energy-efficient EC decoding algorithms
- Favor retrieving original data chunks when available

Research focus to build energy efficient repairs

Reduce the amount of data to repair

- Build storage-space efficient EC schemes

Low energy spend on decoding

Reduce the amount of data to read and transfer to perform repair

- RG code where design to that end but have high encoding/decoding cost

Reduce the data transfer across racks to perform repair

- LRC and SHEC codes allows high repair locality but requires more storage
- Build repair pipeline that perform partial repairs within the rack

Context: Erasure coding

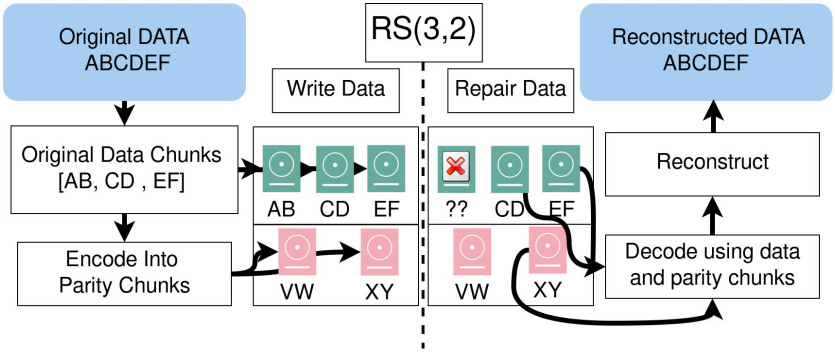


Figure: Write and Repair data under Erasure Coding

- **Data is not available locally**
- **High network usage during repair**