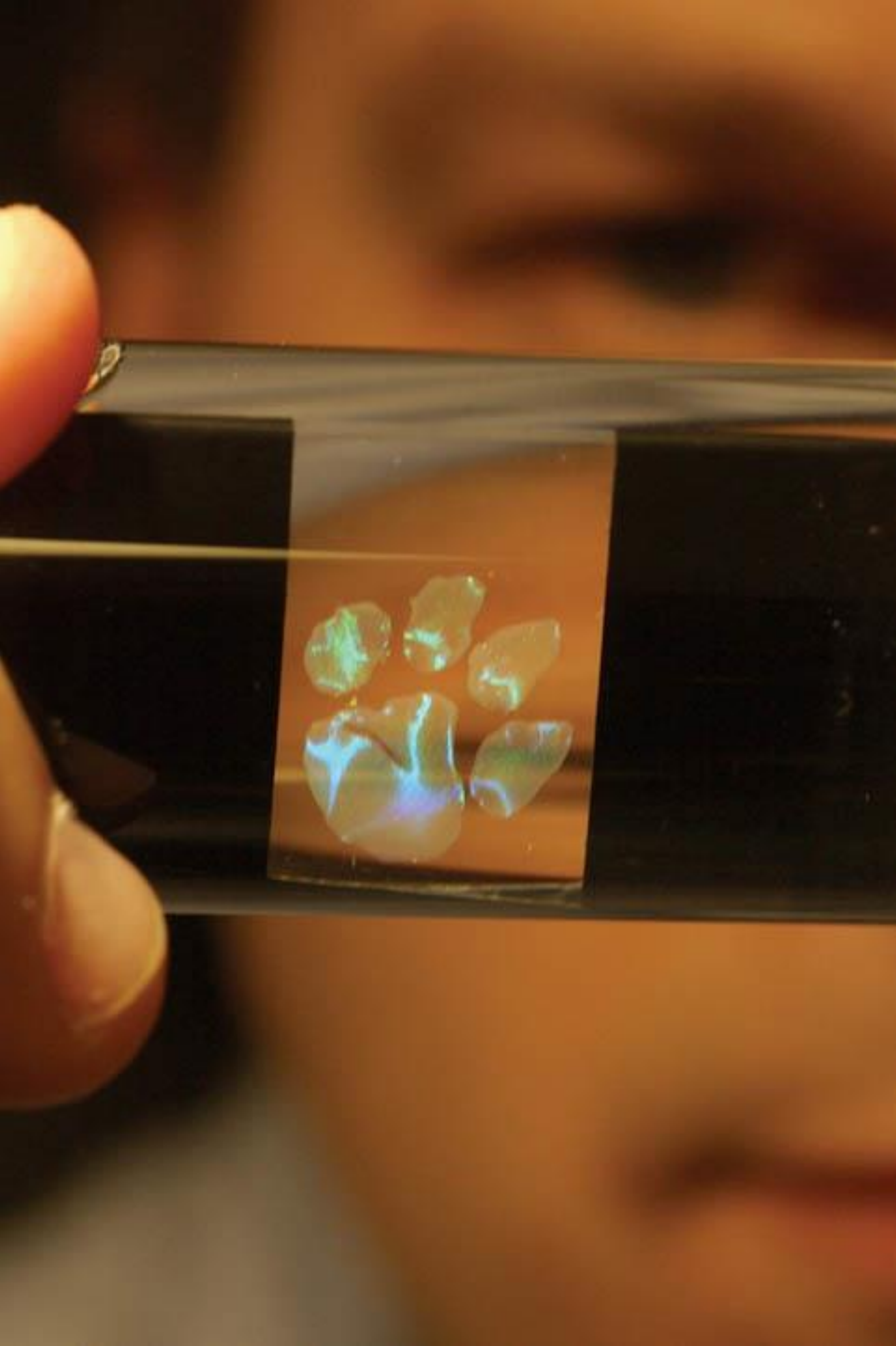


A Popularity-aware Cost-effective Replication Scheme for High Data Durability in Cloud Storage

Jinwei Liu* and Haiying Shen†

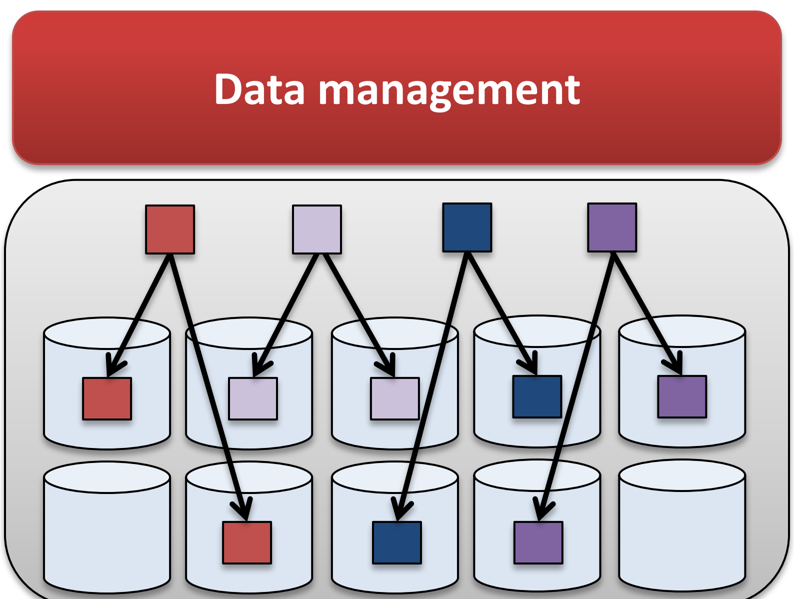
*Dept. of Electrical and Computer Engineering
Clemson University, SC, USA
†Dept. of Computer Science, University of Virginia, Charlottesville, VA, USA

Outline

- Introduction
- Popularity-aware multi-failure resilient and cost-effective replication (PMCR)
- Design of PMCR
- Performance Evaluation
- Conclusions

Introduction

- Data management in cloud storage



amazon web services S3

Google CLOUD STORAGE

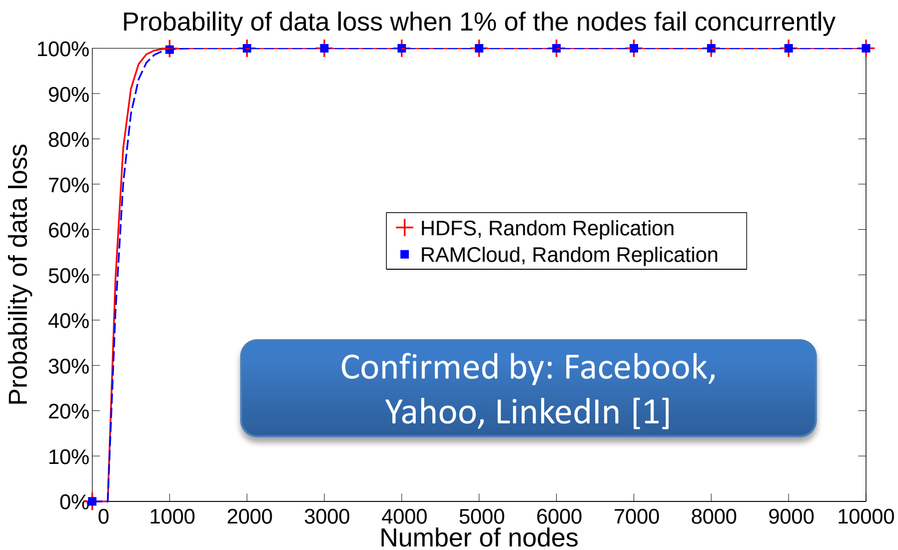
Windows Azure

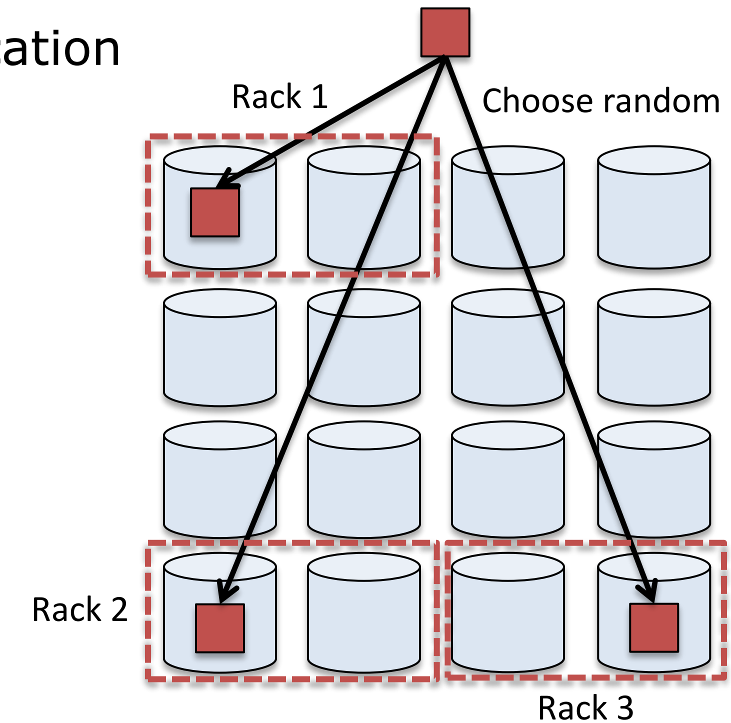
Motivation

- Data loss and machine failures in emerging cloud systems
  - Non-correlated machine failures
    - Multiple machines fail concurrently
  - Correlated machine failures
    - Machines fail individually
      - Power outages
        - 1-2 times a year [Google, LinkedIn, Yahoo]
      - Large scale network failures
        - 5-10 times a year [Google, LinkedIn]
      - And more
        - Rolling software/hardware updates
- Design principle
  - Multi-failure resilient replication scheme

Motivation (cont.)

- Random replication
  - Prob. of data loss in random replication



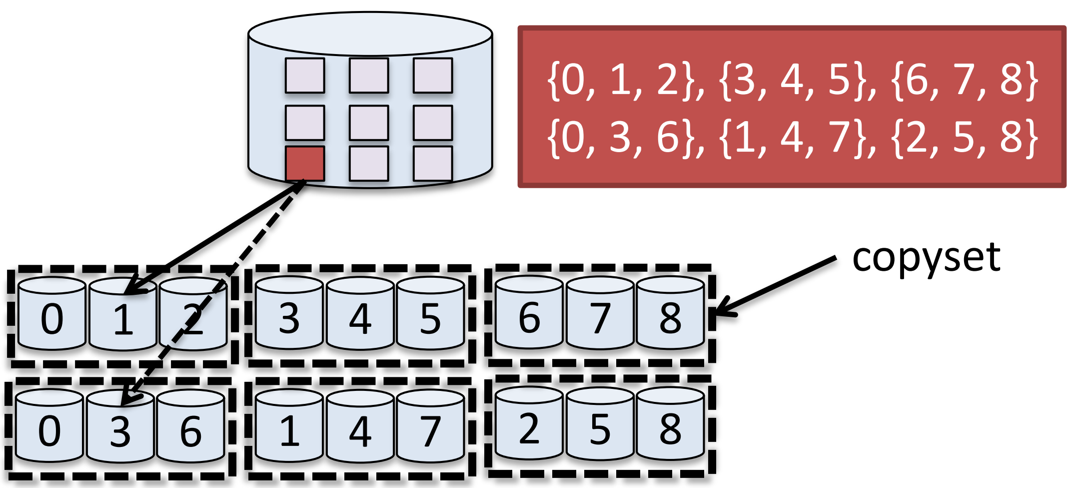


HDFS, GFS, Windows Azure, RAMCloud

[1] A. Cidon, S. Rumble, R. Stutsman, S. Katti, J. Ousterhout, and M. Rosenblum. Copysets: Reducing the frequency of data loss in cloud storage. In *Proc. of ATC*, 2013.

Motivation (cont.)

- Limitation of existing approaches
  - Random Replication
    - High data loss probability, high storage cost and bandwidth cost
  - Copyset Replication & Tiered Replication
    - High storage cost and bandwidth cost

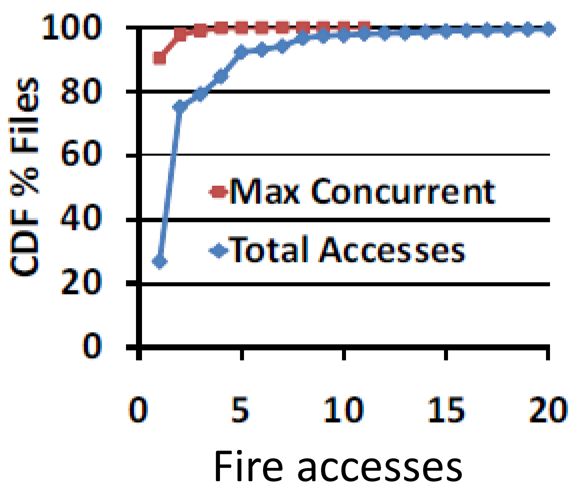


Scatter width (S): # of possible nodes storing the secondary replicas of a chunk

- Design principle
  - Cost-effective replication scheme

Motivation (cont.)

- Data popularity existing in cloud storage systems [2-3]
  - File popularity
    - CDFs of the total # of jobs that access each file and the # of concurrent accesses [2]



- Design principle
  - Popularity-aware replication

[2] G. Ananthanarayanan, S. Agarwal, S. Kandula, A. Greenberg, I. Stoica, D. Harlan, and E. Harris. Scarlett: Coping with skewed content popularity in mapreduce clusters. In *Proc. of EuroSys*, 2011.

[3] A. Khandelwal, R. Agarwal, and I. Stoica. BlowFish: Dynamic Storage-Performance Tradeoff in Data Stores. In *Proc. of NSDI*, 2016.

Outline

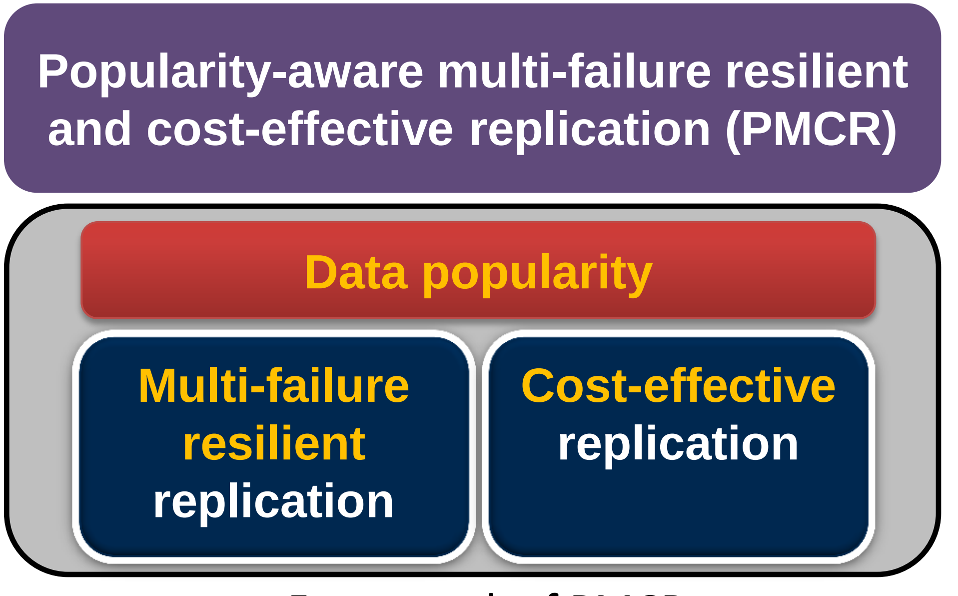
- Introduction
- Popularity-aware multi-failure resilient and cost-effective replication (PMCR)
- Design of PMCR
- Performance Evaluation
- Conclusions

PMCR

- Problem statement
  - Replicate the chunks of data objects so that the request failure probability, storage cost and bandwidth cost are minimized in both correlated failures and non-correlated failures
- Goal
  - Design a popularity-aware replication scheme for achieving high data durability while reducing storage cost and bandwidth cost caused by replication

Proposed Solution

- PMCR: Popularity-aware multi-failure resilient and cost-effective replication
  - Features of PMCR
    - Popularity awareness
    - Multi-failure resilience
    - Cost-effectiveness



Framework of PMCR

Challenges

- Challenges of PMCR design
  - How to significantly reduce data loss probability in both correlated and non-correlated machine failures
  - How to leverage data popularity to reduce cost (storage cost and bandwidth cost) caused by replication without compromising data durability and availability
  - How to determine popularity of data objects
  - How to effectively perform data compression and deduplication for both read-intensive and write-intensive data

Outline

- Introduction
- Popularity-aware multi-failure resilient and cost-effective replication (PMCR)
- Design of PMCR
- Performance Evaluation
- Conclusions

Design of PMCR

- Reduce data loss probability
  - BIBD-based method with data popularity consideration; replicates the first two replicas of each data chunk in primary tier, the third replica in remote backup tier; the three replicas of each data chunk are stored in one FTS
- Reduce cost
  - Compress the third replicas of warm data and cold data in the backup tier
    - For read-intensive data, PMCR uses the Similar Compression (SC); for write-intensive data, PMCR uses the Delta Compression (DC), which records the differences of similar data objects and between sequential data updates
  - Choose storage mediums for data objects based on data popularity

Data Classification

- Determining data popularity value
  - The Popularity φ_i of a data object (d_i) is measured by its visit frequency (denoted by v_i), i.e., # of visits in a time epoch (say epoch t)

$$\varphi_i = \alpha v_i$$

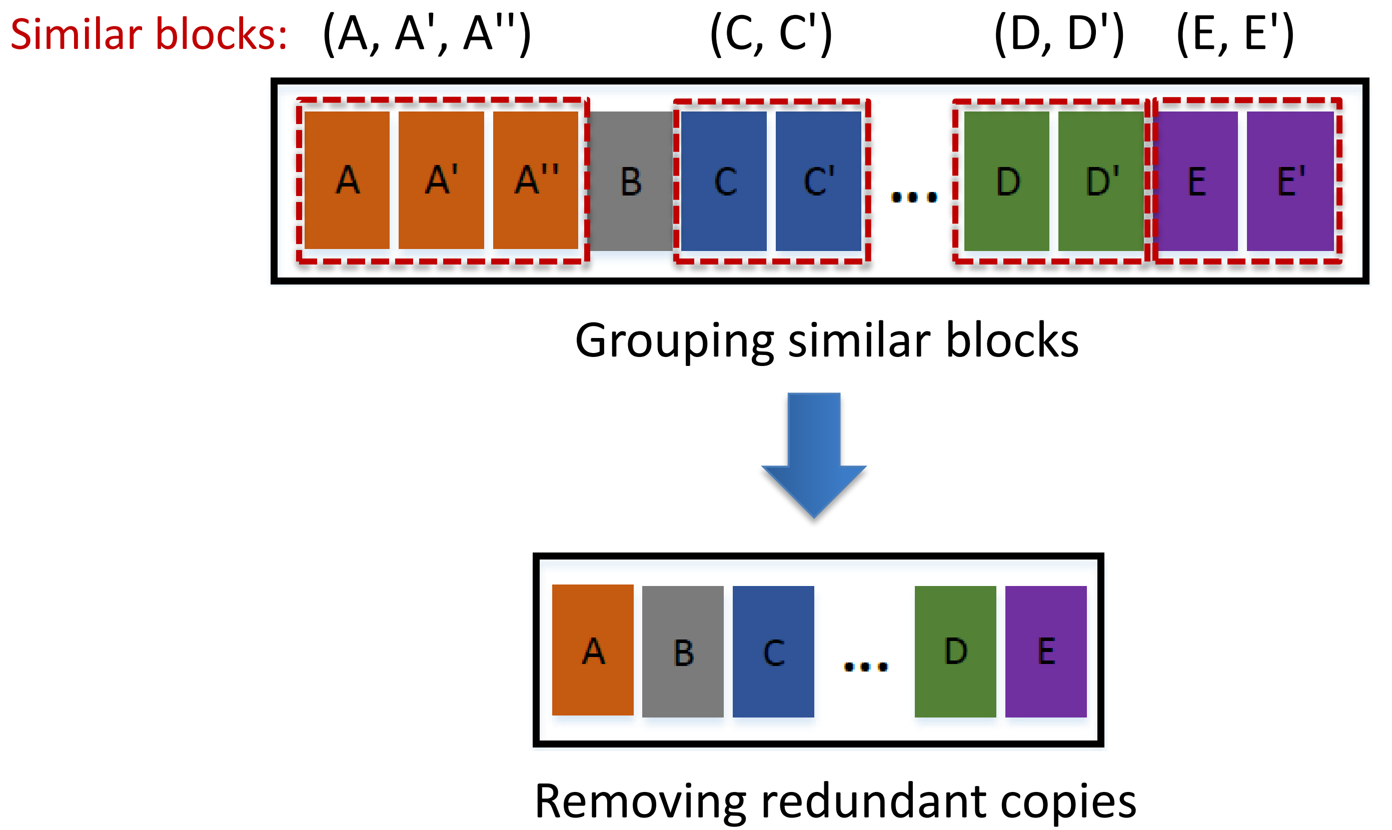
  - where α is a coefficient. The popularity at epoch t+1 is

$$\varphi_i^{t+1}(\cdot) = \beta\varphi_i^t + \alpha v_i$$

  - where β $(0 < \beta < 1)$ is a coefficient
- Determine popularity type
  - Calculate the popularity of each data object; rank them based on their popularity values
  - Hot data: popularity rank within top 25%
  - Warm data: popularity rank in (25%, 50%]
  - Cold data: popularity rank in (50%, 100%]
  -

Similar Compression (SC)

- SC for reducing cost

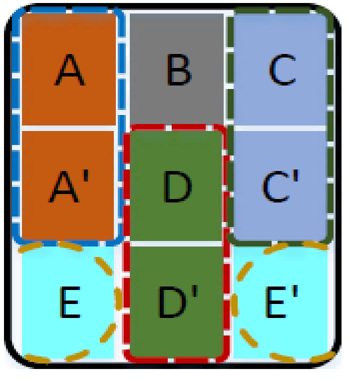


Similar Compression (cont.)

- Extending SC for reducing cost

Similar blocks within a file:

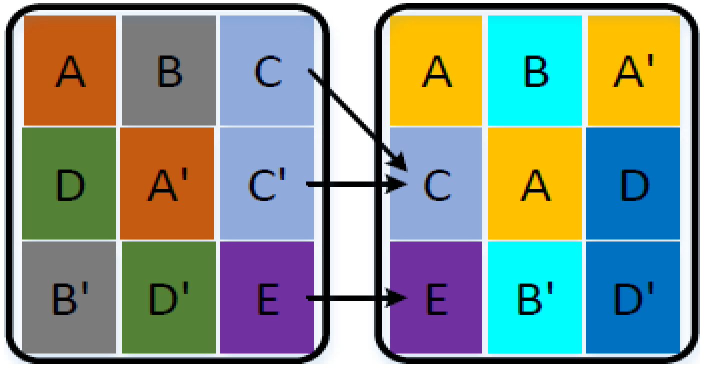
(A, A') (C, C') (D, D')(E, E')



Intra-file similarity

Similar blocks b/w two files

(C, C) (C', C) (E, E')



Inter-file similarity

Similarity Detection

- Bloom filter for similarity detection
  - PMCR uses the Bloom filter to detect similarity b/w data blocks and extends this algorithm for detecting similarity b/w data chunks
  - The chunks can be uniquely identified by SHA-1 hash signature (i.e., fingerprint). As the amount of data increases, more fingerprints need to be generated, which consume more storage space and incur more time overhead for index searching
  - To overcome the scalability of fingerprint-index search, PMCR groups a certain number of chunks into a block, and detects the similarity between blocks
  - The blocks with percentage of common 1s higher than a certain threshold are considered as similar blocks

Outline

- Introduction
- Popularity-aware multi-failure resilient and cost-effective replication (PMCR)
- Design of PMCR
- Performance Evaluation
- Conclusions

Performance Evaluation

- Methods for comparison
  - Random replication (RR)

    Choose secondary replica holders from a window of nodes around the primary node based on Facebook's design

  - Copyset Replication (Copyset) [1]

    [1] A. Cidon, S. Rumble, R. Stutsman, S. Katti, J. Ousterhout, and M. Rosenblum. Copysets: Reducing the frequency of data loss in cloud storage. In *Proc. of ATC*, 2013.

  - Tiered Replication (TR) [4]

    [4] A. Cidon, R. Escriva, S. Katti, M. Rosenblum, and E. G. Sirer. Tiered replication: A cost-effective alternative to full cluster geo-replication. In *Proc. of ATC*, 2015.

  - WAN Optimized Replication (WOR) [5]

    [5] P. Shilane, M. Huang, G. Wallace, and W. Hsu. WAN optimized replication of backup datasets using stream-informed delta compression. In *Proc. of FAST*, 2014.

Experiment Setup

- Set parameters in Facebook and HDFS environments

Parameters from publicly available data [1]

System	Chunks per node	Cluster size	Scatter width
Facebook	10000	1000-5000	10
HDFS	10000	100-10000	200

- Distribution of the file popularity and the updates follow those of FIU trace

- 7 simulated data centers

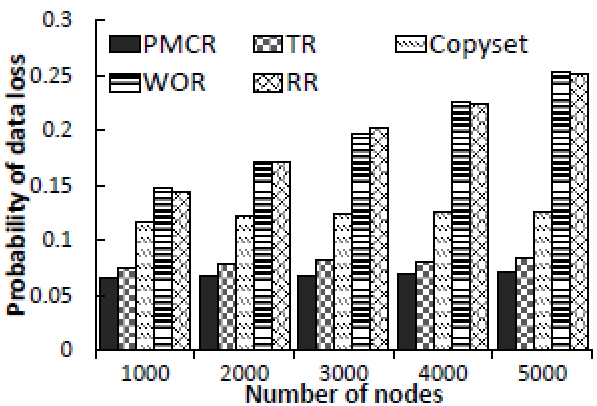
Experiment Setup (cont.)

- Parameter settings

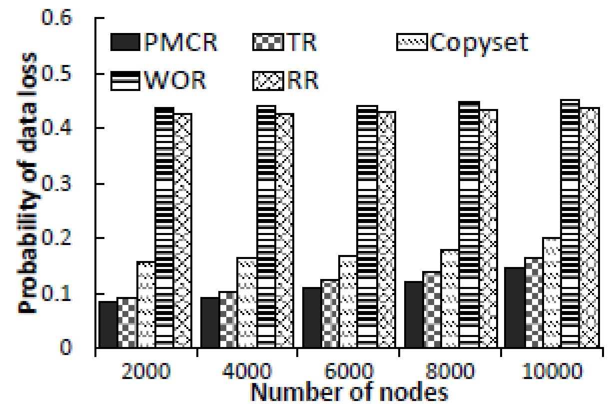
Parameter	Meaning	Setting
N	# of servers	1000-10000
M	# of chunks of a data object	50
R	# of servers in each FTS	3
λ	# of FTSs containing a pair of servers	1
S	Scatter width	4
p	Prob. of a server failure	0.5
m	# of data objects	10000-50000

Evaluation (cont.)

- Probability of data loss



(a) Facebook

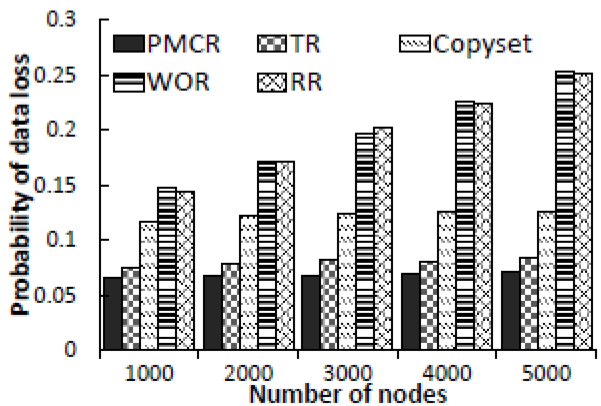


(b) HDFS

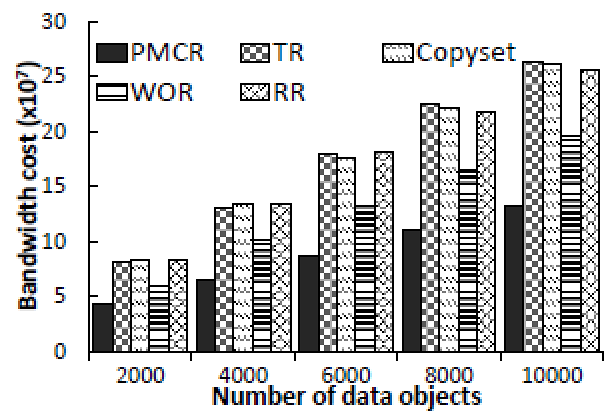
Result: PMCR < TR < Copyset < RR ≈ WOR

Evaluation (cont.)

- Bandwidth cost



(a) Facebook

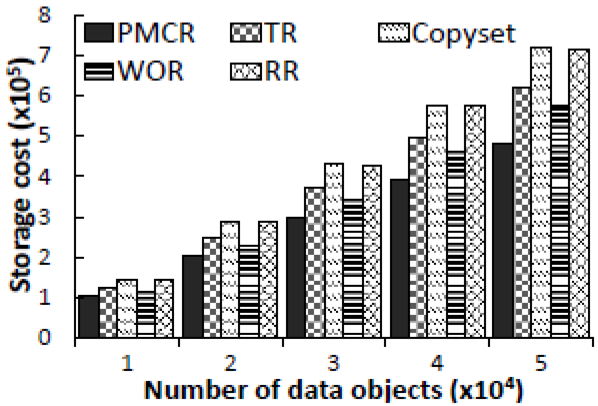


(b) HDFS

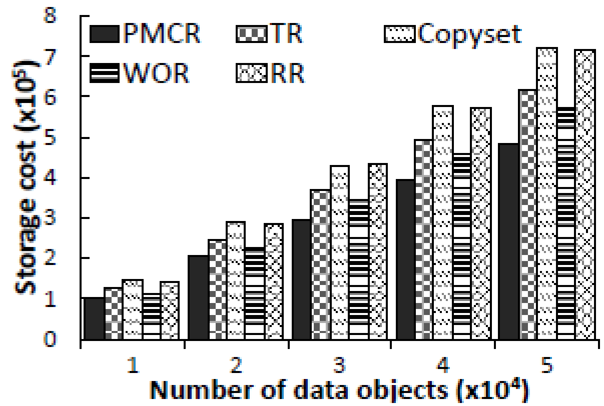
Result: PMCR < WOR < TR ≈ Copyset ≈ RR

Evaluation (cont.)

- Storage cost



(a) Facebook

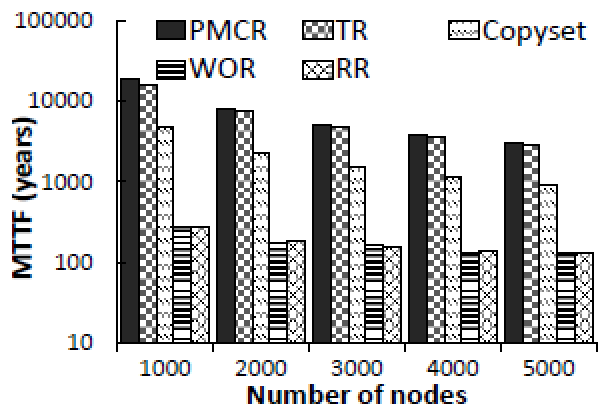


(b) HDFS

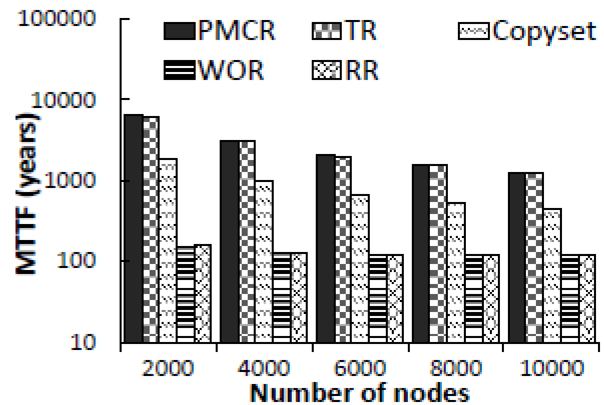
Result: PMCR < WOR < TR < Copyset ≈ RR

Evaluation (cont.)

- Mean time to failure (MTTF)



(a) Facebook



(b) HDFS

Result: PMCR ≈ TR > Copyset > RR ≈ WOR

Outline

- Introduction
- Popularity-aware multi-failure resilient and cost-effective replication (PMCR)
- Design of PMCR
- Performance Evaluation
- Conclusions

Conclusions

- Our contributions
  - PMCR restricts replicas of a data chunk into an FTS and puts the first two replicas in primary tier and the third replica in backup tier, which reduces data loss probability
  - PMCR classifies data into hot data, warm data and cold data, and selectively compresses the third replicas in backup tier to reduce costs; PMCR uses different storage mediums for data objects based on data popularity to further reduce storage cost
  - PMCR enhances SC by eliminating redundant chunks between different data objects
  - Conduct extensive trace-driven experiments to compare PMCR with other state-of-the-art replication schemes

- Future work
  - Consider network failures
  - Node joining and node leaving
  - Power consumption of machines

Thank you!

Questions & Comments?

Jinwei Liu, PhD

jinweil@clemson.edu

Electrical and Computer Engineering

Clemson University