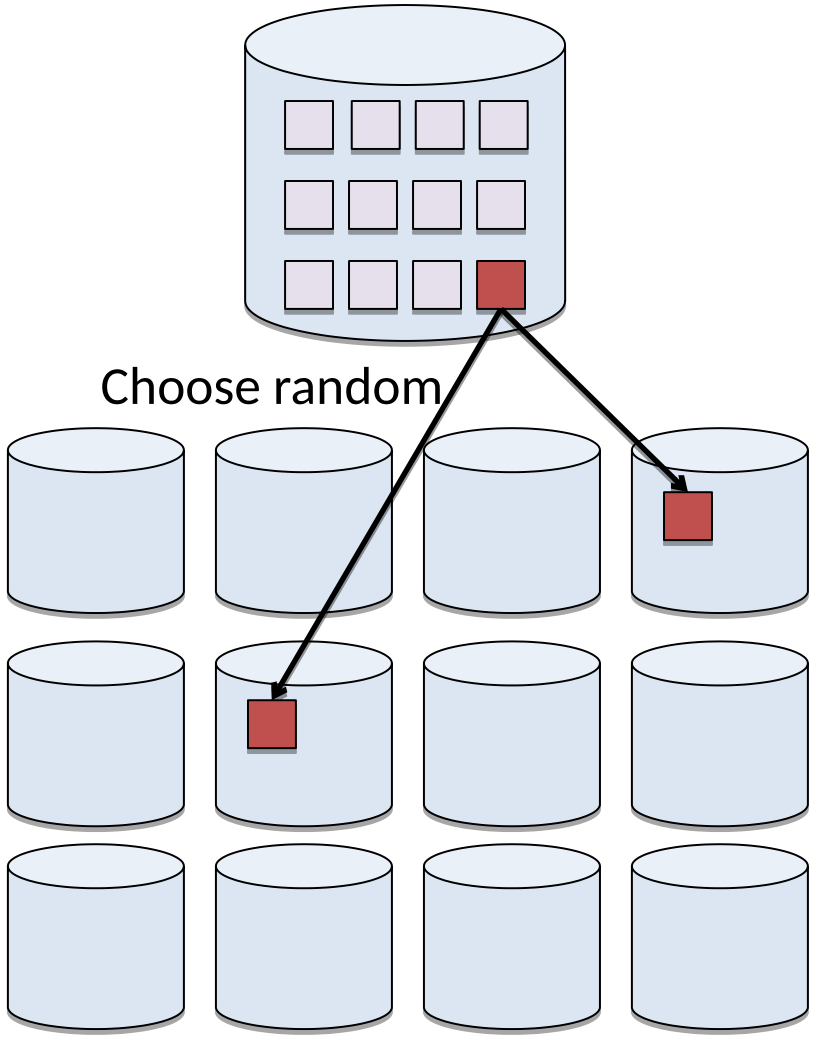
Copysets: Reducing the Frequency of Data Loss in Cloud Storage

Asaf Cidon, Stephen M. Rumble, Ryan Stutsman,
Sachin Katti, John Ousterhout and Mendel Rosenblum

Stanford University

Goal: Tolerate Node Failures



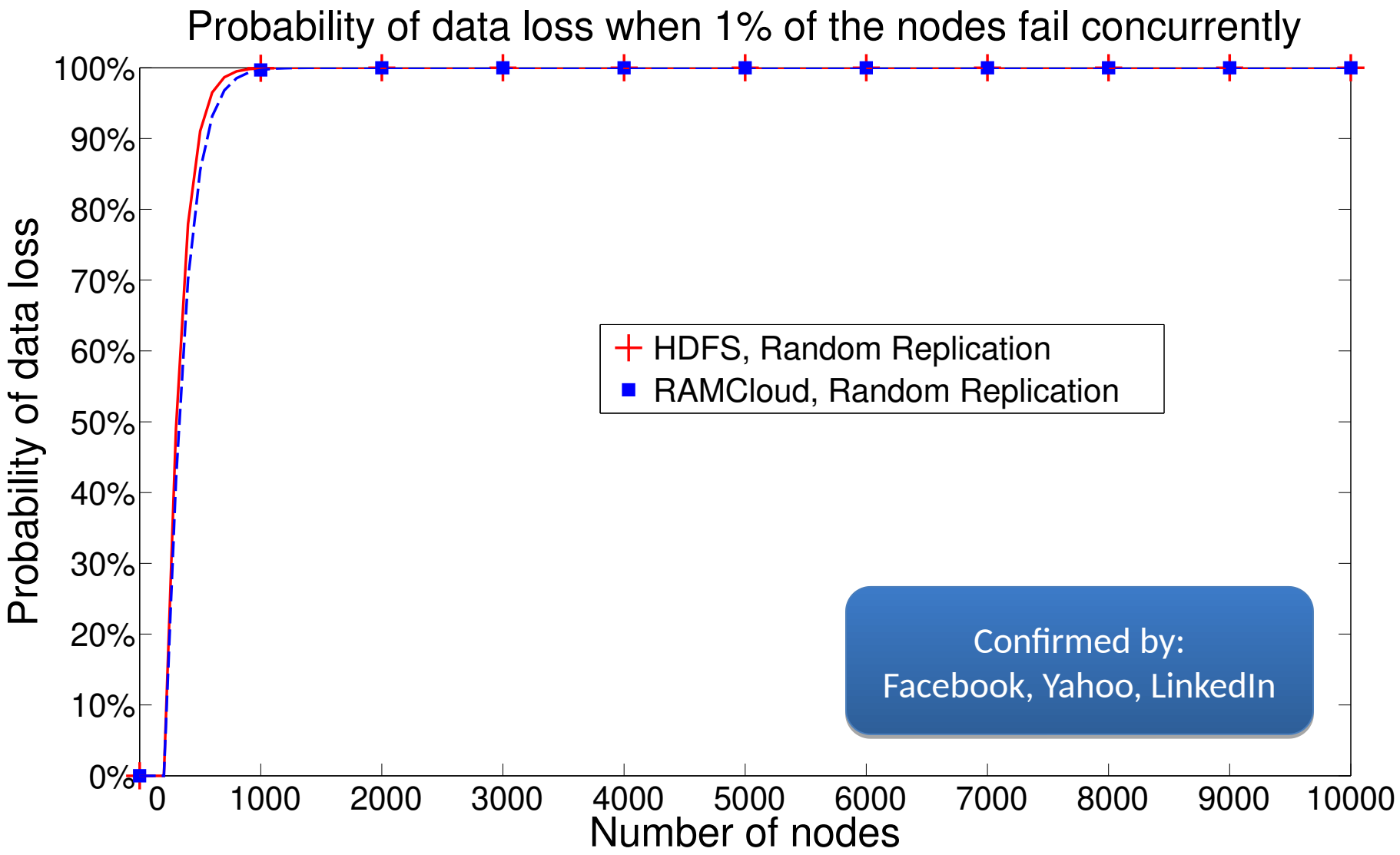
Random replication used by:

- HDFS
- GFS
- Windows Azure
- RAMCloud
- …

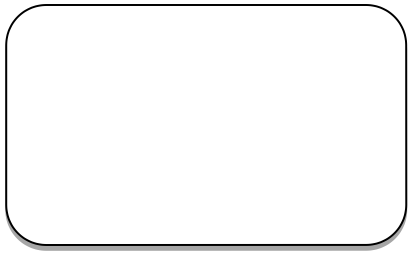
Not All Failures are Independent

- 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 upgrades
  - Power down

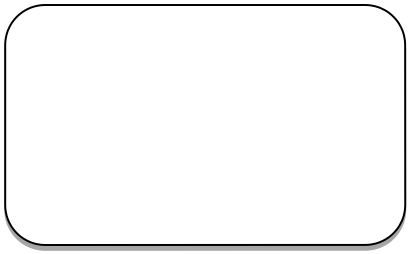
Random Replication Fails Under Simultaneous Failures



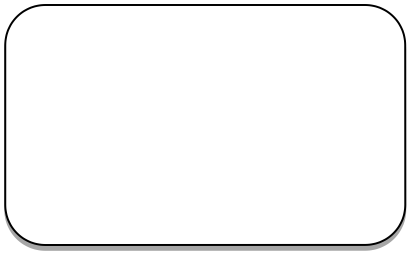
Random Replication



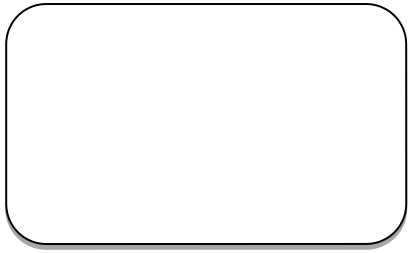
Node 1



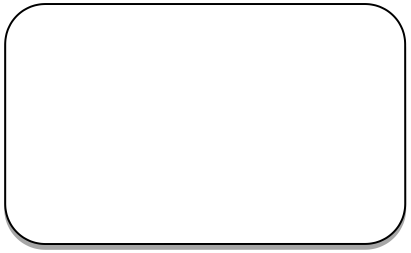
Node 2



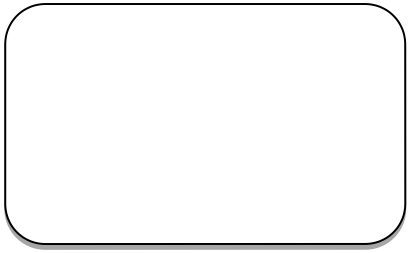
Node 3



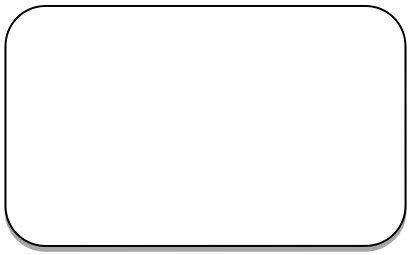
Node 4



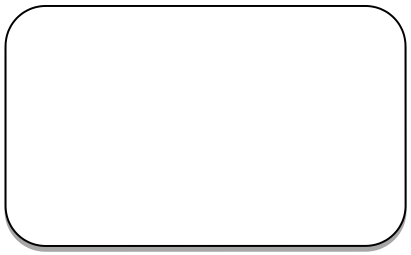
Node 5



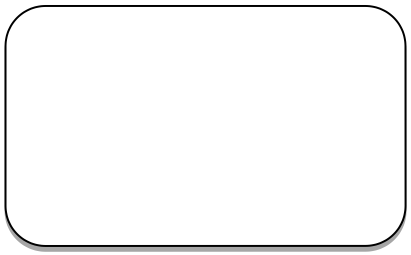
Node 6



Node 7

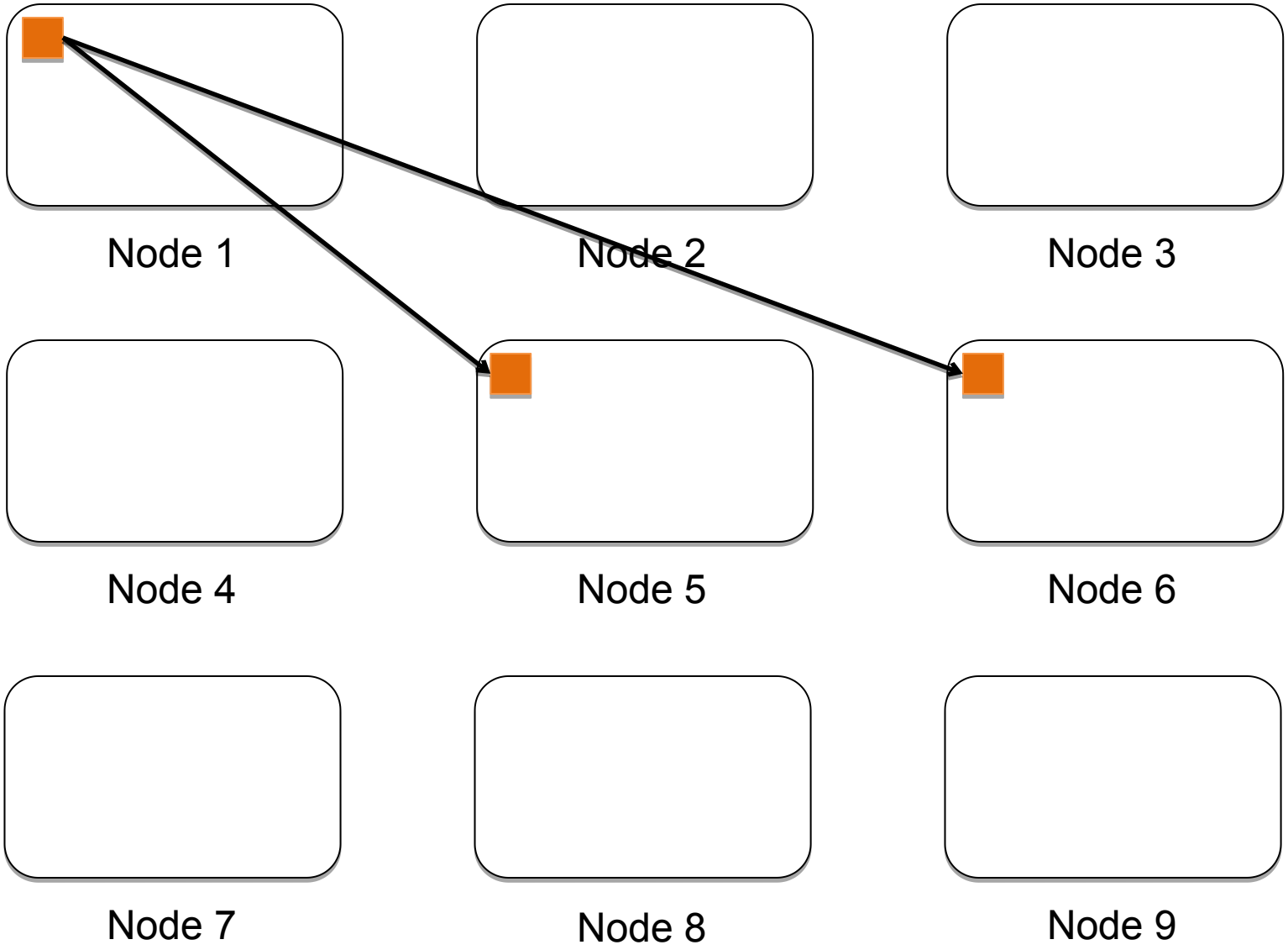


Node 8

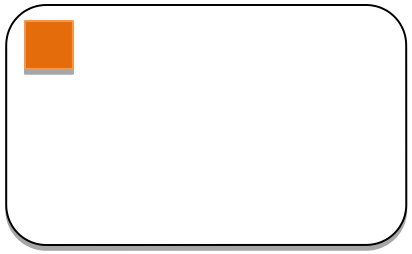


Node 9

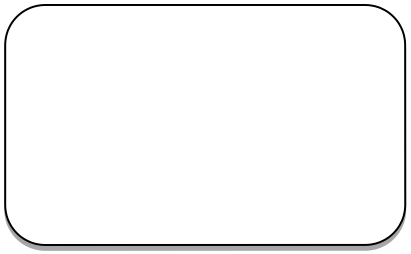
Random Replication



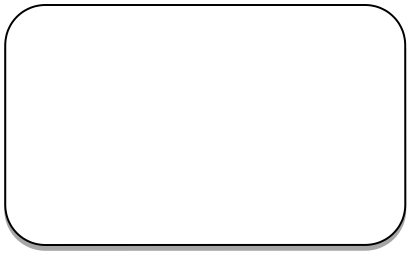
Random Replication



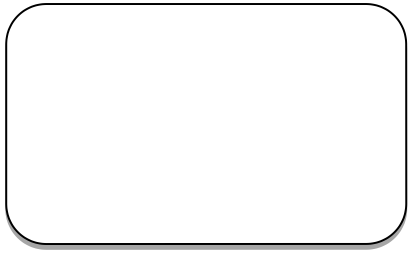
Node 1



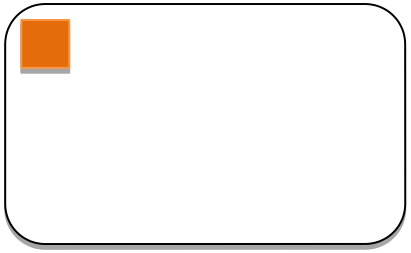
Node 2



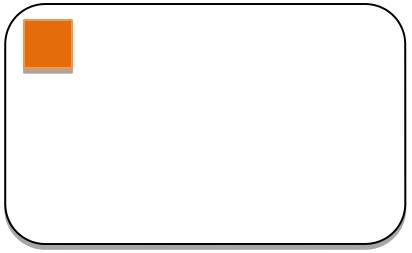
Node 3



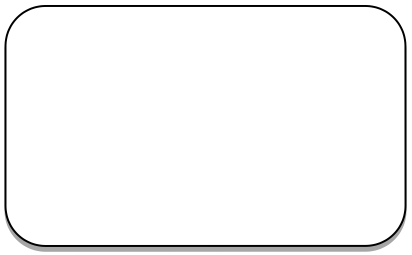
Node 4



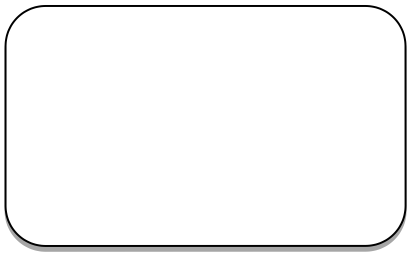
Node 5



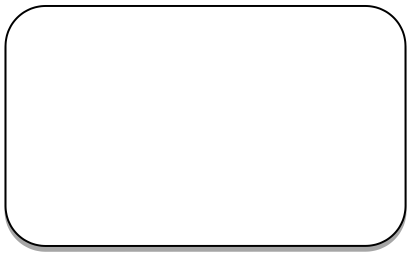
Node 6



Node 7

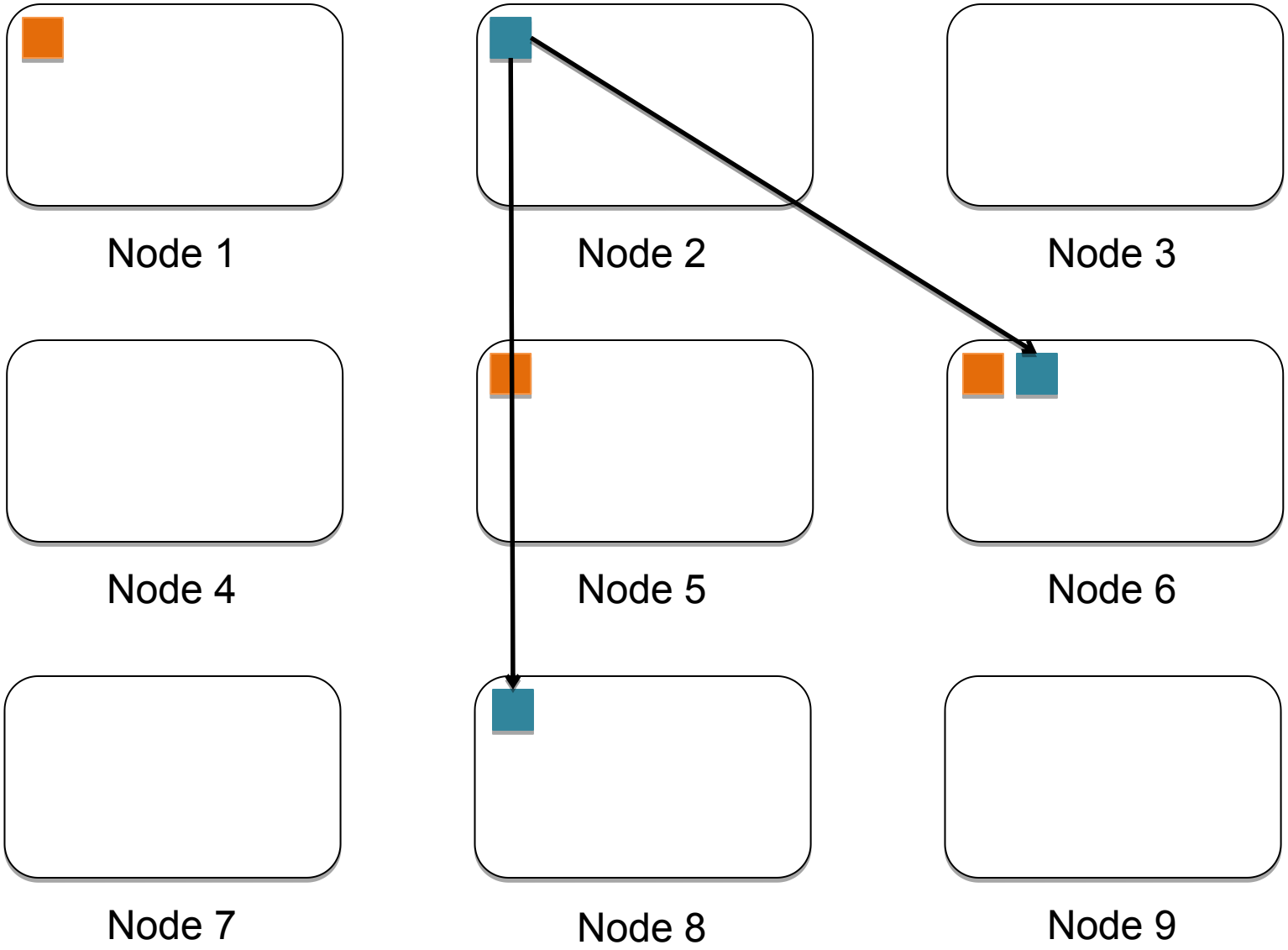


Node 8

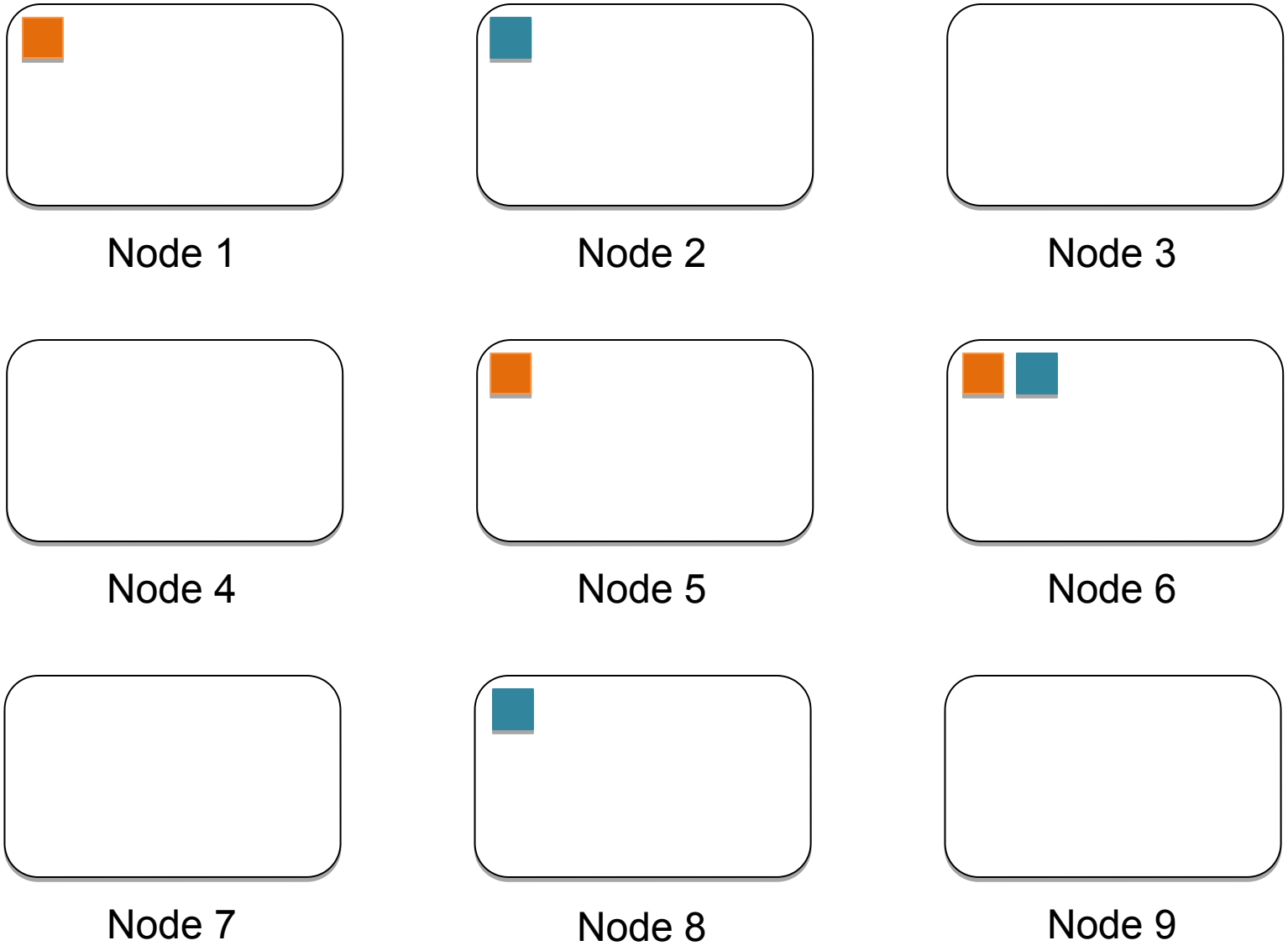


Node 9

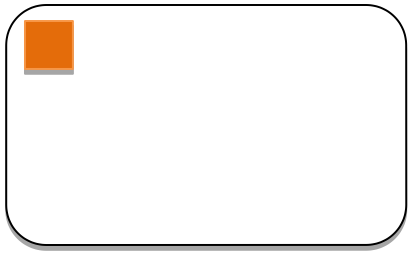
Random Replication



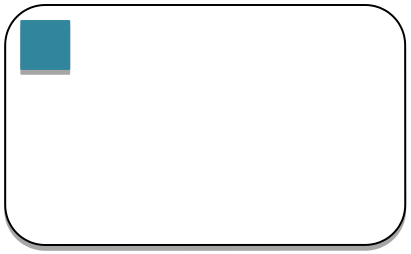
Random Replication



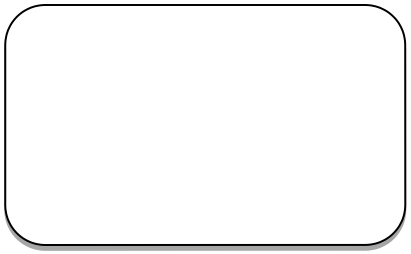
Random Replication



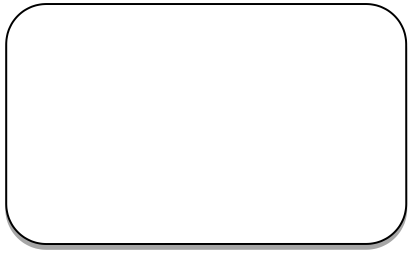
Node 1



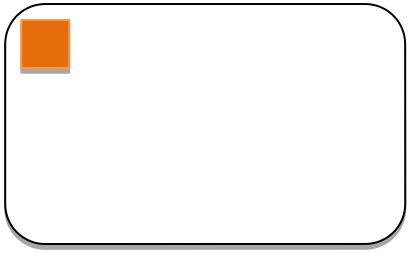
Node 2



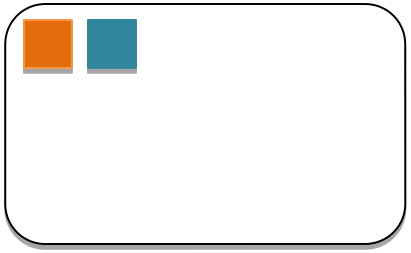
Node 3



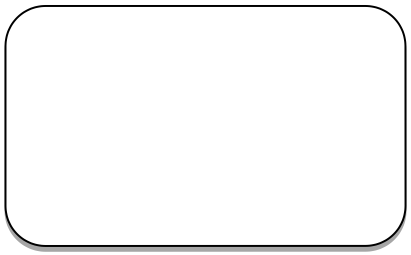
Node 4



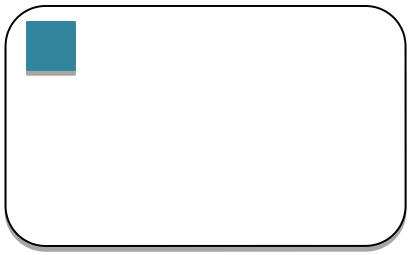
Node 5



Node 6



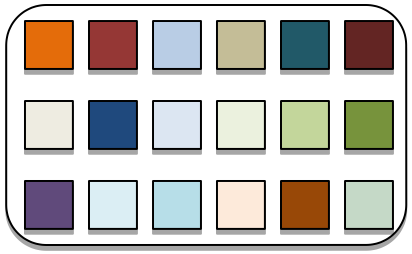
Node 7



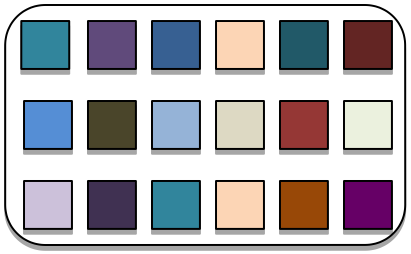
Node 8

Copysets:
{1, 5, 6}, {2, 6, 8}

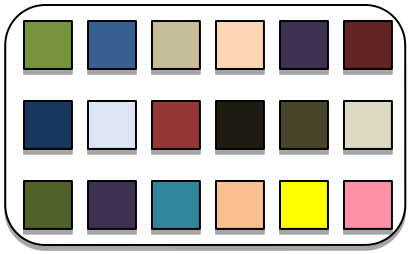
Random Replication



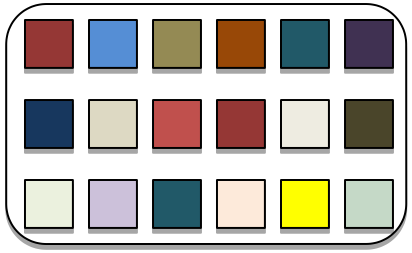
Node 1



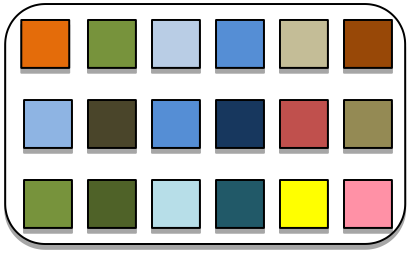
Node 2



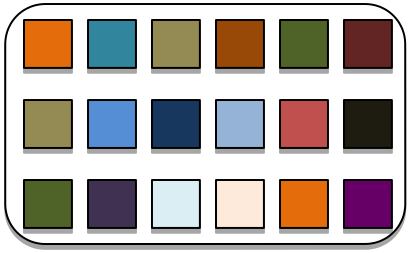
Node 3



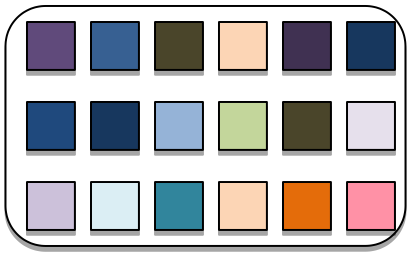
Node 4



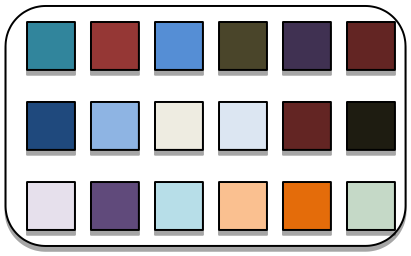
Node 5



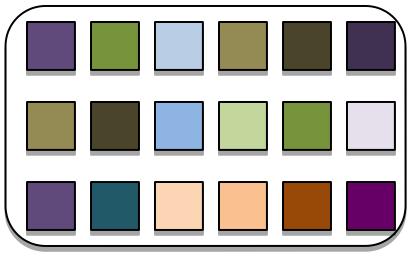
Node 6



Node 7

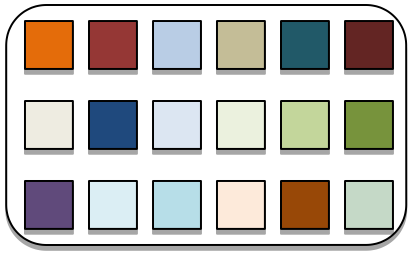


Node 8

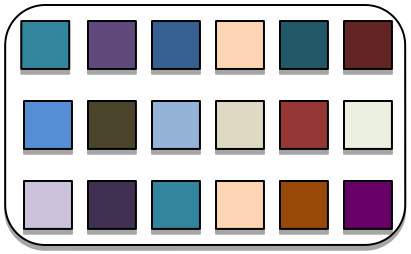


Node 9

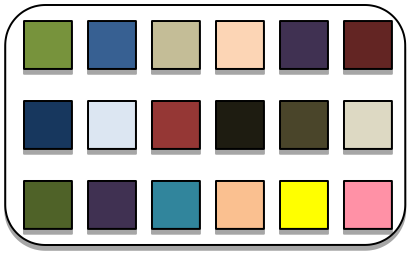
Random Replication



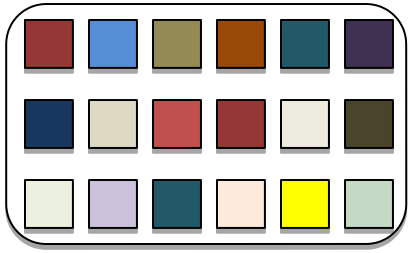
Node 1



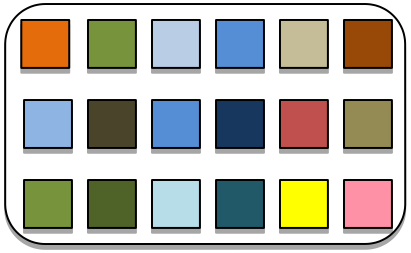
Node 2



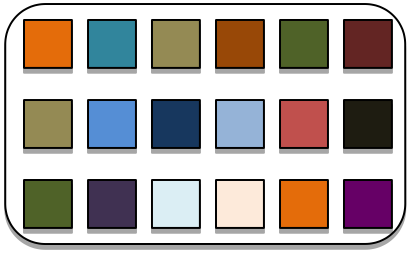
Node 3



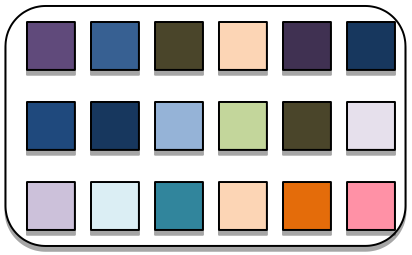
Node 4



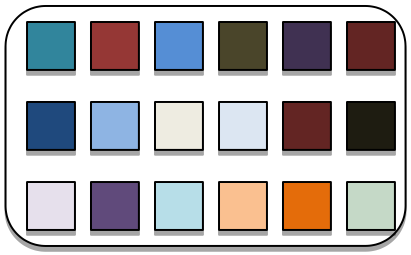
Node 5



Node 6



Node 7



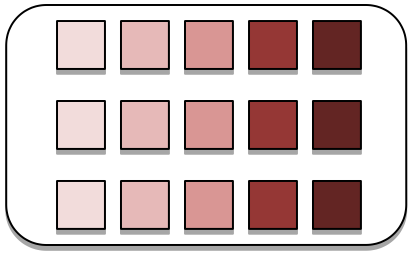
Node 8

Copysets:
{1, 2, 3}, {1, 2, 4}, {1, 2, 5},
{1, 2, 6}, {1, 2, 7}, {1, 2, 8},
...

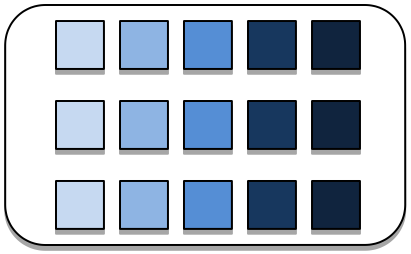
Random Replication Causes Frequent Data Loss

- Random replication eventually creates maximum number of copysets
  - Any combination of 3 nodes
  - $\binom{9}{3}$ = 84 copysets
- **If 3 nodes fail, 100% probability of data loss**
  - $\Pr(\textit{failure}) = \frac{\#\,\textit{copysets}}{\max\#\,\textit{copysets}} = \frac{84}{84}$

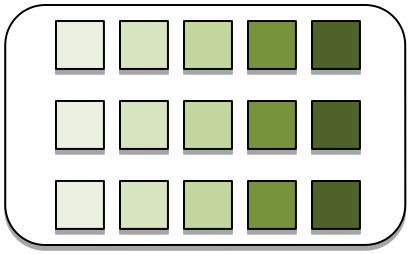
MinCopysets



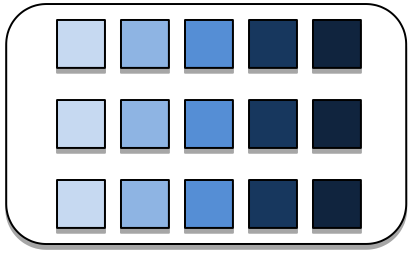
Node 1



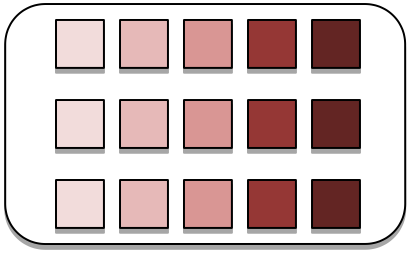
Node 2



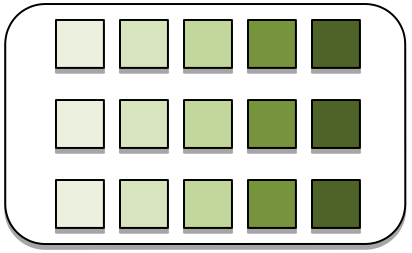
Node 3



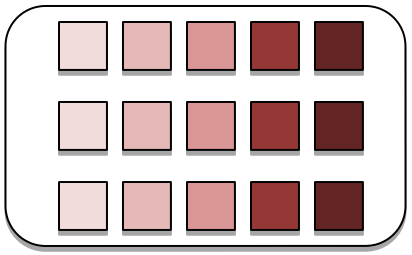
Node 4



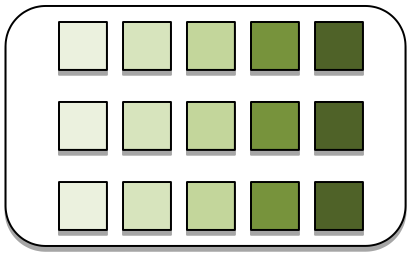
Node 5



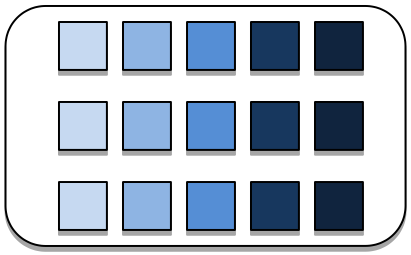
Node 6



Node 7

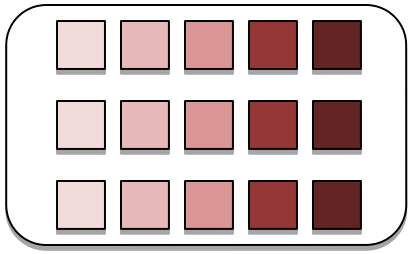


Node 8

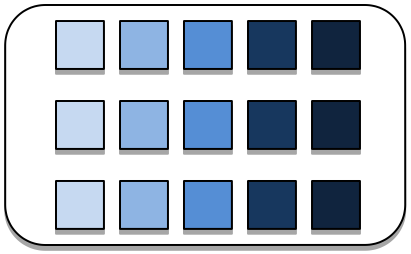


Node 9

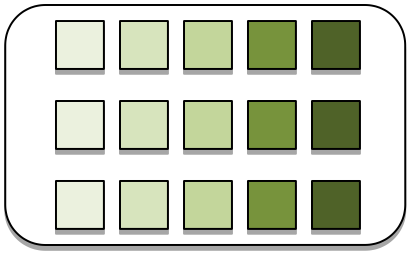
MinCopysets



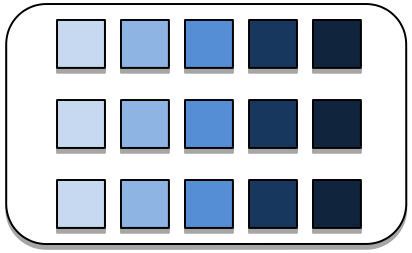
Node 1



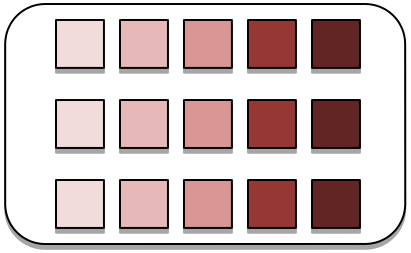
Node 2



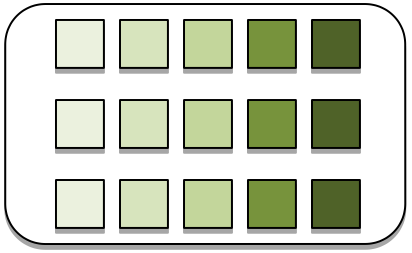
Node 3



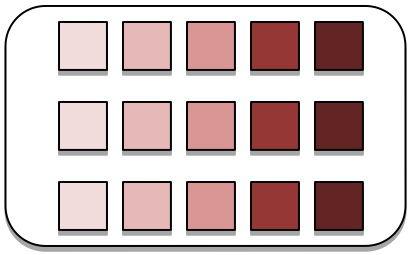
Node 4



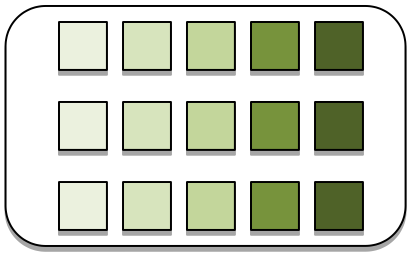
Node 5



Node 6



Node 7



Node 8

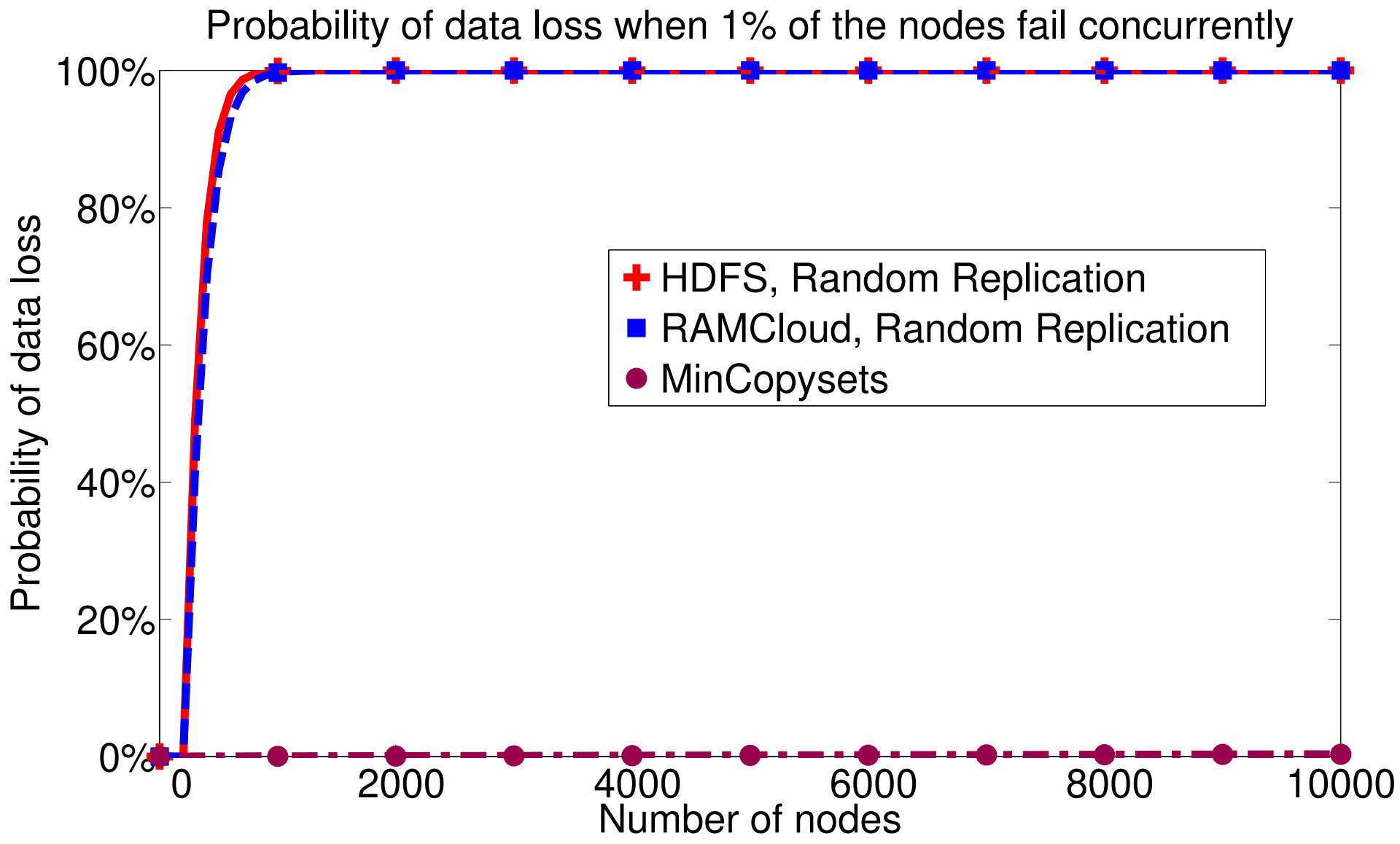
Copysets:
{1, 5, 7}, {2, 4, 9}, {3, 6, 8}

MinCopysets Minimizes Data Loss Frequency

- MinCopysets creates minimum number of copysets
  - Only {1, 5, 7}, {2, 4, 9}, {3, 6, 8}

- **If 3 nodes fail, 3.5% of data loss**
  - $\Pr(\textit{failure}) = \frac{3}{84}$

MinCopysets Reduces Probability of Data Loss



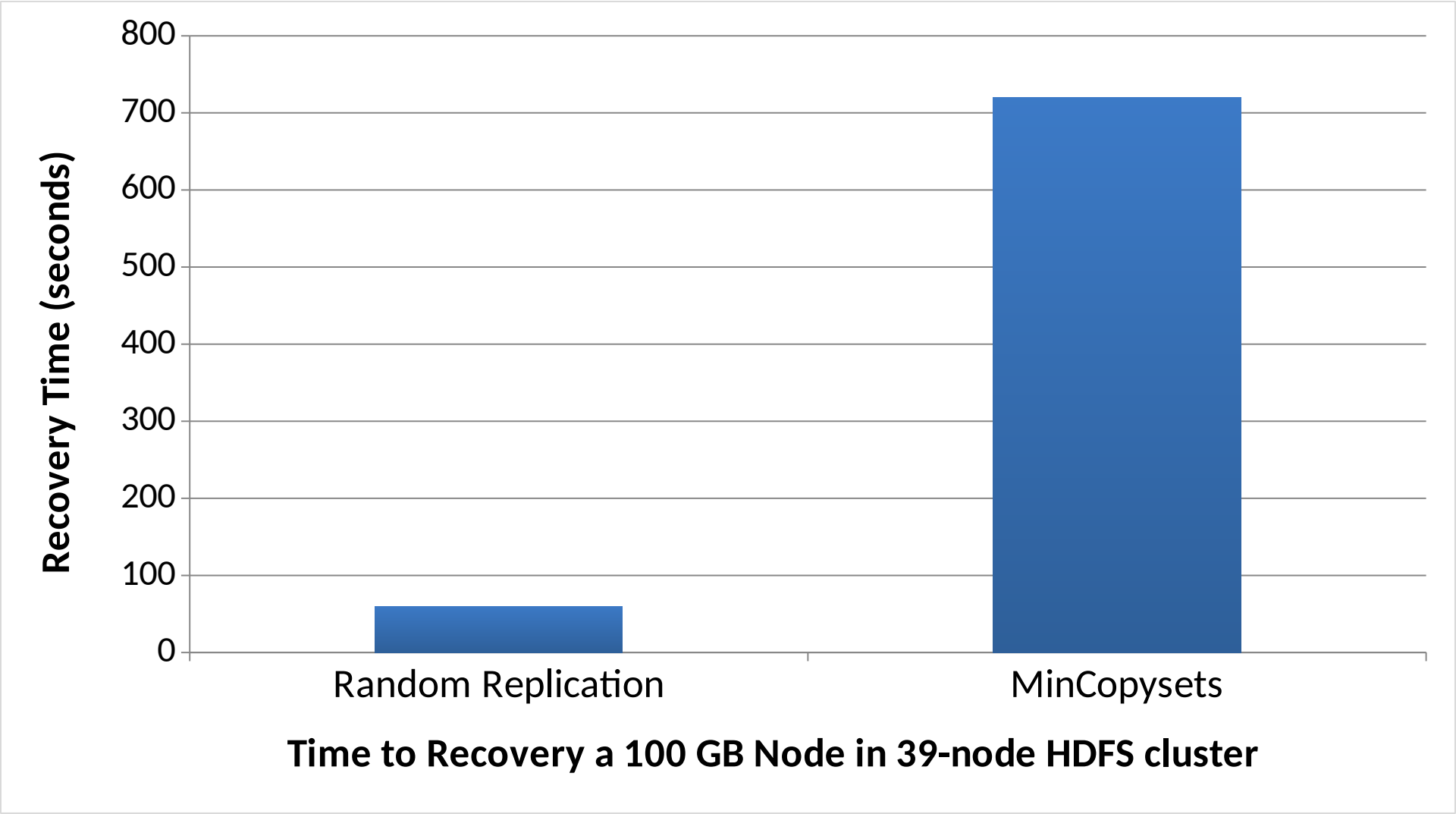
The Trade-off

	MinCopysets	Random Replication
Mean Time to Failure	625 years	1 year
Amount of Data Lost	1 TB	5.5 GB

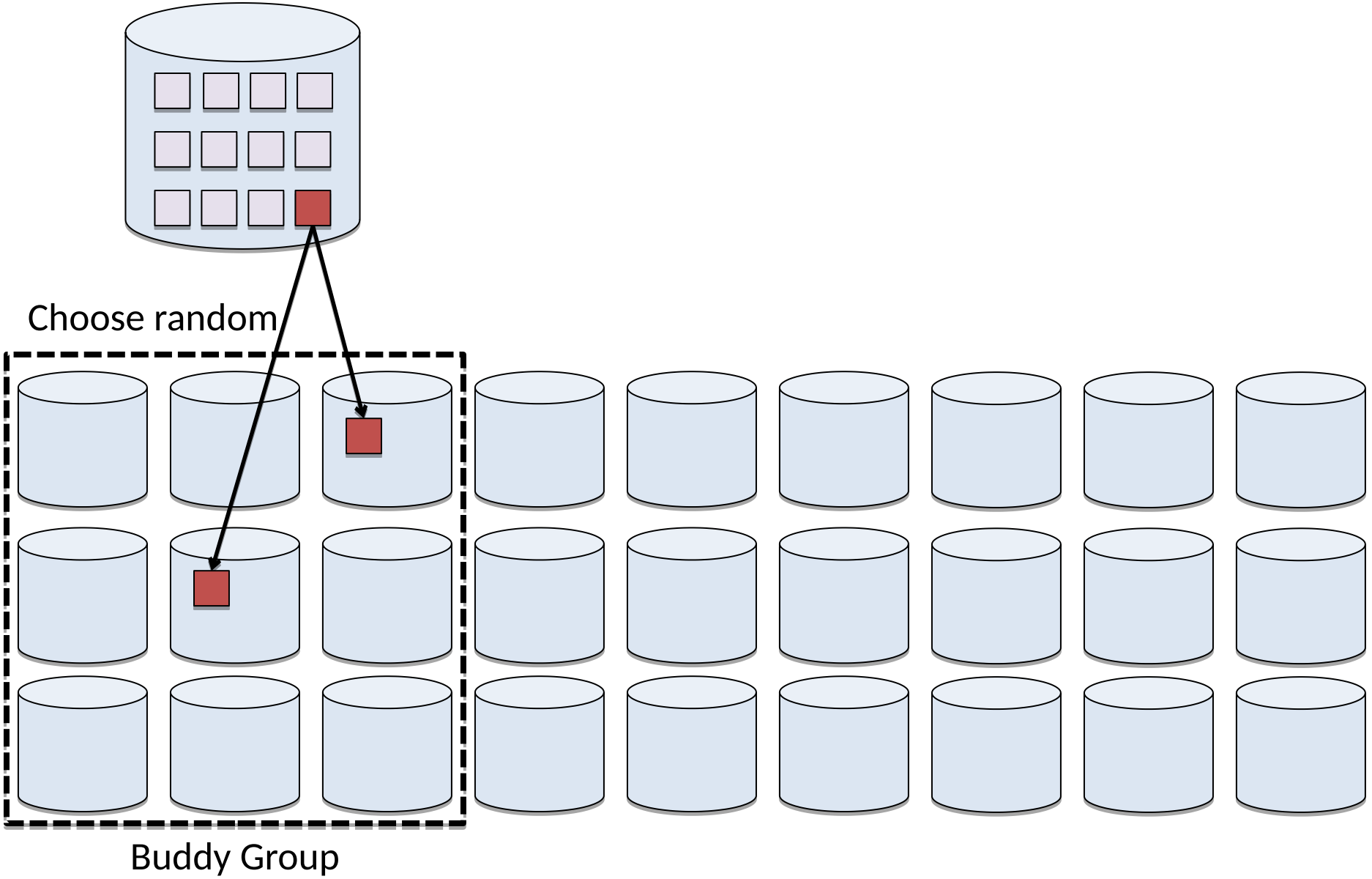
- 5000-node cluster
- Power outage occurs every year

Confirmed by:
Facebook, LinkedIn,
NetApp, Google

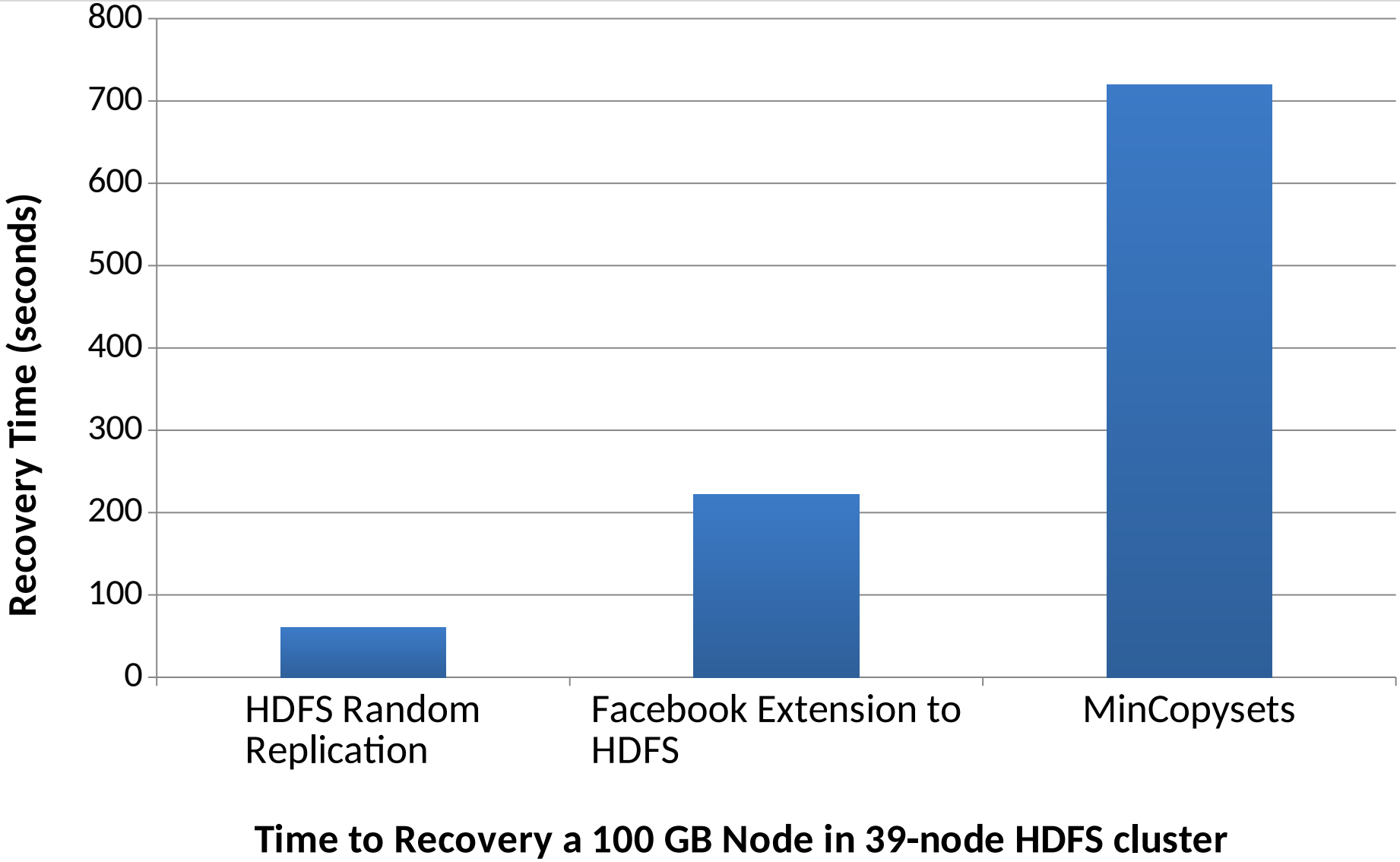
Problem: MinCopysets Increases Single Node Recovery Time



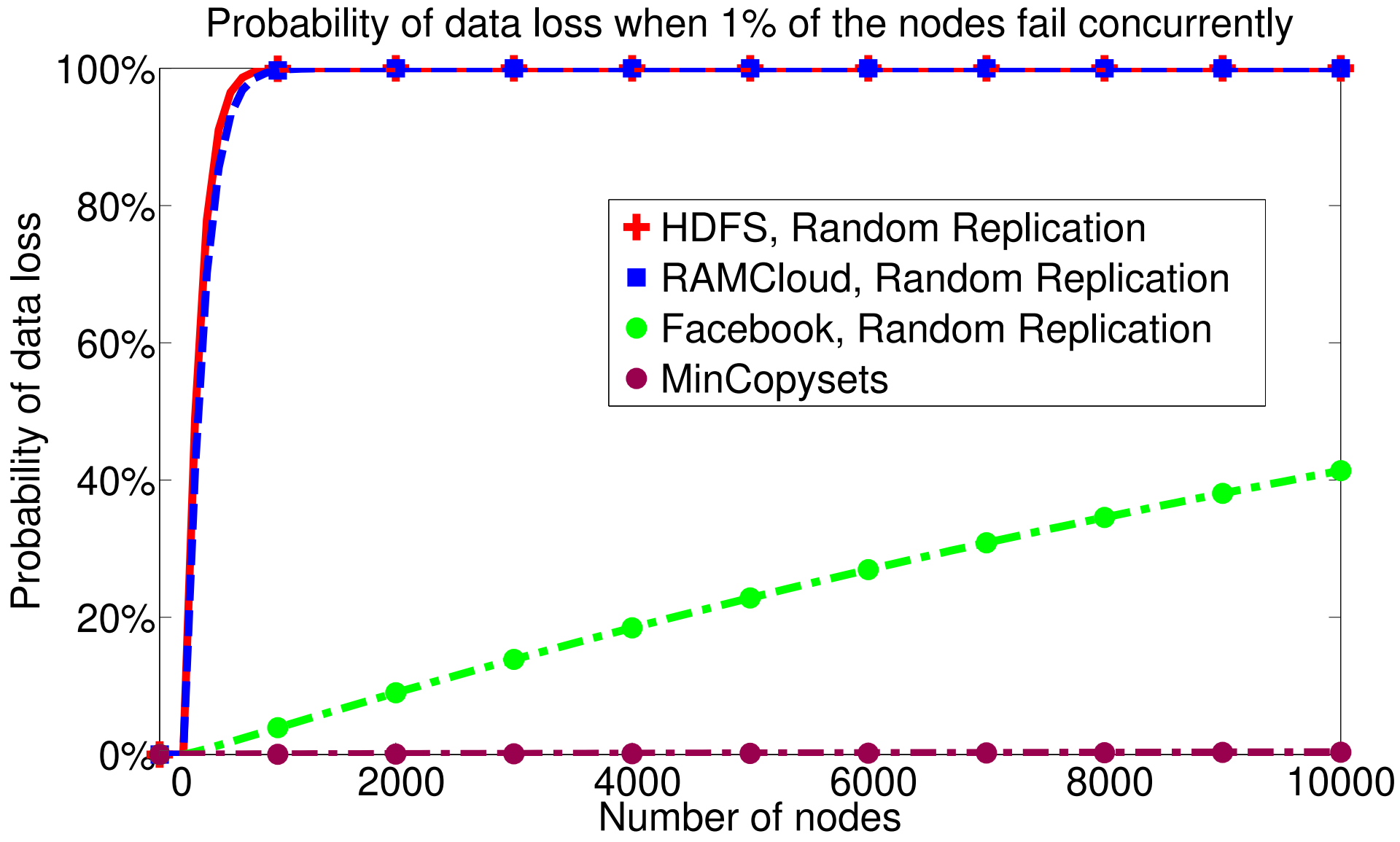
Facebook Extension to HDFS



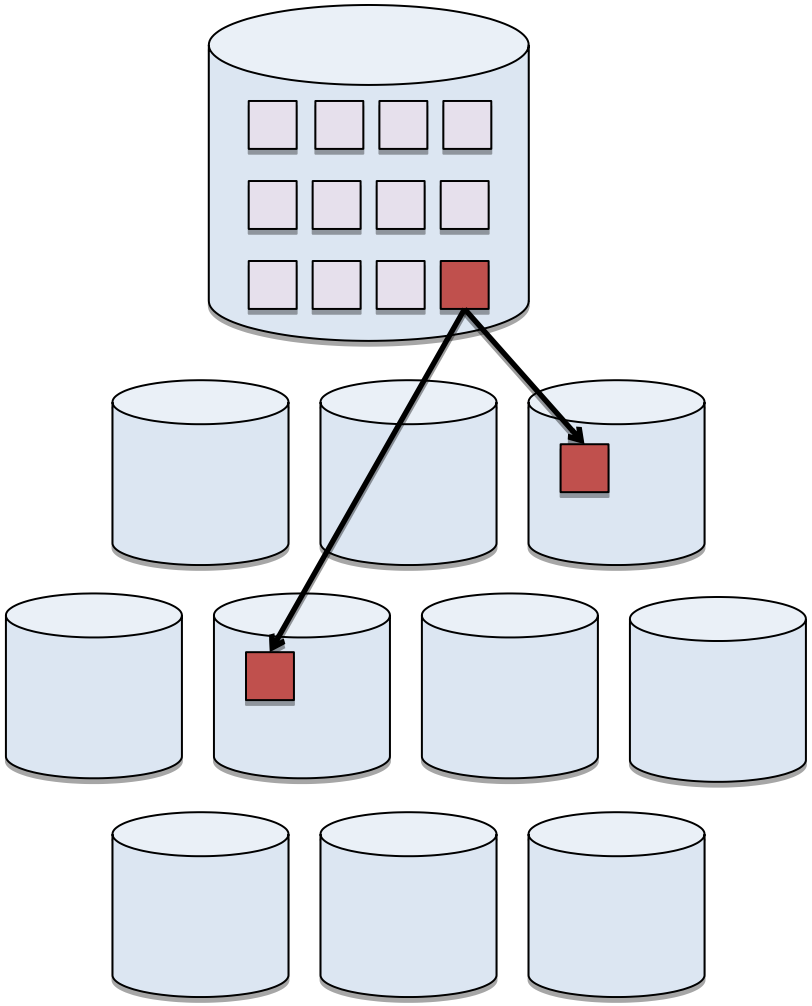
A Compromise



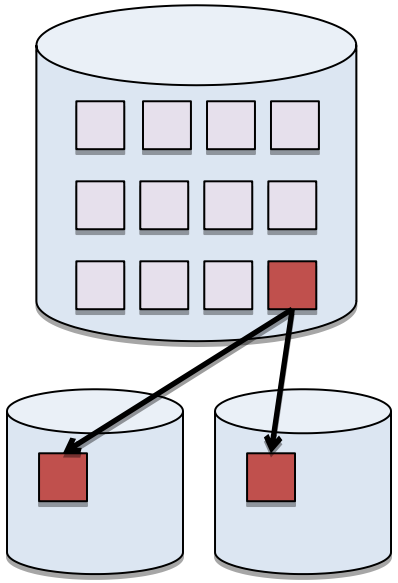
Can We Do Better?



Definition: Scatter Width

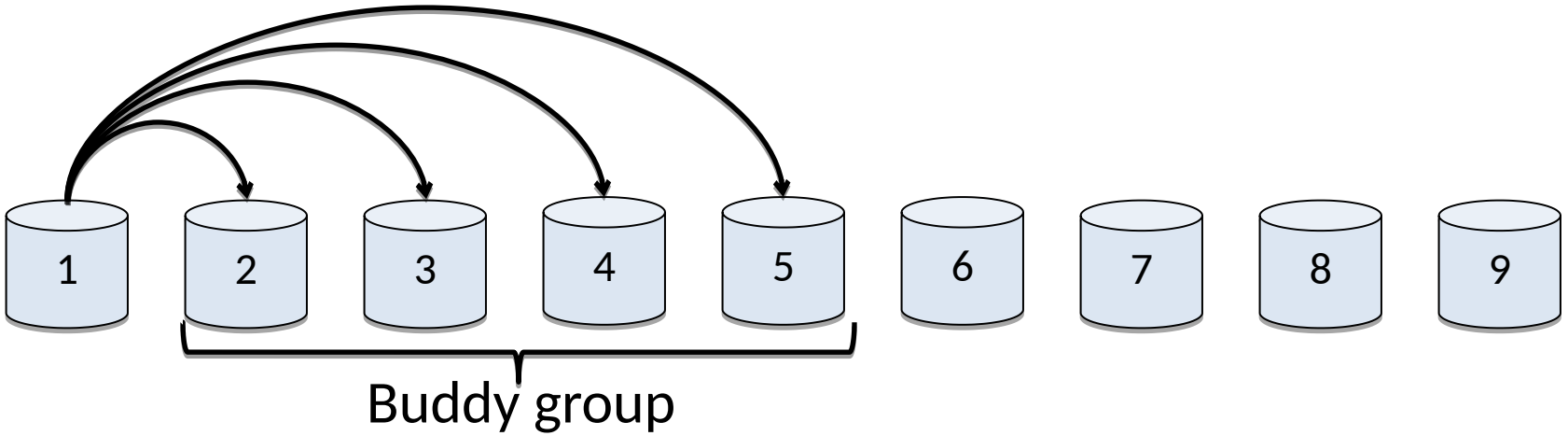


Facebook Extension to HDFS
Scatter Width = 10



MinCopysets
Scatter Width = 2

Facebook Extension to HDFS



- Node 1's copysets:
  - {1, 2, 3}, {1, 2, 4}, {1, 2, 5}, {1, 3, 4}, {1, 3, 5}, {1, 4, 5}
- Overall: 54 copysets
- If 3 nodes fail simultaneously:
- $\Pr(\textit{failure}) = \frac{54}{84} = 64\%$

Copyset Replication: Intuition

- Same scatter width (4), different scheme:

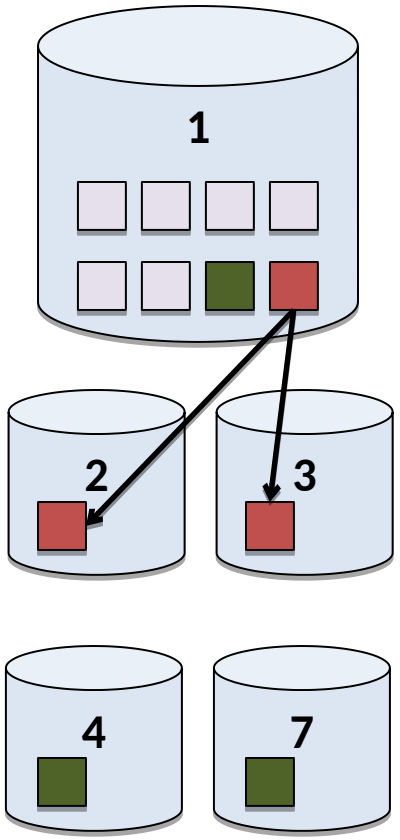
  {1, 2, 3}, {4, 5, 6}, {7, 8, 9}

  {1, 4, 7}, {2, 5, 8}, {3, 6, 9}

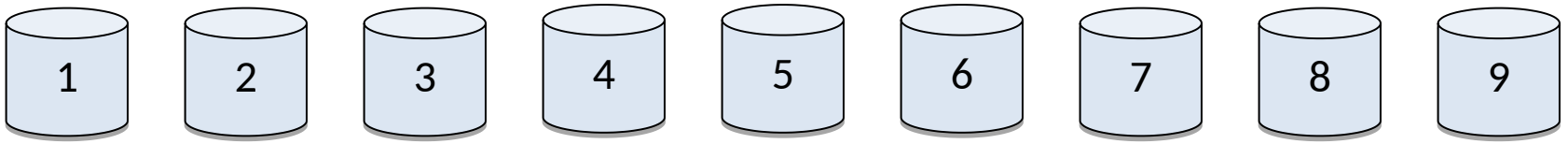
Ingredients of ideal scheme

1. Maximize scatter width
2. Minimize overlaps

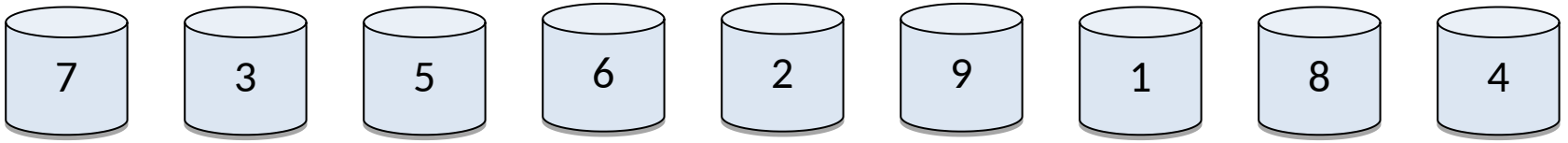
- $\Pr(\textit{failure}) = \frac{6}{84} = 7\% << 64\%$



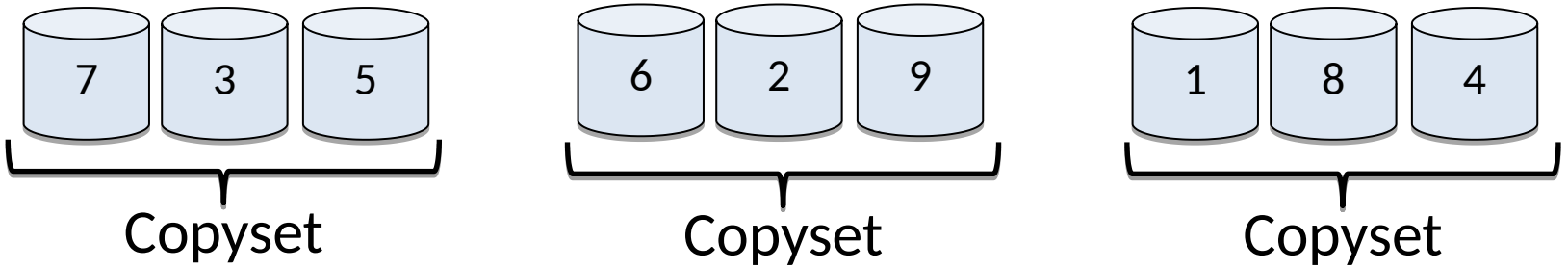
Copyset Replication: Initialization



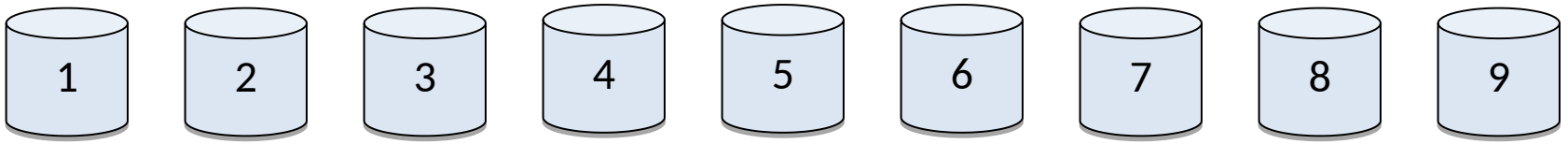
Random Permutation



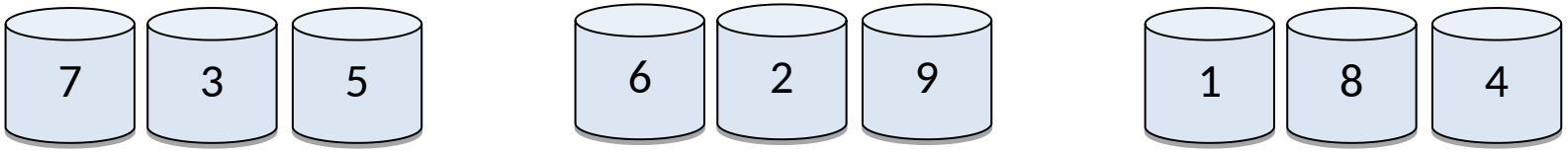
Split into copysets (Scatter width = 2)



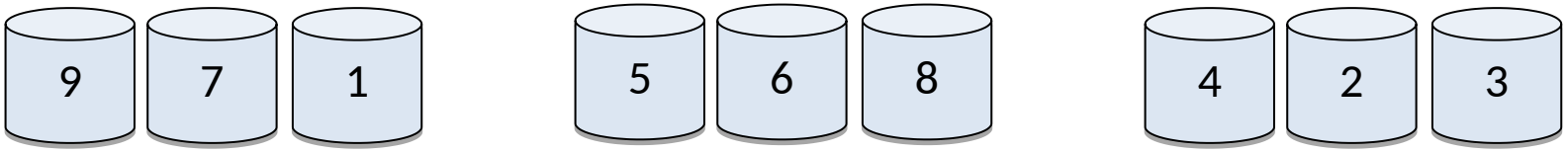
Copyset Replication: Initialization



Permutation 1: Scatter width = 2



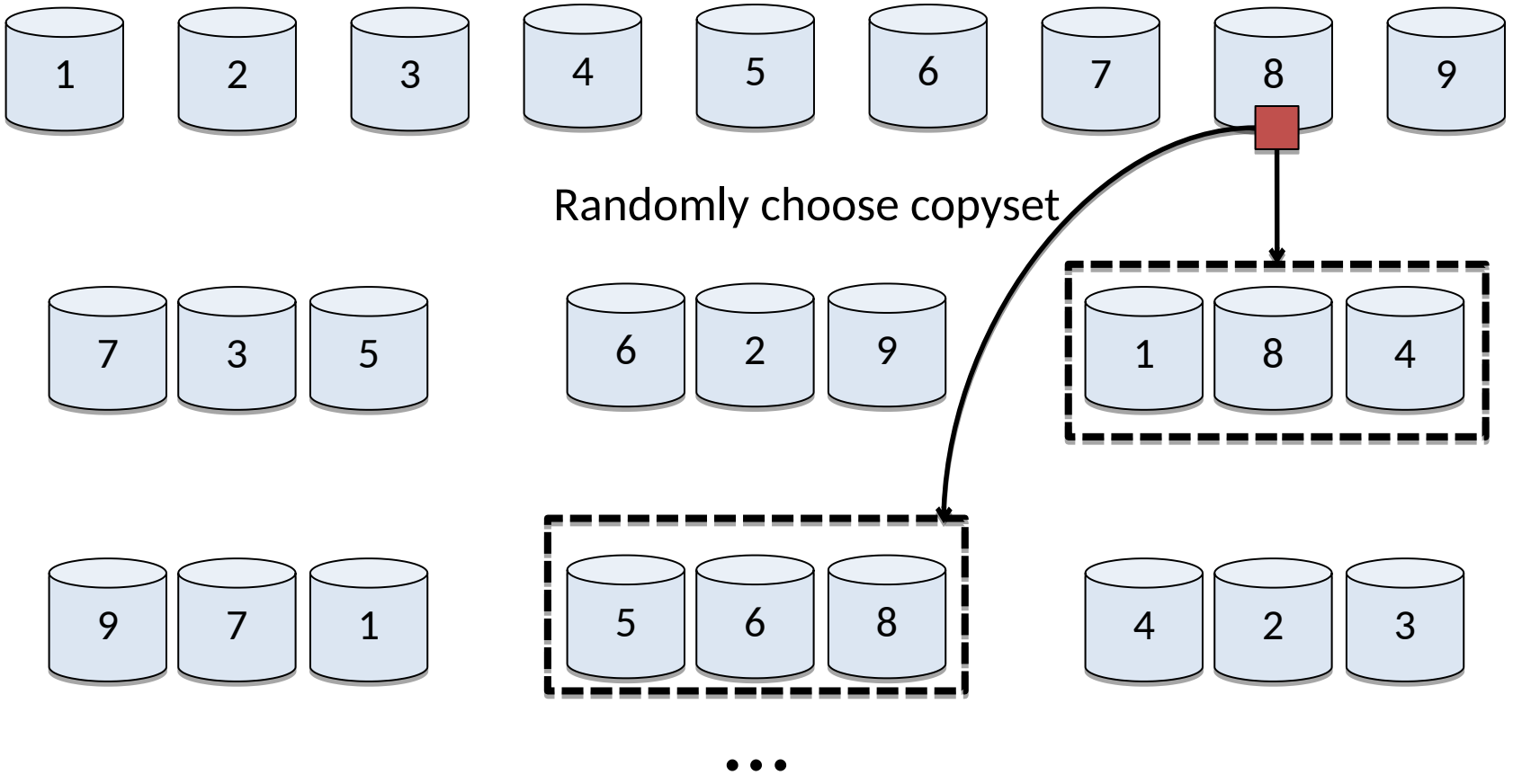
Permutation 2: Scatter width = 4



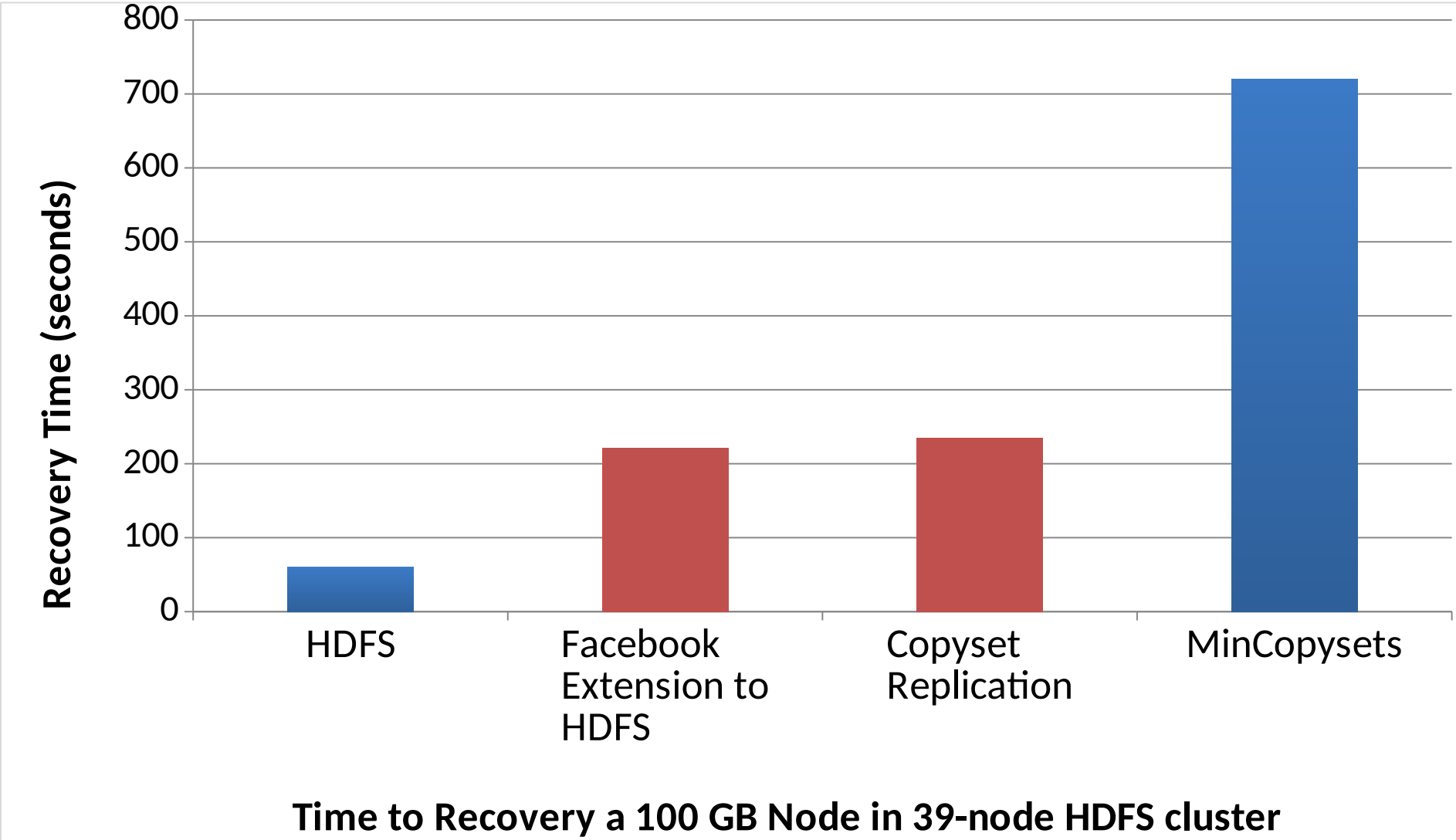
...

Permutation 5: Scatter width = 10

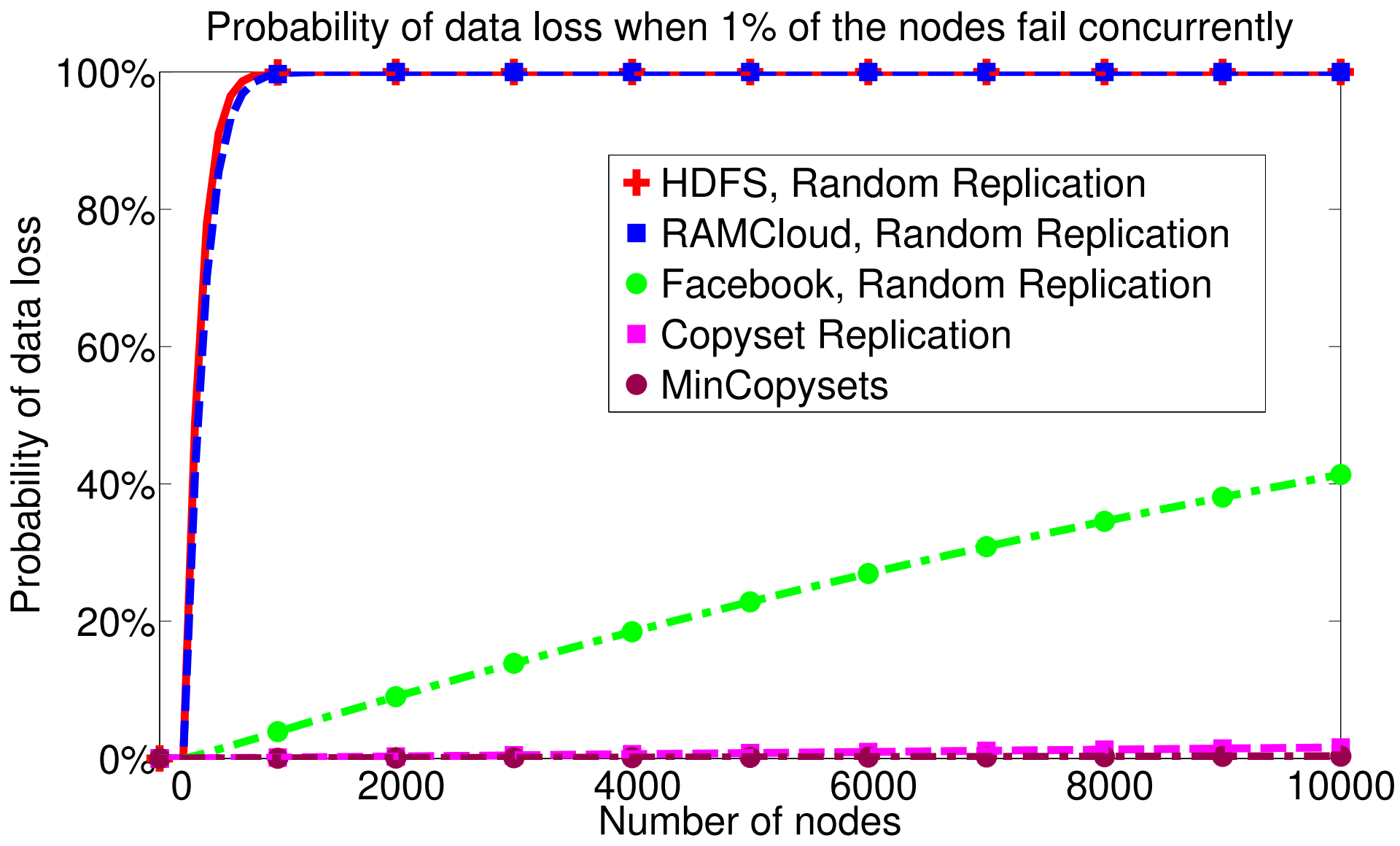
Copyset Replication: Replication



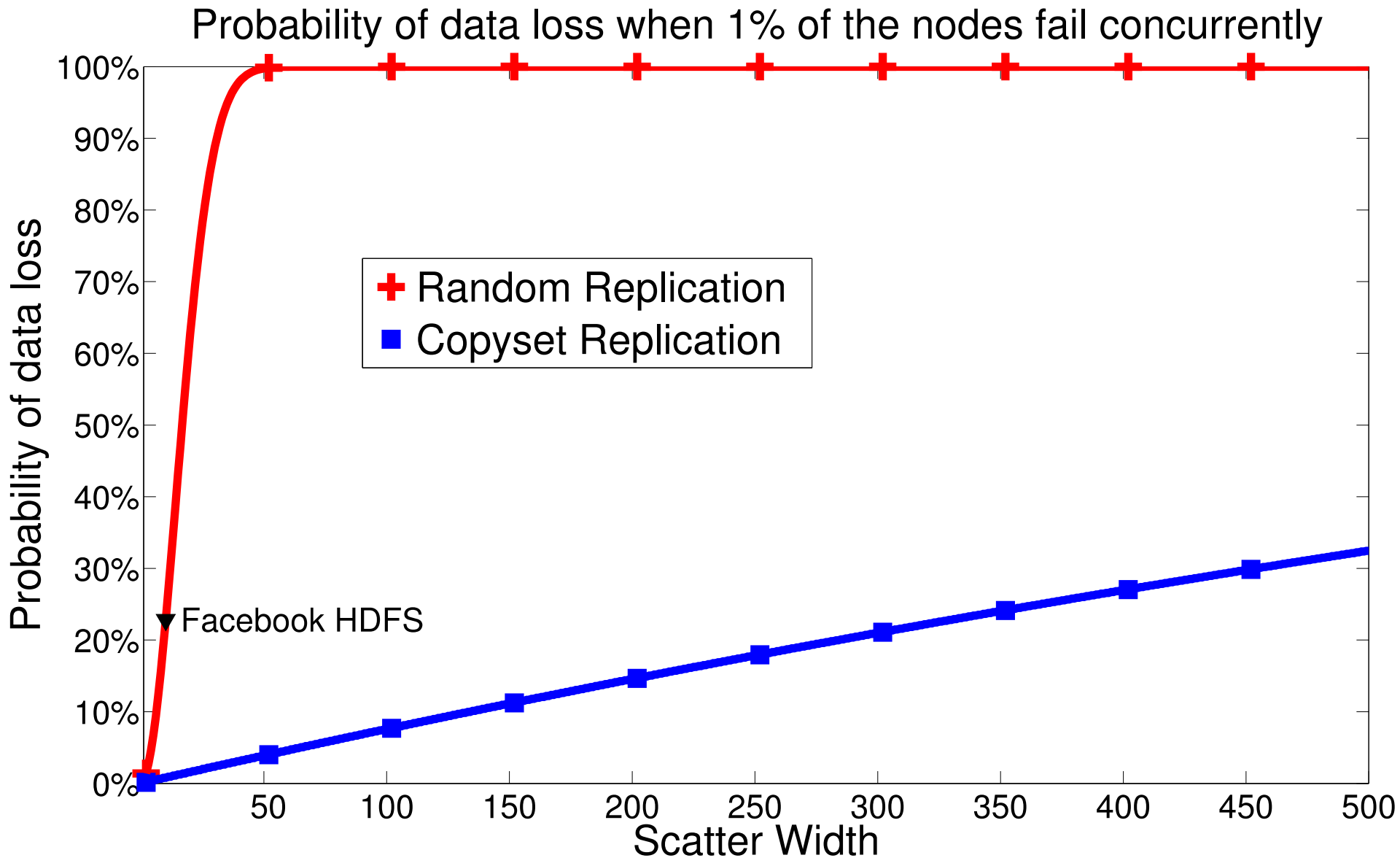
Insignificant Overhead



Copyset Replication



Inherent Trade-off



Related Work

- BIBD (Balanced Incomplete Block Designs)
  - Originally proposed for designing agricultural experiments in the 1930's! [Fisher, '40]
- Other applications
  - Power downs [Harnik et al '09, Leverich et al '10, Thereska '11]
  - Multi-fabric interconnects [Mehra, '99]

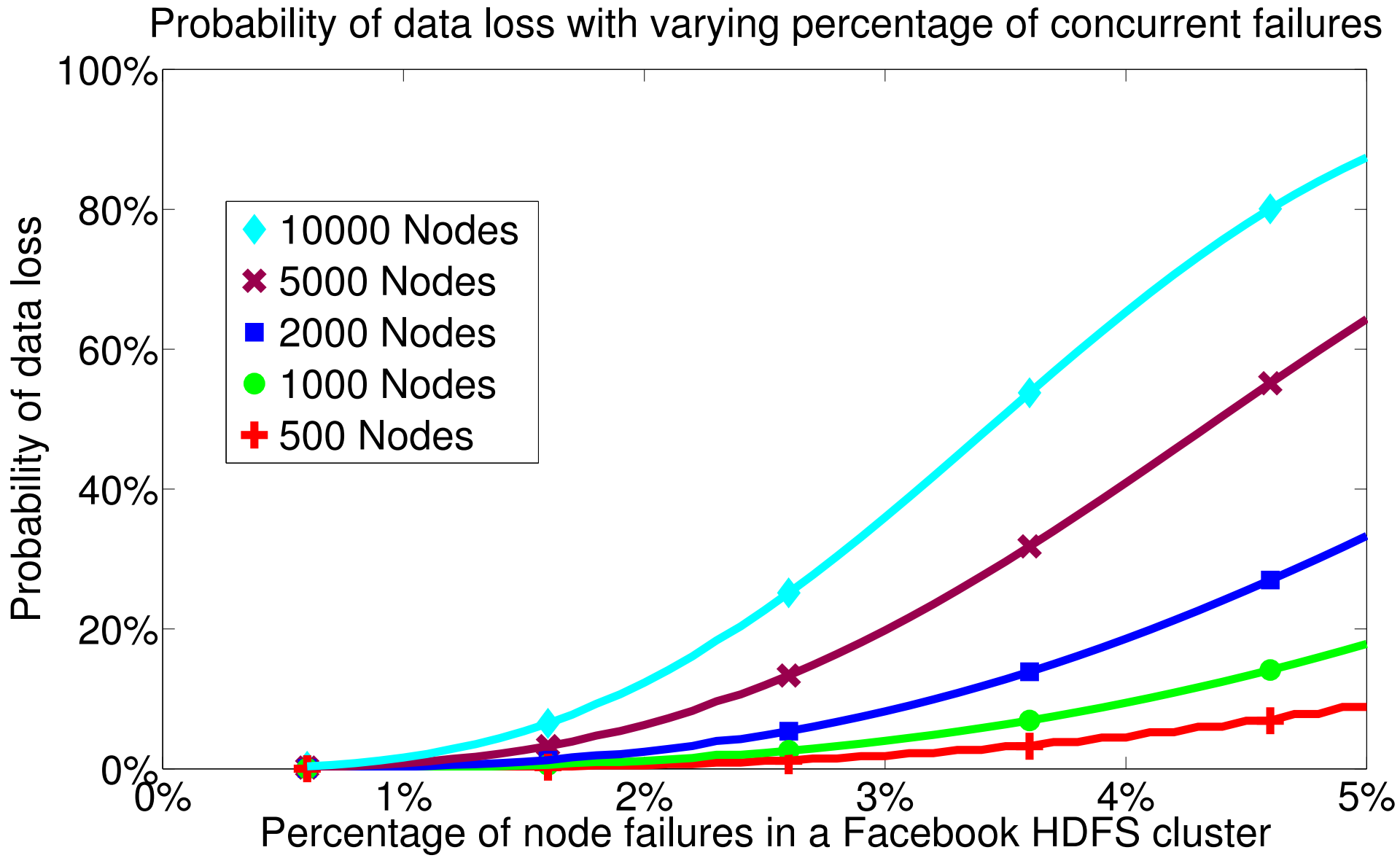
Summary

1. Many storage systems randomly spray their data across a large number of nodes
2. Serious problem with correlated failures
3. Copyset Replication is a better way of spraying data that decreases the probability of correlated failures

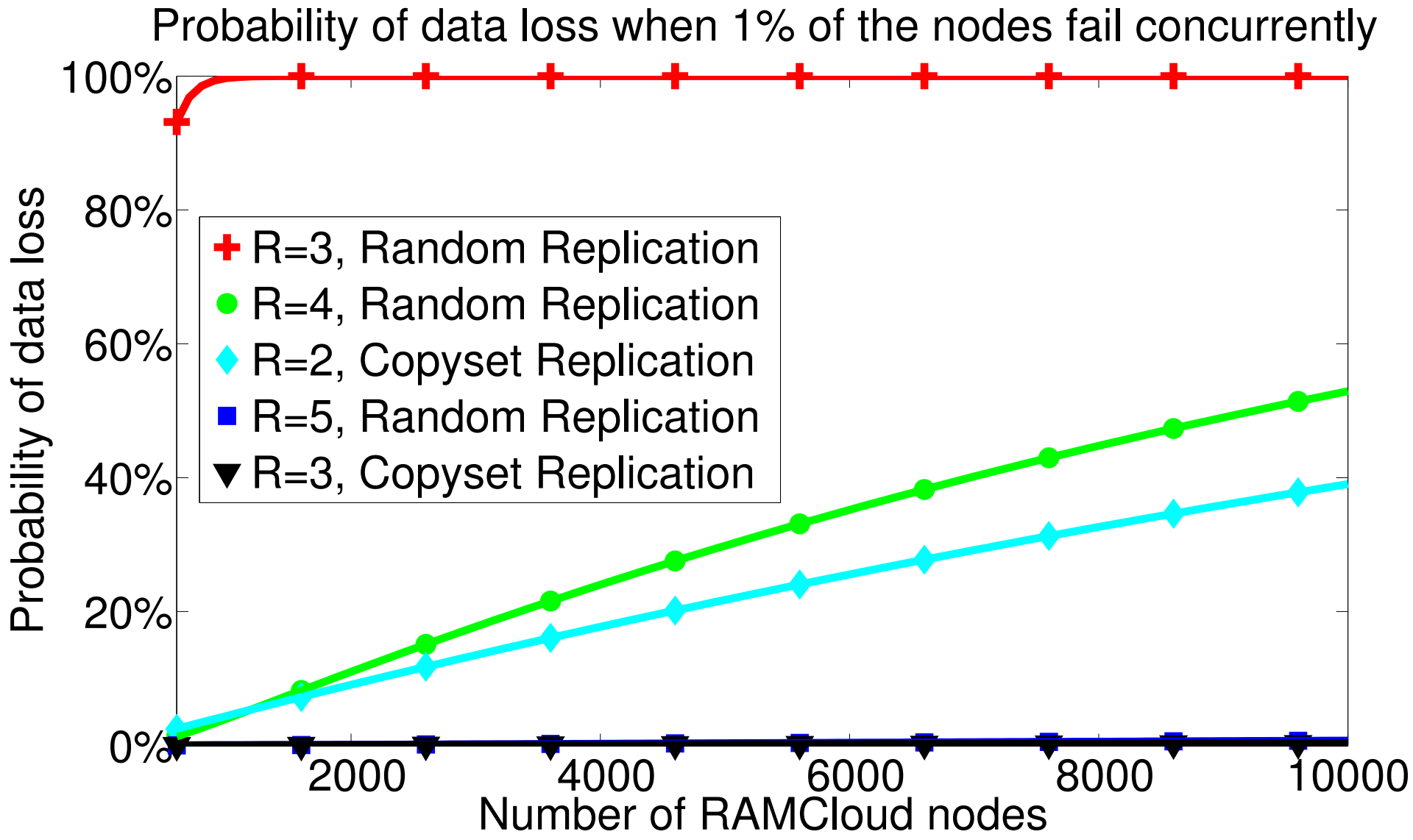
Thank You!

Stanford University

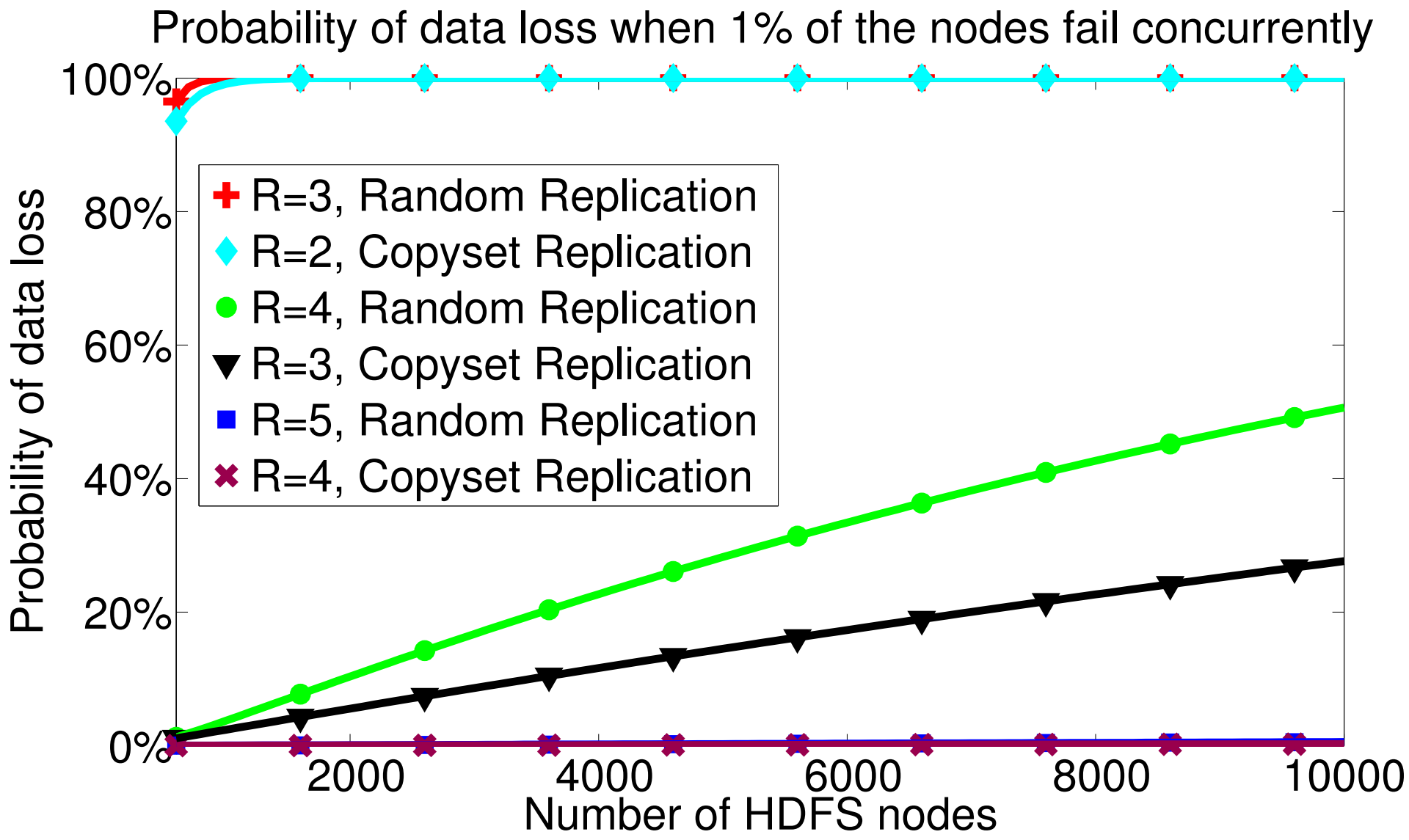
More Failures (Facebook)



RAMCloud


Probability of data loss when 1% of the nodes fail concurrently
Probability of data loss
100%
80%
60%
40%
20%
0%
2000
4000
6000
8000
10000
Number of RAMCloud nodes
R=3, Random Replication
R=4, Random Replication
R=2, Copyset Replication
R=5, Random Replication
R=3, Copyset Replication

HDFS


Probability of data loss when 1% of the nodes fail concurrently
Probability of data loss
100%
80%
60%
40%
20%
0%
2000
4000
6000
8000
10000
Number of HDFS nodes
R=3, Random Replication
R=2, Copyset Replication
R=4, Random Replication
R=3, Copyset Replication
R=5, Random Replication
R=4, Copyset Replication