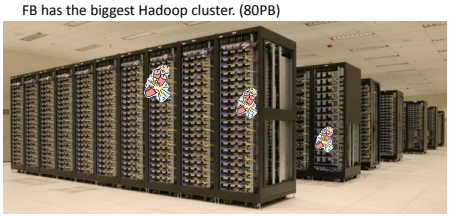
Locally Repairable Codes

Dimitris S. Papailiopoulos and Alexandros G. Dimakis

USC Viterbi School of Engineering

Big Data

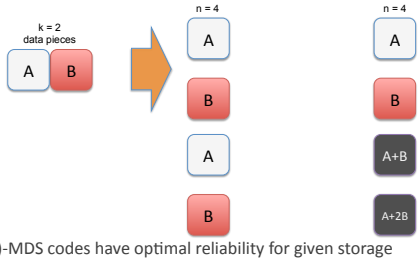
- Big Data Players (Facebook, Amazon, Google, Yahoo, ...)
- FB has the biggest Hadoop cluster. (80PB)



Cluster of machines running Hadoop at Yahoo! (Source: Yahoo!)

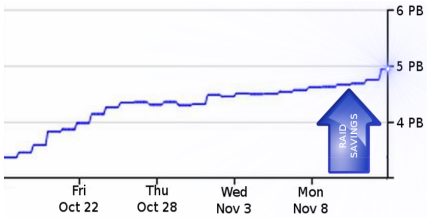
- Failures are the **norm.**
- We need to protect the data: **Introduce redundancy**

Reliability: Replication vs. Codes



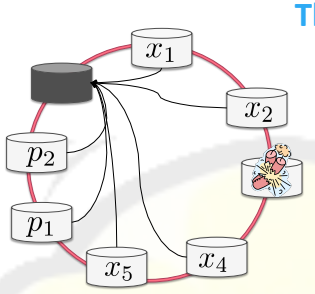
- (n,k)-MDS codes have optimal reliability for given storage
- 8% of Facebook Archival storage uses coding (most is still 3x replication)
- Plans to code 50% of archival data

MDS-Codes: Pros & Cons



Repair Cost

The Code Repair Problem



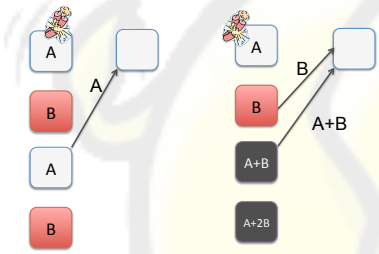
- **A node is lost:** We need to exactly repair it.
- Practice: **ALL** nodes are contacted,
  everything is downloaded for repair [Hadoop]
  (matrix inversions take place)

Naïve repair: 1) generates enormous communication
2) accesses a great number of nodes

Metrics of interest:

- Bits **communicated** for repair
- Bits **read** for repairs
- **Locality = Number of Nodes used** during repair.

Locality of Repairs



Replication
Efficient Repair
Low reliability

MDS Codes
High reliability
Inefficient Repair

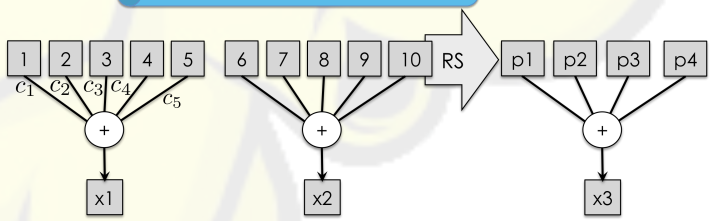
Q: Tradeoff between Locality-Reliability?

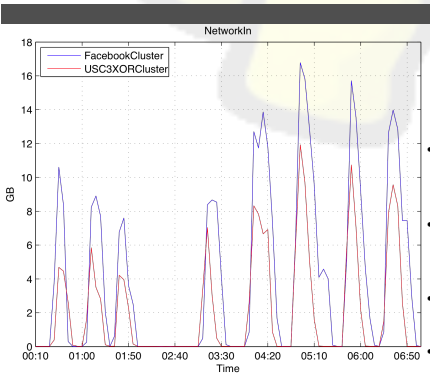
Locally Repairable Codes

Locally-Repairable Codes
Replication-like Repair and MDS-like Reliability

A new reliability-locality-storage trade-off is established

$$d \leq n - \left\lceil \frac{k}{r(1+\epsilon)} \right\rceil - \left\lceil \frac{k}{1+\epsilon} \right\rceil + 2$$





Implementing LRC

[Sathiamoorthy, Asteris, P, Dimakis, facebook]

- LRC was tested on facebook clustehrs and Amazon ec2 clusters (100 machines).
- Reduces disk IO and network bandwidth by approximately 2x
- Available online (Apache licence)
- Under testing for use in production at facebook.