USENIX HotStorage2016

Finding Consistency in an Inconsistent World: Towards Deep Semantic Understanding of Scale-out Distributed Databases

Neville Carvalho, Hyojun Kim, Maohua Lu, Prasenjit Sarkar, Rohit Shekhar, Tarun Thakur, Pin Zhou
DATOS IO

Remzi H. Arpaci-Dusseau
University of Wisconsin-Madison

NECST (***N****ext-generation* ***E****ventually* ***C****onsistent* ***ST****orage systems*)

cassandra

mongoDB

amazon DynamoDB

Why?

Big Data
Internet Scale App.
(IoT, Mobile)

Scale and availability is more important than ACID

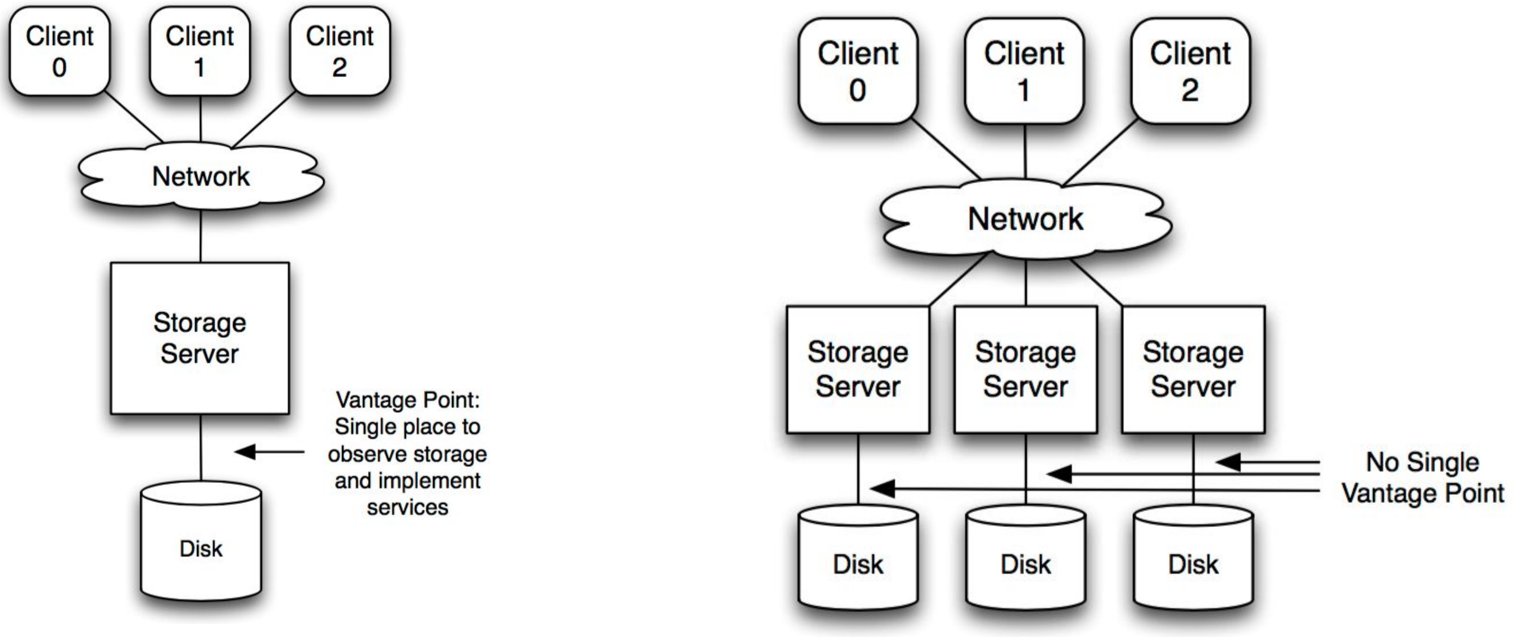
Problem

How to build efficient **backup** and **restore** tools for **NECST** (Next-generation Eventually Consistent STorage systems)

Does NECST require backup?

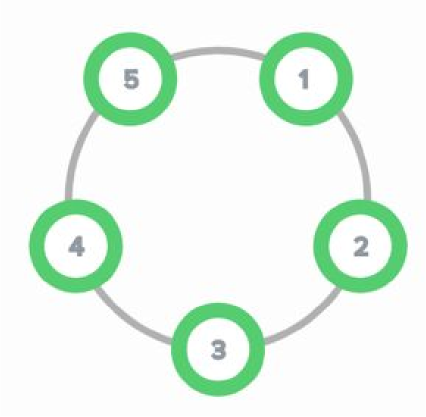
- NECST systems are highly available
  - Data replication, Multi-DC support
- Enterprise organizations have a fundamental need to restore and access particular versions of data from different points in time
  - Operational errors (a.k.a. “Fat fingers”)
  - Operation historian (government regulations)

Why NECST system backup is difficult

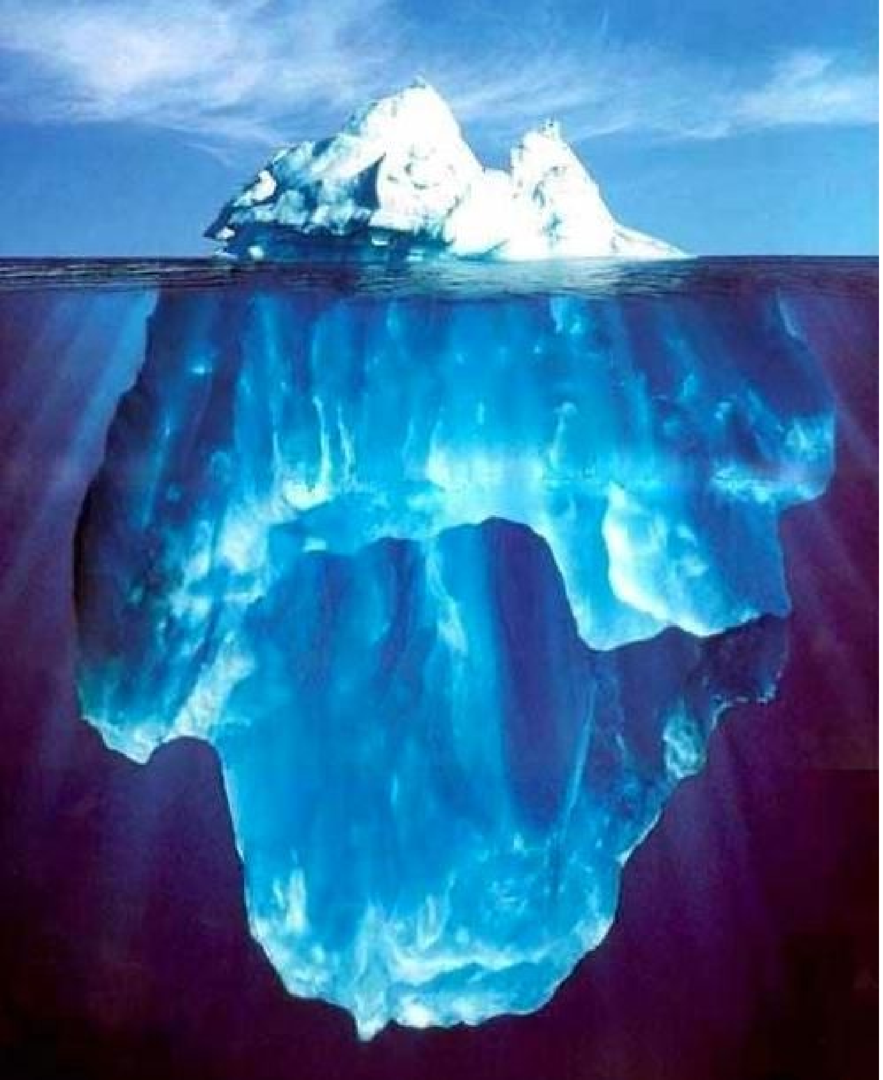


Single node snapshot vs. Distributed system snapshot

Orchestration is needed for backup and restore



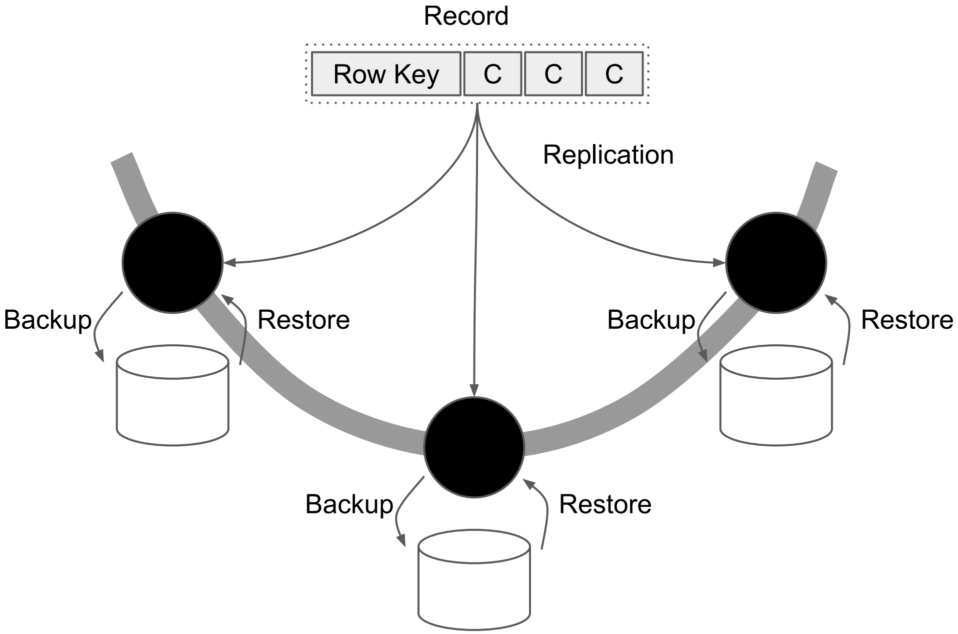
plus, failure handling
plus, topology change support



There are bigger problems

Example: existing backup solution for Cassandra

- Per-node backup & recovery
  - The state of each node can be captured by snapshot command
- Issues
  - Inconsistent backup
  - Topology change
  - Redundant data



Problems of the “per-node” backup approach

- Backup space waste problem
  - Replicated data (normally 3 copies) consumes more space (3x) in a backup

    (if backup files are uploaded to an object store like Swift, space consumption will be 9x)
- Inconsistency problem
  - Creating a consistent snapshot from an eventually consistent DB system
  - Repair operation is very expensive

    (imagine running *fsck* for multiple file systems having terabytes of data)

Goals

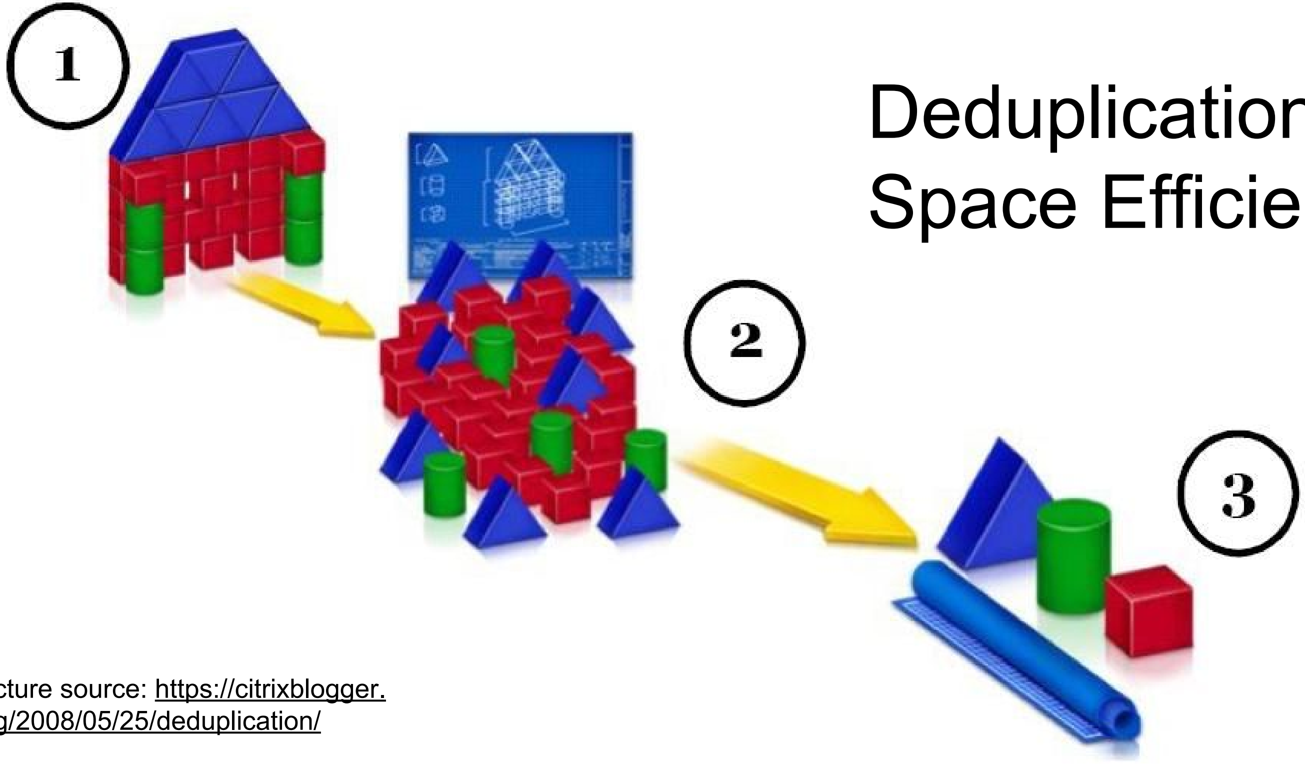
1. Quorum reconciliation (*consistency*)

2. Redundant-copy detection (space efficiency, *deduplication*)

3. Configuration-oblivious backup and restore (*topology change*)

4. *Orchestrated* backup and restore with *failure handling*

Deduplication: Space Efficient Backup



Picture source: https://citrixblogger.org/2008/05/25/deduplication/

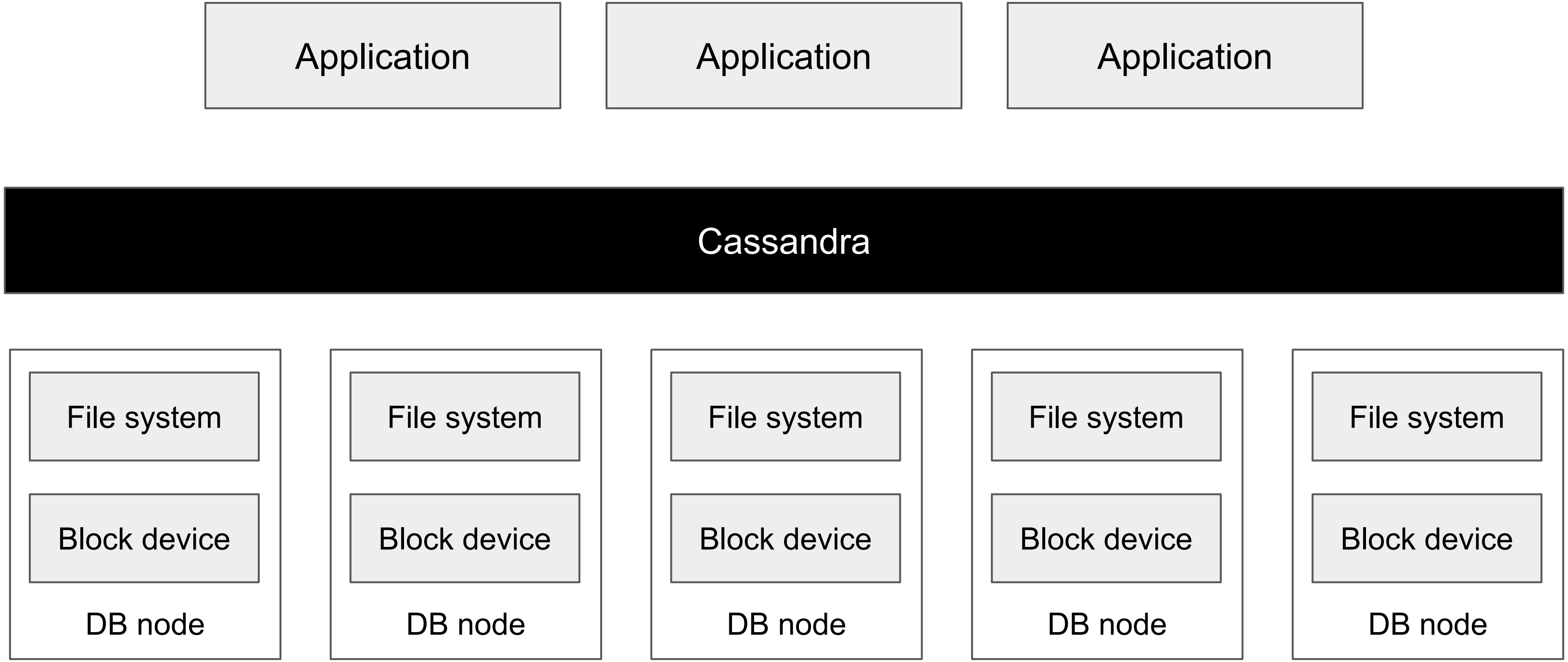
Deduplication

Replace redundant backup data with pointers to shared copy

- Source vs. Target deduplication
- Inline vs. Post-processing deduplication
- File vs. Block level deduplication
- Global deduplication

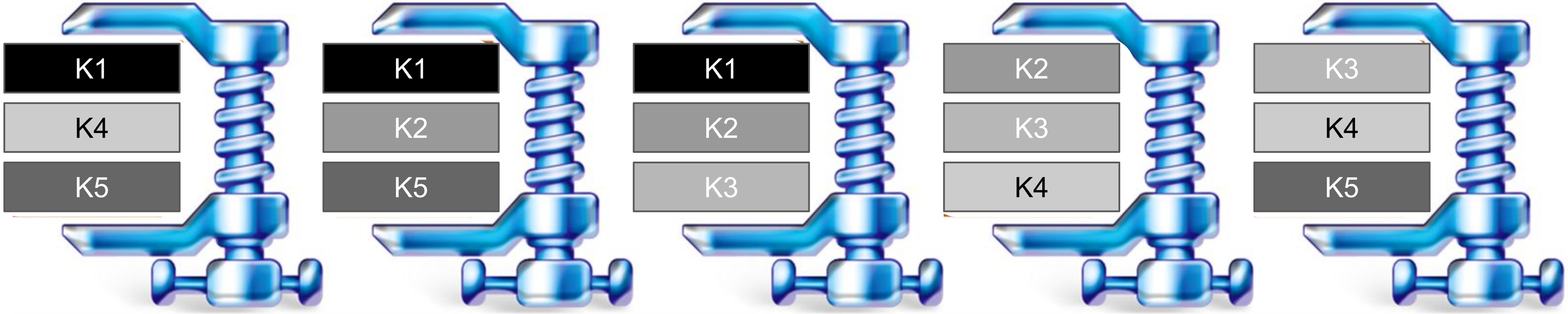
Will existing deduplication solutions work for Cassandra?

Cassandra: Replica exist across nodes



Distributed system based on shared nothing storage

Cassandra: Row based replication + Compression



DB Node 1	DB Node 2	DB Node 3	DB Node 4	DB Node 5
K1	K1	K1	K2	K3
K4	K2	K2	K3	K4
K5	K5	K3	K4	K5

Very low chance to find identical chunks from Cassandra data files

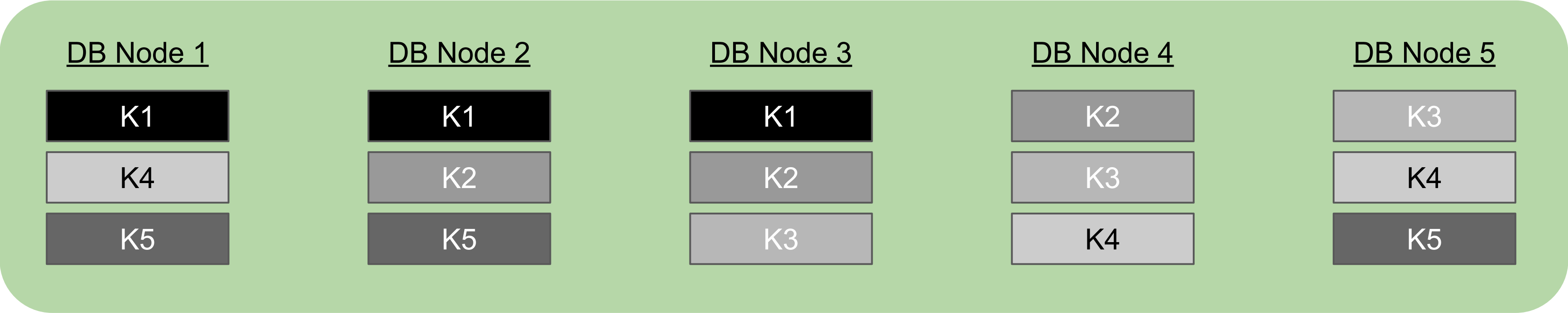
Source: Internet

Consistent Backup

Levels of backup consistency

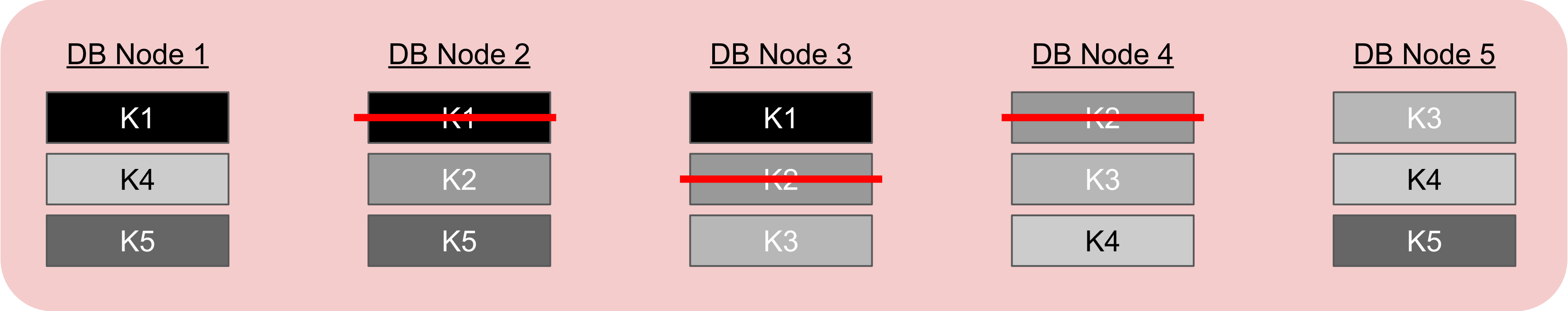
- Inconsistent backup
  - Simple file copy operation
- Crash-consistent backup
  - Backup’s data saved within the same moment of time
  - Memory content and pending I/O will be lost
- Application-consistent backup
  - Capture all data in memory and all transactions in process
  - Quiesce the database application, flush its memory cache, complete all its writes in order and then perform the backup

Consistent status



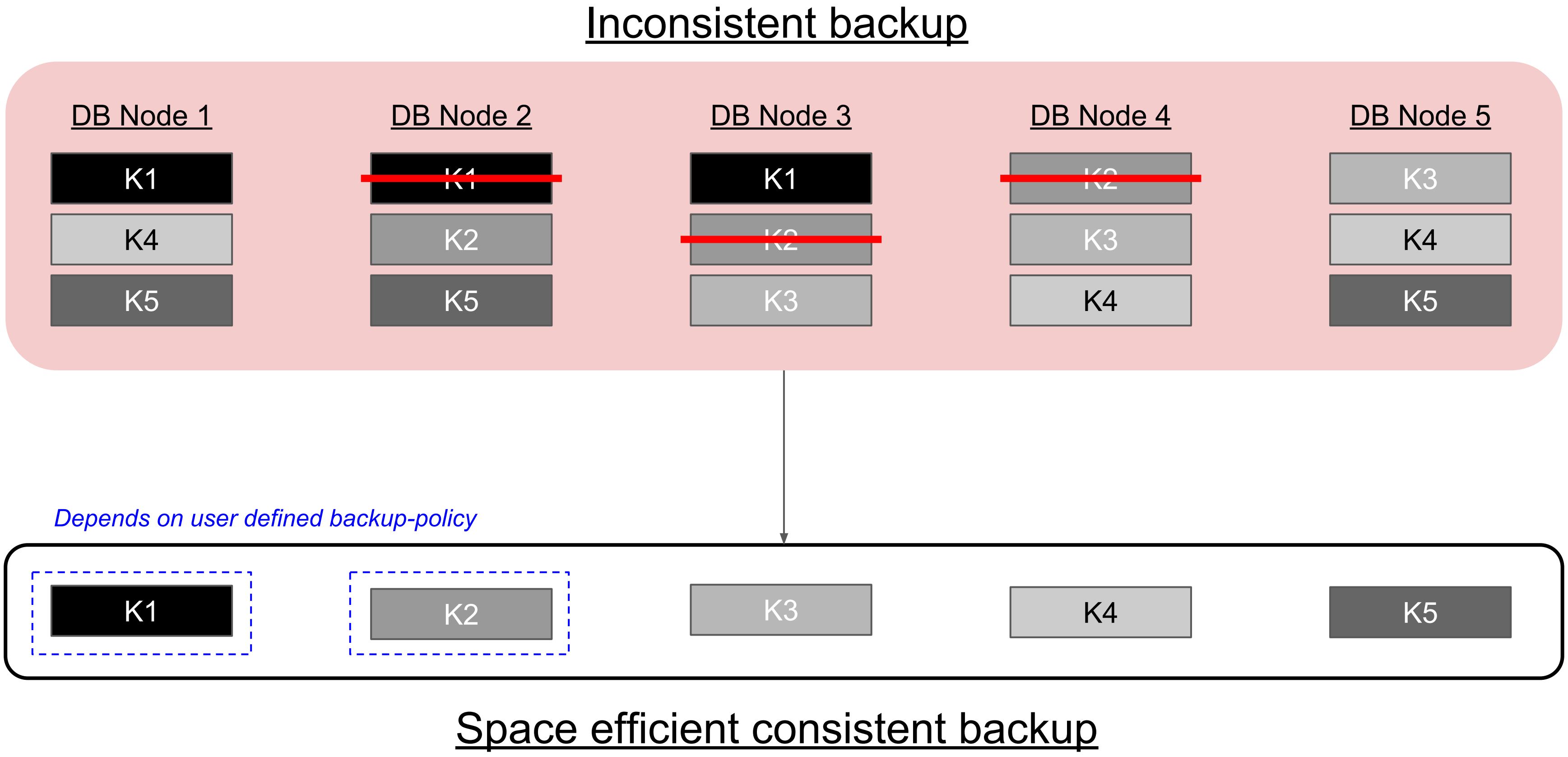
DB Node 1	DB Node 2	DB Node 3	DB Node 4	DB Node 5
K1	K1	K1	K2	K3
K4	K2	K2	K3	K4
K5	K5	K3	K4	K5

Inconsistent status



DB Node 1	DB Node 2	DB Node 3	DB Node 4	DB Node 5
K1	~~K1~~	K1	~~K2~~	K3
K4	K2	~~K2~~	K3	K4
K5	K5	K3	K4	K5

Space efficient consistent version



Two key building blocks

- Deep Semantic Understanding

- Efficient data processing algorithm

Conclusions

- NECST system is becoming important component of the enterprise datacenter.

- NECST backup problem has been introduced: three key parts
  - Backup and restore orchestration
  - Quorum reconciliation for consistent backup
  - Redundant copy detection for space-efficient backup

- Our mission:
  NECST storage management is as easy and effective tomorrow as classic storage management is today

DATOS IO

http://datos.io

Email: info@datos.io

Thank you