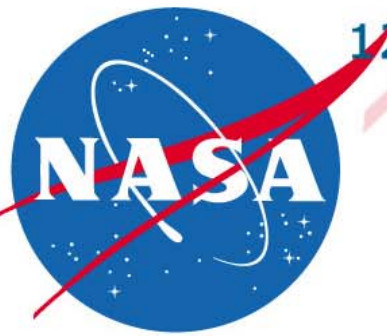
Duplicate Data Elimination in a SAN File System

Bo Hong

University of California, Santa Cruz
Jack Baskin School of Engineering
1156 High Street, Santa Cruz, California 95064
Tel: +1-831-459-4458
hongbo@cse.ucsc.edu

NASA/IEEE MSST 2004
12th NASA Goddard/21st IEEE Conference on
Mass Storage Systems & Technologies
The Inn and Conference Center
University of Maryland University College
Adelphi MD USA
April 13-16, 2004



Co-authored with:

Demyn Plantenberg
IBM Almaden Research Center
Darrell D.E. Long
University of California, Santa Cruz
Miriam Sivan-Zimet
IBM Almaden Research Center

UC Santa Cruz

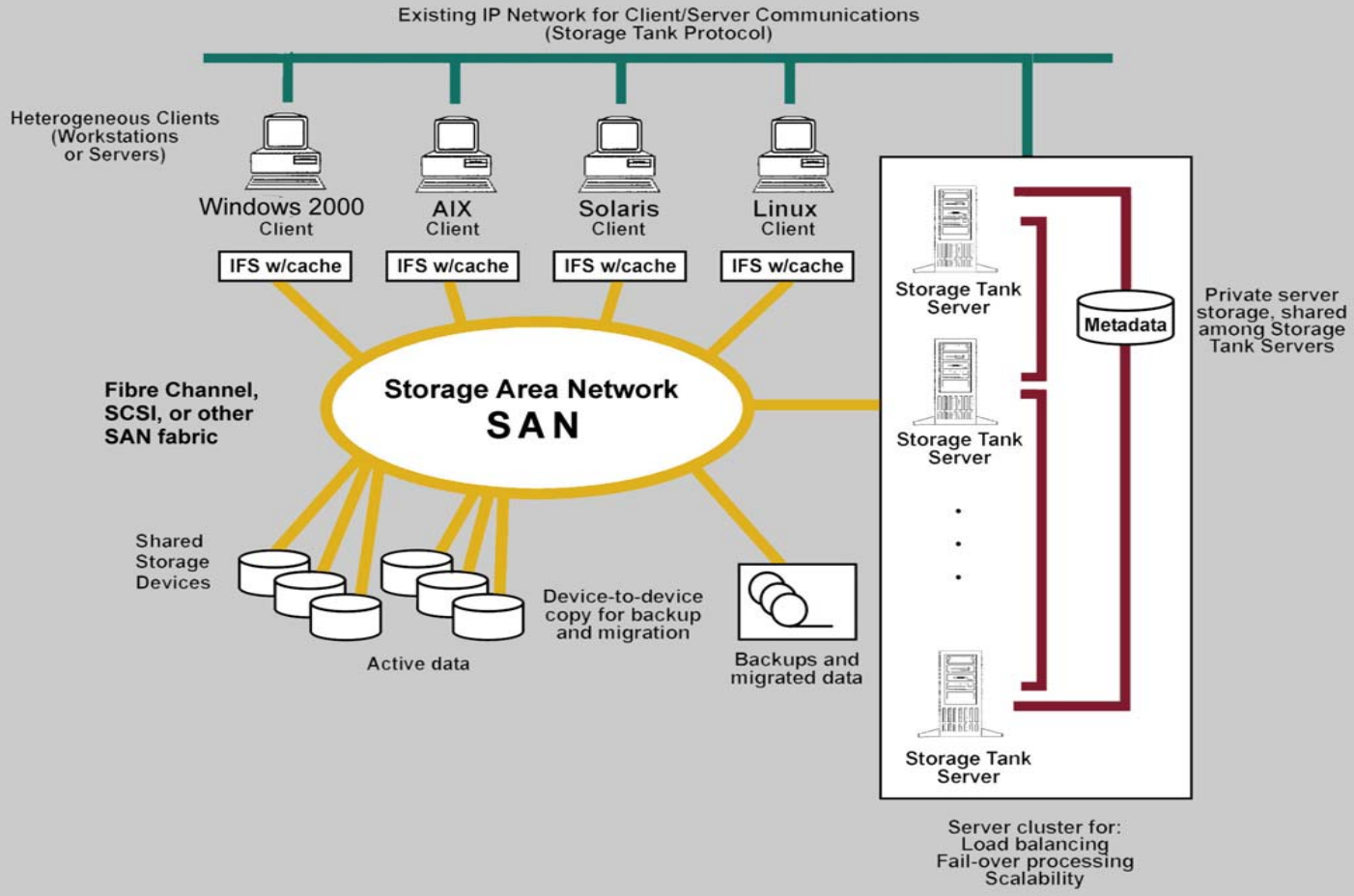
Motivation

- Duplicate data is ubiquitous and generated …
  - intentionally
  - unconsciously
  - systematically
- Disk is cheap but not storage

Goals

- Reduce duplicate data and achieve better storage efficiency
- Support online storage system
- Minimize impact on system performance
- Keep transparent to users

Storage Tank Overview



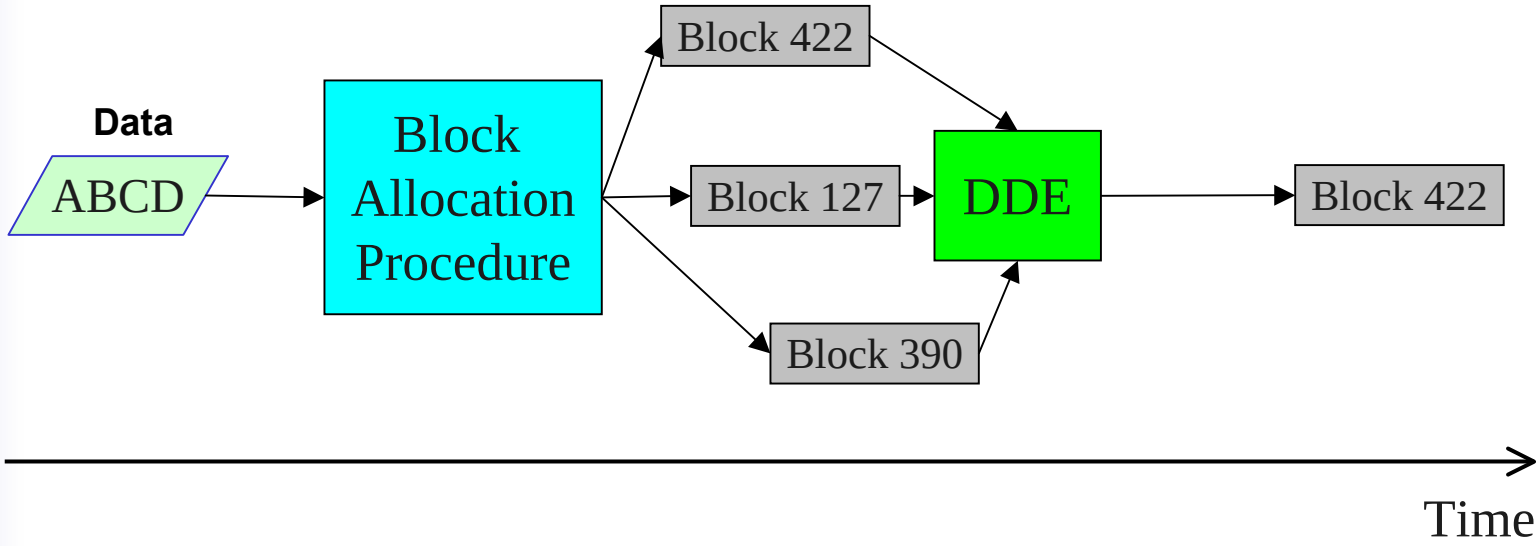
What's the difference from …

- Venti
  - Archival storage
    - Performance is less of a concern
    - Data is immutable
  - Blocks are addressed by the hashes of their contents
  - Write-once policy
- LBFS
  - Reducing network transmission is more important
  - Variable-sized content-based chunk partition
- Microsoft Single Instance Store
  - The file-level duplication is known as a *priori*
- Delta compression
- On-line compression

DDE Design Highlights

Duplicate data elimination (DDE)

- Address-by-block
  - **After-effect effort**

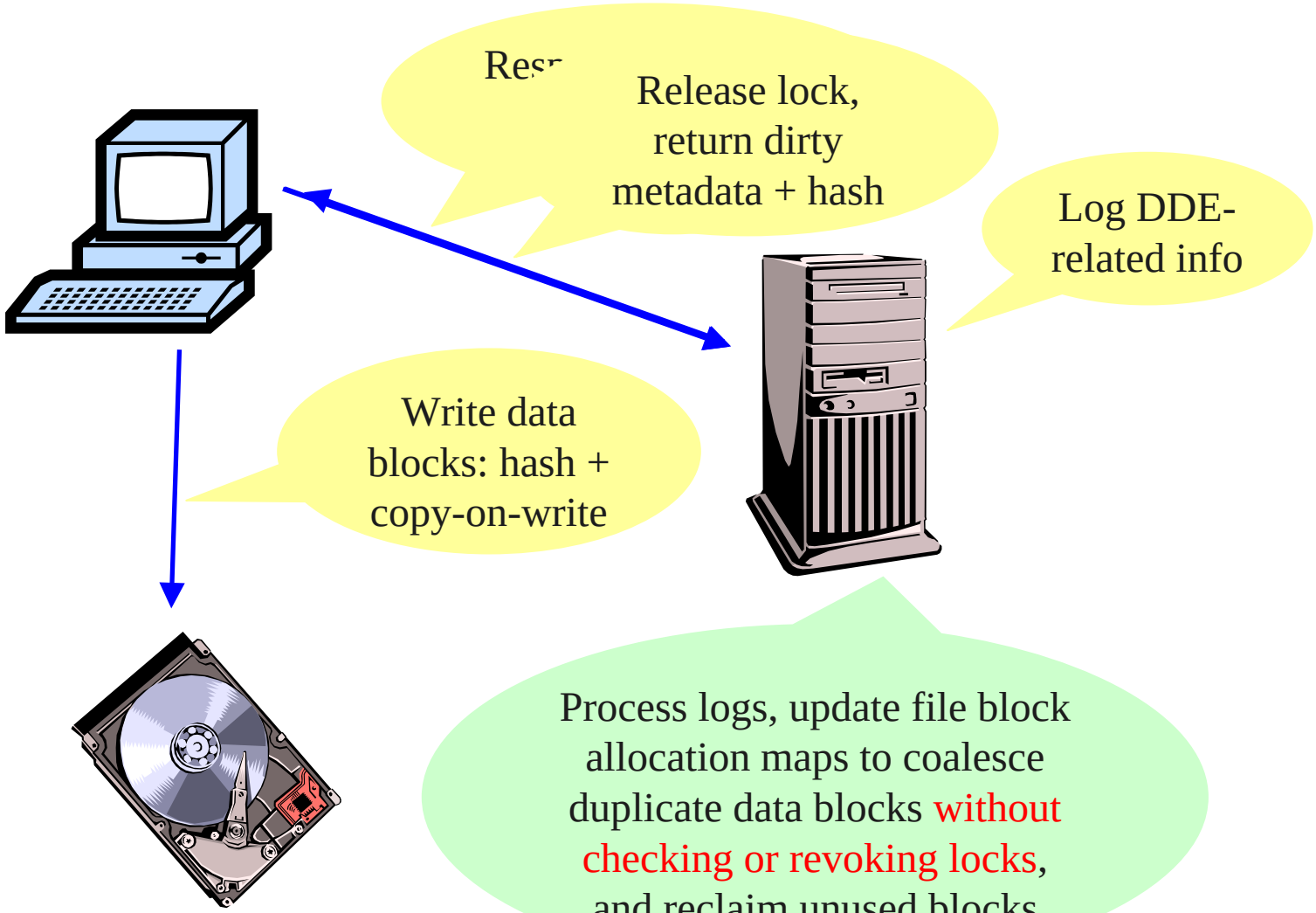


DDE Design Highlights

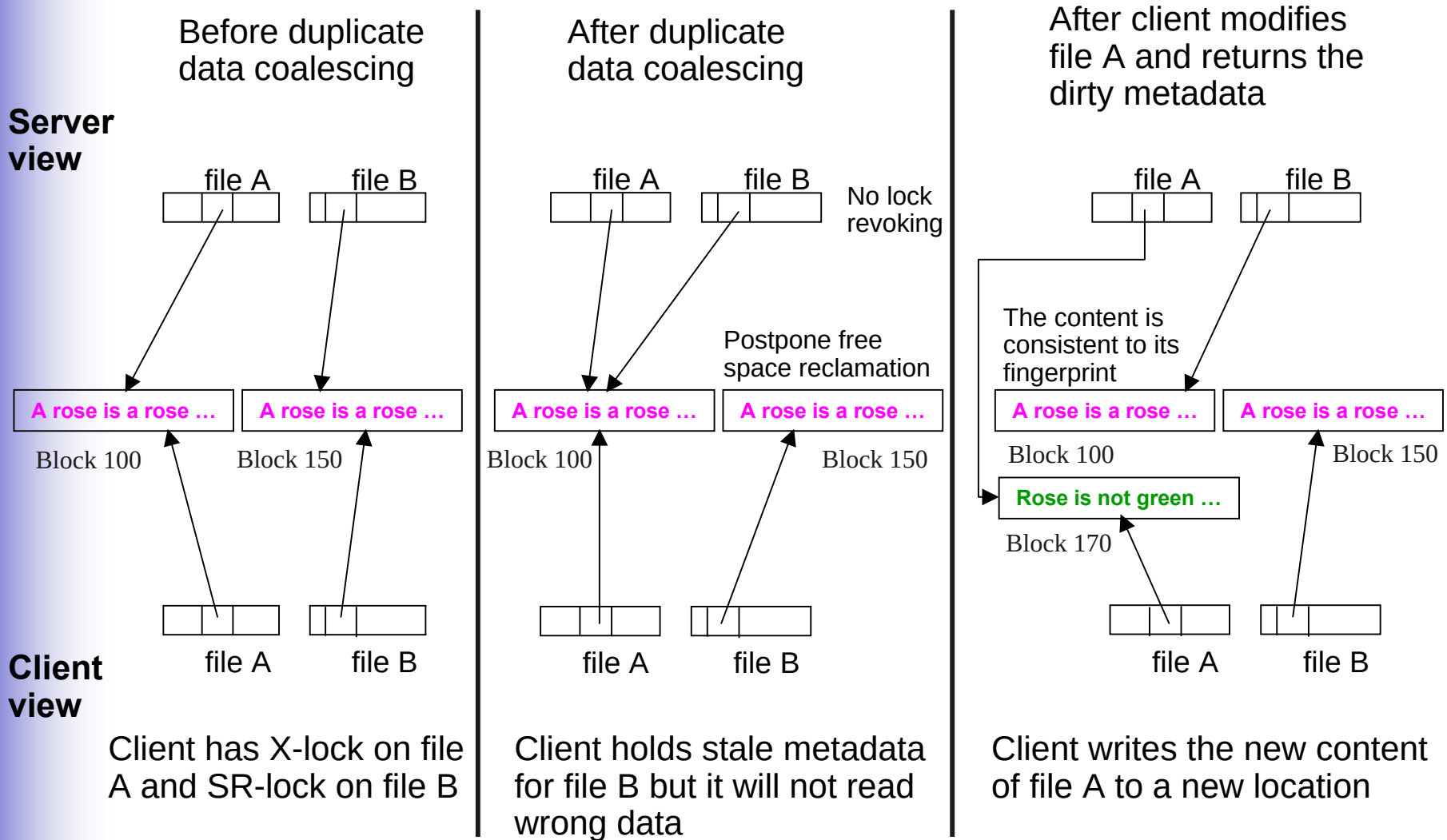
Duplicate data elimination (DDE)

- Address-by-block
  - **After-effect effort**
- Best effort
  - Only operate as a background process
- Block-level content hashing (160 bit Sha-1)
- Copy-on-write (COW)
  - Guarantee consistency between data and data hash
- Lazy update
  - Lazy lock revocation
  - Lazy free block reclamation
  - Minimize system performance overhead

How DDE works?

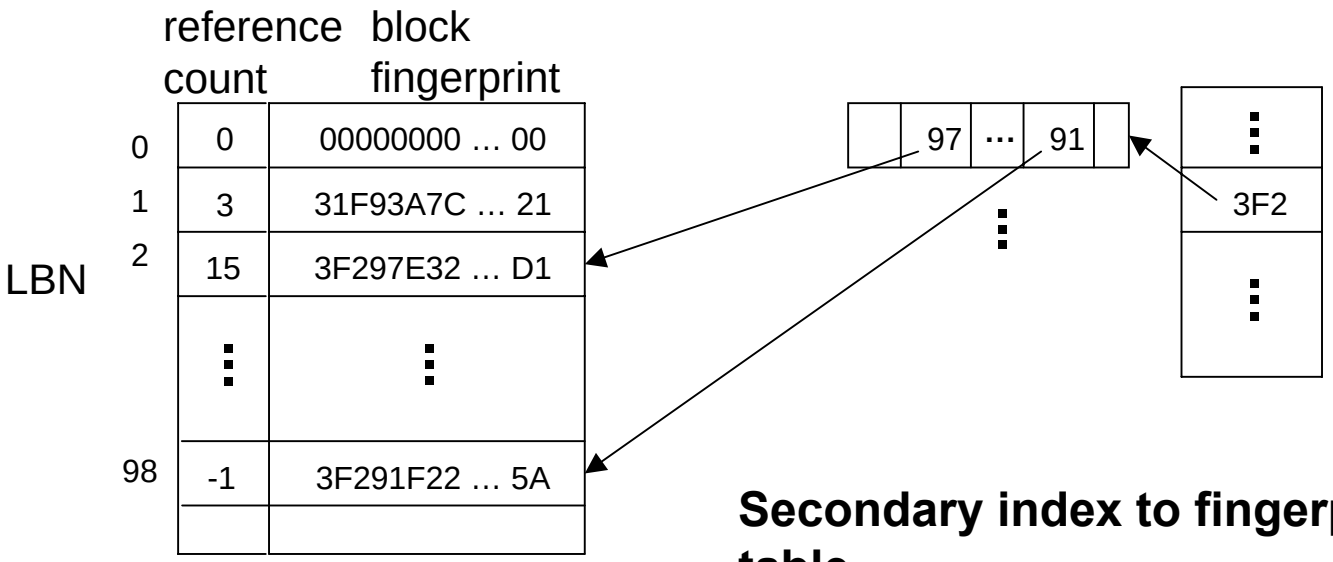


Correctness of DDE



Storing and Retrieving Block Metadata

- A block has two attributes – block metadata
  - Reference count – >= 1, 0, -1
  - Fingerprint – valid only if reference count >= 1



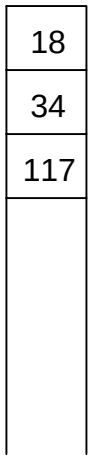
Fingerprint table

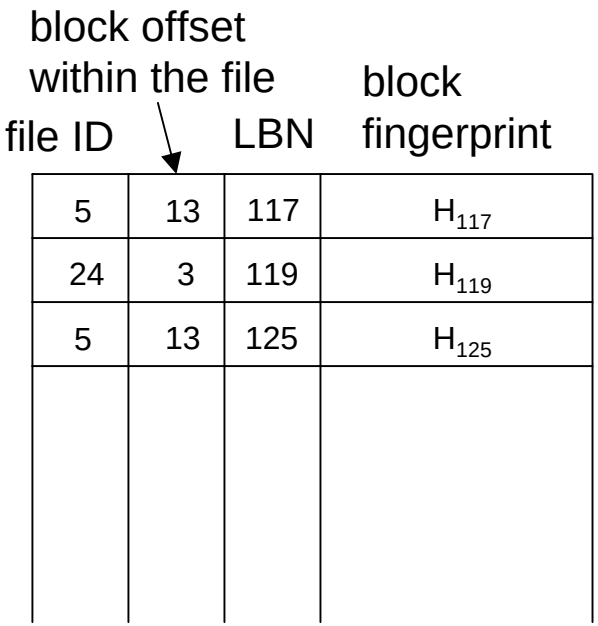
One entry per block. Organized to facilitate comparisons under **sequential** block accesses

Secondary index to fingerprint table

Implemented as a hash table using a portion of the fingerprint as its key. Organized to facilitate **random** fingerprint lookups

Logging Recent Activities on Server





Dereference log (semi-free list)

Log the addresses of blocks recently being deleted or freed due to COW

New fingerprint log

Log recent write activities by clients

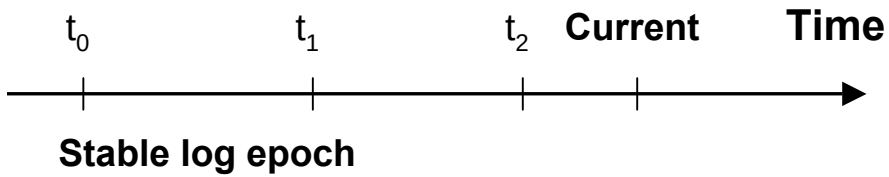
Detecting and Coalescing Duplicate Data

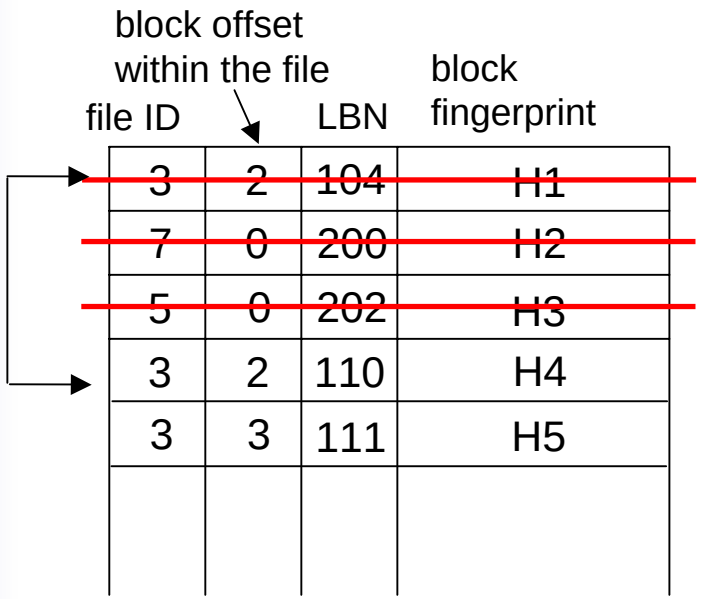
- Add the fingerprint H_A of block A to the fingerprint table. Block A belongs to file F.
  - Check the existence of H_A in the fingerprint table
  - Check the validity of the primary block B ($H_B = H_A$) in the fingerprint table
  - Check whether block A is still referenced by file F
  - Make file F refer to block B instead of A; set ref_A = -1; ref_B ++

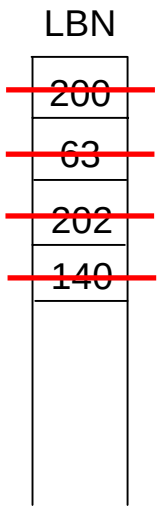
NO lock checking and revocation

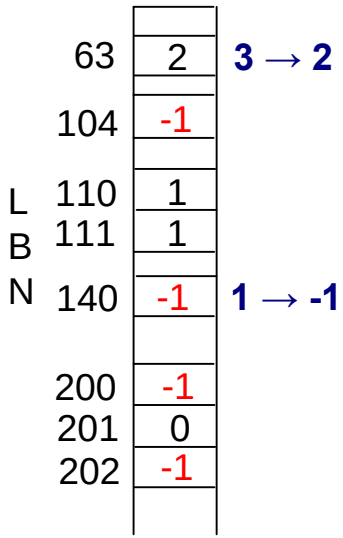
Optimization – Log Preprocessing

- Periodically checkpoint the logs – log epoch









Reclaim Free Space

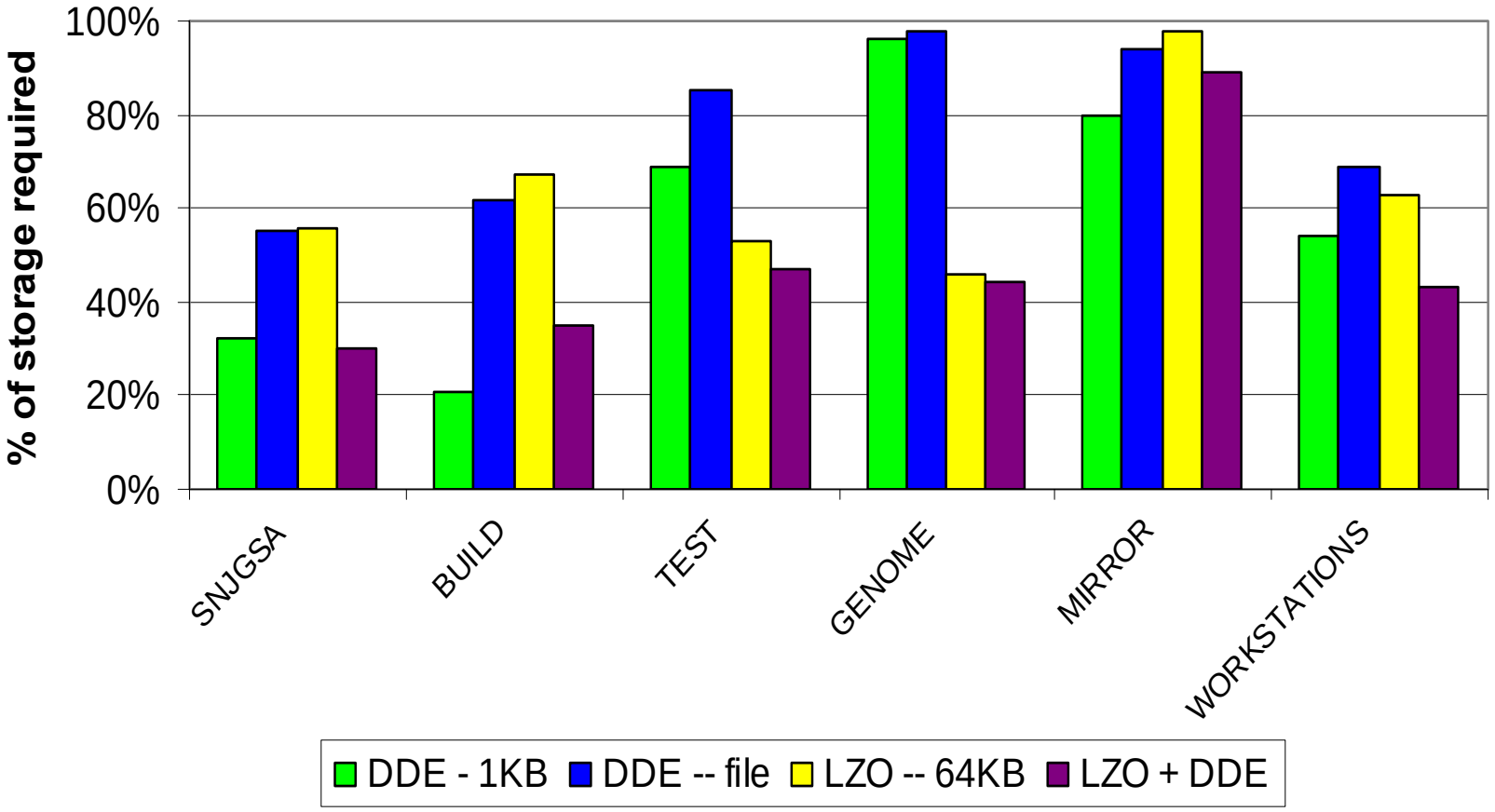
Scan the reference count table in the background

- Collect blocks with reference counts –1; set their reference counts to be 0
- At some particular time (e.g. midnight) revoke all data locks and free those blocks

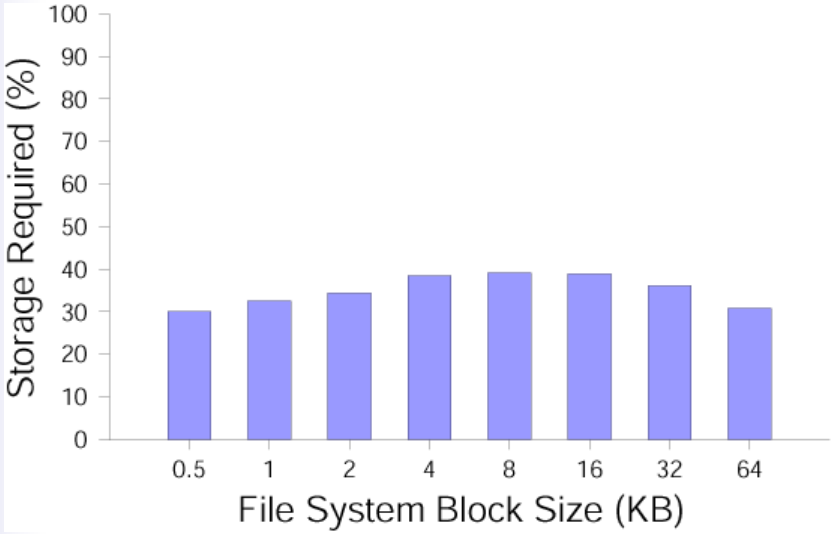
Case Studies – Data Sets

Name	Description	Size (GB)	Number of files
SNJGSA	File server used by a development team	57	661,729
BVRGSA _BUILD	File server used by the development team for code build	344	2,393,795
BVRGSA _TEST	File server used by the development team for testing	215	115,141
GENOME	Human being genome data	348	889,884
LTC_MIR ROR	Local mirror of installation CDs for different Linux versions	261	241,724
WORKST ATIONS	Aggregation of ten personal workstations	123	879,657

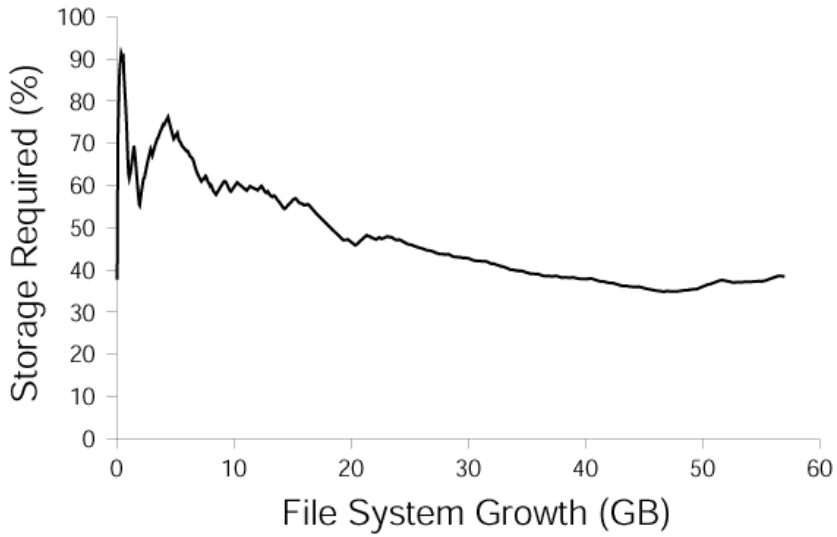
Overall Results



Detailed Study in SNJGSA



Percentage of storage required after DDE under different block sizes



Percentage of unique blocks in a simulated growing file system

- The additional space saving potential of smaller blocks is modest – 5%
- DDE can continuously improve storage efficiency as a data set grows

Future Work

- Implementation
  - Performance measurement
- Data duplication characterization
- Duplicate data coalescing policies
- Alleviate extra allocation cost due to COW
  - Private pool of disk space managed by clients
  - Pre-allocation policy on server
- Client hash cache – a history of write activities
  - No actual I/O when cache hits
- Data migration
- Data integrity check

Conclusions

- Duplicate Data Elimination (DDE)
  - Target – online file system
  - Enabling techniques
    - Block-level content-based hashing
    - Copy-on-write
    - Lazy lock revocation
    - Lazy free space reclamation
- 20-79% of storage savings in some environments

Acknowledgements

- Our shepherd – Curtis Anderson
- Robert Rees, Wayne Hineman, and David Pease from IBM Almaden Research Center
- Scott Brandt, Ethan Miller, Feng Wang, and Lan Xue from Univ. of California, Santa Cruz
- Vijay Sundaram from Univ.of Massachusetts, Amherst
- Terrence Furey and Patrick Gavin from the bioinformatics group of UCSC

Questions

Storage Tank – IBM Almaden Research Center

http://www.almaden.ibm.com/StorageSystems/file_systems/storage_tank/index.shtml

Storage Systems Research Center (SSRC), University of California, Santa Cruz

http://ssrc.cse.ucsc.edu/

Contact author: Bo Hong

http://www.cse.ucsc.edu/~hongbo

Thank you!

Backup Slides

Key Storage Tank Features

- Separated metadata and data management
  - Servers are not in the data path
- Disks provide simple block storage
- File sets, storage pools, and arenas
  - A file set is a sub-tree of the global namespace
  - A storage pool is a collection of one or more volumes
  - An arena provides the mapping between a file set and a storage pool
- Data lock granularity is per file
  - Exclusive (X), Shared-Read (SR), and Shared-Write (SW)
- Copy-on-write and read-only extents support Snapshot

Merging to Fingerprint Table