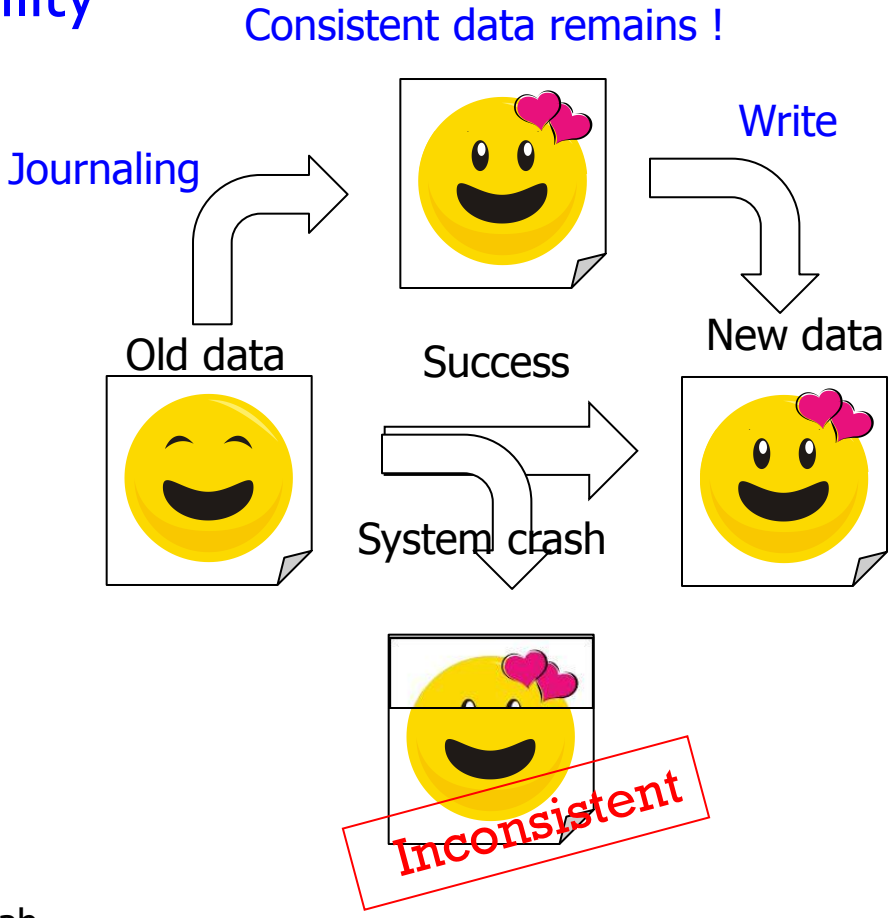
Shortcut-JFS: A New Journaling File System for Phase Change Memory

Apr 19, 2012

Eunji Lee
Seoul National University
(alicia0729@gmail.com)

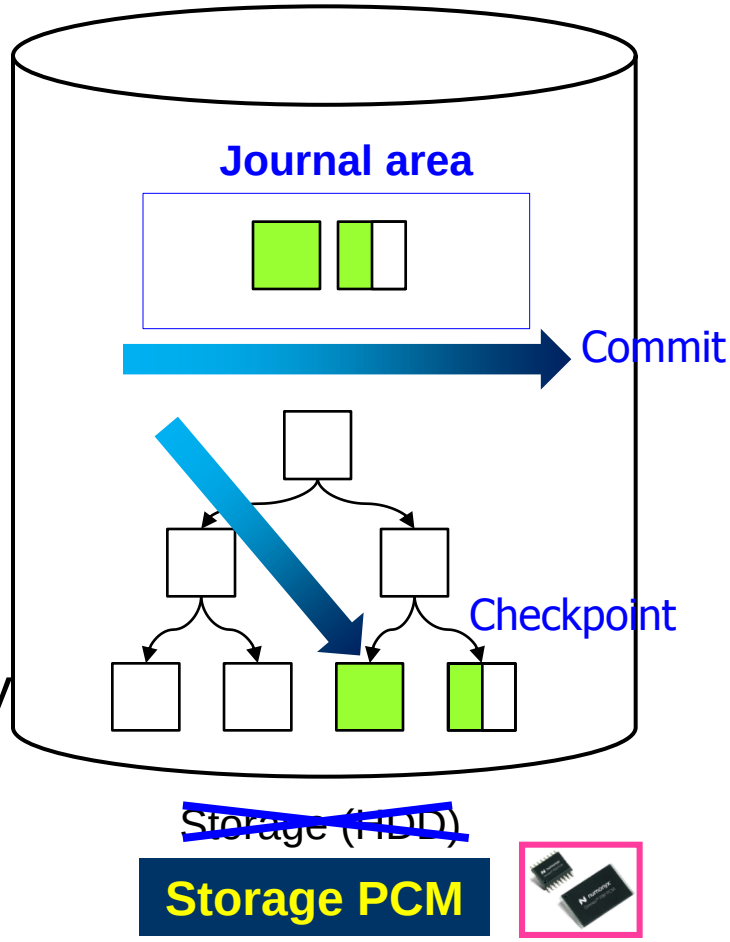
Journaling File system

- Widely used in modern computing systems to provide high reliability



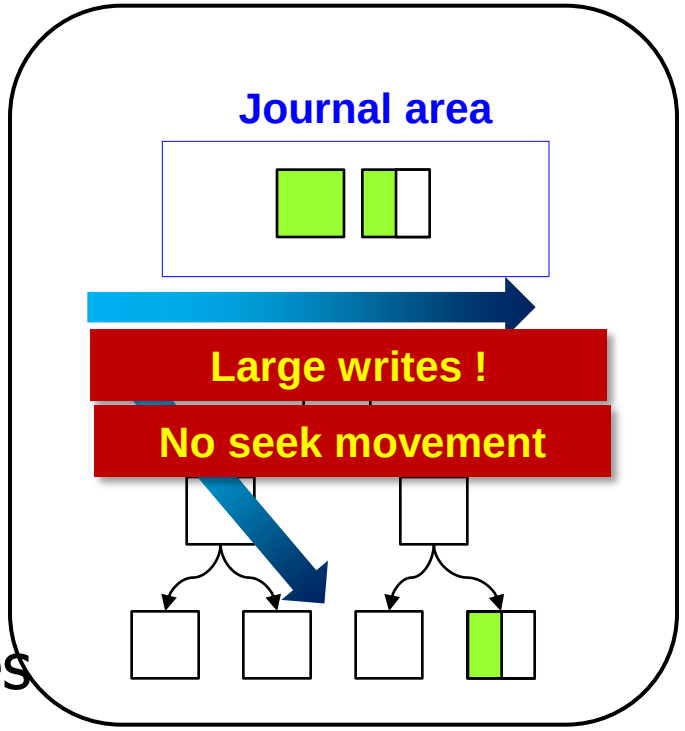
Pros and Cons of Journaling

- Journaling affects I/O performance
- Commit and Checkpoint
  - Write data twice
  - Reasonably efficient in HDD
  - Reduces seek movements
- What if using PCM as storage ?
  - High-speed and nonvolatile memory
  - Expected to be used as secondary storage with growing scalability



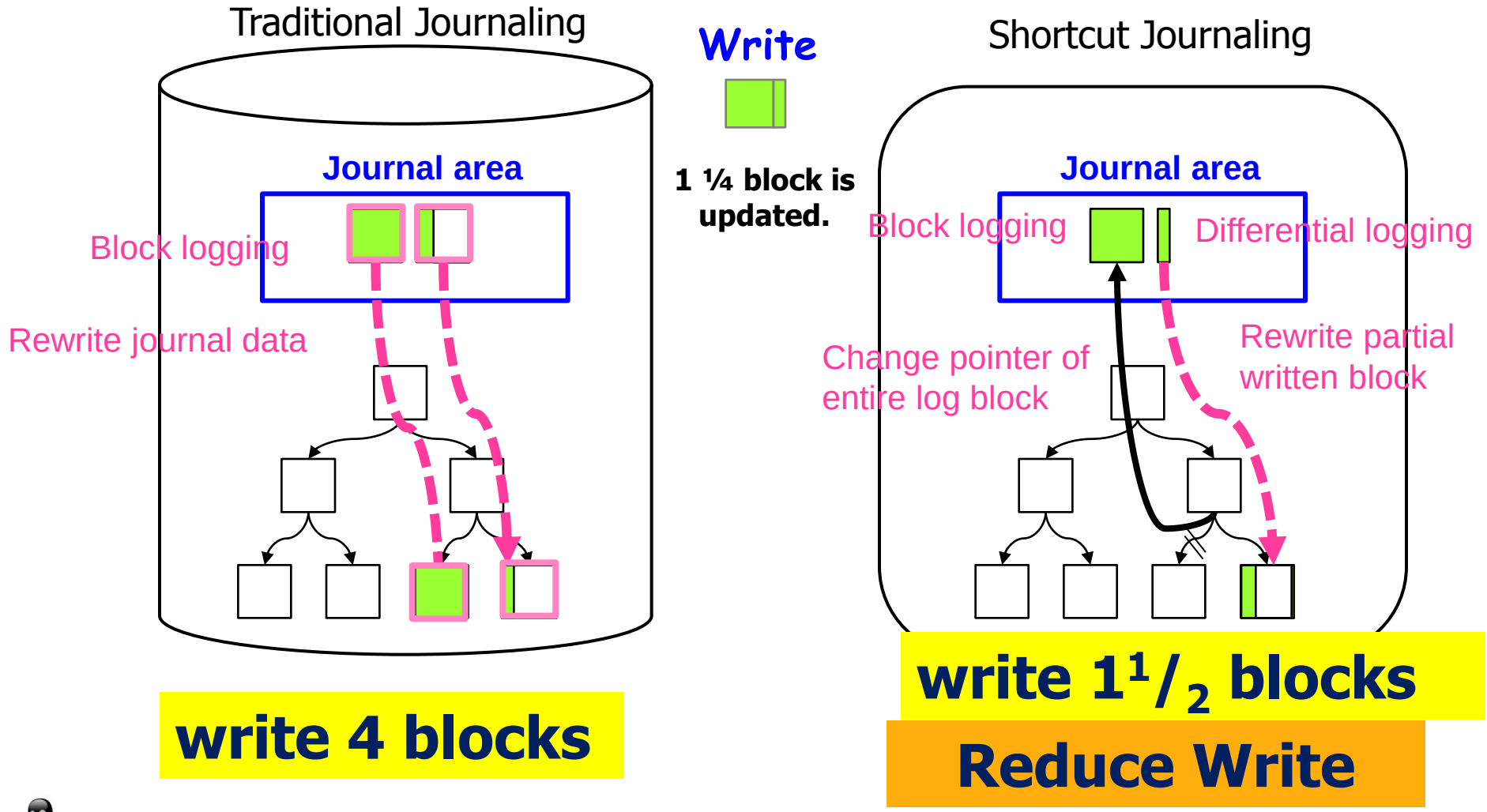
Pros and Cons of Journaling

- NOT efficient in PCM
  - Long write latency
  - Limited write endurance
- Need to reduce additional write amount in PCM
- Shortcut-JFS minimizes writes of journaling by taking advantages of PCM

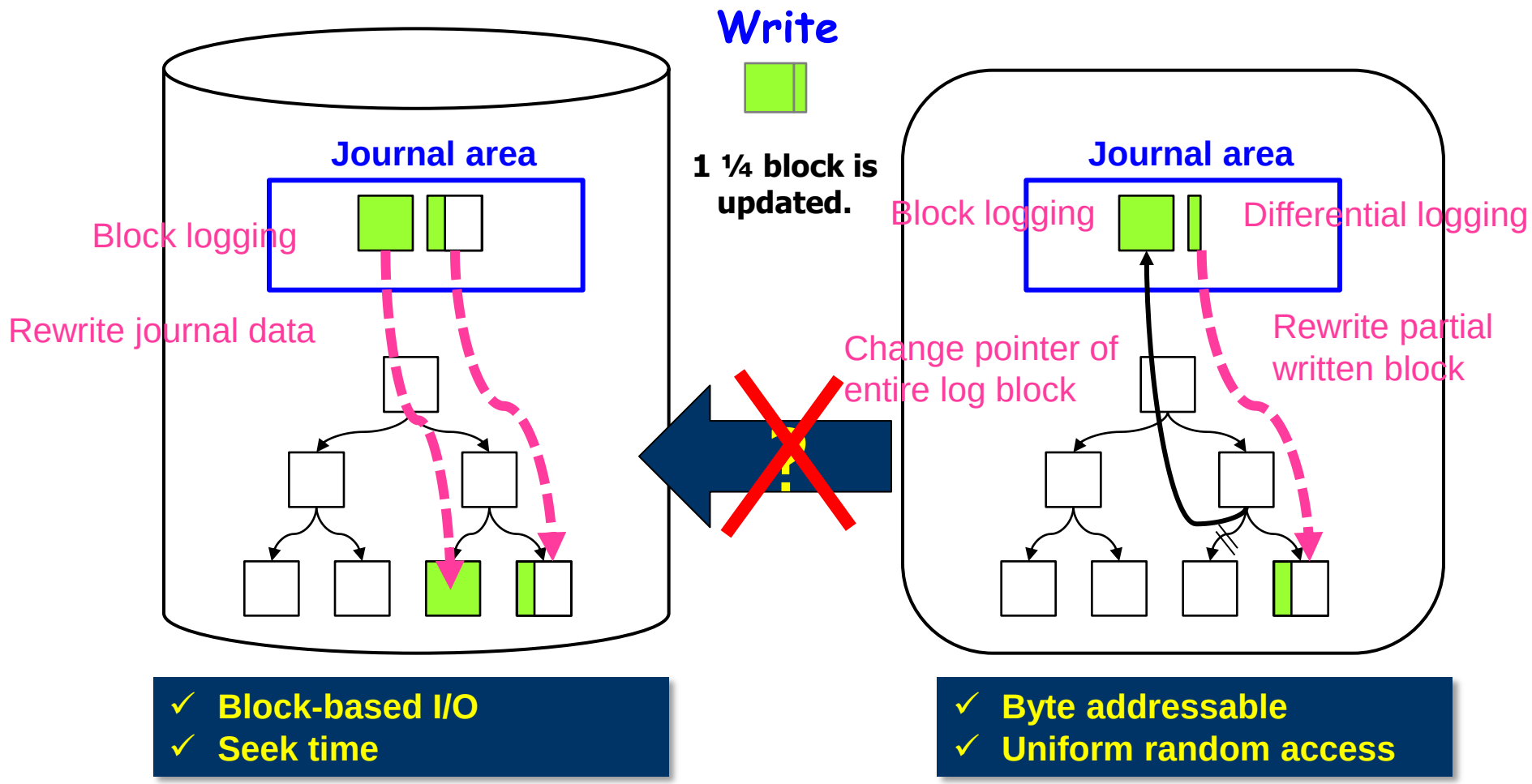


PCM Storage

Shortcut Journaling File system for PCM

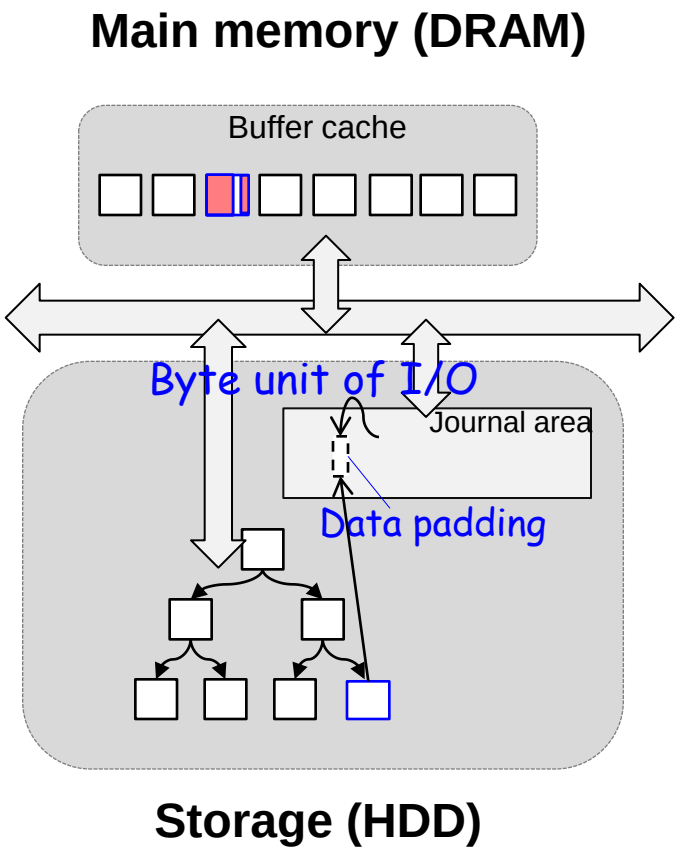


Shortcut Journaling File system for PCM



Data padding in Shortcut-JFS

- Adaptive Journaling Policy depending on write data size
  - When a write size < ½ of a block, Differential logging and Rewriting
  - Otherwise, Block logging and Pointer update
- One implementation Issue
  - Expect that byte unit of I/O interface transfers changed parts only
  - Need to fill out remaining part for block logging "Data padding"
  - Nested data updates incurs complicated data padding
  - Delay data padding until checkpointing



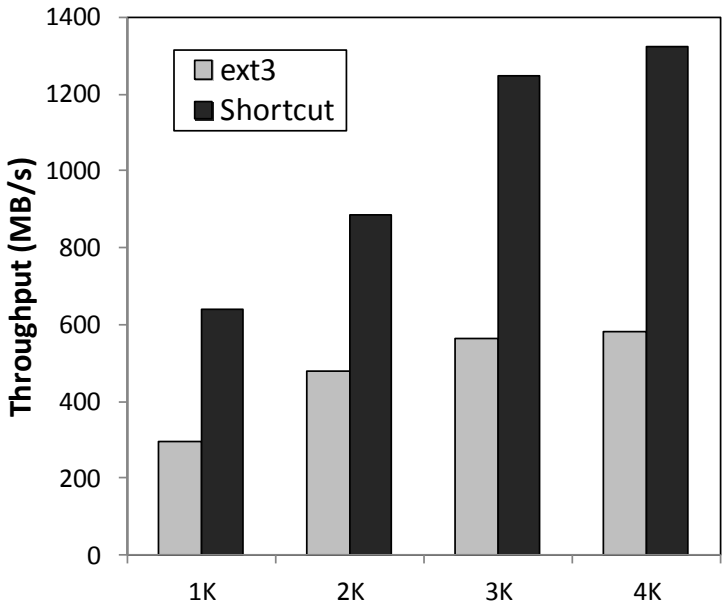
Experimental Setup

- Implemented on Linux

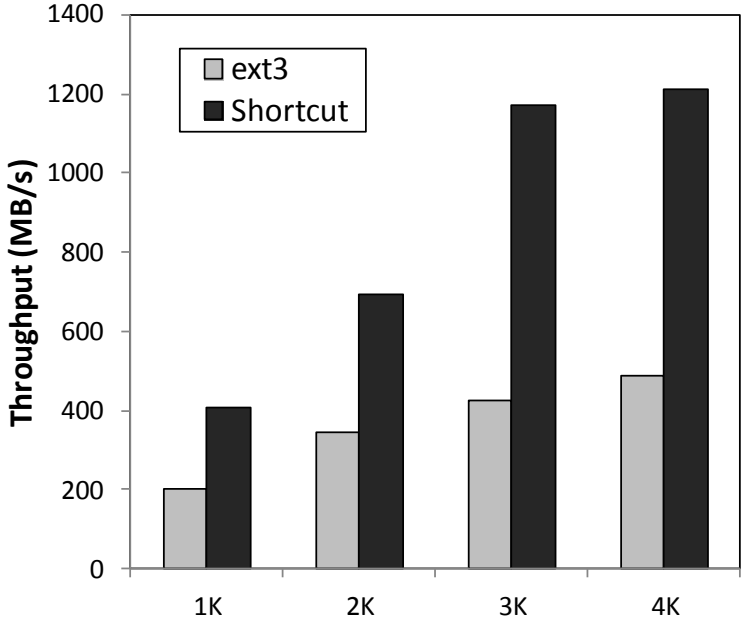
Processor Core	Intel Pentium(R) Dual-core E6500 @ 2.93 GHz
Main memory (ram-disk)	DDR-3 2GB
Operating System	Ubuntu-Server-10.08
Linux version	2.6.32-24
Modified Layer	ramfs based implementation

Experimental Results

- Iozone



(a) Sequential

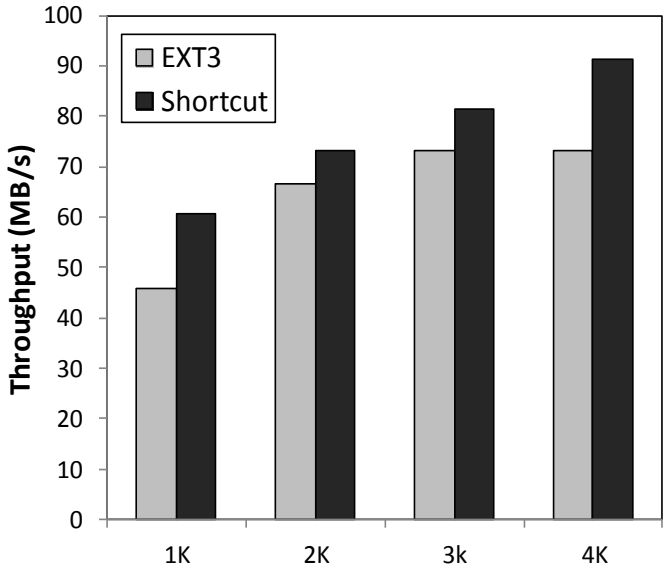


(b) Random

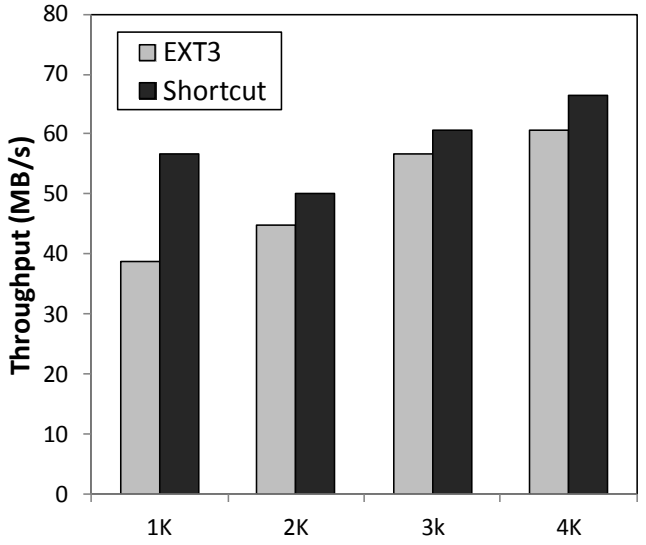
Shortcut-JFS improves I/O performance by 2.1x

Experimental Results

- Postmark



(a) Fileset 1



(b) Fileset 2

Shortcut-JFS improves I/O performance by 1.2x

Thank you~