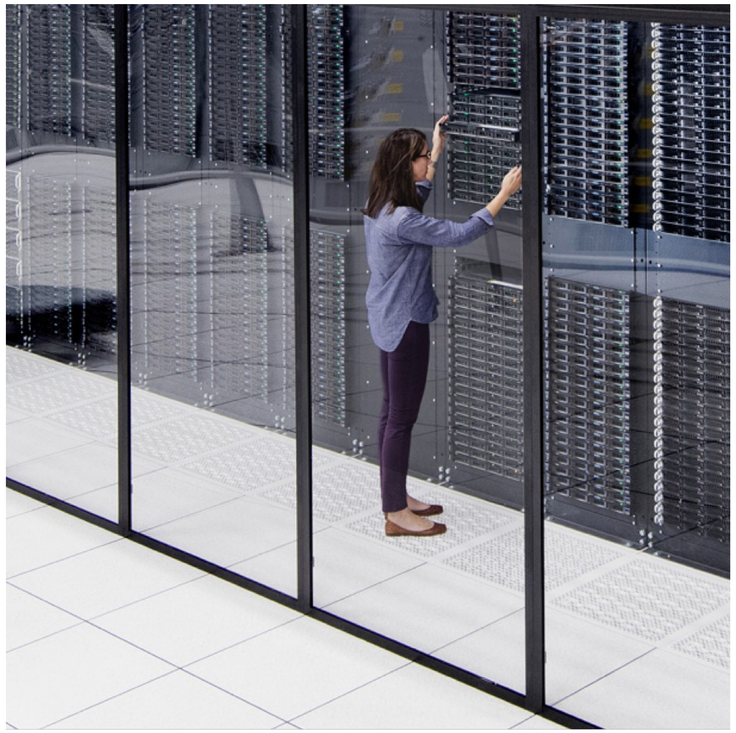
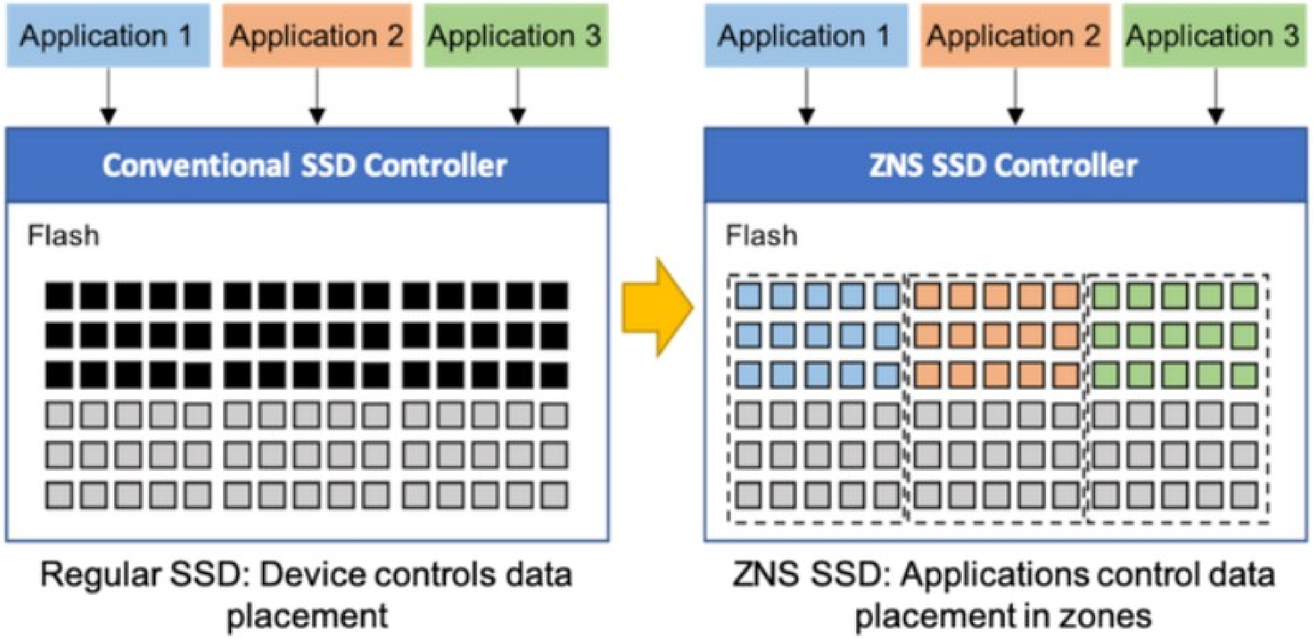
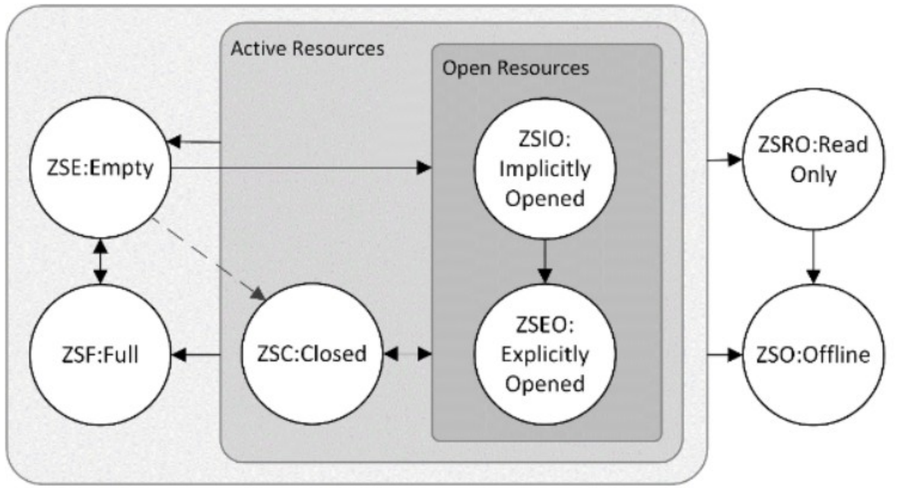
Building an efficient eco-system using ZNS SSD

Viacheslav Dubeyko
PhD, Linux kernel developer
IBM



What is ZNS SSD?





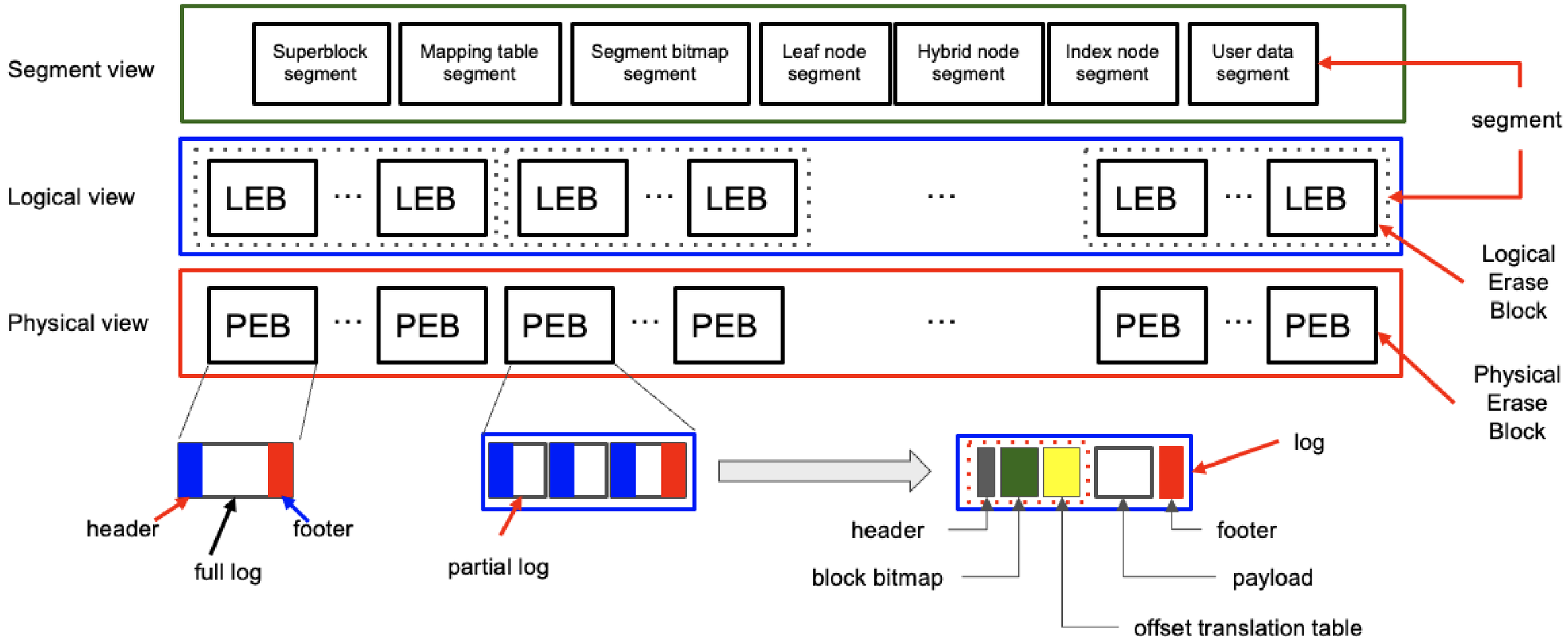
ZNS SSD advantages:

- **Sequential write pattern** reduces the need for internal data shuffling and **garbage collection**.
- ZNS SSDs can achieve **higher throughput**, **lower latency**, and better quality of service (QoS).
- ZNS SSDs reduces the need for **extensive over-provisioning**.
- ZNS SSDs utilize a standard interface defined by NVMe, which allows for a unified approach to storage across both SSDs and HDDs with Shingled Magnetic Recording (SMR).

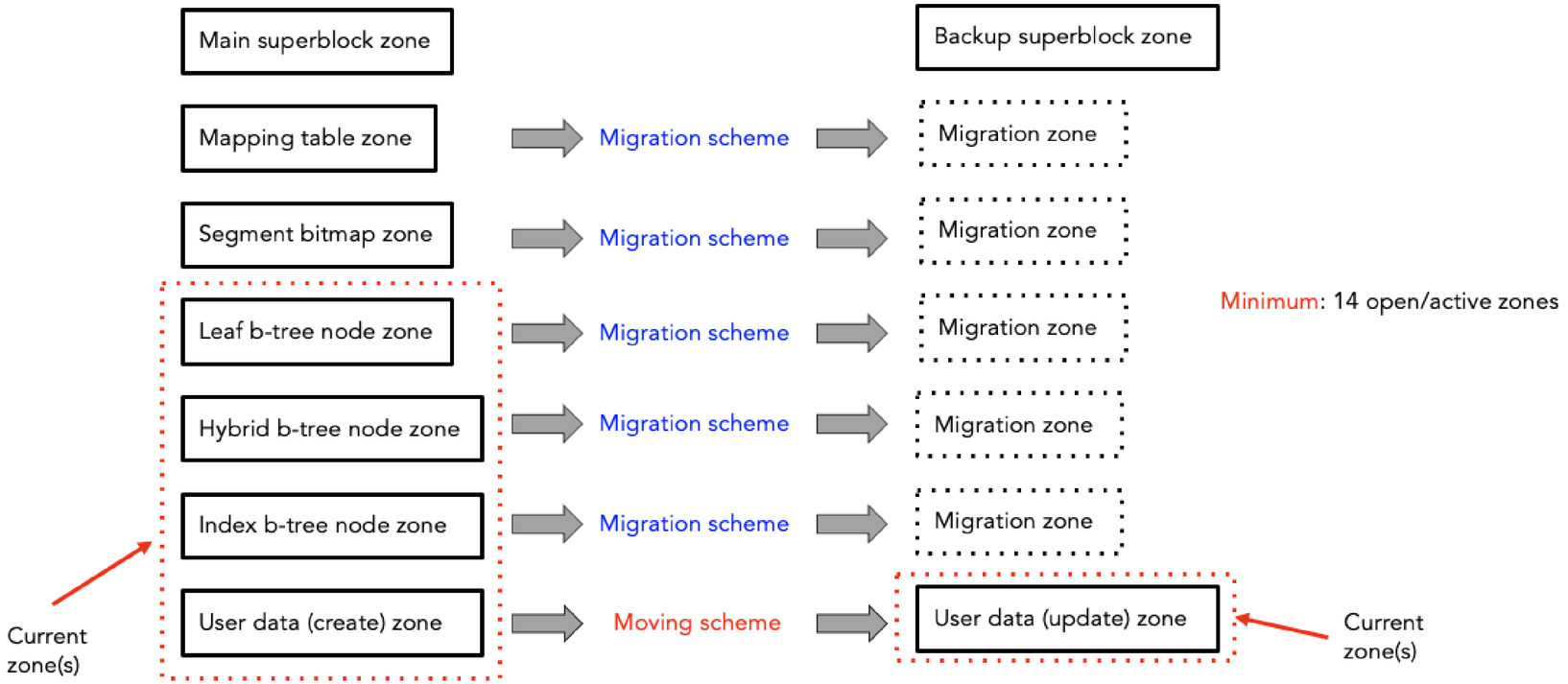
ZNS SSD limitations:

- **Huge zone size**.
- **Append-only mode** of write operations.
- **Limited number** of open/active zones.

SSDFS architecture (logical vs. physical view)



How to manage ZNS SSD limitations?



How to manage the huge zone size?

Compaction scheme

+

Big logical block size (32K, 64K)

+

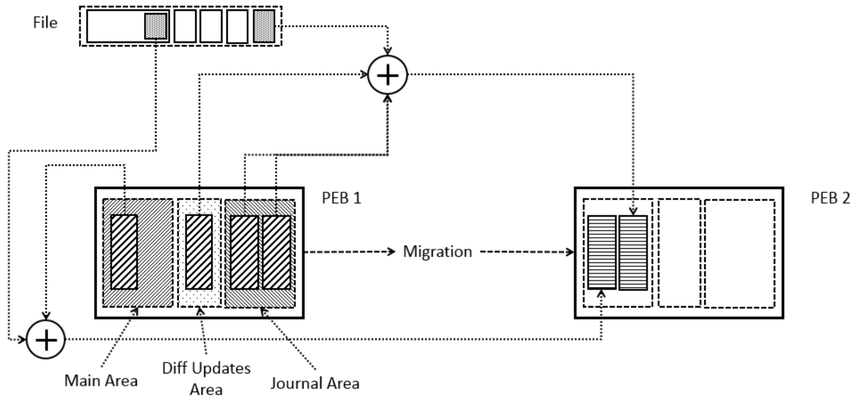
Inflation model

How to manage limited number of open zones?

Migration scheme → metadata

+

Moving scheme → user data



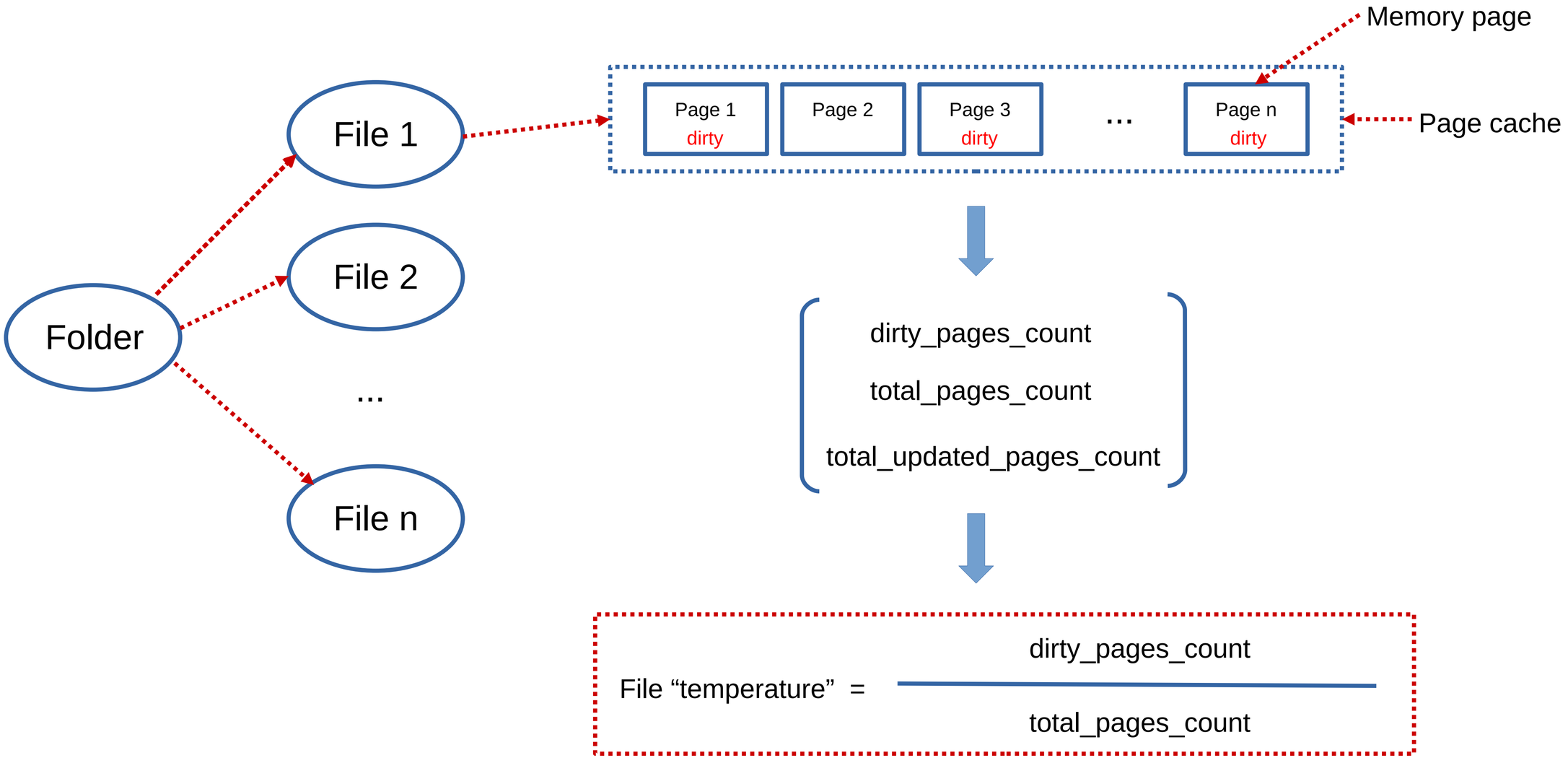
How to manage the append-only mode without introducing GC on file system side?

Copy-On-Write policy

+

Migration scheme

How data temperature can be employed for efficient data classification?



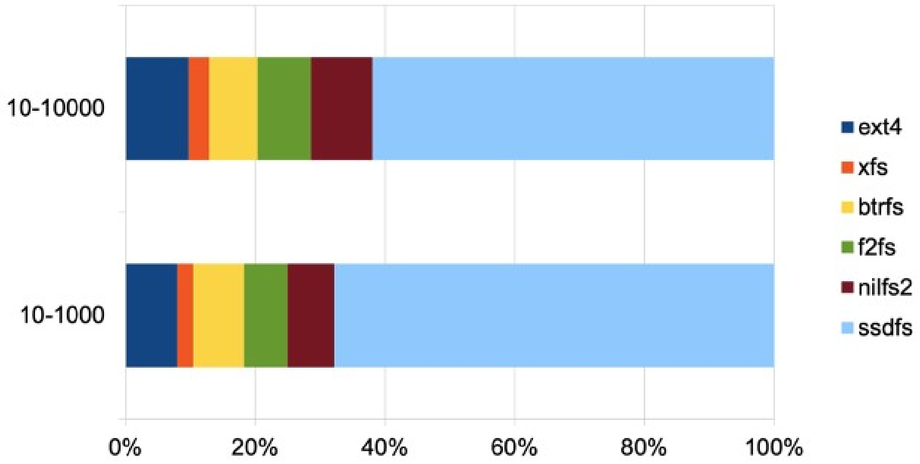
Can file system prolong SSD lifetime?

Prolong SSD lifetime
Decrease number of write/read I/O requests
Eliminate GC activity
Store more data on the same device (increase compression ratio)

Decrease TCO cost
Decrease power consumption

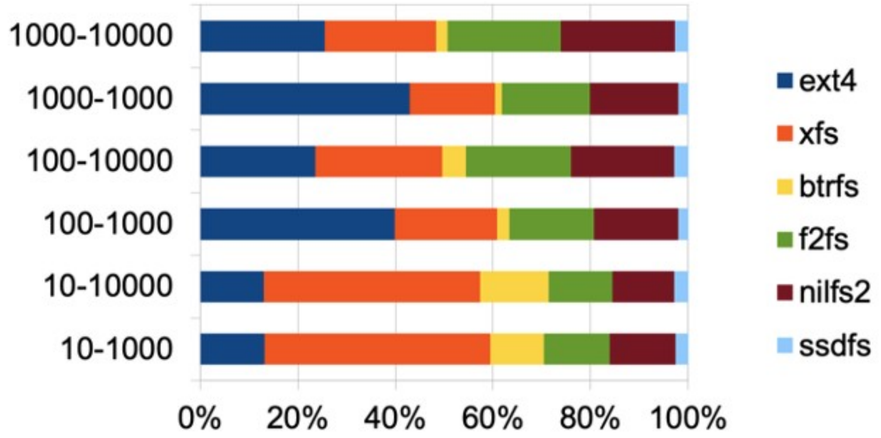
Decrease carbon footprint
Support "green" economy
Save the planet

Lifetime estimation

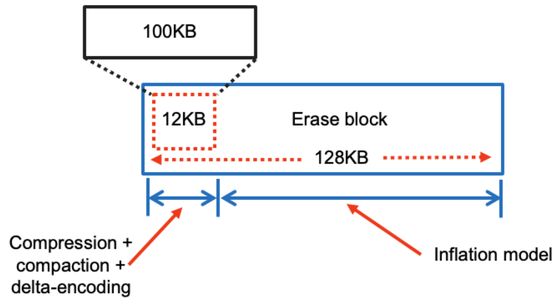


SSDFS can prolong SSD lifetime
2x - 10x for real-life use-cases

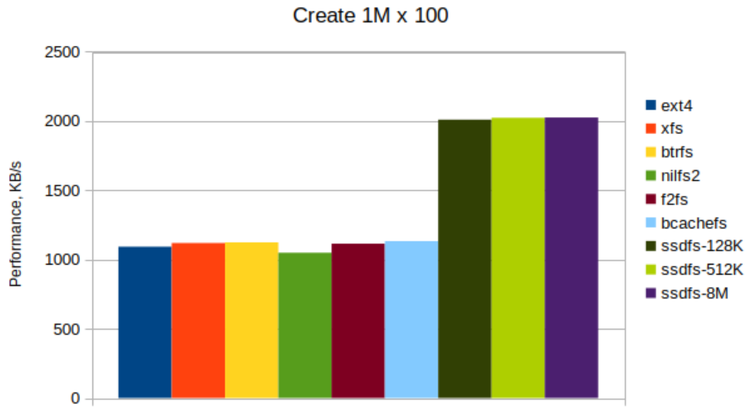
Write I/O requests

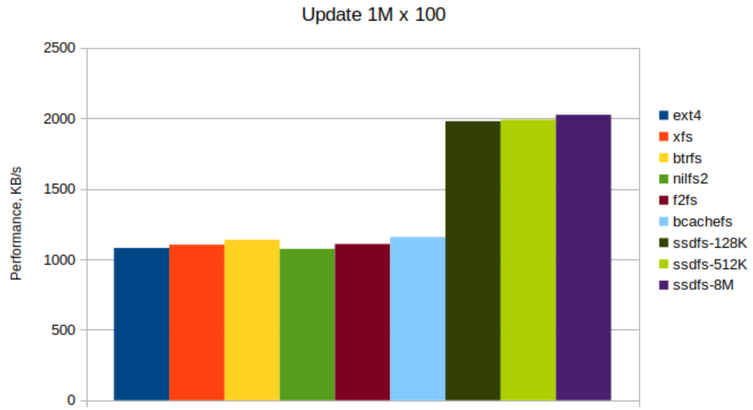


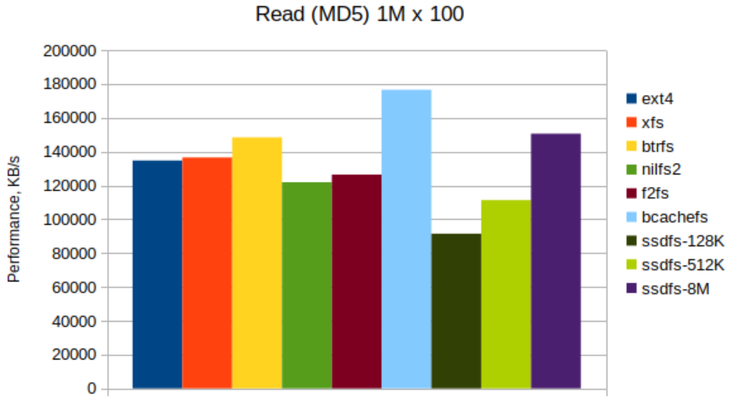
SSDFS is capable to generate **smaller amount** (1.5x - 20x) **of write I/O** requests comparing with other file systems.



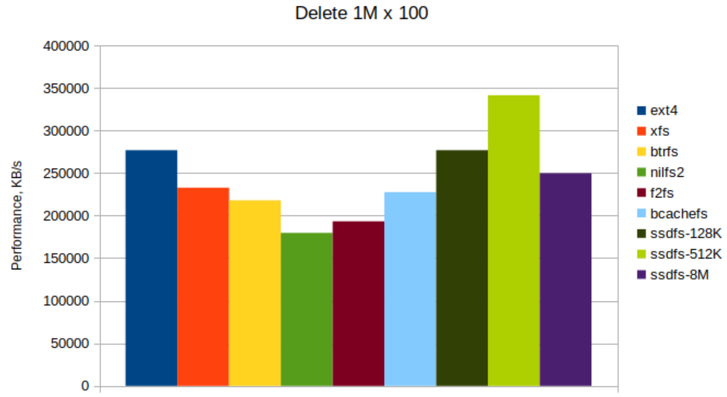
Can file system improve performance?







Segment type + **Compaction scheme** + **Migration stimulation**



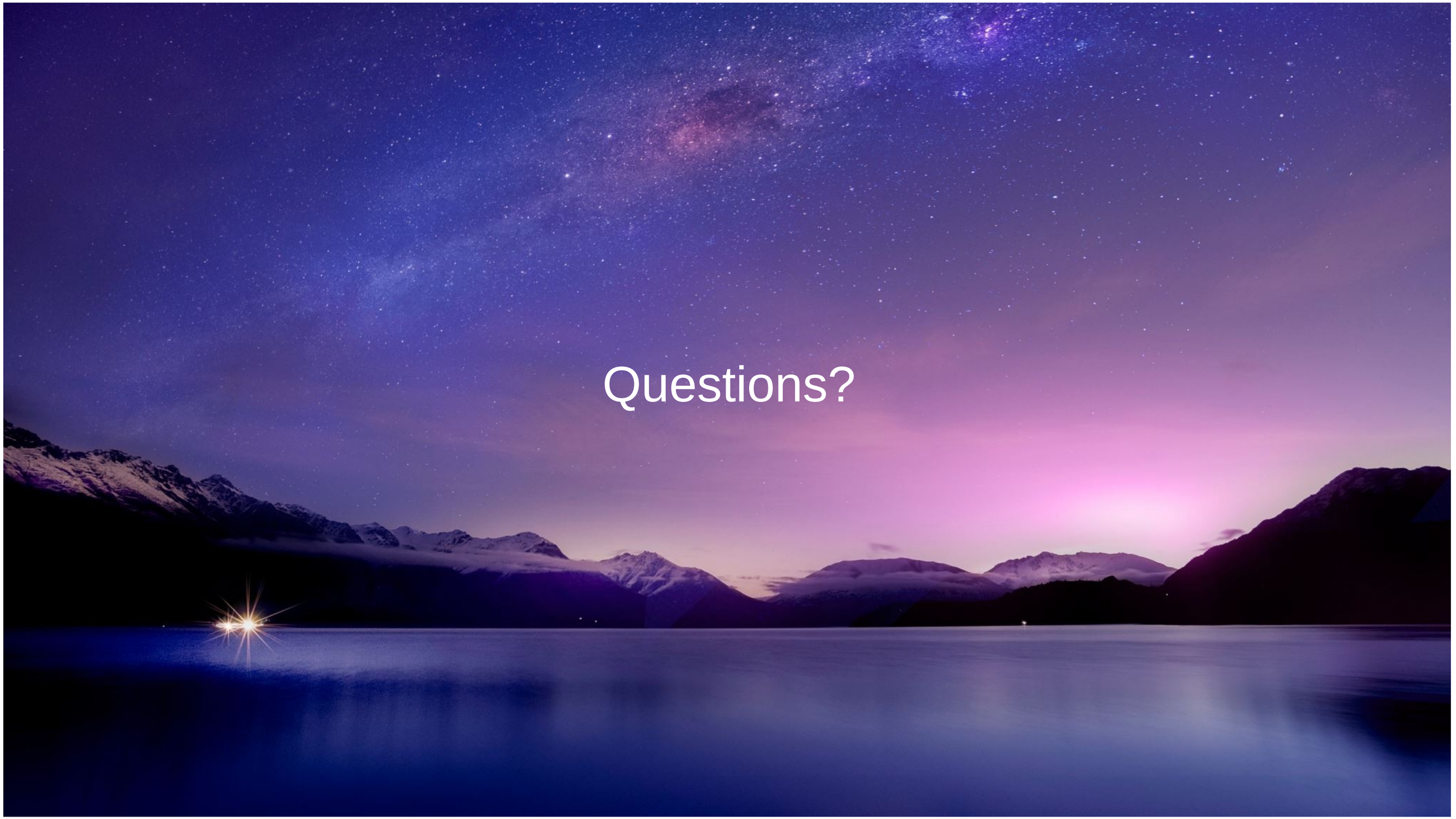
SSDFS is capable to demonstrate a **better performance** for data with good compression ratio:

create
- ext4: **1.2x** - **1.8x**
- xfs: **1.2x** - **1.8x**
- btrfs: **1.3x** - **1.8x**
- nilfs2: **1.1x** - **1.9x**
- f2fs: **1.2x** - **1.8x**
- bcachefs: **0.9x** - **1.7x**

update
- ext4: **1.5x** - **1.8x**
- xfs: **1.5x** - **1.8x**
- btrfs: **1.4x** - **1.7x**
- nilfs2: **1.5x** - **1.8x**
- f2fs: **1.5x** - **1.8x**
- bcachefs: **1.5x** - **1.7x**

read
- ext4: **1.1x** - **3.7x**
- xfs: **1.1x** - **3.7x**
- btrfs: **1x** - **3.4x**
- nilfs2: **1.2x** - **4x**
- f2fs: **1.2x** - **3.9x**
- bcachefs: **0.8x** - **3x**

delete
- ext4: **1.2x** - **2.2x**
- xfs: **1.4x** - **2.2x**
- btrfs: **1.5x** - **2.2x**
- nilfs2: **1.5x** - **2.8x**
- f2fs: **1.7x** - **2.6x**
- bcachefs: **1.5x** - **2.2x**


Questions?