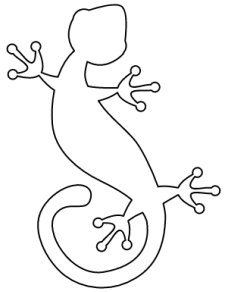


GeckoFTL

Scalable Flash Translation Techniques for Very Large Flash Devices

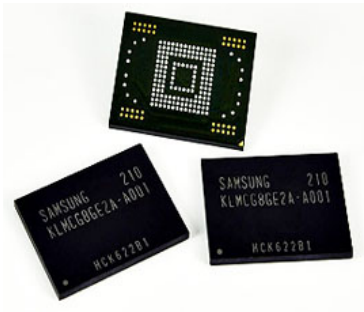
Niv Dayan, Philippe Bonnet, Stratos Idreos

HARVARD
School of Engineering and Applied Sciences

DASlab

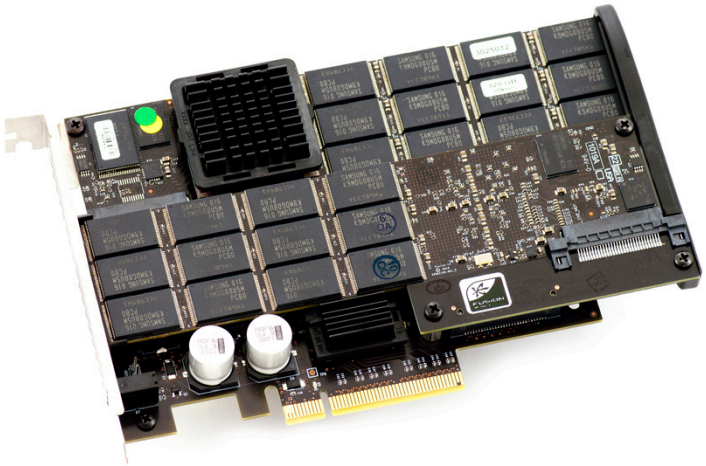
SD Card

eMMC



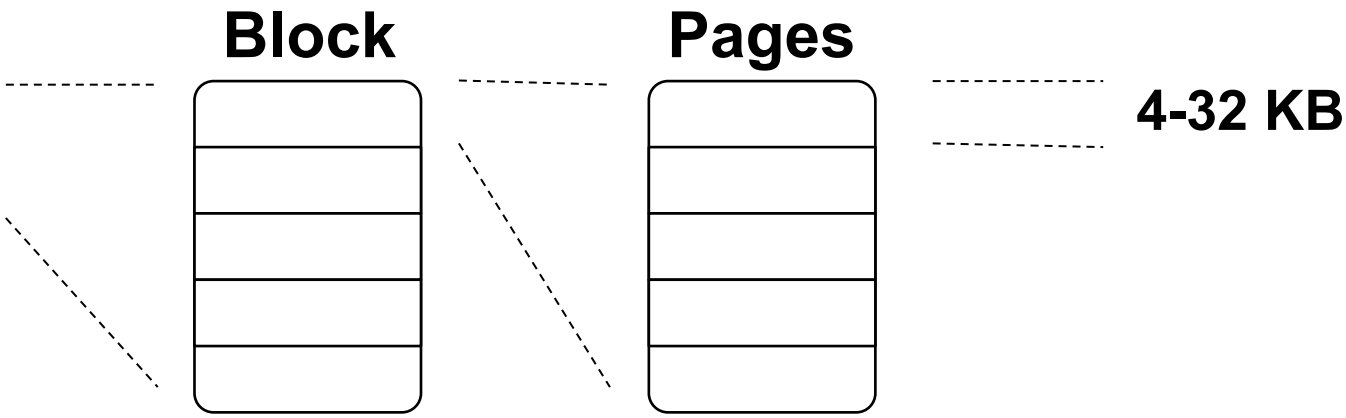
SSD

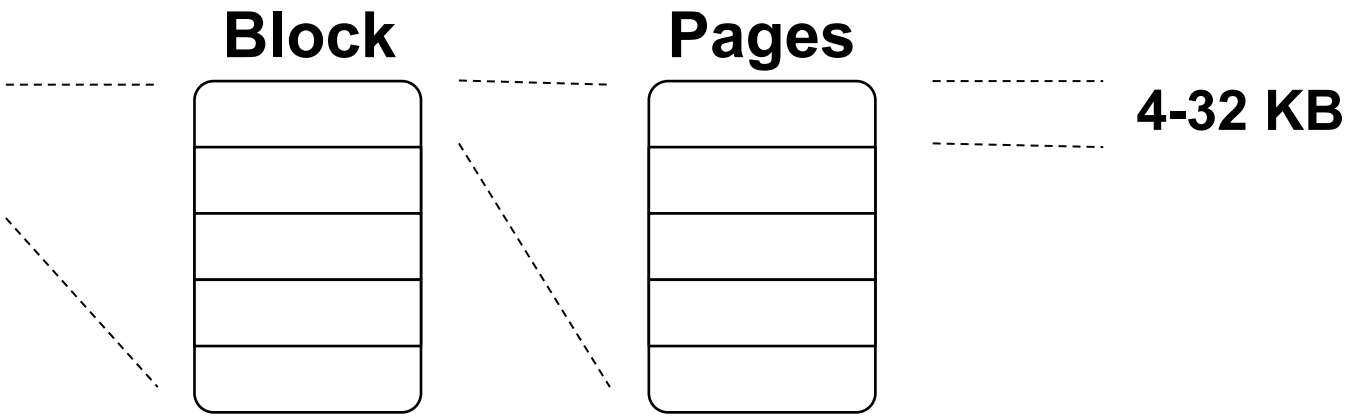
Enterprise SSD



↑ performance

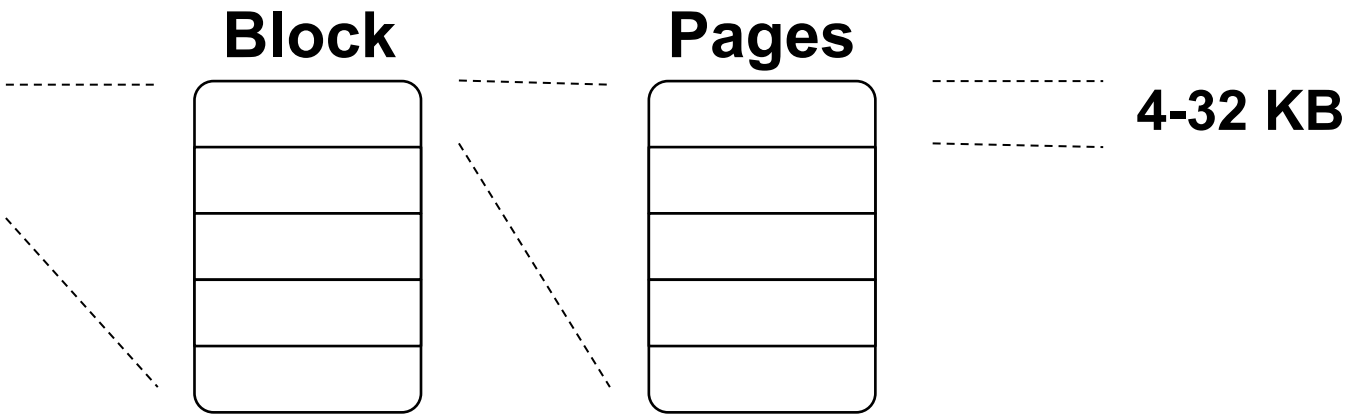
↓ power consumption


Block
Pages
4-32 KB



Constraints

- reads/writes at page granularity
- write latency > read latency
- sequential writes within a block
- block-erase before update
- limited erases per block



Constraints

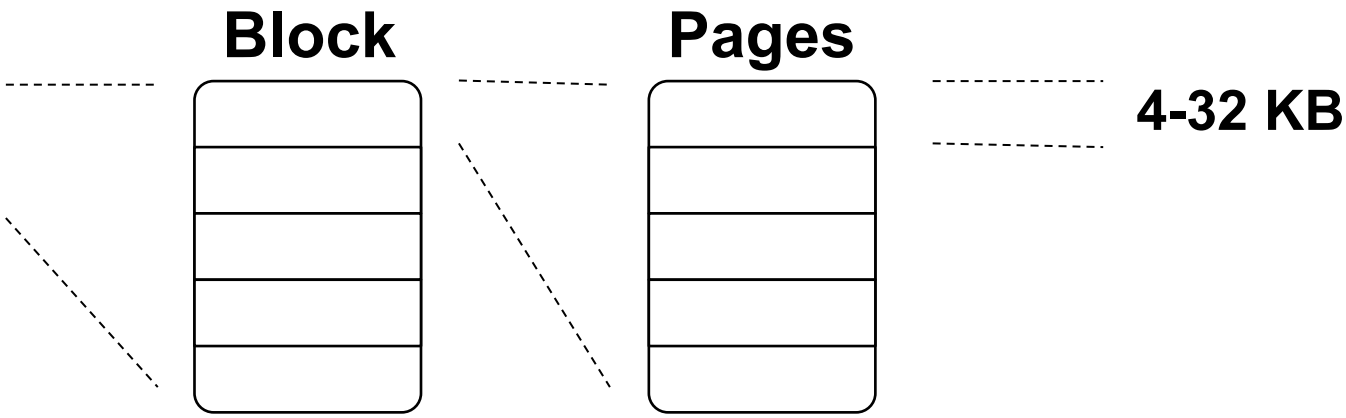
reads/writes at page granularity

write latency > read latency

sequential writes within a block

block-erase before update

limited erases per block



Constraints

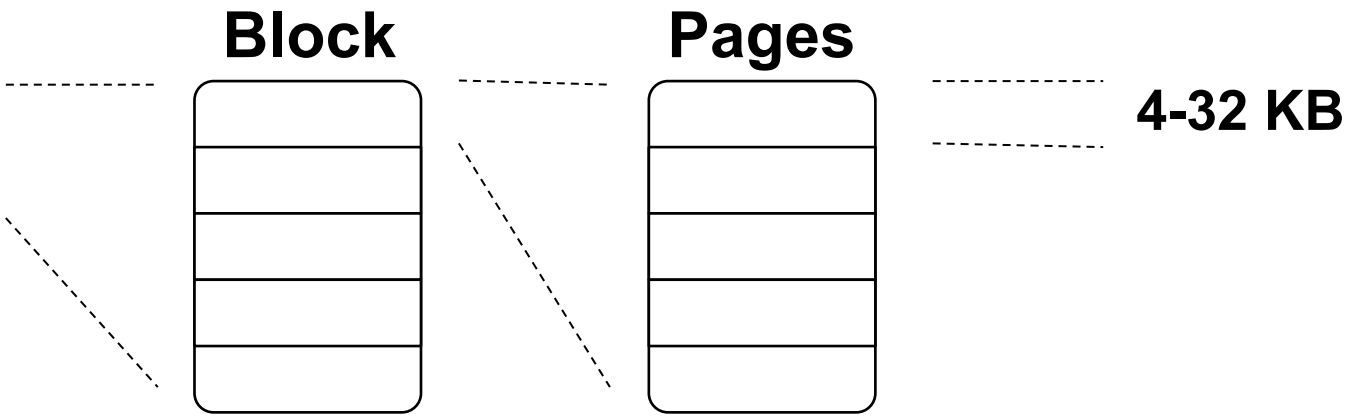
reads/writes at page granularity

write latency > read latency

sequential writes within a block

block-erase before update

limited erases per block



Constraints

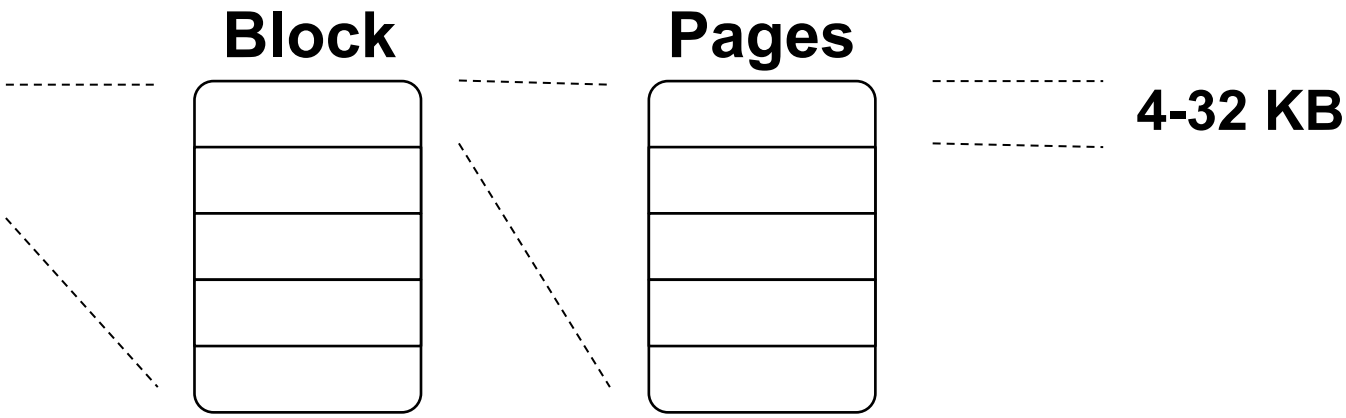
reads/writes at page granularity

write latency > read latency

sequential writes within a block

block-erase before update

limited erases per block



Constraints

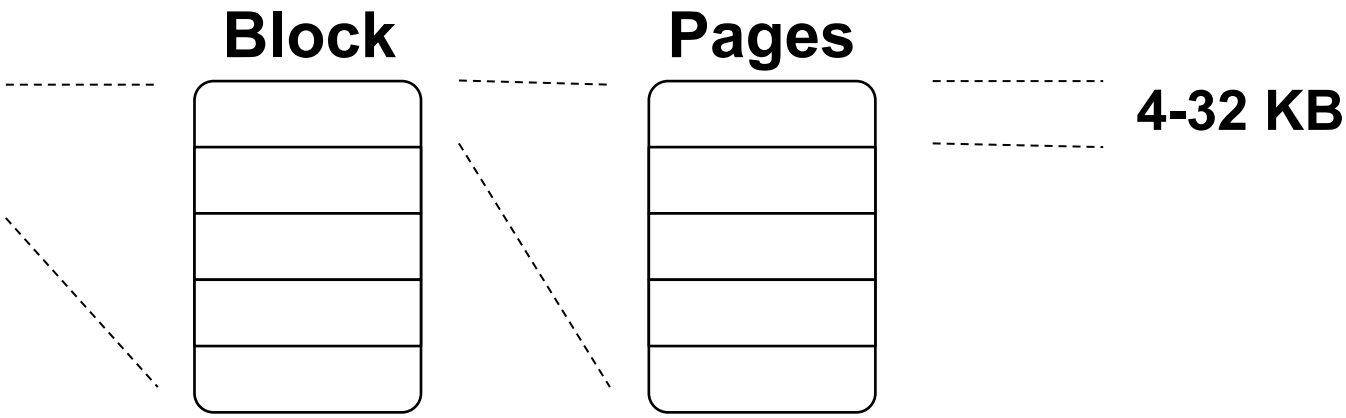
reads/writes at page granularity

write latency > read latency

sequential writes within a block

block-erase before update

limited erases per block



Constraints

reads/writes at page granularity

write latency > read latency

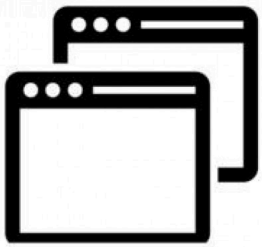
sequential writes within a block

block-erase before update

limited erases per block

Flash Translation Layer (FTL)

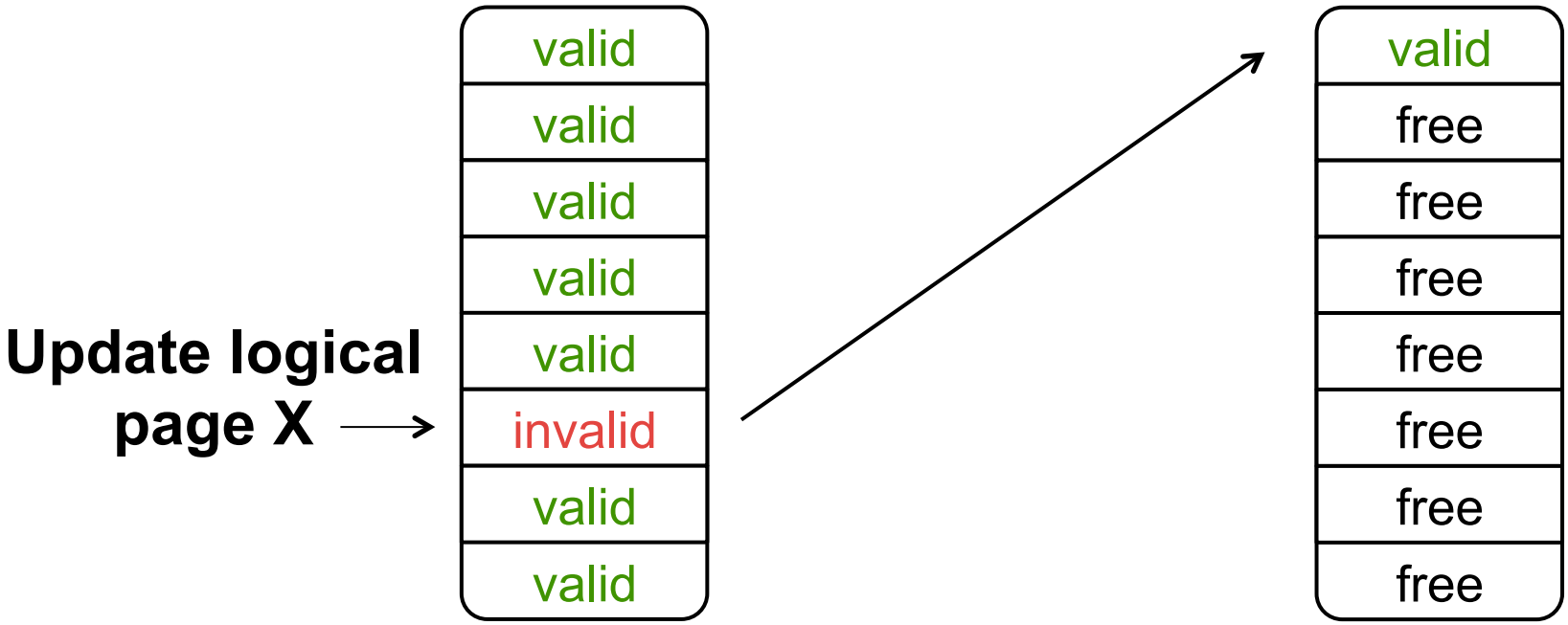
Application



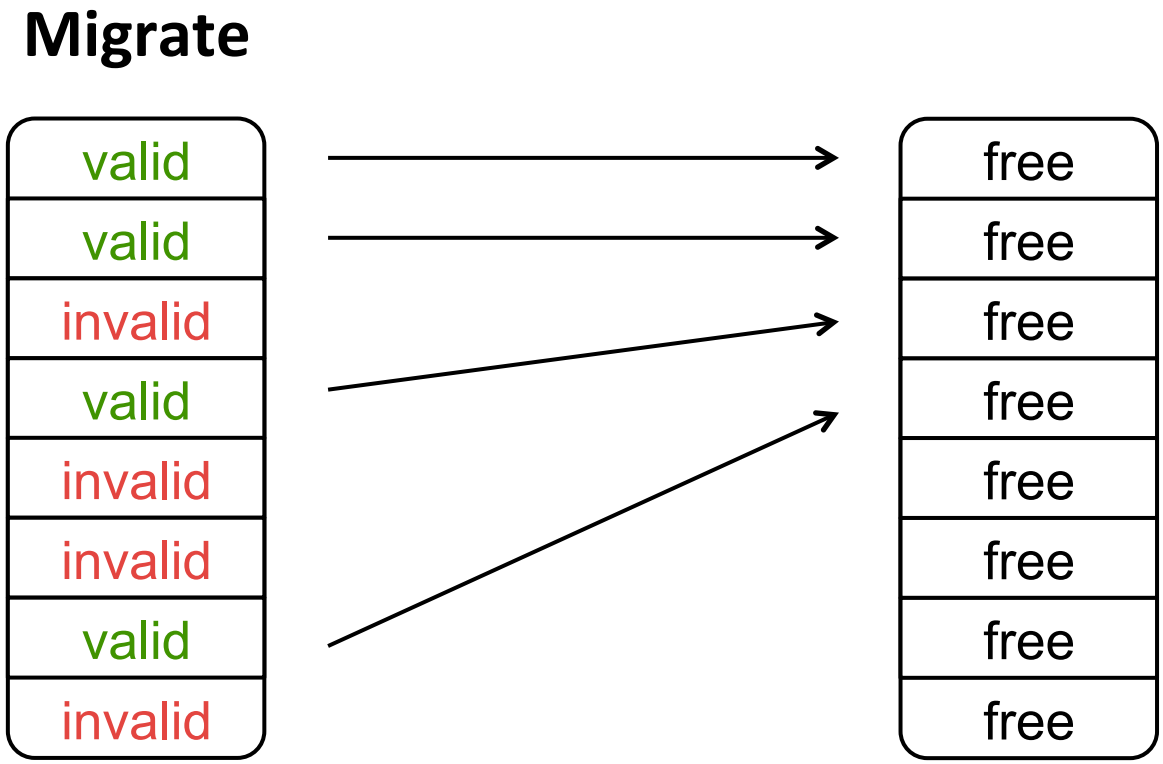
Flash


FTL

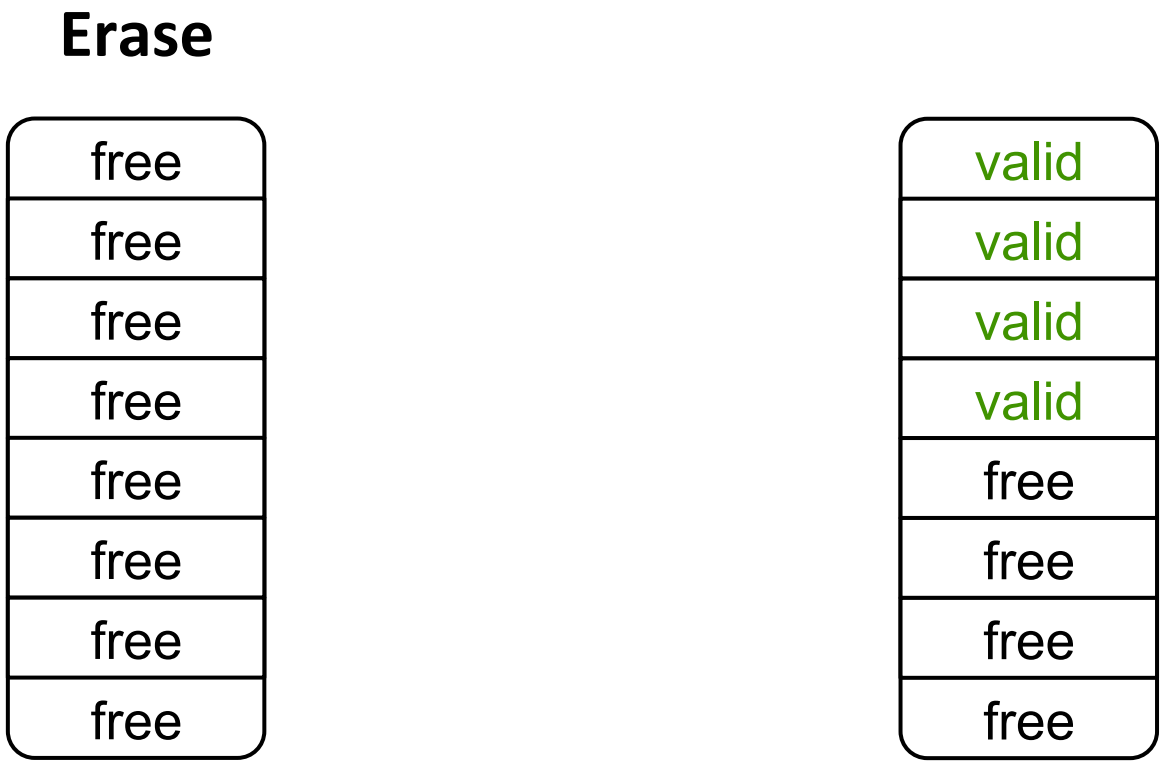
Out-of-place updates



Garbage-collection



Garbage-collection



Page Translation

Integrated RAM

Flash

Mapping Cache

Mapping table

User data

Page Translation

Integrated RAM

Mapping Cache

Page Validity Bitmap (PVB)

... 0 0 0 0 0 1 0 0 ...

Flash

Mapping table

User data

valid valid valid valid valid invalid valid valid

... ...

Two concerns

1. Cost $
2. Recovery ⏳

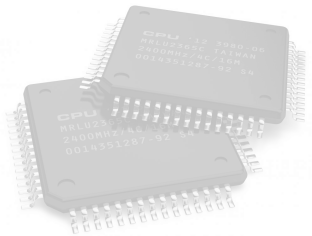
Integrated RAM

Mapping Cache

Page Validity Bitmap (PVB)

… 0 0 0 0 0 1 0 0 …

Flash



Mapping table

User data

valid valid valid valid valid invalid valid valid

… …

Two concerns

1. Cost $
2. Recovery ⧗

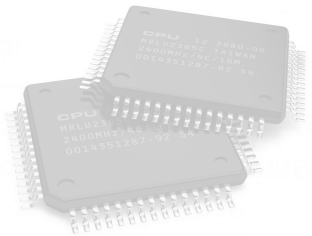
Integrated RAM

Mapping Cache

Page Validity Bitmap (PVB)

… 0 0 0 0 0 1 0 0 …

Flash



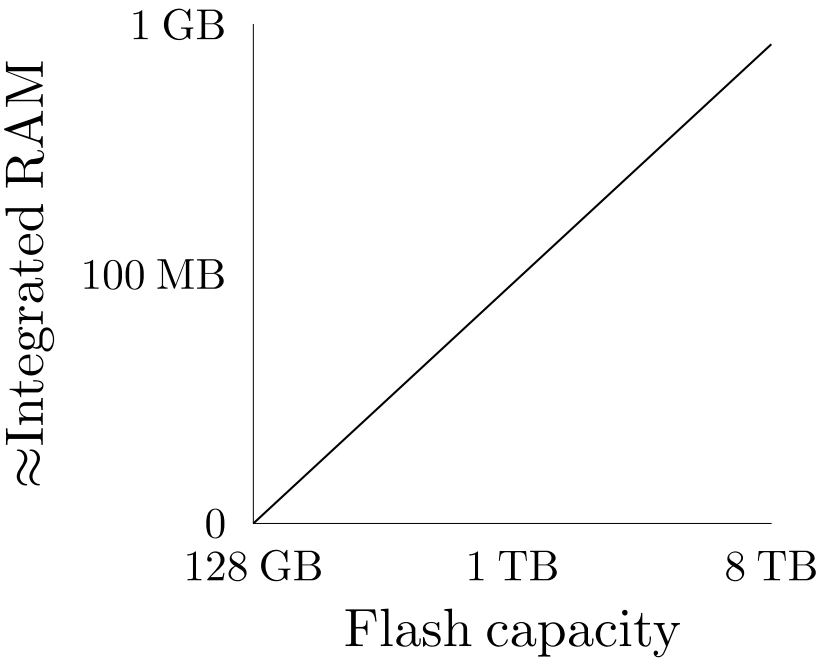
Mapping table

User dat

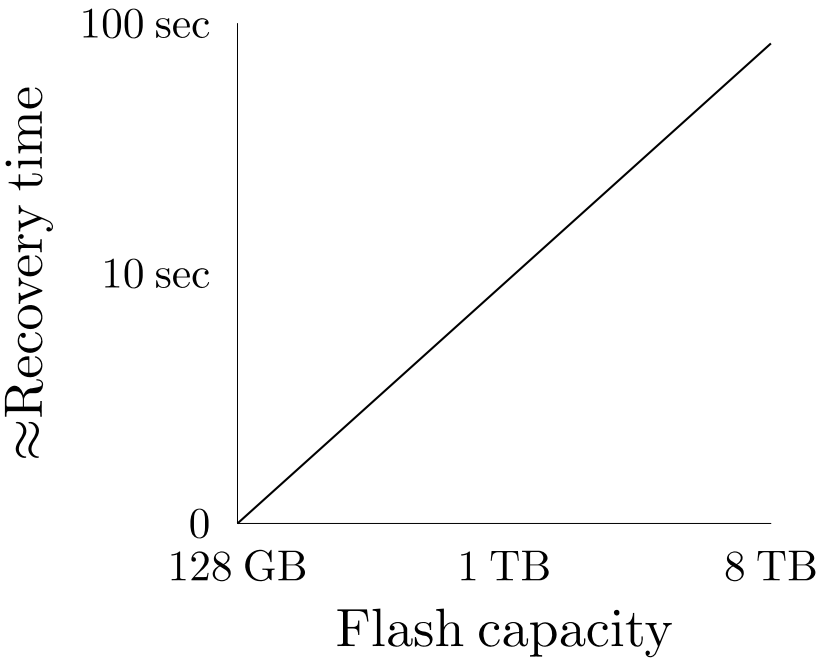
… …

PVB ∝ device capacity

↑ RAM

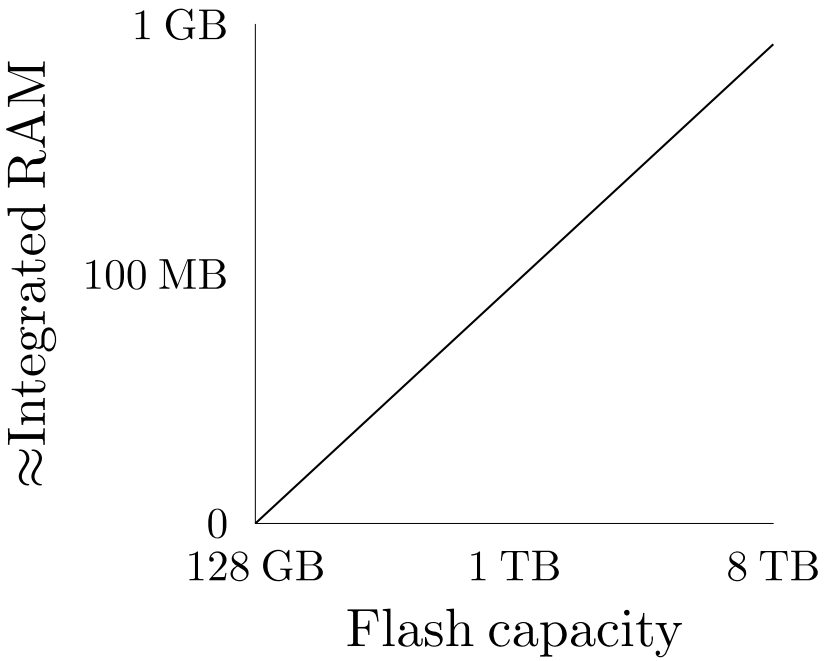


↑ Recovery

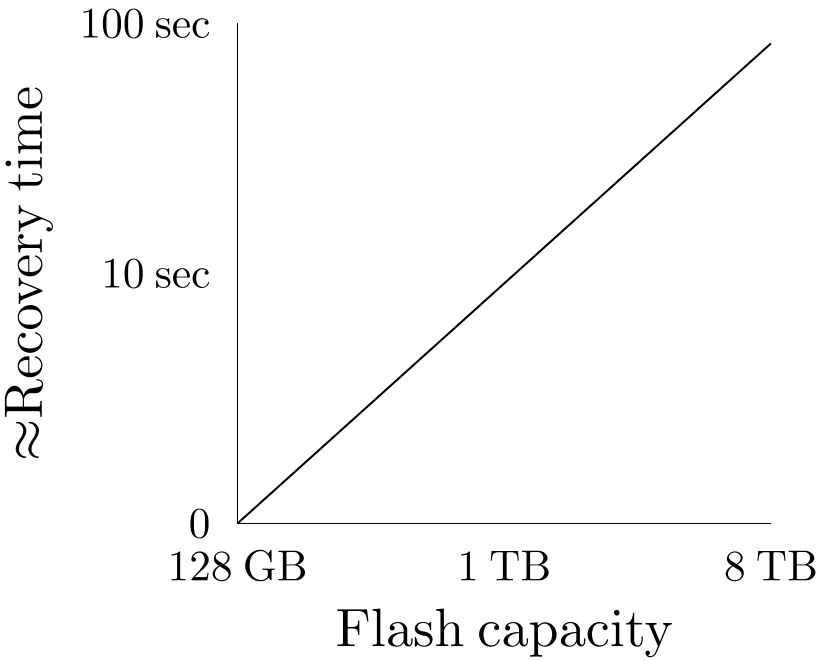


PVB ∝ device capacity

↑ RAM



↑ Recovery



PVB is the bottleneck.

Simple solution: store PVB in flash

- ↓ RAM
- ↓ Performance
- ↓ Recovery
- ↓ Device lifetime

Integrated RAM

Flash

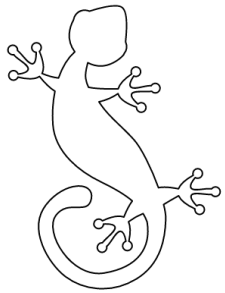
PVB

... 0 0 0 0 0 **1** 0 0 ...

Simple solution: store PVB in flash

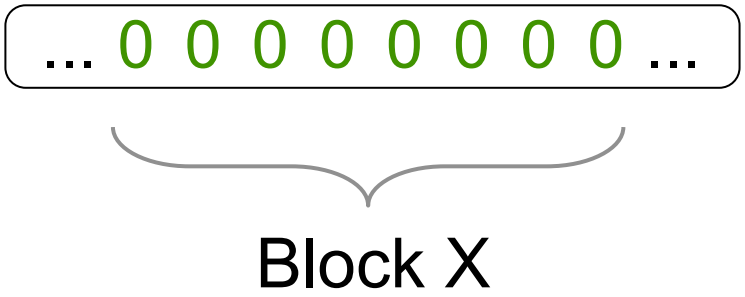
- ↓ RAM
- ↓ Performance
- ↓ Recovery
- ↓ Device lifetime

*How to store PVB in flash without harming **performance** or **device lifetime**?*



Solution:
GeckoFTL

Insight



Insight

Page Invalidation ⟹

PVB

... 0 0 0 0 0 0 0 0 ...

Block X

Insight

Page Invalidation ⟹

PVB

... 0 0 1 0 0 0 0 0 ...

Block X

Insight

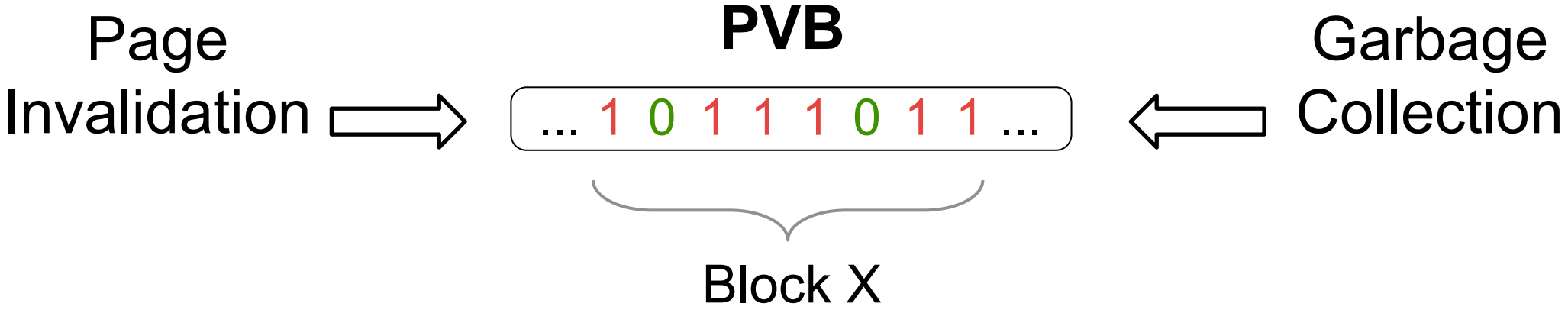
Page Invalidation ⟹

PVB

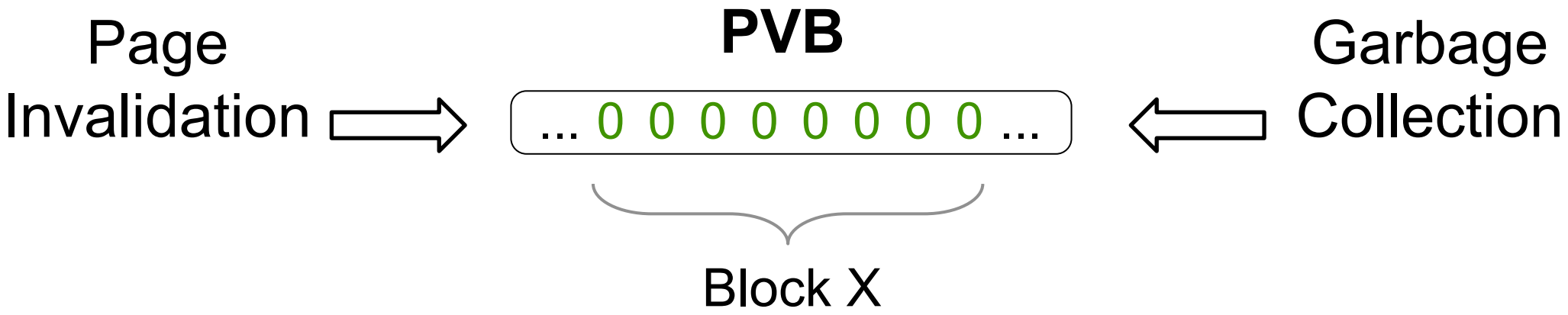
... 1 0 1 1 1 0 1 1 ...

Block X

Insight


Page Invalidation
PVB
... 1 0 1 1 1 0 1 1 ...
Garbage Collection
Block X

Insight

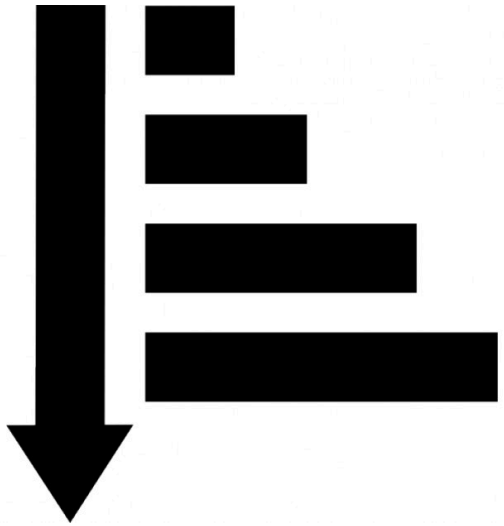


PVB is update-intensive

Optimize for Updates

Buffer modifications in RAM

Reorganize later



LSM-tree variant

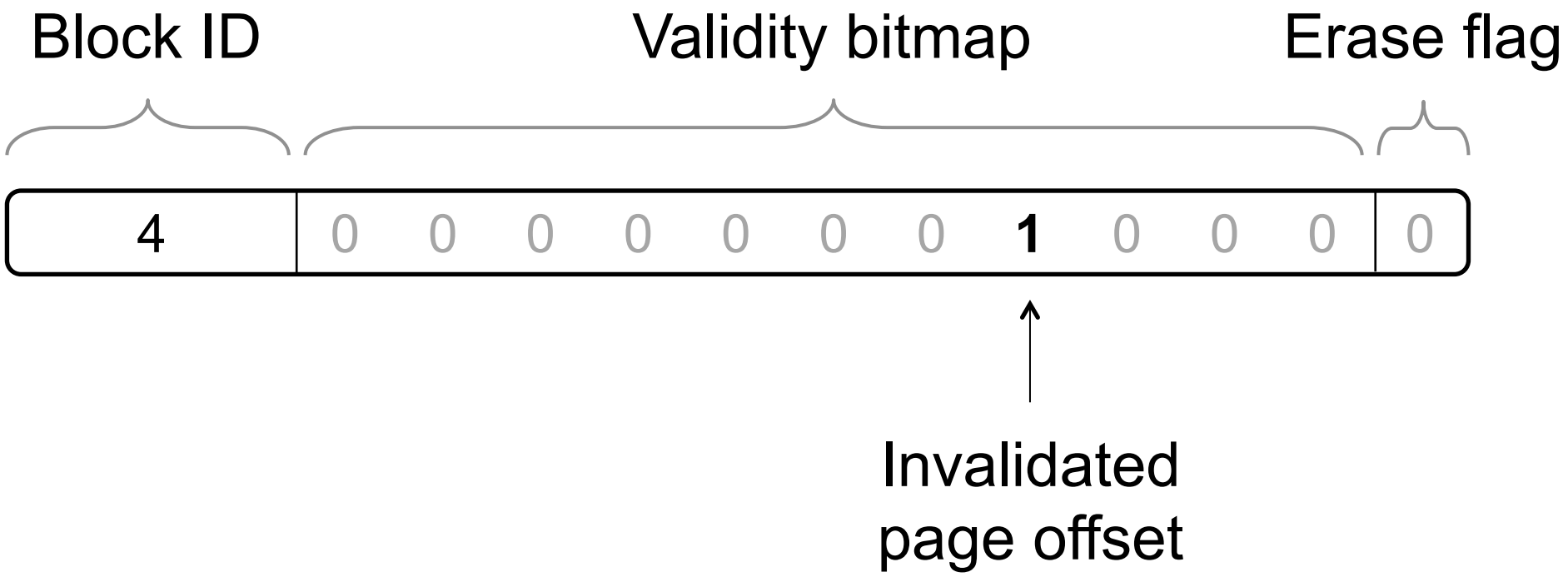
Integrated RAM

buffer

Flash

level 0	
level 1	
level 2	
…	

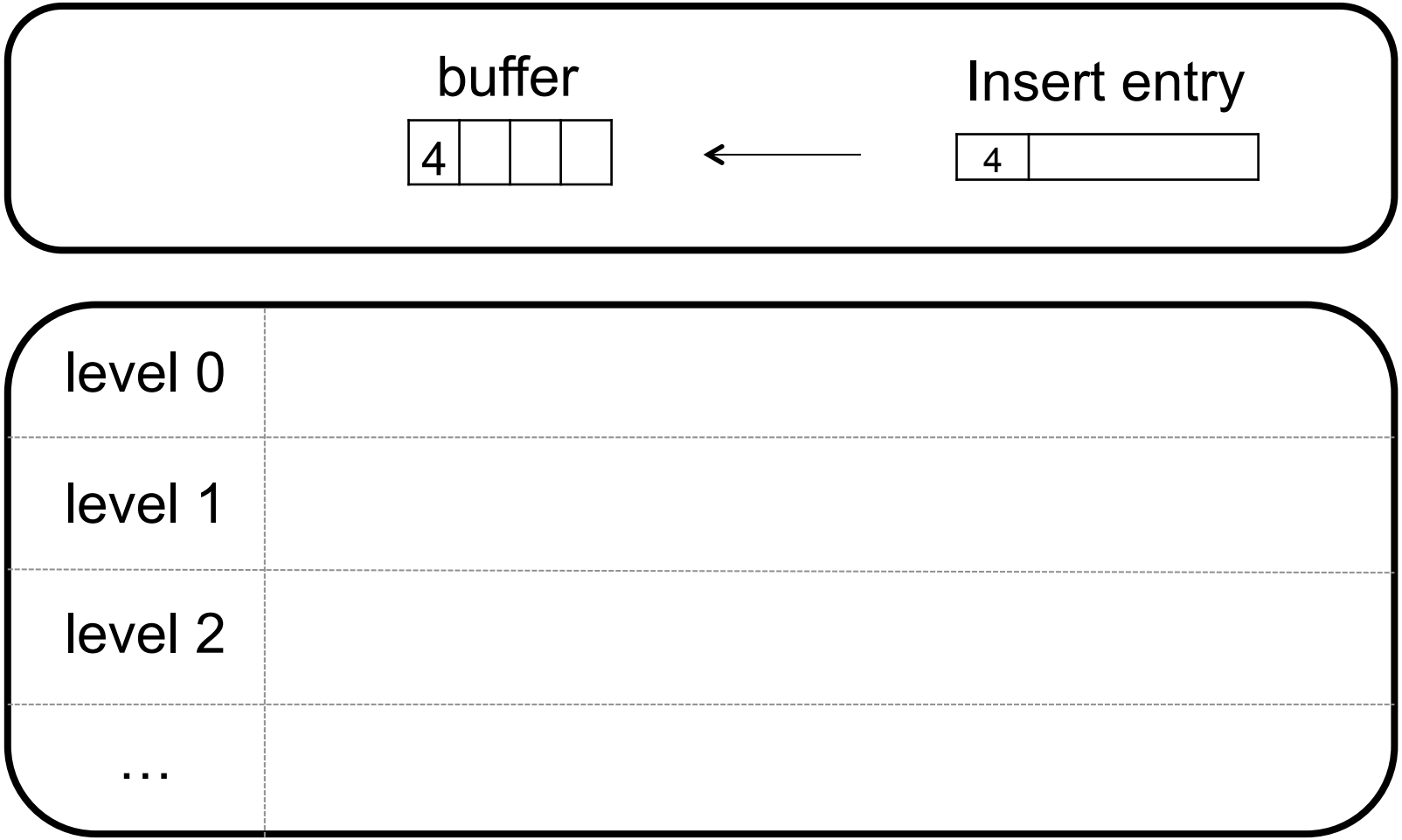
Page Invalidation



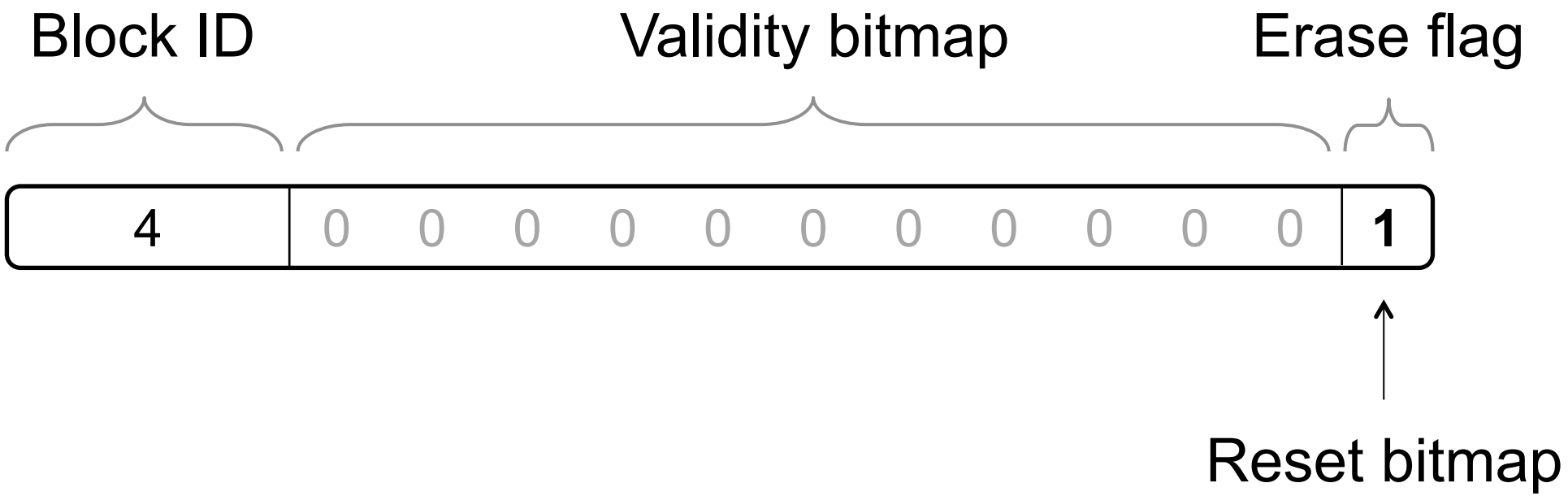
Page Invalidation

Integrated RAM

Flash



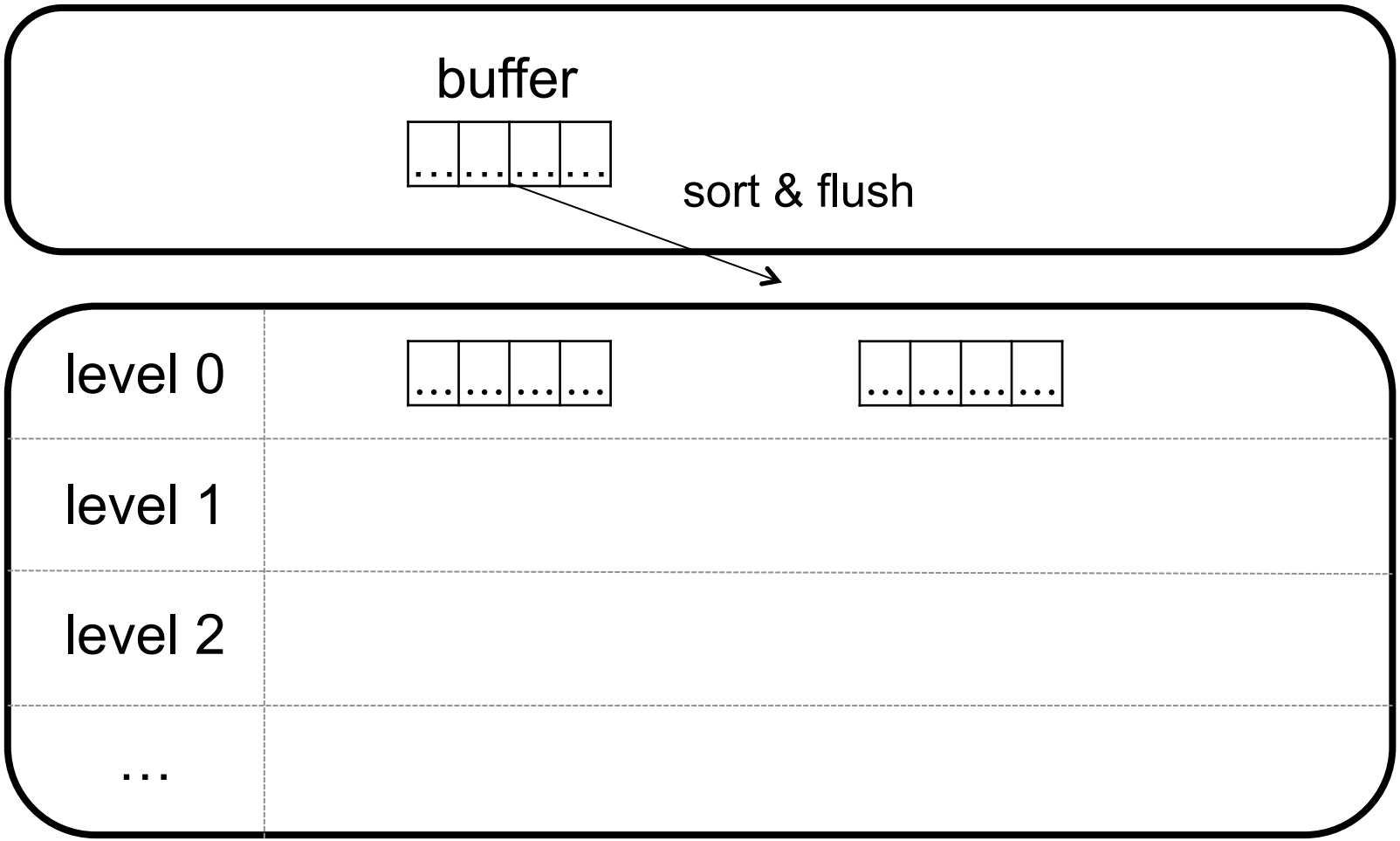
Garbage Collection



Flush & Merge

Integrated RAM

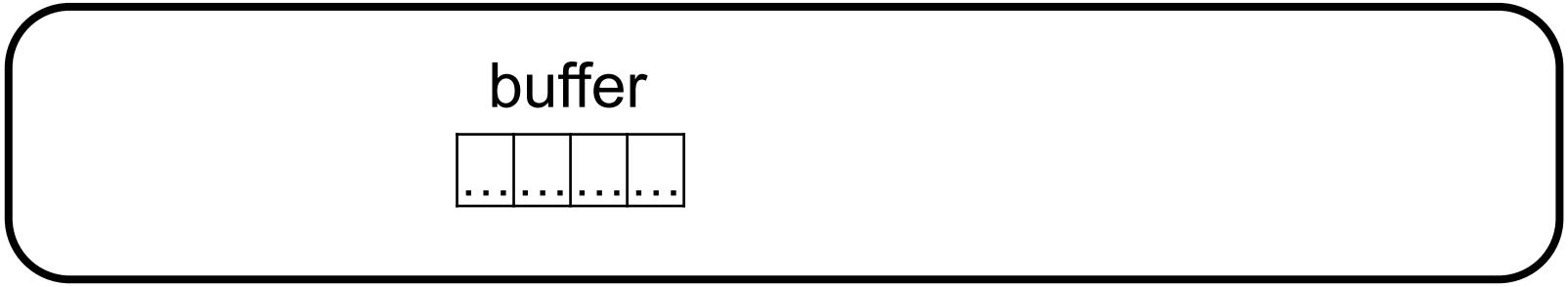
Flash

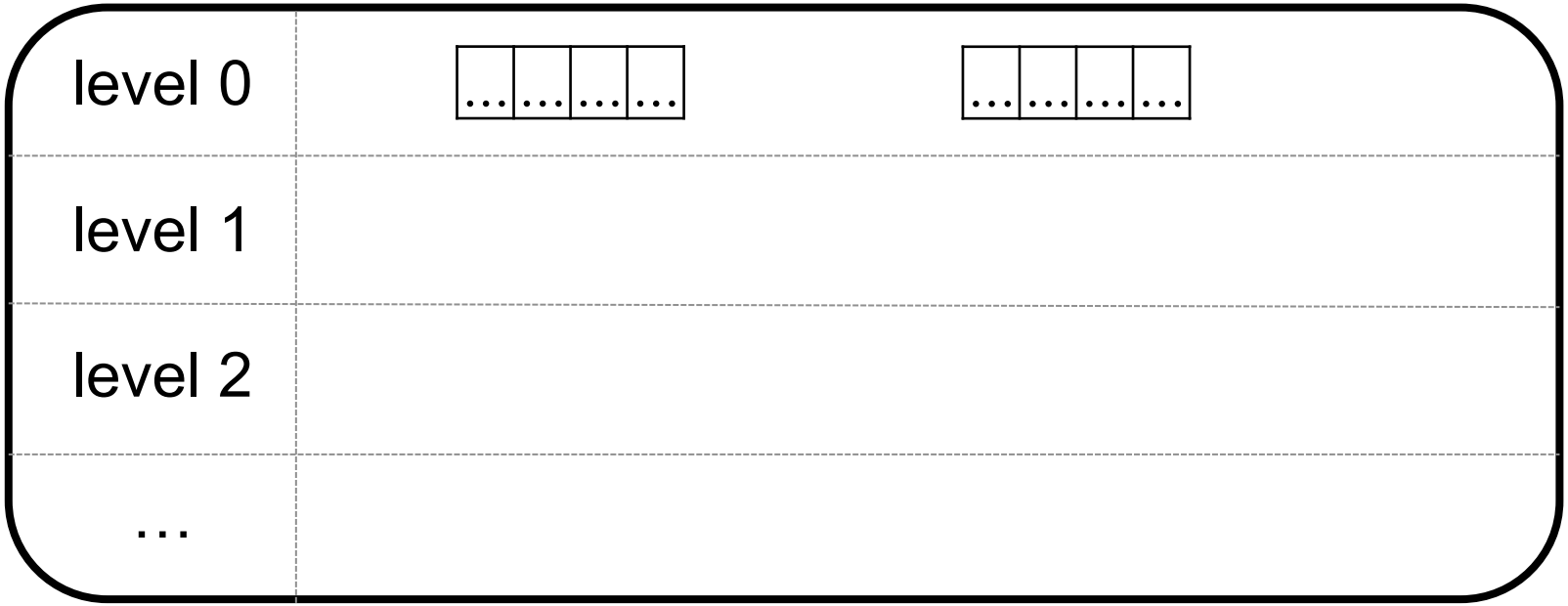


Flush & Merge

Integrated RAM

Flash

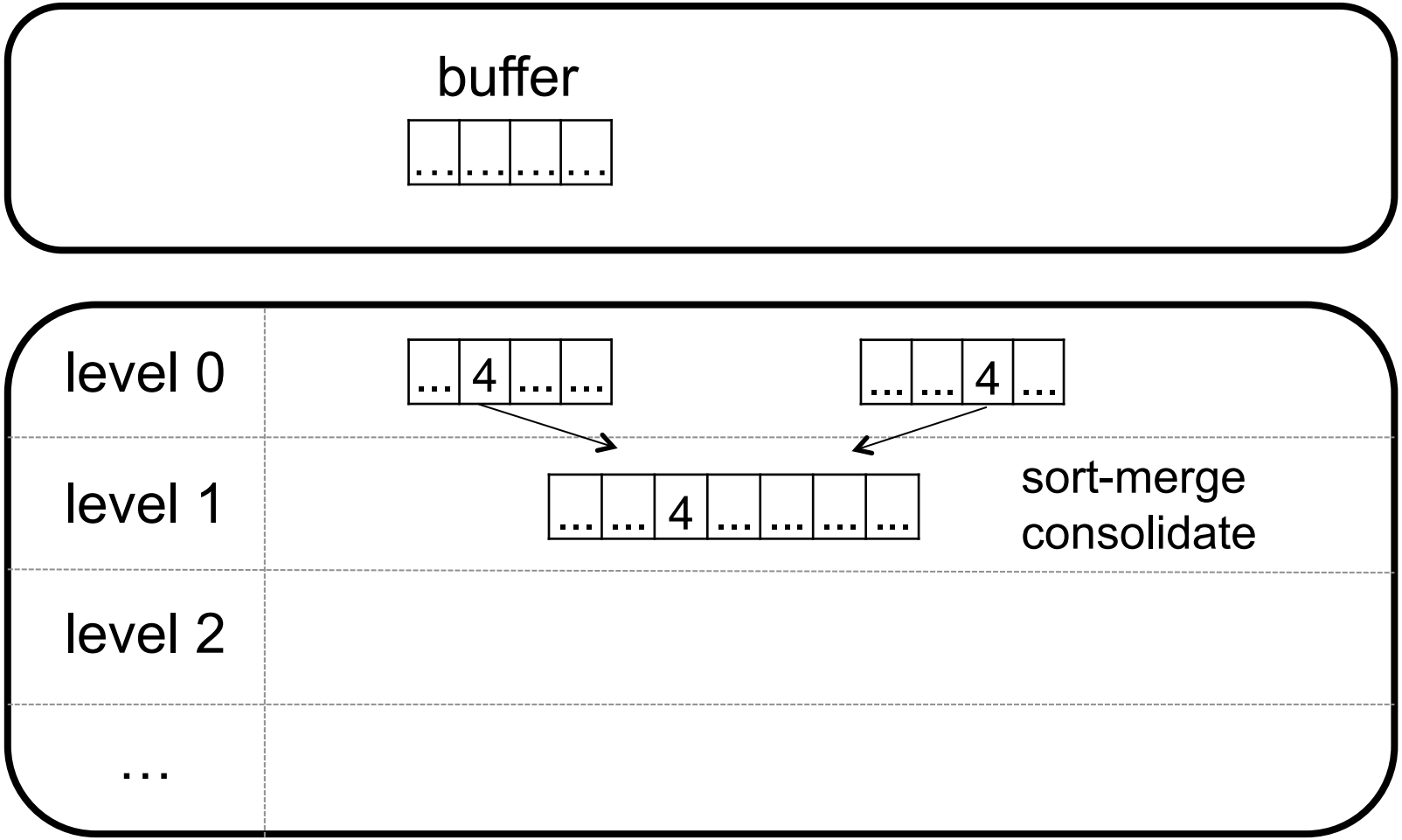




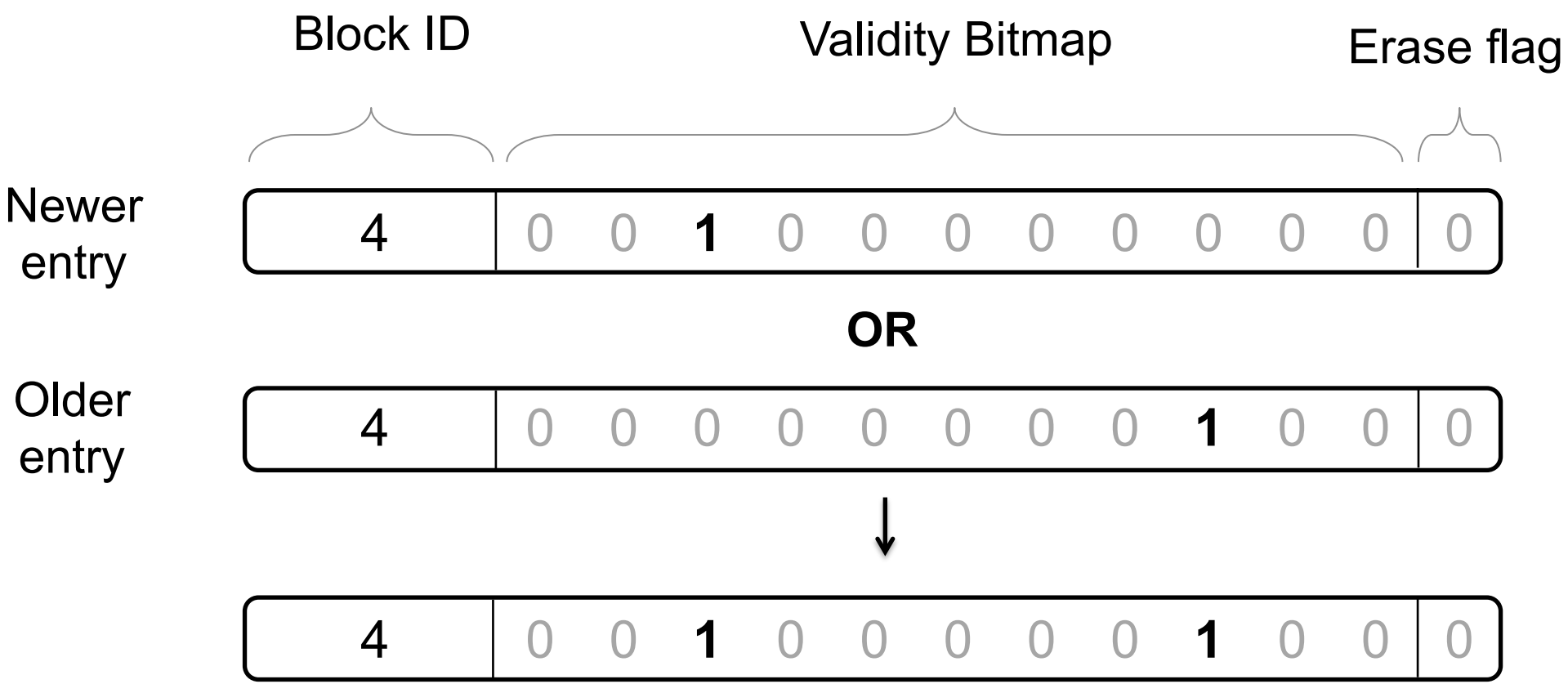
Flush & Merge

Integrated RAM

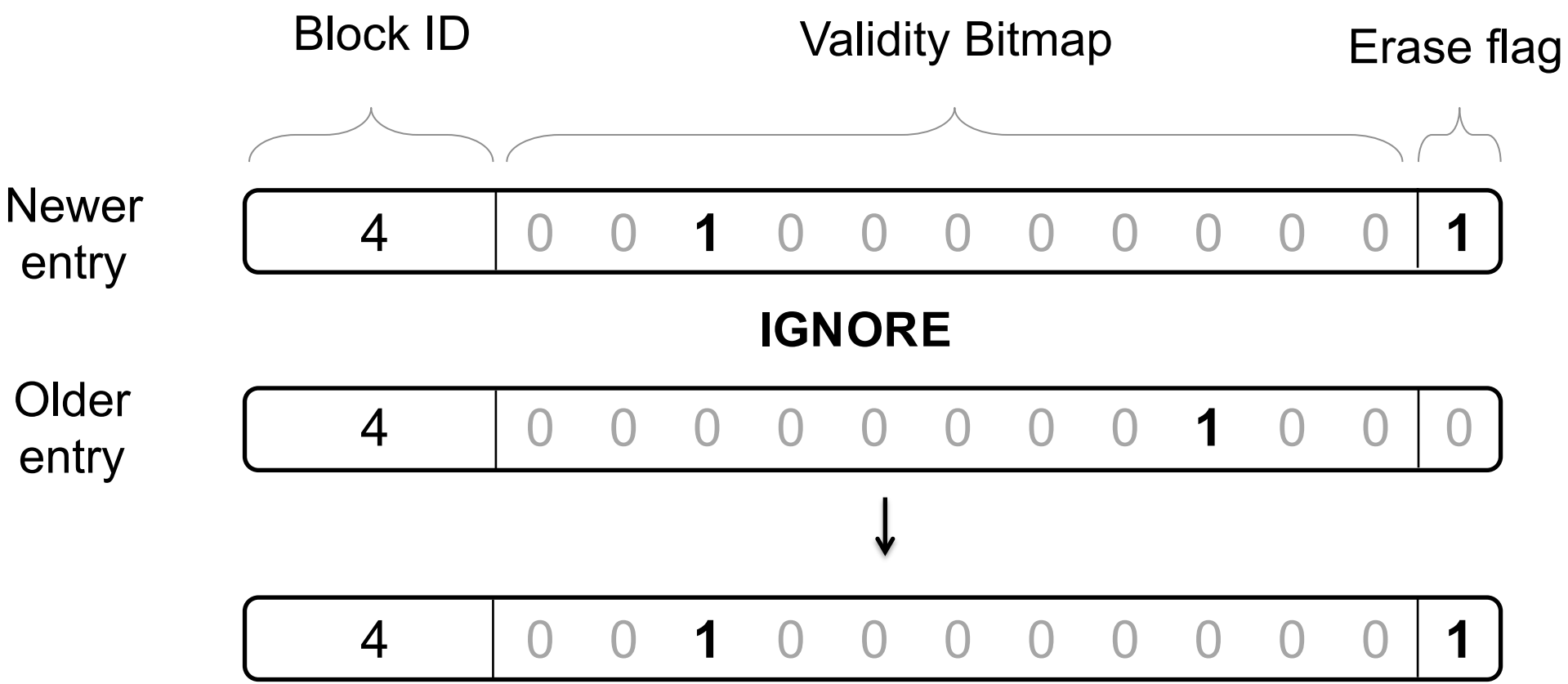
Flash



Merge bitmaps



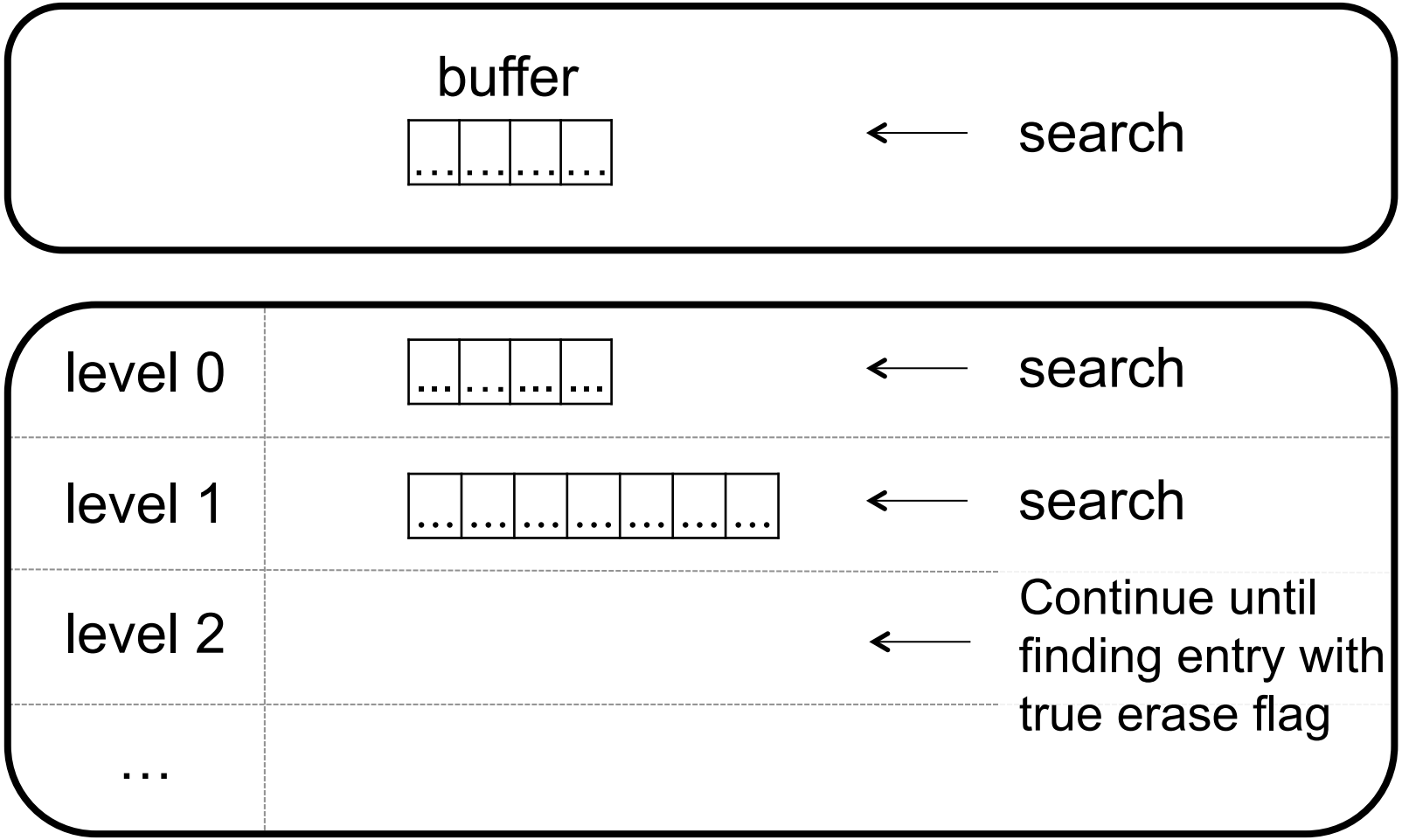
Merge bitmaps



Garbage Collection

Integrated RAM

Flash



Simulation

Baselines:

PVB in RAM

PVB in flash

Simulation

Baselines:

PVB in RAM

PVB in flash

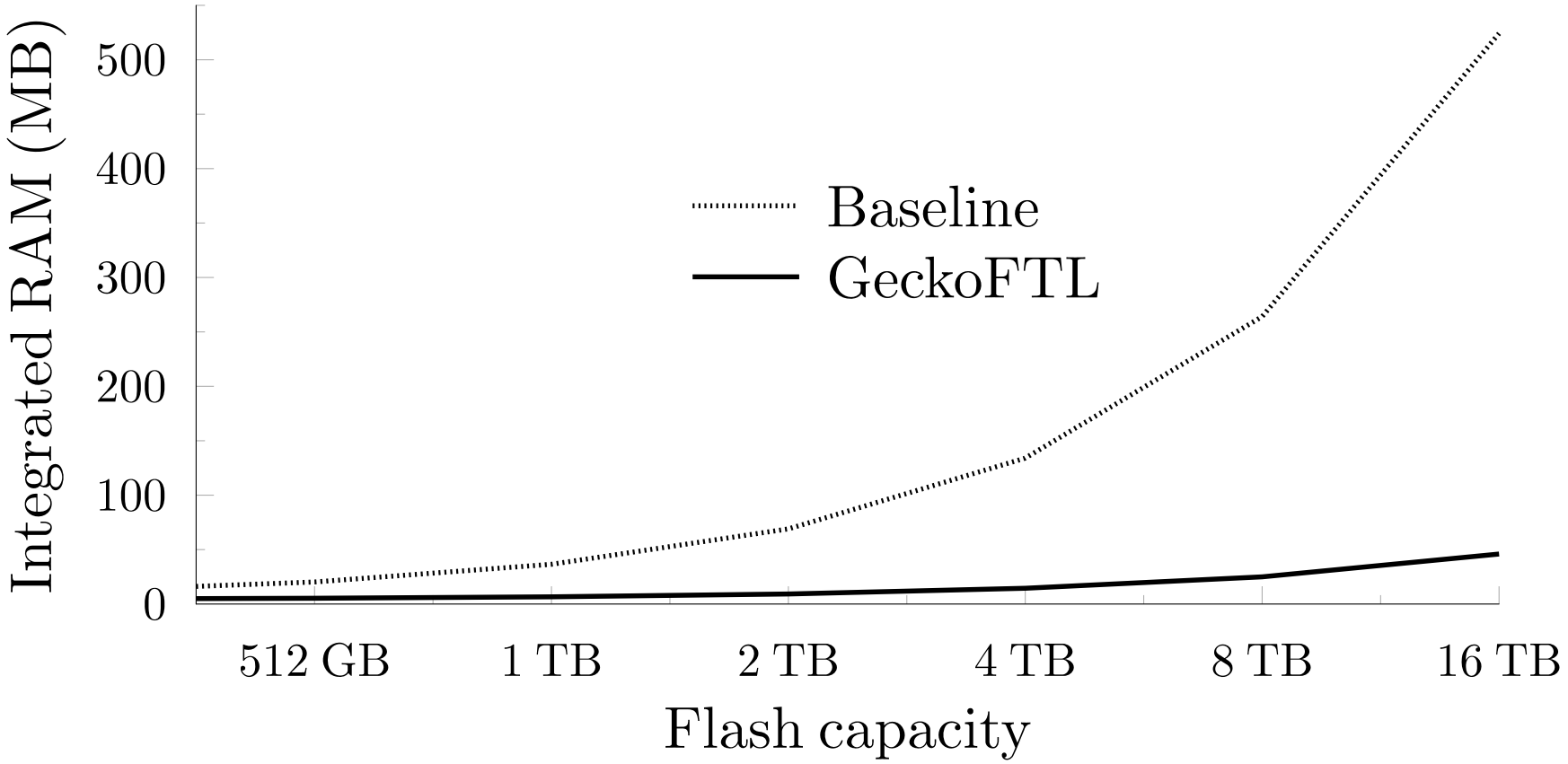
Metrics

RAM required

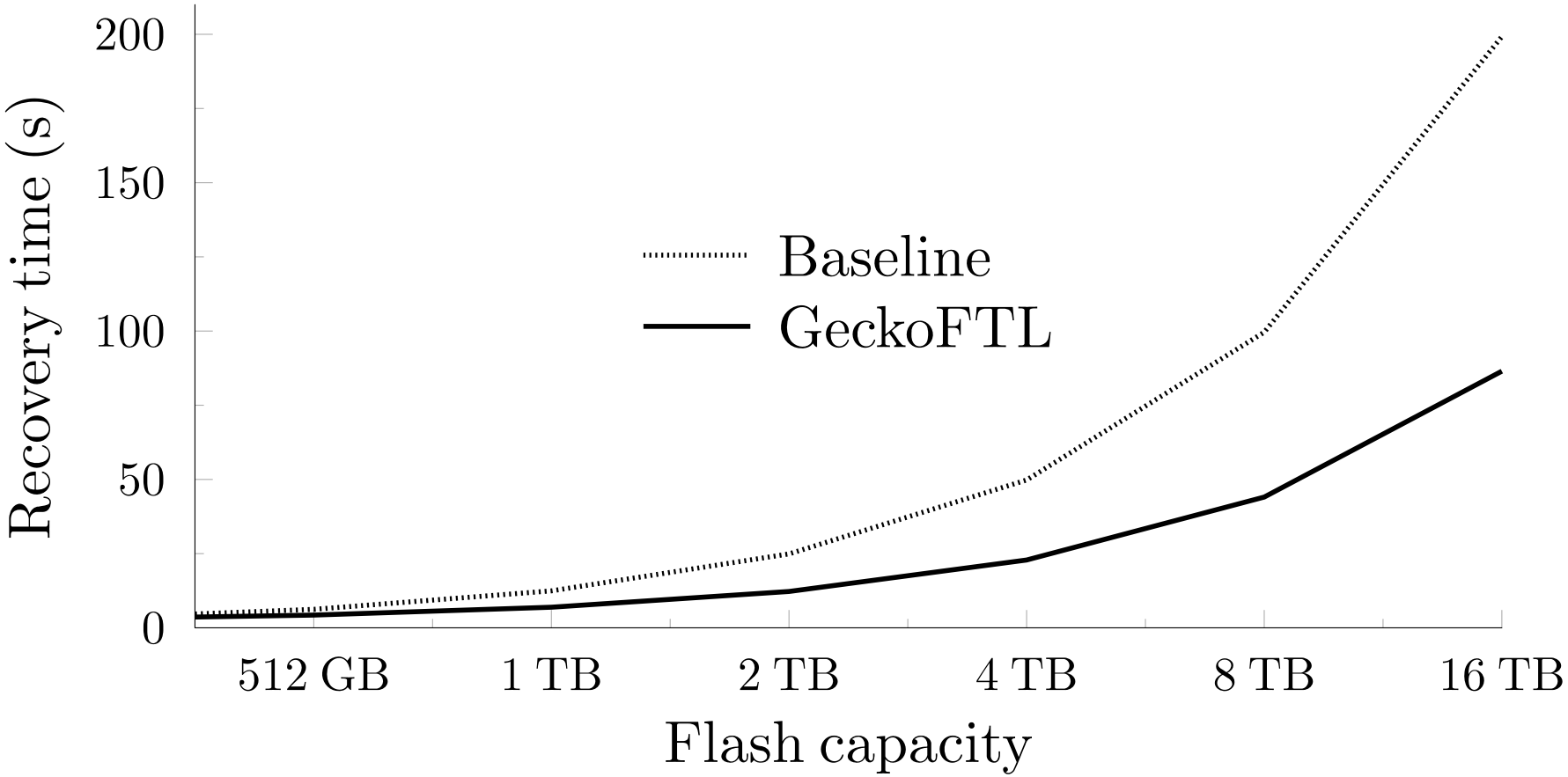
Recovery time

IO overheads

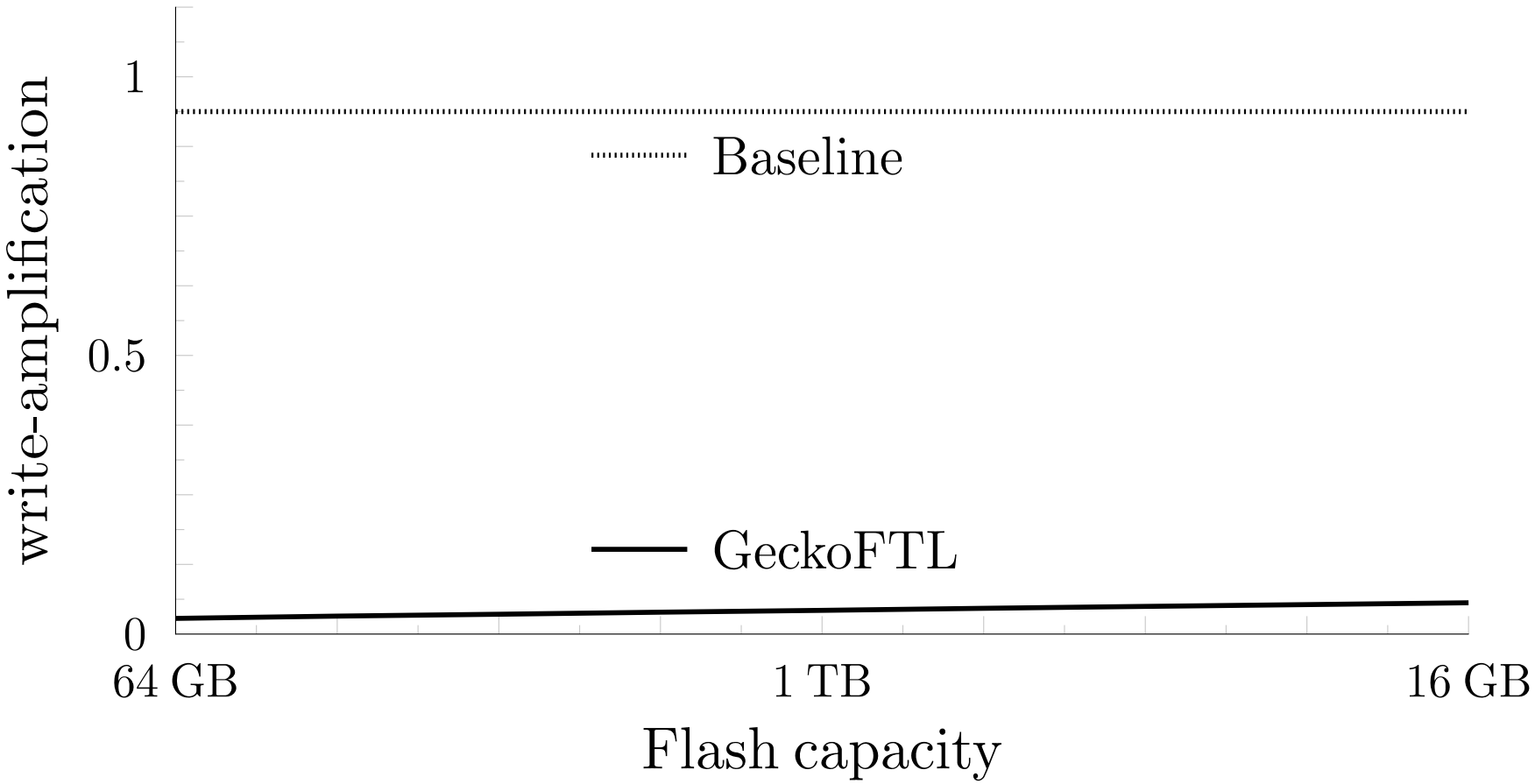
>90% RAM reduction



>50% recovery time reduction



>95% IO overheads reduction

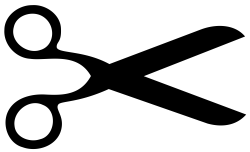


Other Contributions

Garbage collect FTL metadata

Recovering RAM-resident FTL metadata

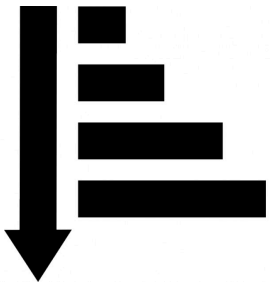
Partition entries to maximize buffer utilization



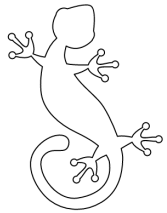
Wear leveling

Conclusion

GeckoFTL stores page validity metadata in flash

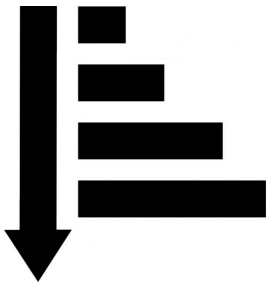


Buffer modifications
Reorganize later



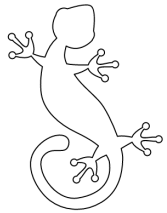
Conclusion

GeckoFTL stores page validity metadata in flash



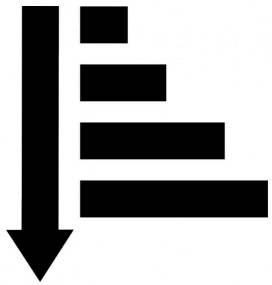
Buffer modifications
Reorganize later

↓ Integrated RAM
↓ Recovery time
↓ IO overheads



Conclusion

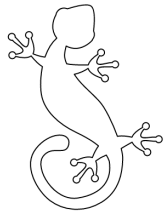
GeckoFTL stores page validity metadata in flash



Buffer modifications
Reorganize later

↓ Integrated RAM
↓ Recovery time
↓ IO overheads

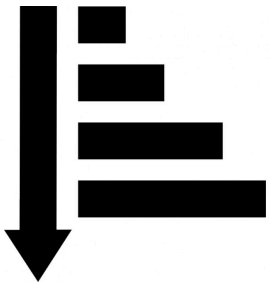
General applicability:



Store an update-intensive bitmap in secondary storage where adjacent bits are reset at the same time

Conclusion

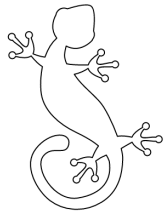
GeckoFTL stores page validity metadata in flash



Buffer modifications
Reorganize later

↓ Integrated RAM
↓ Recovery time
↓ IO overheads

General applicability:



Store an update-intensive bitmap in secondary storage where adjacent bits are reset at the same time

Thank you.