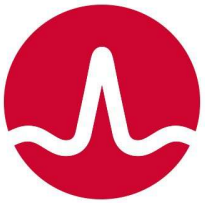
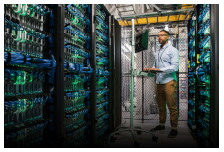
Leveraging Software Fault Tolerance for Longer Flash Hardware Lifespan

Aviad Zuck, Rob Johnson,
Donald E. Porter, Dan Tsafrir



goal: reduce carbon footprint of SSD storage



- **CO2e footprint = operational + embodied**
  (manufacturing)

- For SSD storage: mostly embodied
  (>50% according to literature)
- Result: switching to renewables less helpful than extending device use time
  – beyond inherent flash endurance limits

State of the art

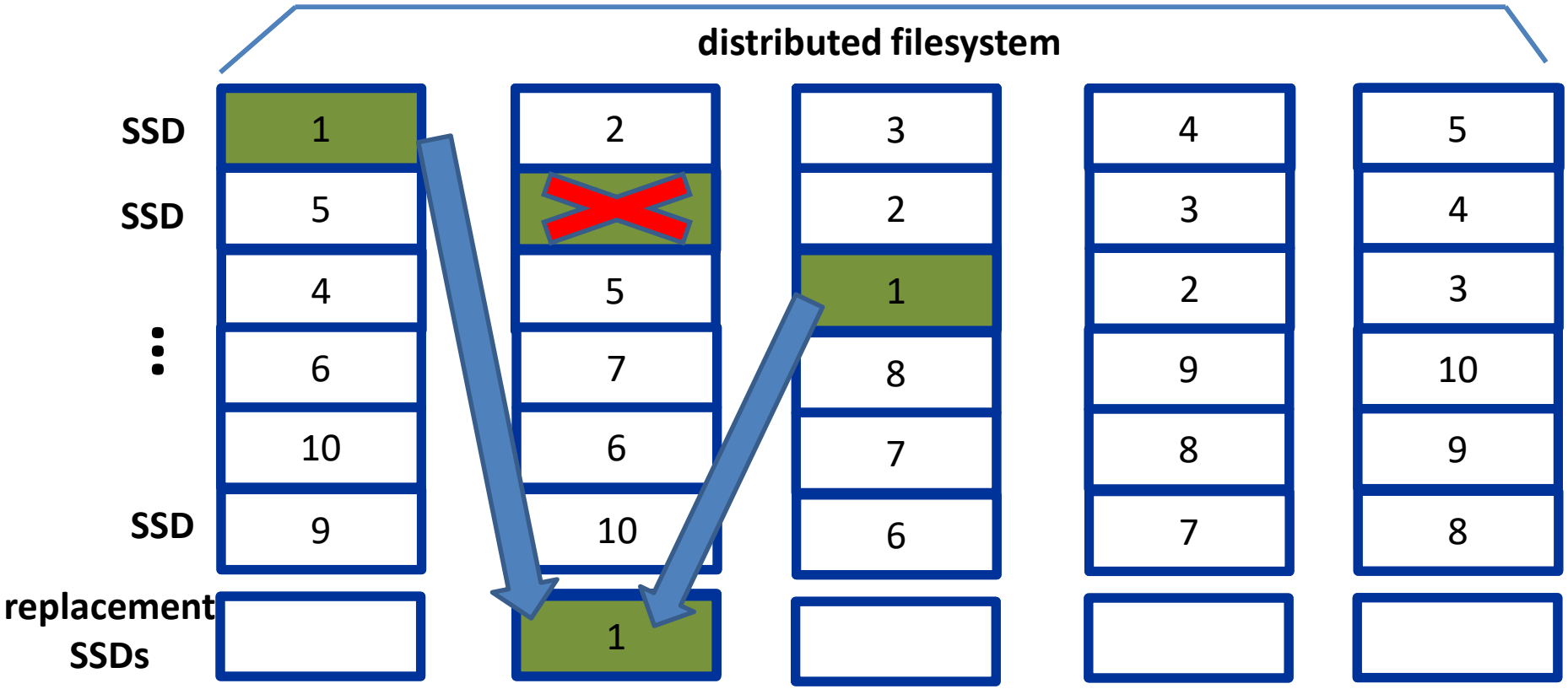
- extends single device use time by either
  - Leave (some) bits to flip beyond what can be corrected (SOS, HotOS'23)
  - Increase overprovisioning by dynamically shrinking capacity in free space (CVSS, FAST'24)

observation

1. Distributed file systems already accommodate both data loss & shrinking capacity
2. Datacenter SSD lifetime highly under-utilized

distributed systems accommodate data loss

- Recover **entire** device after drive failures
- Problem: SSDs never get to fail!



datacenter SSD lifetime highly under-utilized

"99% of systems use < 15% of rated life of drives" **FAST'20 (Netapp)**

at observed usage rates "SSDs will last for more than 100 years in production without wearing out" **FAST'22 (Netapp)**

"98% of failures before drive sees 1500 P/E cycles... manufacturer guarantees 3000 P/E cycles" **SC'19 (30K SSDs, Google)**

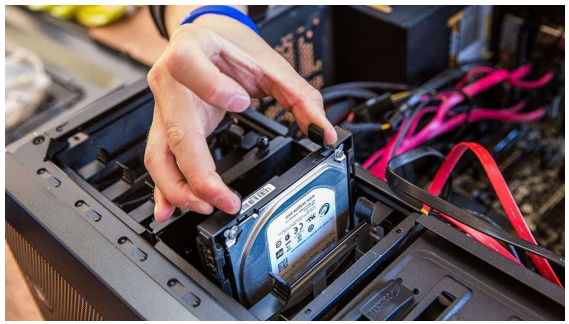
"rated SSD life used increases from 1.6% for 0-1 year old to 67.5% for 5-6 years old on average" **FAST'21 (1M drives, Alibaba)**

"(*retired) datacenter SSDs used well under their ratings****"*** **ServeTheHome (2024)**

reasons for low utilization

- Short datacenter hardware refresh cycles (3-5y)
  - Preemptive failure mitigation

Underlying reason:
static failure behavior of SSDs

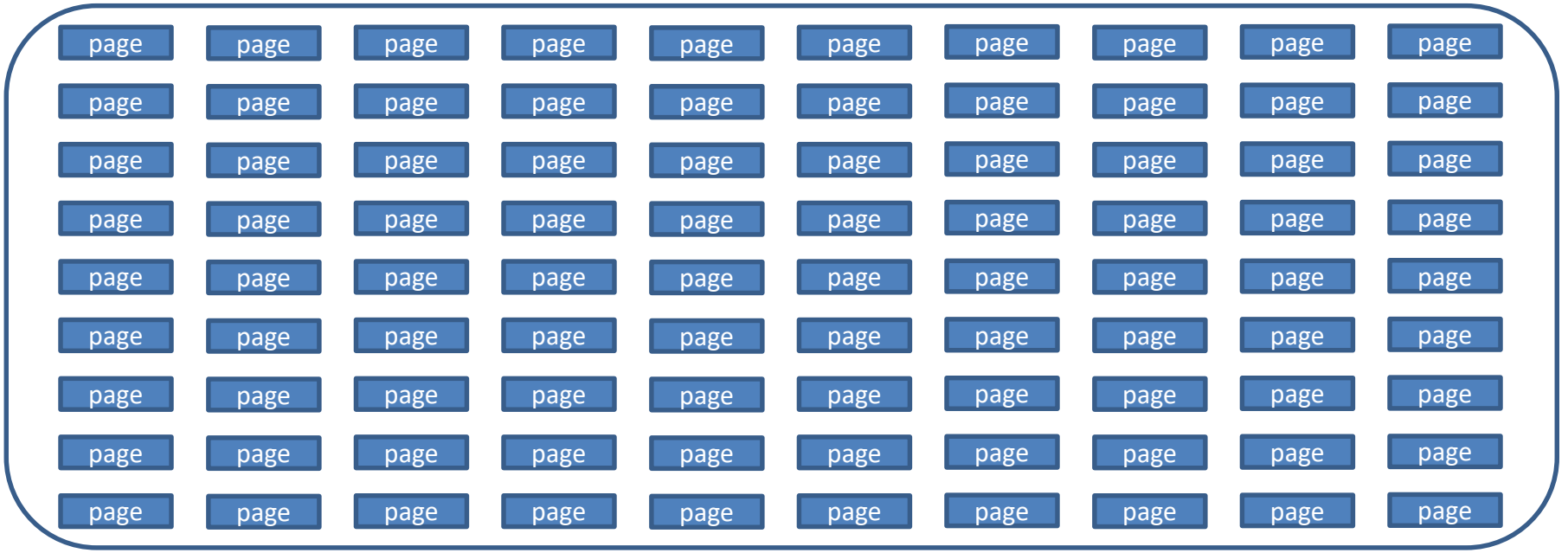


static SSD failure behavior

- Page writes reduce reliability over time (bit flips!)
- **Entire device** fails when **2.5%** of flash worn-out

distributed systems can accommodate shrinking devices

- Regularly handle millions of independent devices
- Just need to find the right abstraction for SSDs
  - Fits the underlying nature of flash storage

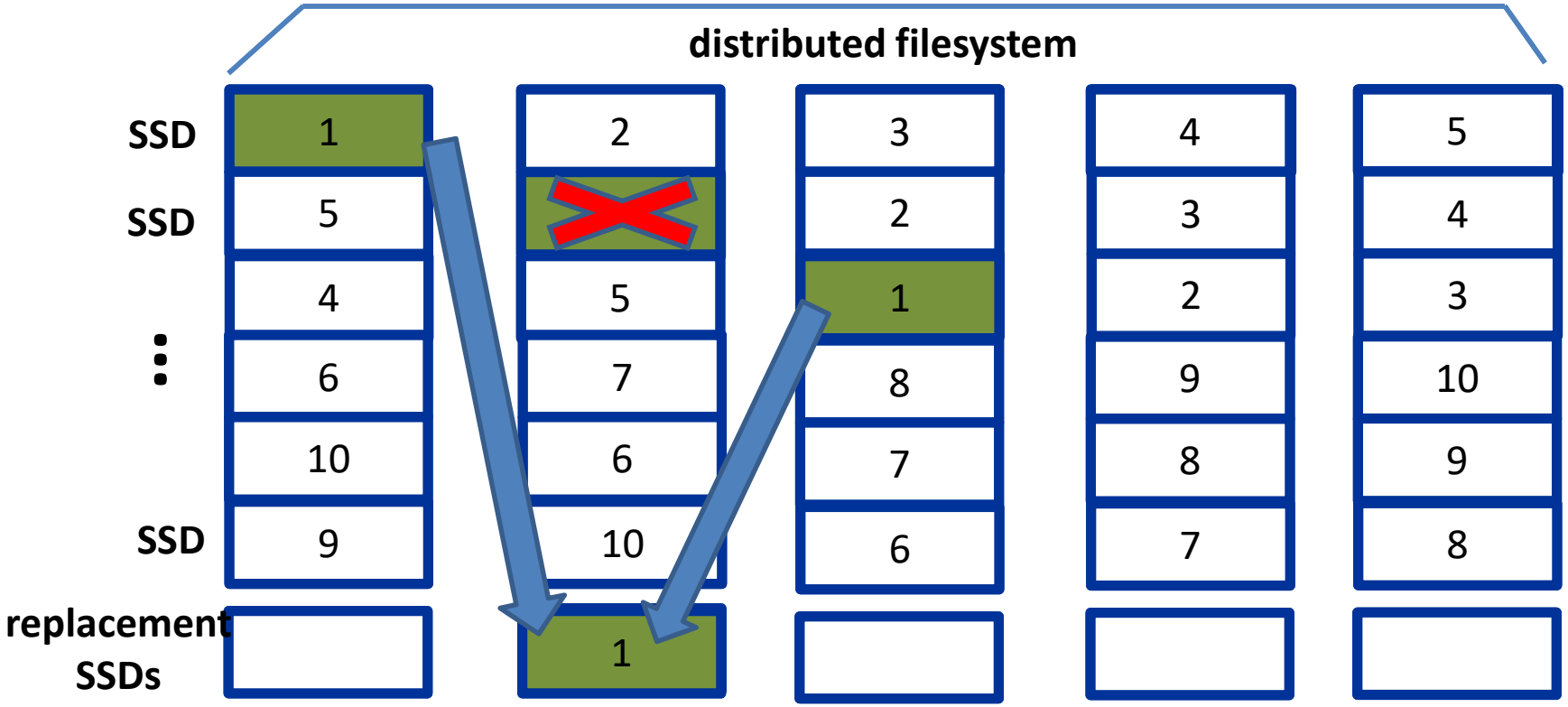


this work

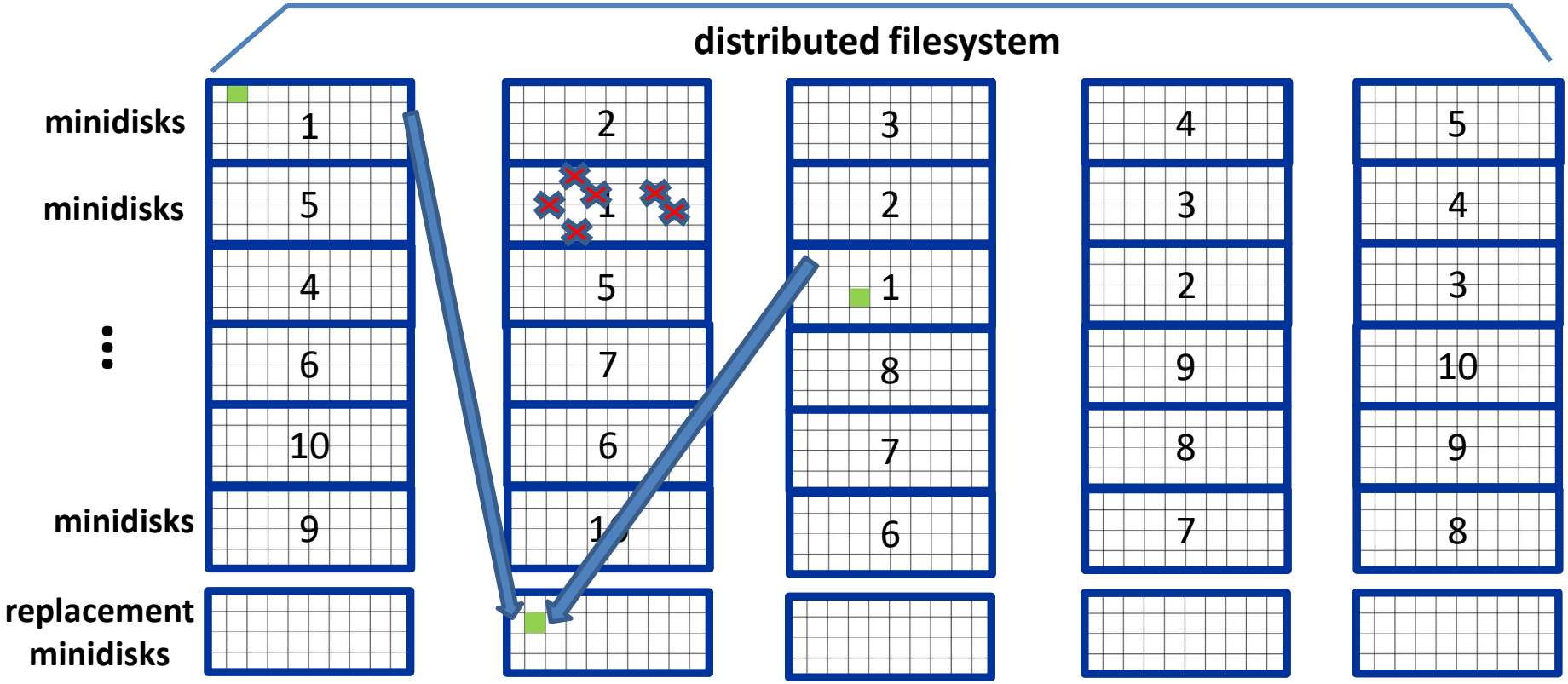
Extend SSD lifetime in distributed systems by supporting incremental and graceful SSD failures

- Obviate need for preemptive full device retirement
- Reduce manufacturing of replacement SSDs and related emissions
- Enable using denser, less-endurant flash (e.g., QLC)
  - Cheaper
  - More sustainable

previously we had disks

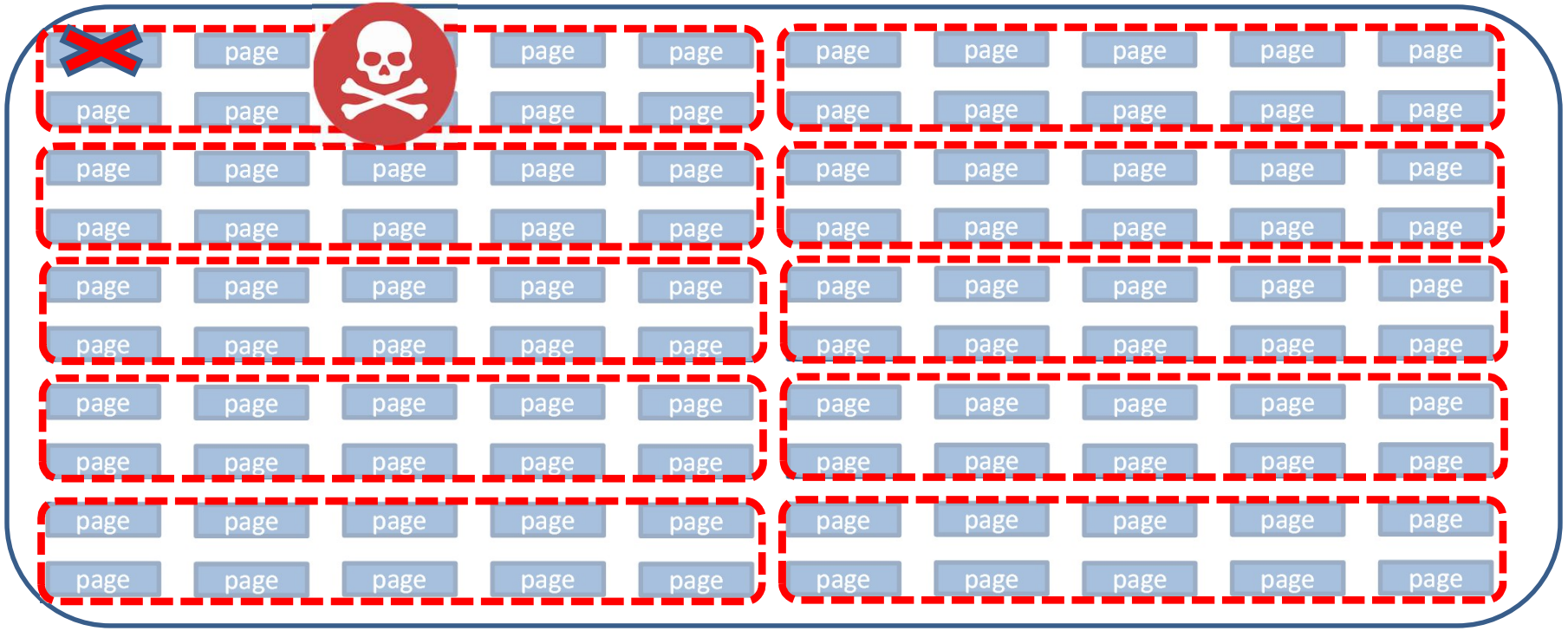


now we have *"minidisks"*



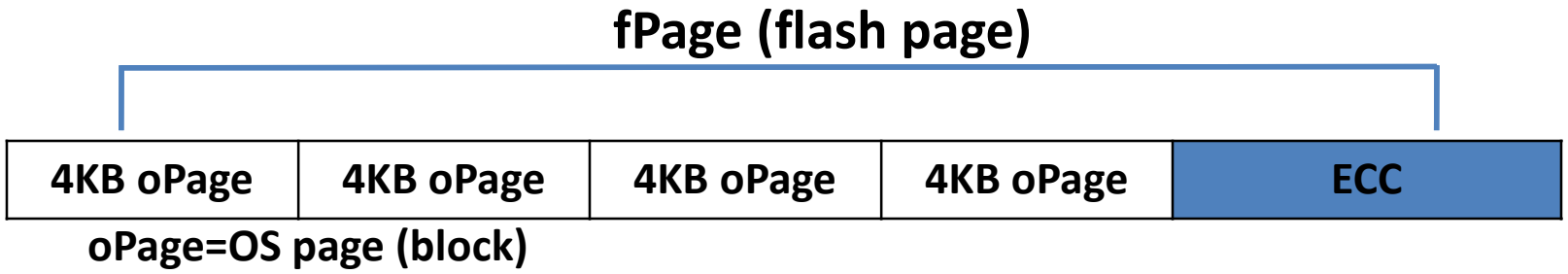
- Physical SSD appears as multiple tiny "minidisks" (e.g. 128MB)
- Minidisk failures induce small **incremental recoveries without retiring entire device**
- Distributed filesystem unchanged: just manages more "disks"

minidisks under the hood



$$\frac{1\text{TB } SSD}{128\text{MB minidisk size}} = 8\text{K minidisks}$$

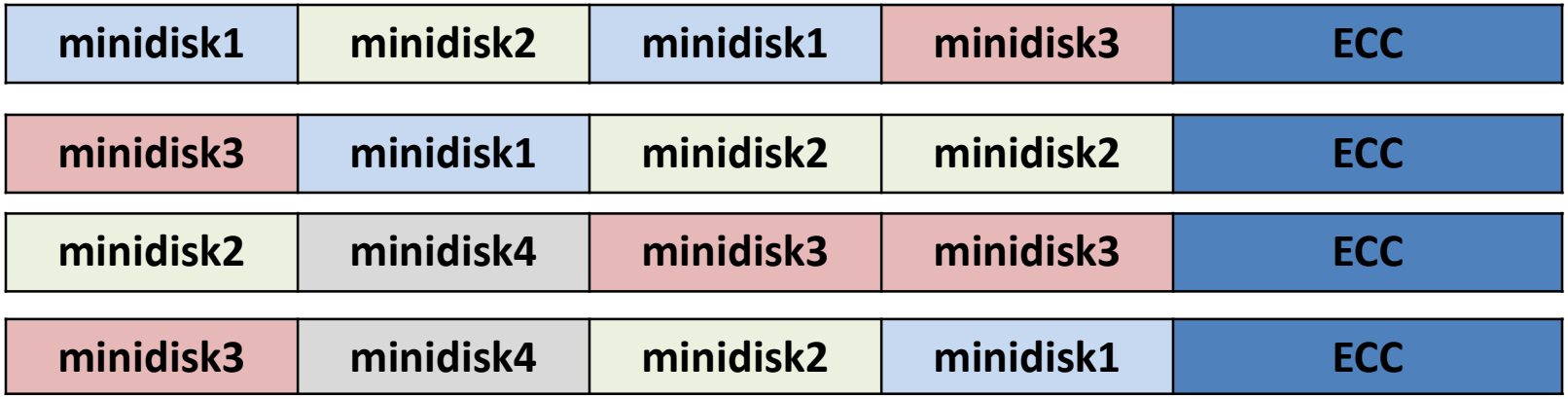
minidisk failures in detail



- Writes induce errors, corrected by ECC
- Lifetime/endurance = max # of writes to page before errors > ECC capability → page worn-out
  → data loss

- What happens when an fPage wears out?

minidisks are a logical abstraction (I)

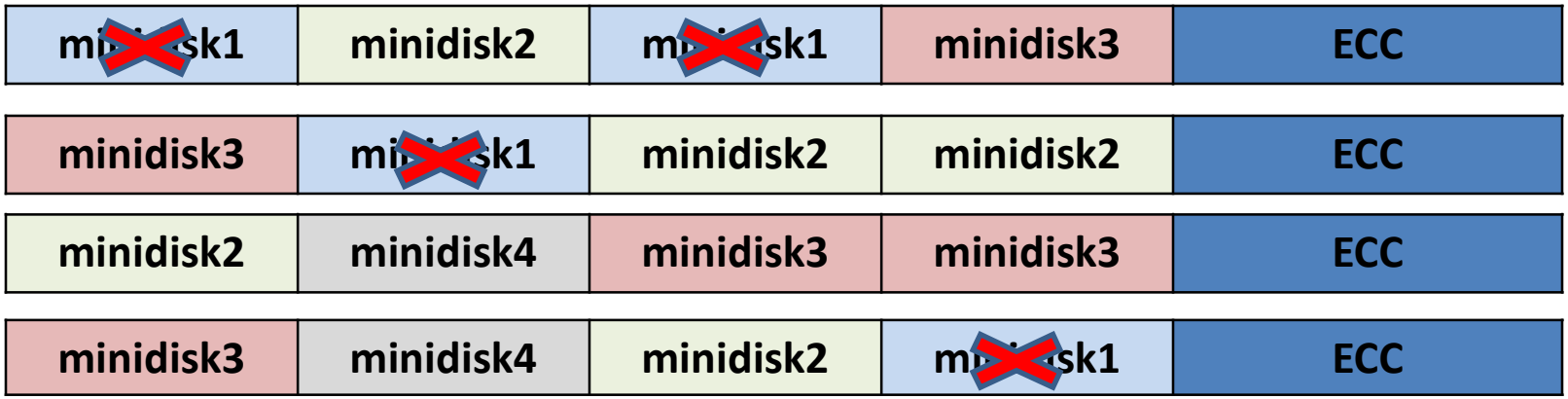


- Every physical *fPage* has logical *oPages* from multiple minidisks
- When an *fPage* wears out the drive should fail all related minidisks

- Let's optimize

minidisks are a logical abstraction (II)

- **logically** deallocate/trim all oPages of a **victim** minidisk



- Firmware correspondingly retires 128MB of **physical** flash from most worn-out pages
  - Can idly use garbage collection to migrate live data
  - Or do it preemptively

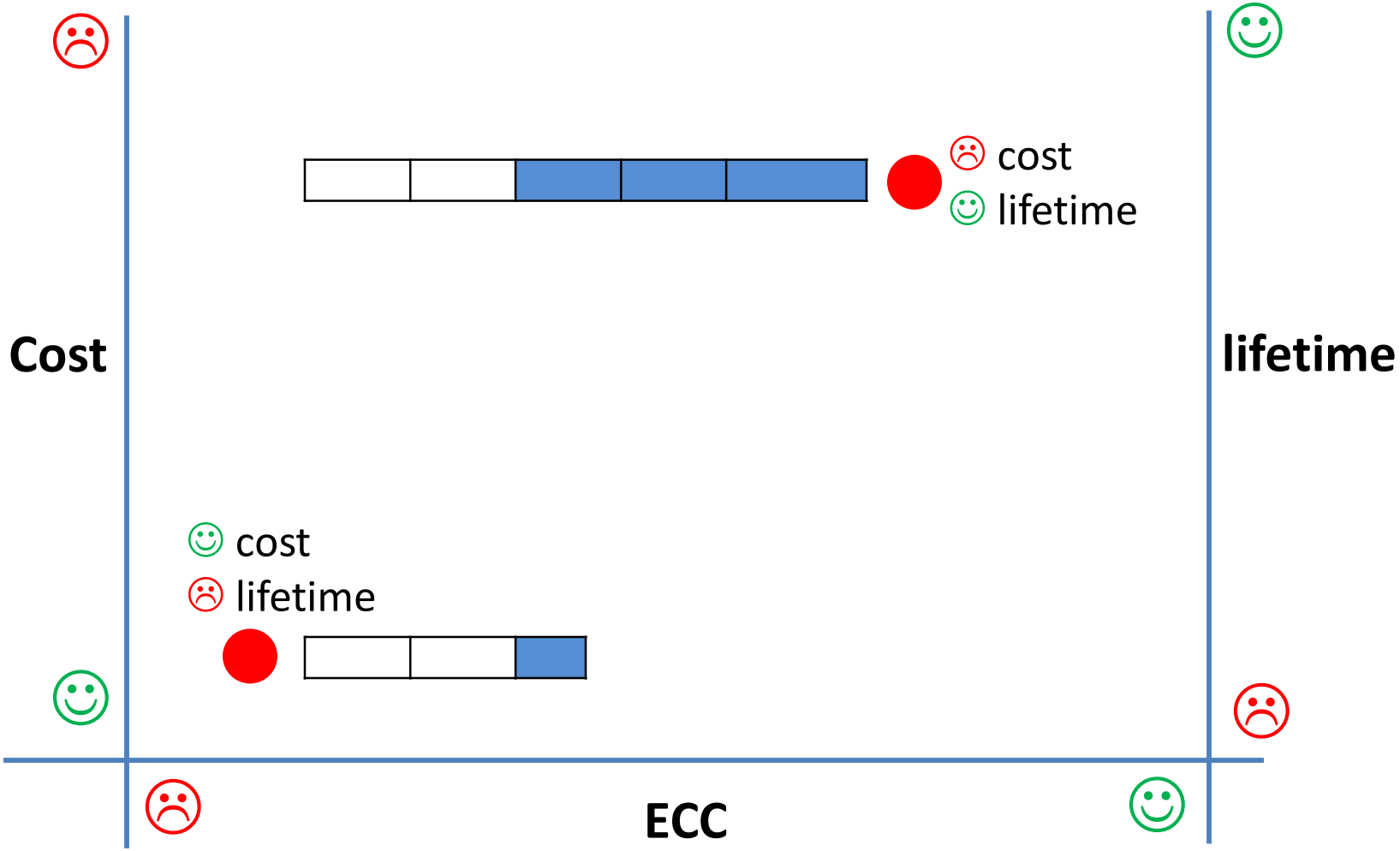
the benefits

	CVSS (FAST'24)	minidisks
Applicable to dist. setup	☹	☺
Software complexity	☹	☺
Space savings	20% (simulated)	>=20% (speculated)

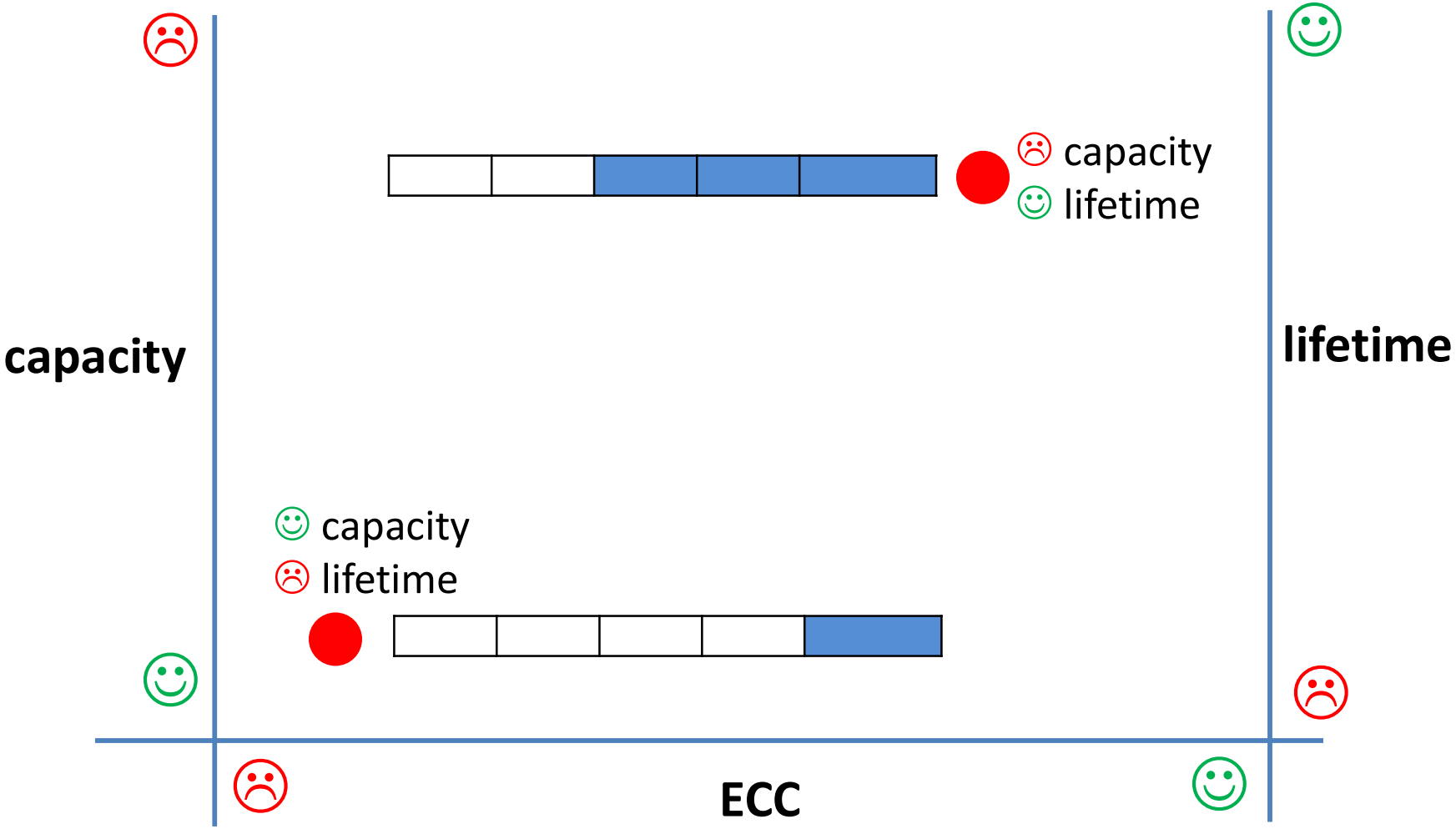
- We did that just by organizing data a bit differently

we can do even better!

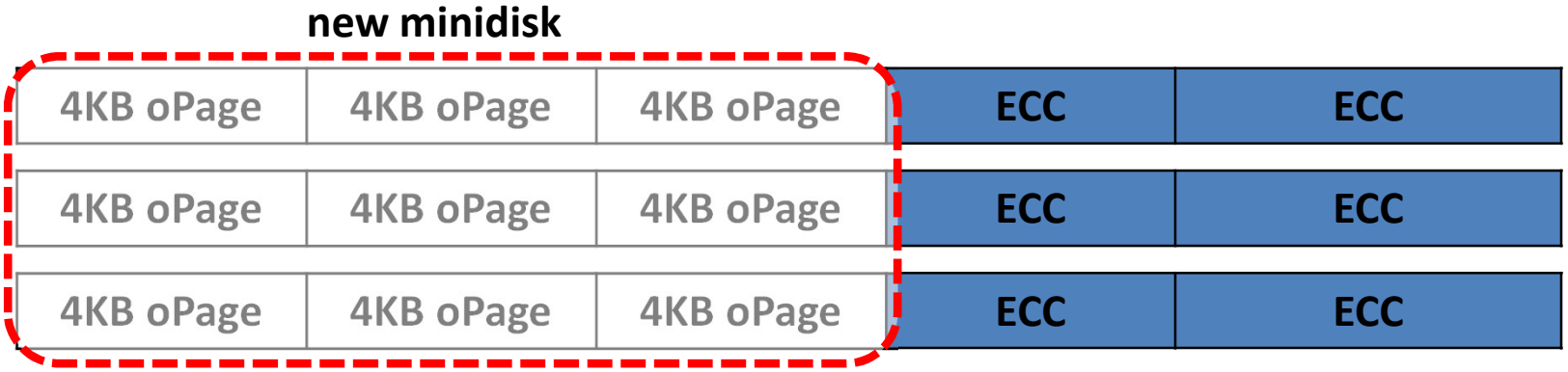
ECC vs. cost static tradeoff



we propose to be dynamic



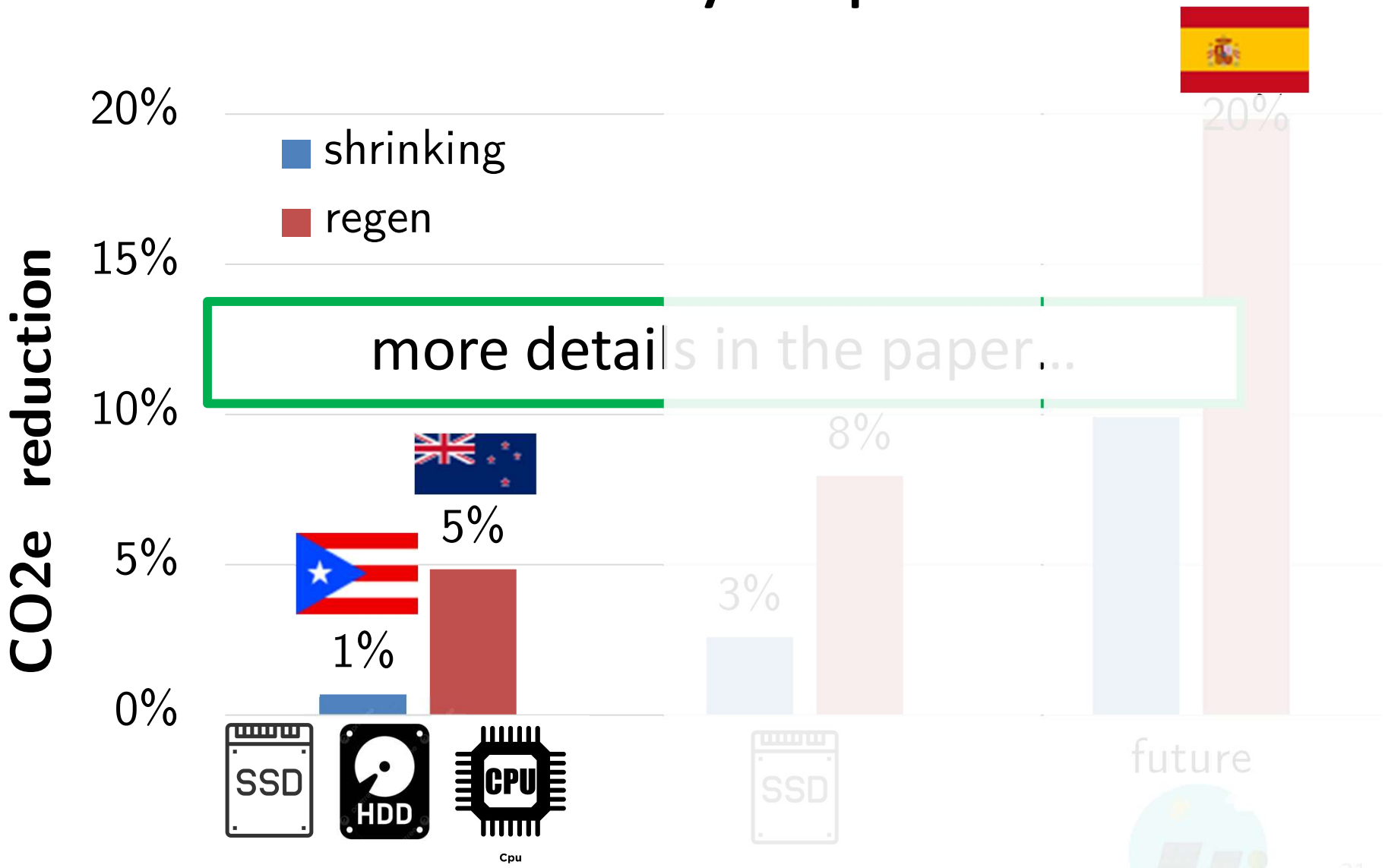
regenerating minidisks



- We can re-purpose one *oPage* to ECC
- At some point we have enough extra *oPages* to **regenerate** a new minidisk
  - Hot-plug into distributed system

- **50% estimated lifetime gains for remaining *oPages*** (see paper)

sustainability implications



summary

- SSDs are an array of parallel hardware
  → should expose independent failure domains
- Current literature: longer flash lifetime use requires overprovisioning and/or tolerating bit flips
- Distributed file systems can already accommodate both!
- Minidisks are easy to implement in an SSD, and easy to use in a distributed file system
- 20-50% lifetime gains, 8-20% carbon gains

Questions?

Aviad Zuck aviadz@cs.technion.ac.il