Solid-state storage & DBMS

CIDR 2013

Manos Athanassoulis

DATA-INTENSIVE APPLICATIONS AND SYSTEMS

EPFL

Flash is replacing disks

But, random writes on flash

- Instability in performance
- Shorter device lifetime

And flash capacity is expensive

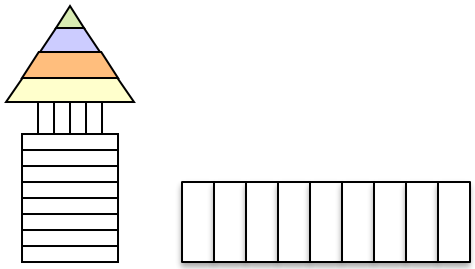
Is flash ready/suitable for secondary storage?

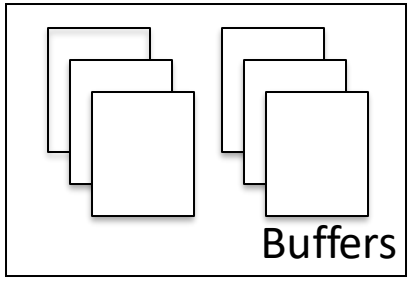
Reverse: How can DBMS exploit flash?

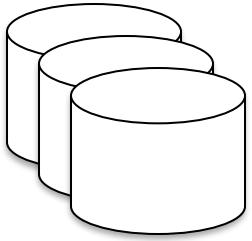
Flash not a drop-in replacement for disks

Disks all over the place

- DBMS modules assumes disks
  - Storage
  - Buffer management
  - Indexing
  - Logging
- Lower-level:
  - Block APIs

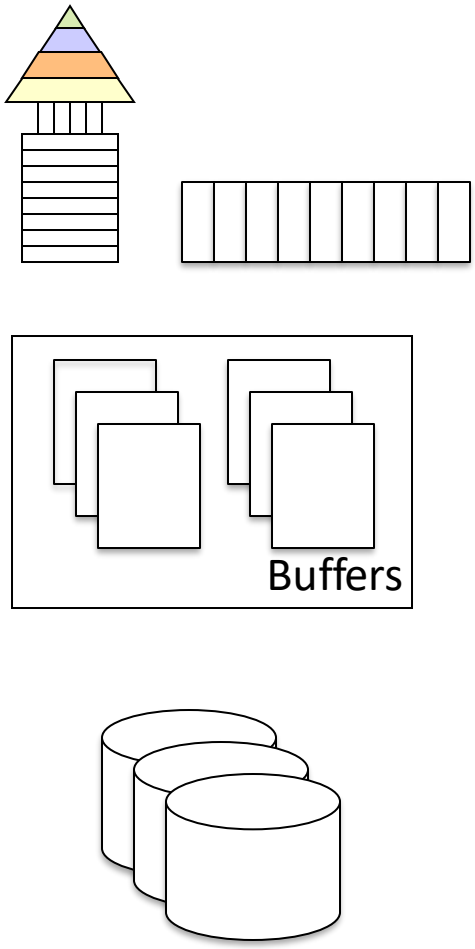






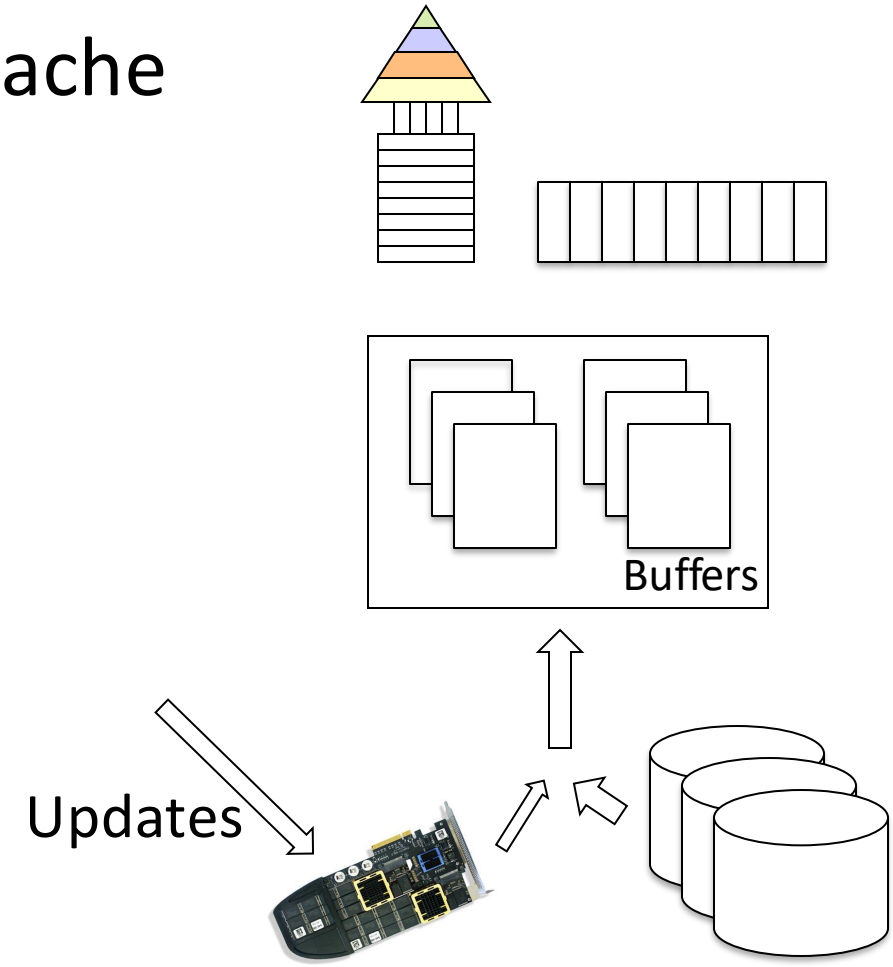
Re-architect DBMS needed for solid-state storage

How to use flash in a “disk-world”?



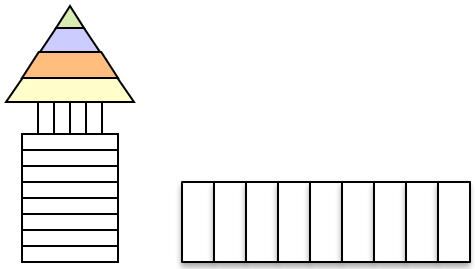
How to use flash in a “disk-world”?

- As a specialized write cache
  - Hides update overhead

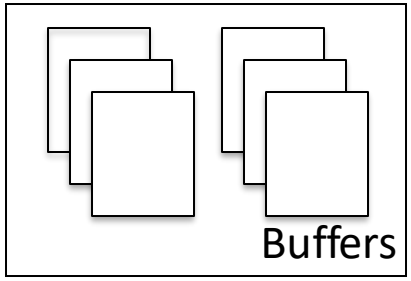


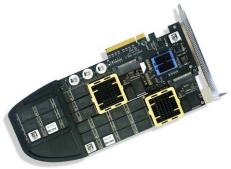
How to use flash in a “disk-world”?

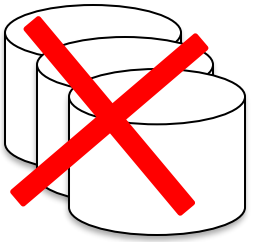
- As a specialized write cache
  - Hides update overhead



- As secondary storage
  - Offers efficient random access

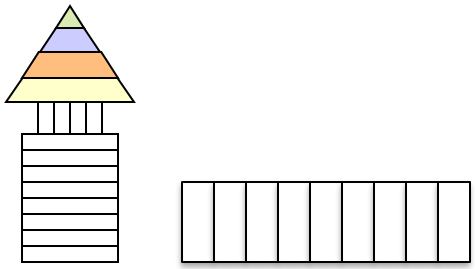




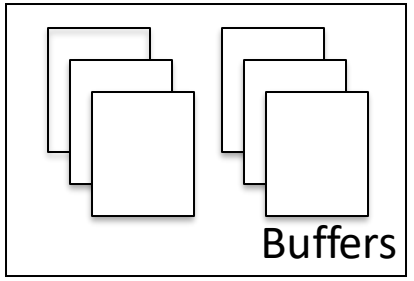


How to use flash in a “disk-world”?

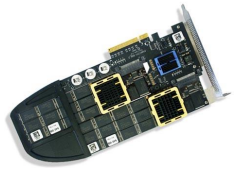
- As a specialized write cache
  - Hides update overhead



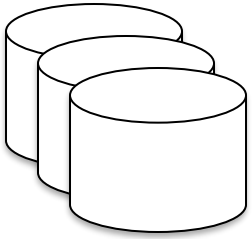
- As secondary storage
  - Offers efficient random access



- As a “regular” cache level

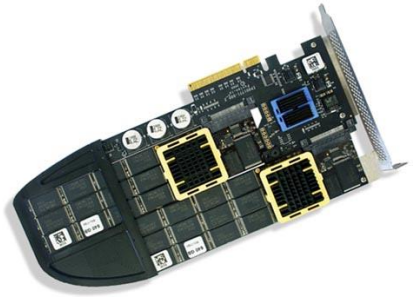


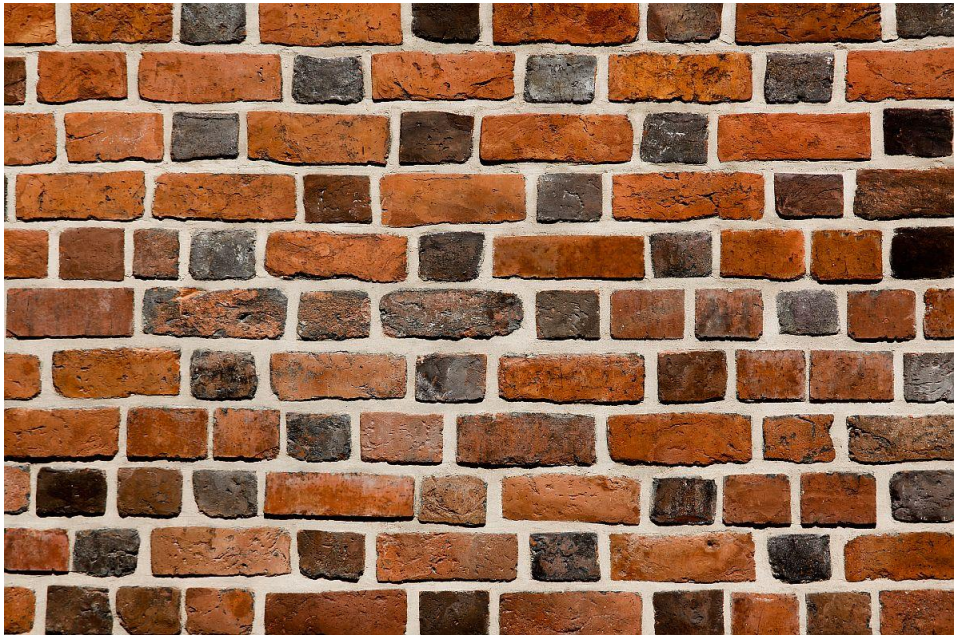
SSD
Buffer



Flash, solid-state, DBMS:
The challenges are far from done …

Flash Wall

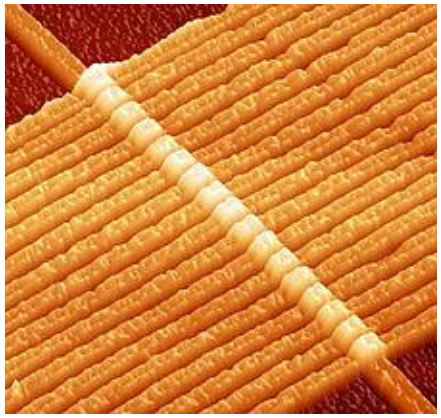




Capacity: Density will plateau (soon)

Endurance (already)

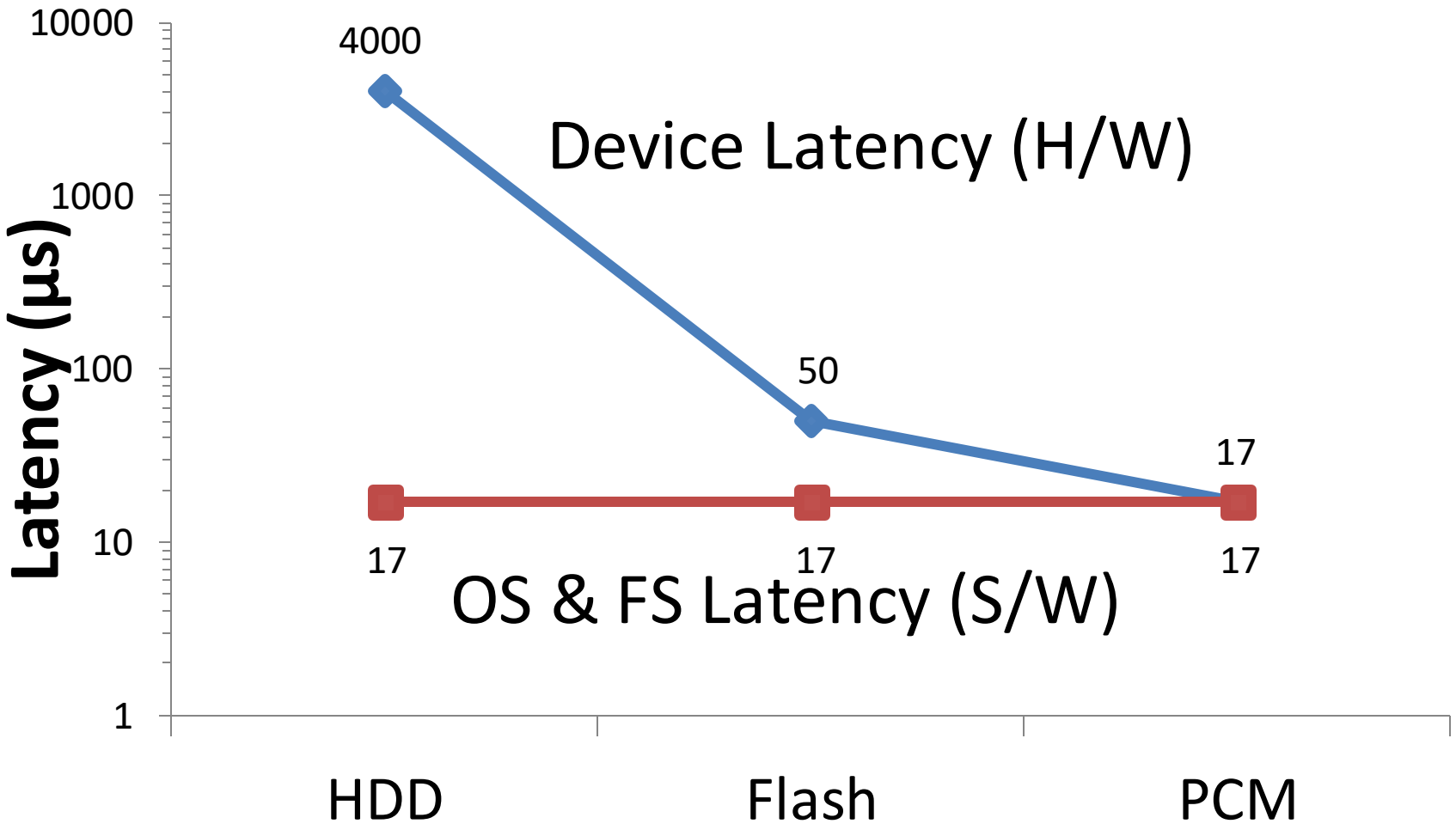
PCM, memristor and co



More than flash:

How to use persistent main memory?

Software stack is too slow



Flash Wall(s)

Persistent main memory

OS & FS too slow

Thank you!

Manos Athanassoulis