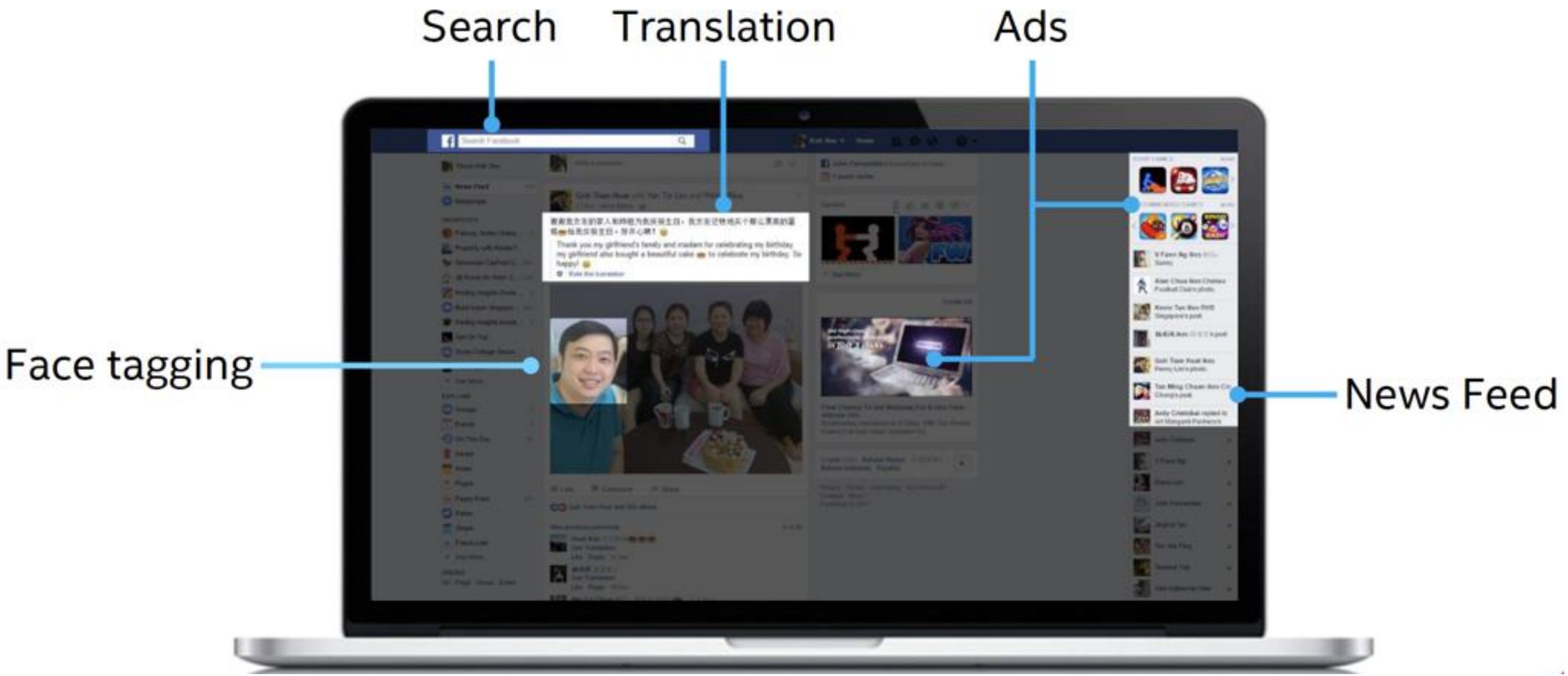
Use an Intelligent SSD to Accelerate Machine Learning

Hung-Wei Tseng

University of California, Riverside

ML is everywhere



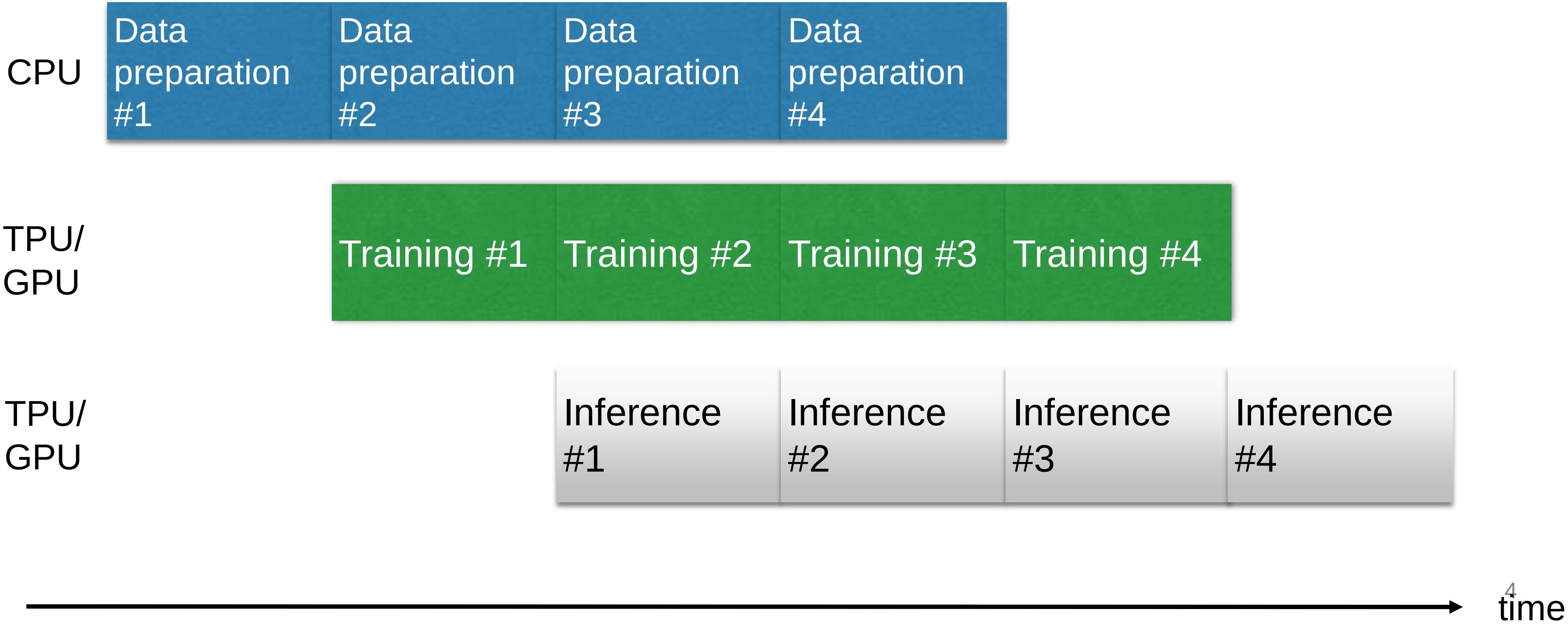
K. Hazelwood et al., "Applied Machine Learning at Facebook: A Datacenter Infrastructure Perspective," 2018 IEEE International Symposium on High Performance Computer Architecture (HPCA), Vienna, 2018, pp. 620-629.

ML is still timing consuming

	Resource	Training Frequency	Training Duration
Facer	GPUs + single socket CPUs	Every N Photos	Seconds
News Feed	Dual Socket CPUs	Daily	Hours
Lumos	GPUs	Multi-monthly	Hours
Search	Vertical Dependent	Hourly	Hours
Language Translation	GPUs	Weekly	Days
Sigma	Dual Socket CPUs	Sub-Daily	Hours
Speech Recognition	GPUs	Weekly	Hours

K. Hazelwood et al., "Applied Machine Learning at Facebook: A Datacenter Infrastructure Perspective," 2018 IEEE International Symposium on High Performance Computer Architecture (HPCA), Vienna, 2018, pp. 620-629.

The ML data processing pipeline



CPU

Data preparation #1

Data preparation #2

Data preparation #3

Data preparation #4

TPU/ GPU

Training #1

Training #2

Training #3

Training #4

TPU/ GPU

Inference #1

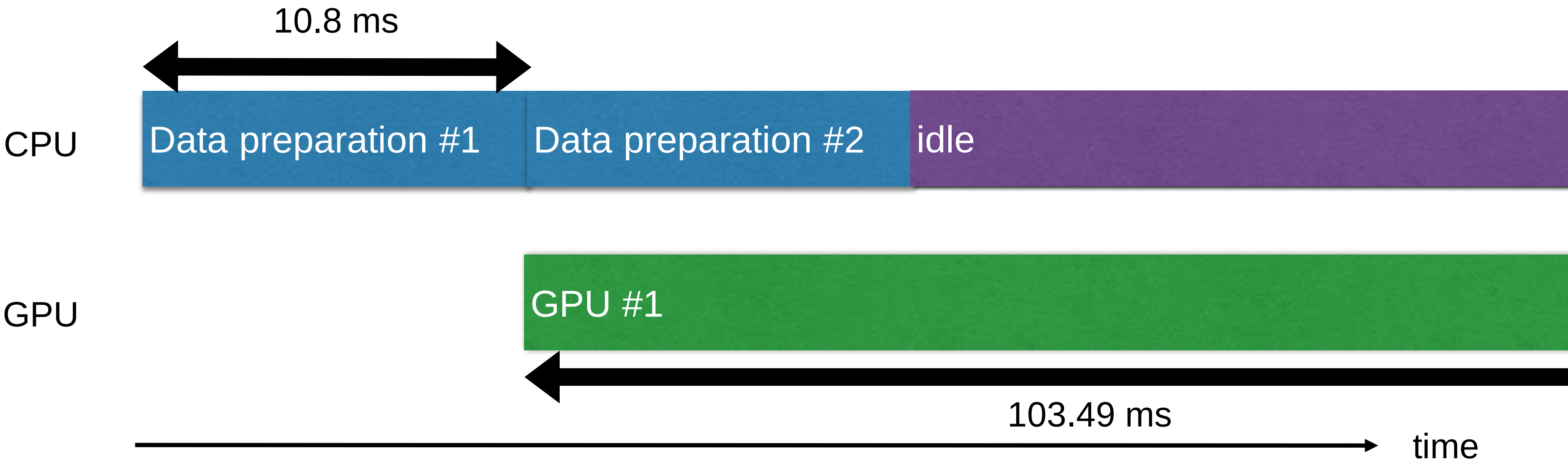
Inference #2

Inference #3

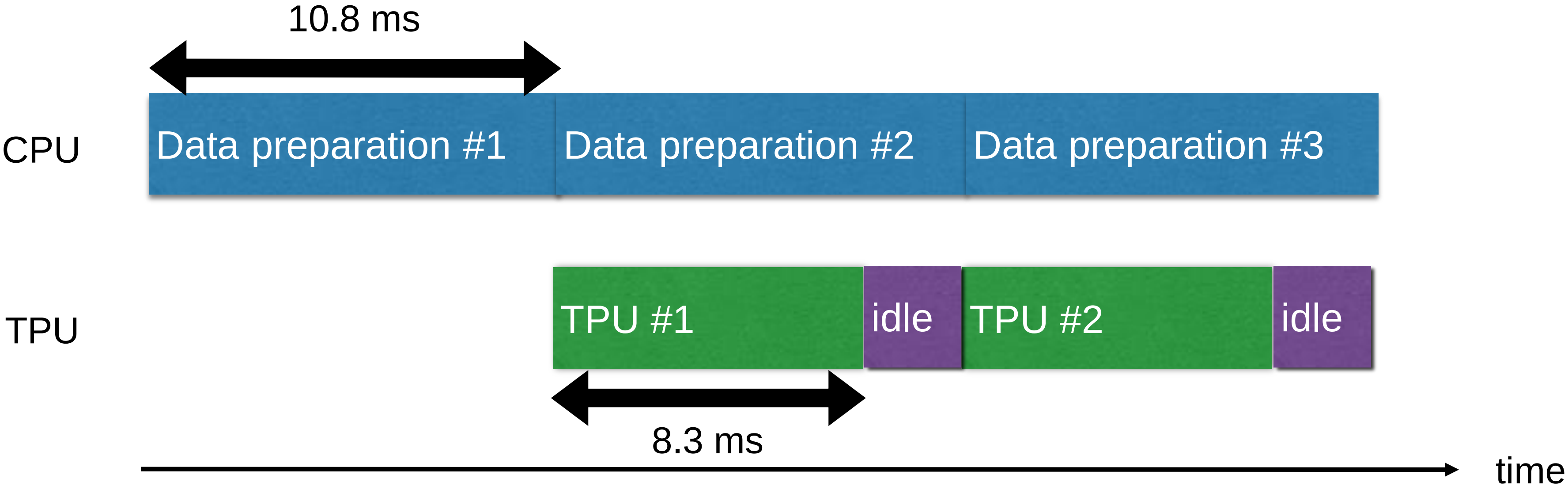
Inference #4

time

The ML data processing pipeline — GPU



The ML data processing pipeline — TPU



Tasks in this new bottleneck

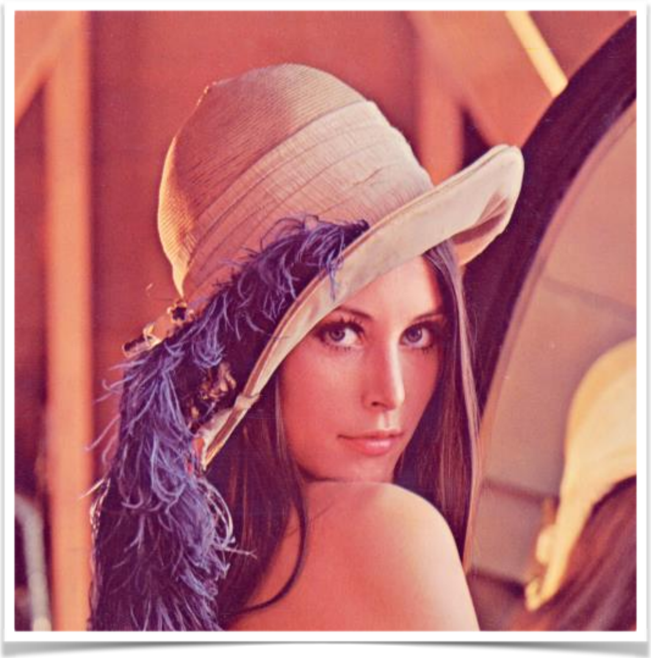
- Reading inputs
- Reduce precisions
- Shuffling data
- Create application objects

Outline

- Adjusting data resolutions in storage -- Varifocal Storage
- Shuffling data in storage
- Conclusion

We don’t need really detailed inputs

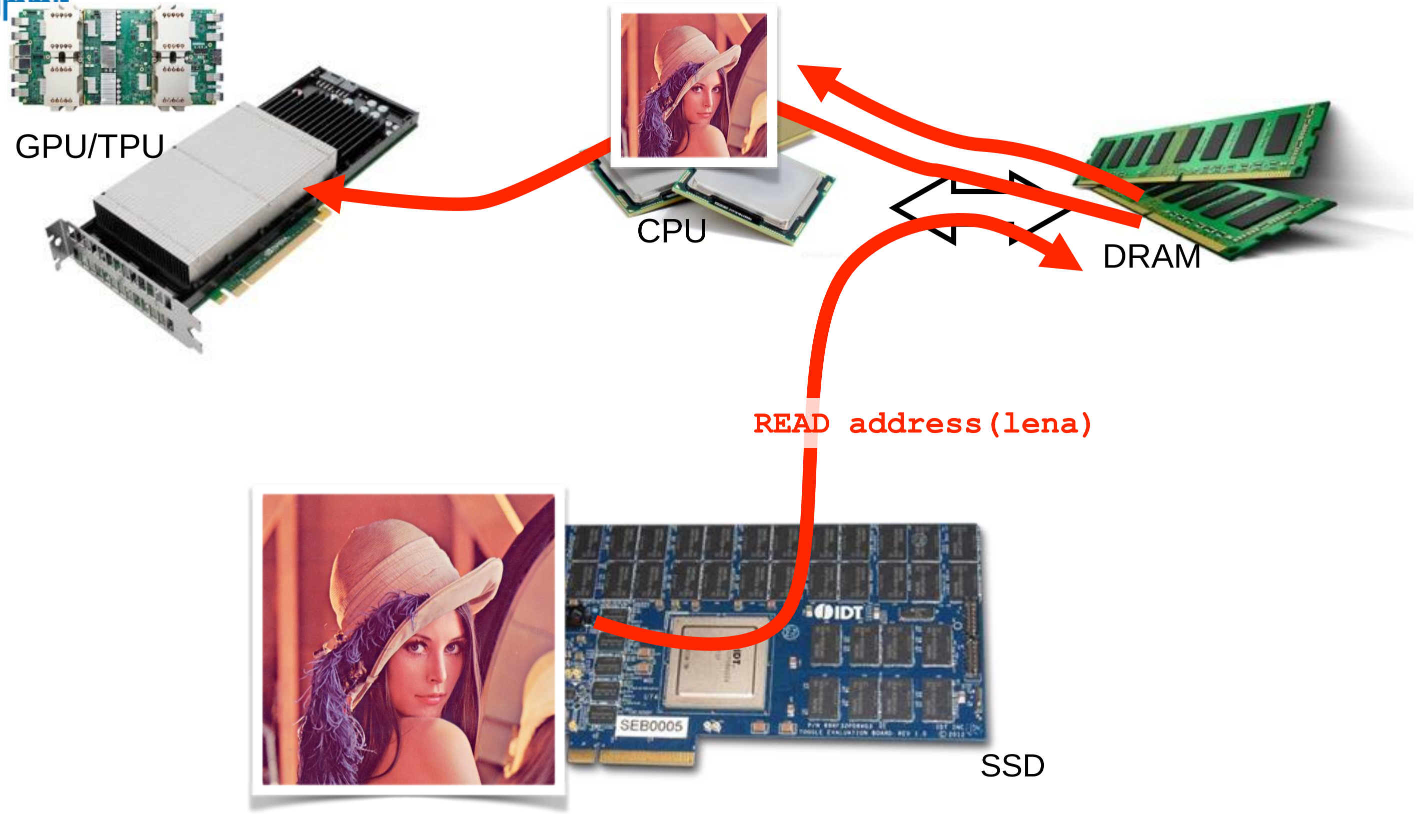
Reduce the resolution by 25%



Approximate Computing

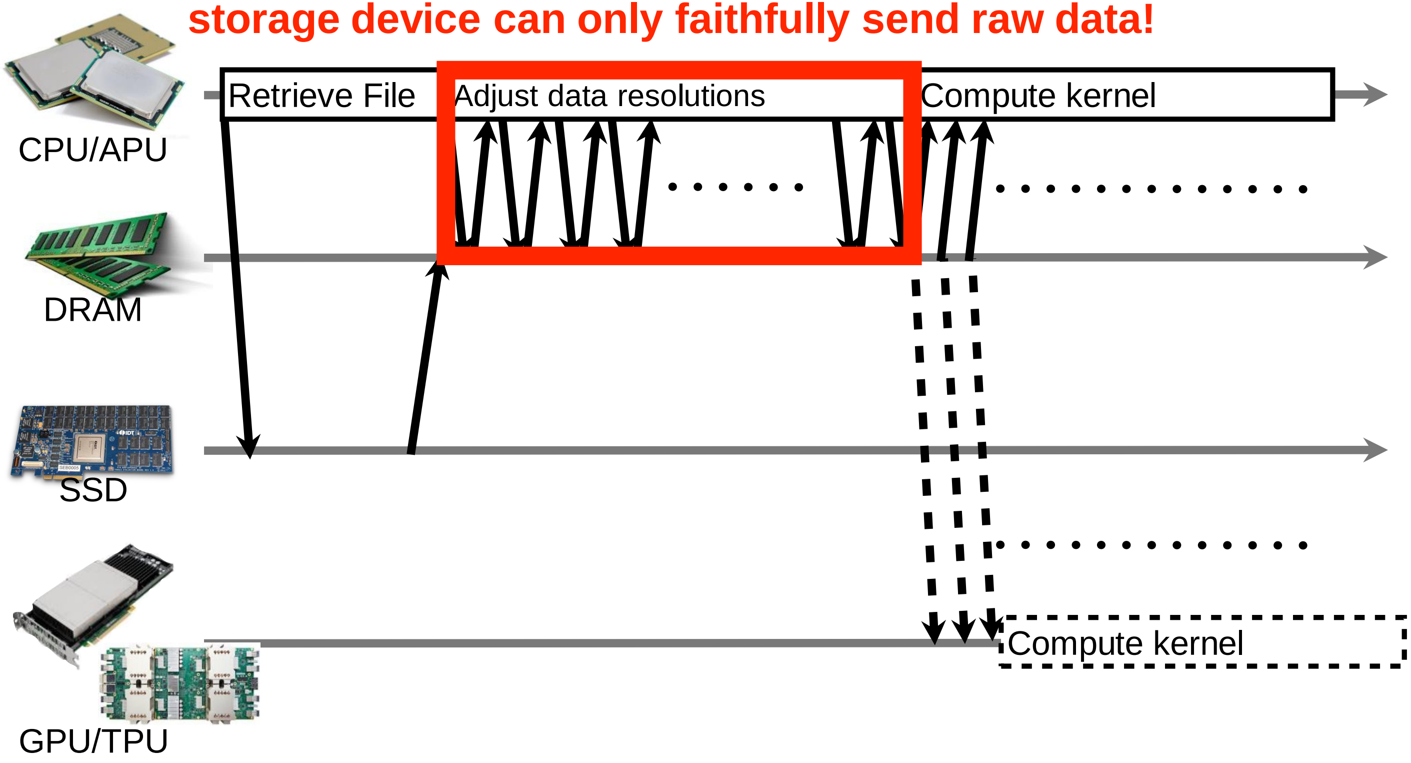
- A large set of applications can tolerate inaccuracies
  - Machine learning
  - Data mining
  - Video/Image processing
  - Scientific computing
- Benefits of approximate computing
  - Reduce the amount of computation
  - Simplify hardware design
  - Deliver higher throughputs
  - Improve the area-efficiency

Conventional Approximate Computing

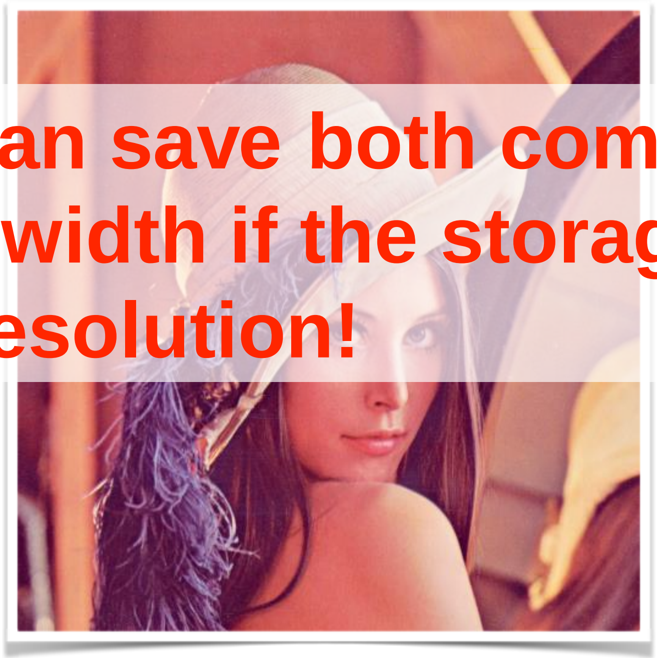


The conventional model

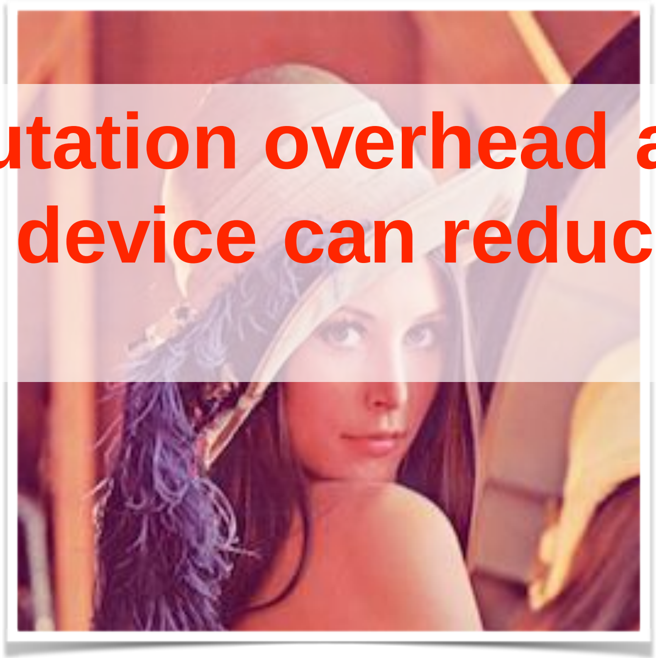
Needs to perform this on the host because the storage device can only faithfully send raw data!



We don’t need really detailed inputs

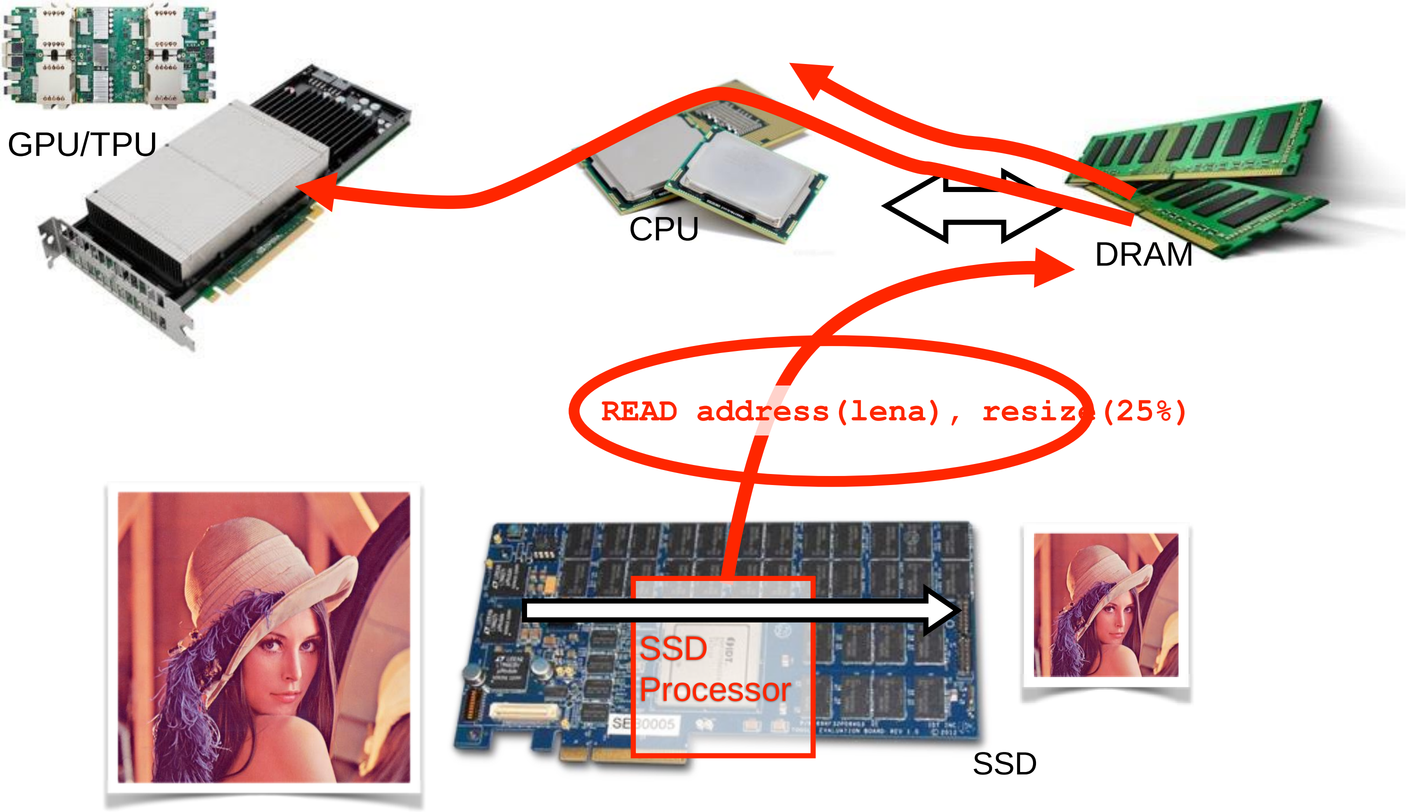


Reduce the resolution by 25%



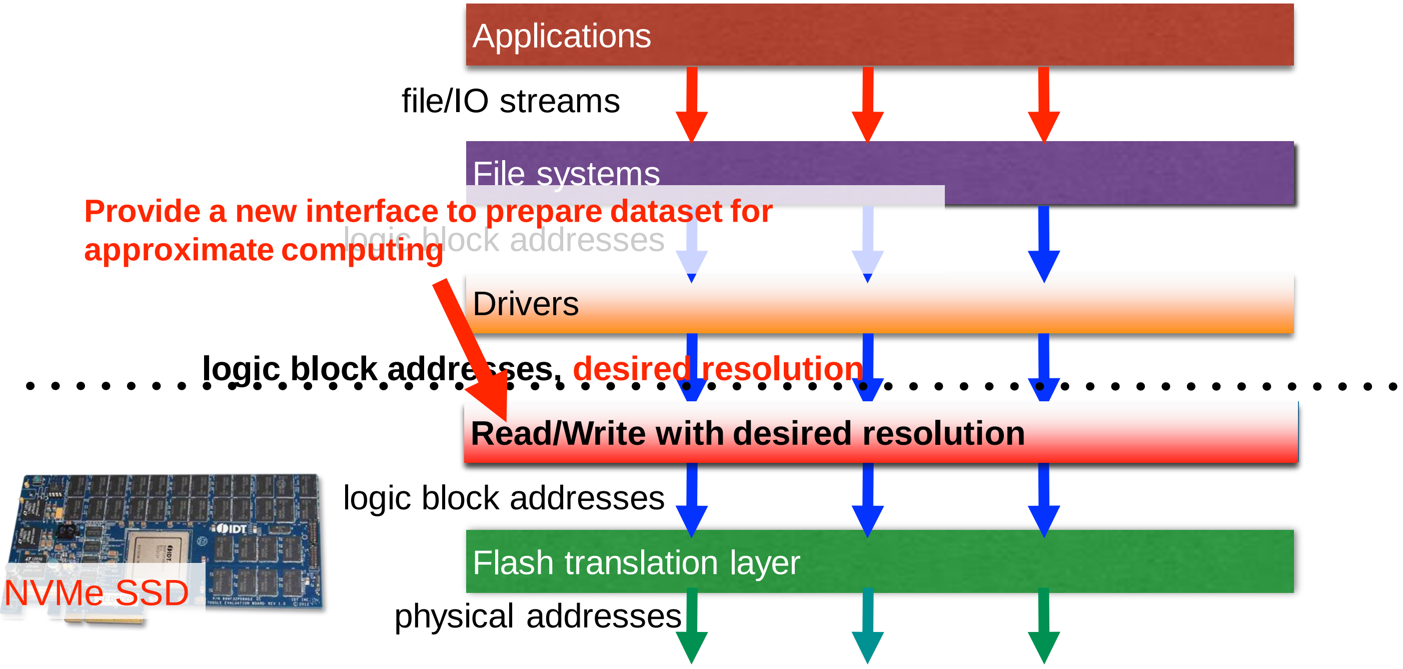
We can save both computation overhead and bandwidth if the storage device can reduce the resolution!

Kannon: dynamic multi-resolution storage



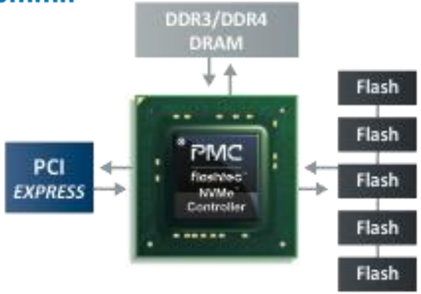
```
READ address(lena), resize(25%)
```

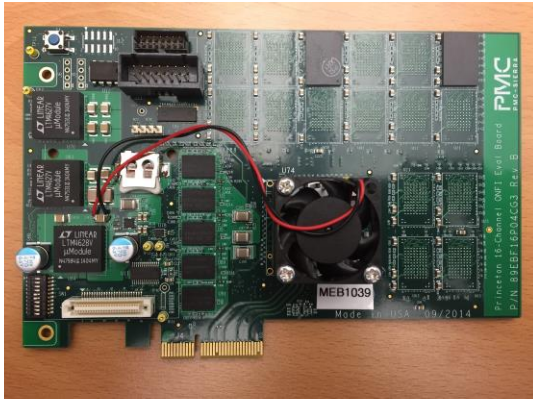
Varifocal Storage: dynamic multi-resolution storage



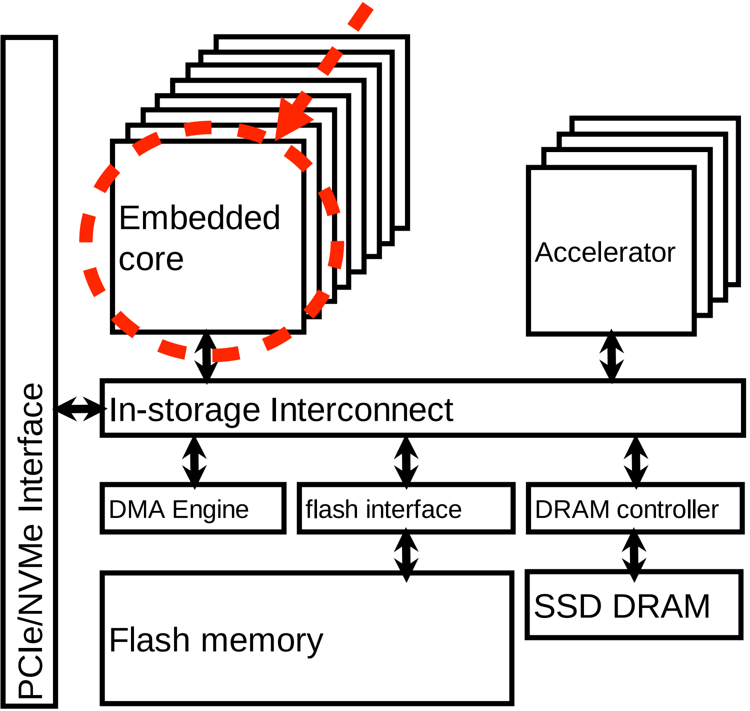
Provide a new interface to prepare dataset for approximate computing

Varifocal Storage

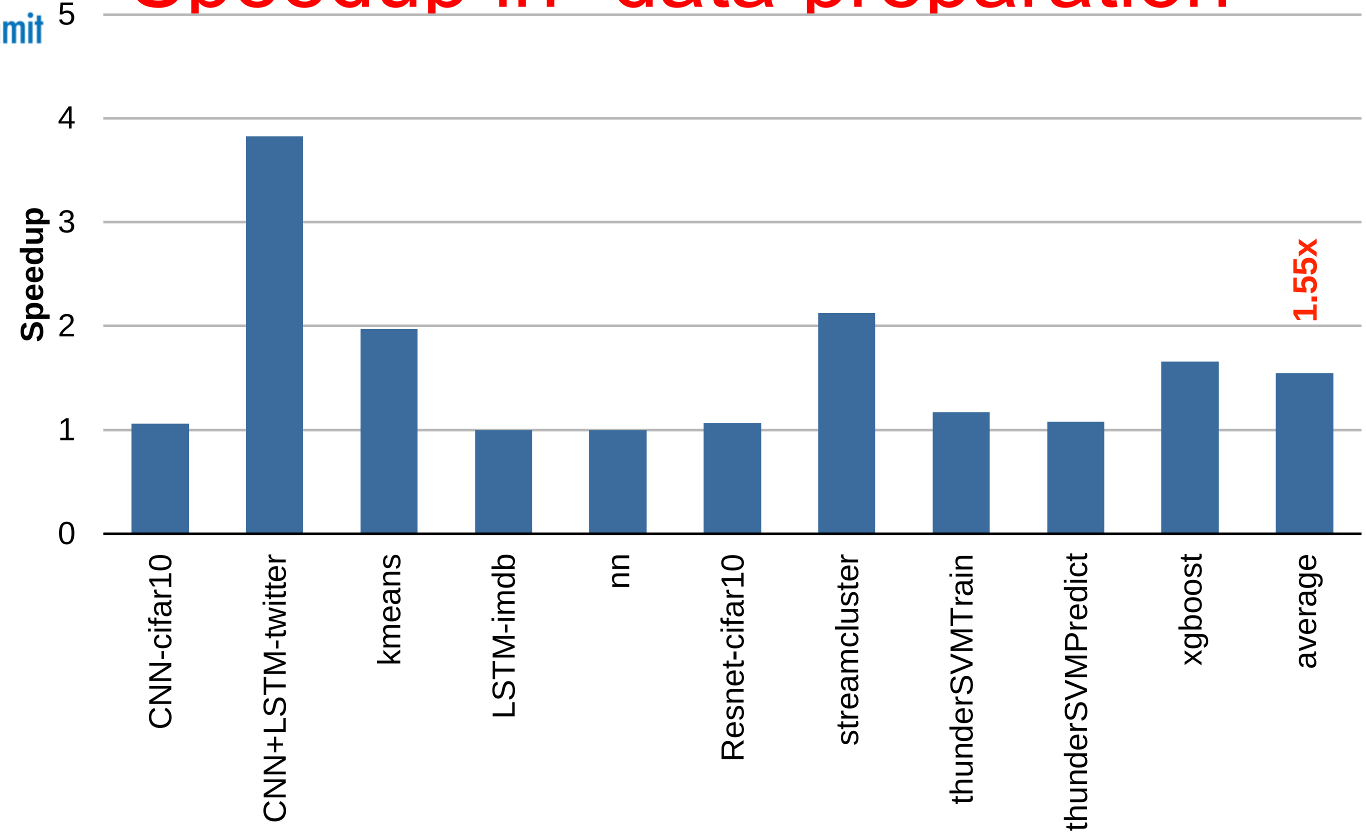




Managing NVMe commands
Adjusting data resolutions

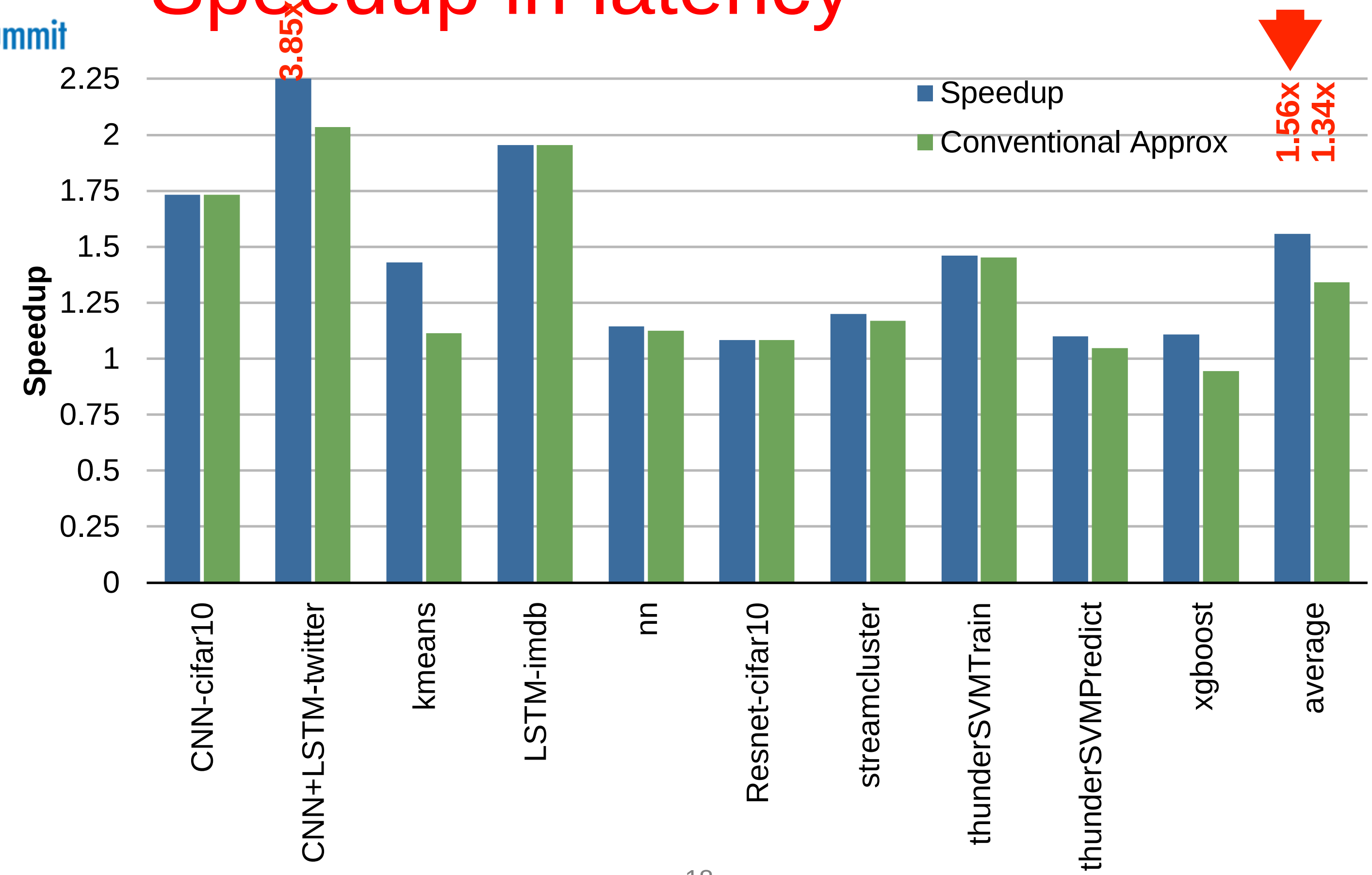


Speedup in “data preparation”



Flash Memory Summit

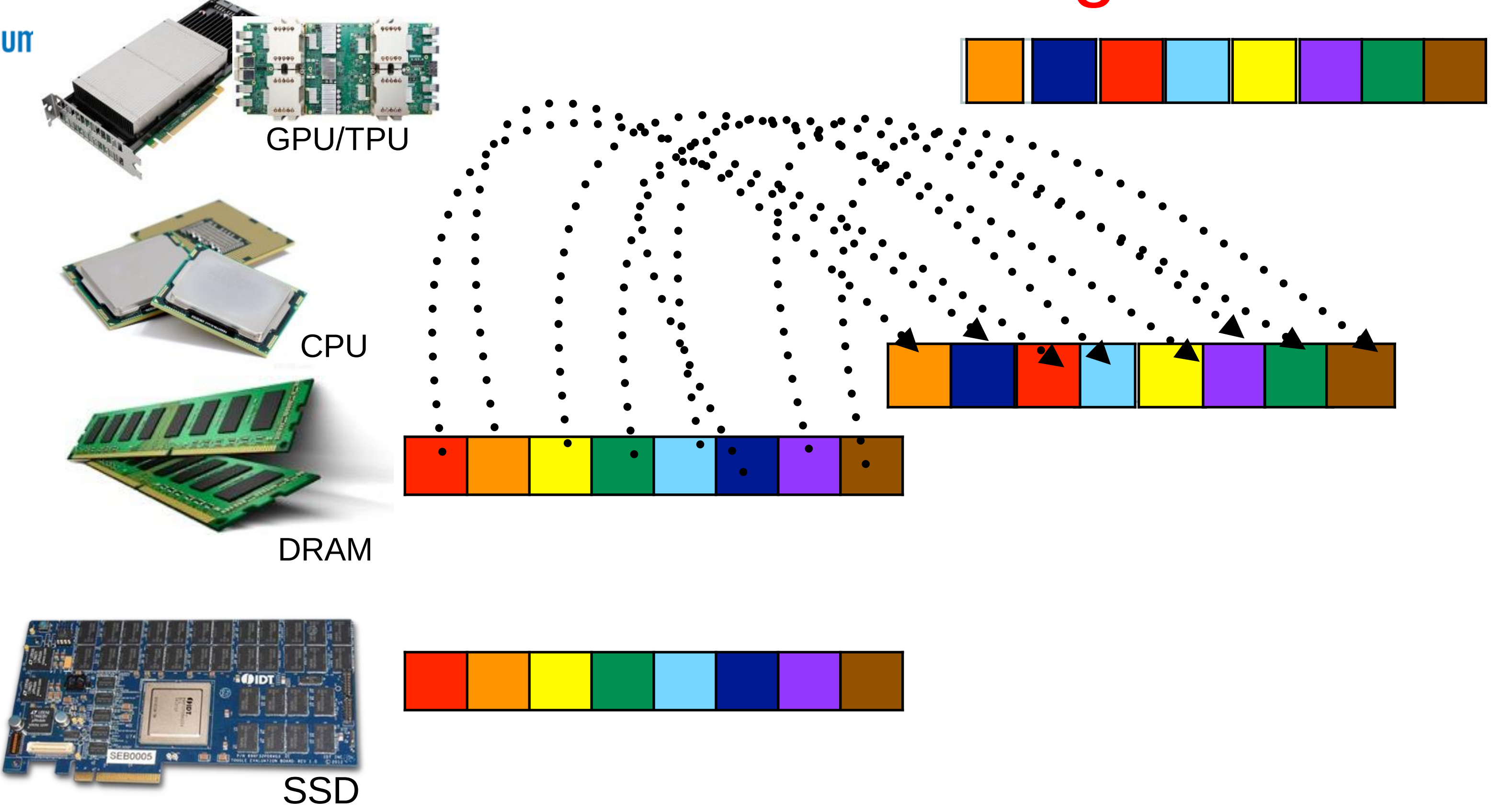
Speedup in latency



Outline

- Adjusting data resolutions in storage -- Varifocal Storage
- Shuffling data in storage
- Conclusion

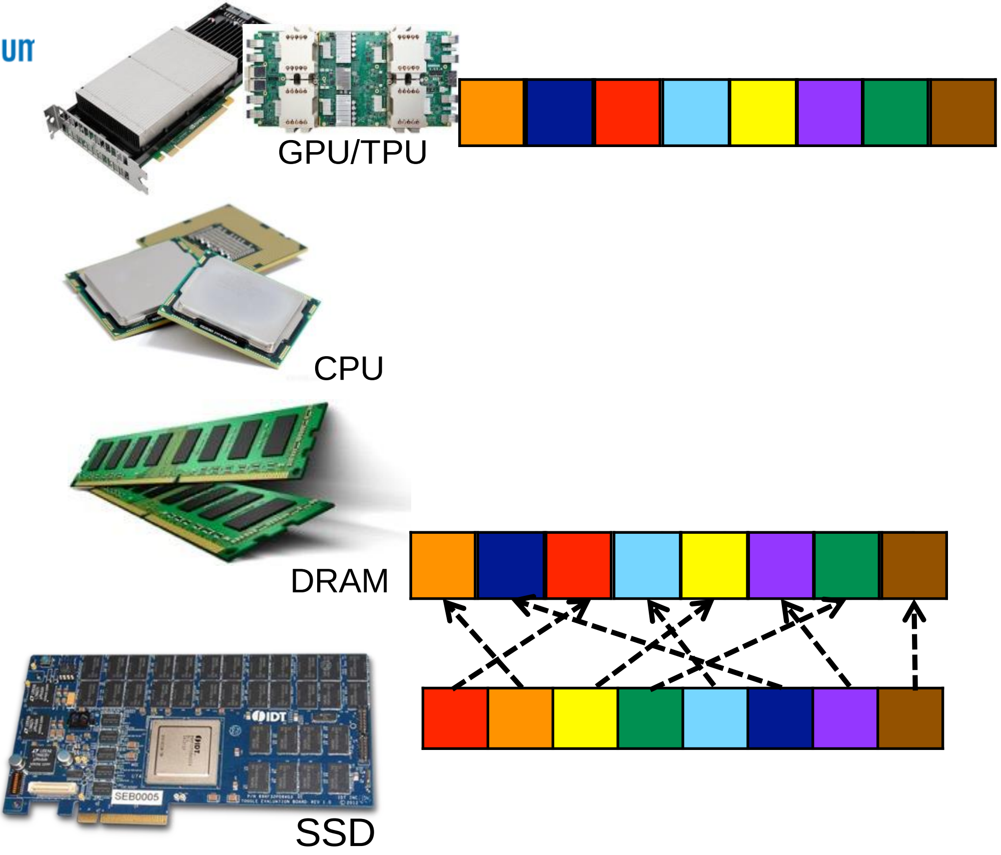
Conventional Data Shuffling



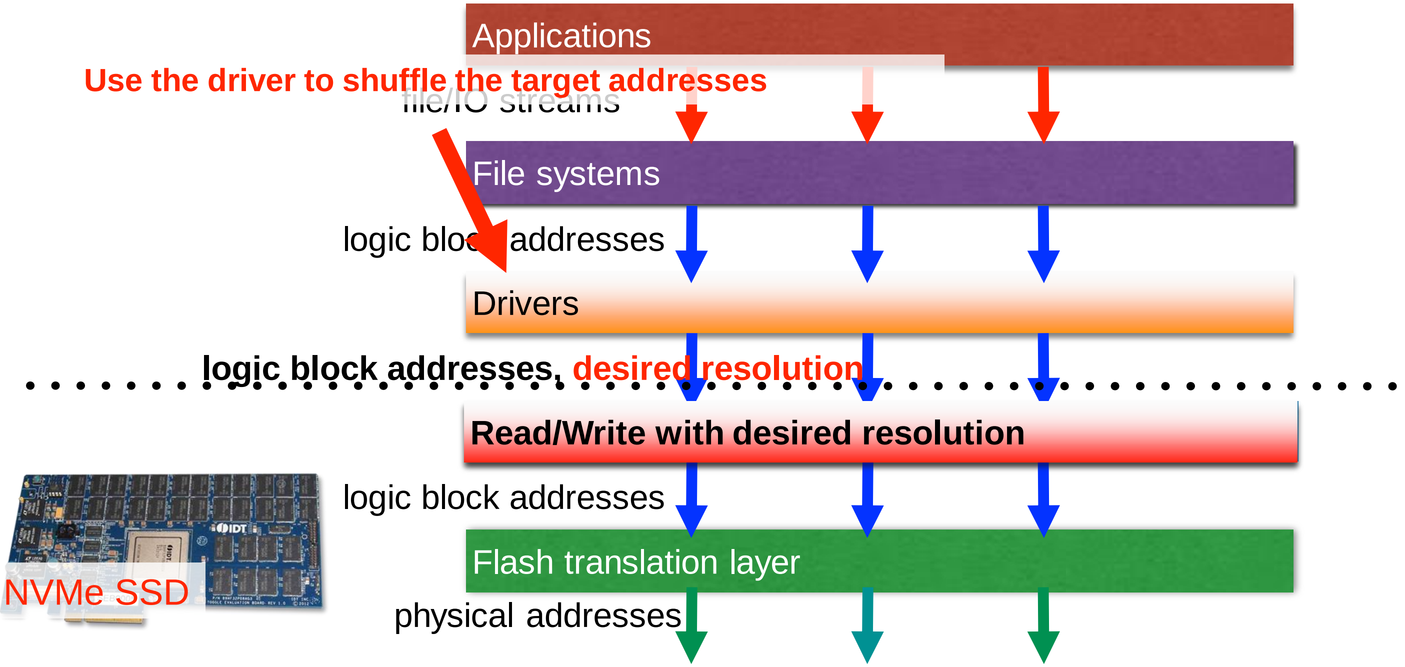
Conventional NVMe Read

- The command sends the starting address in the SSD and the length to read
- The command contains a list of memory locations to receive the reading data
    - These addresses are consecutive in virtual address presented to the application
    - These addresses may not be physically consecutive

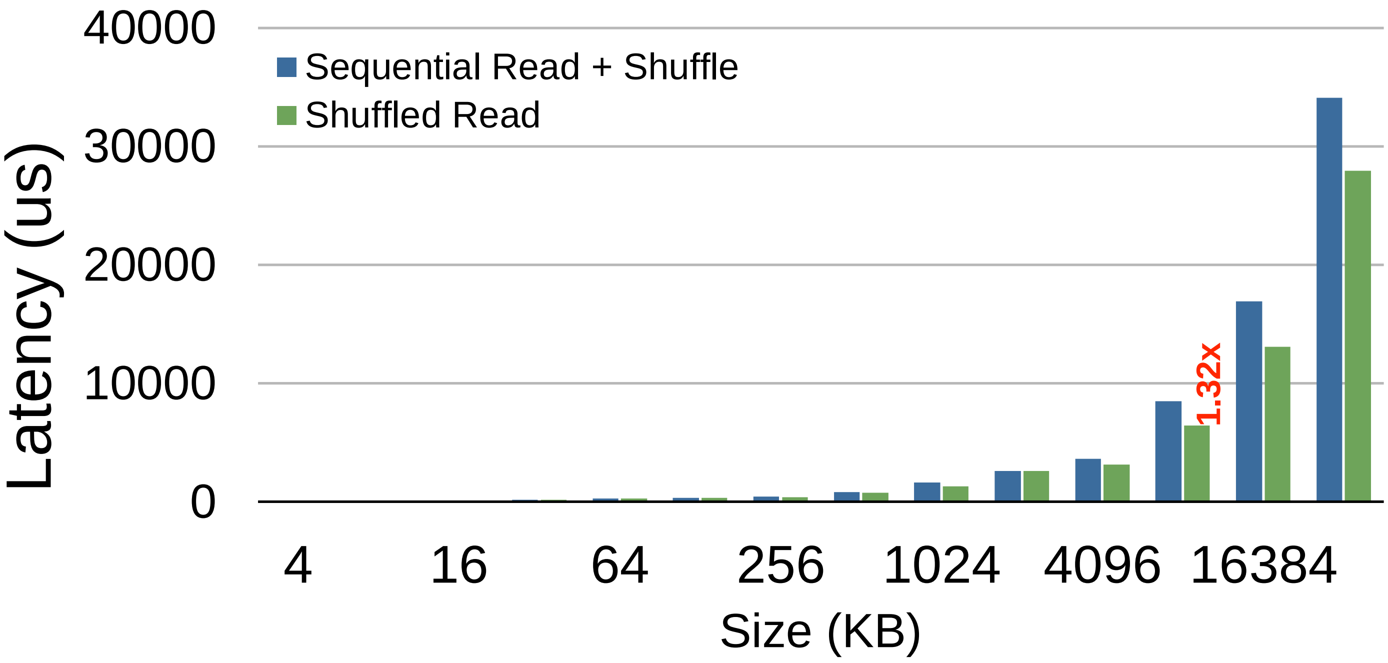
Our Shuffled NVMe Read



Shuffled NVMe Read



Performance of shuffled NVMe read



Flash Memory Summit

Conclusion

- Conventional research focus on single-point design, missing the opportunities for cross-layer, full stack solutions
- I/O stack is becoming the new bottleneck for accelerator-based architectures
- We need to carefully examine the bottleneck in modern applications — they may not be computation-bound

Flash Memory Summit

ESCAL

Extreme Storage & Computer Architecture Lab

UNIVERSITY OF CALIFORNIA

UCRIVERSIDE

https://www.escalab.org/