GL
GOKE US LAB

Developing Low Latency NVMe Systems for Hyperscale Data Centers

Prepared by Engling Yeo
Santa Clara, CA 95054
Date: 08/04/2017

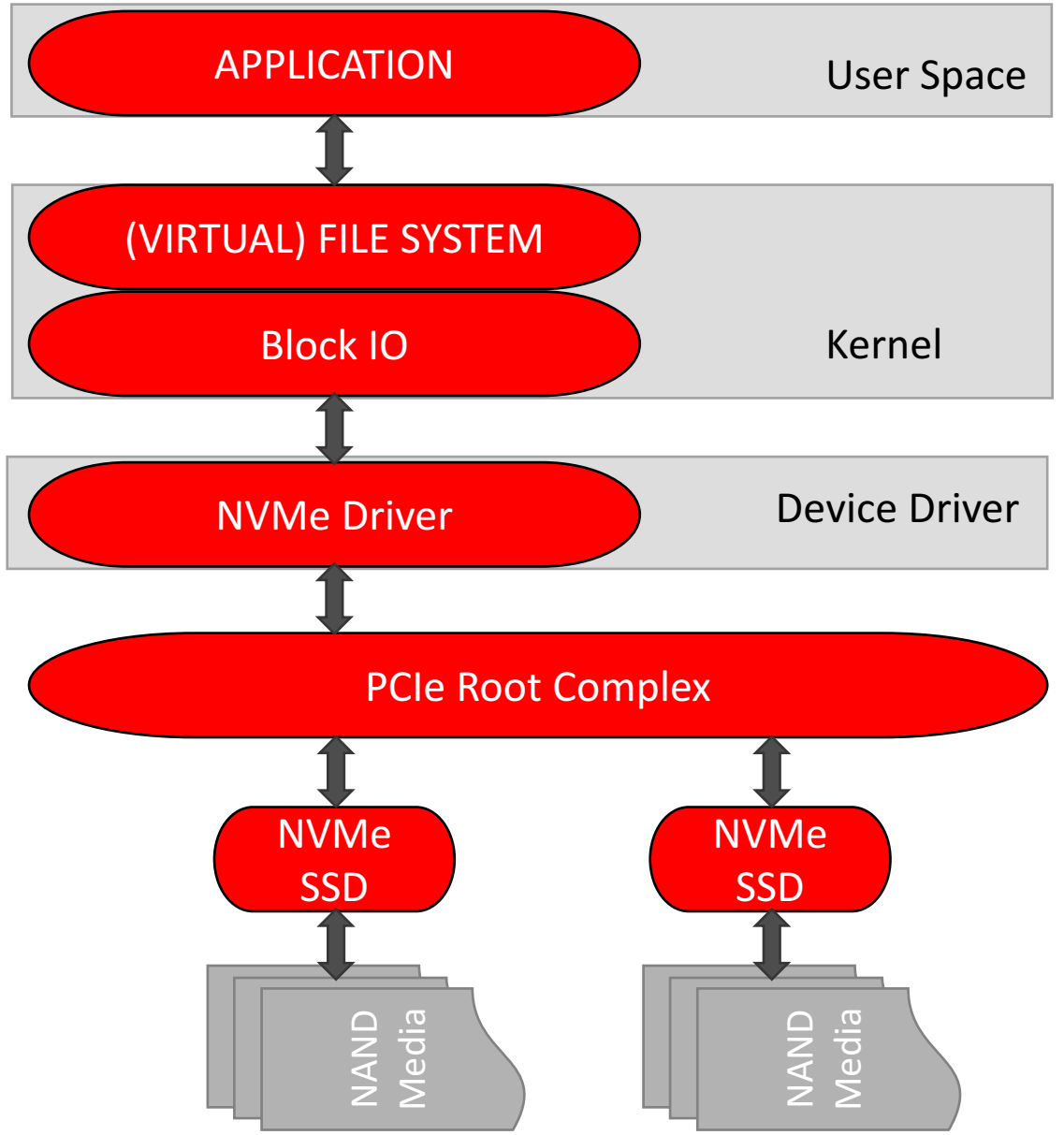
IOPS, Throughput, Latency

- Short predictable read latencies
- Tape Storage. IBM engineers achieve 201 GB/in^2

  200PB fits easily into a truck 5ft x 15ft stacked 100 thick

  - Drive 3 hours to San Francisco
  - Throughput : 18TB/s, 4.5G-IOPS/s

Limit the maximum latency



Hyperscale Storage Directions

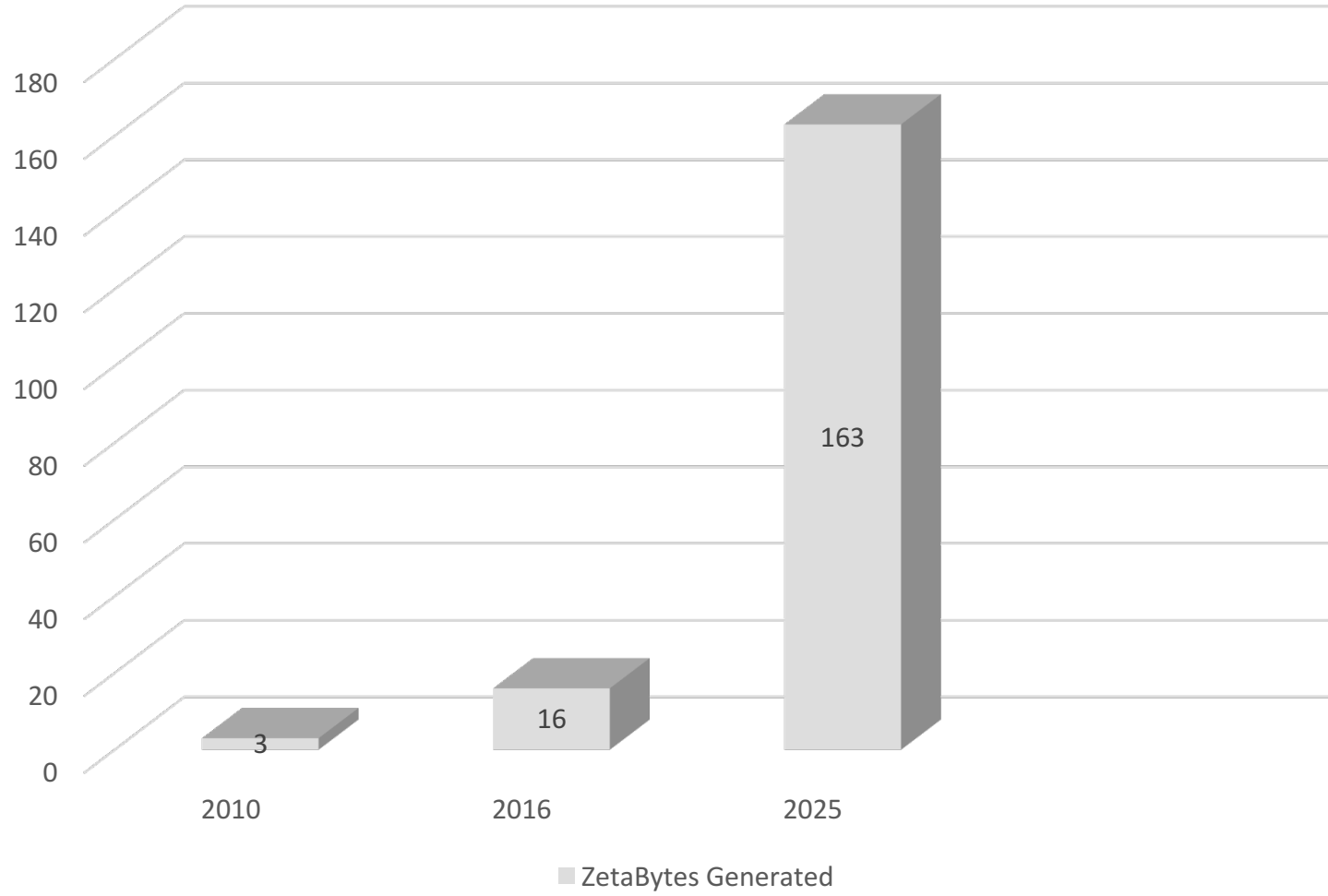
Worldwide data generated

2010 : 70% storage on Mobile/PC

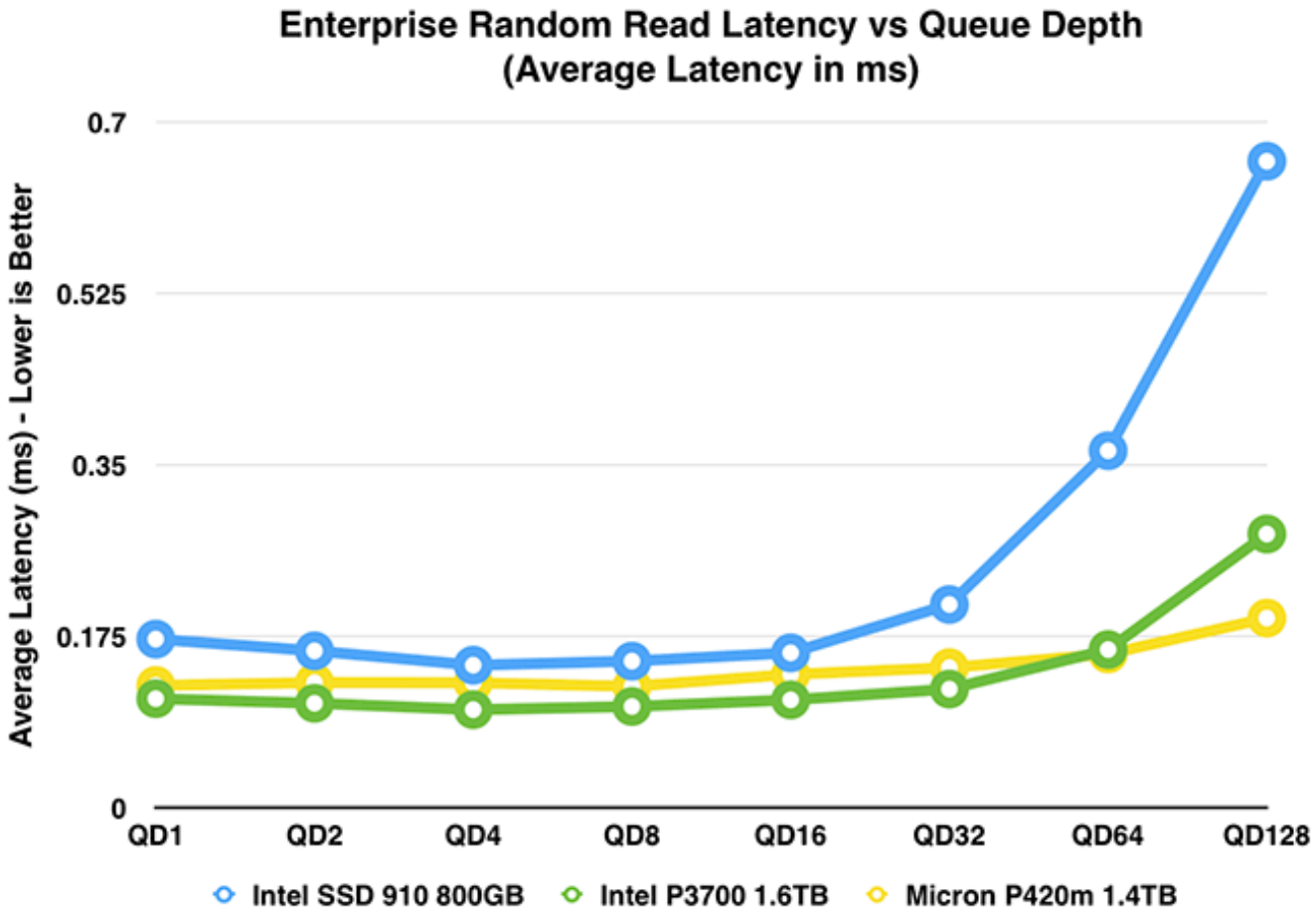
2025 : 50% storage on HyperScale

40% data mining, machine learning, IOT

- Factors Affecting Growth
  - Cost / Capacity
    - Mean time between failures
    - Power concerns
  - Security
    - Configurability. Key management
    - Firmware update
    - Santization and Life Cycles
  - Control over stack. Build vs buy infrastructure
  - Performance



Latency Benchmarks of Several Enterprise PCIe SSDs



Courtesy Anandtech June 2014

Typical Read Latency of a NVMe System

GOKE US LAB

- Typical read latencies for a 4kB Read Access
  - Controller

• PCIe and NVMe frontend HW	1μs
• Firmware interpretation of NVMe command	2μs
• FTL-cache miss; DDR Access	3μs
• T_{read} (TLC)	**100μs**
• Transfer 4kB @ 800MBps	6μs
• ECC Decoding	6μs
• Gen3x4 PCIe transfer and NVMe completion	4μs
Total	~122μs

Compare this latency with DDR4 , e.g. 200ns

Hardware Challenges

Percentage latency attributable to media :

- SLC : 50%
- MLC : 70%
- TLC : 80%

Amdahl's Law

What can the controller design do?

Instead of optimizing best case latencies

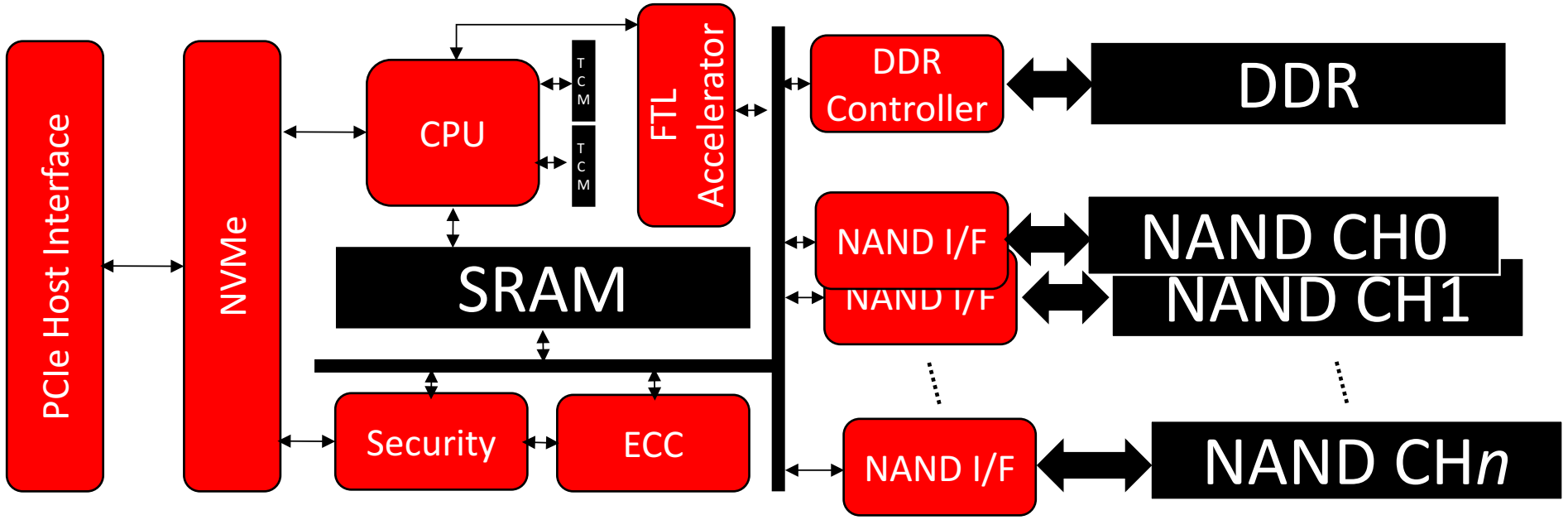
Focus on reducing maximum latency

Garbage collection
Data Cache
User Data

- Configurable FTLs to adapt dynamically to work loads
- Hybrid HW-SW implementation of FTLs
- Trade off dramatic changes in latency with more frequent context switches

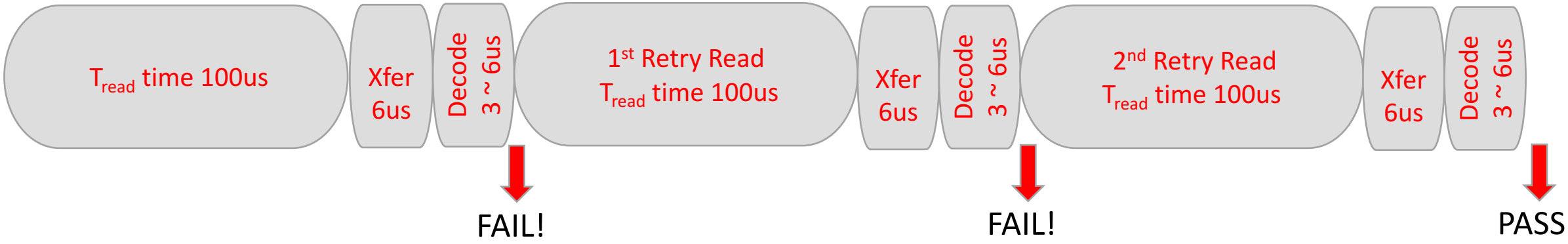
Low Latency NVMe Controller Hardware

- Configurable memories to support the hybrid FTL
- Rapid context switching
- Speculative processing
- Flexibility to issue and maintain control over massively parallel Channels/CEs/LUNs/Planes



GOKE US LAB

LDPC has higher decoding latencies… (not exactly)



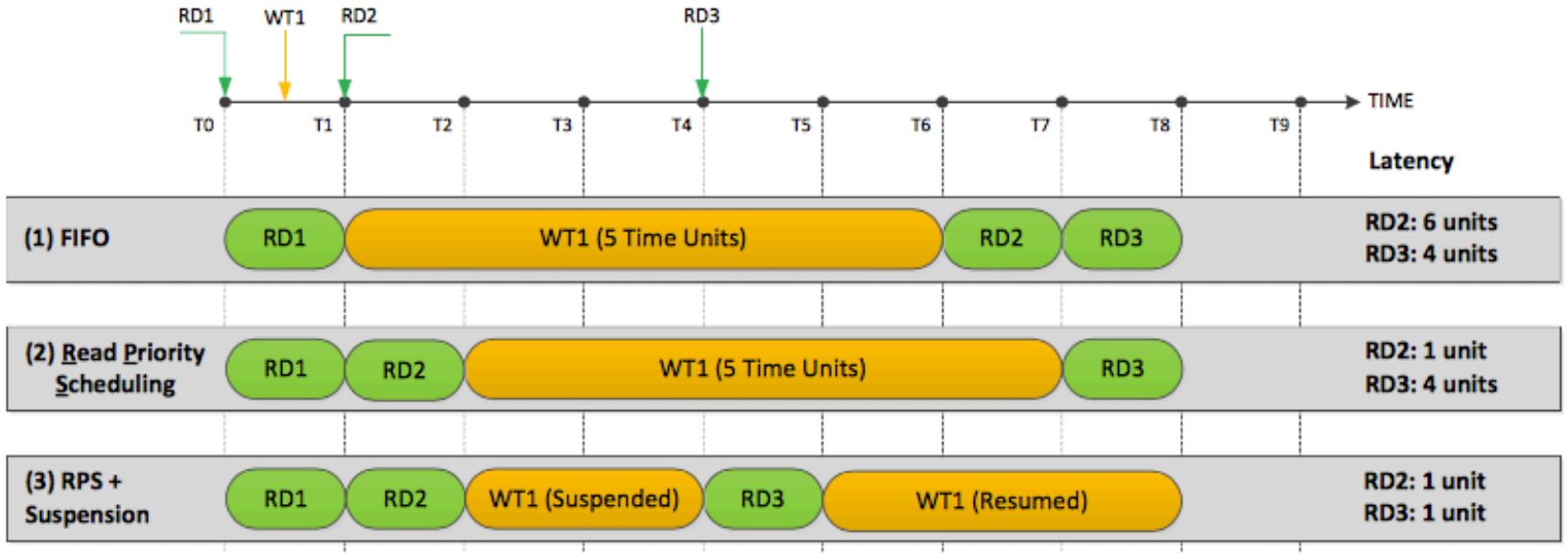
- Read retries are typically >100μs penalty
- Soft-LDPC decoding also requires read retries
- Take advantage of orthogonal Channels / CEs / LUNs/ Planes

Parallel Reads can recover the error frame in significantly reduced latencies

Flash Interface Controller

- Respect the well documented T_{read}, T_{prog}, T_{ber} times
- Poll less, transfer more
- Know when to suspend/abort more time consuming tasks
- Out of order execution

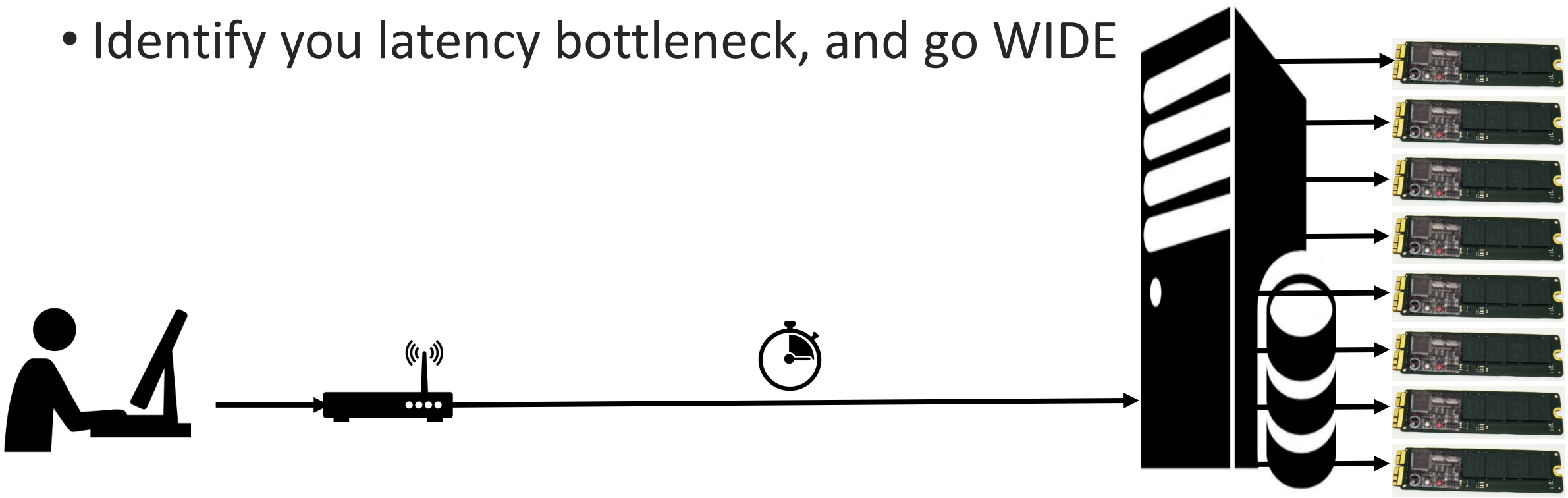
Stop asking. The data is NOT Ready!!



Courtesy Wu,
Virginia Commonwealth University

Latency is Key to QOS

- Always respect Amdahl's Law
- Context Switching
- Control your maximum latency
- Identify you latency bottleneck, and go WIDE



THANK YOU

GOKE US RESEARCH LABORATORY
4655 Old Ironsides Dr, #350
Santa Clara, CA 95054

WWW.GOKEUSLAB.COM

Abstract

- Hyperscale data centers need extremely low latency storage systems to provide predictable high performance over a wide variety of applications at reasonable cost. To be commercially viable, they need a multi-tiered memory system consisting of DRAM for high speed, low-latency non-volatile memory (such as 3D XPoint) for larger amounts of key data, and the more traditional non-volatile NAND flash for mass storage. The realization of such systems involve hardware, software, and driver challenges. The result must be fully scalable, low-power, and capable of handling the most challenging big data applications.