SDC 18
SNIA EMEA
FEBRUARY 2018
TEL AVIV, ISRAEL
STORAGE DEVELOPER CONFERENCE

# ZUFS

## How to Develop an Efficient PM-based File System in User Space

Dr. Golander Amit, NetApp

# Storage Media Generations

IOPS
(even if random…)

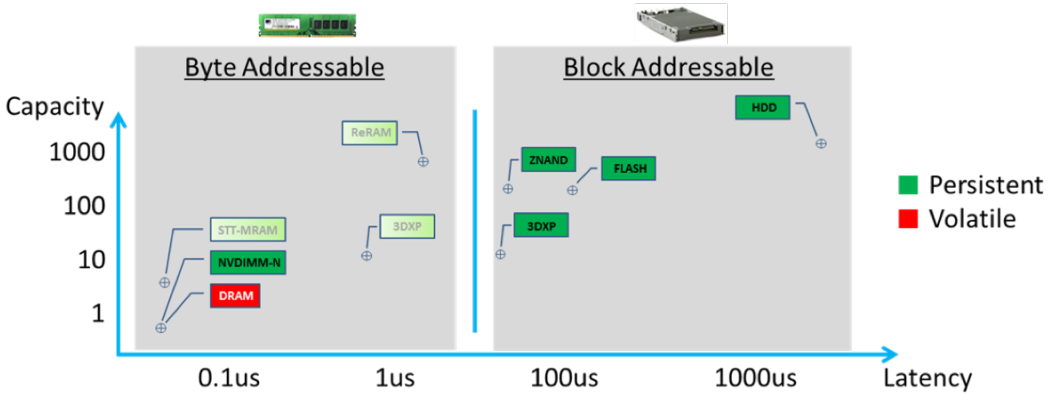
Latency
(even under load…)


**HDD** **FLASH** **NVDIMM / PM**

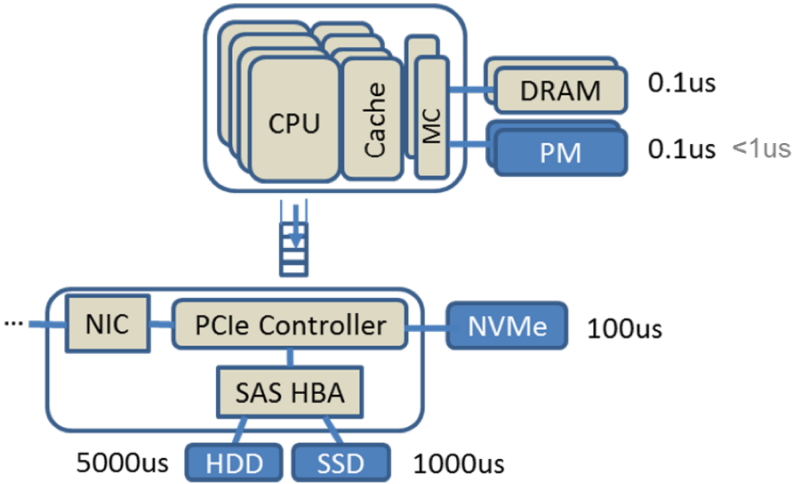
PM marries the best of both worlds: **Memory Speed** + **Storage Persistency**

# PM Hardware

**SCM** (Storage Class Memory)
Byte-addressable **Media**
@ Near-memory speed

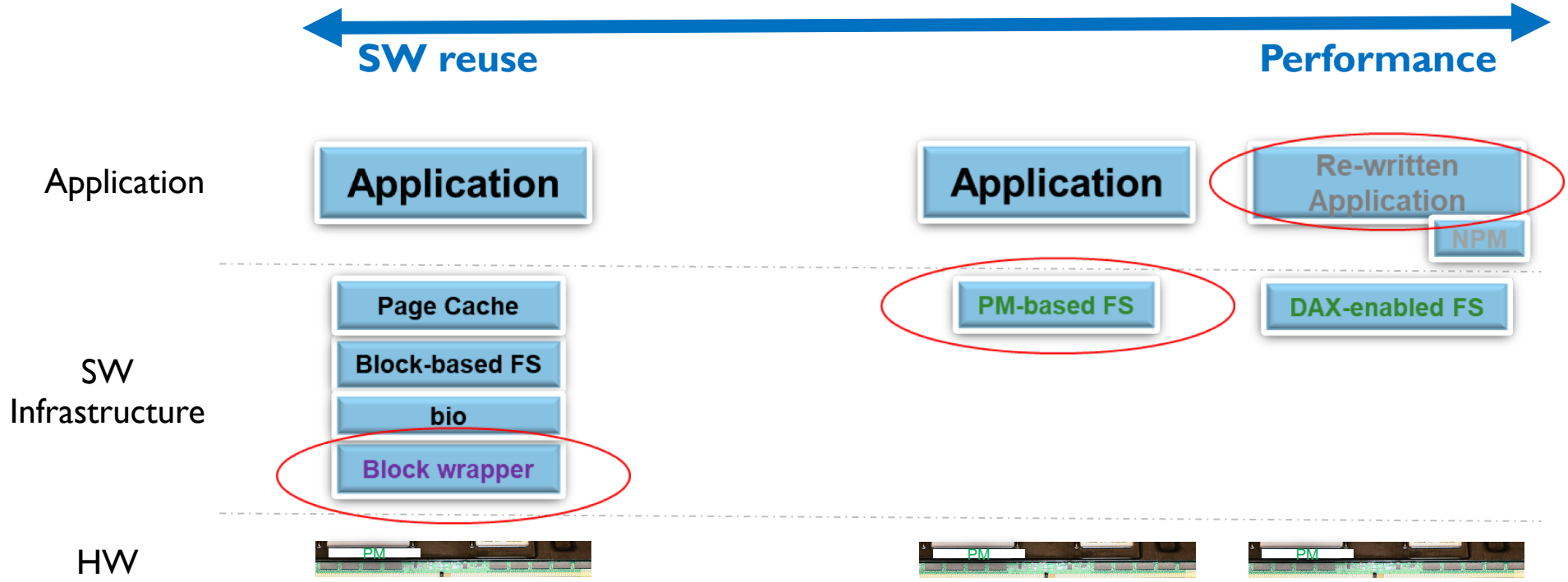



**PM** (Persistent Memory)
Byte-addressable **Device**
@ Near-memory speed, on memory bus



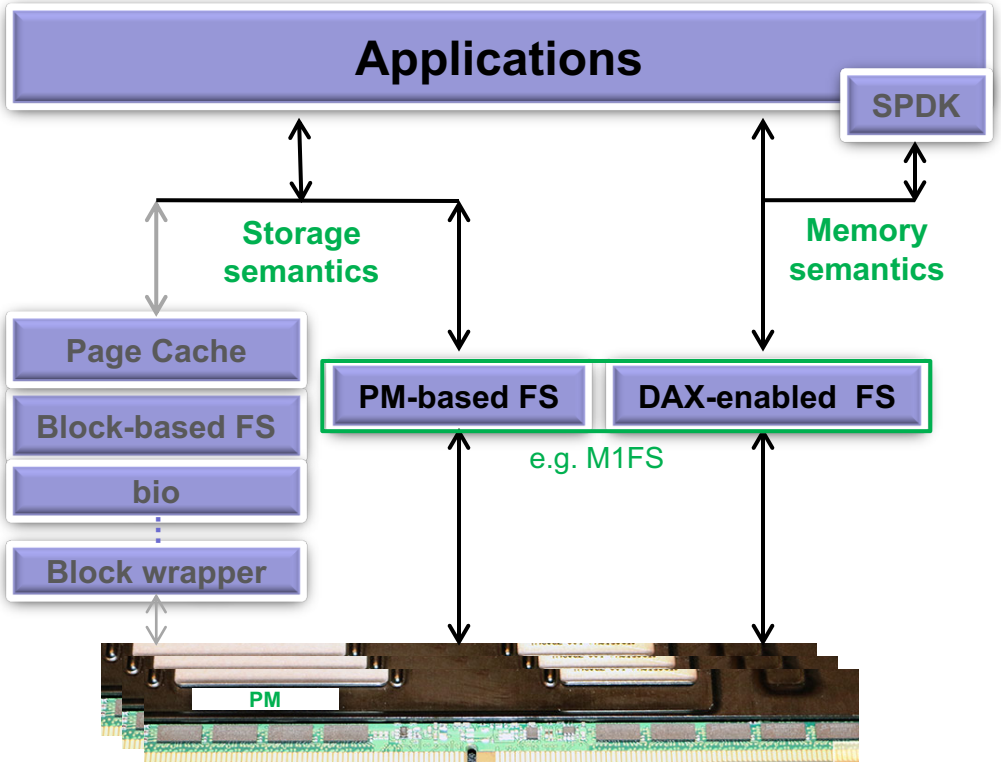

Rounded latency numbers & under typical load

# PM-based SW Approached



# Background

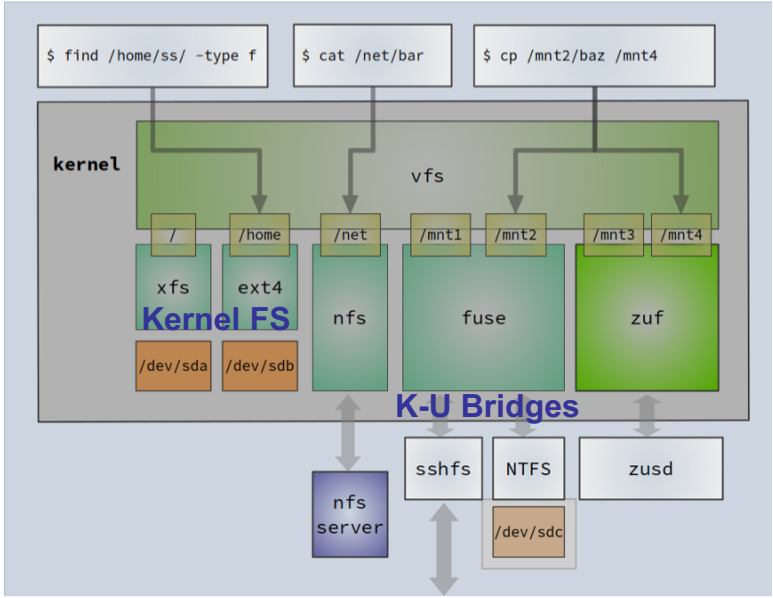
- Plexistor (acquired by NetApp)
  - PM-based FS pioneer since 2013
  - Contributing some of our IP
- Our PM-based FS approach:
  - Support legacy applications & Enable NPM (e.g. SPDK)
  - Feature rich
  - Integrate with NetApp product portfolio



# Kernel Vs. User Space FS Implementation



| Kernel | User space |
|---|---|
| Fast (shortest path) | Portable |
| | Resilient (contained) |
| | Simpler to add functionality & Debug |
| | Fewer licensing restrictions |

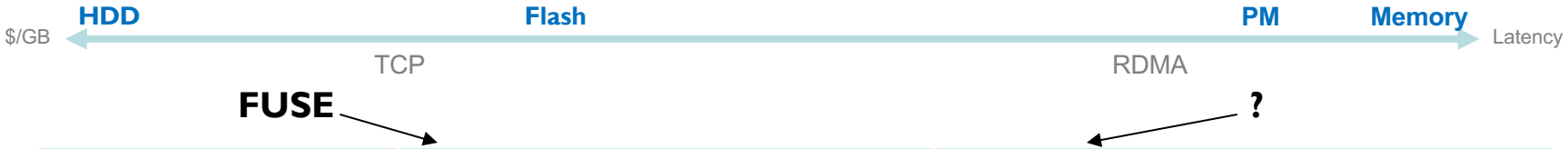



The gap:
Near-memory speed Kernel-to-User bridge

# Why not extend FUSE to PM?



FUSE architecture is great for HDDs and ok(ish) for SSDs, but not suitable for PM

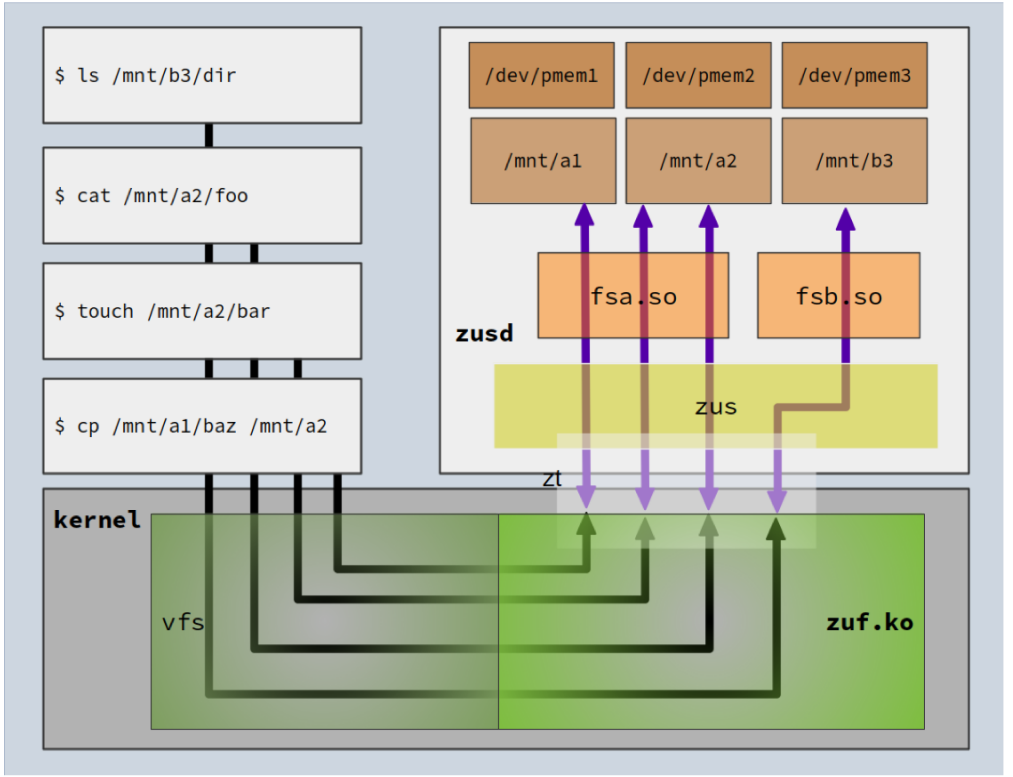



Design Assumptions

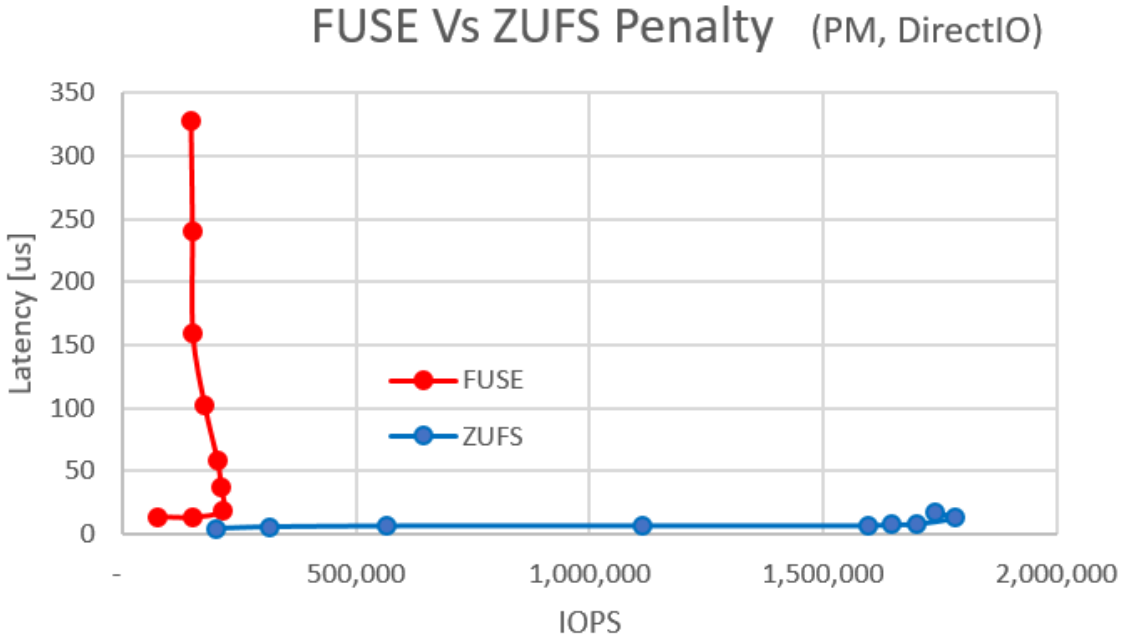
| | FUSE | ZUFS |
|---|---|---|
| Typical medias | Built for HDDs & extended to Flash | Built for PM/NVDIMMs and DRAM |
| SW Perf. goals | • Secondary (High latency media)<br>• Async I/O Throughput | • SW is the bottleneck<br>• Latency is everything |
| SW caching | Slow media -><br>Rely on OS Page Cache | Near-memory speed media -><br>Bypass OS Page Cache |
| Access method | I/O only | I/O and mmap (DAX) |
| Cost of redundant copy / context switch | Negligible | The bottleneck -><br>Avoid copies, queues & remain on core |
| Latency penalty under load | 100s of µs | 3-4 µs |

# ZUFS Features & Architecture

- Low latency & Efficient
  - Core & L1 cache affinity
  - Zero data copy
- Manages devices
  - Optimal pmem access
  - NUMA aware
  - Data mover to lower tier devices
- Page table mapping supports I/O & DAX semantics
- Misc
  - Async hook available
  - System service



# Preliminary Results (for PM)




- Measured on
  - Dual socket, XEON 2650v4 (48HT)
  - DRAM-backed PMEM type
- Random 4KB DirectIO write access

# Summary

- ZUFS is a Kernel-to-User bridge designed for PM
  - Enables NetApp solutions
- Open Source, being contributed upstream
  - Hope to accelerate PM adoption and innovation
  - https://github.com/NetApp/zufs-zus & zufs-zuf
  - You're welcome to use, review and contribute code

# Thank you