The Case for NFS-eSSDs

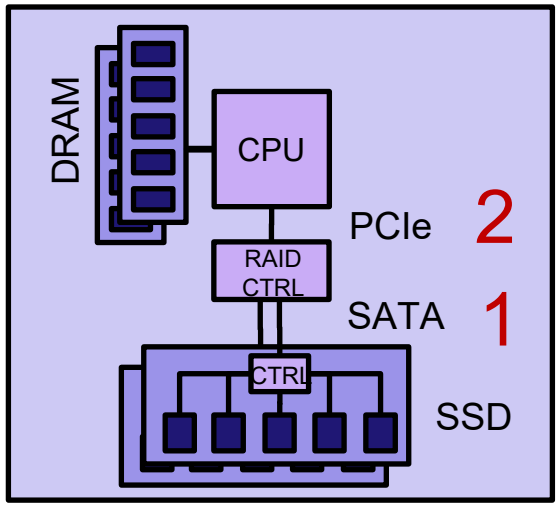
MSST, May 2023

David Flynn

Founder and CEO of Hammerspace,
Founder and CEO of Fusion-io

HAMMERSPACE

Direct Attached Storage

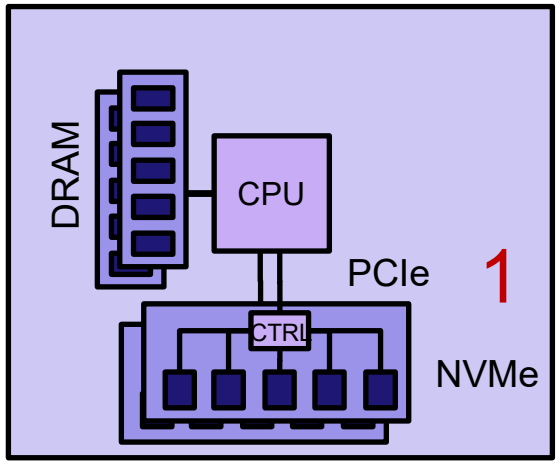


The RAID controller is the bottleneck and adds an additional serial data retransmission.

Direct Attached Storage – Using NVMe

NVMe only fixes direct attached storage.

What about shared storage?

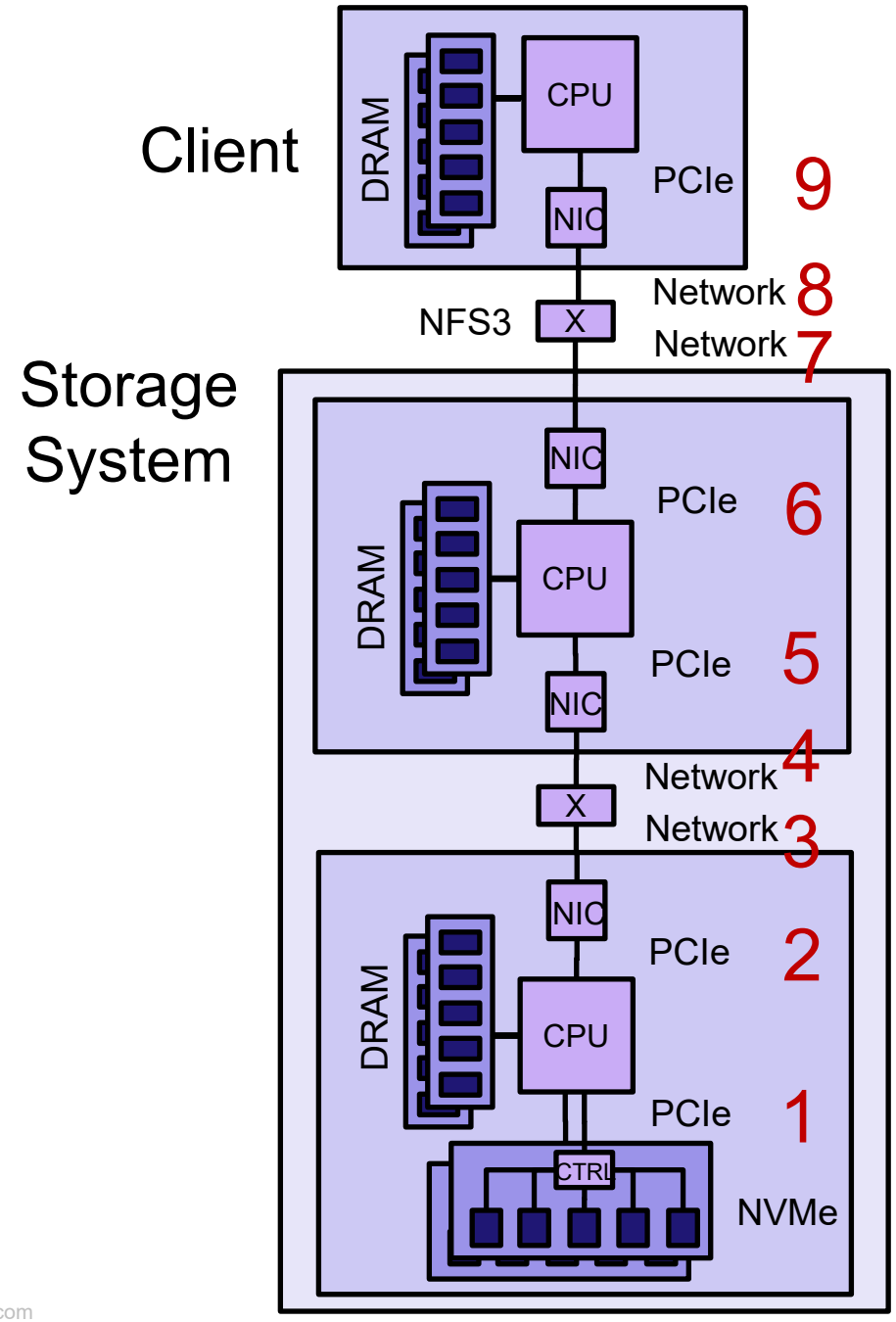


- DENTRY to INODE mapping
- File offset to block mapping
- Block to flash address mapping

HAMMERSPACE

Network Attached Storage

(e.g. NetApp, Isilon, Pure, Qumulo, Ceph)



DENTRY to INODE mapping

File offset to block mapping

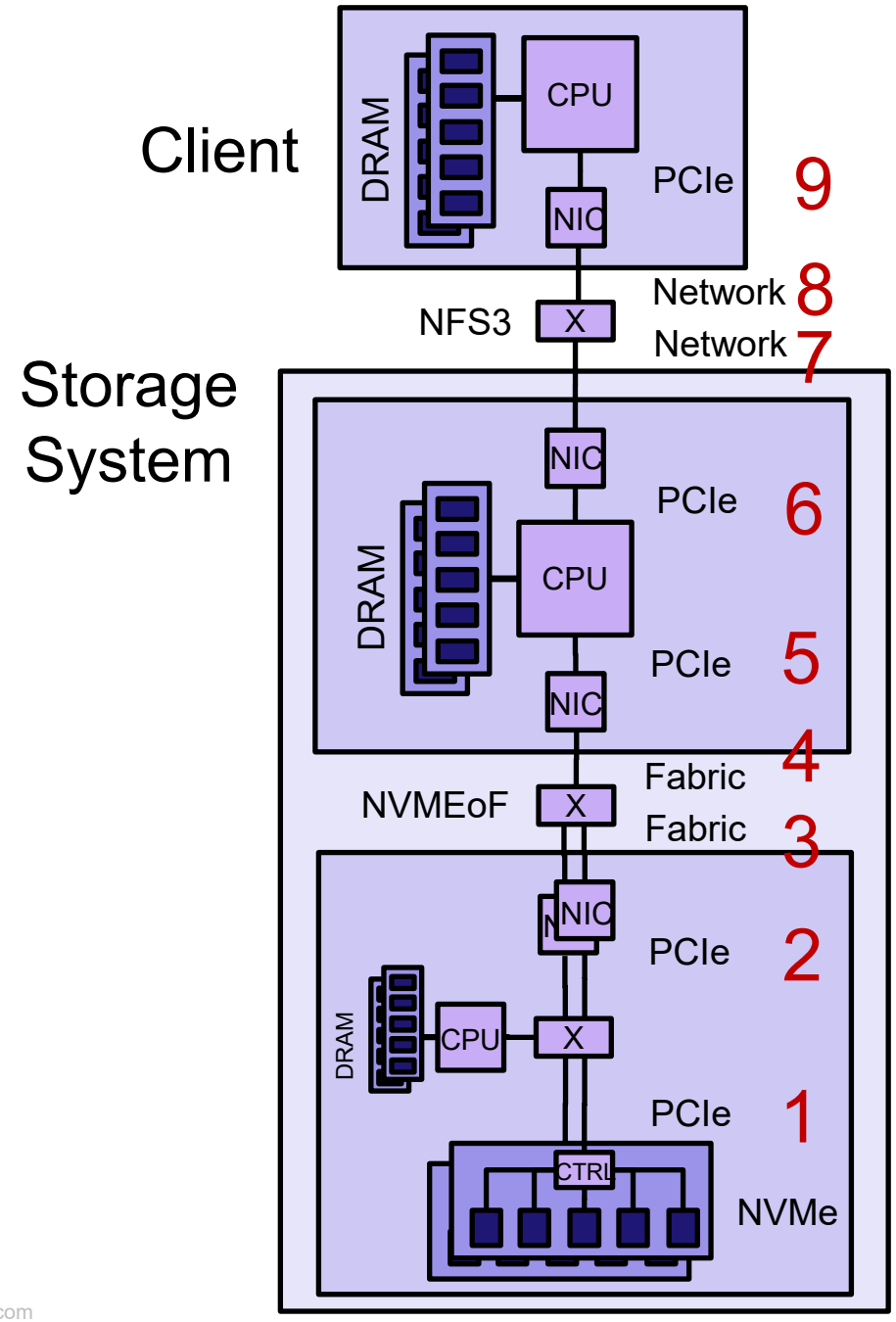
The storage back-end (CPU and NICs) are the first bottlenecks.

Block to flash address mapping

HAMMERSPACE

Network Attached Storage

(using NVMEoF, e.g. Vast, Weka)



DENTRY to INODE mapping

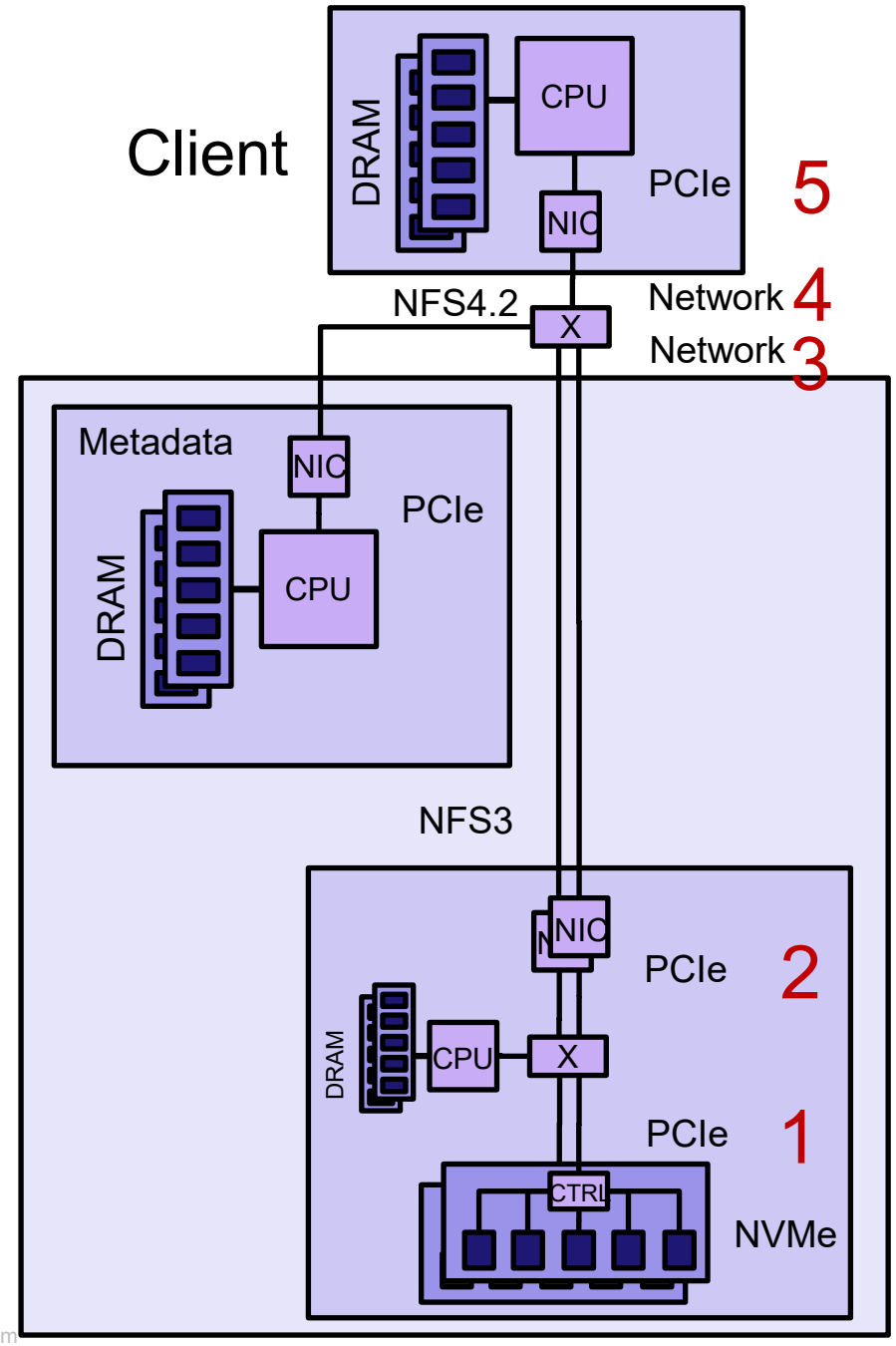
File offset to block mapping

Block to flash address mapping

The file server front end is an even bigger bottleneck.

HAMMERSPACE

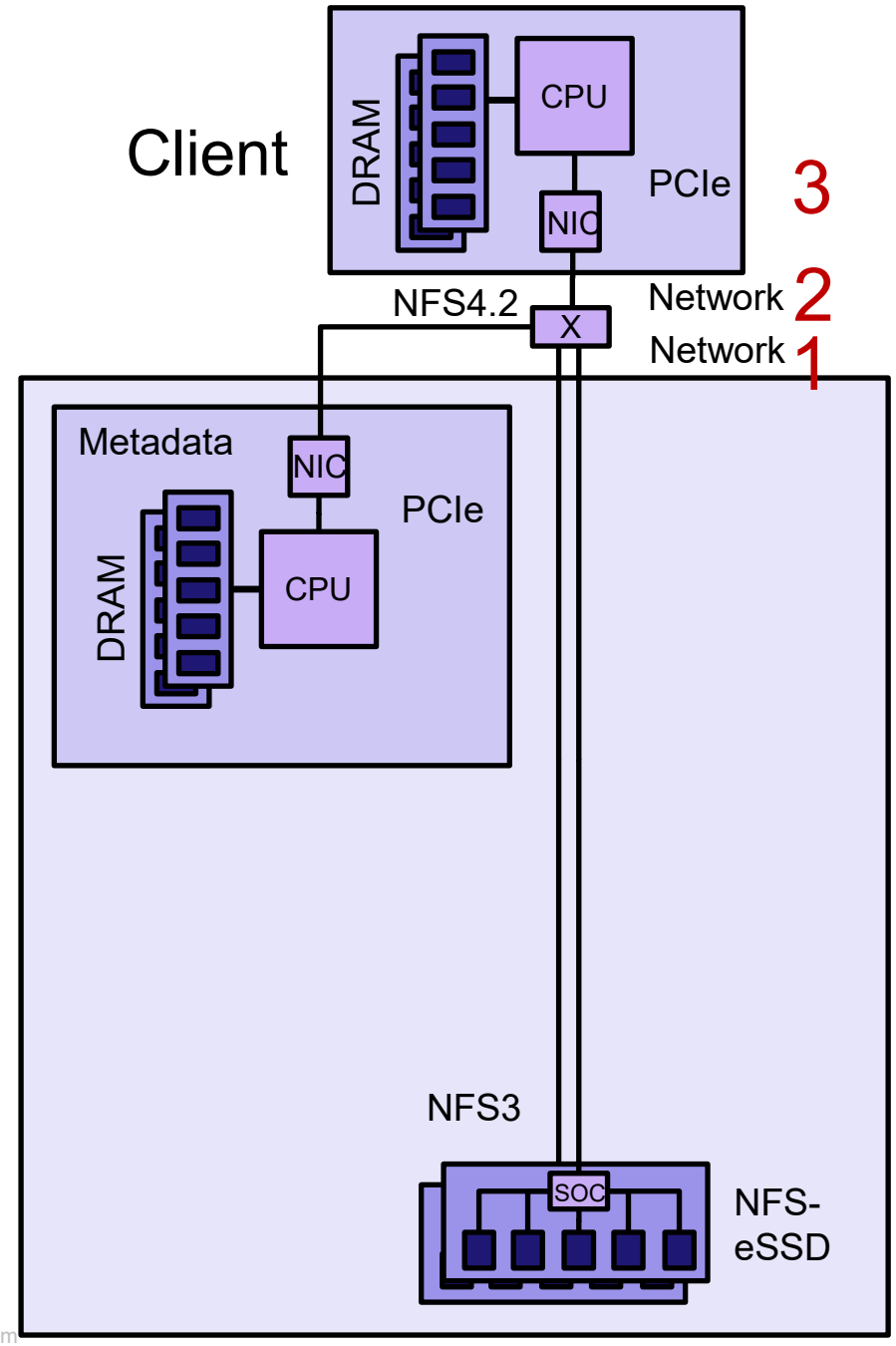
Network Attached Storage
(using NFS4.2, e.g. Hammerspace)



NFS4.2 has no bottlenecks, eliminates 4 of 9 data retransmissions, and doesn't need NVMEoF – or even an internal network!

HAMMERSPACE

Network Attached Storage
(using NFS4.2 and NFS-eSSD)



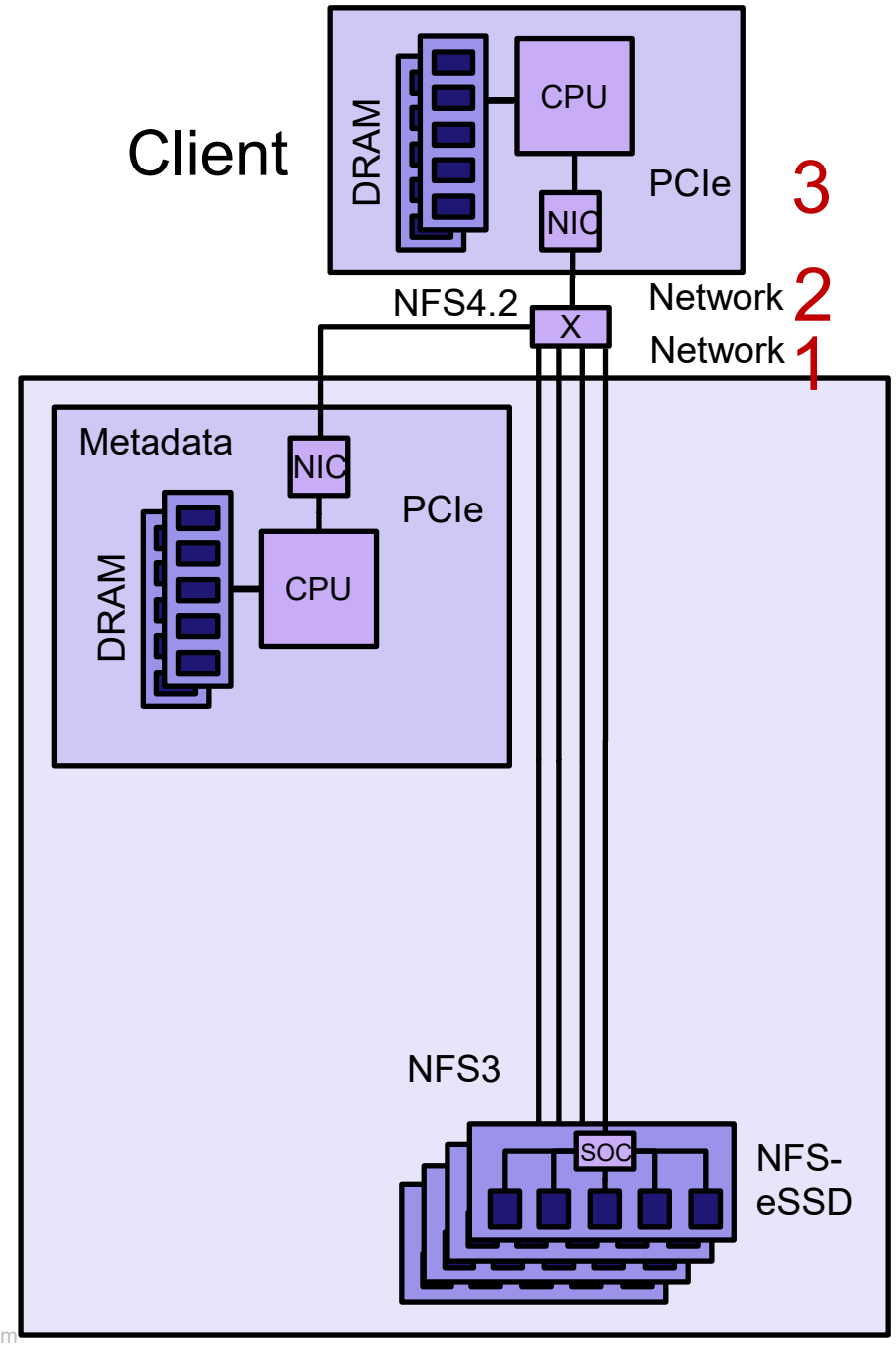
DENTRY to INODE mapping

File offset to block mapping

Block to flash address mapping

HAMMERSPACE

Network Attached Storage
(using NFS4.2 and NFS-eSSD)

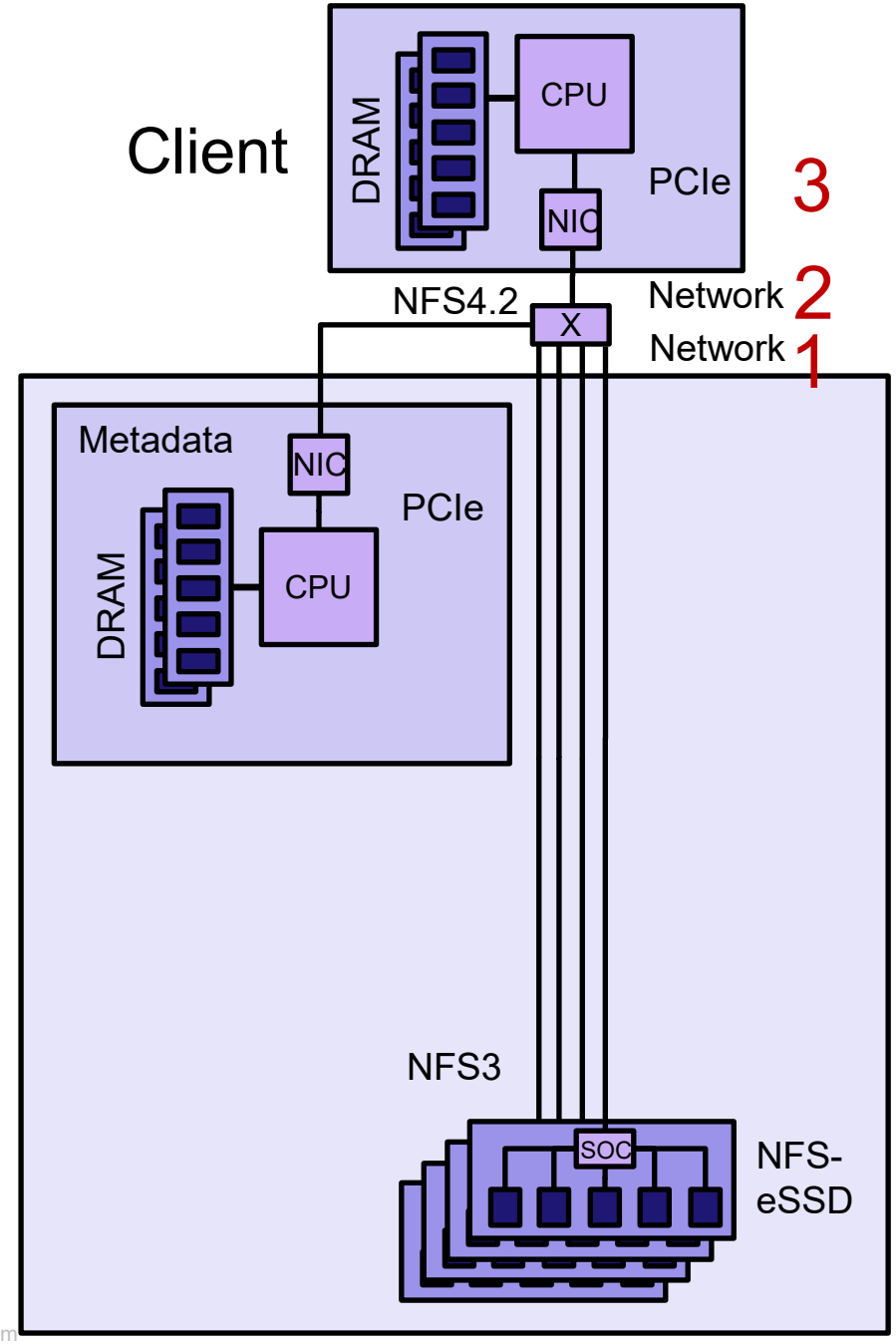


DENTRY to INODE mapping

NFS4.2 with proposed NFS-eSSDs eliminates 6 of 9 data retransmissions, eliminates the double mapping layers, and scales 1x1 with network ports!

File offset to flash address mapping

HAMMERSPACE



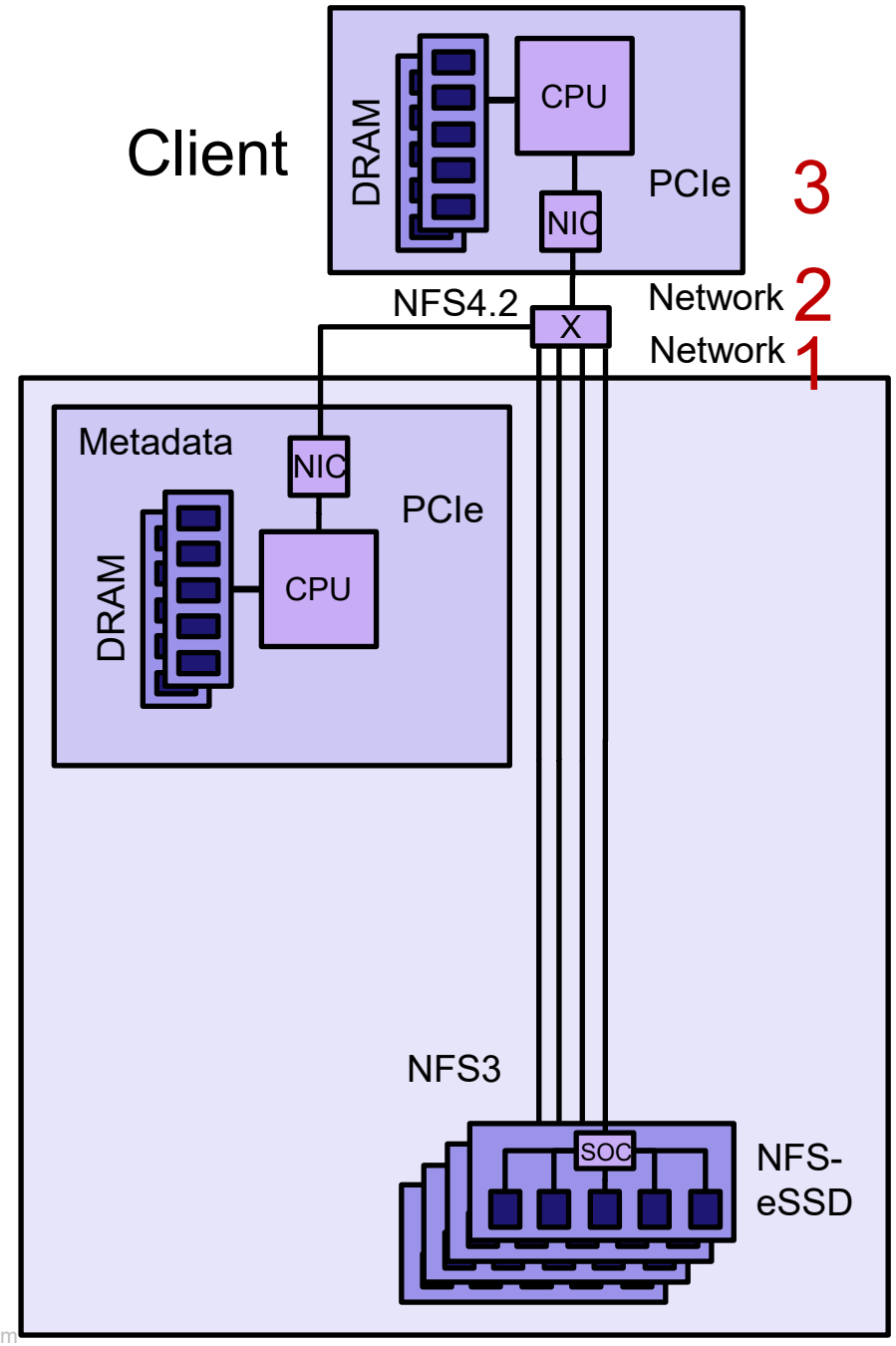
DENTRY to INODE mapping

File offset to flash address mapping

Benefits

- Lower latency
- Lower power consumption
- Lower operational (and capital) costs
- Lower write amplification
- Higher density without sacrifice of potential performance
- Higher access density
- Better inherent reliability, availability and serviceability
- Much wider dynamic range of scale
  - Scale up (hyperscale)
  - Scale down (SOHO, maybe on USB-C)
- Enables Computational Storage
  - Compression, deduplication, encryption, erasure / error coding, copy / clone, filter, search, join, map reduce, etc. can be offloaded to the SSD now that it understands file layout

HAMMERSPACE



DENTRY to INODE mapping

Why Now

AI/ML workloads demanding efficient performance

Data governance / cloud computing needs orchestration

Flash performance can easily saturate PCIe/Ethernet

E1.S and other form factors (density and power)

64-bit processor IP availability

Processor performance density

IPv6, RoCE

Embedded Linux with

High performance, lightweight filesystems (XFS)

High performance, lightweight NFS server (kNFSd)

Standardized Parallel NFS 4.2 Flexible Files

File offset to flash address mapping

Thank you

david.flynn@hammerspace.com

HAMMERSPACE