Ustore Lightning Talk

Storage Scale User Group Meeting @ SC23
Denver, CO – Nov 12, 2023

Presented by John Lewars (IBM)

Storage Scale

Disclaimer

Storage Scale

Ustore

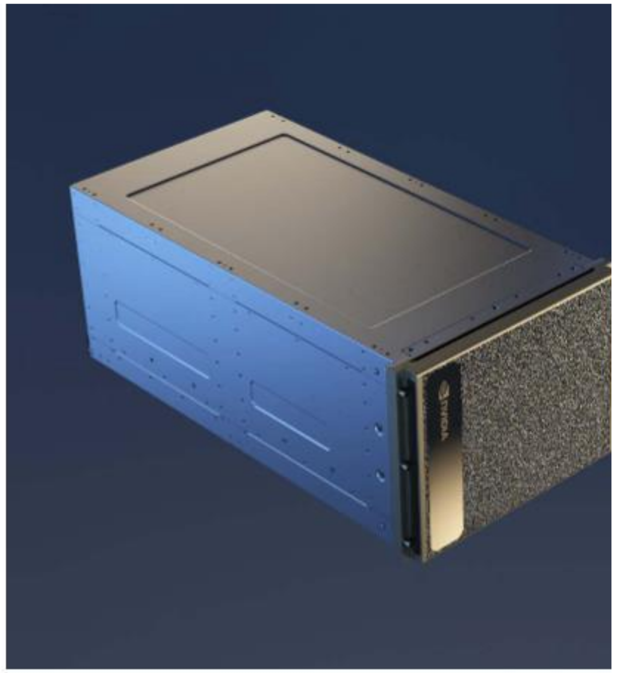
Accelerate AI and Analytics by allowing data to be stored in two pools at once, facilitating both a GNR reliable copy AND (i) a copy of data as close to the compute as possible or (ii) a copy of the data that leverages the small I/O performance benefits of NVMeoF.

Local storage can be shared/local NVMe, Persistent Memory or DRAM.

Ustore supports **Asymmetric Replication** with one erasure-encoded copy of the data and one performance copy for very fast access.

Ustore creates a **Shared Co-operative Cache** across all compute nodes. Any node can access all cached data, regardless of physical location.

NVIDIA DGX SuperPOD with IBM ESS



Specifications	
GPU	8x NVIDIA H100 Tensor Core GPUs
GPU memory	640GB total
Performance	32 petaFLOPS FP8
NVIDIA® NVSwitch™	4x

Ustore Value Proposition

Local storage configurations (particularly NVMe drives in client nodes) have become more common and IBM Storage Scale provides multiple solutions that leverage local storage:

Existing solutions that use local storage to create a filesystem:

- SNC (Shared-nothing cluster)
- FPO (File Placement Optimizer: SNC with write-affinity)
- ECE (Erasure Code Edition)

Existing solution that uses local storage to cache data:

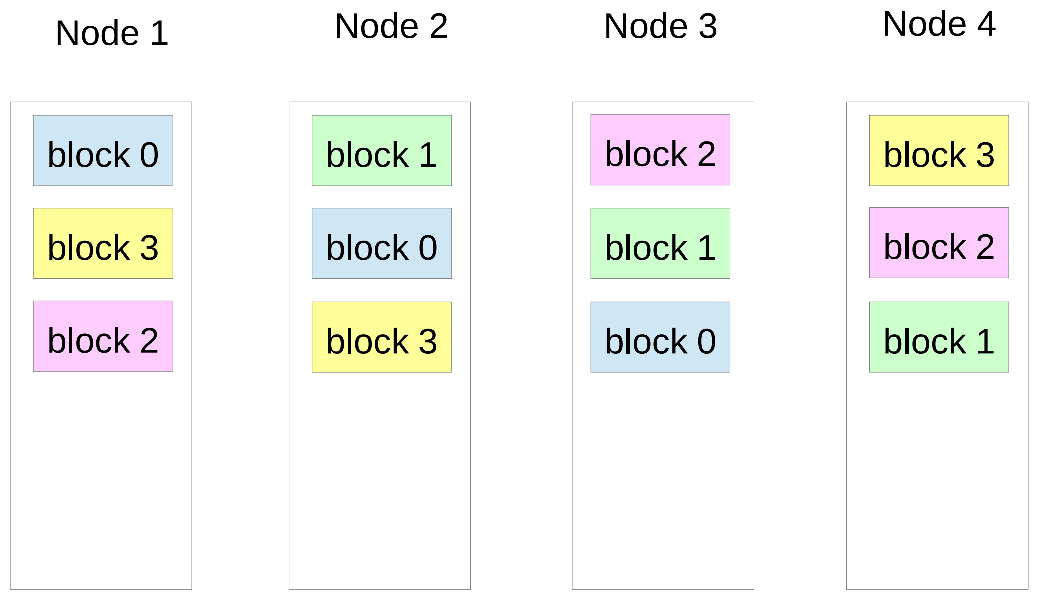
- LROC (Local Read-Only Cache)

Ustore combines key features of these solutions to create a filesystem and cache data in a persistent, non-redundant manner.

We are first targeting support for a shared storage solution in which drives from an IBM Storage Scale System are made available to clients via the NVMeoF protocol, and then we're targeting support for the use of clients' local storage in a follow-on release

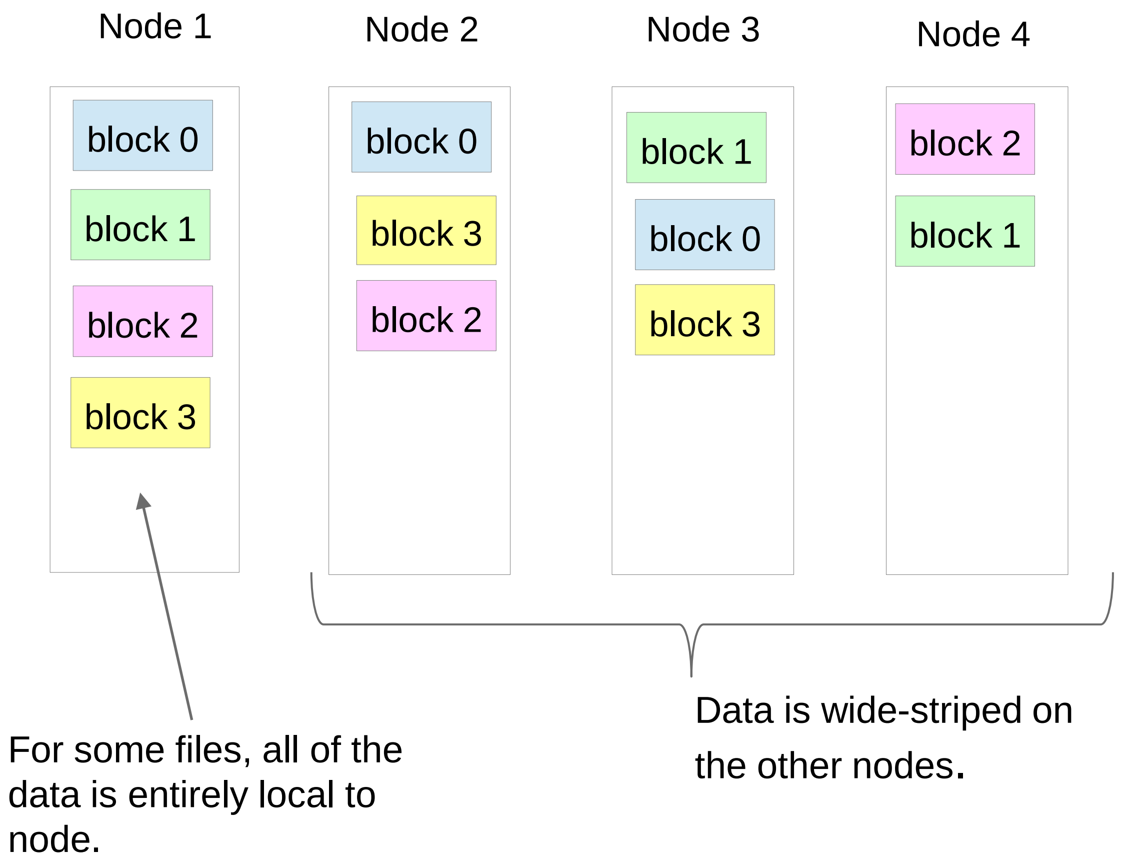
Workloads on local storage with SNC

- Data Access: Workloads that use data-aware schedulers (e.g., Hadoop) can take advantage of locally stored data.
- Data Reliability/Availability: 3-way replication. Metadata scan needed after a failed node rejoins the cluster.
- Cluster Resources: Each unit of user-consumable storage uses 3x units of storage for replication overhead.



Workloads on local storage with FPO

- Data access: Provides a method to affinitize data so a wider range of applications can exploit data locality; e.g., databases can ensure that one replica of a target fileset is entirely stored on a given node.
- Data reliability/availability: 3-way replication. Metadata scan needed after a failed node rejoins the cluster.
- Cluster resources: Each unit of user-consumable storage uses 3x units of storage for replication overhead.



Workloads with ECE storage-rich servers

Employs GNR declustered RAID software across local storage in the cluster. Each data block is split into N data strips and M parity strips, and these strips are spread across nodes.

– Data Access: Most data not local and there is no control over data placement (data chunks too small for schedulers like Hadoop to manage). A read typically requires gathering strips from other ECE servers (multiple network hops).

b0 D1	b0 D2	b0 D3	b0 P
b1 P	b1 D1	b1 D2	b1 D3
b2 D3	b2 P	b2 D1	b2 D2
b3 D2	b3 D3	b3 P	b3 D1

– Data Reliability/Availability: Uses erasure encoding and parity strips to provide data reliability.

– Cluster Resources: Each unit of user-consumable storage uses 1.25x of storage resources for parity overhead (assuming 8+2P).

Workloads on local storage with Ustore

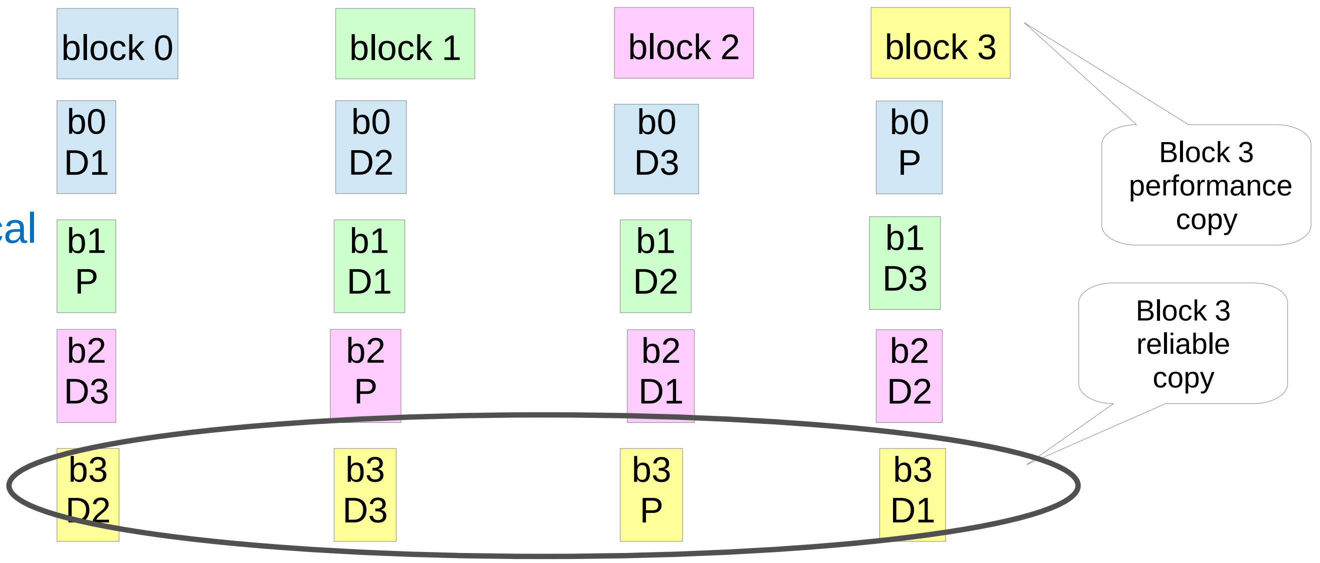
Maintain two copies of each data block in two different pools with "asymmetric replication".

- "reliable copy": protected by ECE erasure encoding
- "performance copy": stored entirely (as one unit) on a single node, providing fast reads for workloads that leverage data locality

The performance copy is a persistent, non-redundant cache.

Data Access: The performance copy of data is entirely local to a client (as with FPO); the reliable copy is erasure-encoded.

Data Reliability/Availability:
The reliable copy is erasure-encoded. The performance copy is maintained with "*lazy consistency*" such that missing updates due to a disk being down are fixed on the next read after the disk is back online.
(No metadata scan needed before bringing the disk back.)



Cluster resources: Each unit of user-consumable storage uses 2.25x of storage resources, which is higher than 1.25x ECE but lower than 3x FPO.

Workloads on local storage with LROC

LROC uses local storage (typically SSD/NVMe) to support caching.

Ustore has 2 advantages over LROC:

- LROC cache is potentially redundant because the same data can be cached on multiple nodes, but no more than a single copy of the data will be contained in the Ustore performance pool.

- LROC is not a persistent cache because a reload is needed after a failed node is back up. With Ustore, the performance copy is persistent except when a drive goes down (the reliable copy is used until the performance copy is available and any missing updates are applied).

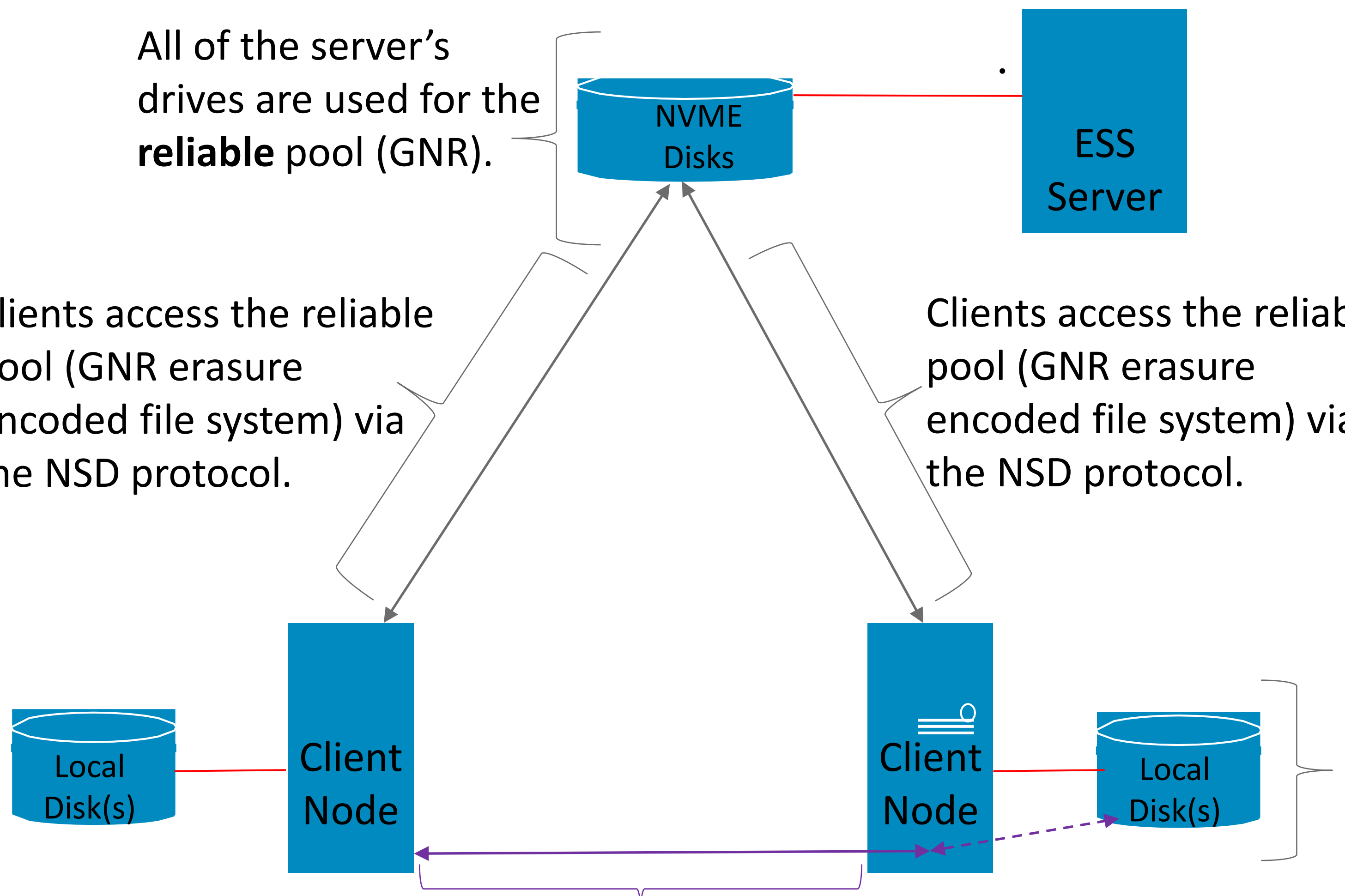
Ustore use case: Exploiting local storage

All of the server's drives are used for the **reliable** pool (GNR).

Clients access the reliable pool (GNR erasure encoded file system) via the NSD protocol.

Clients access the reliable pool (GNR erasure encoded file system) via the NSD protocol.

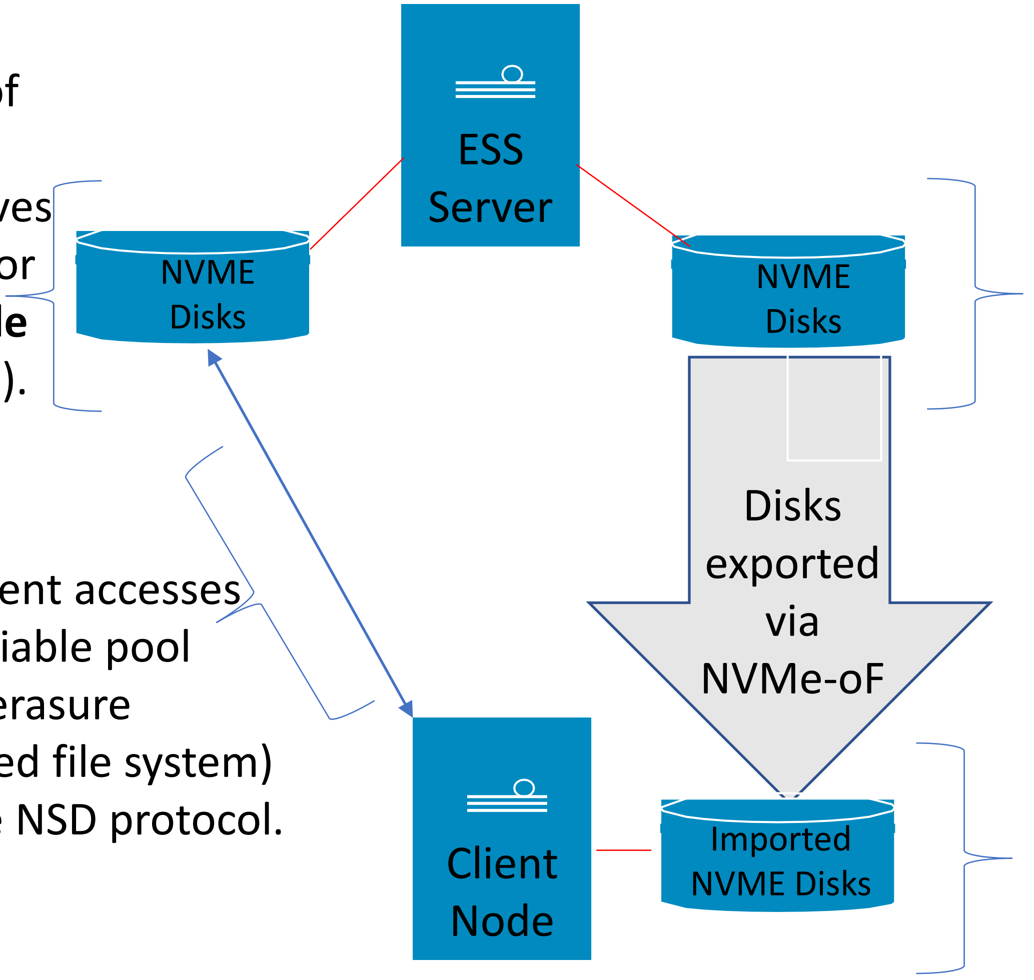
Storage physically present on the client (not exported via NVMeoF) comprises the performance pool. Clients access this local storage directly or can access local storage from other clients via the NSD protocol.



Example of client accessing another client's local storage via NSD protocol.

Ustore use case: Exploiting shared storage via NVMeoF

.



A subset of server's NVME drives are used for the **reliable** pool (GNR).

The complementary subset of the server's drives are used for the **performance** pool.

The client accesses the reliable pool (GNR erasure encoded file system) via the NSD protocol.

The Performance pool drives are exported to the client(s) via NVMeoF, appearing like local drives to the client (e.g /dev/nvmeXn1). A client accesses the drives via the local I/O path (not NSD protocol) and there is no erasure encoding or checksum provided by GNR.

Additional Ustore Features and Enhancements Being Considered

- Reliable tier with higher capacity:
  - Performance pool content managed by policy (e.g., when performance pool smaller than reliable pool).
  - Cooperative caching: Reading the performance copy from another node is typically faster than reading the reliable copy but this may not be true when there are many concurrent readers of the performance copy.
- Create local copies of different filesets on different subset of clients:
  - Supports performance isolation, separating read traffic of different projects.
- Propagate updates asynchronously to the reliable copy, allowing writes to return after the performance copy is updated.
  - Eventual reliability (eg, “Burst Buffer”)
- When missing updates are repaired, handle all missing updates in the given indirect block.
- Persistent memory support enables new programming paradigms:
  - Map performance copy in persistent memory into application address space.
  - Replicate to reliable tier asynchronously or on demand (msync).

Thank you for using

Storage Scale

Storage Scale System