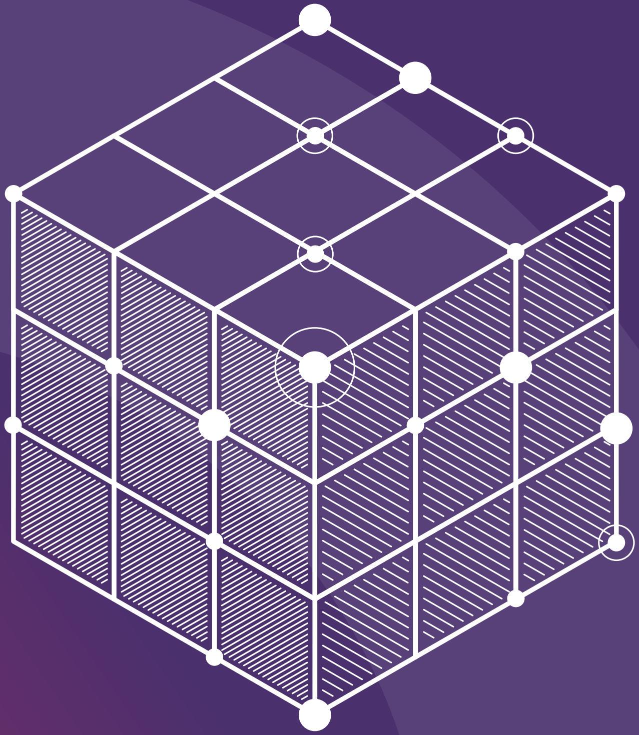
Parallel Distributed Storage

TECHNICAL CAPABILITY SHEET



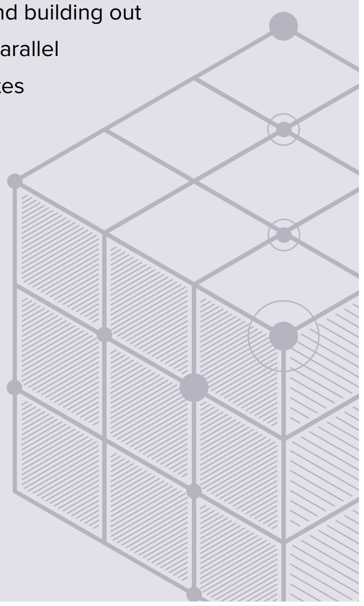
open DRIVES

Capability Definition

In a standard scale-up storage solution, you can add resources (capacity, memory, CPU upgrades) or even swap them out for newer component generations to create ever-higher levels of performance. Scaling up enables you to support more demanding workflows and workloads in dynamic environments where resource consumption can be extreme due to the constant changes in data sets. This scale-up approach works well to a certain point, but ultimately scale-up systems can and do reach a maximum threshold of performance. You are then no longer able to add more resources to continue growing and evolving the storage system. Yes, your storage solution is maxed out!

Fortunately, another architectural approach can alleviate the upper performance limitations you might encounter with scale-up solutions. Scaling out your storage infrastructure means that you add new nodes, stacks, and clusters in parallel that all operate in tandem with one another, sharing the workload among all the clusters. A parallel file system, however, is required to manage resources across the clusters and to make sure that every component operates within the single namespace which is desirable for a scale-out architecture. Our Atlas Core software implements Lustre, which creates parallel distributed clusters and allows for massively large scaling and high-performance computing capabilities.

Note: Lustre is an open-source project built around managing and building out the application's working code base. Lustre enables massively parallel distributed storage clusters with an overlay file system that creates a single namespace, facilitating data access across the distributed environment. As with most other open-source applications, an entire community of technical researchers and developers collaborates and contributes to adding new technical capabilities to the Lustre code base.



Did You Know? Lustre powers some of the world's largest supercomputing projects in high-tech, scientific research, government, and academia. OpenDrives has a strong commitment to working with the Lustre open-source community to make it a more robust and usable tool. Other collaborators include governmental agencies such as the US Department of Energy as well as major tech companies, including Amazon, Hewlett-Packard, and Intel.

Outcome & Value

The outcome of parallel distributed storage clusters is the ability to add resources horizontally to an enterprise compute and storage infrastructure instead of scaling up and ultimately hitting the upper threshold of scale-up architecture. Ultimately, more computing power, memory, and storage capacity can help you tackle larger workflows, increase speed and throughput dramatically, and manage the single namespace and capacity storage pools more effectively. As a matter of fact, the larger the storage infrastructure becomes, the more cost-efficient it is to add new high-availability stacks and clusters to the parallel architecture instead of investing resources into scale-up upgrades.

How It Works

Lustre incorporates services that provide data and metadata management functionality as well as access to distributed and shared storage targets throughout the clustered environment. The main servers in this architecture include the Object Store Service (OSS), Metadata Service (MDS), and Management Service (MGS). Lustre uses a handful of targets for data (OSTs, or object storage targets) and metadata (MDTs, or metadata targets). Physical data resides on the persistent OSTs, while metadata is stored on the persistent MDTs.

In this architectural design, the OSS manages a number of capacity targets (OSTs), fielding and handling all data read/write requests to the OSTs within the distributed environment. All of this is done in conjunction with ZFS, which is the Atlas Core file system necessary to handle local data requests on each individual OST. The role of the MDS and MDT is to accept client requests, locate the physical location of requested data, and then pass that along to the requesting client. The subsequent I/O process is managed by the OSS and associated OST, which carries out the read/write request for the client.

Here is what a standard parallel distributed storage deployment might look like. Assuming four separate high-availability (HA) stacks, each stack consisting of two compute modules and also sharing a set of disks. The first stack (stack A) would have the MGS, OSS, and MDS services running on it. The B, C, and D stacks would have just the OSS and OSTs running on them. Of course, the real power of Lustre is that in this multi-stack environment we are taking all the capacity spread among the four stacks and turning that capacity into a single namespace. Of course, you can add more stacks quite easily—bring up the HA stack, create the ZFS pool, then create an OSS and OST to add all this to the main cluster. This is how scale-out parallel distribution functions—quick and easy expansion to accommodate whatever your data ecosystem demands.

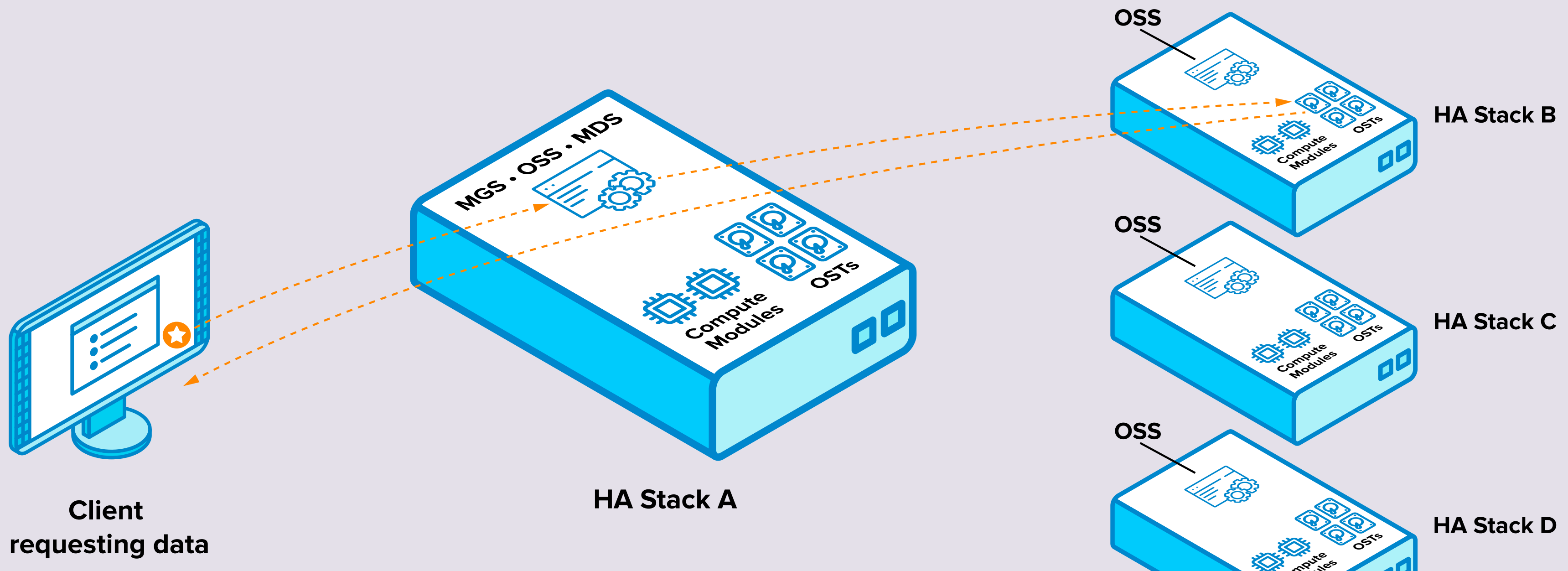
Note: Keep in mind that Lustre works in conjunction with ZFS to create a parallel distributed environment. Lustre performs the global management across the entire storage infrastructure (across the clusters), while ZFS handles transactions locally within the stack. The good news is that Lustre has hooks into Atlas Core's ZFS capabilities, so these two technologies work seamlessly together to scale out as much as is needed to handle workflow requirements.

Characteristics

Our parallel distributed storage capability, powered by Lustre, has the following characteristics:

- Leverages Lustre from the open-source community to create a parallel file system overlay to manage multiple or many stacks and clusters working in parallel.
- Creates and manages a single global namespace, which means that it unifies the storage capabilities inherent to each individual storage cluster that otherwise would exist as separate and discrete filesystems, or data silos.
- We refer to this capability as a parallel distributed single namespace, which means that you can concurrently talk to more than one node (parallel) within a topology consisting of more than one node (distributed stacks) and a single namespace consisting of an underlying single directory tree unifying all the file system attributes.
- Lustre has direct hooks into ZFS, which we use within our Atlas Core software to manage the block storage area. Therefore, Lustre is the single namespace overlay that interfaces directly with our implementation of ZFS for the block storage area on the local capacity nodes.
- We even have scale-in capabilities to make it easy to phase out older-generation equipment without hard outages or immediately decommissioning production equipment. You can put a stack into evacuation mode, which means that as a background process it begins moving data from the node being evacuated to new nodes. Once all data has been evacuated, you can decommission the evacuated stack from the larger cluster without any effect on the larger environment.
- In a multi-stack environment, we make sure that all capacity is available throughout the clusters via the single namespace entity.
- Our capability makes it easy to add stacks when your current ones can't accommodate your operational workflow demands anymore. Just add more to the parallel environment without the need to upgrade vertically (scale up).

The following diagram illustrates a simplified version of the standard deployment and also indicates the roles of the major services, including the OSS/OST, the MDS/MDT, and the MGS. Clients talk to the MDS for a lookup of data, then to the OSS which works with the OST to read or write data as requested.



Further Reading

In this sheet, we mention quite a bit about Atlas Core's ZFS capabilities which are needed for local filesystem operation with the distributed infrastructure managed by Lustre. For more information about ZFS functions such as caching and prefetching, please refer to the Caching Capability Sheet or the Prefetch Capability Sheet which explain in more detail the mechanics behind these features.

sales@opendrives.com | +1 888-778-5491

open DRIVES