SOLUTION BRIEF

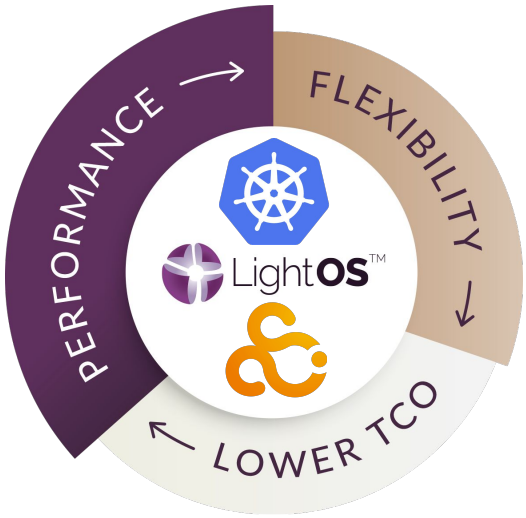
Kubernetes® and LightOS™
Performance, Persistence, Simplicity

Containers and Kubernetes have taken the application world by storm as a way to offer a simplistic way to scale and manage applications efficiently by dividing them into a set of fine-grained services ("microservices") that are loosely coupled. By separating the services into their own containers, it's possible to scale them independently.

Lightbits Labs LightOS™ is software-defined block storage bringing hyperscale efficiency and flexibility to all. It delivers composable, high-performance, scale-out and redundant NVMe/TCP storage that performs like local flash.

LightOS persistent storage for Kubernetes, via the Container Storage Interface (CSI), provides a storage solution that extends the efficiency of containers to NVMe flash while maintaining performance, flexibility and application portability.

WHY LIGHTBITS?



Cloud native storage for Kubernetes. Simplify high performance, scalable and persistent container storage while lowering cost

ADHERING TO PORTABILITY PHILOSOPHY

Containers themselves were designed to make application services portable, freeing services from server physicality. They can move between hosts, data centers and environments. They can scale up or down, or be started elsewhere if a server fails.

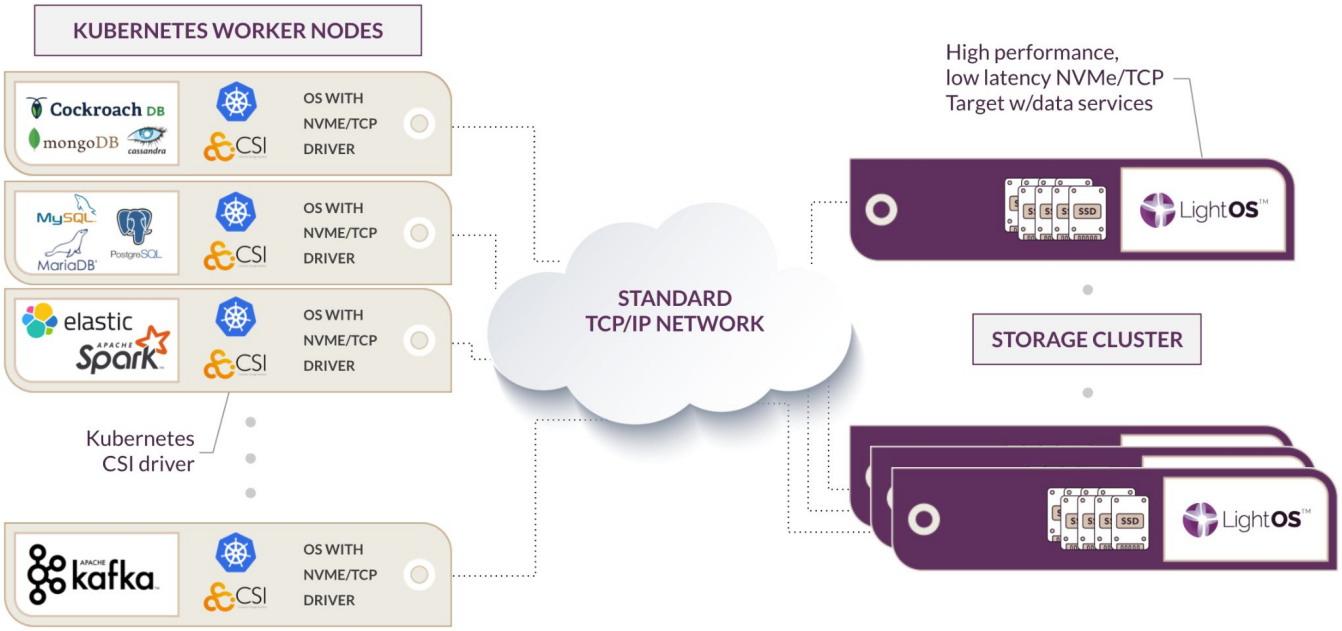
Kubernetes, originally designed by Google, is an open-source container-orchestration system for automating the deployment, management and scaling of container based applications. As more and more stateful services such as Apache Kafka, Apache Cassandra, and MongoDB move to container deployment under Kubernetes, they need persistent, low latency, high performance storage. It's tempting to use Local Persistent Volumes for local NVMe SSDs due to their performance characteristics but this breaks the Kubernetes model of container/pod portability.

LightOS storage for Kubernetes offers the best of both worlds. The performance of Local Persistent Volumes while retaining pod autonomy and freedom from server physicality.

APPLICATION PORTABILITY

LightOS logical volumes, via CSI, allow for container with persistent volume claims to move freely from one physical Kubernetes server to any other server on the network. Pod movement to a different physical server due to failure, maintenance or upgrade will trigger the same logical volume to be attached and made available to the pod.

Just as containers can't rely on special hardware to ensure portability, LightOS persistent volumes must be able to be attached to Kubernetes compute servers without requiring special hardware or protocols. Most NVMe-oF implementations require RDMA via the RoCE protocol, requiring specialized network interface cards (NICs) and often special drivers. NVMe/TCP drivers are part of all major Linux distributions and because they are 100% open source and in the upstream kernel, the drivers are readily available even for older kernels/distributions. NVMe/TCP works on any Ethernet NIC. Thus you have complete portability in your server infrastructure as there are no special requirements for NIC drivers, block drivers or special NICs.



Overview of a Kubernetes deployment utilizing LightOS NVMe/TCP targets for persistent volumes

APPLICATION ACCELERATION

As more and more stateful applications move to containers under Kubernetes orchestration the need for low latency, high performance persistent volumes is growing. Most of the popular stateful applications call for local SSDs, with NVMe highly recommended when deployed on bare metal:

- MySQL
- PostgreSQL
- MariaDB
- MongoDB
- Redis
- Apache Cassandra
- Splunk
- Apache Kafka
- Apache Spark

As described above, utilizing local SSDs with the Kubernetes Local Persistent Volume functionality would meet the storage requirements of these applications but would break container/application portability. Lots of storage solutions offer persistent volumes but not anywhere near local flash performance.

LightOS persistent volumes perform like local NVMe flash. On Kubernetes servers with 25Gbps Ethernet ports or faster, LightOS persistent volumes may even outperform a single local NVMe drive. This is due to the relatively low write performance of NVMe drives compared to their read performance as well as a reduction in contention for resources if multiple containers were utilizing different namespaces on the same NVMe drive.

Thus, with LightOS persistent volumes in your Kubernetes environment, applications whose best practices call for local NVMe flash can get the same performance yet maintain the portability associated with containers and pods.

BETTER PROTECTION AGAINST FAILURES

In a Kubernetes deployment with applications that perform their own data protection via replication, that replication takes CPU and network resources. If a server or drive fails, even temporarily, the resulting rebuild takes both precious network and CPU resources on both the surviving replicas, and replica being repopulated or synchronized. With such an application running in Kubernetes, a Local Persistent Volume drive failure would result in a full data rebuild either on a new application server (if the container/pod moves) or on a new drive after replacement. This rebuild would come from the surviving replicas on other servers, over the network. This can have a long and detrimental effect on the network and the application service itself.

LightOS protects against drive failures with Elastic RAID on the NVMe/TCP target servers themselves. In the case of a drive failure, any data written to the drive will be seamlessly rebuilt into spare capacity within the target server resulting in minimal service disruption. This means the recovery takes much less time and does not impact network performance or burden application services with additional load.

With LightOS persistent volumes, in the case of a Kubernetes server failure, pods will move to alternate servers, the persistent volumes will be attached and the applications will only need to synchronize changes that were missed while the service was offline.

LightOS 2.0 and above is built on a scale-out, highly available architecture and volumes designated with either 2X or 3X replication will remain available if a target NVMe/TCP server fails.

LOWERING TOTAL COST OF OWNERSHIP (TCO)

LightOS lowers your total cost of ownership both for the initial purchase, as well as over time with greater operational efficiency.

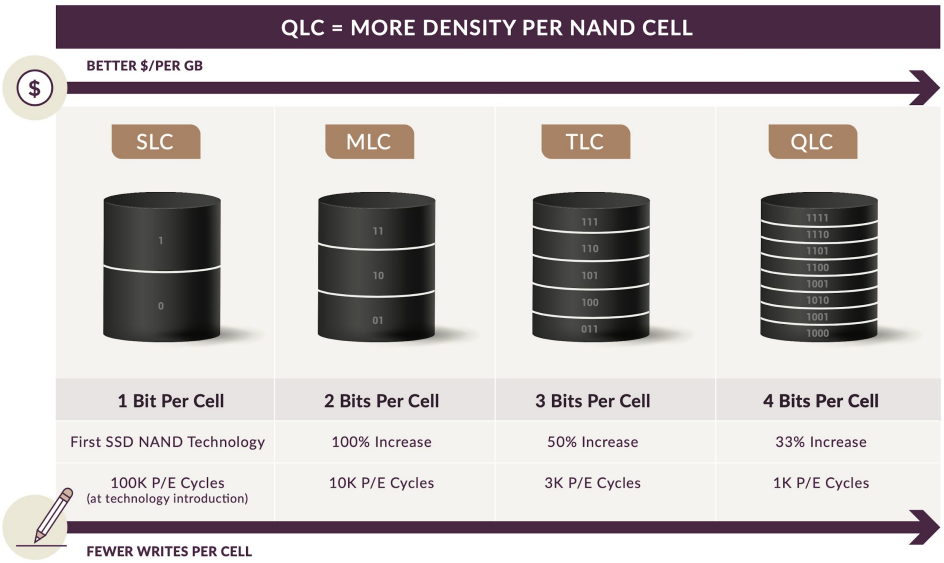
Kubernetes environments utilizing local NVMe with or without Local Persistent Volumes are often only 15-25% utilized. When moving to a LightOS persistent volume storage service there are vast improvements in capacity and performance utilization. This means less money is spent on NVMe flash while providing a more operationally efficient Kubernetes environment.

RICH DATA SERVICES

Lower TCO is not only achieved by improving capacity and performance utilization. LightOS offers rich data services that are not generally associated with NVMe storage, at NVMe performance latencies. All LightOS persistent volumes are thin provisioned and when combined with compression support (that can be enabled/disabled on a per-volume basis), LightOS can achieve total data reduction levels as high as 10:1 in service provider and private cloud environments. Additionally, thin snapshots and clones allow for DevOps functionality in dynamic container environments by making development datasets/databases instantly available with the same performance as their parent.

ENABLING QLC FLASH

QLC flash is inexpensive, but often not suitable for use in Kubernetes servers where the write pattern is unpredictable. Write performance of QLC flash is poor when compared to more expensive TLC and MLC devices. Lastly, unless writes are sequential and in large chunks, it's possible to wear out QLC media quickly. Thus, it's difficult to take advantage of the lower cost of QLC flash directly in application servers, especially in service environments based on Kubernetes.

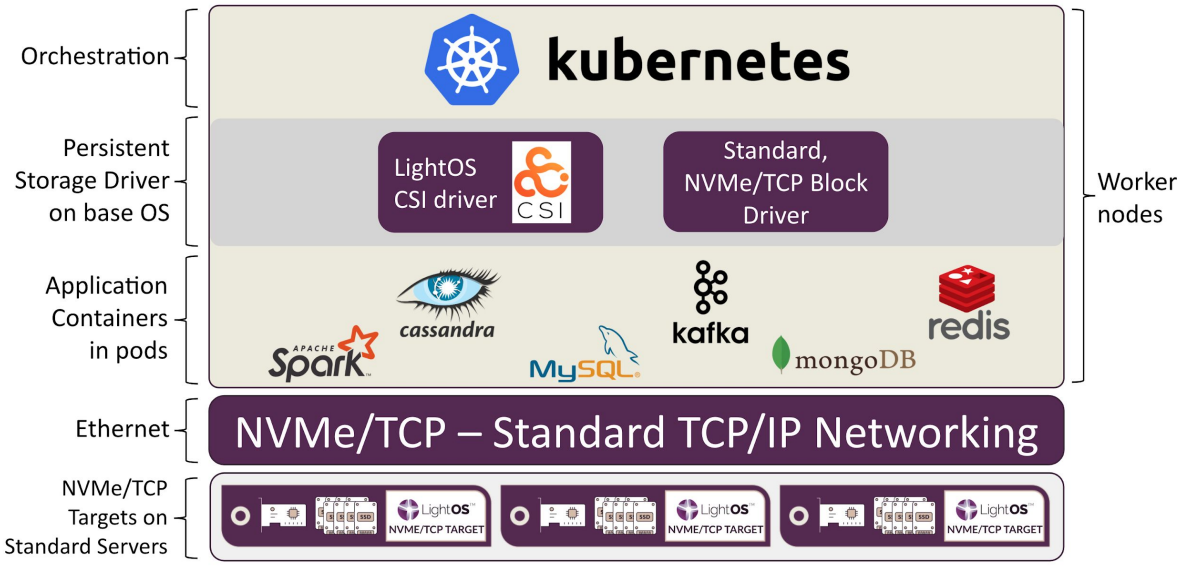


LightOS makes QLC flash viable in a Kubernetes service environment

DEPLOYING NVME/TCP ON KUBERNETES SERVERS

Lightbits Labs developed NVMe/TCP and contributed it to the upstream kernel so it's open source and has been part of the standard Linux kernel since kernel 5.0. Many enterprise Linux variants have since backported the block driver to earlier kernels in their distributions. These include Red Hat Enterprise Linux, CentOS, Ubuntu, SUSE SLES, OpenSuse, Debian and Fedora. Regardless of base Linux distribution used under Kubernetes, the NVMe/TCP driver is already there or readily obtainable.

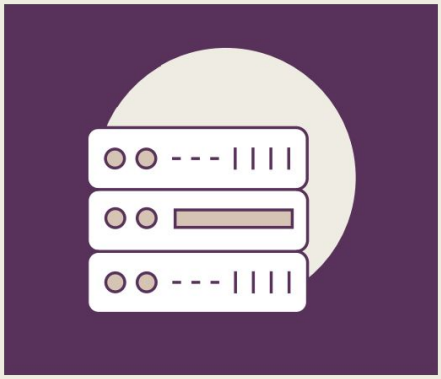
NVMe/TCP utilizes the same network cards and infrastructure that might be used by protocols like iSCSI. Network switches don't require any special settings for the NVMe/TCP storage protocol. The LightOS NVMe/TCP target software is a set of software packages applied to a base Linux distribution such as those listed above. It's a target side only solution that works with the networking hardware and practices already in place in the data center.



The Kubernetes, NVMe/TCP initiator, CSI and LightOS NVMe/TCP target stack

DEPLOYING LIGHTOS NVME/TCP TARGETS FOR KUBERNETES

THE POWER OF CHOICE



Select the x86 server platform hardware and NVMe drives from a vendor of choice

SuperSSD™ makes deployment easy with full hardware and software support

SOFTWARE-DEFINED STORAGE

LightOS as software is licensed per storage server on an annual subscription basis. It runs on x86 servers and utilizes standard Ethernet cards and NVMe drives. In general, the minimum CPU requirement is 10 cores. Target servers, in general, should use 100Gbps Ethernet interfaces with each interface capable of supporting up to 8-10 GB/s of storage bandwidth, largely depending on the number of CPU cores. Lightbits Labs is happy to provide reference platform guides and/or consulting on the right server configuration tailored to workload requirements.

DEPLOYMENT READY APPLIANCES

For those that value convenience in a heavily tested and optimized platform, LightOS is available as a SuperSSD appliance. This 2U, 24 drive platform comes in various pre-configured capacities with software and hardware support. This is the fastest and easiest way to get LightOS deployed and is backed by world-wide warranty and support services.

WHY LIGHTOS FOR KUBERNETES

What's needed for optimal Kubernetes persistent storage is a solution that is as flexible and portable as containers yet performs like local NVMe SSDs. To preserve container portability, it must speak common network protocols and cannot require special NICs. It must be standards based, managed via an API and run on standard servers.

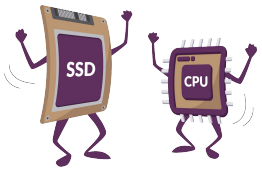
LightOS meets both the philosophical and technical requirements to be the best high performance persistent storage solution for Kubernetes. It supercharges your Kubernetes based applications while increasing reliability and flexibility by providing:

- The same performance as flash Local Persistent Volumes with greater utilization of your storage investment
- Improved service levels and a better user experience with consistent latency
- Faster rebuild time with higher resiliency levels
- Standard, simple, secure storage access to any of your Kubernetes application servers
- No changes to your TCP/IP network with no proprietary drivers on Kubernetes servers

Finally, you can separate storage from compute without all the drama, and at lower cost.

FIND OUT MORE

To learn more, please visit our website, www.lightbitslabs.com
To contact our team, email us at info@lightbitslabs.com



kubernetes

CSI
Container Storage Interface