

# Container-Native Storage for DevOps

Dynamically provision high-performance persistent storage to stateful applications on Kubernetes clusters

## BENEFITS

- **Accelerate responsiveness of stateful workloads** through low-latency access to NVMe drives
- **Streamline DevOps CI/CD pipeline** with containerized storage services orchestrated directly from Kubernetes
- **Replace complex storage management** with native Kubernetes automation
- **Gain observability** to storage behavior not possible with external systems
- **Ensure portability** of storage services by removing kernel/OS dependencies
- **Enhance data resilience** through volume replication across K8s nodes

Modern cloud-native applications require modern containerized solutions to run and manage them. Traditional monolithic storage alternatives cannot address the dynamic provisioning requirements of microservices at scale, nor the responsiveness and location independence expected of them. DevOps personnel and site reliability engineers (SRE) depend on a fast, reliable, and flexible infrastructure to streamline CI/CD operations and deliver high-performing applications under varied conditions.

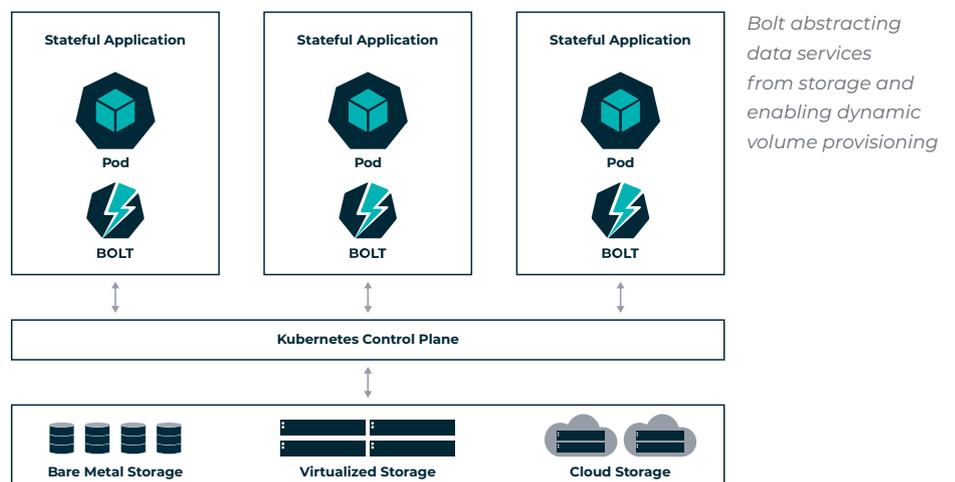
While Kubernetes provides the orchestration platform to automate and manage the container lifecycle, many projects underestimate critical dependencies on the data storage layer, especially in terms of performance, integration, and portability. Storage related problems that are not apparent during proof of concept become evidently clear in production when spread out over hundreds or thousands of pods in diverse environments.

Many of those unforeseen problems stem from relying on outside control of storage resources from systems and processes designed for much more static and centralized operations.

## Ensure Responsiveness, Manageability, and Portability of Critical Storage Services for Consistent Behavior Across Diverse Container Deployments

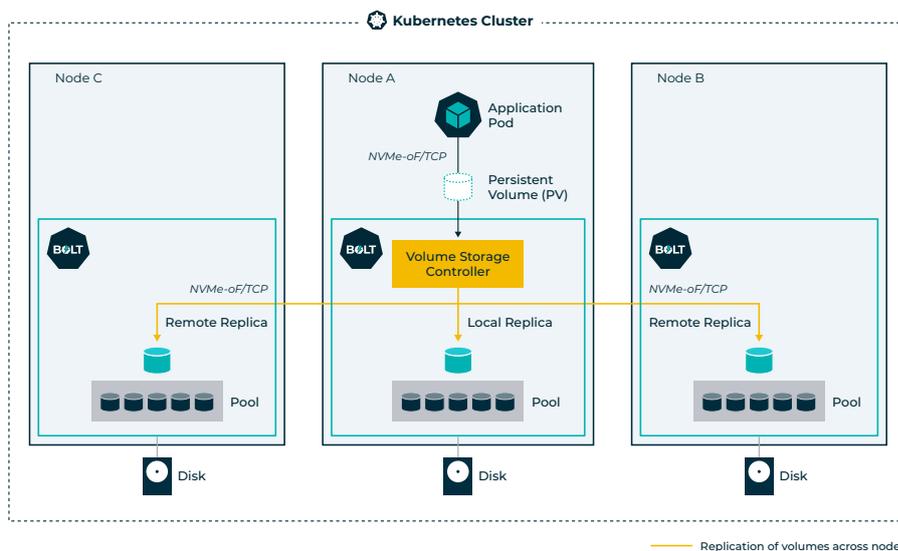
A sure way to avoid external storage dependencies and their undesired surprises is by letting Kubernetes manage storage allocation, load balancing and failover as an integral part of maintaining your desired state. For that you must first containerize the most sought-after storage services. **DataCore Bolt** software offers such a solution, borne out of our extensive experience with the top open-source storage package, OpenEBS.

No need to be concerned with special OS or kernel calls. Bolt runs completely in user space, ensuring widespread portability. The code you build and test in one environment will behave the same way in another. Deploy with equal confidence in multi-cloud scenarios, on-premises, and at the edge.



You will enjoy dynamic provisioning of persistent volumes for your stateful workloads, that also happen to be super-fast. Fully exploit the raw speed and low latency of NVMe SSD hardware by eliminating much of the overhead and interrupts that other alternatives introduce.

Just deploy the lightweight Bolt container-native storage software on your Kubernetes nodes – either bare metal or virtual servers. Bolt abstracts the disk attached to the node and creates a storage pool. Use conventional declarations to claim persistent volumes which are logically partitioned and provisioned to pods just like any other resource. Leverage volume replicas within the cluster to protect against hardware failures and data loss.



*Bolt protecting volumes against failure by creating replicas across K8s nodes*

These simple steps facilitate the deployment and release of production-ready software while enabling enterprise-scale agility. Minimize wait time and accelerate code progression to meet your goals of continuous integration, delivery, and deployment. Harnessing the benefits of NVMe and NVMe-oF puts you in a position to support a composable and disaggregated storage architecture that delivers low-latency and enhanced responsiveness for stateful workloads.

No storage expertise is needed to deploy and manage Bolt: your DevOps engineer/SRE or K8s admin can focus on their core tasks while Bolt provides platform-independent storage services to fulfill the needs of containers. Simply schedule Bolt pods to run on the same worker nodes as the application, drawing capacity from a local disk to create persistent volumes. Those logical volumes will be dynamically provisioned on-demand to stateful workloads and may be replicated to other nodes for added resilience and throughput. NVMe over TCP is used for high-speed, synchronous replication between nodes. The type of disk varies depending on whether the node is in the cloud, on a virtual machine, or on a bare metal server.

### Why Choose DataCore Bolt for Your Kubernetes Environment



**Persistent Volumes for Kubernetes**



**High Performance for Stateful Apps**



**Easy Capacity Scaling**



**Circumvent Storage Outages & Failures**



**Platform-Agnostic Storage**



**Software-Defined Automation**

You can seamlessly snap Bolt into your DevOps ecosystem alongside other tools used for version control, configuration management, CI/CD automation, code repository and artifact management, test automation, and so on. Out of the box integration with monitoring tools (such as Prometheus) provides logs and metrics for complete observability.

Enhance the experience and efficiency of I/O-intensive applications using Bolt. Empower your organization with the agility to scale the Kubernetes infrastructure to support stateful containerized applications and the DevOps pipeline. Contact DataCore to learn more about Bolt container-native storage.



### Discover the Ultimate Flexibility of DataCore Software

DataCore Software delivers the industry’s most flexible, intelligent, and powerful software-defined storage solutions for block, file, and object storage, helping more than 10,000 customers worldwide modernize how they store, protect, and access data. With a comprehensive product suite, intellectual property portfolio, and unrivaled experience in storage virtualization and advanced data services, DataCore is The Authority on Software-Defined Storage. [www.datacore.com](http://www.datacore.com)

**GET STARTED**