

## Technical Review

# ROBIN Storage for Containers: Enabling Stateful Applications on Kubernetes

Date: July 2019 Author: Kerry Dolan Senior IT Validation Analyst

## Abstract

This report describes how ROBIN Storage delivers bare-metal performance and enterprise data management for stateful containerized applications on Kubernetes.

## The Challenges

As organizations continue to pursue digital transformation initiatives, many have adopted container technologies to streamline application needs, get applications to market faster, and make them more portable. At the same time, Kubernetes has become the orchestrator of choice for deploying, managing, and scaling containers. While development remains a key container target, more organizations are deploying containers in production applications. When ESG asked IT managers about their production container usage in 2018, 56% reported having already deployed applications in production, 24% reported testing with a plan to deploy within a year, and another 16% reported that they expected to start testing production containers in the next year.<sup>1</sup>



Had deployed containers in production



Were testing and plan to deploy within 12 months



Intended to start testing within 12 months

Why the increased interest? Container technologies abstract applications from hardware by virtualizing the operating system, which is a lightweight design that makes them efficient, reliable, scalable, and portable. Containers enable development autonomy and agility, as developers can do more on their own without IT provisioning or management. The infrastructure and staffing efficiency of containers result in lower costs and streamlined processes.

Stateless containers have no need to keep data persistent once the processes they are executing have finished. A key challenge for running enterprise-class, container-based production applications is that they are most often stateful: that is, the applications maintain data from each compute session, even when the container terminates. As a result, running applications such as databases, artificial intelligence/machine learning (AI/ML), or custom-built applications on Kubernetes requires external storage that outlasts the container. When running mission-critical processes, these applications need swift storage provisioning, predictable performance, full data protection and security, easy data sharing, and the flexibility to leverage hybrid/multi-cloud deployments.

There are numerous external storage solutions that support containers through Container Storage Interface (CSI)-compliant APIs to manage interactions between container orchestrators such as Kubernetes and storage arrays. However, these solutions do not provide the performance and data management capabilities that enterprise production applications demand. They provide storage at the volume level but cannot deliver application-level data services.

## The Solution: ROBIN Storage

ROBIN Storage is a CSI-compliant, container-native, software-defined block storage solution that offers enterprise-class performance and data management capabilities for Kubernetes-orchestrated containers. It provides resilient storage (supporting HDD, SSD, and NVMe) with bare-metal performance, and has built-in data rebalancing, disk and I/O error-detection, volume rebuilds, and hotspot detection. Data services include automated provisioning, point-in-time snapshots,

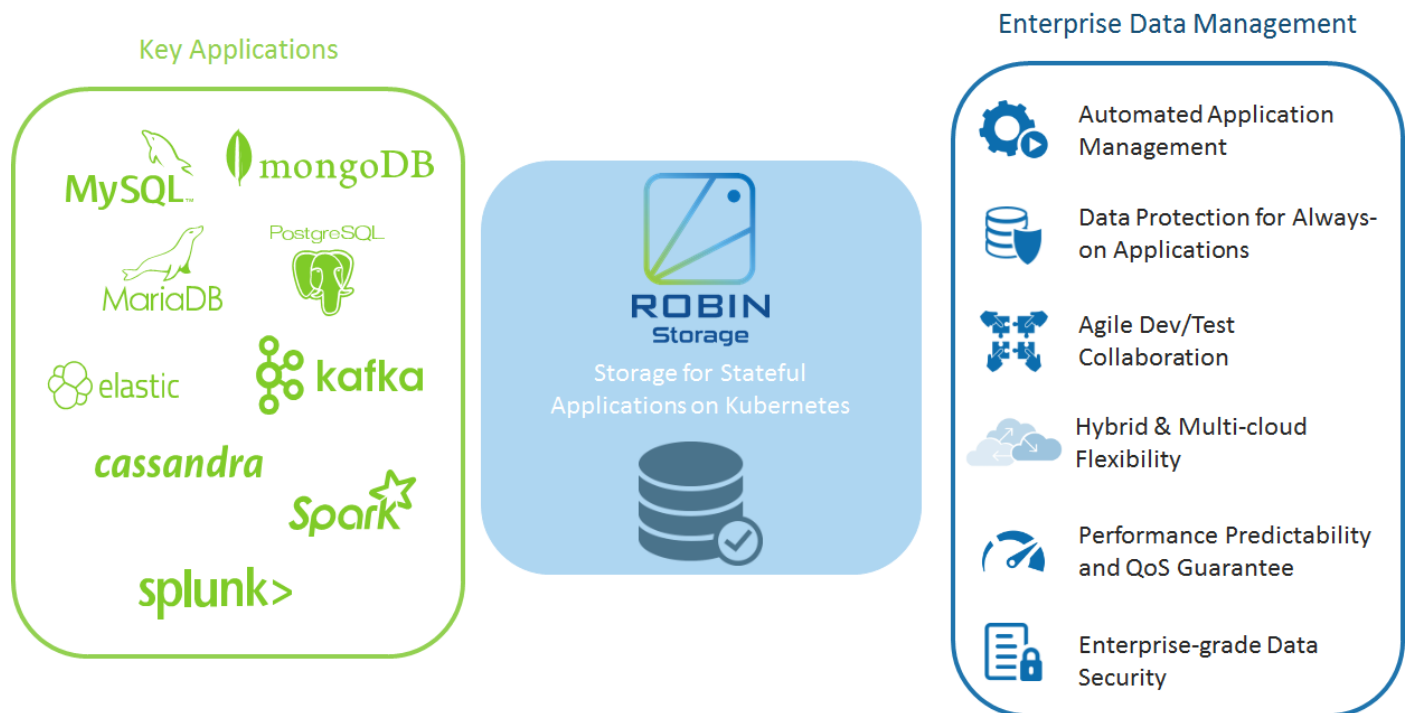
<sup>1</sup> Source: ESG Brief, [The Growth in the Use of Application Containers](#), May 2018.

backup/recovery, cloning, guaranteed QoS, and multi-cloud migration for stateful applications on Kubernetes. ROBIN Storage leverages standard APIs that are integrated with tools with which DevOps teams are familiar, enabling provisioning and management via Kubectl, Helm, and Operator frameworks. Data management operations throughout the application lifecycle are automated, simple, and cover the complete application, not just the data. ROBIN Storage has been validated to support enterprise workloads on Google Kubernetes Engine (GKE), Google Anthos, and Red Hat OpenShift. ROBIN Storage was designed to easily integrate with any Kubernetes deployment, and integrations with EKS, AKS, PKS, and IKS are in process.

Enterprise applications in production are stateful and require storage features that are considered “table stakes” for any application today. ROBIN Storage brings these features to stateful containers, letting organizations:

- *Automate application management.* Lifecycle operations include application configurations and metadata, not just storage volumes. For example, when DevOps staff snapshot an on-premises MariaDB, they can move it to the Google Cloud and the entire application will come up, configured properly.
- *Protect always-on applications.* Built-in replication ensures high availability, while snapshot, backup, and recovery functionalities provide protection from errors and failures.
- *Collaborate easily.* Cloning of entire application environments makes them easy to share across teams for consistent development.
- *Leverage hybrid and multi-cloud flexibility.* The ability to move applications and data between on-premises data centers and multiple clouds gives organizations choice and flexibility to run applications where they are most efficient and cost-effective.
- *Ensure bare-metal performance with QoS guarantees.* Bare-metal storage performance lets organizations run mission-critical applications with high performance requirements on containers, while ensuring that SLAs are met with QoS guarantees. This enables application consolidation for reduced costs.
- *Ensure data security.* Kubernetes-native authentication is supported, and data is encrypted at rest and in transit. Checksum algorithms ensure that data is tamper-proof. These are essential capabilities for mission-critical applications.

**Figure 1. ROBIN Storage for Stateful Containers on Kubernetes**



Source: Enterprise Strategy Group

## Developer-focused

ROBIN Storage makes these data management operations simple for developers to execute, enabling databases such as PostgreSQL, MySQL, MongoDB, MariaDB, Cassandra, Elastic and many more. They are the ones interacting with these applications on Kubernetes; without ROBIN Storage, ongoing management of these stateful applications on Kubernetes would require assistance from storage and other administrators that would interfere with productivity.

## Application-aware

ROBIN Storage was designed from the ground up to be application-aware. Designed for Kubernetes clusters, it handles database shards running on different pods and different nodes for resilience. This is in contrast to traditional storage solutions that carve up storage volumes from the same disk, denying the database the resilience it demands. Because ROBIN Storage is application-aware, it knows the mappings of applications to pods to volumes to disks, enabling consistent application-level snapshots. It also provides QoS guarantees to eliminate “noisy neighbor” performance impacts, while traditional storage can only cap IOPS to specific volumes.

## ESG Tested

ESG validated how ROBIN Storage simplifies application management, protection, and portability.

Testing began with a three-node, on-premises Kubernetes cluster configured with ROBIN Storage. We began by creating a MySQL application from a stable Helm repository, which is simple using a single command line (`helm install - -name esg stable/mysql - -tls`). ROBIN recognized the three physical servers and virtualized their storage. We logged into the pod and populated the database with data, creating a table called *Cloud* and inserting the name *Google* as an entry on Row 1.

Next, we registered the application with ROBIN and named it *esg*, and ROBIN started tracking the details of the application. This includes not just the container and a volume, but all container details including the config map, secrets, persistent volume claims (PVCs), deployment, and services. ROBIN’s ability to track these details enables it to execute data management and lifecycle operations such as application-consistent snapshots, clones, migration, etc.

We created a snapshot with a simple command (`robin snapshot create esgsnap1`), which pointed to the Helm Chart to capture the entire application, not just data volumes. Figure 2 shows the details of the snapshot.

**Figure 2. Kubernetes MySQL Application Snapshot on ROBIN Storage**

```

+-----+-----+-----+-----+-----+
| Snapshot ID | State | App Name | App type | Snapshot name |
+-----+-----+-----+-----+-----+
| f7755430773a11e9a8c1f39e936e6e58 | ONLINE | mdb | helm | mdb_snap1 |
| 125b6f4c773d11e99c126fc09b0b9668 | ONLINE | maria | flexapp | maria_snap-maria |
| 5e1281c6781111e9bd70fb115810644d | ONLINE | test1 | helm | test1_snap1 |
| 089dce18781511e9aeec8f9c434c6036 | ONLINE | esg | helm | esg_esgsnap1 |
+-----+-----+-----+-----+-----+
[robinds@eqx01-flash03 ~]# robin snapshot info 089dce18781511e9aeec8f9c434c6036
Name          : esg_esgsnap1
Id            : 089dce18781511e9aeec8f9c434c6036
Application Name : esg
Application Kind  : helm
State          : ONLINE
Description     :
Number of pvcs  : 1
Number of backups : 0

Spec:
+-----+-----+
| Kind | Name |
+-----+-----+
| ConfigMap | esg-mysql-test |
| Secret | esg-mysql |
| PersistentVolumeClaim | esg-mysql |
| Deployment | esg-mysql |
| Service | esg-mysql |
+-----+-----+

```

Source: Enterprise Strategy Group

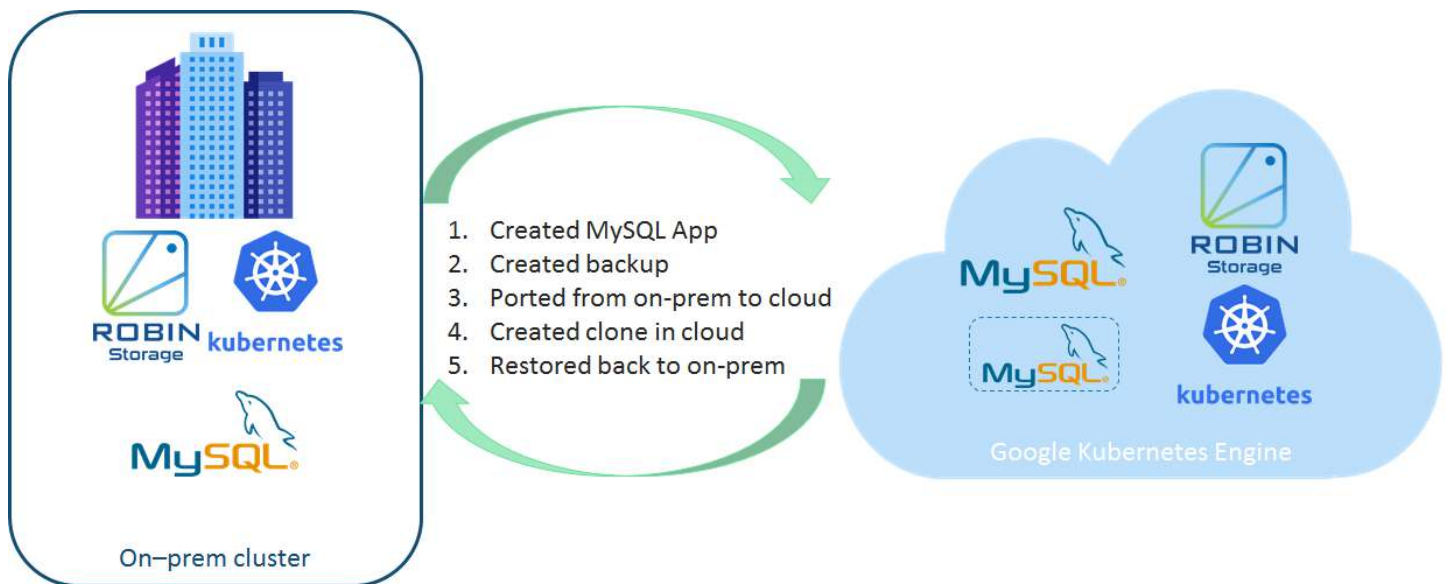
Next, we registered a cloud object store with ROBIN so we could back up the snapshot to the cloud by using the *backup create* command, providing the snapshot ID, selecting the Google Cloud repository, and getting permission to back up to the cloud.

Next, we viewed a Google Kubernetes Engine instance and a three-node cluster with ROBIN deployed as a storage class. We exported the backup to the cloud after creating a key to give the GKE administrator permission to import the backup. The entire application was then streamed to the cloud. Once it was imported, we created a clone, and watched as ROBIN cloned the entire MySQL application that we had on-premises. Once that was complete, we viewed the data and confirmed that the table and data we had created was in the backup.

- This demonstrated how simple it is to burst your test/dev clones to the cloud, and to port the entire app from on-premises to the cloud and between cloud regions to improve collaboration. ROBIN Storage eliminates the pain of waiting weeks for IT to provision storage and move applications, or creating scripts for Kubernetes configurations and secrets, right sizing templates for pods, etc. With ROBIN Storage, developers can get what they need, when they need it, without any esoteric storage knowledge.

Finally, we deleted our table from the on-premises MySQL cluster and then restored from the cloud-based snapshot. This rolled back all the data, MySQL Settings, configset, secrets, etc.

**Figure 3. ROBIN Storage Testing**



Source: Enterprise Strategy Group

## The Bigger Truth

Today's IT is no longer about discrete silos of infrastructure, each managed by specialists who take in requests and spend months procuring and configuring before delivering services. The focus is on immediate IT services, applications driven by their owners and developers, and simple management. Efficiency and agility are essential today; when asked their most important objectives for digital transformation initiatives, 55% of ESG survey respondents cited "become more operationally efficient," making it the most cited response.<sup>2</sup> Container technologies can deliver that for the right applications.

Containers and Kubernetes orchestration bring well-documented improvements in efficiency, portability, agility, and lower cost. The opportunity to gain these benefits for both stateless *and* stateful applications maximizes container advantages, particularly when they can both be run on the same infrastructure. This streamlines both capital and operational costs.

There are many vendors that offer CSI-compliant container storage that operates at the storage volume level. This is fine for stateless apps, but stateful production applications that drive businesses need enterprise-class storage features that operate at the application level. Traditional storage was not designed for containers, and a redesign can better support containerized, multi-pod workloads, which have different needs than traditional on-prem applications or VMs.

ROBIN Storage was designed to bring enterprise-class storage performance and data management to stateful application containers on Kubernetes. It is application-aware, enabling developers to get what they need quickly, share it easily, and manage its lifecycle without specialized knowledge or waiting for storage administrators. Organizations can now leverage containers and Kubernetes to get new products out faster, collaborate easily across teams, recover from errors and failures, avoid infrastructure lock-in, deliver predictable performance, and keep data protected and secure.

ESG validated that ROBIN Storage can bring a new set of applications into the containerized world: Enterprise, stateful, legacy production applications can now gain the benefits of containerization that include efficiency, flexibility, agility, portability, and lower costs. Our testing validated the ease of management, data protection, and application portability across on-premises and cloud deployments.

In describing the solution's value, one ROBIN Storage employee commented, "We don't want IT infrastructure to get in the way of developers turning great ideas into reality." ROBIN Storage delivers enterprise data management and bare-metal performance that fit into the developer's workflow for stateful applications on Kubernetes. If you are looking for an opportunity to increase efficiency, agility, and portability for stateful applications using containers and Kubernetes, ROBIN Storage can get you there now.

All trademark names are property of their respective companies. Information contained in this publication has been obtained by sources The Enterprise Strategy Group (ESG) considers to be reliable but is not warranted by ESG. This publication may contain opinions of ESG, which are subject to change. This publication is copyrighted by The Enterprise Strategy Group, Inc. Any reproduction or redistribution of this publication, in whole or in part, whether in hard-copy format, electronically, or otherwise to persons not authorized to receive it, without the express consent of The Enterprise Strategy Group, Inc., is in violation of U.S. copyright law and will be subject to an action for civil damages and, if applicable, criminal prosecution. Should you have any questions, please contact ESG Client Relations at 508.482.0188.

The goal of ESG Validation reports is to educate IT professionals about information technology solutions for companies of all types and sizes. ESG Validation reports are not meant to replace the evaluation process that should be conducted before making purchasing decisions, but rather to provide insight into these emerging technologies. Our objectives are to explore some of the more valuable features and functions of IT solutions, show how they can be used to solve real customer problems, and identify any areas needing improvement. The ESG Validation Team's expert third-party perspective is based on our own hands-on testing as well as on interviews with customers who use these products in production environments.

<sup>2</sup> Source: ESG Master Survey Results, [2019 Technology Spending Intentions Survey](#), March 2019.