DIAMANTI

Best Practices For Building High-Performance Stateful Databases On Kubernetes

Wallie Leung, Sr. Dir. Solution Engineering

Agenda

- Why Run Databases In Containers?
- Advantages / Disadvantages Of Different Kubernetes Infrastructure Types
- A Look At DIY Kubernetes Infrastructure
- Persistent Storage For Databases
- MariaDB: Architectural Advantages
- Case Study
  - Fortune 50 Bank Adopts Containerized DBs, Deploys On Bare Metal Infrastructure

Why Run Databases On Bare Metal Kubernetes?

- Better performance
- Higher density per node (no VM tax)
- Accelerated time-to-production, updates, fixes
- Easier, more granular scalability
- Leveraging open source technologies eliminates hefty license costs
- Conducive to DBaaS

Where Should You Deploy Your Containerized Databases?

DIY Infrastructure

PROS:

- Low complexity at small scale
- Experience with traditional technology components

CONS:

- Longer time-to-value
- With a VM layer, resource efficiency is reduced and complexity increases
- High complexity and cost at scale
- Difficult to manage at scale

Public Cloud

PROS:

- 100% OPEX means lowest cost at small scale
- Proven, resilient infrastructure as a service

CONS:

- Highest costs at large scale
- Difficult to optimize efficiency
- Security and compliance concerns
- No bare metal support

DIY Bare Metal

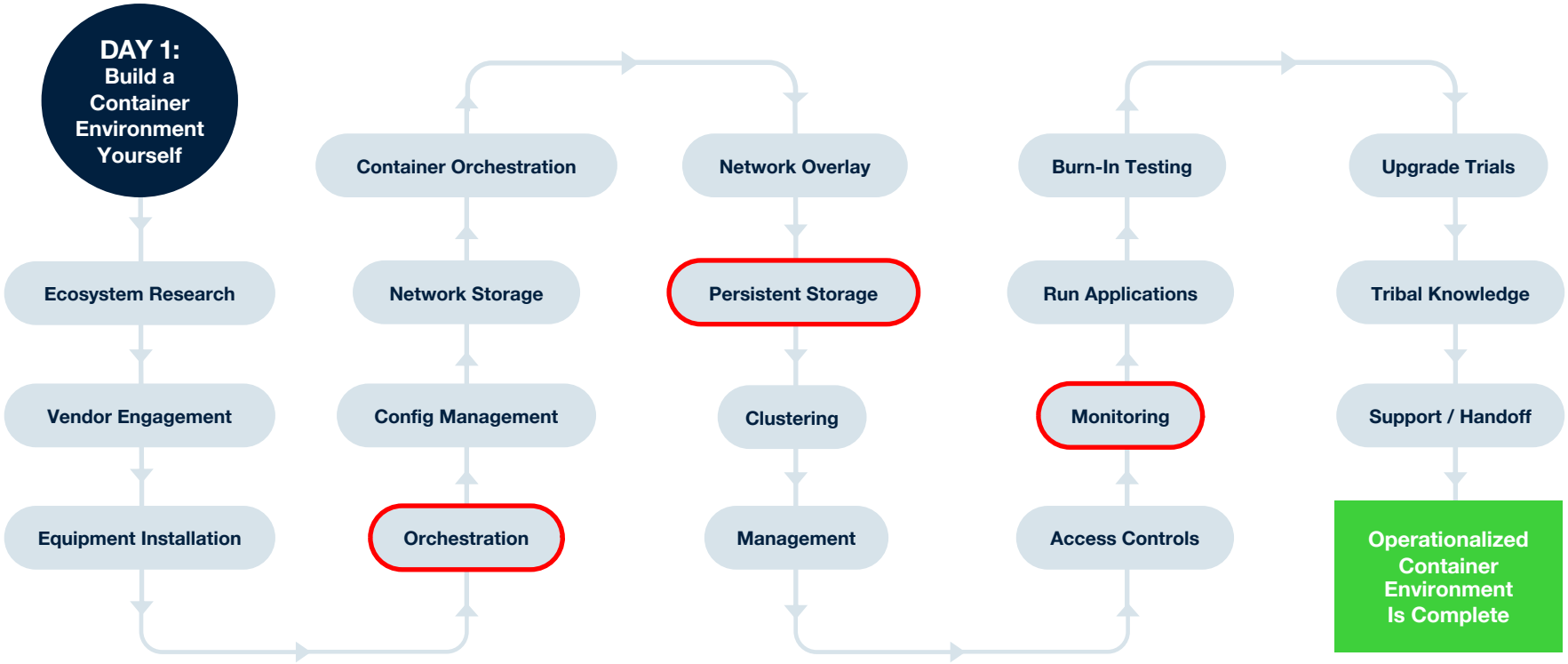
PROS:

- Flexible, seamlessly scalable infrastructure
- Better overall performance at scale
- TCO advantages are realized

CONS:

- Specialized skill sets required at different areas of the stack

Do-It-Yourself Approach to Container Infrastructure



Storage Remains a Top Container Adoption Challenge

Container persistent storage for stateful applications has proven particularly difficult for the industry

"A modern storage solution must provide DevOps teams with persistent, stateful application data; allow the consumption of storage on-demand; and deliver the same level of availability and performance provided to traditional application infrastructures."

Source: https://containerjournal.com/2019/02/20/new-storage-challenges-emerge-as-container-adoption-increases/

Drilling Down: Storage For Stateful Containers

Local Storage

- Host paths can be mounted to containers in Docker and Kubernetes

CHALLENGES

- Persistence
- High Availability (HA)
- Who is managing it?

Traditional SAN Storage

- Either connect by iSCSI or FC

CHALLENGES

- Persistence

Network File System (NFS)

- Offers persistent storage

CHALLENGES

- Performance isn't adequate

Connecting The Ecosystem With FlexVolume / CSI

- FlexVolume is Kubernetes plug-in facility for 3rd party storage
- Allows a standard way of integrating 3rd party storage, including storage with enterprise features, into Kubernetes
- Developed and contributed to Kubernetes by Diamanti
- Being supplanted by CSI (Container Storage Interface), which Diamanti continues to contribute to
- Now developers and architects have storage that is accessible and configurable from the K8s API and PodSpecs

Database Container Challenges, According To MariaDB

StatefulSet

- Definition: StatefulSet manages Pods that are based on an identical container spec
- Identical pods: additional scripting required to instantiate different roles
- Complete enterprise cluster requires more StatefulSets (i.e. MaxScale+Master/Slave)

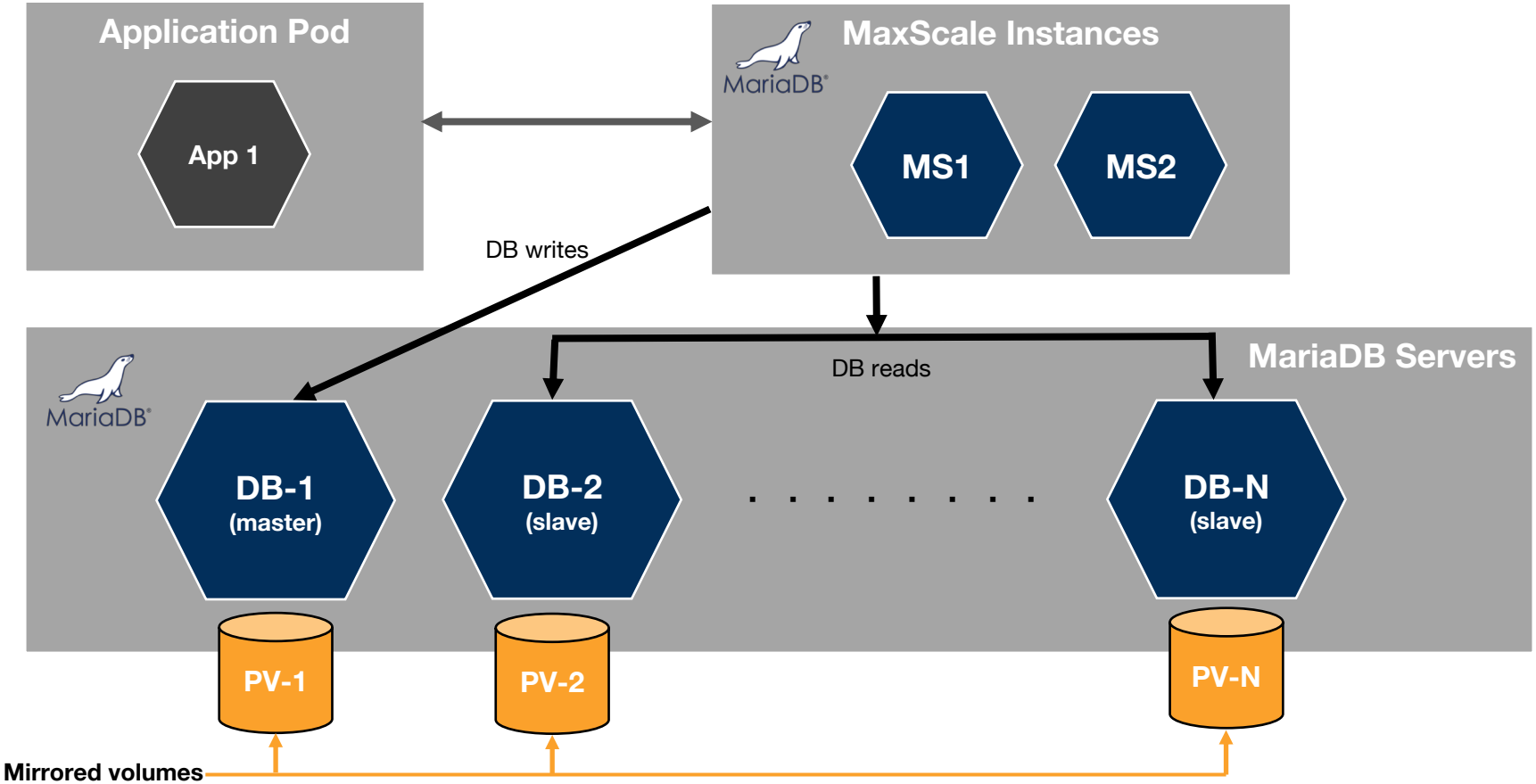
Persistence

- Persistent volumes can add new layers of abstraction (and latency)
- Provisioning / managing of IOPS or throughput is backend storage vendor-specific

Resource Management

- DB instances contend with other apps for CPU, RAM, network and storage I/O resources

DBaaS: Sample Architecture



Why MariaDB Is Suited For Kubernetes

- Lightweight, portable database runtime
  - Well-suited for containers
- Implementation in Kubernetes enables load balancing of reads and writes
  - Master DB and Slave DBs can be scaled independently
- Offer different topologies, depending on specific use case
- Offer CPU/Memory usage resources management

Containers And Database-as-a-Service

The Goal of DBaaS

- Service model that provides users the ability to provision DBs without the need for setting up physical hardware, installing software or configuring for performance

- Operational advantages with container-based DBaaS
  - App developers can check out DBs without burdening DBAs or IT Admins
  - Scaling capacity on-demand
  - Auto-recovery via Statefulset (in Kubernetes)
  - Built-in resource management for memory and CPU per pod

Containers And Database-as-a-Service On Diamanti

- Operational advantages with container-based DBaaS
  - App developers can check out DBs without burdening DBAs or IT Admins
  - Scaling capacity on-demand
  - Auto-recovery via Statefulset (in Kubernetes)
  - Built-in resource management for memory and CPU per pod
  - User defined policy for QoS – Network and Storage - per pod
  - Out-of-the-box high-performance persistent storage (500k IOPS per node)
  - Plug-n-play networking setup

Diamanti Bare-Metal Kubernetes Platform

Complete turnkey Kubernetes stack

- Hyperconverged 1U appliance built on x86 architecture
- Features container-optimized networking and storage models
- 24/7 full-stack support

Built for public cloud experience, on-prem

- Per container pod network and storage QoS
- Enterprise DP/DR features: mirroring/synchronous replication, snapshots/asynchronous replication
- No vendor lock-in: freedom for moving workload between on-perm Diamant to the cloud
- IaaS for containers

Benefits:

- High performance
- Efficient
- Secure
- Installs in minutes
- Low TCO

Customer Success: Fortune 50 Bank

Profile:

- Large global bank serving 16M Canadian, American, and international customers
- 81,000 employees, worldwide

Challenge:

- Migration away from Oracle for cost, agility
- ~4 weeks to deploy new DBs on traditional infrastructure

Initial approach:

- DIY infrastructure to support containerized DBaaS
- 2 FTE for 18 months and still cannot operationalize the platform

Minimal Downtime

16x
Footprint reduction

23x
Faster deployment

$14M
TCO savings over 5 years

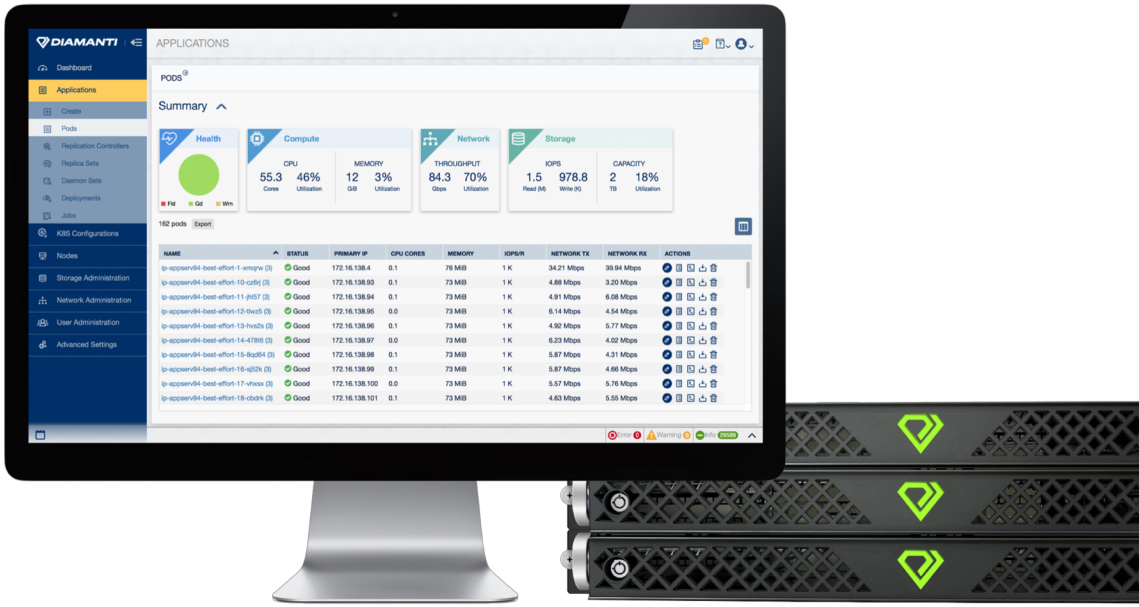
Solution:

- Installed and configured 3-node Diamanti D10 cluster
- Deployed PostgreSQL, MongoDB, and MariaDB on Diamanti in a day, vs. weeks
- Added addition Diamanti nodes dynamically with zero service disruption

DIAMANTI

Next Steps

- Visit www.diamanti.com
- Follow @diamanticom
- Email demo@diamanti.com for a briefing and live demo
- Stop by Diamanti Booth for more detailed discussion




DIAMANTI
Thank You