Cloud Native Technologies : The Building Blocks of Modern Database Software

Divine Odazie | 8th March 2025

Divine Odazie

- Ex Technology Evangelist at Severalnines

- Electrical/Electronics Engineer

- Certified Kubernetes Engineer (CKAD - CKA)

- Data on Kubernetes Ambassador

- I'm excited to speak at SCaLE 22x for the first time!

1. **Cloud Native Technologies**
2. **Introduction to Stateful Workloads on Kubernetes**
3. **Building a DBaaS on Kubernetes**
4. **Building a DBaaS with Kubernetes**
5. **Conclusion**

AGENDA

several9s

1

Cloud Native Technologies

Definitions

What is Cloud Native?

Cloud native practices empower organizations to develop, build, and deploy workloads in computing environments (public, private, hybrid cloud) to meet their organizational needs at scale in a programmatic and repeatable manner.

Cloud native technologies

Cloud native technologies and architectures typically consist of some combination of containers, service meshes, multi-tenancy, microservices, immutable infrastructure, serverless, and declarative APIs — this list is non-exhaustive.

Source: Cloud Native Computing Foundation (CNCF)

The Cloud Native Computing Foundation (CNCF)

The CNCF seeks to drive adoption of the Cloud Native paradigm by fostering and sustaining an ecosystem of open source, vendor-neutral projects.

It democratize state-of-the-art patterns to make these innovations accessible for everyone.

2

Introduction to Stateful Workloads on Kubernetes

What is a Stateful workload?

A workload that requires persistent data and must maintain its state across restarts or failures. Examples include:

- **Databases**: Store transactional data, user information, etc.
- **Game servers**: Persist player data, game progress, etc.
- **AI/ML applications:** Store training data, model checkpoints, etc.

Stateful workloads require meticulous orchestration!

Challenges Stateful workloads face

- Maintaining a consistent identity
  - Often for connection to other services
- High & Consistent Availability
  - Upgrades must be handled carefully to avoid disruption
  - Stateful workloads often have complex start and shutdown processes

Source: [Stateful Workloads in Kubernetes: A Deep Dive](#)

Kubernetes was originally designed for stateless workloads.

How is Kubernetes addressing these challenges?

Deployment

- **StatefulSets**
  - A StatefulSet ensures each of its pods retains a sticky, stable, unique identity.
- **Jobs**
  - Training datasets, models in ML
  - Backups

+ Other components

Lifecycle and Day 2 Management

- Custom Resource Definitions (CRD)
- Operators (How Kubernetes runs CRDs)
  - PostgreSQL,
  - Kafka,
  - MySQL,
  - Redis, etc.

How does Kubernetes handle storage?
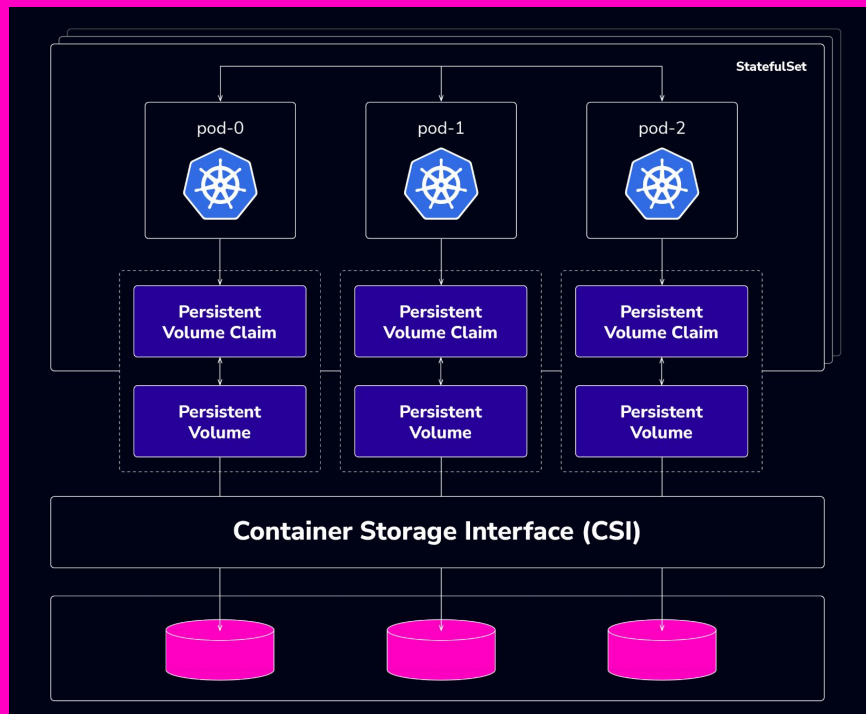
Persisting the Data

Persistent Volumes (PVs): Kubernetes objects that represent storage resources in your cluster.

Persistent Volume Claims (PVCs): a request for storage by a user. PVCs consume PV resources.

Container Storage Interface (CSI): is a standard for exposing arbitrary block and file storage systems to containerized workloads

Getting features like:

- Dynamic provisioning, resizing
- Snapshots, cloning etc.

Source: DoK Whitepaper

Putting it all together

- The user submits a configuration…
- The Operator processes the request…
- The StatefulSet creates pods in order…
- The PVCs are bound to Persistent Volumes…
- The database application within the pods…
- The Operator continuously monitors the cluster…

Source: [DoK Whitepaper](#)
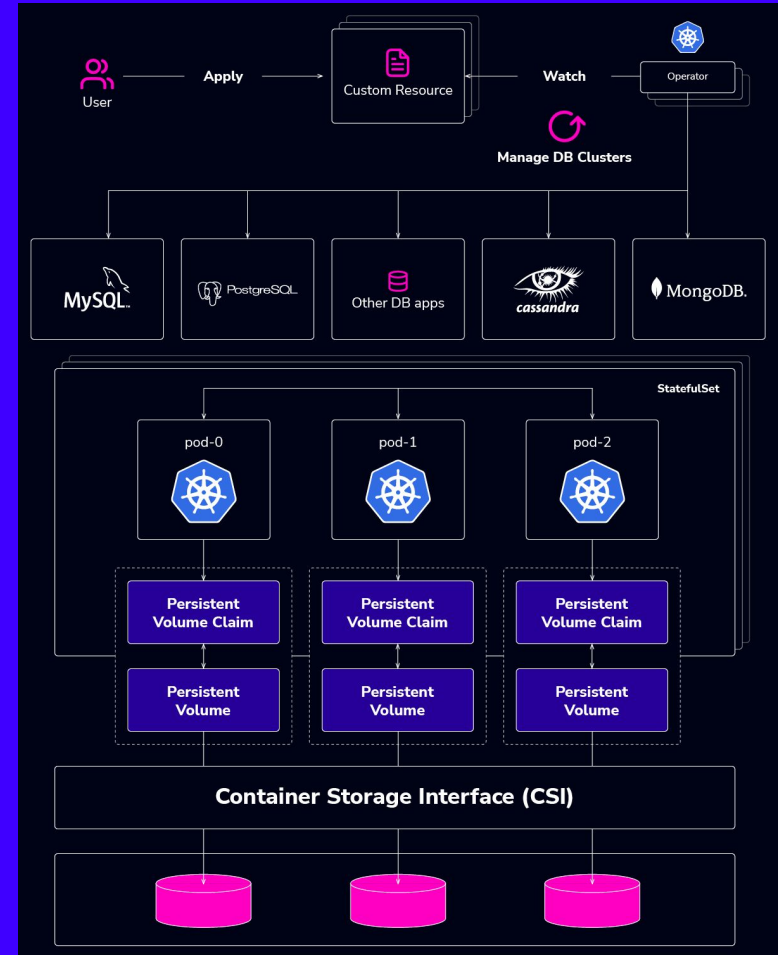
3 Building a DBaaS on Kubernetes

What does a DBaaS on Kubernetes need?

- Requires a system to manage users and databases
- Needs a method to apply schemas
- Must generate secrets with authentication data and distribute them to clusters and applications using those databases
- More importantly the ability to automate day 2 ops and standardize daily lifecycle ops
- Requires a user-friendly interface

You choose

Being Independent

- Involves procuring your own infrastructure, building custom software to handle each job within the DBaaS framework.

- Requires expertise and significant investment in time and resources.

Being Interdependent

- A middle ground between buying a solution and building everything from scratch.

- You would integrate your chosen infrastructure, code, and tools with off-the-shelf software to function as the control plane.

Interdependency with Cloud Native Technologies
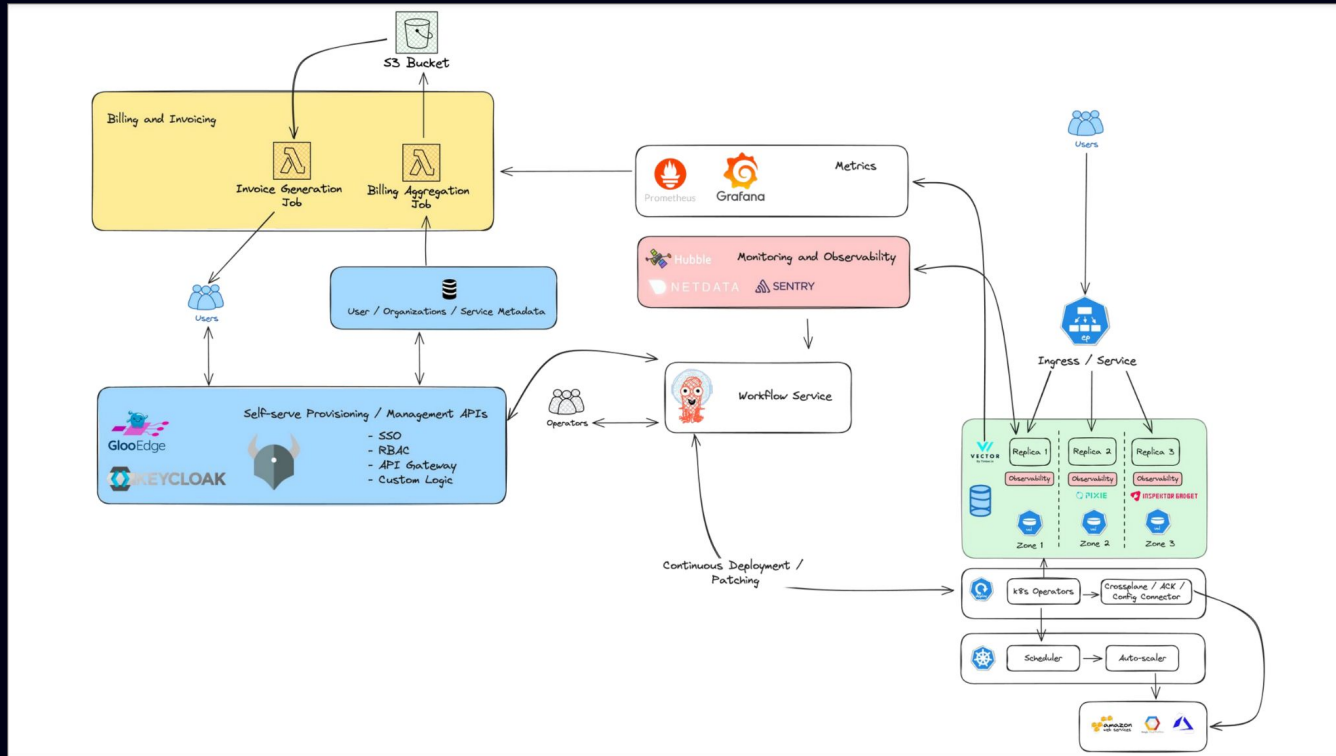
Do-It-Yourself DBaaS

Day 0
- Requirements
- Architecture
- Design

Day 1
- Installation
- Setup
- Configuration

Day 2
- Lights-on
- Maintenance
- Housekeeping
- Optimize

End
- Delete
- Clean up

Who built a DBaaS on Kubernetes?

Zalando

- PostgreSQL-as-a-Service on Kubernetes since as far back as 2018. Learn more in this talk.

- Deploying and managing 500+ PostgreSQL HA clusters.

- Creators of the Postgres operator and Patroni.

Shopify

- KateSQL – Shopify's custom-built Database-as-a-Service platform on GKE.

- Supporting hundreds of production MySQL instances.

- Kubernetes for efficient resource utilization and scaling.

Not everyone wants to put their databases on Kubernetes…
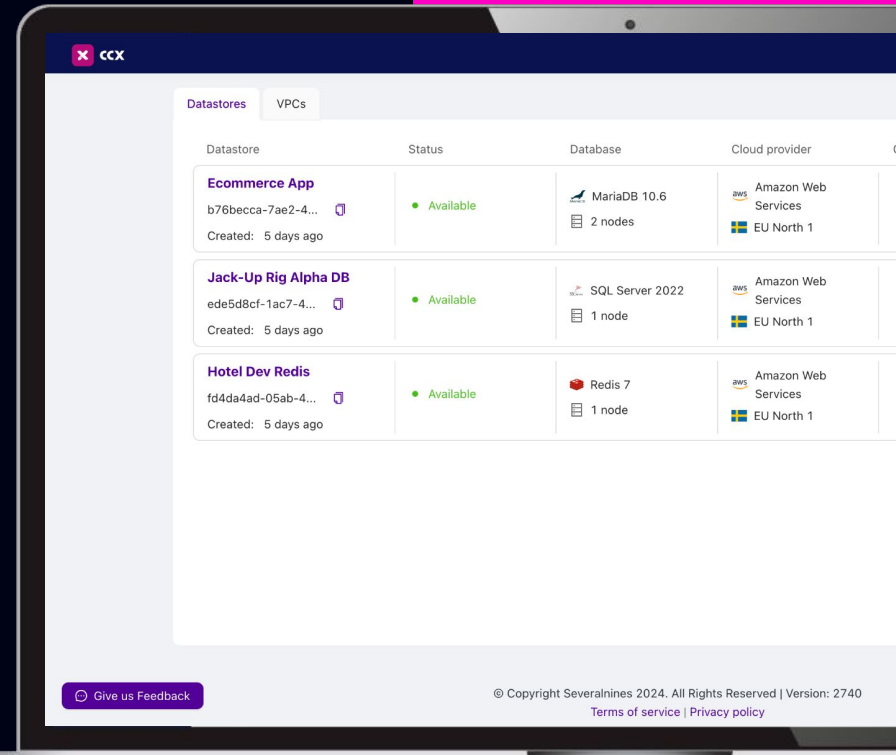
YET!

4

Building a DBaaS with Kubernetes

Introducing **CCX**

CCX is a drop-in, white-label DBaaS that allows CSPs to offer managed open-source databases to their end users, using their own infrastructure.
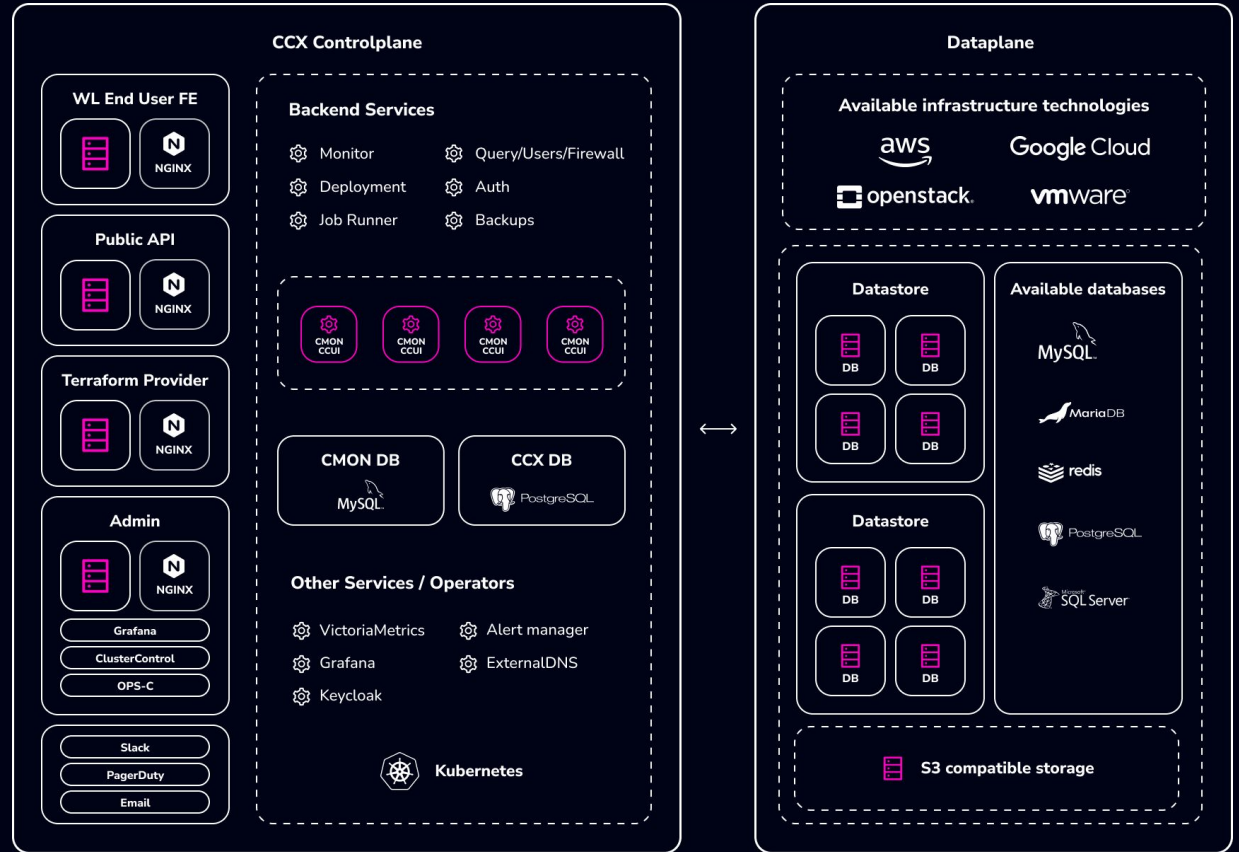
Details:

- Kubernetes-based
- Built on the industry-leading ClusterControl (CC) platform
- Prioritizes data sovereignty through a loosely coupled architecture and regulatory compliance
- Quick to set up and integrate
- Provides 24/7 support and high SLA
- Offers an easy-to-use ops and end-user interface.

several9s | Database ops, your way

Architecture & Overview

CCX Cloud Service Provider Architecture

CCX Controlplane

WL End User FE

NGINX

Public API

NGINX

Terraform Provider

NGINX

Admin

NGINX

Grafana

ClusterControl

OPS-C

Slack

PagerDuty

Email

Backend Services

- ⚙ Monitor
- ⚙ Deployment
- ⚙ Job Runner
- ⚙ Query/Users/Firewall
- ⚙ Auth
- ⚙ Backups

CMON CCUI · CMON CCUI · CMON CCUI · CMON CCUI

CMON DB
MySQL

CCX DB
PostgreSQL

Other Services / Operators

- ⚙ VictoriaMetrics
- ⚙ Grafana
- ⚙ Keycloak
- ⚙ Alert manager
- ⚙ ExternalDNS

Kubernetes

Dataplane

Available infrastructure technologies

aws · Google Cloud · openstack · vmware

Datastore

DB · DB
DB · DB

Available databases

MySQL

MariaDB

redis

PostgreSQL

SQL Server

Datastore

DB · DB
DB · DB

S3 compatible storage

CCX Installation in 3 Steps

1

Deploy CCX Dependencies

- CCX core DB operators
- NATS
- ExternalDNS
- Ingress controller

2

Deploy CCX Core Services

- CCXUI
- Authentication service, etc.
- DB Orchestration service

3

Deploy CCX Observability Stack

- Grafana
- Alert manager
- Prometheus node exporter

CCX Documentation | Public On Github | ArtifactHUB

several9s | Database ops, your way

Who built a DBaaS with **CCX**?

Elastx: A Swedish Cloud Provider **|| Lintasarta Cloudeka:**
An Indonesian Cloud Provider

- They implemented a full-fledged, premium service quickly instead of committing a ton of R&D resources and time to build one from scratch.

- They are now offering DBaaS to thousands of end users.

- They fully customized the platform to meet their specific needs.

> "As a Cloud Platform Partner, Severalnines has helped us confidently step into the DBaaS space, making our solutions catalog more extensive, and helping our customers be more successful. However, it's not just about simplifying database automation, but ensuring that our customers can focus more on their core business by choosing a compliant provider and a robust platform that offers more than just IaaS."

elastx   **Joakim Öhman,** CEO of Elastx

# Does this make the role of a traditional DBA defunct?

Absolutely Not!

Wrapping up

✔ The journey of database deployment is far from over.

✔ Cloud-native technologies, particularly Kubernetes, offer powerful tools for managing Stateful workloads like databases.

✔ If you are interested in a drop in DBaaS solution, scan the QR code.

several9s | Database ops, your way

several9s

THANK YOU!
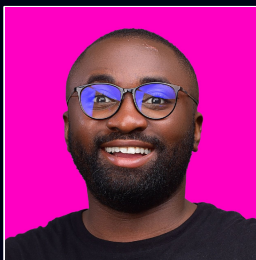
Tech Evangelist
Divine Odazie

𝕏 @_Odazie

Check out our podcast:
Sovereign DBaaS Decoded

▶ @Severalnines

● Sovereign DBaaS Decoded