PERCONA .CONNECT

Empowering Platform Engineering with Percona Everest: Cloud Native Database Platform for MongoD PostgreSQL, and MySQL

Takis Stathopoulos
Enterprise Architect, Percona

Agenda

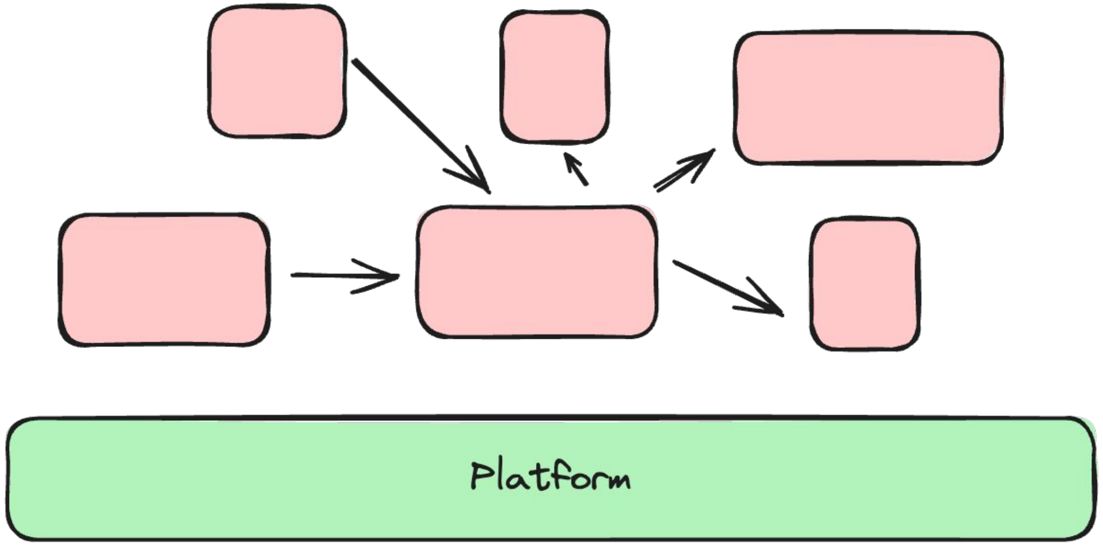
1. Platforms and Platform Engineering
2. Percona Everest
    a. What is it about?
    b. Architecture
    c. Demo
3. Roadmap
4. Get Involved!

Q1: DevOps anyone?

Q2: Platform Engineering anyone?

Platforms

Platform? What do you mean?

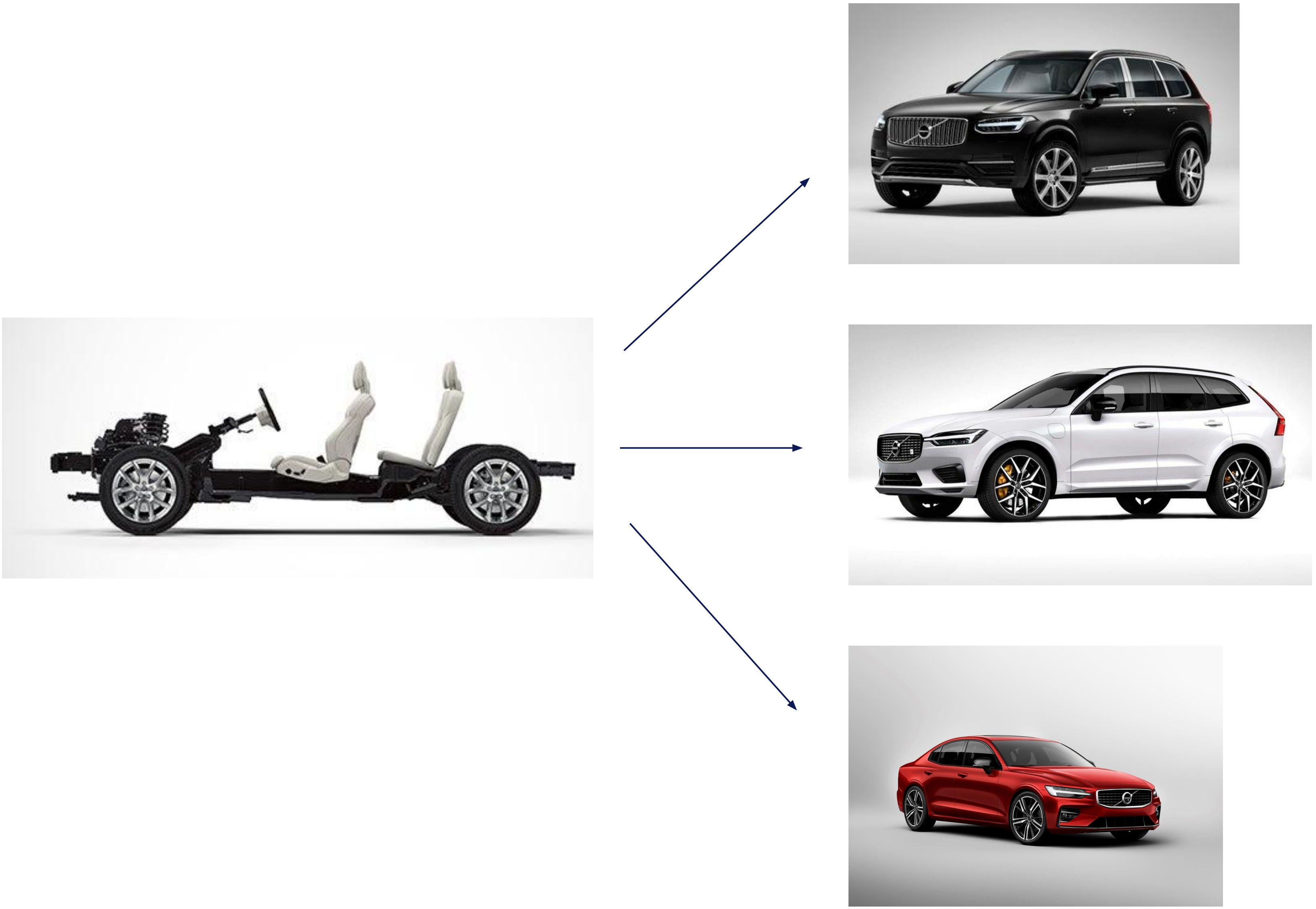


“A platform is a group of technologies that are used as a base upon which other applications, processes or technologies are developed.”

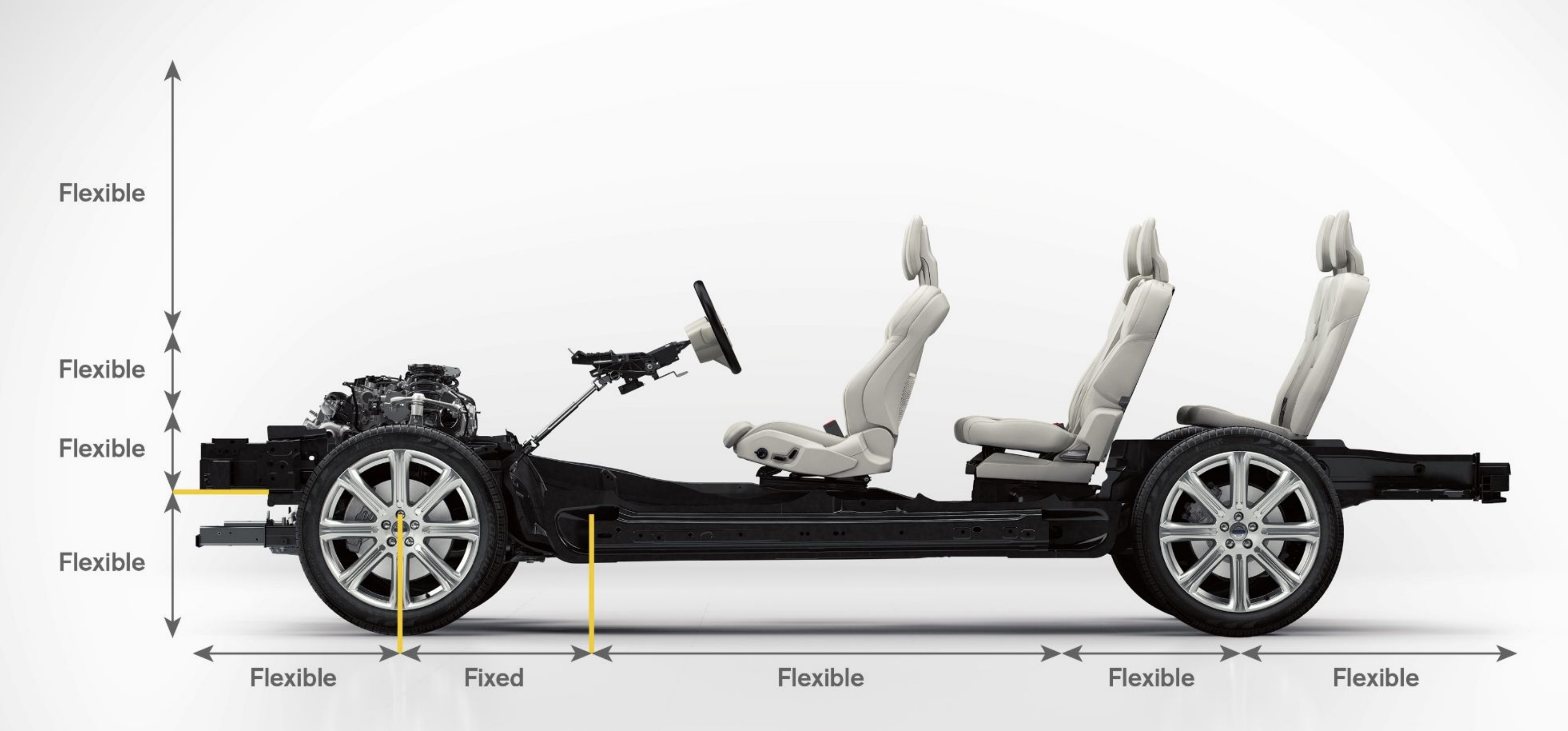
Source: https://www.techopedia.com/definition/3411/platform-computing

Automotive platforms

- High investment
- Long R&D
- Standardization



Enable (not limit) innovation



Enable (not limit) innovation

amazon web services
salesforce
Azure
ORACLE
IBM Cloud

Do standards limit or enable innovation?

“Platforms are a means of centralizing expertise, while decentralizing innovation to the customer or user.”

-Peter Gillard-Moss, ThoughtWorks

Platform Engineering

Platform Engineering is a **practice** that builds on internal platforms, enabling **self-service** capabilities for software engineering teams following a **cloud-native approach**.

A platform encompasses a **set of tools, services, and infrastructure** that enables developers to build, test, deploy, and monitor software applications. These platforms are known as **internal developer platforms (IDPs)** and are **tailored** specifically to an organization's needs, demands, and goals.

Simply put, in a cloud-native world, Platform Engineering is the application of DevOps principles at scale.

Source: https://www.dynatrace.com/monitoring/solutions/platform-engineering/

Why Platform Engineering?

The forever end-goal

Improve developer experience and productivity

Faster time to market

Today's challenges

Increasing complexity of software architectures

Operational scalability

Future outlook

"By 2026, 80% of large software engineering organizations will establish platform engineering teams **as internal providers of reusable services, components and tools for application delivery.**

Platform engineering will ultimately solve the central problem of cooperation between software developers and operators."

Source: *https://www.gartner.com/en/articles/what-is-platform-engineering*

How?

Self-service capabilities with automated infrastructure operations

Components

- Developer Portals
- API-driven infrastructures
- Reusable components
- Cloud-Nativeness
- Enforcement of standards

Source: https://www.gartner.com/en/articles/what-is-platform-engineering

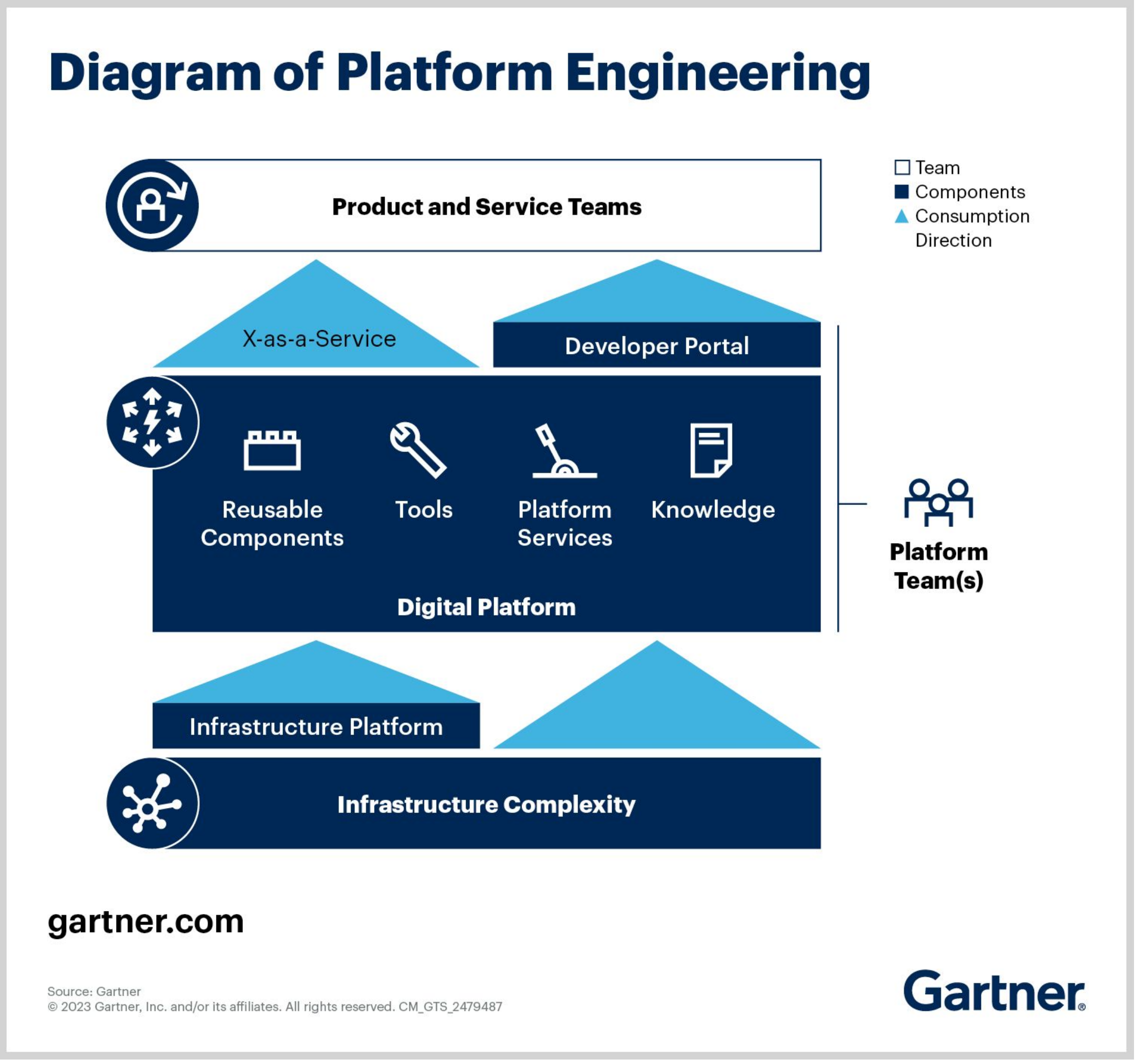


Diagram of Platform Engineering

gartner.com

Source: Gartner
© 2023 Gartner, Inc. and/or its affiliates. All rights reserved. CM_GTS_2479487

Gartner.

Value

DevOps principles at scale in a Cloud Native manner

- Drive standardization and consistency
- Utilize resources more efficiently
- Enhance developer productivity
- Scale DevOps approach
- Security and compliance “enforcement”

PERCONA .CONNECT

Establishing a (data) platform in your organization

	Build	DB-as-a-Service	Buy
Cost	Headcount, opportunity, and time	High overhead at scale	Cost of product, often high
Time to market	(Very) high	Now-ish	Now-ish
Internal resources	Headcount and skills	Almost none	Significantly less - just adopt
Control & compatibility	Total control	Limitations	Limitations
Maintenance and support	On you	Done by vendor	Done by vendor
Competitive advantage	Not your core business	Focus on your core business	Focus on your core business
Vendor lock-in	Non-existent	Real (specific platform)	Real (and painful)
Business Continuity	Succession planning for silos of knowledge	Easy to achieve	Vendor-dependent

	Build	DB-as-a-Service	Buy
Cost	Headcount, opportunity, and time	High overhead at scale	Cost of product, often high
Time to market	(Very) high	Now-ish	Now-ish
Internal resources	Headcount and skills	Almost none	Significantly less - just adopt
Control & compatibility	Total control	Limitations	Limitations
Maintenance and support	On you	Done by vendor	Done by vendor
Competitive advantage	Not your core business	Focus on your core business	Focus on your core business
Vendor lock-in	Non-existent	Real (specific platform)	Real (and painful)
Business Continuity	Succession planning for silos of knowledge	Easy to achieve	Vendor-dependent

	Build	DB-as-a-Service	Buy
Cost	Headcount, opportunity, and time	High overhead at scale	Cost of product, often high
Time to market	(Very) high	Now-ish	Now-ish
Internal resources	Headcount and skills	Almost none	Significantly less - just adopt
Control & compatibility	Total control	Limitations	Limitations
Maintenance and support	On you	Done by vendor	Done by vendor
Competitive advantage	Not your core business	Focus on your core business	Focus on your core business
Vendor lock-in	Non-existent	Real (specific platform)	Real (and painful)
Business Continuity	Succession planning for silos of knowledge	Easy to achieve	Vendor-dependent

	Build	DB-as-a-Service	Buy
Cost	**Headcount, opportunity, and time**	**High overhead at scale**	**Cost of product, often high**
Time to market	**(Very) high**	Now-ish	Now-ish
Internal resources	**Headcount and skills**	Almost none	Significantly less - just adopt
Control & compatibility	Total control	**Limitations**	**Limitations**
Maintenance and support	**On you**	Done by vendor	Done by vendor
Competitive advantage	**Not your core business**	Focus on your core business	Focus on your core business
Vendor lock-in	Non-existent	**Real (specific platform)**	**Real (and painful)**
Business Continuity	**Succession planning for silos of knowledge**	Easy to achieve	**Vendor-dependent**

	Build	DB-as-a-Service	Buy
Cost	Headcount, opportunity, and time	High overhead at scale	Cost of product, often high
Time to market	(Very) high	**Now-ish**	**Now-ish**
Internal resources	Headcount and skills	**Almost none**	Significantly less - just adopt
Control & compatibility	**Total control**	Limitations	Limitations
Maintenance and support	On you	**Done by vendor**	**Done by vendor**
Competitive advantage	Not your core business	**Focus on your core business**	**Focus on your core business**
Vendor lock-in	**Non-existent**	Real (specific platform)	Real (and painful)
Business Continuity	Succession planning for silos of knowledge	**Easy to achieve**	Vendor-dependent

Build (vs DBaaS) vs Buy:
is there anything in between?

Percona Everest:
The Cloud Native Database Platform

Percona Everest - open source, cloud-native database platform that helps organizations and their IT teams regain control over data access, configuration flexibility, and costs of DBaaS.

1. **DevOps/Platform Engineers** deploy and maintain Private DBaaS on K8s infrastructure
2. **Database Engineers** define standardized deployments and assist with challenges
3. **App Teams/Devs** provision and maintain highly performant database clusters through self-service portal/APIs

All the database heavy lifting is taken care of and codified, but you keep things under control.

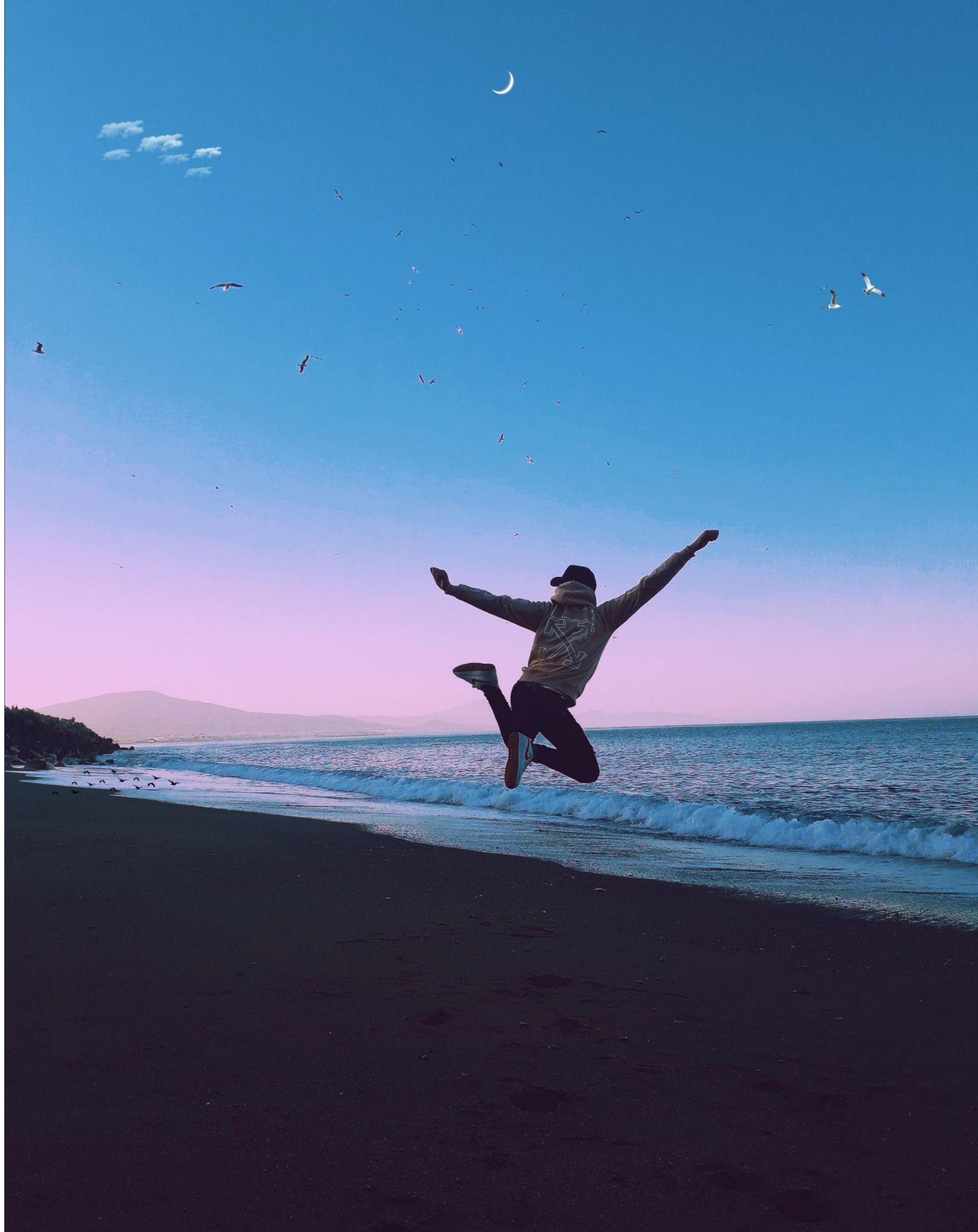
PERCONA

We believe (we are biased):

- Open source is a better choice
- Investing into open source pays back
- You can achieve your goals with OSS

All software we provide is open source.

MySQL PostgreSQL mongoDB



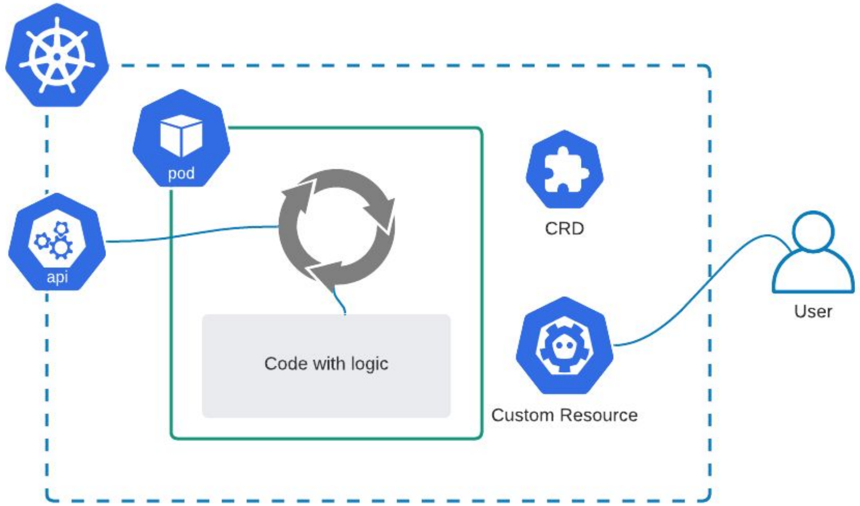
Cloud Native Definition

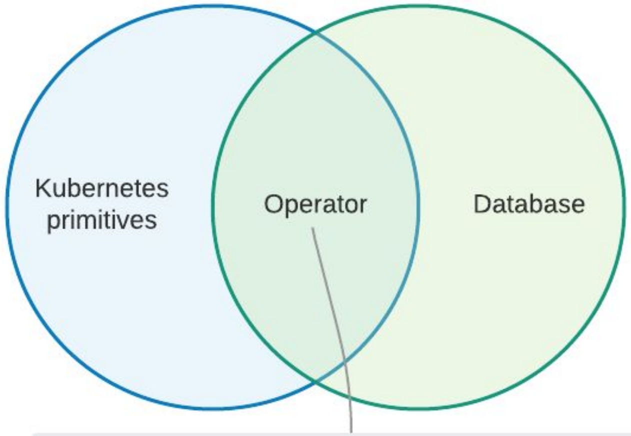
Source: https://github.com/cncf/toc/blob/main/DEFINITION.md

Cloud native technologies empower organizations to build and run scalable applications in modern, dynamic environments such as public, private, and hybrid clouds. Containers, service meshes, microservices, immutable infrastructure, and declarative APIs exemplify this approach.

These techniques enable loosely coupled systems that are resilient, manageable, and observable. Combined with robust automation, they allow engineers to make high-impact changes frequently and predictably with minimal toil.

Database heavy lifting is on us





```
spec:
  image: percona/percona-server-mongodb:4.4.6-8
  replsets:
   - name: rs0
      size: 3
    sharding:
       mongos:
          size: 3
        configsvrReplSet:
           size: 3
      backups:
         ...
```

Operators are software extensions to Kubernetes that make use of Custom Resources Definitions (CRDs) to manage applications and their components.

Operators abstract and automate Database - level concepts to K8s primitive transparently for the end - user

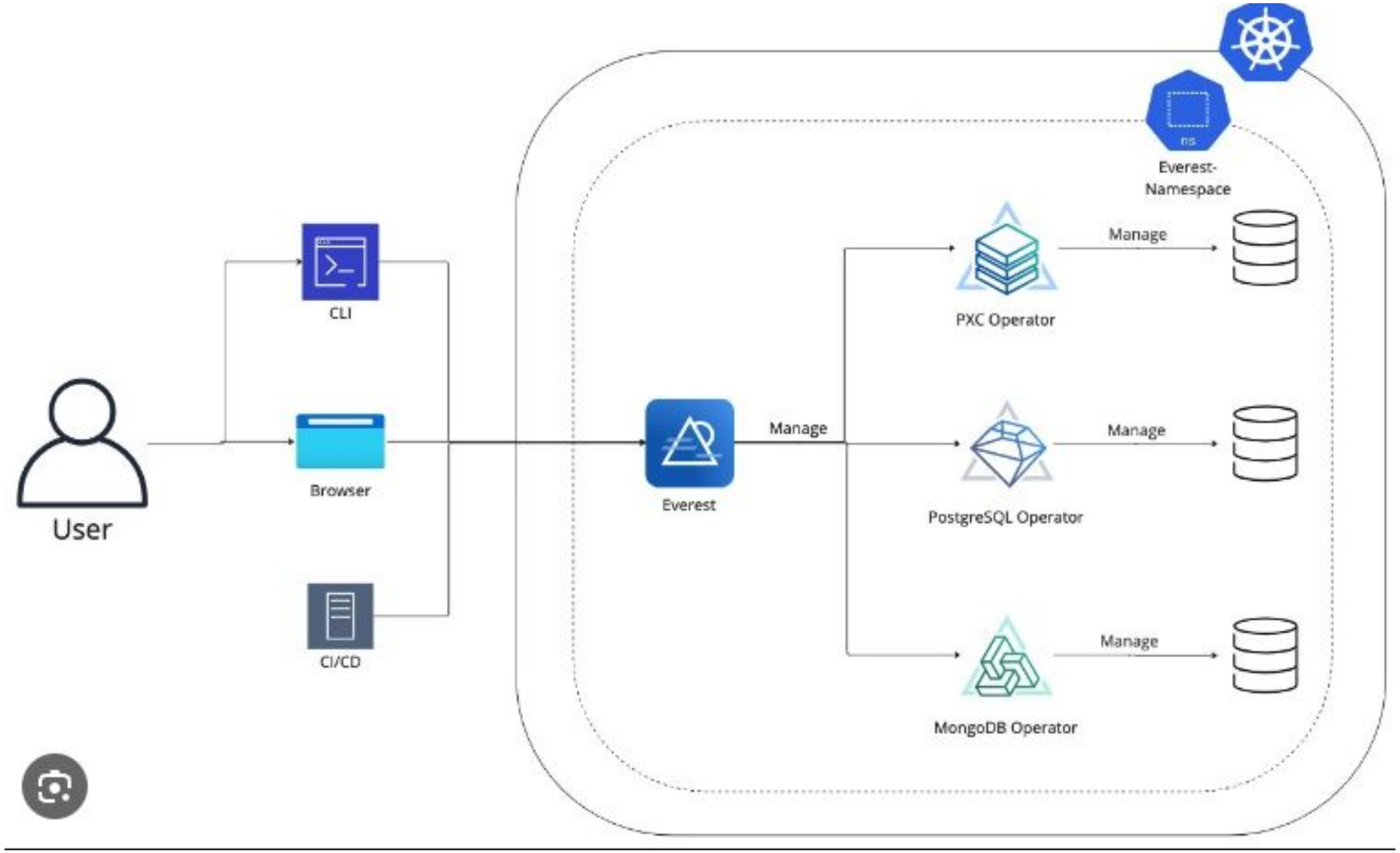
Everest Capabilities

- **Multi-Cloud** and **On-Premise** Deployments: AWS, GCP, OpenShift (any K8s)
- **Multi-database support**: MySQL, MongoDB, or PostgreSQL (option to extend)
- Private deployments and **customizable database setup**: control plane and data plane hosted privately by you
- **Horizontal and Vertical Scaling** - with K8s autoscaling

- Recovery capabilities: scheduled, on-demand **backups, PITR**, **restores**
- Centralized configuration management
- Activity tracking and audit logs
- Native integration with **Percona Monitoring and Management**
- Policies definition and enforcement
- **CLI management tool and REST APIs**

Full-stack expert support with **15 min SLA** available from Percona! (optional)

How Everest works

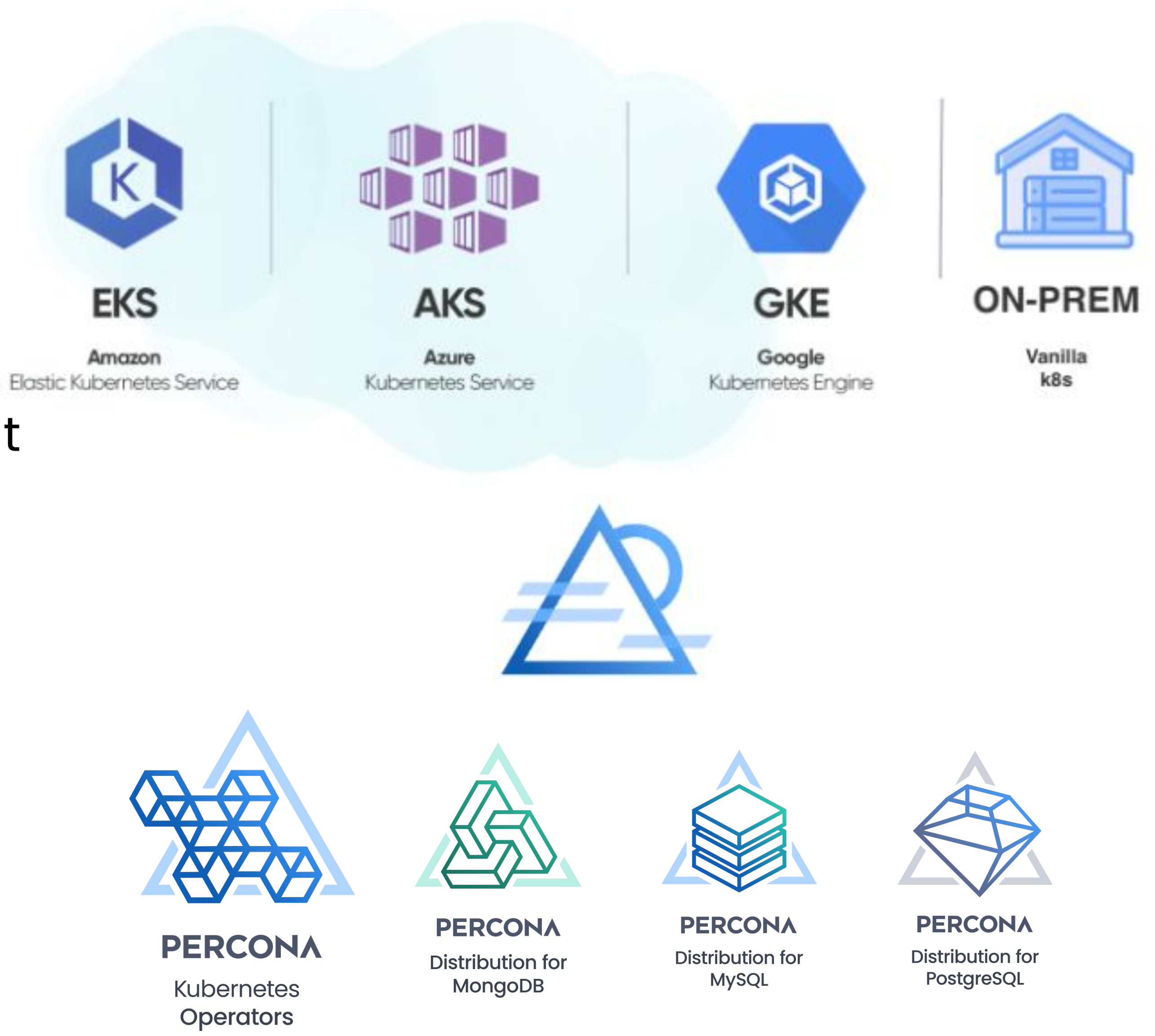


Databases

The most popular open source databases deployed privately within your cloud or on-premise environment

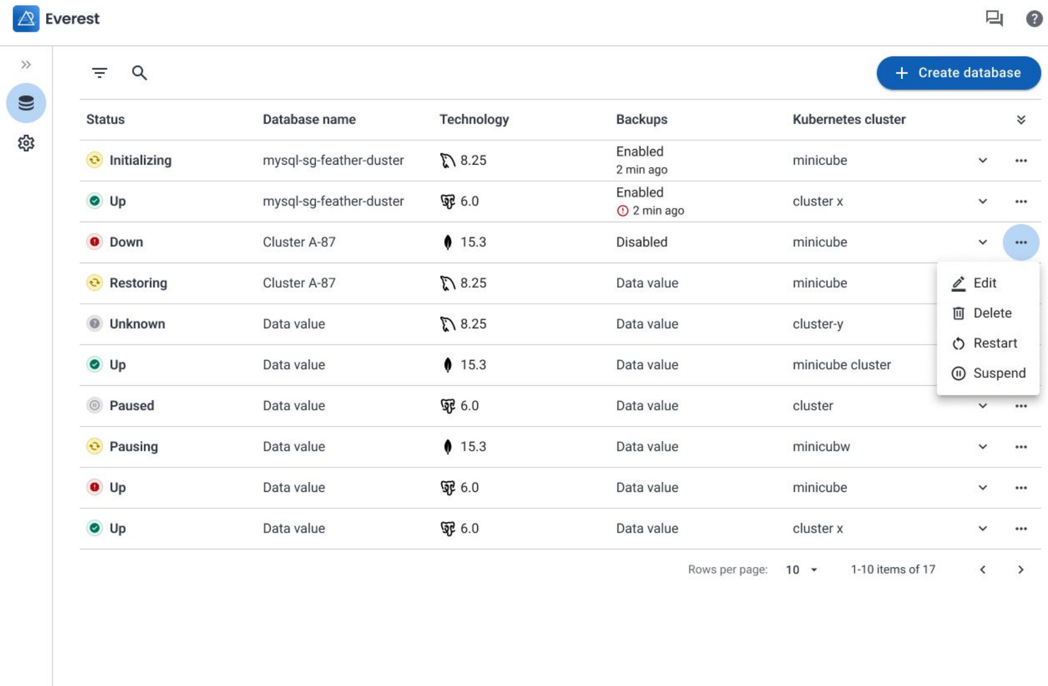
Potential community contributions:

- **Valley**
- **Clickhouse**



Automation

Backups, failover, horizontal and vertical scaling with Percona expertise productized

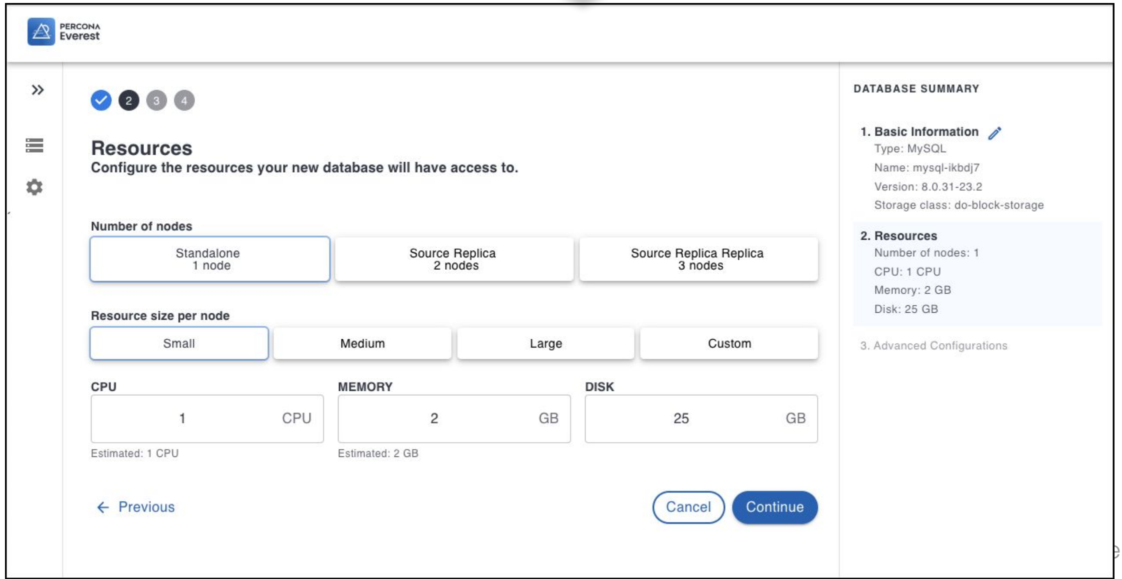


Everest

+ Create database

Status	Database name	Technology	Backups	Kubernetes cluster
Initializing	mysql-sg-feather-duster	8.25	Enabled 2 min ago	minicube
Up	mysql-sg-feather-duster	6.0	Enabled 2 min ago	cluster x
Down	Cluster A-87	15.3	Disabled	minicube
Restoring	Cluster A-87	8.25	Data value	minicube
Unknown	Data value	8.25	Data value	cluster-y
Up	Data value	15.3	Data value	minicube cluster
Paused	Data value	6.0	Data value	cluster
Pausing	Data value	15.3	Data value	minicubw
Up	Data value	6.0	Data value	minicube
Up	Data value	6.0	Data value	cluster x

Rows per page: 10 1-10 items of 17



PERCONA Everest

Resources
Configure the resources your new database will have access to.

Number of nodes
Standalone 1 node | Source Replica 2 nodes | Source Replica Replica 3 nodes

Resource size per node
Small | Medium | Large | Custom

CPU: 1 CPU — Estimated: 1 CPU
MEMORY: 2 GB — Estimated: 2 GB
DISK: 25 GB

DATABASE SUMMARY

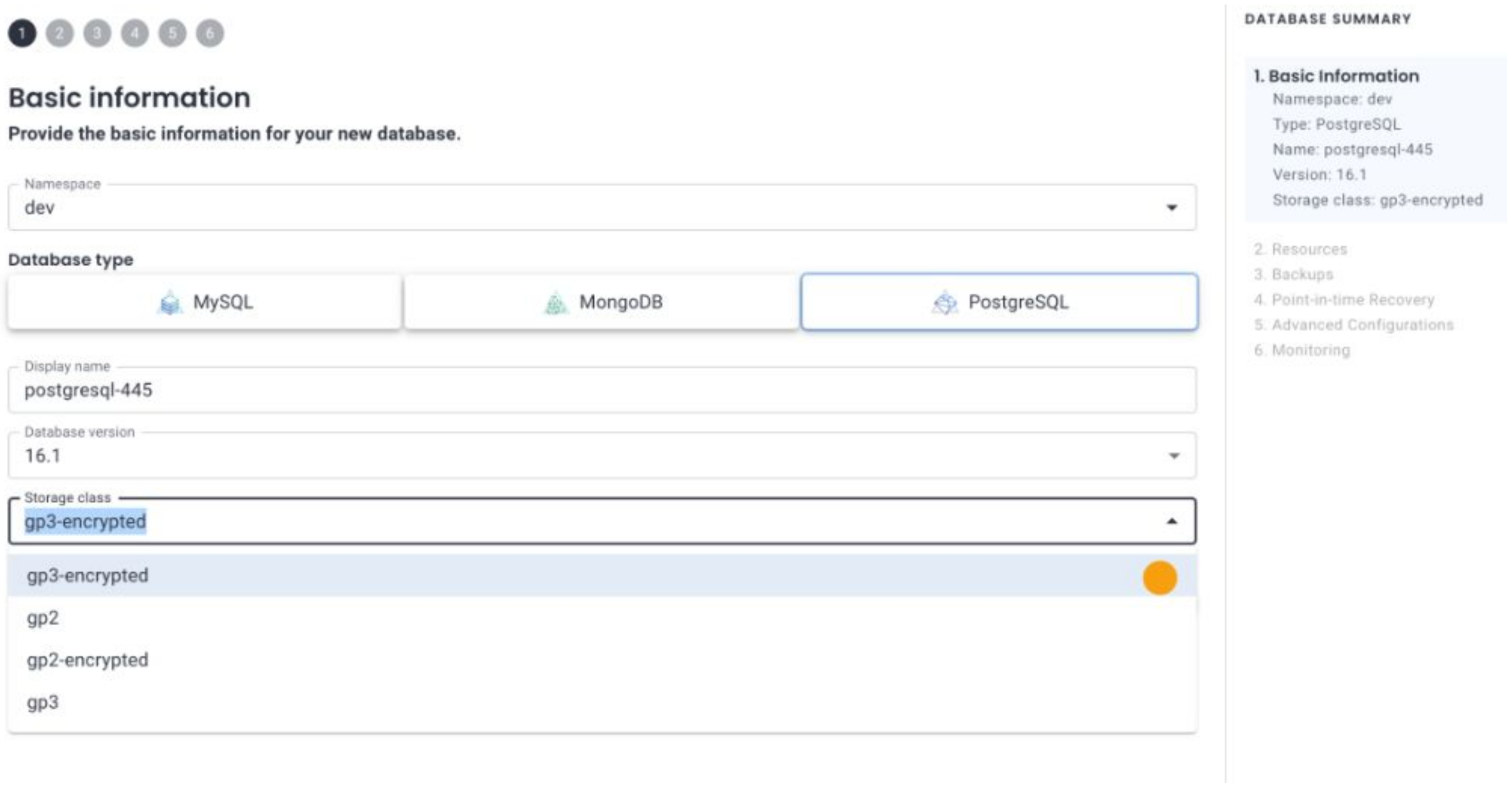
1. Basic Information
Type: MySQL
Name: mysql-ikbdj7
Version: 8.0.31-23.2
Storage class: do-block-storage

2. Resources
Number of nodes: 1
CPU: 1 CPU
Memory: 2 GB
Disk: 25 GB

3. Advanced Configurations

Customization

Default or advanced configuration options for ease and complete control over storage and database parameters



Basic information

Provide the basic information for your new database.

Namespace: dev

Database type: MySQL | MongoDB | PostgreSQL

Display name: postgresql-445

Database version: 16.1

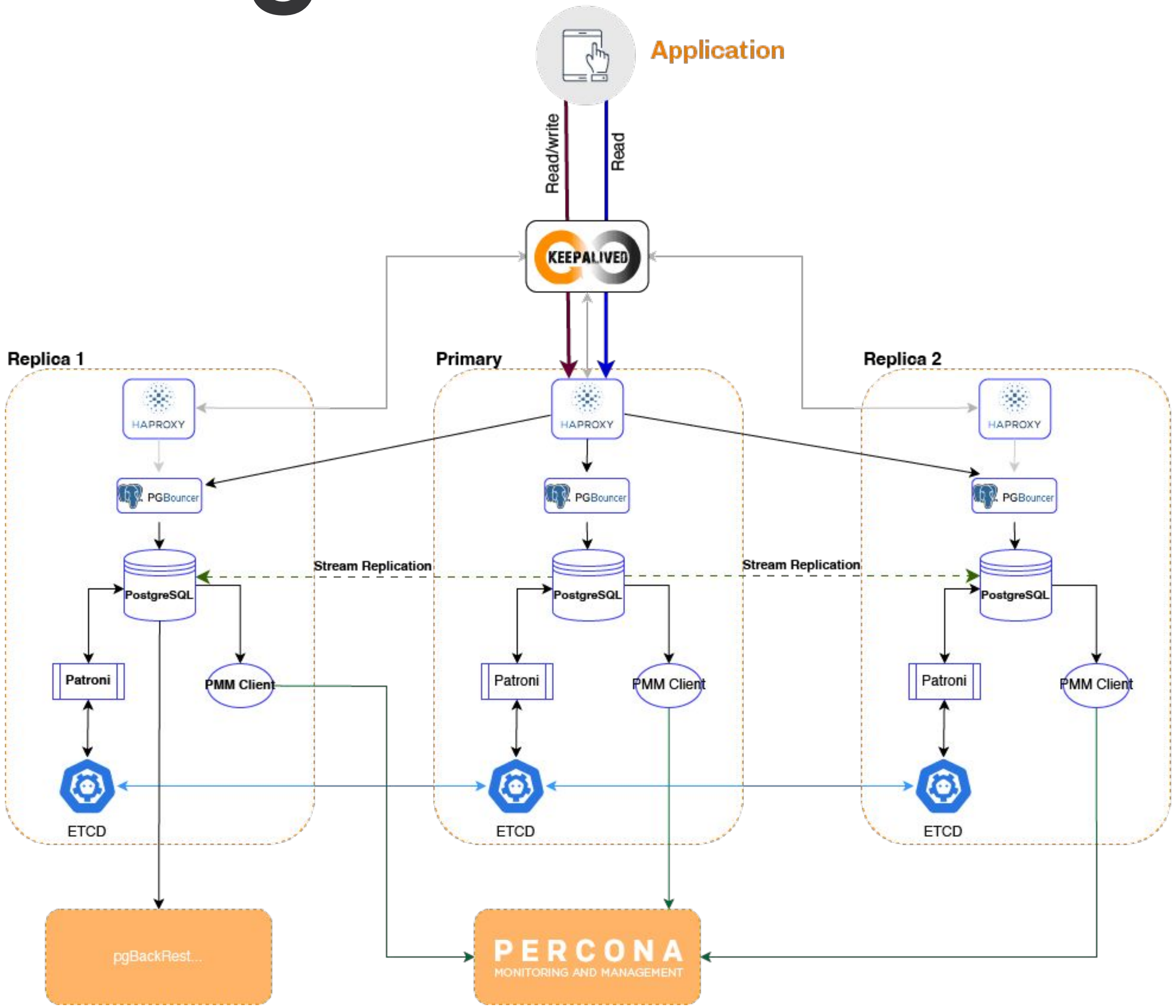
Storage class: gp3-encrypted

- gp3-encrypted
- gp2
- gp2-encrypted
- gp3

DATABASE SUMMARY

1. Basic Information
   - Namespace: dev
   - Type: PostgreSQL
   - Name: postgresql-445
   - Version: 16.1
   - Storage class: gp3-encrypted
2. Resources
3. Backups
4. Point-in-time Recovery
5. Advanced Configurations
6. Monitoring

Indicative PostgreSQL Architecture



Demo Time

Try it yourselves:

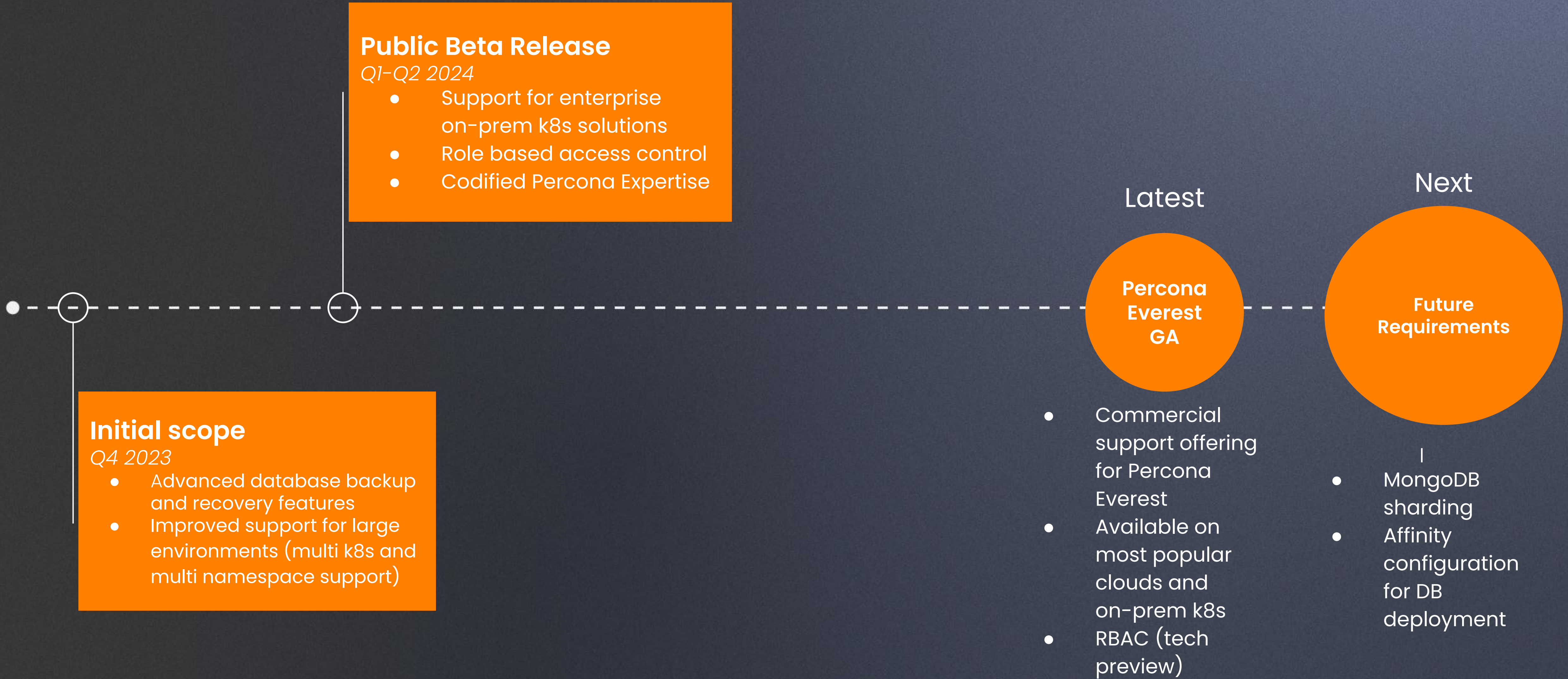
https://learn.percona.com/percona-everest-demo

Percona Everest: Benefits

- Database best practices - codified
- Ready to integrate with your Platform
- Free from vendor lock-in
- Self-service for everyone
- Open and customizable
- Support from Percona and Community

Best of “Build”, “DB-as-a-Service” and “Buy”

Everest milestones



Initial scope
Q4 2023
- Advanced database backup and recovery features
- Improved support for large environments (multi k8s and multi namespace support)

Public Beta Release
Q1-Q2 2024
- Support for enterprise on-prem k8s solutions
- Role based access control
- Codified Percona Expertise

Latest

Percona Everest GA
- Commercial support offering for Percona Everest
- Available on most popular clouds and on-prem k8s
- RBAC (tech preview)

Next

Future Requirements
- MongoDB sharding
- Affinity configuration for DB deployment

Try it and get involved!

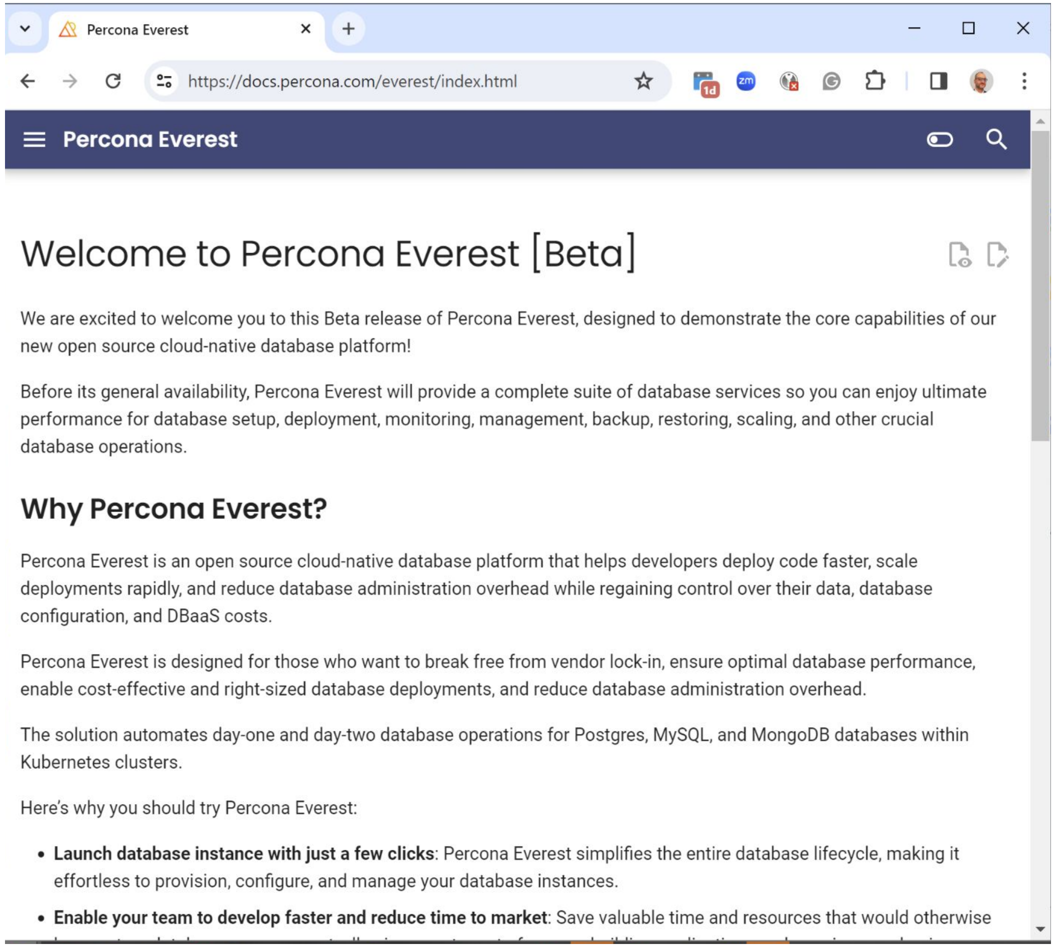
https://percona.community/projects/everest/
https://docs.percona.com/everest

Try it!

- Build on rock-solid DB operators
- Lots of features
- Provide feedback and drive development direction

Contribute!

- Everest is open source project
- Contributions are welcomed and encouraged
- What about additional databases?



Welcome to Percona Everest [Beta]

We are excited to welcome you to this Beta release of Percona Everest, designed to demonstrate the core capabilities of our new open source cloud-native database platform!

Before its general availability, Percona Everest will provide a complete suite of database services so you can enjoy ultimate performance for database setup, deployment, monitoring, management, backup, restoring, scaling, and other crucial database operations.

Why Percona Everest?

Percona Everest is an open source cloud-native database platform that helps developers deploy code faster, scale deployments rapidly, and reduce database administration overhead while regaining control over their data, database configuration, and DBaaS costs.

Percona Everest is designed for those who want to break free from vendor lock-in, ensure optimal database performance, enable cost-effective and right-sized database deployments, and reduce database administration overhead.

The solution automates day-one and day-two database operations for Postgres, MySQL, and MongoDB databases within Kubernetes clusters.

Here's why you should try Percona Everest:

- **Launch database instance with just a few clicks**: Percona Everest simplifies the entire database lifecycle, making it effortless to provision, configure, and manage your database instances.
- **Enable your team to develop faster and reduce time to market**: Save valuable time and resources that would otherwise

PERCONA
.CONNECT

Thank you

Feedback? Would you like to learn more?

https://www.linkedin.com/in/pgstathopoulos/