SAOUG
CONFERENCE 2025

PURESTORAGE®

Revolutionising Database Deployment with the Oracle Database Operator for Kubernetes

HELLO
AFRICA

Ron Ekins

Director, Field Solutions Architecture
EMEA & LATAM

Outsurance In the House Party

Oracle ACE Director

Ron Ekins
Director Field Solution Architecture, EMEA & LATAM

- **Email: ron@purestorage.com**
- **X: @RonEkins**
- **BlueSky: @ronekins.com**
- **Blog: https://ronekins.com**
- **GitHub: https://github.com/raekins**

ORACLE | Partner

Oracle ACE Director

PURESTORAGE®

Positioned Highest in Execution, Furthest in Vision

2025 Gartner® Magic Quadrant™ for Enterprise Storage Platforms

Gartner

Oracle ACE

The Oracle ACE Program

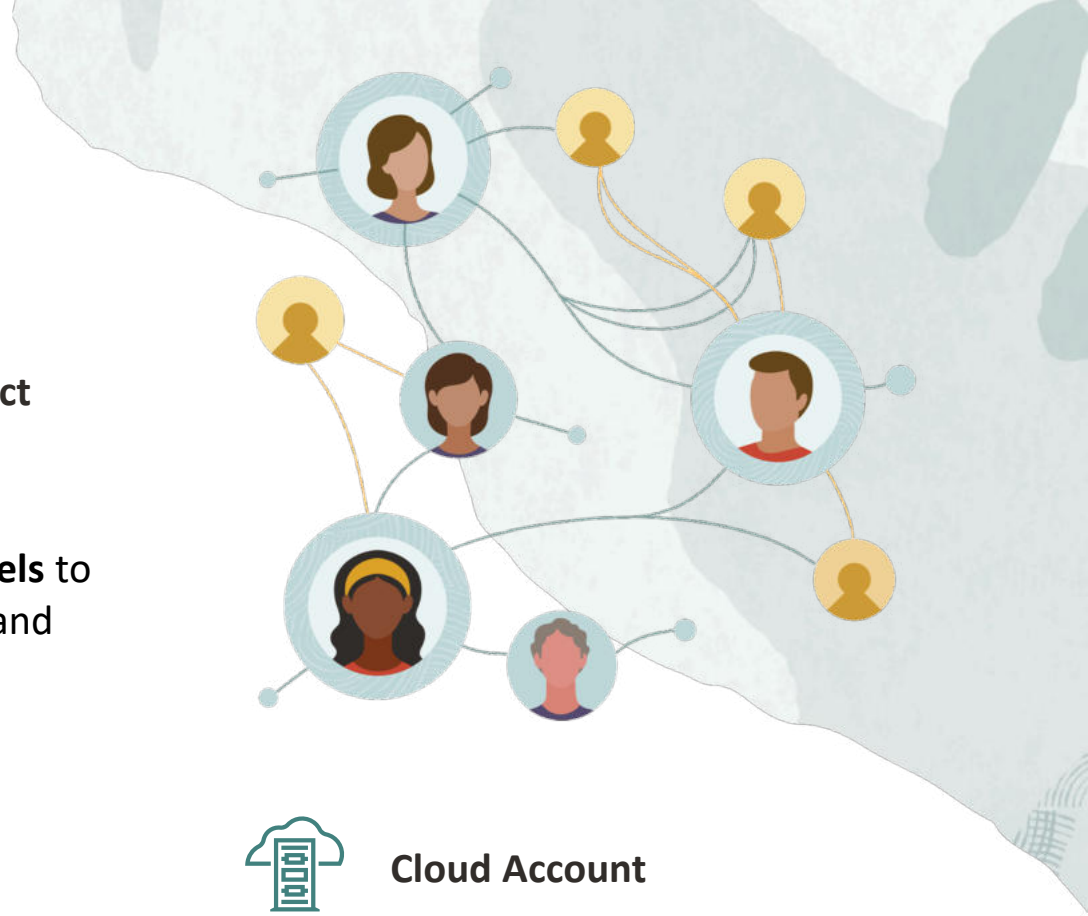
600+ technical experts helping peers globally

- The Oracle ACE Program recognizes and rewards community members for their technical and community contributions to the Oracle community
- 3 membership levels: Director, Pro, and Associate
- Nominate yourself or a colleague at ace.oracle.com/nominate
- Learn more at ace.oracle.com

aceprogram_ww@oracle.com Facebook.com/OracleACEs @oracleace Oracle ACE Program Group

Oracle ACE

ACE Member Benefits



Key Benefits

Cool **swag***, Digital awards for social media, Oracle **CloudWorld pass***, & more

Direct Access to Product Management

Multiple **direct communication channels** to product management and fellow ACEs

Exclusive Content

Exclusive **monthly virtual meetings** with product development teams + engaging guest speakers

Networking

In-person & virtual networking opportunities for ACEs to **connect with product development** and each other.

Cloud Account

$5k USD Cloud account*

Travel Support

ACE Directors are eligible for travel support to give presentations or lead workshops at conferences globally

* for Pro and Director levels

Agenda

Introduction to Oracle and Containerisation

The Oracle Database Kubernetes Operator

Oracle 23ai on OKE

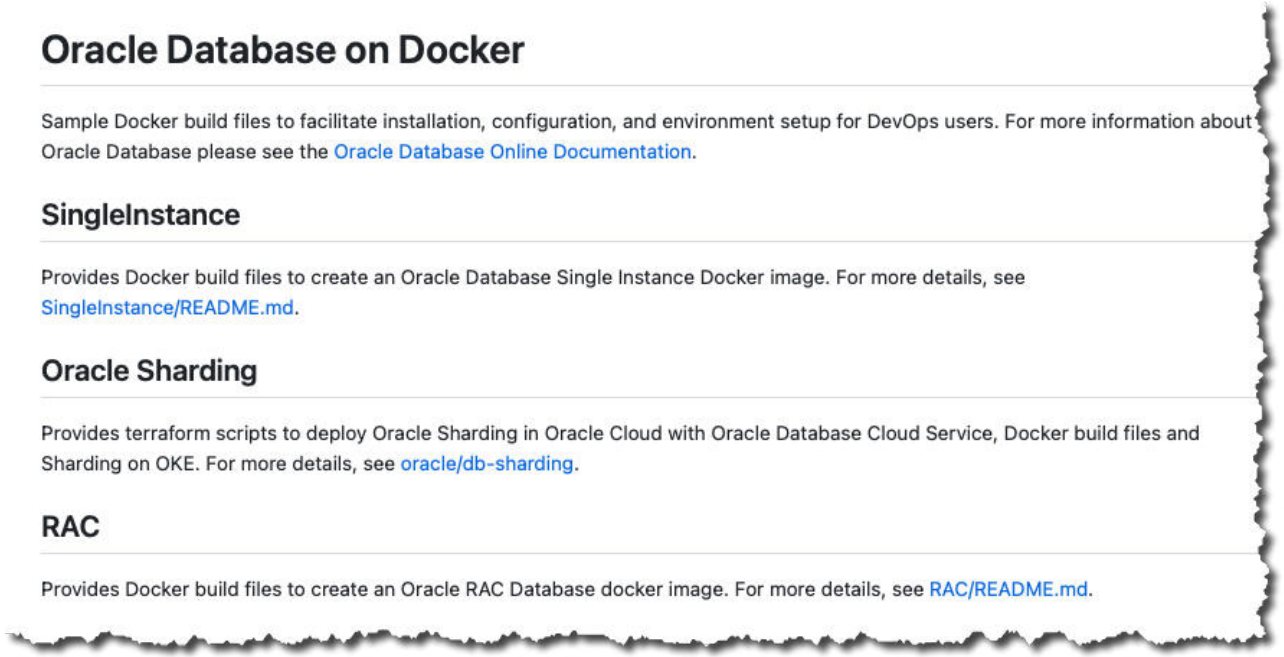
PURESTORAGE®

Oracle and Containerisation

Oracle Database on GitHub

Oracle provided Docker Build Scripts

- 26ai (23.9.0) FREE
- 21c (21.3.0) EE and SE2
- 19c (19.3.0) EE and SE2
- 18c (18.4.0) XE
- 18c (18.3.0) EE and SE2
- 12c Release 2 (12.2.0.2) EE and SE2
- 12c Release 1 (12.1.0.2) EE and SE2
- 11g Release 2 (11.2.0.2) XE



Oracle Database on Docker

Sample Docker build files to facilitate installation, configuration, and environment setup for DevOps users. For more information about Oracle Database please see the Oracle Database Online Documentation.

SingleInstance

Provides Docker build files to create an Oracle Database Single Instance Docker image. For more details, see SingleInstance/README.md.

Oracle Sharding

Provides terraform scripts to deploy Oracle Sharding in Oracle Cloud with Oracle Database Cloud Service, Docker build files and Sharding on OKE. For more details, see oracle/db-sharding.

RAC

Provides Docker build files to create an Oracle RAC Database docker image. For more details, see RAC/README.md.

https://github.com/oracle/docker-images/tree/main/OracleDatabase

Oracle Container Registry

https://container-registry.oracle.com/

Oracle Support for Database Running in Docker

MOS Doc ID 2216342.1

Oracle Support for Database Running on Docker (Doc ID 2216342.1)

APPLIES TO:

Oracle Database - Enterprise Edition - Version 12.1.0.2 and later
Oracle Database - Standard Edition - Version 12.1.0.2 and later
Linux x86-64

PURPOSE

Clarify Oracle's support of Oracle Database running on Docker

SCOPE

For customers running Oracle Database (single instance configuration) in Docker containerized environments.

DETAILS

Oracle plans to certify the latest versions of Oracle Database to run in Docker containers which are built and supported with Oracle Linux as the host.

Additionally, Oracle does support customers running Oracle Database (single instance) in Docker containers running on Oracle Linux 7 with UEK4 (and later) or Red Hat Enterprise Linux 7. Docker binaries are available in the Addons channel for Oracle Linux. Details on Installation can be found in Chapter 2 of the Oracle Linux 7 Docker Users Guide. Other details on support for Docker on Oracle Linux can be found in Support for Docker Running on Oracle Linux (Doc ID 1921163.1).

For information on Oracle RAC on Docker, see Note 2488326.1 Oracle RAC on Docker - Released Versions and Known Issues

If you are planing to deploy Oracle database on Oracle Linux 8, you need to use Podman with UEKR6. Podman binaries are available in the Oracle Linux 8 on ULN and the Oracle Linux yum server. You can install all of these packages by installing the container-tools module using the dnf command. Podman in Oracle Linux 8 is a drop-in replacement for Oracle Container Runtime for Docker. However, Docker-produced images will continue to work with Open Container Initiative compatible container runtimes.

Additional information which might be helpful:

- Support for Docker Running on Oracle Linux (Doc ID 1921163.1)
- Information on Docker for Oracle Linux can be found at Oracle Linux Cloud Native Environment
- Certified Virtualization and Partitioning Technologies for Oracle Database and RAC Product Releases
- Creating an Oracle Database Docker Image (blog post)

Oracle RAC Database Support Now includes 21.3.0/19.16.0

MOS Doc ID 2488326.1

APPLIES TO:

Oracle Database - Enterprise Edition - Version 12.2.0.1 and later
Linux OS - Version Oracle Linux 7.5 with Unbreakable Enterprise Kernel [4.14.35] to Oracle Linux 7.9 with Unbreakable Enterprise Kernel [5.4.17] [Release OL7U5 to OL7U9]
Linux x86-64

PURPOSE

This article is about

SCOPE

The contents of this

DETAILS

Docker lets you pac
virtually anywhere.
overcomes the limit
Docker provides a r

[redacted]

Released Versions	Available on GitHub	Available on OCR	Oracle Documentation Link	Support
21.3.0	Yes	Yes	https://docs.oracle.com/en/database/oracle/oracle-database/21/racdk/index.html	Production
19.16.0	Yes	No	https://docs.oracle.com/en/database/oracle/oracle-database/19/racdk/	Production
19.3.0	Yes	Yes	None	Dev and Test
18.3.0	Yes	Yes	None	Dev and Test
12.2.0	Yes	Yes	None	Dev and Test

Time to try containerised Oracle Database 26ai

PURESTORAGE®

An Introduction to Kubernetes Operators

What is a Kubernetes Operator

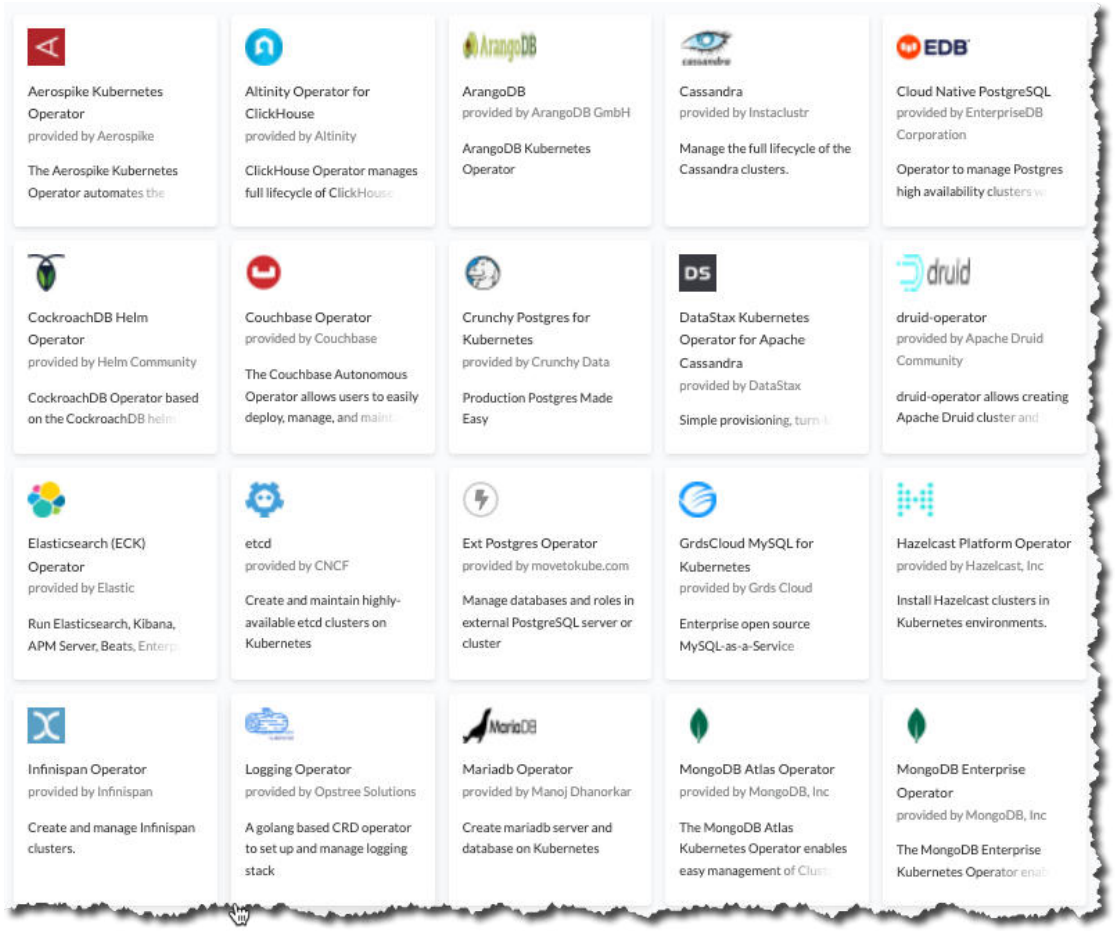
Kubernetes Operators provide a way to package, deploy, and manage a Kubernetes application.

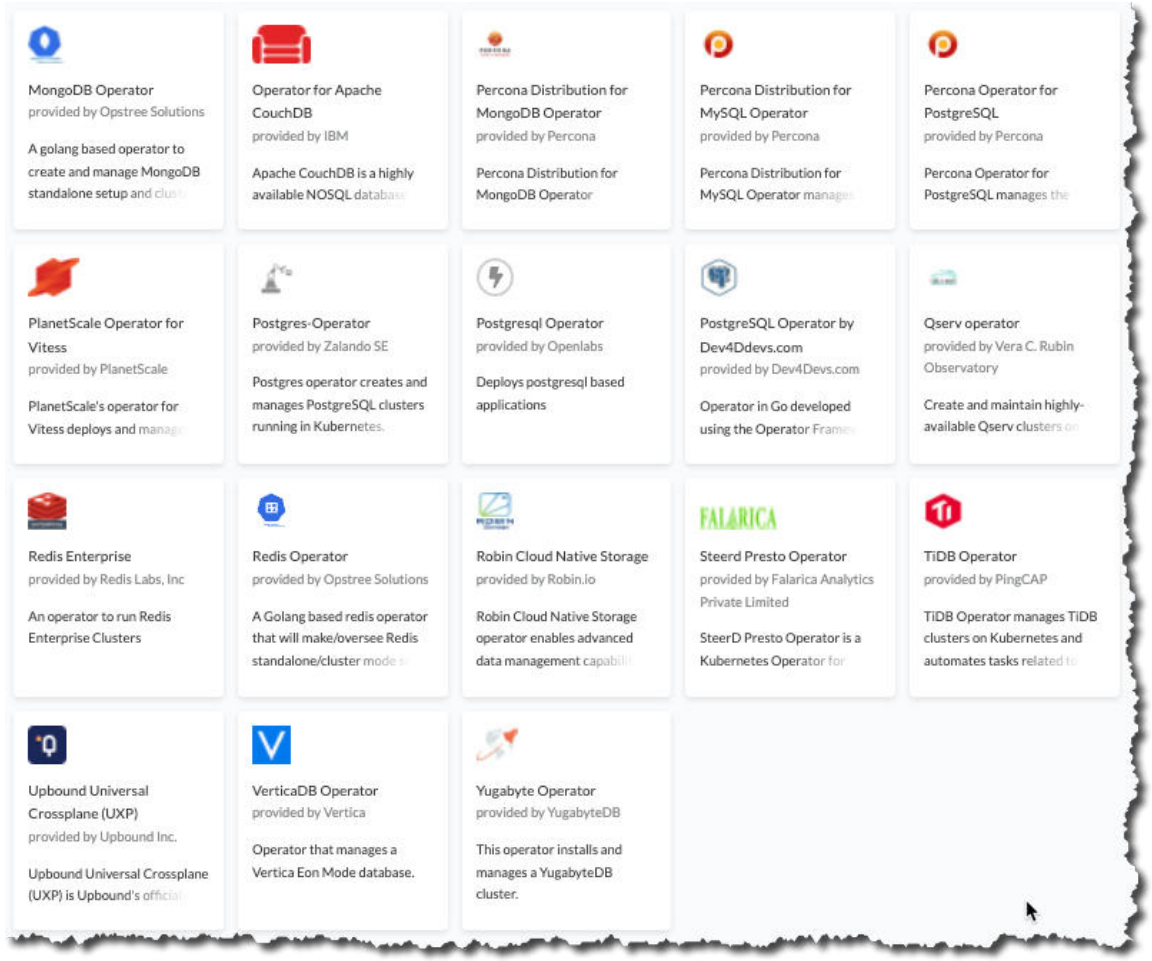
Kubernetes Operators help in building cloud-native applications by automating deployment, scaling, backup and location independent restores.

Kubernetes Operators provide a way of delivering consistent deployments across cloud vendors and on-premises.

OperatorHub.IO

52 Databases Including Oracle





https://operatorhub.io/?category=Database

Multiple Providers offering Different Features

Crunchy Postgres for Kubernetes

Production Postgres Made Easy

Home › Crunchy Postgres for Kubernetes

Crunchy Postgres for Kubernetes

PGO, the Postgres Operator from Crunchy Data, gives you a **declarative Postgres** solution that automatically

Designed for your GitOps workflows, it is easy to get started with Postgres on Kubernetes with PGO. Within cluster complete with high availability, disaster recovery, and monitoring, all over secure TLS communication cluster to tailor it to your workload!

With conveniences like cloning Postgres clusters to using rolling updates to roll out disruptive changes with at every stage of your release pipeline. Built for resiliency and uptime, PGO will keep your desired Postgres i

PGO is developed with many years of production experience in automating Postgres management on Kuber keep your data always available.

- **PostgreSQL Cluster Provisioning**: Create, Scale, & Delete PostgreSQL clusters with ease, while fully c
- **High-Availability**: Safe, automated failover backed by a distributed consensus based high-availability configure how aggressive this can be! Failed primaries automatically heal, allowing for faster recovery and set your backup retention policy
- **Disaster Recovery**: Backups and restores leverage the open source pgBackRest utility and includes su efficient delta restores. Set how long you want your backups retained for. Works great with very large
- **Monitoring**: Track the health of your PostgreSQL clusters using the open source pgMonitor library.
- **Clone**: Create new clusters from your existing clusters or backups with efficient data cloning.
- **TLS**: All connections are over TLS. You can also bring your own TLS infrastructure if you do not want to
- **Connection Pooling**: Advanced connection pooling support using pgBouncer.
- **Affinity and Tolerations**: Have your PostgreSQL clusters deployed to Kubernetes Nodes of your prefe and more rules to customize your deployment topology!
- **Full Customizability**: Crunchy PostgreSQL for Kubernetes makes it easy to get your own PostgreSQL- deployments, including:
  - Choose the resources for your Postgres cluster: container resources and storage size. Resize at
  - Use your own container image repository, including support `imagePullSecrets` and private
  - Customize your PostgreSQL configuration

and much more!

Percona Operator for Postg

Percona Operator for PostgreSQL manages the lifecycle of Percona PostgreSQL

Home › Percona Operator for PostgreSQL

Percona Operator for PostgreSQL

Percona is Cloud Native

Percona Operator for PostgreSQL automates and simplifies deploying and managing open source Po based on Postgres Operator developed by Crunchy Data. Whether you need to get a simple Postgre tolerant cluster in production, or are running your own database-as-a-service, the Operator provide

Consult the documentation on the Percona Operator for PostgreSQL for complete details on capabil

Supported Features

- **PostgreSQL Cluster Provisioning** Create, Scale, & Delete PostgreSQL clusters with ease, whil
- **High Availability**

  Safe, automated failover backed by a distributed consensus based high-availability solution. U enforced this can be. Failed primaries automatically heal, allowing for faster recovery time. Su across multiple Kubernetes clusters.
- **Disaster Recovery**

  Backups and restores leverage the open source pgBackRest utility and includes support for fu restores. Set how long you want your backups retained for. Works great with very large datab
- **Communication Security**

  Secure communication between your applications and data servers by enabling TLS for your P connections to use TLS.
- **Monitoring**

  Track the health of your PostgreSQL clusters with Percona Monitoring and Management (PM
- **PostgreSQL User Management**

  Quickly add and remove users from your PostgreSQL clusters with powerful commands. Man

Cloud Native PostgreSQL

Operator to manage Postgres high availability clusters with a primary/standby architecture.

Home › Cloud Native PostgreSQL

Cloud Native PostgreSQL

Cloud Native PostgreSQL is an operator designed by EnterpriseDB to manage PostgreSQL workloads on any supported Kubernetes cluster running in private, public, or hybrid cloud environments, including OpenShift. Cloud Native PostgreSQL adheres to DevOps principles and concepts such as declarative configuration and immutable infrastructure.

It defines a new Kubernetes resource called "Cluster" representing a PostgreSQL cluster made up of a single primary and an optional number of replicas that co-exist in a chosen Kubernetes namespace for High Availability and offloading of read-only queries.

Applications that reside in the same Kubernetes cluster can access the PostgreSQL database using a service which is solely managed by the operator, without having to worry about changes of the primary role following a failover or a switchover. Applications that reside outside the Kubernetes cluster need to configure an Ingress object t expose the service via TCP.

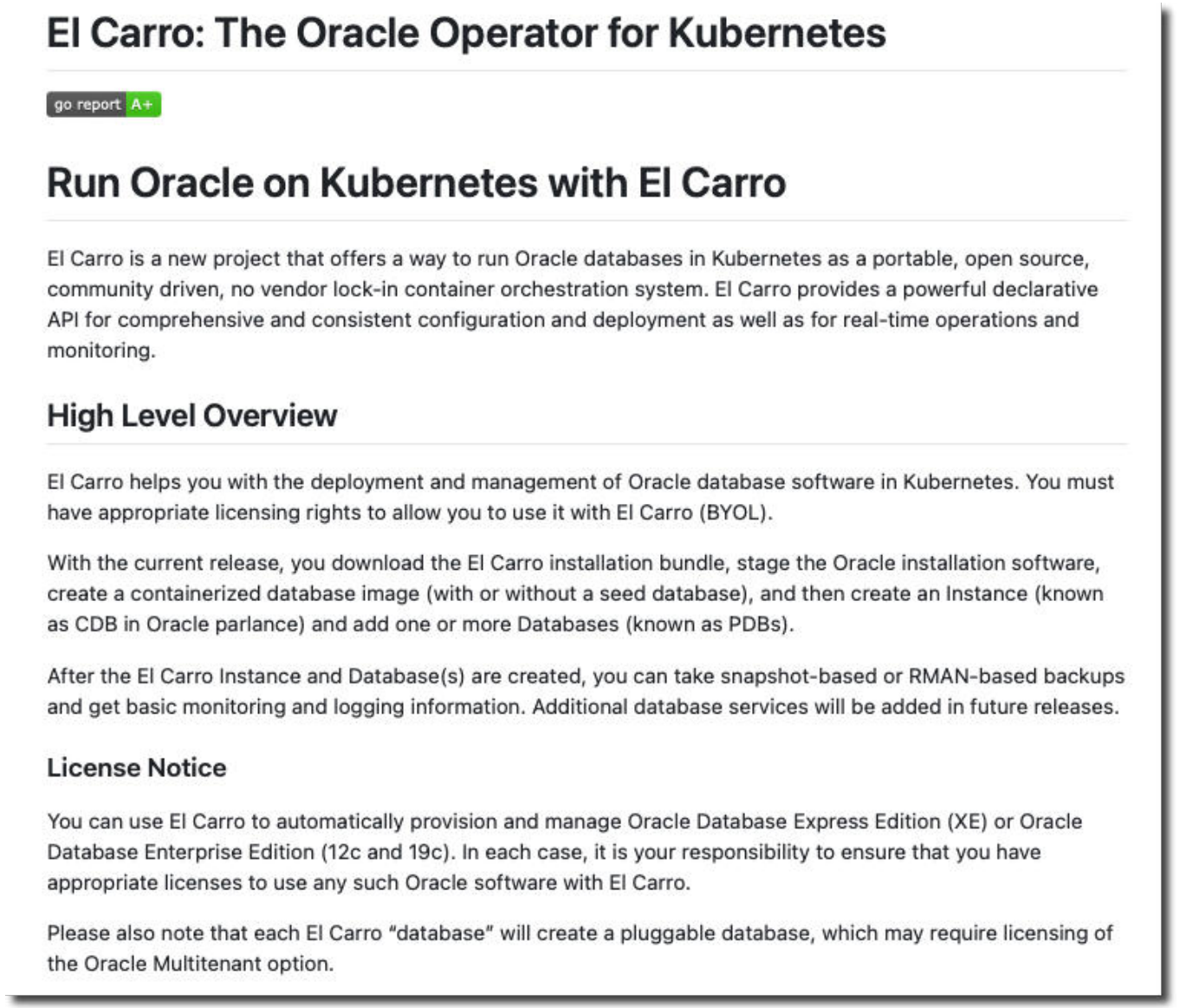
Cloud Native PostgreSQL works with PostgreSQL (14, 13, 12, 11 and 10) and EDB Postgres Advanced (14, 13, 12, 11 and 10), and it is available under the EnterpriseDB Limited Use License. You can evaluate Cloud Native PostgreSQL for free. You need a valid license key to use Cloud Native PostgreSQL in production.

Main features

- Direct integration with Kubernetes API server for High Availability, without requiring an external tool
- Self-Healing capability, through:
  - failover of the primary instance by promoting the most aligned replica
  - automated recreation of a replica
- Planned switchover of the primary instance by promoting a selected replica
- Scale up/down capabilities
- Definition of an arbitrary number of instances (minimum 1 - one primary server)
- Definition of the *read-write* service, to connect your applications to the only primary server of the cluster
- Definition of the *read-only* service, to connect your applications to any of the instances for reading workloads
- Declarative management of PostgreSQL configuration, including certain popular Postgres extensions through the cluster `spec: pg_audit, auto_explain, and pg_stat_statements`
- Support for Local Persistent Volumes with PVC templates
- Reuse of Persistent Volumes storage in Pods
- Rolling updates for PostgreSQL minor versions
- In-place or rolling updates for operator upgrades
- TLS connections and client certificate authentication
- Support for custom TLS certificates (including integration with cert-manager)
- Continuous backup to an object store (AWS S3 and S3-compatible, Azure Blob Storage, and Google Cloud Storage)
- Backup retention po

Google El Carro

- Google Developed Oracle Database Operator for Kubernetes ‘El Carro’
- Designed for Google Kubernetes Engine (GKE)
- Supports:
  - Oracle 19c EE
  - Oracle 18c XE
  - Oracle 12c EE



El Carro: The Oracle Operator for Kubernetes

go report A+

Run Oracle on Kubernetes with El Carro

El Carro is a new project that offers a way to run Oracle databases in Kubernetes as a portable, open source, community driven, no vendor lock-in container orchestration system. El Carro provides a powerful declarative API for comprehensive and consistent configuration and deployment as well as for real-time operations and monitoring.

High Level Overview

El Carro helps you with the deployment and management of Oracle database software in Kubernetes. You must have appropriate licensing rights to allow you to use it with El Carro (BYOL).

With the current release, you download the El Carro installation bundle, stage the Oracle installation software, create a containerized database image (with or without a seed database), and then create an Instance (known as CDB in Oracle parlance) and add one or more Databases (known as PDBs).

After the El Carro Instance and Database(s) are created, you can take snapshot-based or RMAN-based backups and get basic monitoring and logging information. Additional database services will be added in future releases.

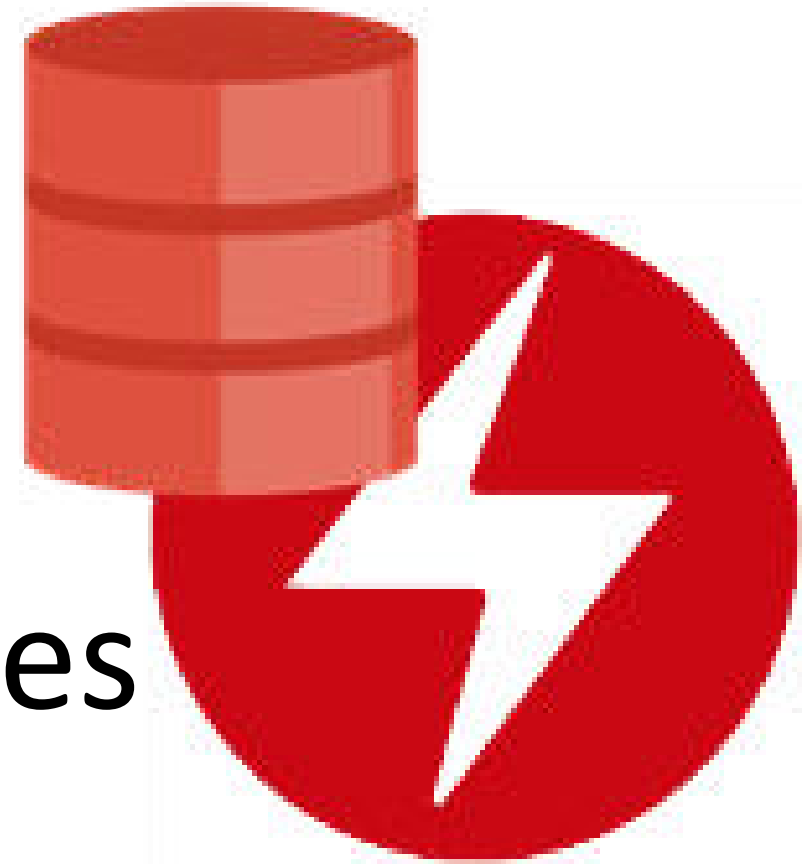
License Notice

You can use El Carro to automatically provision and manage Oracle Database Express Edition (XE) or Oracle Database Enterprise Edition (12c and 19c). In each case, it is your responsibility to ensure that you have appropriate licenses to use any such Oracle software with El Carro.

Please also note that each El Carro “database” will create a pluggable database, which may require licensing of the Oracle Multitenant option.

https://github.com/GoogleCloudPlatform/elcarro-oracle-operator

PURESTORAGE®



The Oracle Database Operator for Kubernetes

Aka 'OraOperator’

The Evolution of Containerisation for the Oracle Database

Docker

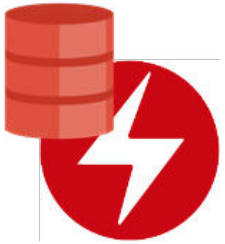
- Running Containers on Docker Engine
- Oracle Images (Single Instance, Sharding, Oracle RAC & CMAN)
- Standalone Environments
- OL7 Support
- Docker Compose

Podman

- Alternative to Docker
- Secure
- Oracle Images (Single Instance, Sharding, Oracle RAC & CMAN)
- Standalone Environments
- OL8 Support
- Podman Compose

Kubernetes

- Container Orchestration
- Software Defined Networking
- Observability and Scaling
- Single Instance & Sharding Support
- Helm

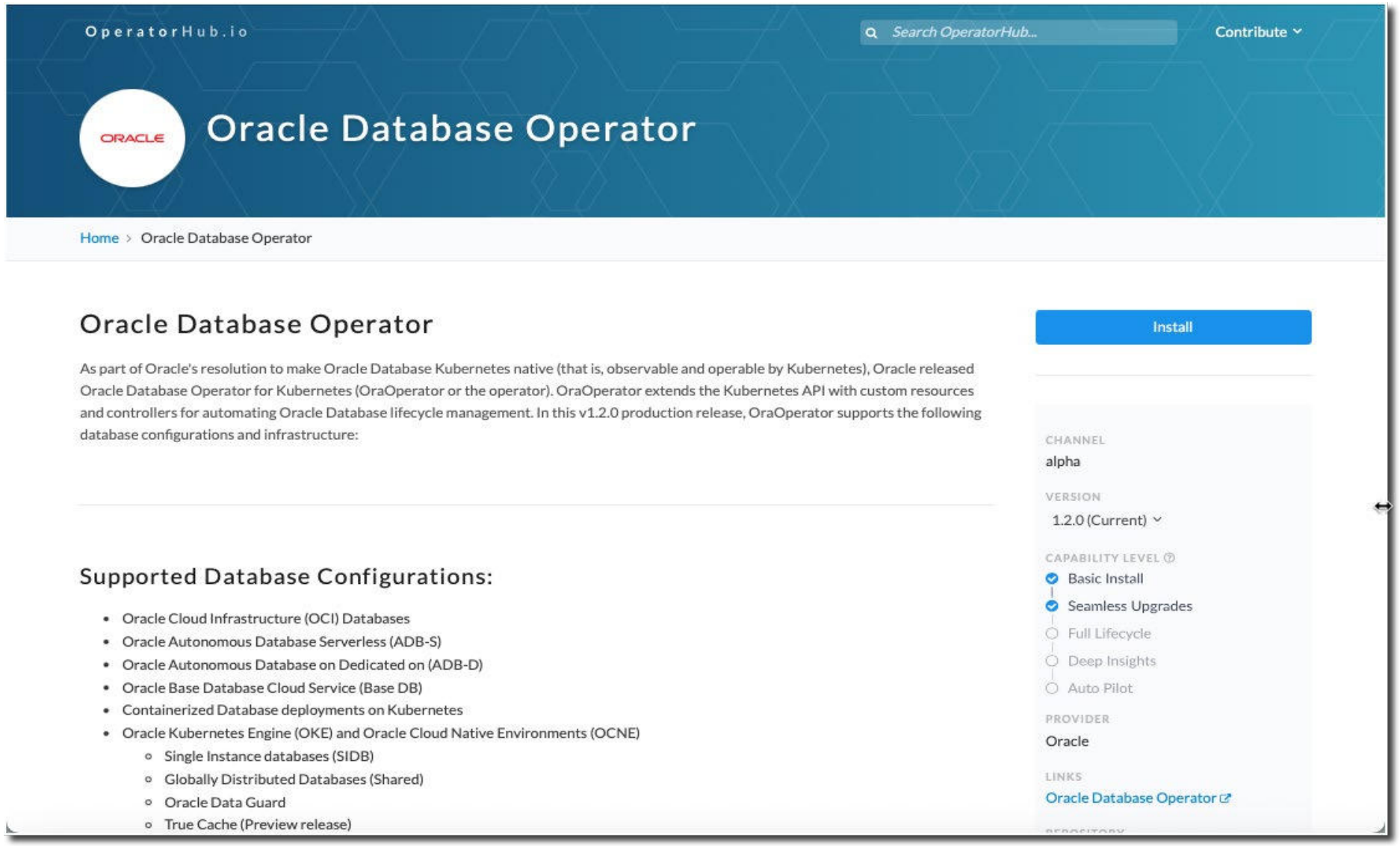


DB Operator

- Database-specific Controller
- Software extension of the Kubernetes APIs
- ADB, BaseDB, Single Instance database, Sharding & Multitenant
- Go

The Oracle Database Operator

Version 1.2.0



https://operatorhub.io/operator/oracle-database-operator

Oracle Database Operator for Kubernetes v2.0.0

Oracle Database Operator for Kubernetes

Make Oracle Database Kubernetes Native

As part of Oracle's resolution to make Oracle Database Kubernetes native (that is, observable and operable by Kubernetes), Oracle released the *Oracle Database Operator for Kubernetes* (`OraOperator` or the operator). OraOperator extends the Kubernetes API with custom resources and controllers for automating the management of the Oracle Database lifecycle.

What's New in v2.0.0

- **RedHat OpenShift**
  - Validation of OraOperator and Controllers
  - Inclusion in RedHat Operators Catalog
- **Restart Controller**
  - Provision, add & delete asm disks, and more
- **ORDS Service**
  - ServiceAccount and OpenShift support
  - Auto download of APEX installation files and APEX image on a Persistent Volume
- **Integrations**
  - Private Cloud Appliance (PCA)
  - Compute Cloud@Customer (C3)
- **Bug fixes**
  - Bugs filed through Oracle Support
  - GitHub issues

Supported Database Configurations v2.0.0

In this v2.0 production release, `OraOperator` supports the following database configurations and controllers:

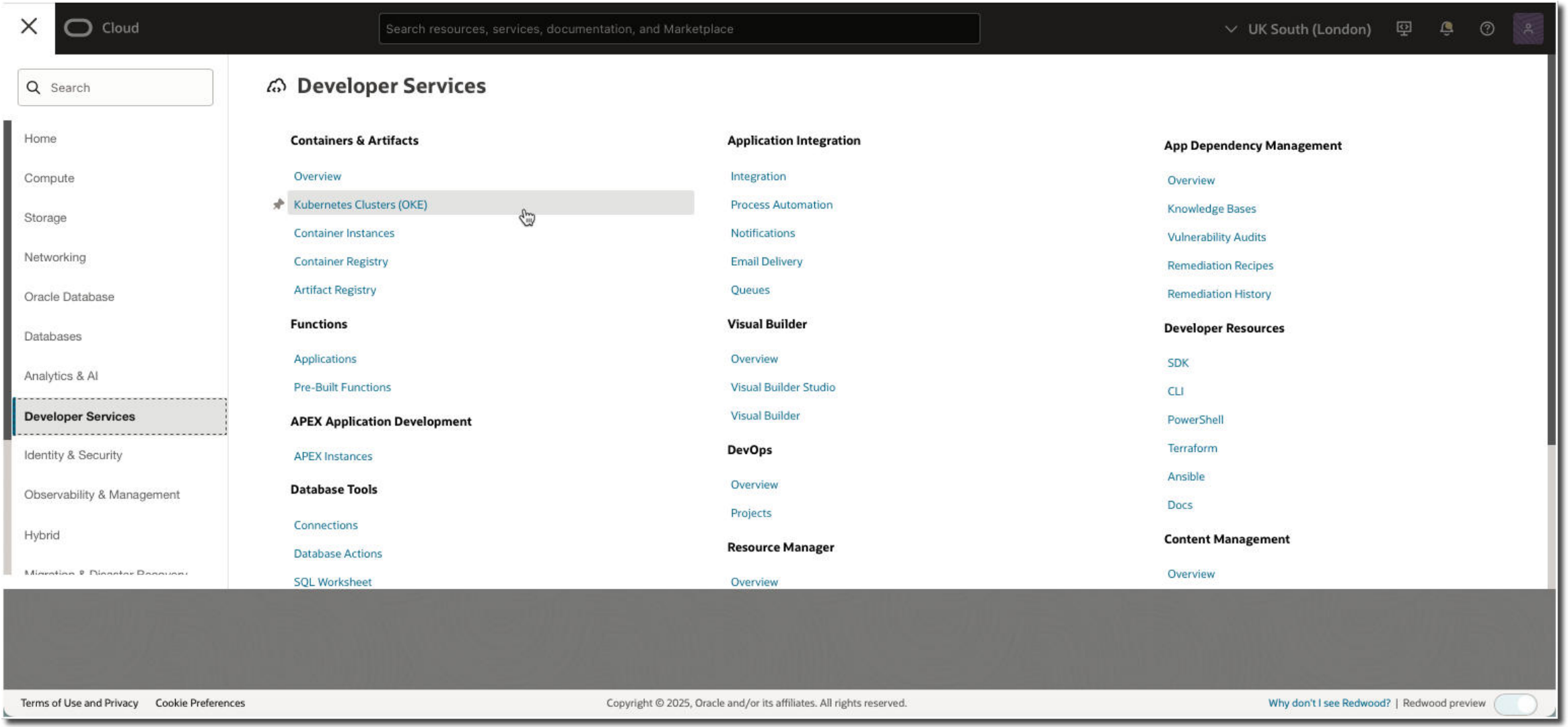
- **Oracle Autonomous Database:**
  - Oracle Autonomous Database shared Oracle Cloud Infrastructure (OCI) (ADB-S)
  - Oracle Autonomous Database on dedicated Cloud infrastructure (ADB-D)
  - Oracle Autonomous Container Database (ACD), the infrastructure for provisioning Autonomous Databases
- **Containerized Single Instance databases (SIDB)** deployed in the Oracle Kubernetes Engine (OKE) and any Kubernetes where OraOperator is deployed
- **Containerized Oracle Globally Distributed Databases (GDD)** deployed in OKE and any Kubernetes where OraOperator is deployed
- **Oracle Multitenant Databases (CDB/PDBs)**
- **Oracle Base Database Service (OBDS)** on Oracle Cloud Infrastructure (OCI)
- **Oracle Data Guard**
- **Oracle Database Observability**
- **Oracle Database Rest Service (ORDS) instances**
- **Oracle Restart**
- **Oracle Globally Distributed Database**

https://github.com/oracle/oracle-database-operator

PURESTORAGE®

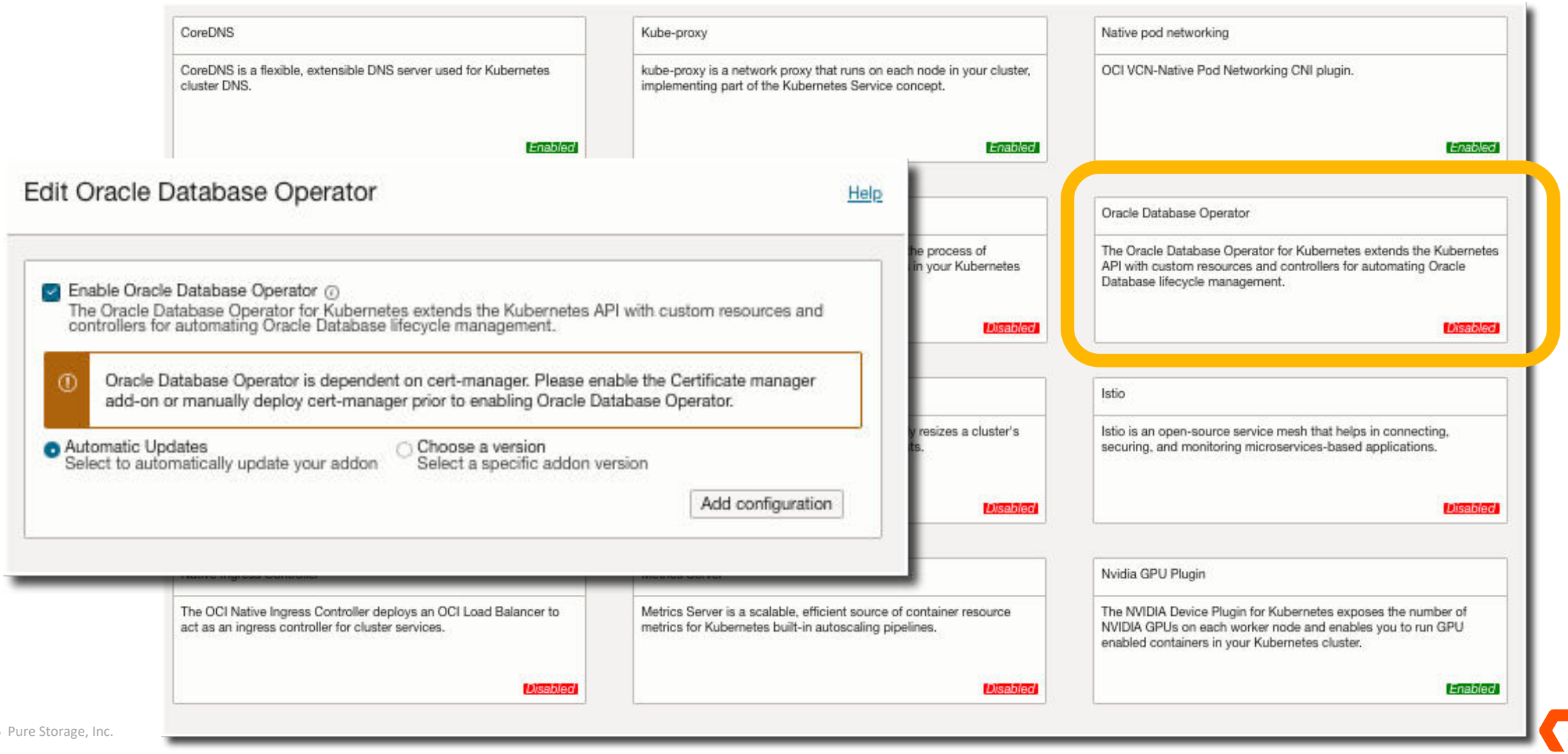
OKE OraOperator Installation

OKE Developer Services



OKE Oracle Database Operator Add-On

Only v1.1.0 & v1.2.0 available



PURESTORAGE®

Containerized Oracle Database 26ai using: the Oracle Database Operator for Kubernetes + Portworx Enterprise on OKE

Oracle Database 26ai Free

Using v2.0.0

Install cert-manager

- kubectl apply -f https://github.com/cert-manager/cert-manager/releases/download/v1.14.2/cert-manager.yaml

Install Database Operator

- kubectl apply -f https://raw.githubusercontent.com/oracle/oracle-database-operator/main/oracle-database-operator.yaml

SingleInstanceDatabase

name

namespace

secretName

stor

loac

vi

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: px-csi-ora
provisioner: pxd.portworx.com
parameters:
  repl: "3"
  io_profile: "auto"
  priority_io: "high"
allowVolumeExpansion: true
```

```
1  apiVersion: database.oracle.com/v1alpha1
2  kind: SingleInstanceDatabase
3  metadata:
4    name: freedb-livelabs
5    namespace: oracle23ai
6  spec:
7
8    ## Use only alphanumeric characters for sid
9    sid: FREE
10
11   ## DB edition
12   edition: free
13
14   ## Secret containing SIDB password mapped to secretKey
15   adminPassword:
16     secretName: freedb-admin-secret
17
18   ## Database image details
19   image:
20   ## Oracle Database Free is only supported from DB version 23.2 onwards
21     pullFrom: container-registry.oracle.com/database/free:latest
22     prebuiltDB: true
23
24   ## size is the required minimum size of the persistent volume
25                                                          provisioning
26   ## accessMode can only accept one of ReadWriteOnce,  ReadWriteMany
27   persistence:
28     size: 50Gi
29     ## oci-bv applies to OCI block volumes,
30     ## update as required
31     storageClass: "px-csi-ora"
32     accessMode: "ReadWriteOnce"
33
```

Provision containerized Oracle Database 26ai Free

- Provision Database
  - kubectl create –f singleinstancedatabase_free.yaml

```
-zsh                                                    ⌥⌘1
rekins@rekins--Mac14 23ai % kubectl create -f singleinstancedatabase_free.yaml
singleinstancedatabase.database.oracle.com/freedb-livelabs created
rekins@rekins--Mac14 23ai %
rekins@rekins--Mac14 23ai %
```

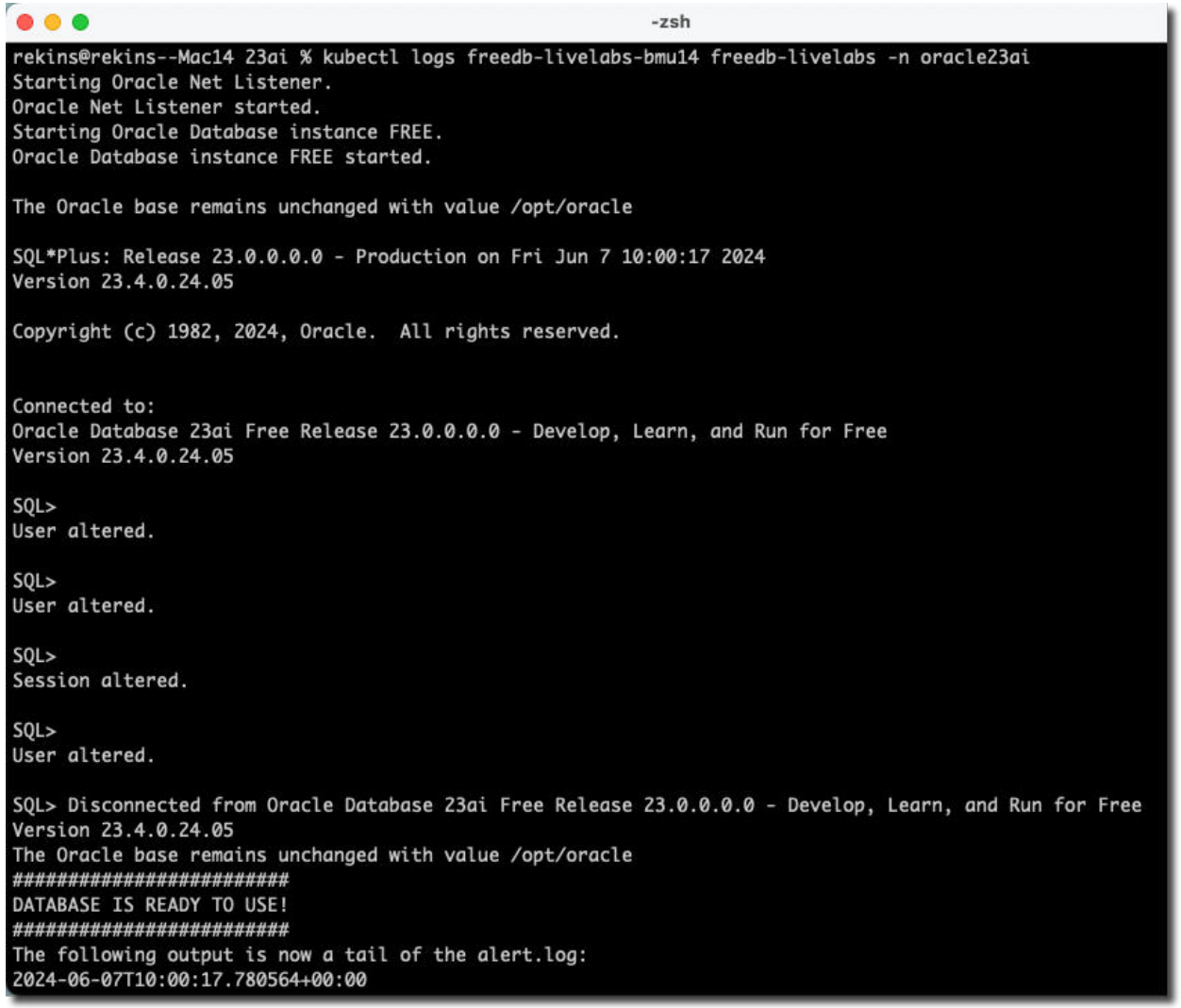
- Check Pod Status
  - kubectl get pods -n <namespace>

```
-zsh                                                    ⌥⌘1
rekins@rekins--Mac14 23ai % kubectl get pods -n oracle23ai
NAME                   READY   STATUS    RESTARTS   AGE
freedb-livelabs-bmu14  1/1     Running   0          4m2s
rekins@rekins--Mac14 23ai %
```

Database Logs

Kubectl logs <podname> -n <namespace>

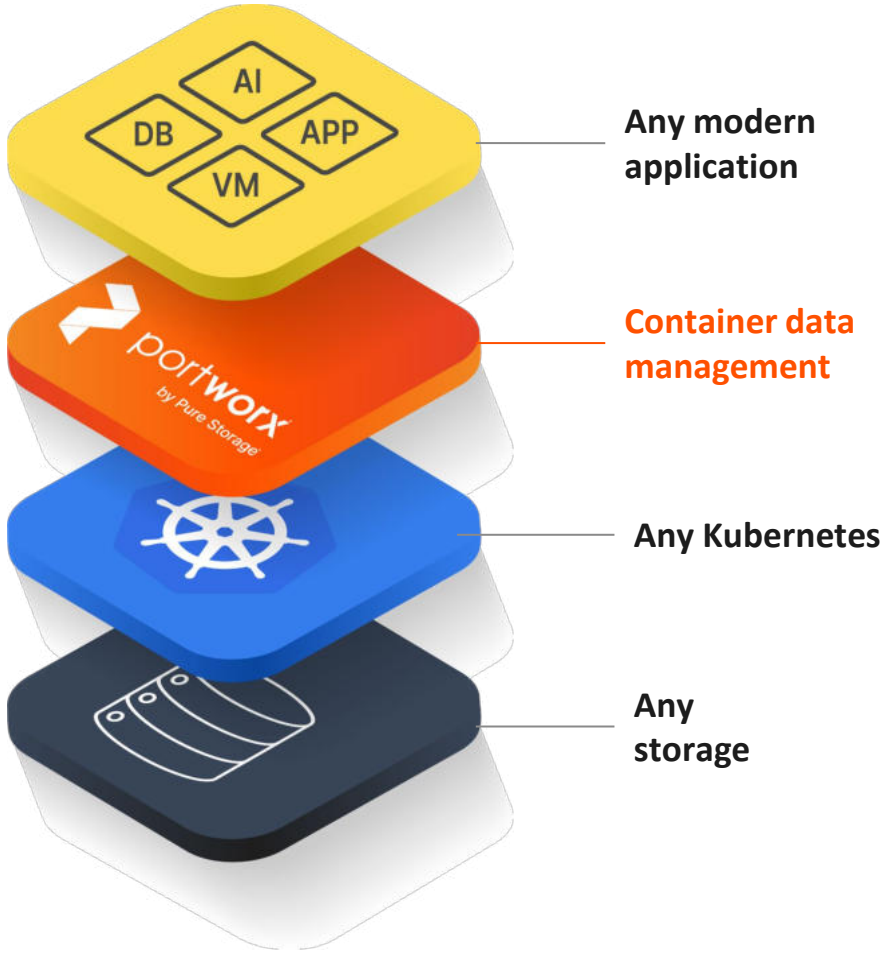
- Use kubectl logs to view the database alert.log file
- Use –follow to view database installation progress.
- Check
  - Database name
  - Oracle version
  - Status
  - …



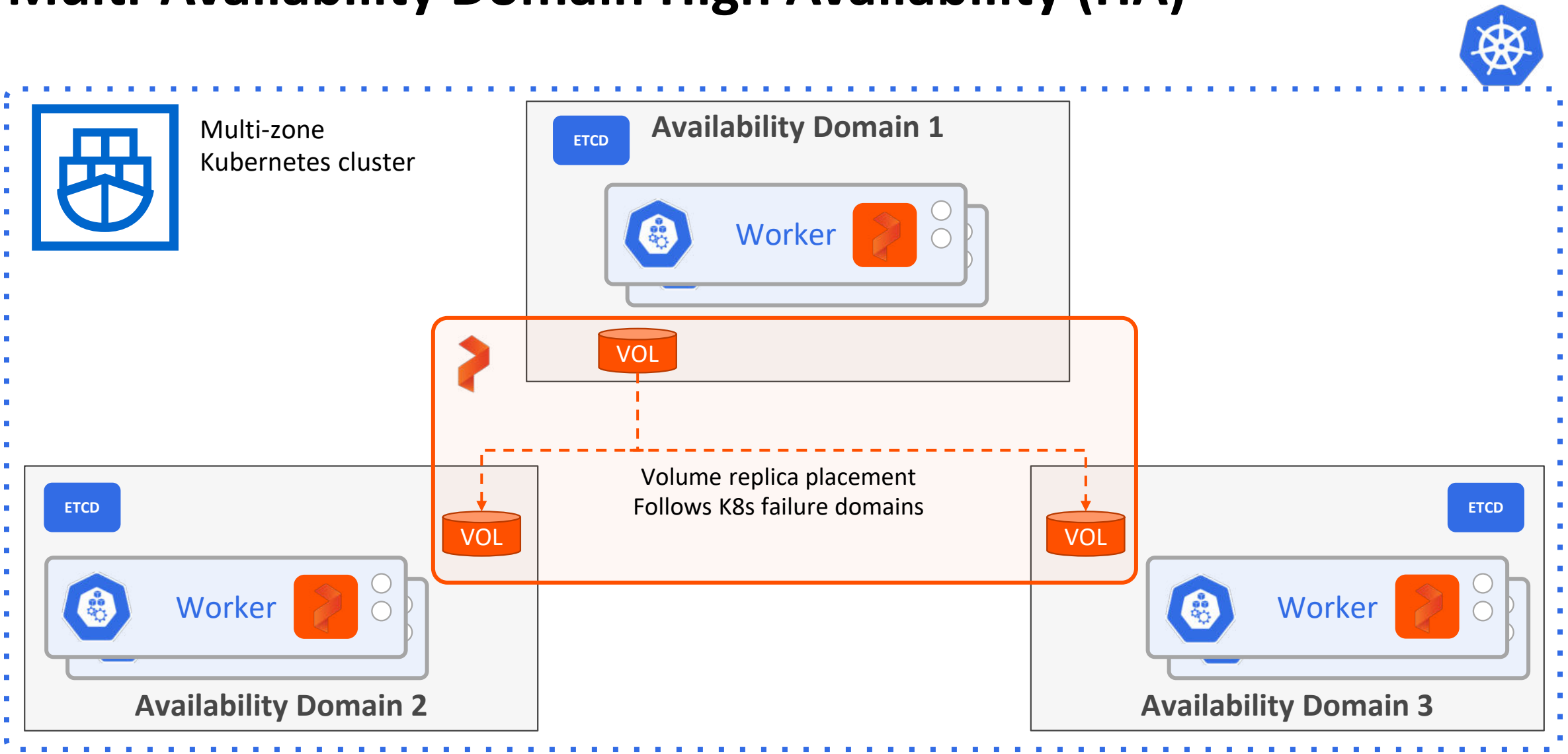
PURESTORAGE®

portworx® by Pure Storage

Oracle Database 26ai High Availability (HA) with the Oracle Database Operator for Kubernetes and Portworx



Multi-Availability Domain High Availability (HA)



Automate

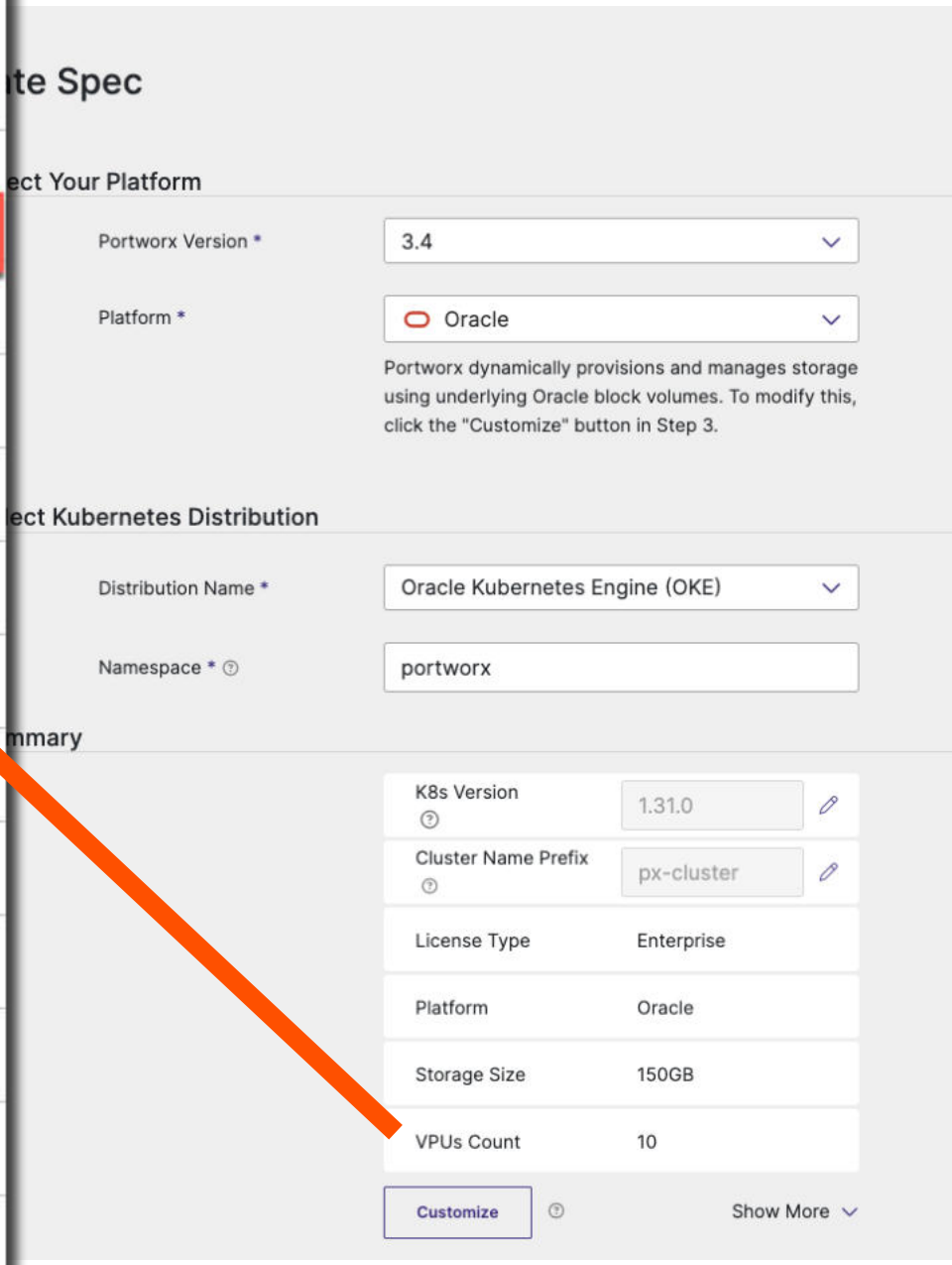
Day 0 Operatio

Streamlined installation a
for storage and data serv

Streamlined installatio

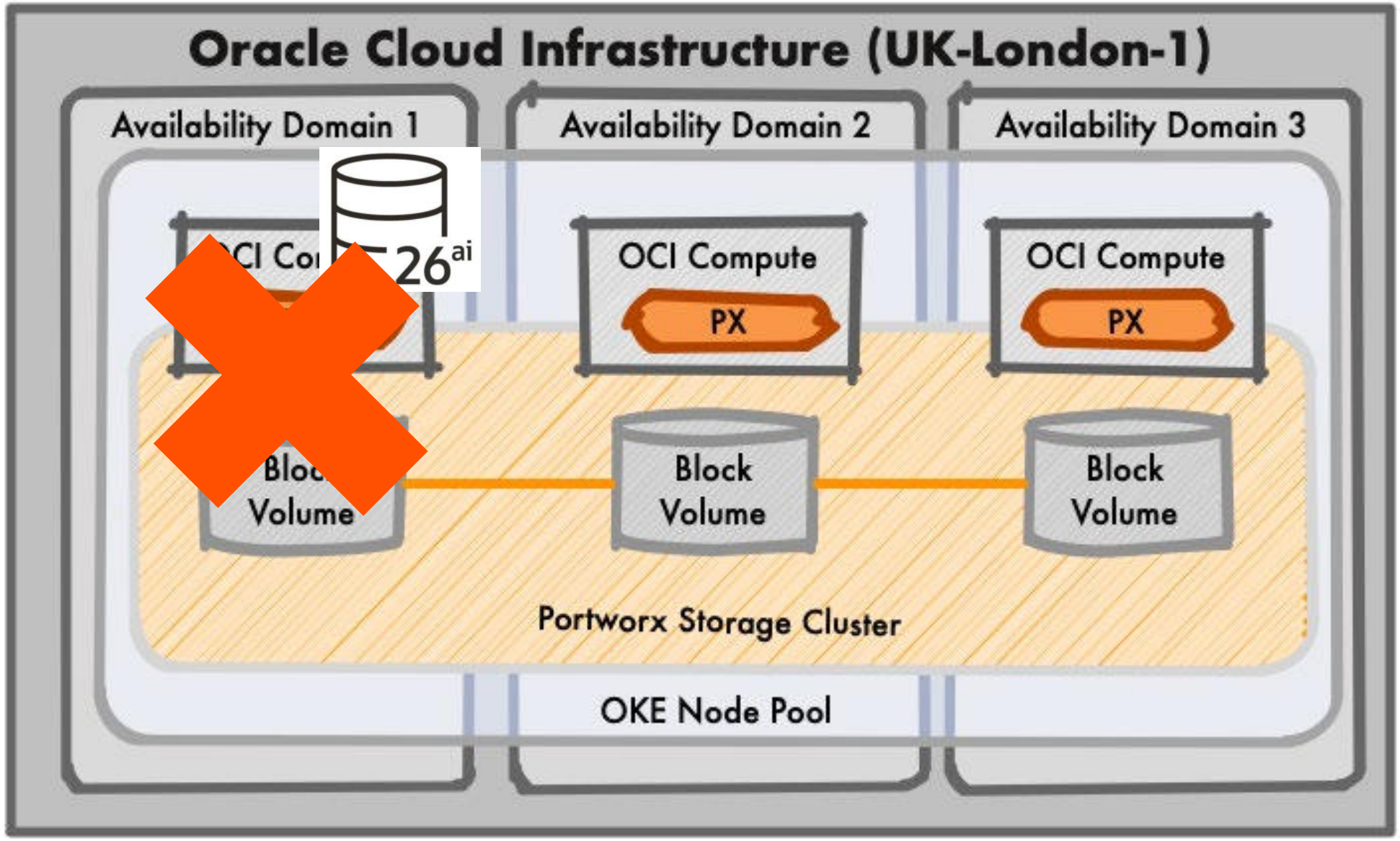
- Define storage policie
  premises, cloud, or hy
  environment.
- Select OCI Block Volu
  Performance using O
  Performance Units (V
- Install and deploy via
  extensions for Red Ha
  SUSE Rancher
- Support for GitOps w
  APIs, Terraform provi

Elastic Performance Level	Volume Performance Units (VPUs)	IOPS per GB	Max IOPS per Volume	Size for Max IOPS (GB)	KBPS per GB	Max MBPS per Volume
Lower Cost	0	2	3,000	1,500	240	480
Balanced	10	60	25,000	417	480	480
Higher Performance	20		50,000	667	600	680
Ultra High Performance	30	90	75,000	833	720	880
Ultra High Performance	40	105	100,000	952	840	1,080
Ultra High Performance	50	120	125,000	1,042	960	1,280
Ultra High Performance	60	135	150,000	1,111	1,080	480
Ultra High Performance	70	150	175,000	1,167	1,200	1,680
Ultra High Performance	80	165	200,000	1,212	1,320	1,880
Ultra High Performance	90	180	225,000	1,250	1,440	2,080
Ultra High Performance	100	195	250,000	1,282	1,560	2,280
Ultra High Performance	110	210	275,000	1,310	1,680	2,480
Ultra High Performance	120	225	300,000	1,333	1,800	2,680



https://docs.oracle.com/en-us/iaas/Content/Block/Concepts/blockvolumeperformance.htm

Oracle Database 26ai Planned / Managed Failover

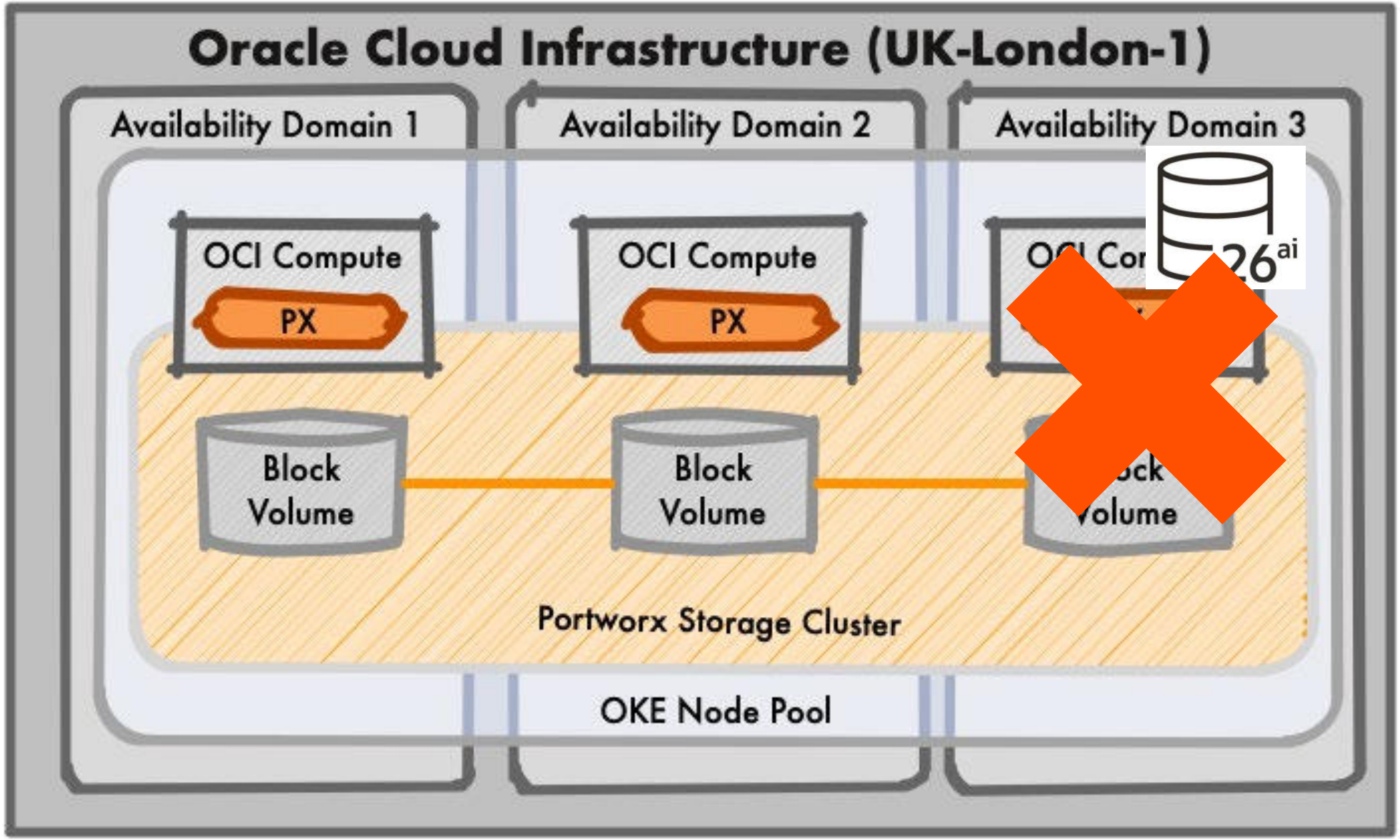


PURESTORAGE®

Containerised Oracle 26ai Unplanned Failover

Using Portworx Cloud Native Software Defined Storage

Oracle Database 26ai Unplanned Failover



Oracle Licensing

Overview of Licensing of Oracle Programs in Containers and Kubernetes

Container images that have been pulled to a host or a Kubernetes node may contain Oracle Programs. This may lead to licensing requirements on that host or that Kubernetes node for the Oracle Programs encapsulated within the image. As stated in the Oracle Partitioning Policy document:

> "Once a container image (e.g. a Docker image) containing Oracle Programs has been pulled to a host, or to a Kubernetes node in a Kubernetes cluster, (either a virtual machine or a physical machine), that host or Kubernetes node must be licensed for the Oracle Programs for the number of processors on that host or Kubernetes node. If the host or Kubernetes node is a physical machine, the number of processors on that host or Kubernetes node equals the number of processors on that physical machine. If the host or Kubernetes node is a virtual machine, then the number of processors on that host or Kubernetes node is subject to the guidelines documented in this Partitioning Policy."

Oracle recommends use of the procedures below for configuring environments that run Oracle Programs in containers and Kubernetes, in order to determine and manage licensing requirements for the Oracle Programs that may run in these environments.

https://www.oracle.com/a/tech/docs/running-and-licensing-programs-in-containers-and-kubernetes.pdf