

Revolutionising Database Deployment with the Oracle Database Operator for Kubernetes



Ron Ekins

Director, Field Solutions Architecture
EMEA & LATAM



Outsurance In the House Party



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>





Positioned Highest in Execution, Furthest in Vision

2025 Gartner® Magic Quadrant™ for Enterprise Storage Platforms

Figure 1: Magic Quadrant for Enterprise Storage Platforms



© Gartner, Inc.

Gartner.

Gartner®

Gartner, Inc. and/or its affiliates in the U.S. and internationally, and MAGIC QUADRANT is a registered trademark of Gartner, Inc. and/or its affiliates and are used herein with permission. All rights reserved.
Gartner does not endorse any vendor, product or service depicted in its research publications, and does not advise technology users to select only those vendors with the highest ratings or other designation. Gartner research publications consist of the opinions of Gartner's research organization and should not be construed as statements of fact. Gartner disclaims all warranties, expressed or implied, with respect to this research, including any warranties of merchantability or fitness for a particular purpose.
This graphic was published by Gartner, Inc. as part of a larger research document and should be evaluated in the context of the entire document. [The Gartner document is available upon request from Pure Storage.](#)



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



ACE Member Benefits



Key Benefits

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



Exclusive Content

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



Direct Access to Product Management

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



Networking

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



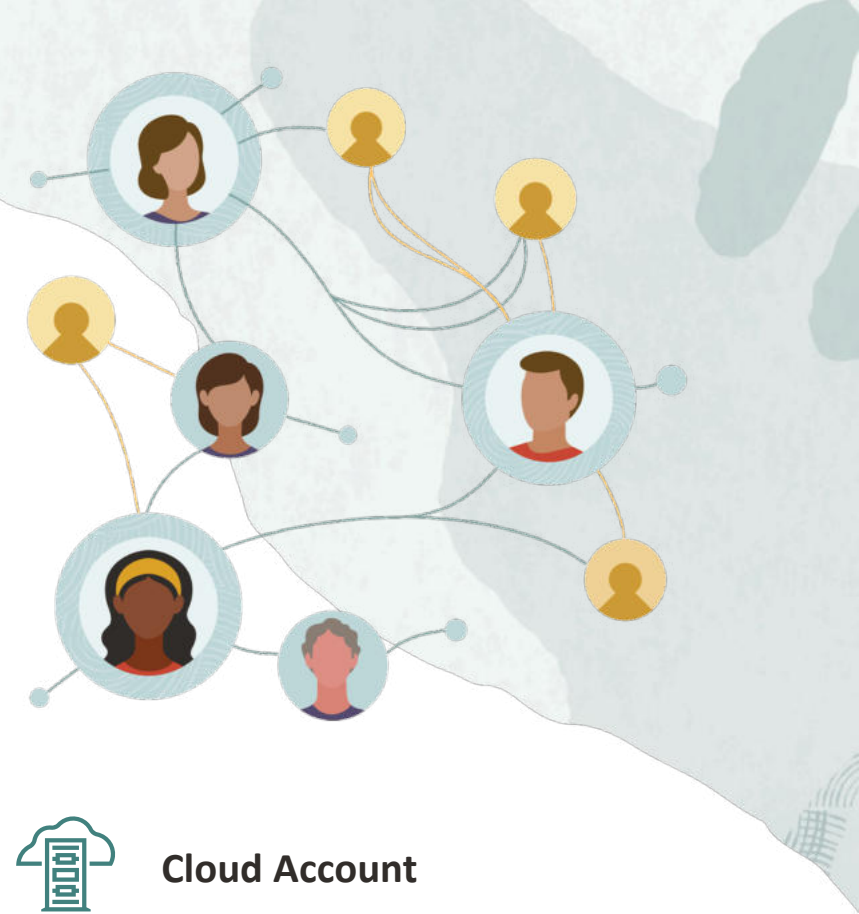
Cloud Account

\$5k USD Cloud account*



Travel Support

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



Agenda



Introduction to Oracle and
Containerisation



The Oracle Database
Kubernetes Operator



Oracle 23ai on OKE

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](#).



Oracle Container Registry

ORACLE[®]

Oracle Container Registry

Database Repositories \

Oracle Database Enterprise Edition

Description

Oracle Database Server Release 19c Docker Image Documentation

Oracle Database Server Release 19c is an industry leading relational database server. The Oracle Database server docker image contains Oracle Database server Release 19c (19.3.0.0) Enterprise or Standard Edition running on Oracle Linux 7. This image contains a default database in a multitenant configuration with one pluggable database.

For more information on Oracle Database server Release 19c, refer to <http://docs.oracle.com/en/database/>.

ORACLE[®]

Oracle Container Registry

Database Repositories \

Oracle Database Enterprise Edition

Release Image

Database server. The Oracle Database 21c (21.3.0.0) Enterprise Edition contains a default database in a

ORACLE[®]

Oracle Container Registry

Database Repositories \

Oracle Database Free

Description

Oracle AI Database 26ai Free Container Image Documentation

Oracle AI Database 26ai Free is the free edition of the industry-leading database. The Oracle AI Database 26ai Free Container Image contains Oracle AI Database 26ai Free based on an Oracle Linux 8 base image.

Two flavors of the image are supported:

1. The **Full** image: supports all the database features provided by Oracle AI Database 26ai Free.
2. The **Lite** image: smaller image size with a stripped-down installation of the database.

The **Lite** image has a smaller storage footprint than the **Full** image (~80% image size reduction) and a substantial improvement in image pull time. This image is useful in CI/CD scenarios and for simpler use cases where advanced database features are not required.

For more information on Oracle AI Database 26ai Free, see: <https://oracle.com/database/free>



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 planning 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 pack
virtually anywhere.
overcomes the limit
Docker provides a r

Related Resource:

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**





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

 Aerospike Kubernetes Operator provided by Aerospike The Aerospike Kubernetes Operator automates the	 Altinity Operator for ClickHouse provided by Altinity ClickHouse Operator manages full lifecycle of ClickHouse	 ArangoDB provided by ArangoDB GmbH ArangoDB Kubernetes Operator	 Cassandra provided by InstaClustr Manage the full lifecycle of the Cassandra clusters.	 Cloud Native PostgreSQL provided by EnterpriseDB Corporation Operator to manage Postgres high availability clusters w
 CockroachDB Helm Operator provided by Helm Community CockroachDB Operator based on the CockroachDB helm	 Couchbase Operator provided by Couchbase The Couchbase Autonomous Operator allows users to easily deploy, manage, and maint	 Crunchy Postgres for Kubernetes provided by Crunchy Data Production Postgres Made Easy	 DataStax Kubernetes Operator for Apache Cassandra provided by DataStax Simple provisioning, turn k	 druid-operator provided by Apache Druid Community druid-operator allows creating Apache Druid cluster and
 Elasticsearch (ECK) Operator provided by Elastic Run Elasticsearch, Kibana, APM Server, Beats, Enterp	 etcd provided by CNCF Create and maintain highly-available etcd clusters on Kubernetes	 Ext Postgres Operator provided by movetokube.com Manage databases and roles in external PostgreSQL server or cluster	 GrdsCloud MySQL for Kubernetes provided by Grds Cloud Enterprise open source MySQL-as-a-Service	 Hazelcast Platform Operator provided by Hazelcast, Inc Install Hazelcast clusters in Kubernetes environments.
 Infinispan Operator provided by Infinispan Create and manage Infinispan clusters.	 Logging Operator provided by Opstree Solutions A golang based CRD operator to set up and manage logging stack	 Mariadb Operator provided by Manoj Dhanorkar Create mariadb server and database on Kubernetes	 MongoDB Atlas Operator provided by MongoDB, Inc The MongoDB Atlas Kubernetes Operator enables easy management of Clus	 MongoDB Enterprise Operator provided by MongoDB, Inc The MongoDB Enterprise Kubernetes Operator enab

 MongoDB Operator provided by Opstree Solutions A golang based operator to create and manage MongoDB standalone setup and clus	 Operator for Apache CouchDB provided by IBM Apache CouchDB is a highly available NOSQL database	 Percona Distribution for MongoDB Operator provided by Percona Percona Distribution for MongoDB Operator	 Percona Distribution for MySQL Operator provided by Percona Percona Distribution for MySQL Operator manages	 Percona Operator for PostgreSQL provided by Percona Percona Operator for PostgreSQL manages the
 PlanetScale Operator for Vitess provided by PlanetScale PlanetScale's operator for Vitess deploys and manage	 Postgres-Operator provided by Zalando SE Postgres operator creates and manages PostgreSQL clusters running in Kubernetes.	 Postgresql Operator provided by Openlabs Deploys postgresql based applications	 PostgreSQL Operator by Dev4DDevs.com provided by Dev4Devs.com Operator in Go developed using the Operator Frame	 Qserv operator provided by Vera C. Rubin Observatory Create and maintain highly-available Qserv clusters on
 Redis Enterprise provided by Redis Labs, Inc An operator to run Redis Enterprise Clusters	 Redis Operator provided by Opstree Solutions A Golang based redis operator that will make/oversee Redis standalone/cluster mode s	 Robin Cloud Native Storage provided by Robin.io Robin Cloud Native Storage operator enables advanced data management capabili	 Steerd Presto Operator provided by Falarica Analytics Private Limited Steerd Presto Operator is a Kubernetes Operator for	 TiDB Operator provided by PingCAP TiDB Operator manages TiDB clusters on Kubernetes and automates tasks related to
 Upbound Universal Crossplane (UXP) provided by Upbound Inc. Upbound Universal Crossplane (UXP) is Upbound's official	 VerticaDB Operator provided by Vertica Operator that manages a Vertica Eon Mode database.	 Yugabyte Operator provided by YugabyteDB This operator installs and manages a YugabyteDB cluster.		



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 provisions, manages, and backs up your PostgreSQL clusters on Kubernetes. Designed for your GitOps workflows, it is [easy to get started](#) with Postgres on Kubernetes with PGO. Within minutes, you can have a 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 i 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 Kuberi 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 PostgreSQL

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 to 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`, `spec.pg_explain`, and `spec.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 policy (configurable retention window)



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

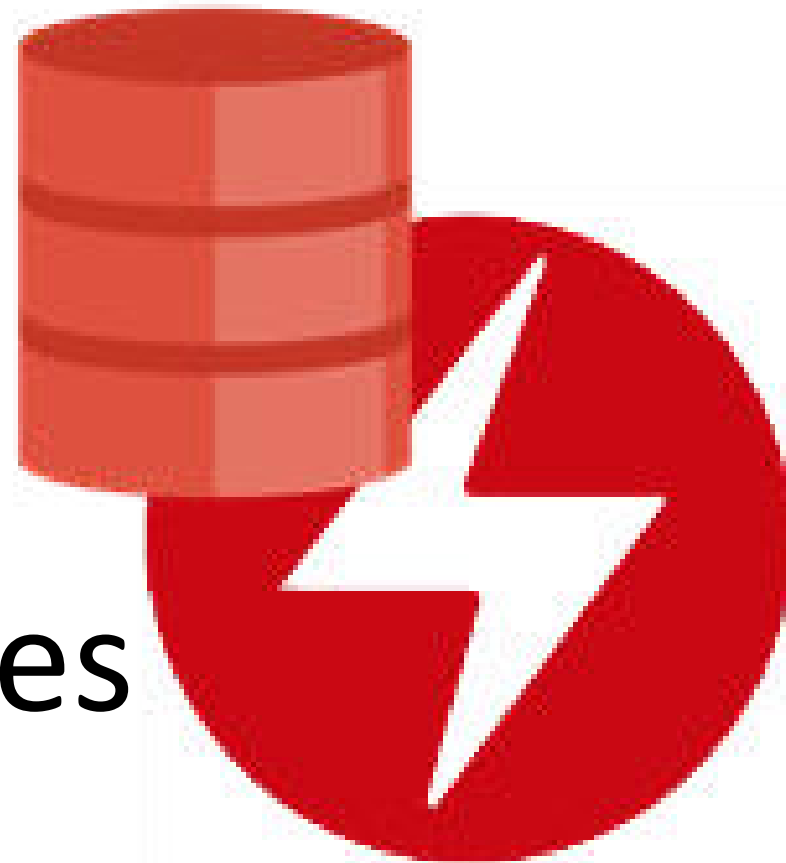
You can use El Carro to automatically provision and manage Oracle Database Express Edition (XE) or Oracle Database Enterprise Edition (12c and 19c). In each case, it is your responsibility to ensure that you have appropriate licenses to use any such Oracle software with El Carro.

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



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



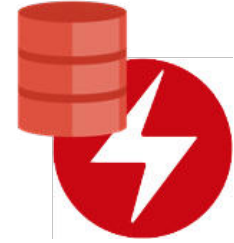
Podman

- 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



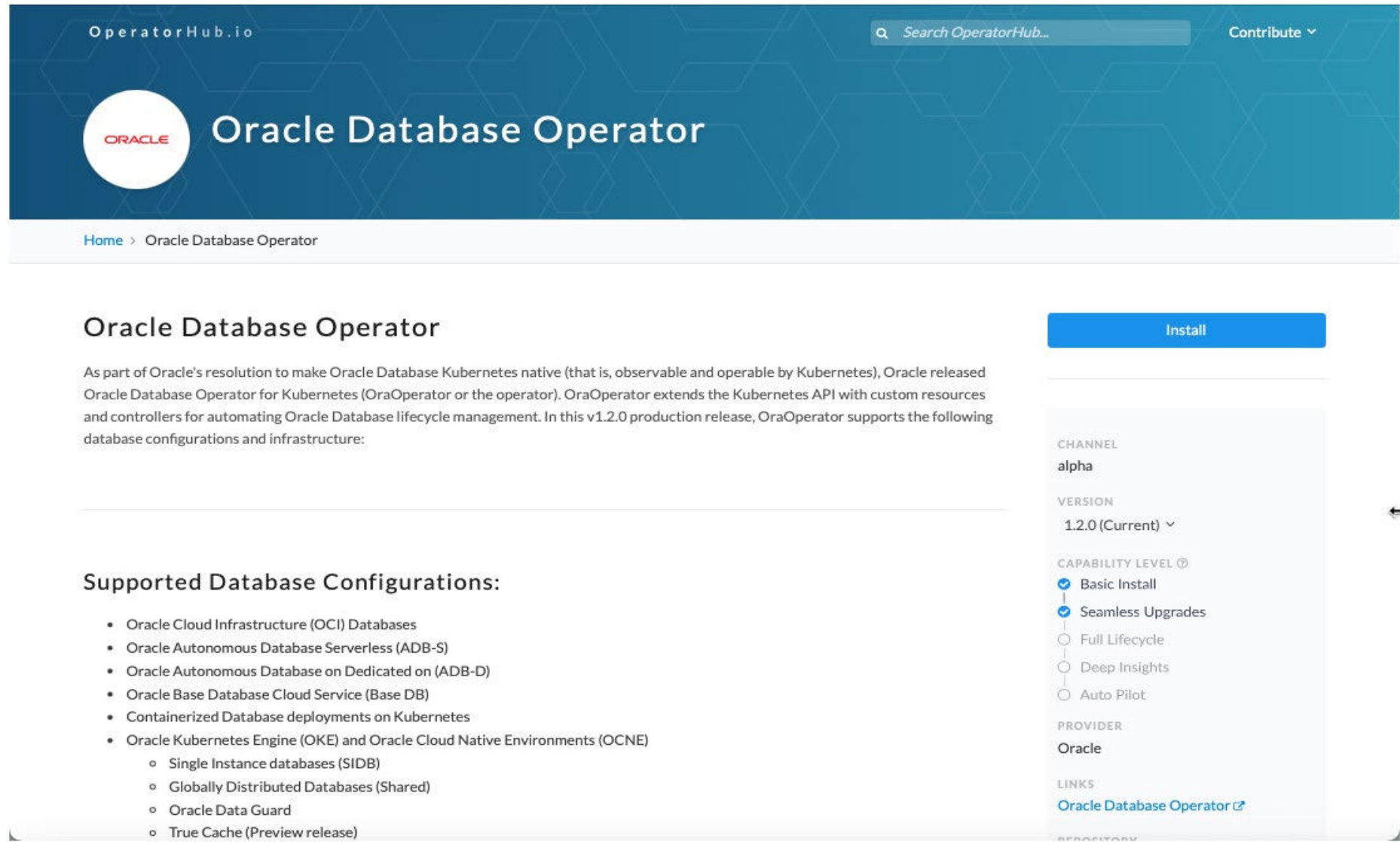
DB Operator

- 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



The screenshot shows the Oracle Database Operator page on OperatorHub.io. The header features the OperatorHub.io logo, a search bar, and a 'Contribute' button. The main heading is 'Oracle Database Operator' with the Oracle logo. Below this is a breadcrumb 'Home > Oracle Database Operator'. The main content area has a title 'Oracle Database Operator' followed by a paragraph describing its purpose: 'As part of Oracle's resolution to make Oracle Database Kubernetes native (that is, observable and operable by Kubernetes), Oracle released Oracle Database Operator for Kubernetes (OraOperator or the operator). OraOperator extends the Kubernetes API with custom resources and controllers for automating Oracle Database lifecycle management. In this v1.2.0 production release, OraOperator supports the following database configurations and infrastructure:'. To the right of this text is a blue 'Install' button. Below the paragraph is a section titled 'Supported Database Configurations:' followed by a bulleted list of supported configurations. On the right side of the page, there is a sidebar with configuration options: 'CHANNEL' set to 'alpha', 'VERSION' set to '1.2.0 (Current)', 'CAPABILITY LEVEL' with 'Basic Install' and 'Seamless Upgrades' selected, 'PROVIDER' set to 'Oracle', and a 'LINKS' section with a link to 'Oracle Database Operator'.

OperatorHub.io

Search OperatorHub...

Contribute

ORACLE

Oracle Database Operator

Home > Oracle Database Operator

Oracle Database Operator

As part of Oracle's resolution to make Oracle Database Kubernetes native (that is, observable and operable by Kubernetes), Oracle released Oracle Database Operator for Kubernetes (OraOperator or the operator). OraOperator extends the Kubernetes API with custom resources and controllers for automating Oracle Database lifecycle management. In this v1.2.0 production release, OraOperator supports the following database configurations and infrastructure:

Supported Database Configurations:

- Oracle Cloud Infrastructure (OCI) Databases
- Oracle Autonomous Database Serverless (ADB-S)
- Oracle Autonomous Database on Dedicated on (ADB-D)
- Oracle Base Database Cloud Service (Base DB)
- Containerized Database deployments on Kubernetes
- Oracle Kubernetes Engine (OKE) and Oracle Cloud Native Environments (OCNE)
 - Single Instance databases (SIDB)
 - Globally Distributed Databases (Shared)
 - Oracle Data Guard
 - True Cache (Preview release)

Install

CHANNEL
alpha

VERSION
1.2.0 (Current)

CAPABILITY LEVEL

- ☒ Basic Install
- ☒ Seamless Upgrades
- ☐ Full Lifecycle
- ☐ Deep Insights
- ☐ Auto Pilot

PROVIDER
Oracle

LINKS
[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**



OKE OraOperator Installation

OKE Developer Services

The screenshot displays the Oracle Cloud Developer Services console. The top navigation bar includes a search bar, the region 'UK South (London)', and user profile icons. The left sidebar lists various service categories, with 'Developer Services' highlighted. The main content area is titled 'Developer Services' and is organized into four columns of links. In the first column, under 'Containers & Artifacts', the 'Kubernetes Clusters (OKE)' link is highlighted with a mouse cursor. Other links in this column include 'Overview', 'Container Instances', 'Container Registry', and 'Artifact Registry'. The second column, 'Functions', includes 'Applications' and 'Pre-Built Functions'. The third column, 'APEX Application Development', includes 'APEX Instances'. The fourth column, 'Database Tools', includes 'Connections', 'Database Actions', and 'SQL Worksheet'. The fifth column, 'Application Integration', includes 'Integration', 'Process Automation', 'Notifications', 'Email Delivery', and 'Queues'. The sixth column, 'Visual Builder', includes 'Overview', 'Visual Builder Studio', and 'Visual Builder'. The seventh column, 'DevOps', includes 'Overview' and 'Projects'. The eighth column, 'Resource Manager', includes 'Overview'. The ninth column, 'App Dependency Management', includes 'Overview', 'Knowledge Bases', 'Vulnerability Audits', 'Remediation Recipes', and 'Remediation History'. The tenth column, 'Developer Resources', includes 'SDK', 'CLI', 'PowerShell', 'Terraform', 'Ansible', and 'Docs'. The eleventh column, 'Content Management', includes 'Overview'. The footer contains links for 'Terms of Use and Privacy', 'Cookie Preferences', a copyright notice for 2025, and a toggle for 'Redwood preview'.

Developer Services

- Containers & Artifacts**
 - [Overview](#)
 - [Kubernetes Clusters \(OKE\)](#)
 - [Container Instances](#)
 - [Container Registry](#)
 - [Artifact Registry](#)
- Functions**
 - [Applications](#)
 - [Pre-Built Functions](#)
- APEX Application Development**
 - [APEX Instances](#)
- Database Tools**
 - [Connections](#)
 - [Database Actions](#)
 - [SQL Worksheet](#)
- Application Integration**
 - [Integration](#)
 - [Process Automation](#)
 - [Notifications](#)
 - [Email Delivery](#)
 - [Queues](#)
- Visual Builder**
 - [Overview](#)
 - [Visual Builder Studio](#)
 - [Visual Builder](#)
- DevOps**
 - [Overview](#)
 - [Projects](#)
- Resource Manager**
 - [Overview](#)
- App Dependency Management**
 - [Overview](#)
 - [Knowledge Bases](#)
 - [Vulnerability Audits](#)
 - [Remediation Recipes](#)
 - [Remediation History](#)
- Developer Resources**
 - [SDK](#)
 - [CLI](#)
 - [PowerShell](#)
 - [Terraform](#)
 - [Ansible](#)
 - [Docs](#)
- Content Management**
 - [Overview](#)



OKE Oracle Database Operator Add-On

Only v1.1.0 & v1.2.0 available

The screenshot displays the 'Edit Oracle Database Operator' configuration window in the Oracle Cloud console. The window is titled 'Edit Oracle Database Operator' and includes a 'Help' link. It features a checkbox for 'Enable Oracle Database Operator' which is checked. Below this, a description states: 'The Oracle Database Operator for Kubernetes extends the Kubernetes API with custom resources and controllers for automating Oracle Database lifecycle management.' A warning message is displayed in a yellow box: 'Oracle Database Operator is dependent on cert-manager. Please enable the Certificate manager add-on or manually deploy cert-manager prior to enabling Oracle Database Operator.' Below the warning, there are two radio buttons: 'Automatic Updates' (selected) and 'Choose a version' (unselected). The 'Automatic Updates' option has a sub-label 'Select to automatically update your addon'. The 'Choose a version' option has a sub-label 'Select a specific addon version'. An 'Add configuration' button is located at the bottom right of the configuration window. In the background, the 'Add-Ons' page is visible, showing a grid of add-ons. The 'Oracle Database Operator' add-on is highlighted with a yellow border and is currently 'Disabled'. Other add-ons shown include CoreDNS (Enabled), Kube-proxy (Enabled), Native pod networking (Enabled), Istio (Disabled), Nvidia GPU Plugin (Enabled), and Metrics Server (Disabled).

Add-On	Status
CoreDNS	Enabled
Kube-proxy	Enabled
Native pod networking	Enabled
Oracle Database Operator	Disabled
Istio	Disabled
Nvidia GPU Plugin	Enabled
Metrics Server	Disabled



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

storageClass

accessMode

```
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    ## size is only applicable for persistent volume 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`

```
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>`

```
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
 - ...

```
-zsh
rekins@rekins--Mac14 23ai % kubectl logs freedb-livelabs-bmu14 freedb-livelabs -n oracle23ai
Starting Oracle Net Listener.
Oracle Net Listener started.
Starting Oracle Database instance FREE.
Oracle Database instance FREE started.

The Oracle base remains unchanged with value /opt/oracle

SQL*Plus: Release 23.0.0.0.0 - Production on Fri Jun 7 10:00:17 2024
Version 23.4.0.24.05

Copyright (c) 1982, 2024, Oracle. All rights reserved.

Connected to:
Oracle Database 23ai Free Release 23.0.0.0.0 - Develop, Learn, and Run for Free
Version 23.4.0.24.05

SQL>
User altered.

SQL>
User altered.

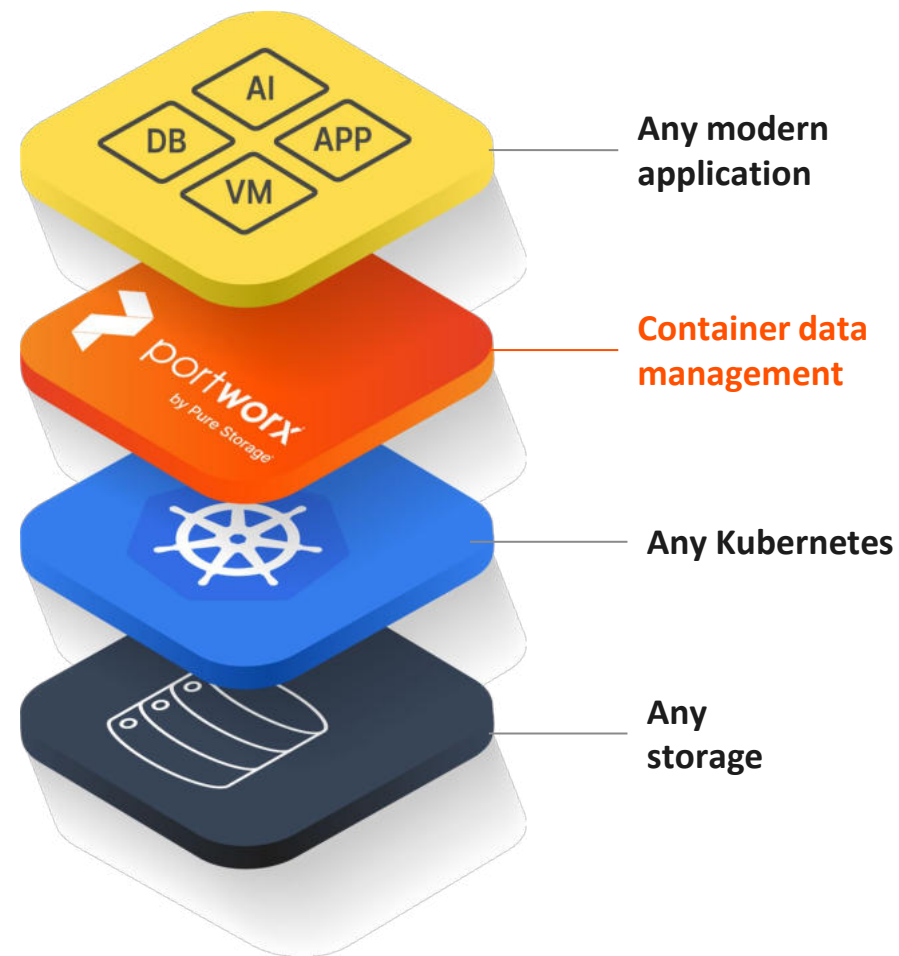
SQL>
Session altered.

SQL>
User altered.

SQL> Disconnected from Oracle Database 23ai Free Release 23.0.0.0.0 - Develop, Learn, and Run for Free
Version 23.4.0.24.05
The Oracle base remains unchanged with value /opt/oracle
#####
DATABASE IS READY TO USE!
#####
The following output is now a tail of the alert.log:
2024-06-07T10:00:17.780564+00:00
```



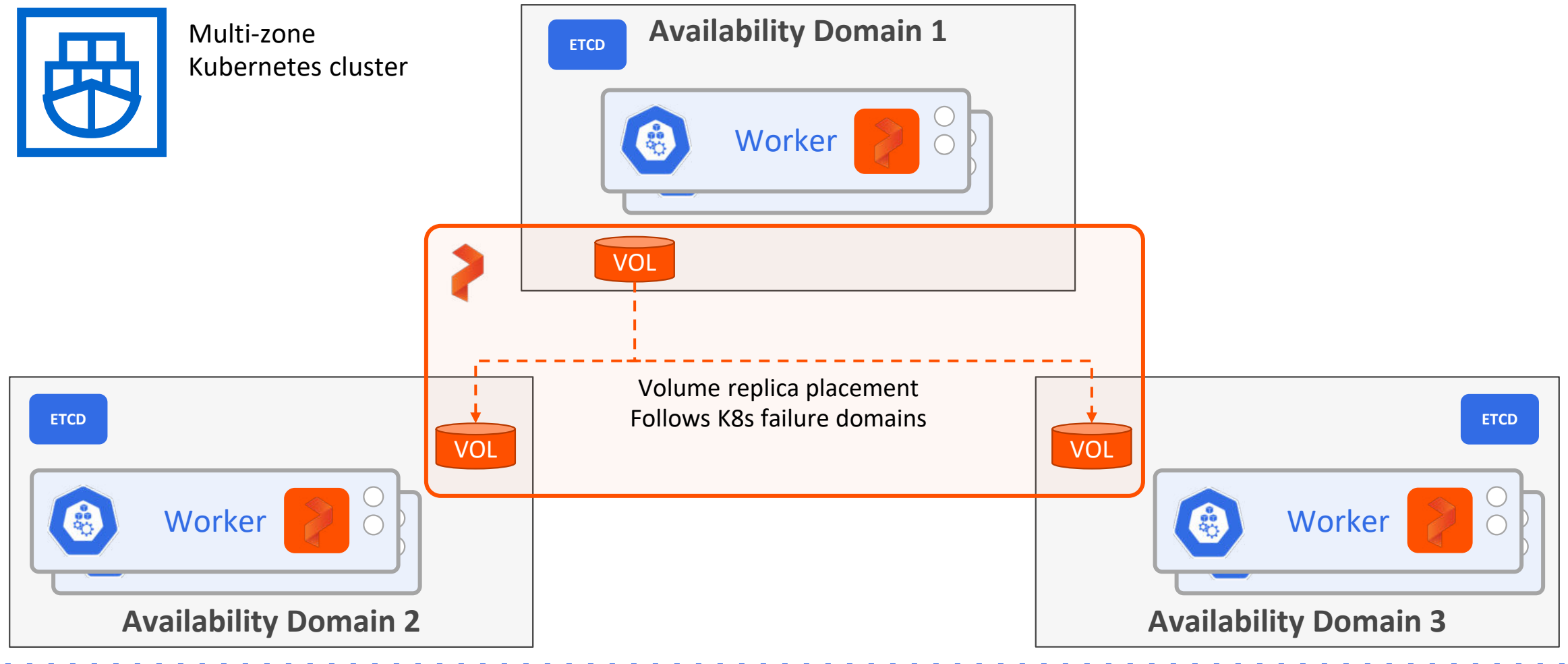
Oracle Database 26ai High Availability (HA) with the Oracle Database Operator for Kubernetes and Portworx



Multi-Availability Domain High Availability (HA)



Multi-zone
Kubernetes cluster



Automate Day 0 Operation

Streamlined installation and configuration for storage and data services

Streamlined installation

- Define storage policies for on-premises, cloud, or hybrid environment.
- Select OCI Block Volume Performance using Oracle Performance Units (VPU)
- Install and deploy via extensions for Red Hat OpenShift or SUSE Rancher
- Support for GitOps workflows, Helm charts, Terraform providers

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	120	50,000	667	600	680
Ultra High Performance	30	90	75,000	833	720	880
Ultra High Performance	40	105	100,000	1,000	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	1,440
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

Create Spec

Select Your Platform

Portworx Version *

Platform *

Portworx dynamically provisions and manages storage using underlying Oracle block volumes. To modify this, click the "Customize" button in Step 3.

Select Kubernetes Distribution

Distribution Name *

Namespace *

Summary

K8s Version

Cluster Name Prefix

License Type

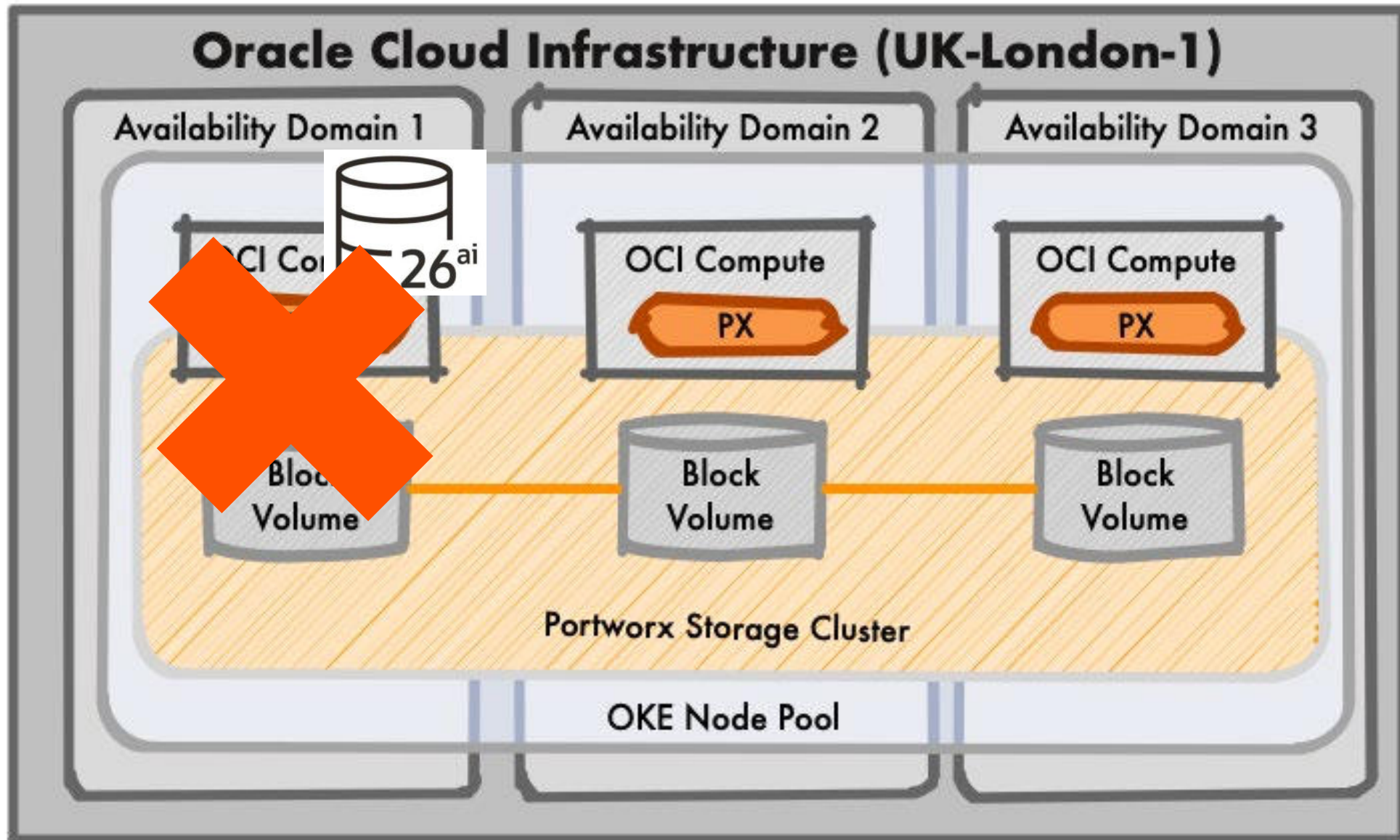
Platform

Storage Size

VPUs Count



Oracle Database 26ai Planned / Managed Failover

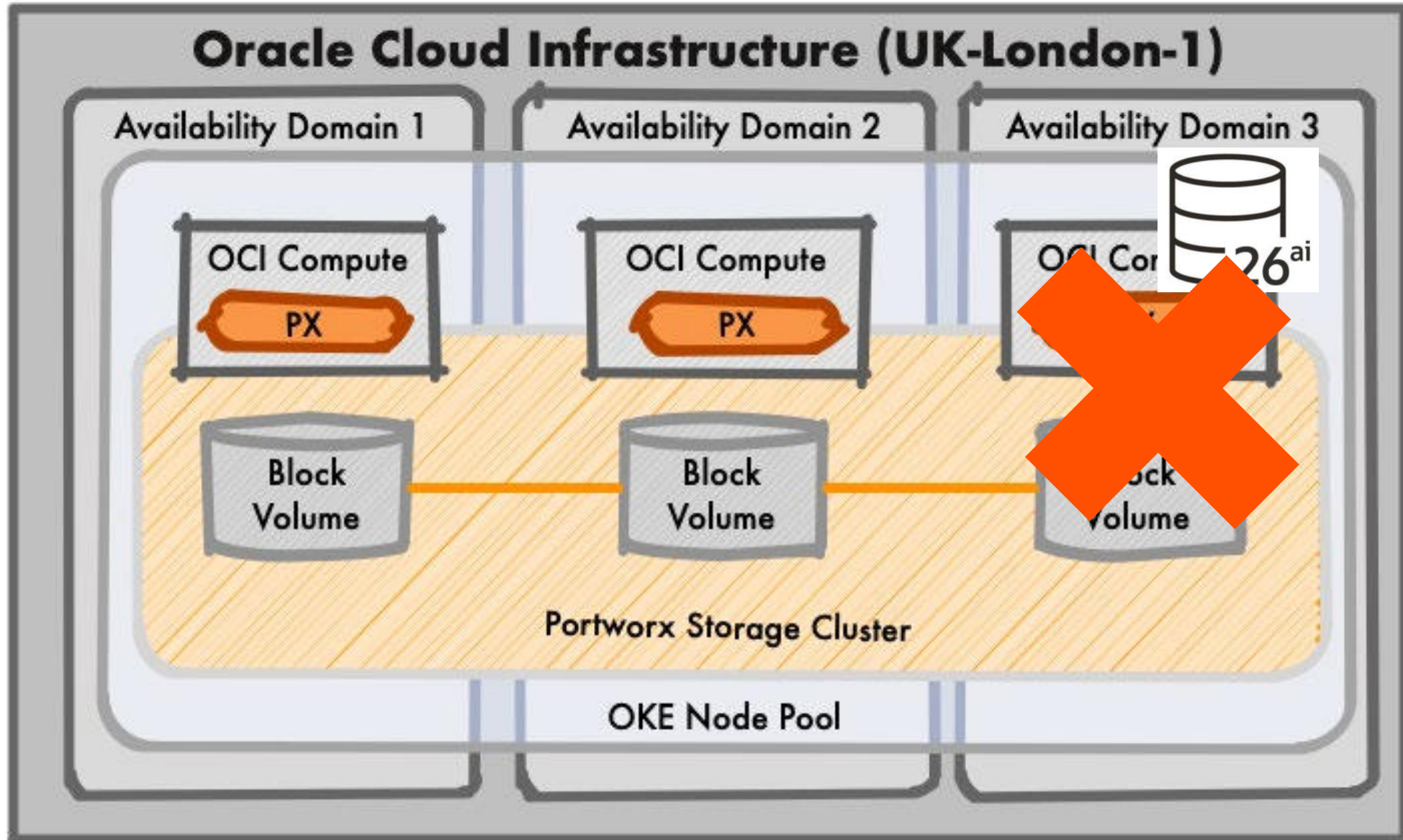




Containerised Oracle 26ai Unplanned Failover

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.





Thank You

Any Questions ?