|  |  |  |  |
| --- | --- | --- | --- |
| Technical White Paper |  |  |  |

# Title

Oracle Database in Docker Container

# Author

Kumar Viswanathan

# Product(s)

Oracle Database and Docker Container

# Purpose of the white paper

The white paper is intended for audiences who want to understand the scenarios, pros and cons while considering implementing Oracle Database in a On-Premise / Virtual Machine (IAAS cloud solution) as against Docker Container.

# Introduction

Oracle Database is a widely used relational database software robust to store enterprise and custom data. The purpose of the Oracle Database is to store and retrieve information. Oracle Database is used as a database for OLTP and OLAP systems.

Oracle Database is the first database designed for enterprise grid computing, the most flexible and effective way to manage information and applications. Enterprise grid computing creates large pools of industry-standard, modular storage and servers. With this architecture, each new system can be rapidly provisioned from the pool of components. There is no need for peak workloads, because capacity can be easily added or reallocated from the resource pools as needed. The database has logical structures and physical structures. Because the physical and logical structures are separate, the physical storage of data can be managed without affecting the access to logical storage structures.

Docker is an open platform for developing, shipping, and running applications. Docker enables you to separate your applications from your infrastructure so you can deliver software quickly. With Docker, you can manage your infrastructure in the same ways you manage your applications. By taking advantage of Docker’s methodologies for shipping, testing, and deploying code quickly, you can significantly reduce the delay between writing code and running it in production.

Docker provides the ability to package and run an application in a loosely isolated environment called a container. The isolation and security allow you to run many containers simultaneously on a given host. Containers are lightweight because they don’t need the extra load of a hypervisor but run directly within the host machine’s kernel. This means you can run more containers on a given hardware combination than if you were using virtual machines.

Let’s understand the difference between running any service in Virtual Machine against Docker Container

# Virtual Machine Vs Docker Container

The below are the differences of implementing any software /service considering

|  |  |
| --- | --- |
| **Virtual Machine** | **Docker Container** |
| Operating System is completely allocated to the Virtual Machine and runs the single workload. | Operating System is shared among many applications and runs multiple workloads. |
| Cost is high as compared to Virtual Machine as it is dedicated to the software which is installed in it. | Cost is shared among applications or database that is installed in the Docker Container, so the overhead is less. |
| Secured environment as systems can be isolated in a separate cloud network. | Less secured as they cannot be isolated completely as many applications are deployed in single or multiple containers. |
| Can be used to configure and install any services, applications etc. | It is more application focused. |
| Deployments, boots and drop are slower as compared to Docker Container | Deployments, boots and drop are faster as compared to Virtual Machine. |
| Storage provisioning is more flexible as it allows many types of storage to be provisioned. | Docker Container supports copy on write feature where in we can only use the base image stored and |
| Virtual machines allow flexibility in re-deploying, re-writing and maintaining the code of the application. | Docker containers are normally used where the application is expected not to be re-written. |

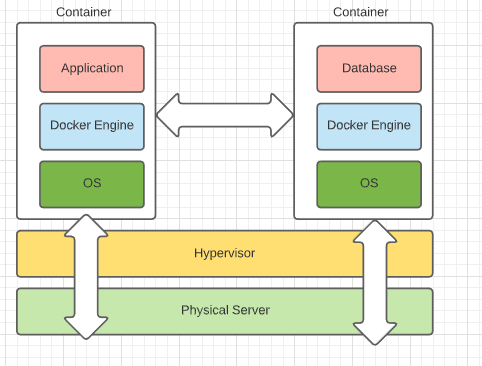
# Features of Oracle Database for Microservices

The following are the features available with the Oracle Database which can be used for implementing Microservices:

* Oracle Database provides Multitenant Architecture that allow microservices to have separate databases (PDB’s – Pluggable Databases) which can be clubbed together and managed inside a Container Database (CDB).
* For managing messages and transfer of data, Oracle Advanced Queuing simplifies microservices design by providing transactional messaging of events.
* Oracle Database provides a Multi Model engine which supports Relational, OLAP, JSON, Document, XML, Graph and Spatial.
* Oracle Materialized Views, multi-shard and cross PDB queries support real time analysis
* Oracle Database provides industry leading security, scalability, high availability and data analytics solutions.

# Architectural Diagram of Oracle Database in Docker Container

Figure shows a logical view of Oracle Database implementation in Docker Container:



The above diagram shows a setup where in we can implement multiple applications and databases (including Oracle) inside a physical or a virtual server but above the hypervisor. The database setup inside the Docker Container is a monolith database architecture.

Currently, the Oracle Database Server Docker image is available in the Docker Hub (the path is mentioned below) and contains Oracle Database Server 12.2.0.1 Enterprise Edition running on Oracle Linux 7. This image contains a default database in a multi-tenant configuration with a single pluggable database.

<https://hub.docker.com/_/oracle-database-enterprise-edition>

# Advantages of Oracle Database in Docker Container

The below are some of the advantages when considering implementing Oracle Database in Docker Container.

* Supports Microservices architecture
* Organize application as a set of well defined, loosely coupled services
* Improves parallelization of development of an application
* Makes application easier to understand.
* Allows teams to deploy their services independently
* Adheres to continuous delivery model
* Useful when the data does not grow much
* Low N/w Latency between the application and database
* Less cost overhead as infrastructure is shared among application and database.
* Preferable for databases that are smaller in size
* Supports multi-tenant architecture where in each microservice when build can communicate with it’s own PDB’s (Pluggable Databases) making it as a private data model.
* Multitenant database containers manage many databases as one, secure separation of data, easy sharing and querying of data.
* Can be implemented in scenarios where we need to implement microservices, as a repository for Business Intelligence applications, the database upgrades and patching can be avoided or is required at a bare minimum, the database would undergo sharding where the database growth is controlled.
* Minimum backup and recovery procedures are required
* Also suited for environments such as DEV and UAT which normally are used by few users and don’t need an additional server saving cost.

# Disadvantages of Oracle Database in Docker Container

The following disadvantages of having the Oracle when considering implementing Oracle Database in Docker Container.

* The architecture makes it difficult to troubleshoot basic Oracle errors as well as internal errors (eg: ORA- errors).
* Oracle Enterprise Manager would not be able to provide deep level monitoring of the database which are installed inside the Docker Container.
* Security update patches, upgrades and critical patch updates are not yet been tested and applied in the Oracle Database that is installed in a Docker Container. As such this technology is as of now at a premature stage.
* The architecture does not support big application whose database size can range or seen to be more than 200 GB.
* Performance Tuning of a database installed in Docker Container is very difficult to achieve when the explain plan changes.
* Database revision changes are different from application changes and it is still not clear to achieve the same in this monolith architecture. The changes have to be implemented manually and cannot be automated as of now.
* CPU and memory used by the Oracle Database is more that 90% at all times in the configuration leaving very bare minimum memory to be used by the Operating System. Application if hosted in the same server on top of Docker Container would require appropriate sizing and scaling of the server on them it is installed.
* Isolation of the Database and implementing stringent security Database features
* High Availability solutions such as Physical Standby database, Active Data Guard, Real Application Clusters cannot be implemented effectively in the Docker architecture.
* Direct SSH connectivity is not possible to the Oracle Database and additional access is required in the Docker container to fulfill the access.

# Summary

In this white paper we have covered the basic introduction to the architecture of Oracle Database in Docker container, its benefits in setting up Microservices environments. We also noticed the advantages and disadvantages of implementing the setup inside a Docker Container.

The document provides an insight of using the Oracle Database in a Docker Container and provides only a point of view of using this architecture. It is important for the architect to understand the client requirements, use this document as a base to evaluate if and when required to implement the Oracle Database in a Docker Container.