Contents

[What is Docker? 2](#_Toc55985414)

Introduction to Containers and Docker

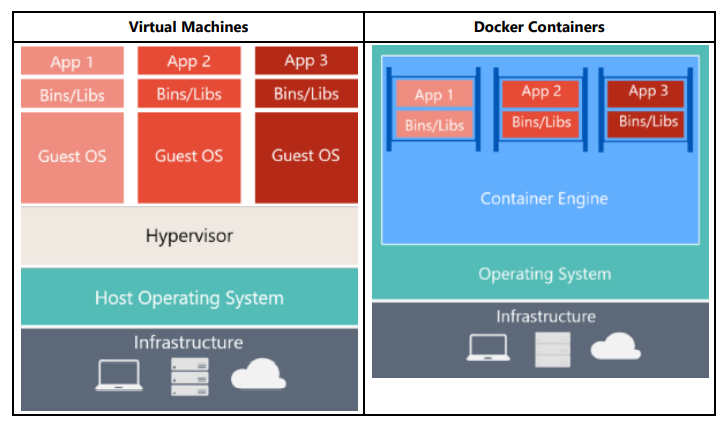
Containerization is an approach to software development in which an application or service, its dependencies and its configuration are packaged together as a container image. The containerized application can be tested as a unit and deployed as a container image instance to the host operating system.

Containerization also isolate applications from each other on a shared OS.

Another benefit of containerization is scalability.

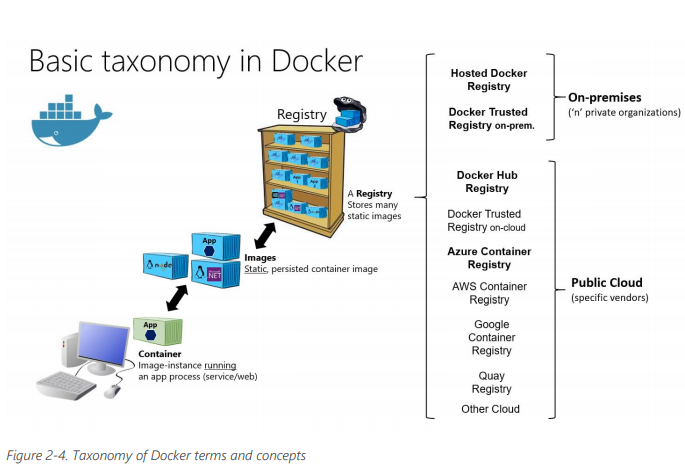
# What is Docker?

Docker is an open-source project for automating the deployment of applications as portable, self-sufficient containers that can run on the cloud or on-premises.

Docker containers can run anywhere, on-premises in the customer datacentre, in an external service provider or in the cloud.

|  |  |
| --- | --- |
| Virtual Machines | Docker Containers |
| Virtual Machines include the application, the required libraries or binaries and a full guest OS. Full virtualization requires more resources than containerization. | Containers include the application and all its dependencies. However, they share the OS kernel with other containers, running as isolated processes in user space on the host OS. |

Docker Terminology:

* **Container Image**
* **Dockerfile**
* **Build**
* **Container**
* **Volume**
* **Tag**
* **Multi-Stage Build**
* **Repository (repo)**
* **Registry:** A service that provides access to repositories.
* **Multi-arch Image**: For multi-architecture, it’s a feature that simplifies the selection of the appropriate image, according to the platform where Docker is running.
* Docker Hub
* **Docker Trusted Registry (DTR)**: could be installed on -premises so it lives within the organization’s datacenter and network.
* **Compose**: A command-line tool and YAML file format with metadata for defining and running multi-container applications.
* **Cluster**: A collection of Docker hosts exposed as if it were a single virtual Docker host, so that the application can scale to multiple instances of the services spread across multiple hosts within the cluster.
* **Orchestrator**: A tool that simplifies management of clusters and Docker hosts.