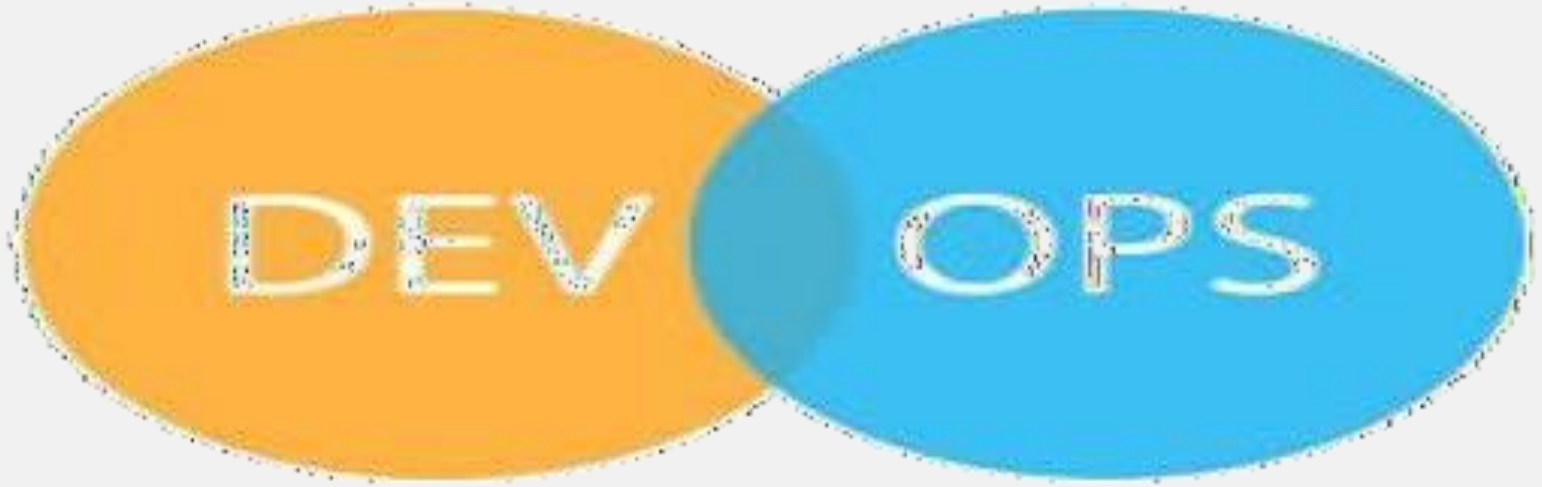
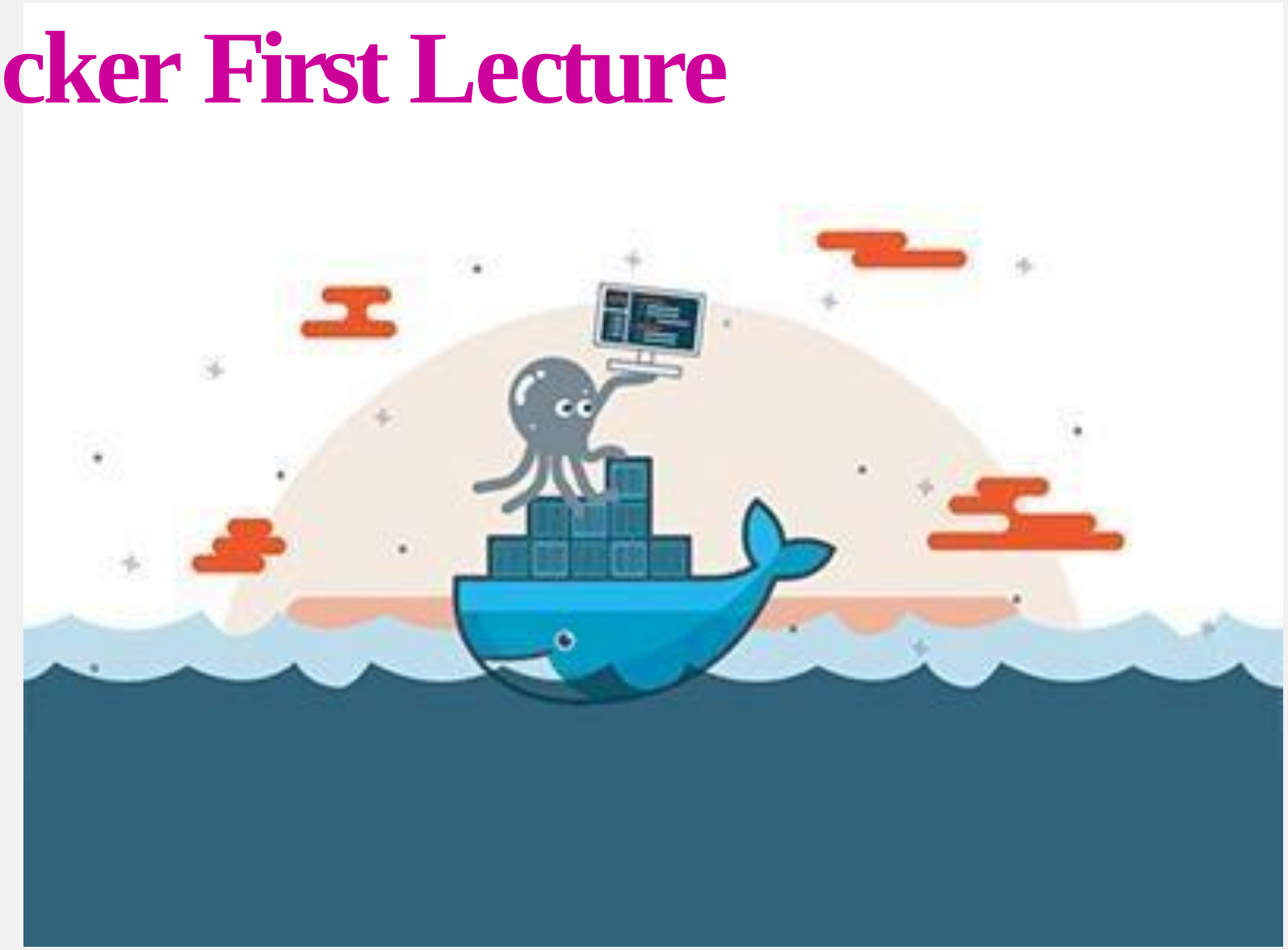
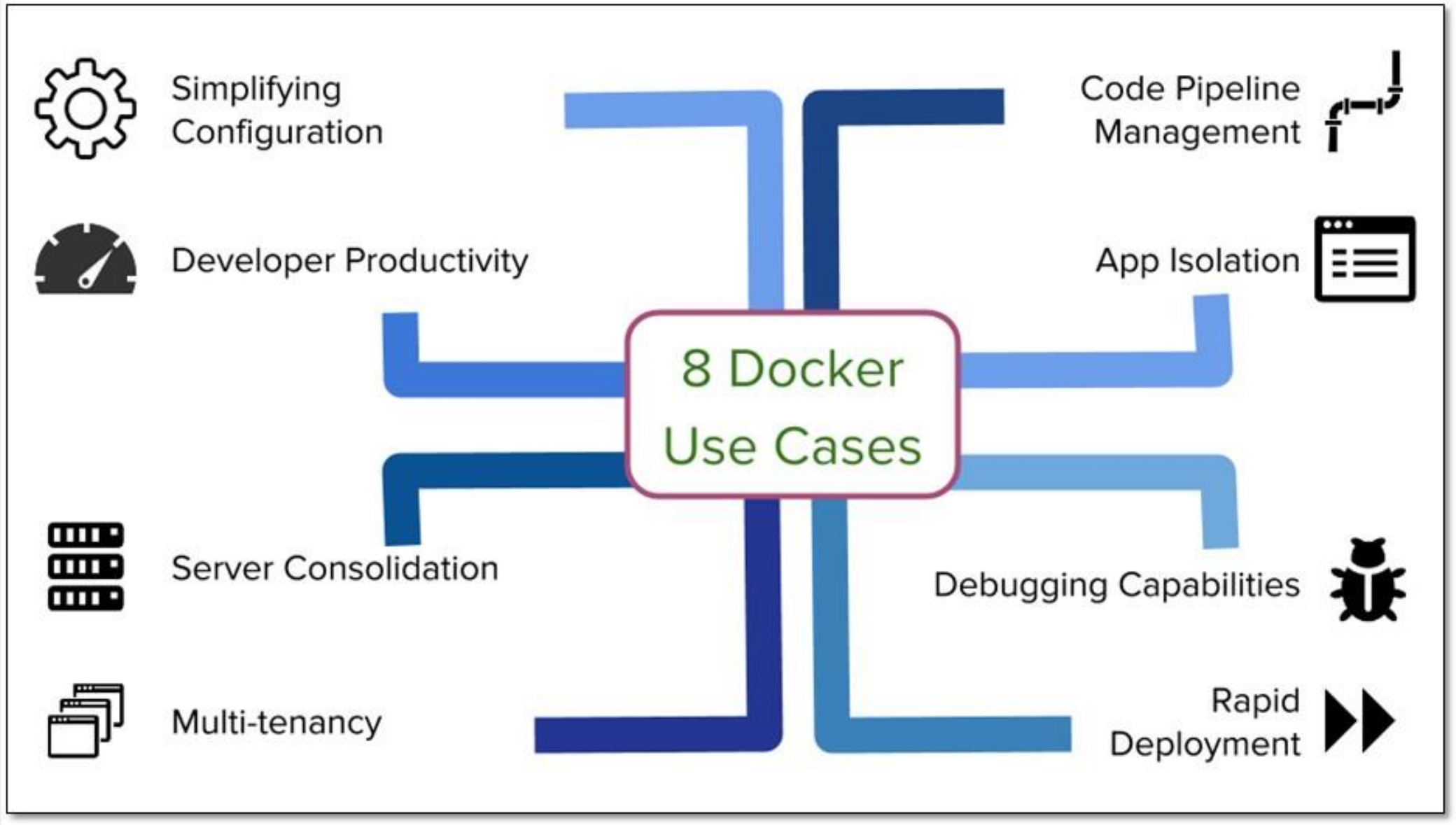

DEV
OPS

Docker First Lecture




Simplifying Configuration
Developer Productivity
Server Consolidation
Multi-tenancy
8 Docker Use Cases
Code Pipeline Management
App Isolation
Debugging Capabilities
Rapid Deployment

Docker's Benefits

1. Environment standardization: **Docker would make sure all created environments are consistent.**

2. Faster configuration with consistency: **Put our configurations into code and deploy it.** And **save a lot of time from preparing the setup and deployment documentation.**

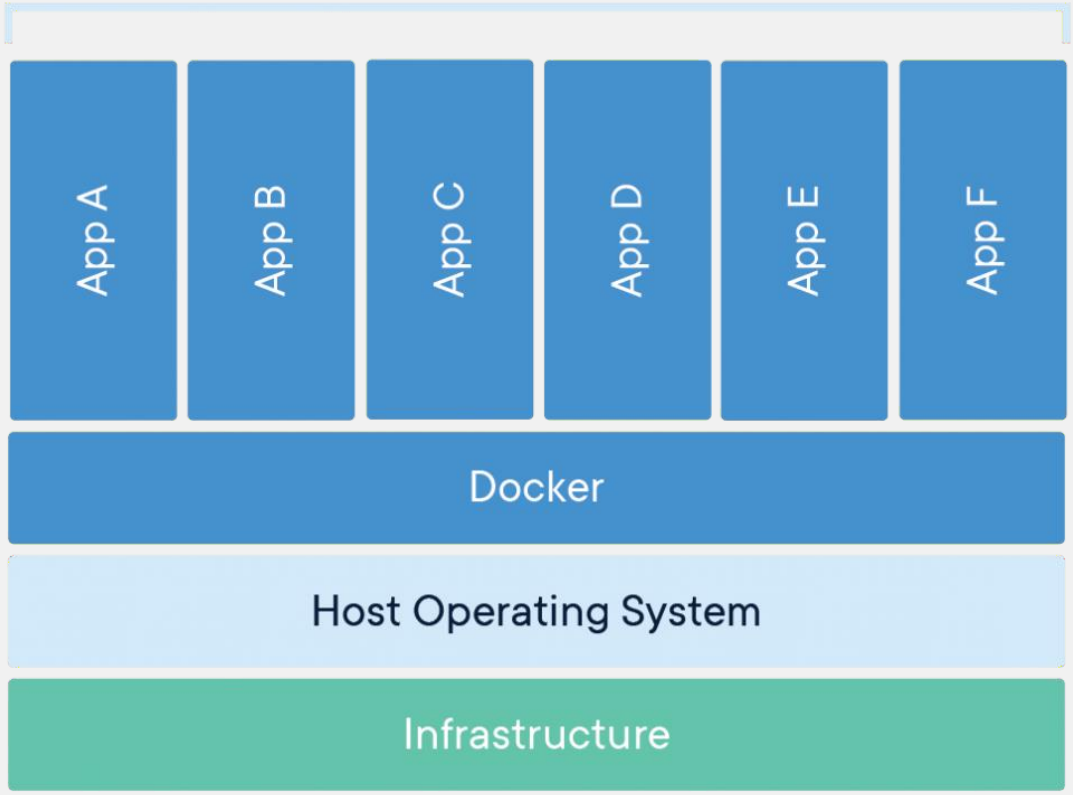
3. Better disaster recovery: **Easily replicate the file to the new hardware.**

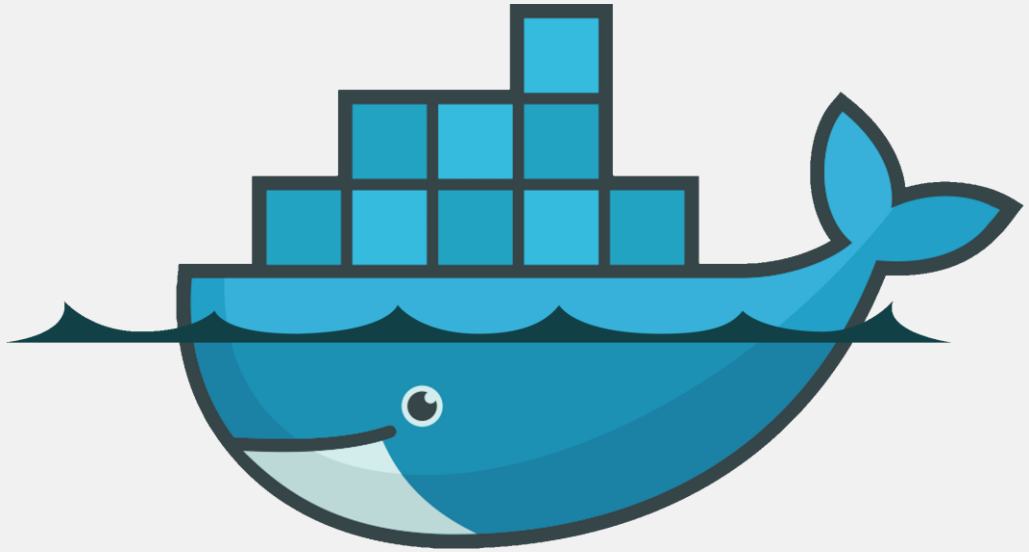
4. Improvement in adoption of DevOps: **Docker simplifies DevOps by standardizing the configuration interface and makes machine setup simpler**

What is a Container

A standardized unit of software

Containerized Applications

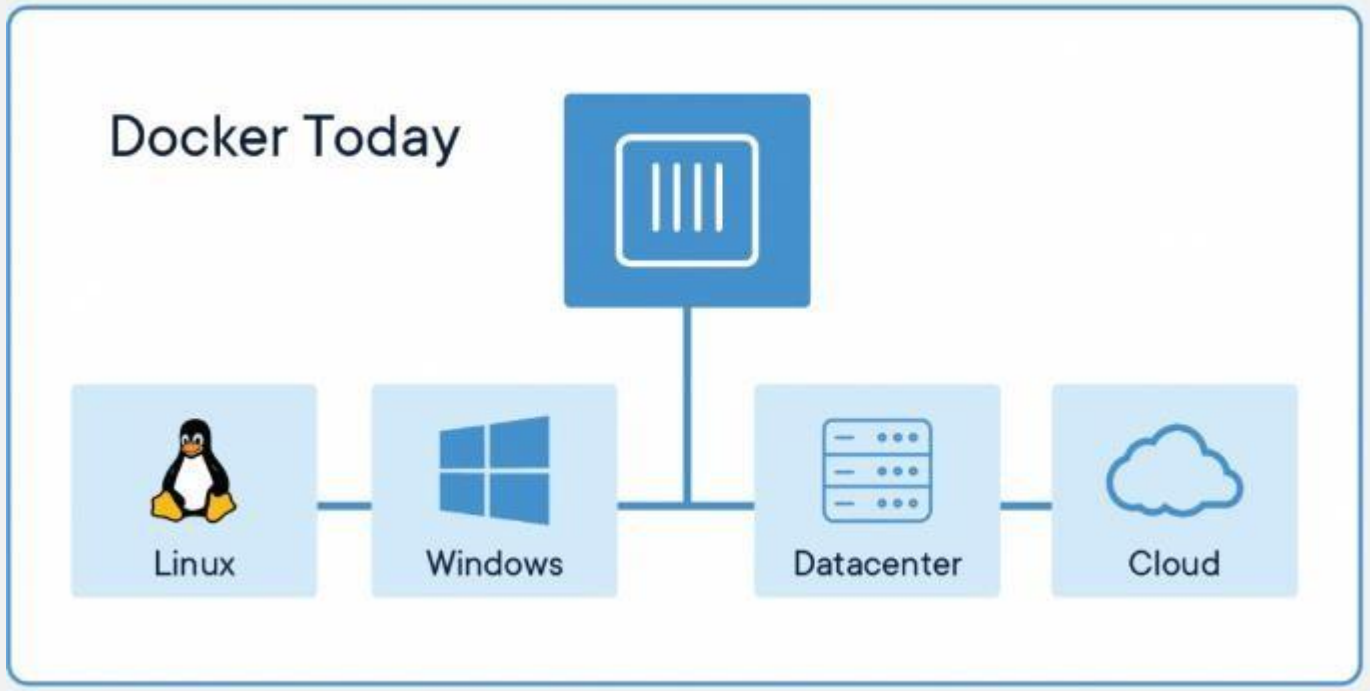




A container is a standard unit of software that packages up code and all its dependencies so the application runs quickly and reliably from one computing environment to another. A Docker container image is a lightweight, standalone, executable package of software that includes everything needed to run an application: code, runtime, system tools, system libraries and settings.

Docker containers that run on Docker Engine:

- **Standard:** Docker created the industry standard for containers, so they could be portable anywhere
- **Lightweight:** Containers share the machine's OS system kernel and therefore do not require an OS per application, driving higher server efficiencies and reducing server and licensing costs
- **Secure:** Applications are safer in containers and Docker provides the strongest default isolation capabilities in the industry



docker

docker

docker

Docker Installation:

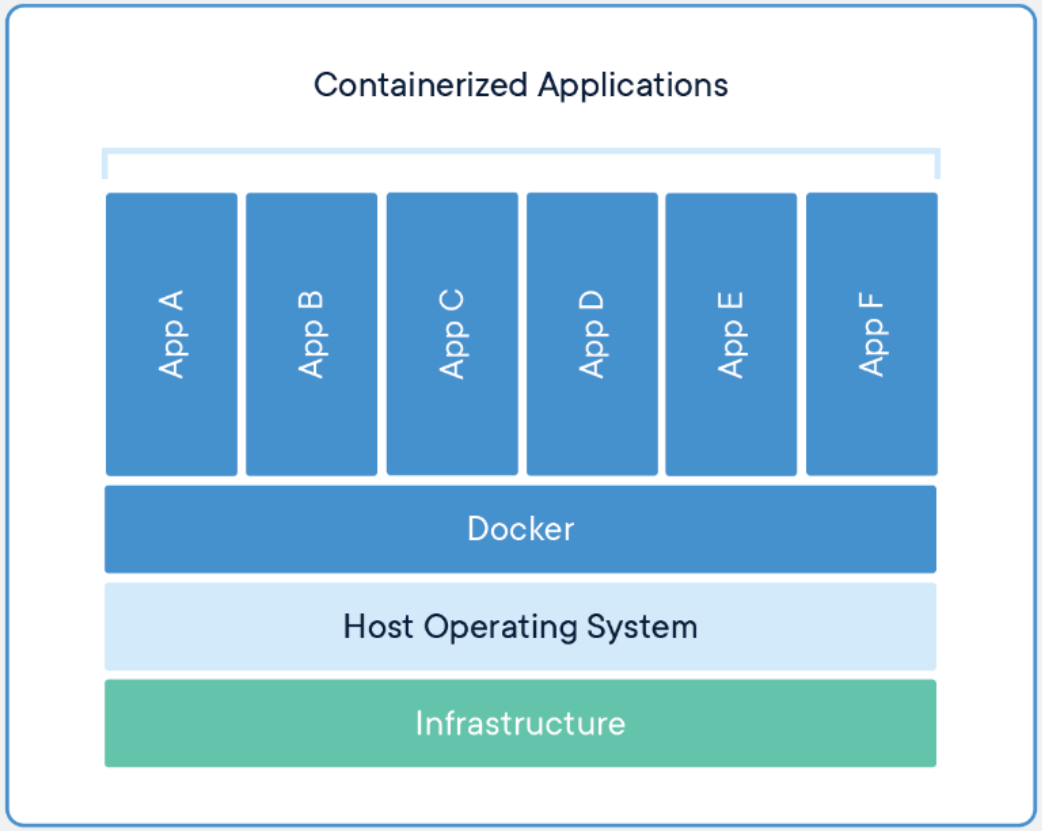
URL:https://docs.docker.com/engine/install/centos/

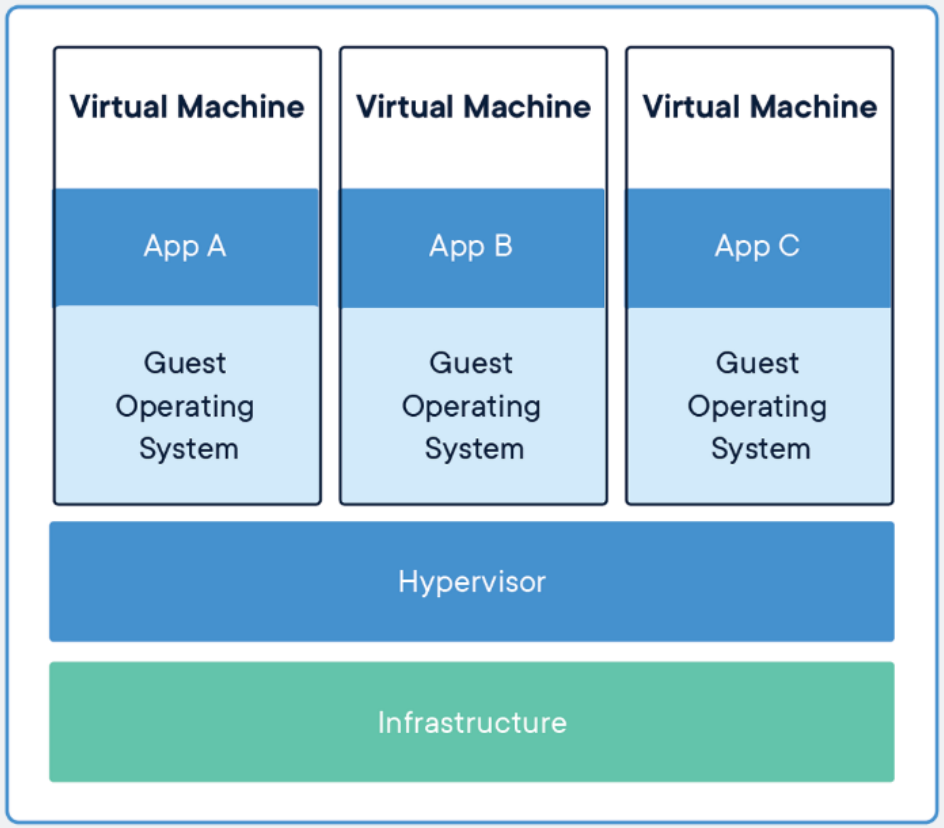
Commands:

```
#  service docker status
#  service docker stop
#  docker run hello-world
#  docker images ls
#  docker container ps
# docker container ps –a
# docker –version
#  docker version
```

Comparing Containers and Virtual Machines

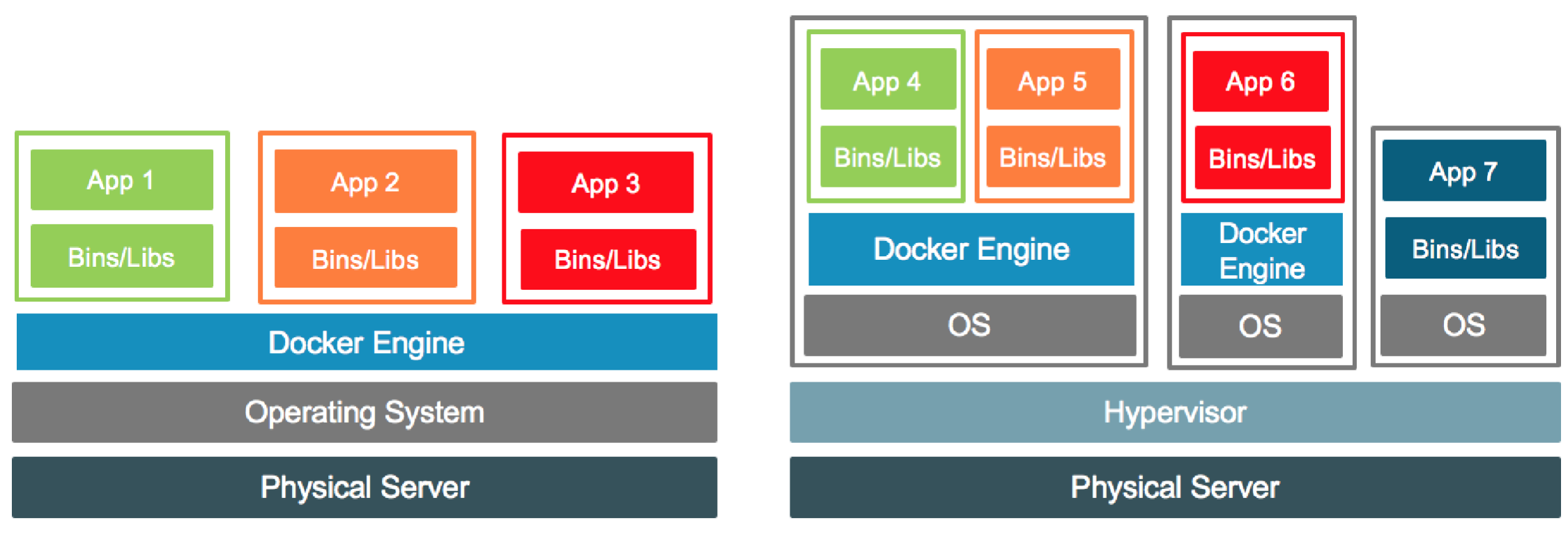
Containers and virtual machines have similar resource isolation and allocation benefits, but function differently because containers virtualize the operating system instead of hardware. Containers are more portable and efficient.



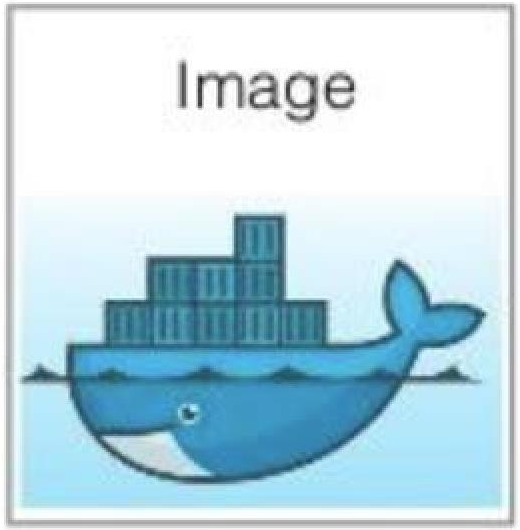


Containers and Virtual Machines Together

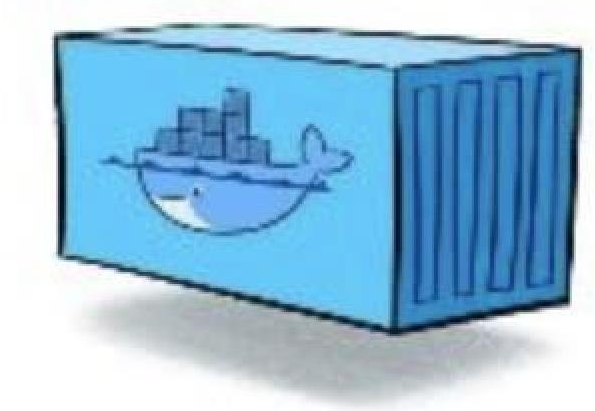
Containers and VMs used together provide a great deal of flexibility in deploying and managing app



build



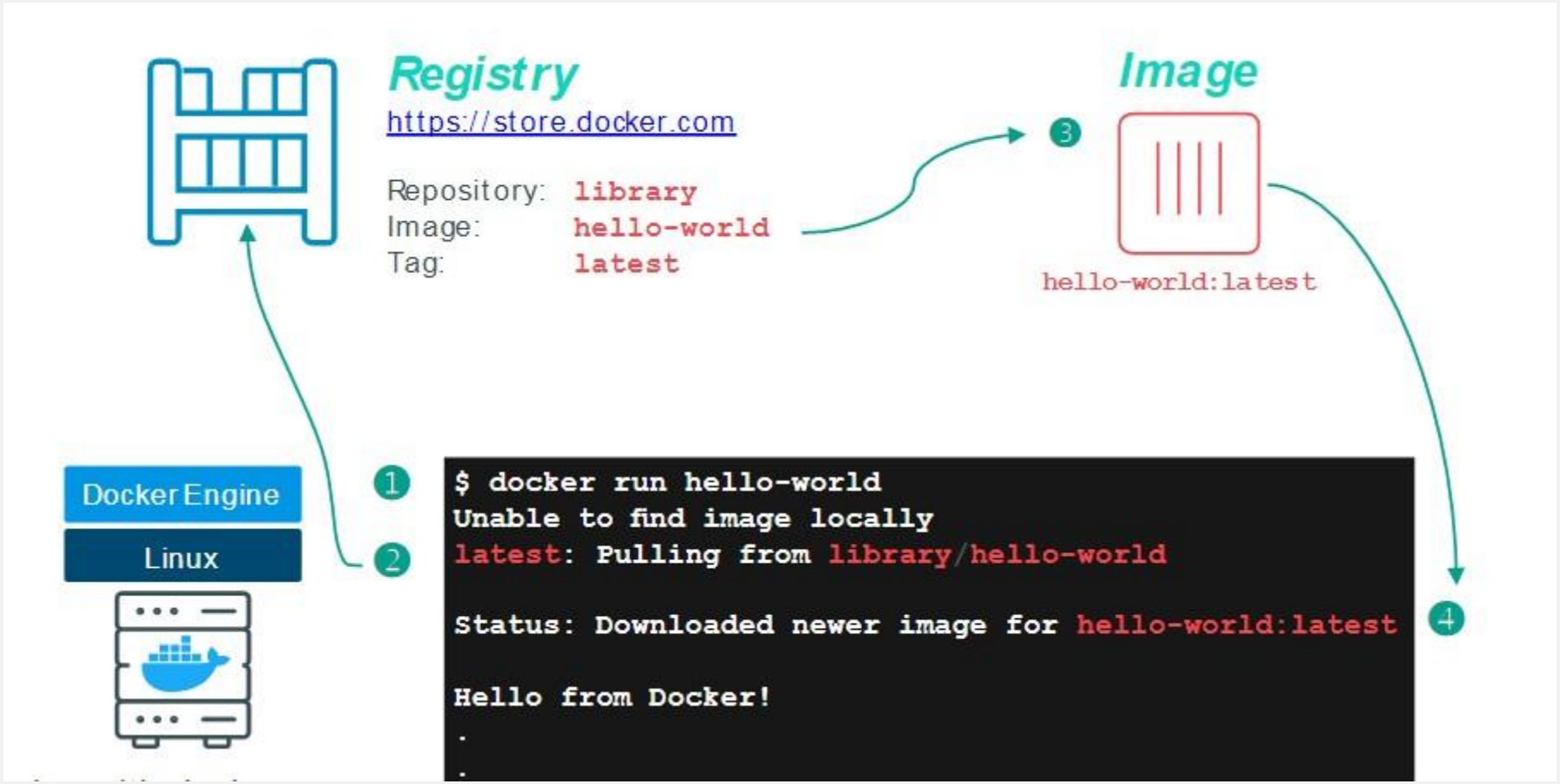
run



Dockerfile

Docker Image

Docker Container


Registry
https://store.docker.com
Repository: library
Image: hello-world
Tag: latest
Image
3
hello-world:latest
Docker Engine
Linux
1
2
$ docker run hello-world
Unable to find image locally
latest: Pulling from library/hello-world
Status: Downloaded newer image for hello-world:latest
Hello from Docker!
.
.
4