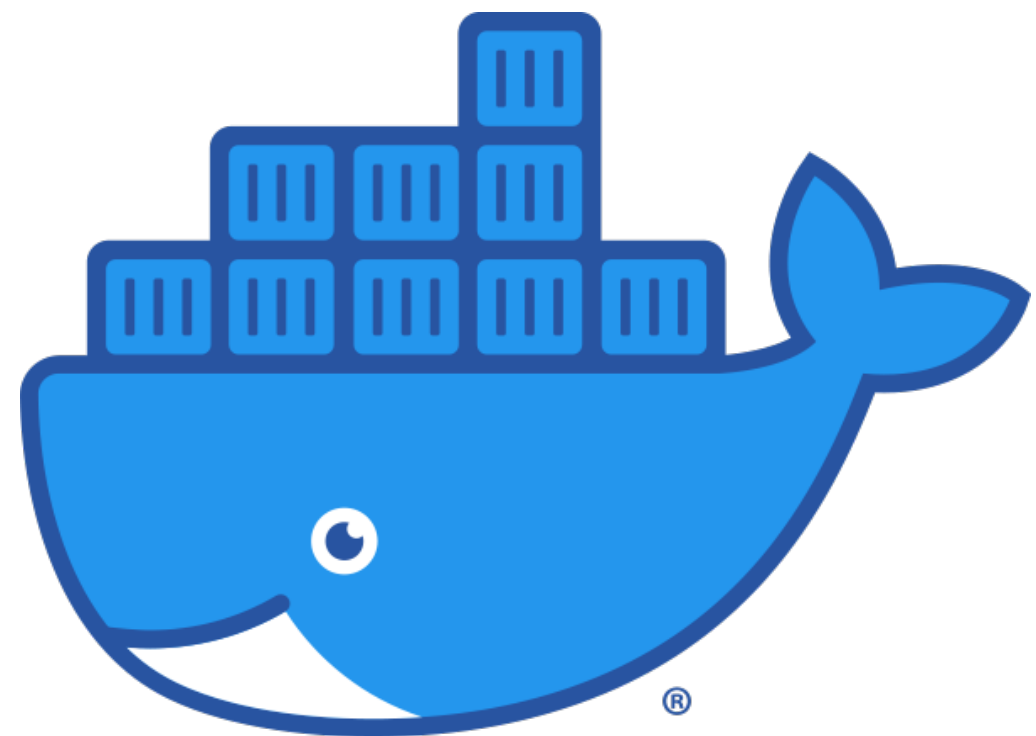


<https://www.docker.com/>

- “the de facto developer standard for building and sharing apps that enable simplicity, agility and choice for software development across any infrastructure...”



Basic links

<https://www.docker.com/get-started>

Get Started with Docker

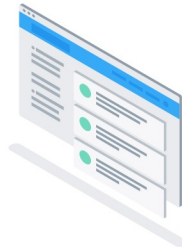
We have a complete container solution for you – no matter who you are and where you are on your containerization journey.



Docker Desktop

Developer productivity tools and a local Kubernetes environment.

Download for
Mac – Intel
Chip



Docker Hub

Cloud-based application registry and development team collaboration services.

Signup



Get started

Learn Docker basics and the benefits of containerizing your applications.



Download and install

Download and install Docker on your machine in a few easy steps.



Guides

Learn how to set up your Docker environment and start containerizing your applications.



Language-specific guides

Learn how to containerize language-specific applications using Docker.



Manuals

Browse through the manuals and learn how to use Docker products.



Reference

Browse through the CLI and API reference documentation.

<https://docs.docker.com>

How to follow the course

- Create your own Docker ID at <https://hub.docker.com/signup>
- And...

Option 1:

- You can download and install Docker on multiple platforms.
- Refer to the following link: <https://docs.docker.com/get-docker/> and choose the best installation path for you.

Option 2:

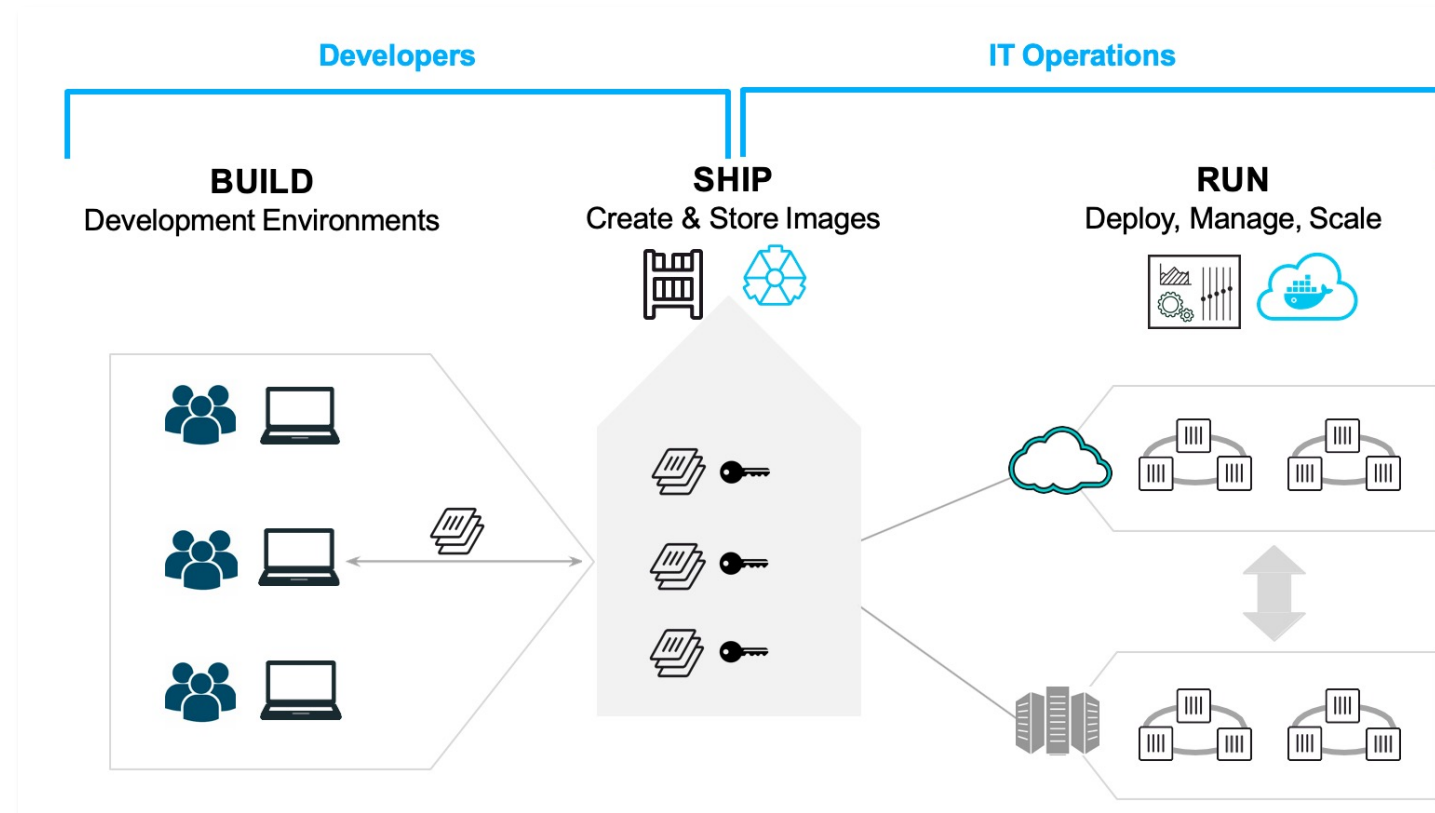
- You can use a virtual machine that you can download here:
 - https://bit.ly/dockervm_2022
- You have to use VirtualBox with the option: "Files / Import virtualized service".
- The VM has user "docker" with password "docker"

Option 3:

- You can execute it online: <https://labs.play-with-docker.com/>

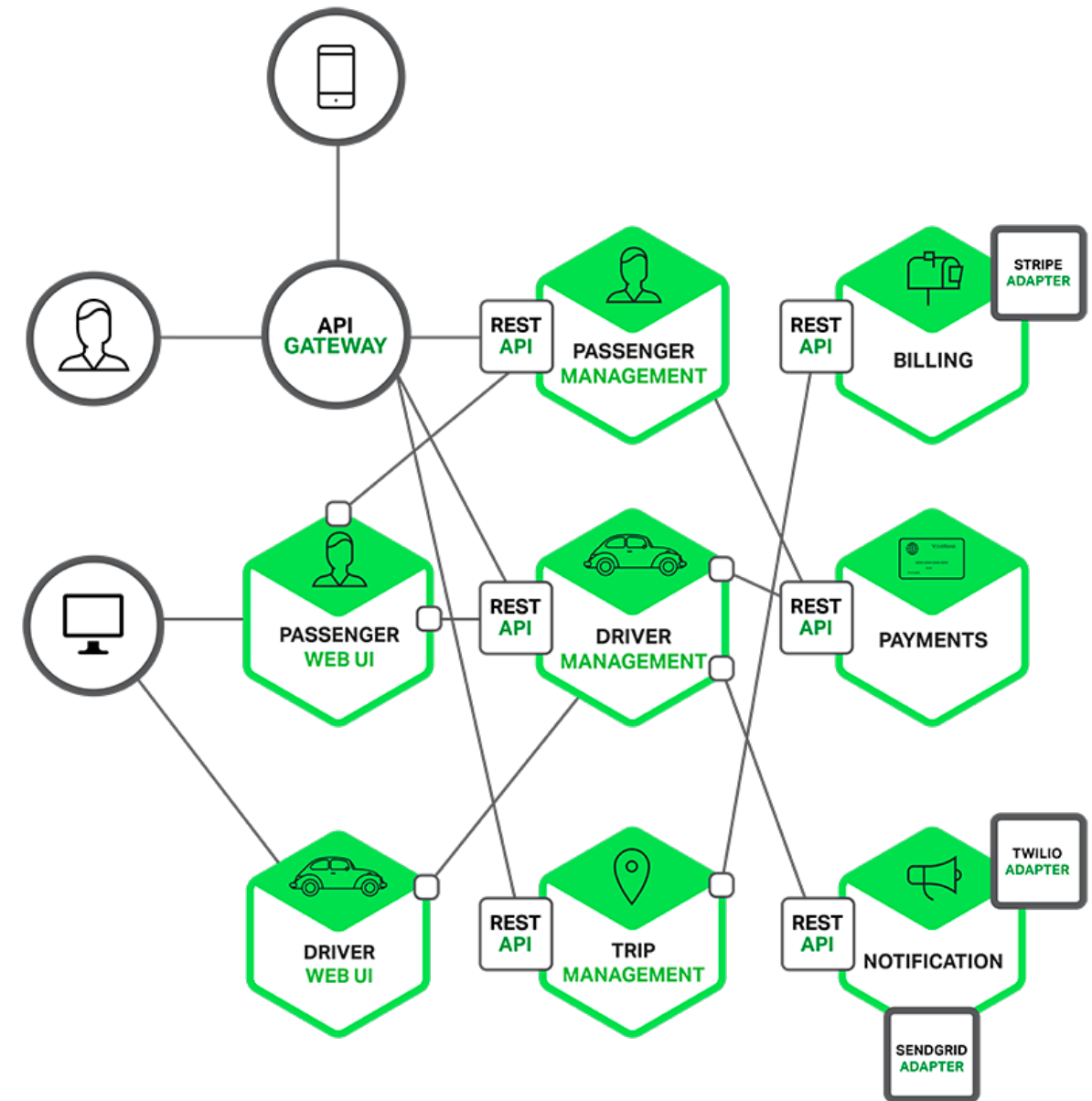
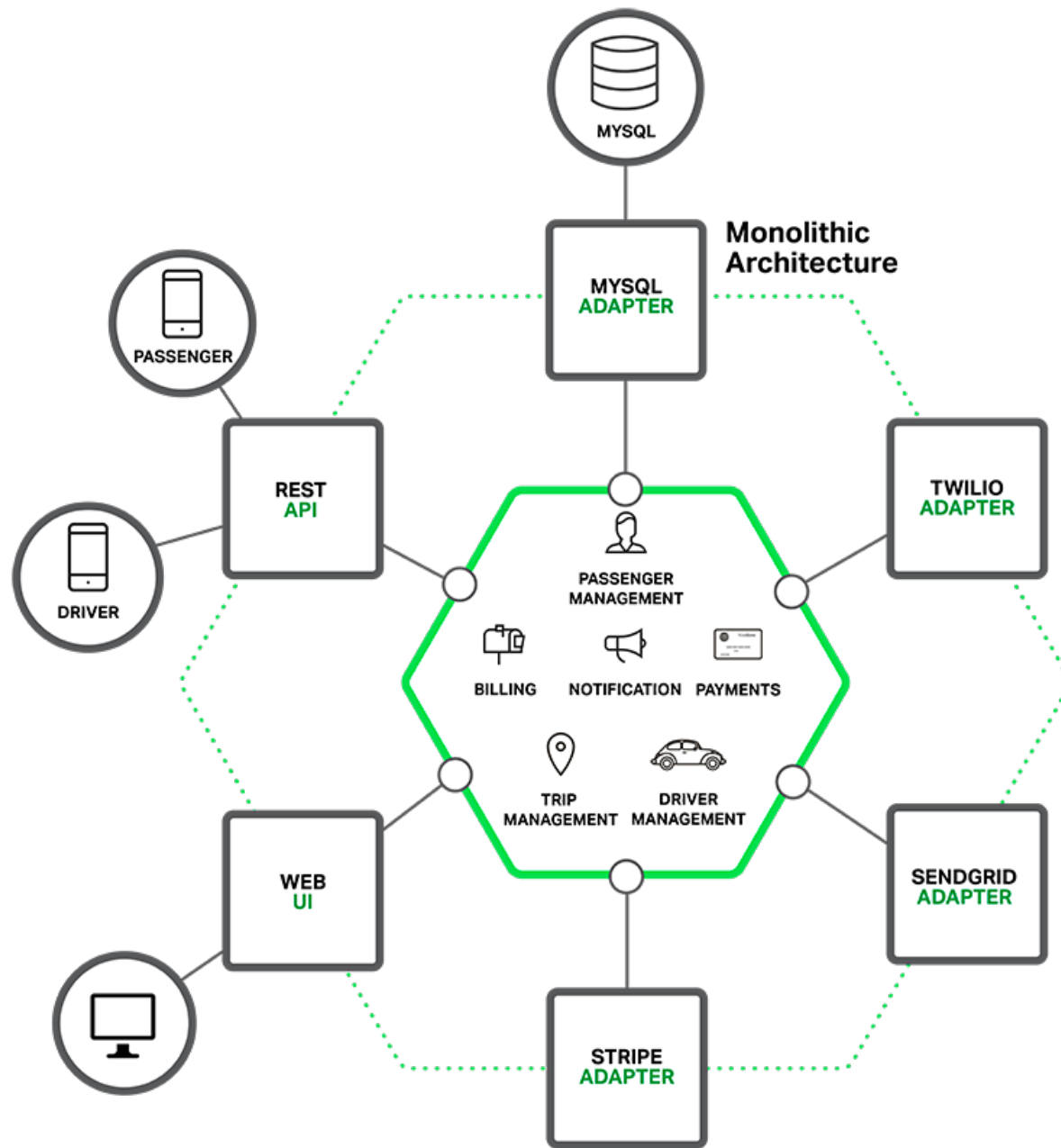
Docker overview

- Docker is an open platform for developing, shipping, and running applications.

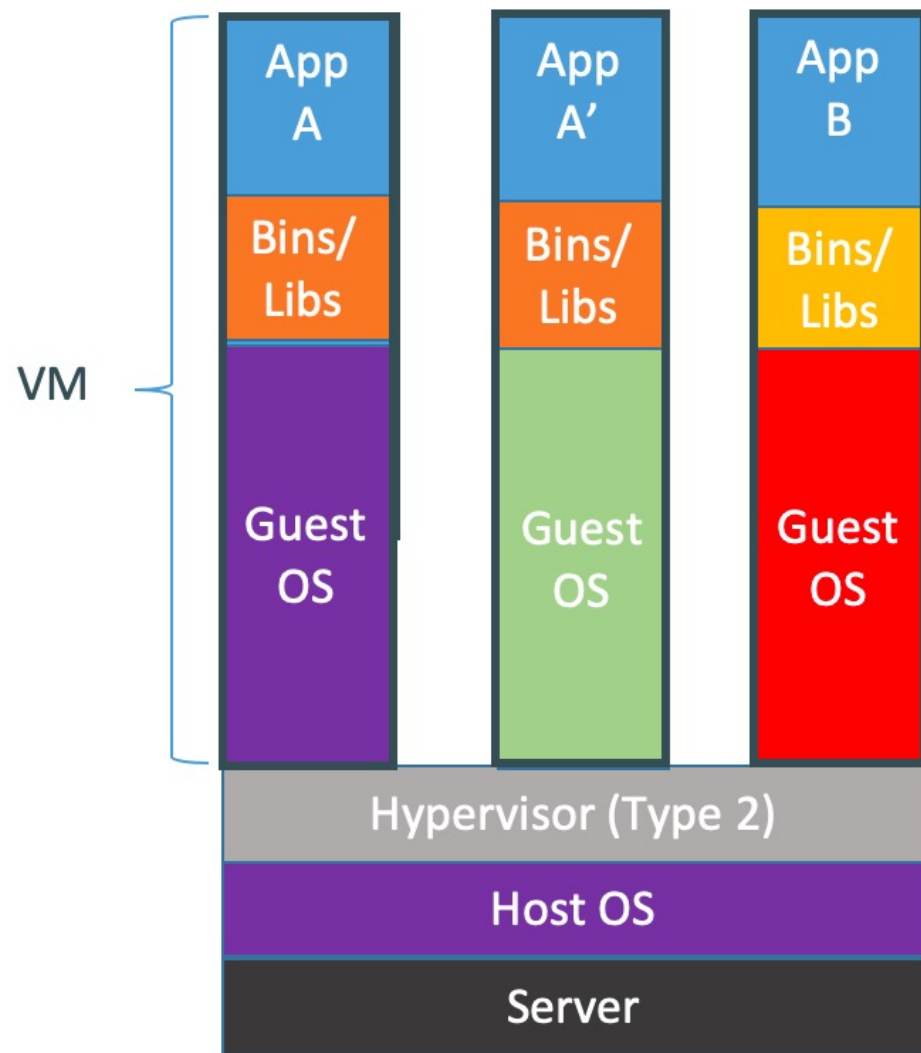


- Docker enables you to separate your applications from your infrastructure so you can deliver software quickly.
- <https://docs.docker.com/get-started/overview/>

Monolithic vs. Microservices

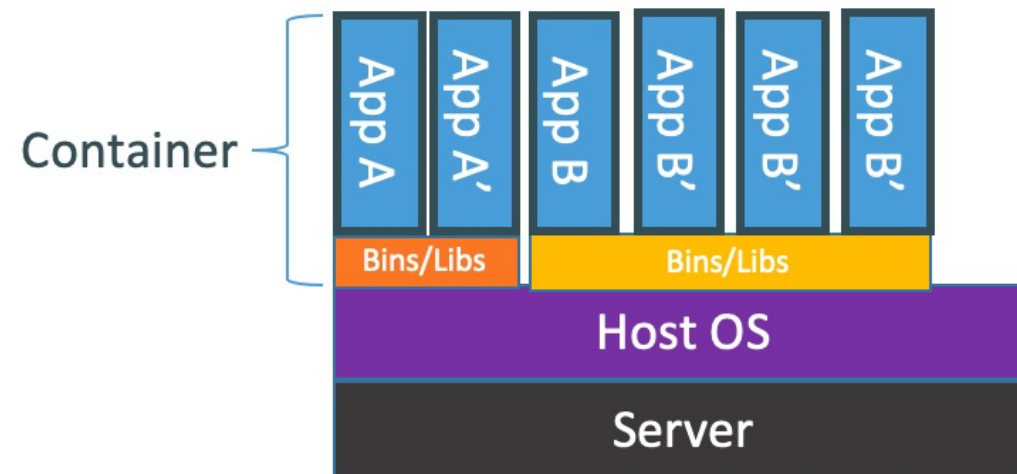


Docker vs. Virtual Machine



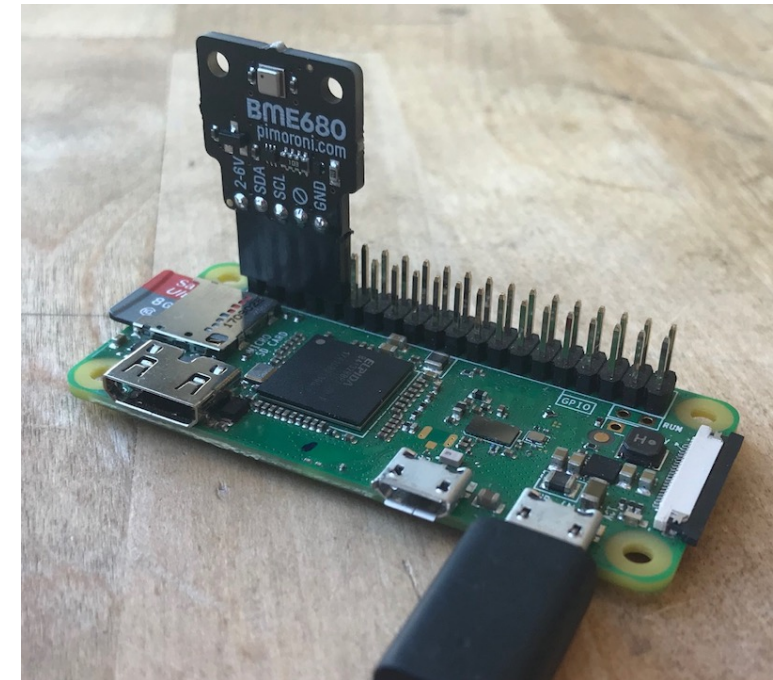
Containers are isolated, but share OS kernel and, where appropriate, bins/libraries

...result is significantly faster deployment, much less overhead, easier migration, faster restart



Deployment and scaling

- Docker containers can run on a developer's local laptop, on physical or virtual machines in a data center, on cloud providers, or in a mixture of environments.
- Docker's portability and lightweight nature also make it easy to dynamically manage workloads, scaling up or tearing down applications and services as business needs dictate, in near real time.



<https://www.balena.io/blog/build-an-environment-and-air-quality-monitor-with-raspberry-pi/>



<https://blog.alexellis.io/getting-started-with-docker-on-raspberry-pi/>

Docker containers on embedded devices

<https://www.balena.io/>

The image shows a screenshot of the Balena website. The main page has a blue header with the Balena logo and navigation links: "What is balena?", "balenaCloud", "More products", "Resources", "Pricing", and "Custom". The main content area has a large blue background with the text "Build your IoT project with balena." and a subtext "We provide a full technology stack to help you develop, deploy, and manage projects at any scale." with a "Learn more" button. Below this is an illustration of IoT devices connected to a central hub. On the right, a sidebar lists several products with icons and descriptions:

- openbalena** → Open source software to manage IoT devices
- balenaOS** → Run Docker containers on embedded devices
- balenaEtcher** → Flash OS images to SD cards & USB drives
- balenaEngine** → A Container Engine for IoT, based on Moby technology from Docker
- balenaFin** → A carrier board for the Raspberry Pi Compute Module

Docker containers on embedded devices

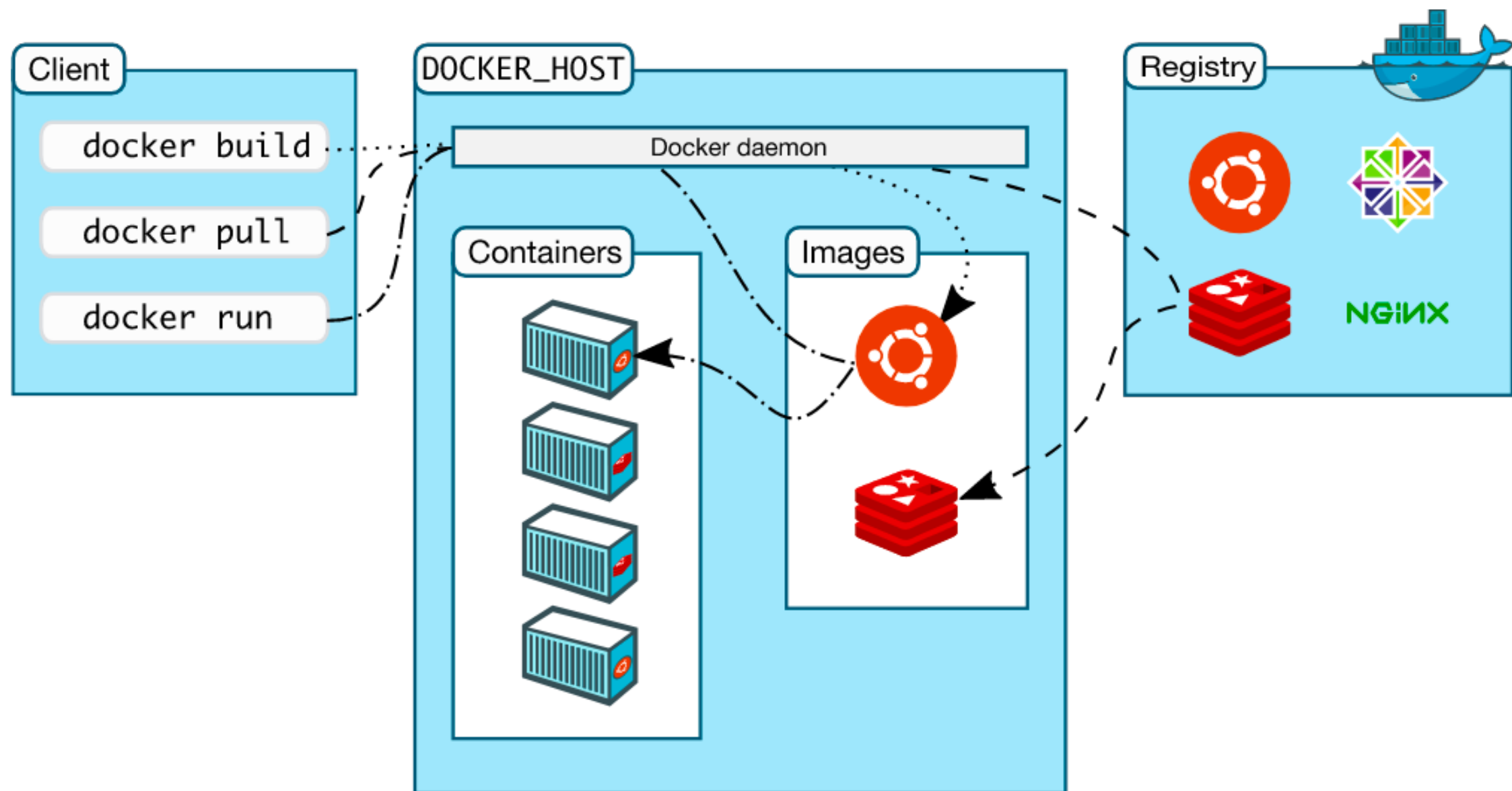
<https://toit.io>

The image shows the Toit website and a preview of its dashboard. The website has a yellow background with the Toit logo in the top left. Navigation links include Product, Pricing, Developers, and Company. There are buttons for Sign in and Start now. The main heading reads "Cloud-managed containers on the ESP32". Below this, a subheading states: "Secure the code on your ESP32 microcontrollers with lightweight containers and orchestrate them through our cloud API." A large "Start now" button is centered, with a callout box below it saying "Free forever on 10 devices".

The dashboard preview at the bottom shows a sidebar with navigation options: Dashboard, Devices (selected), Applications, Serial, Organization, and System. The main content area is titled "Devices" and includes a search bar and a table of devices. The table has columns for Device, Firmware, Last seen, Id, and Status. One device is listed: "Thermal Sensor-1" with firmware "v1.0.2", last seen "Sep 15 09:09", id "532728ff-2d2...", and status "Healthy".

Device	Firmware	Last seen	Id	Status
Thermal Sensor-1	v1.0.2	Sep 15 09:09	532728ff-2d2...	Healthy

Docker architecture

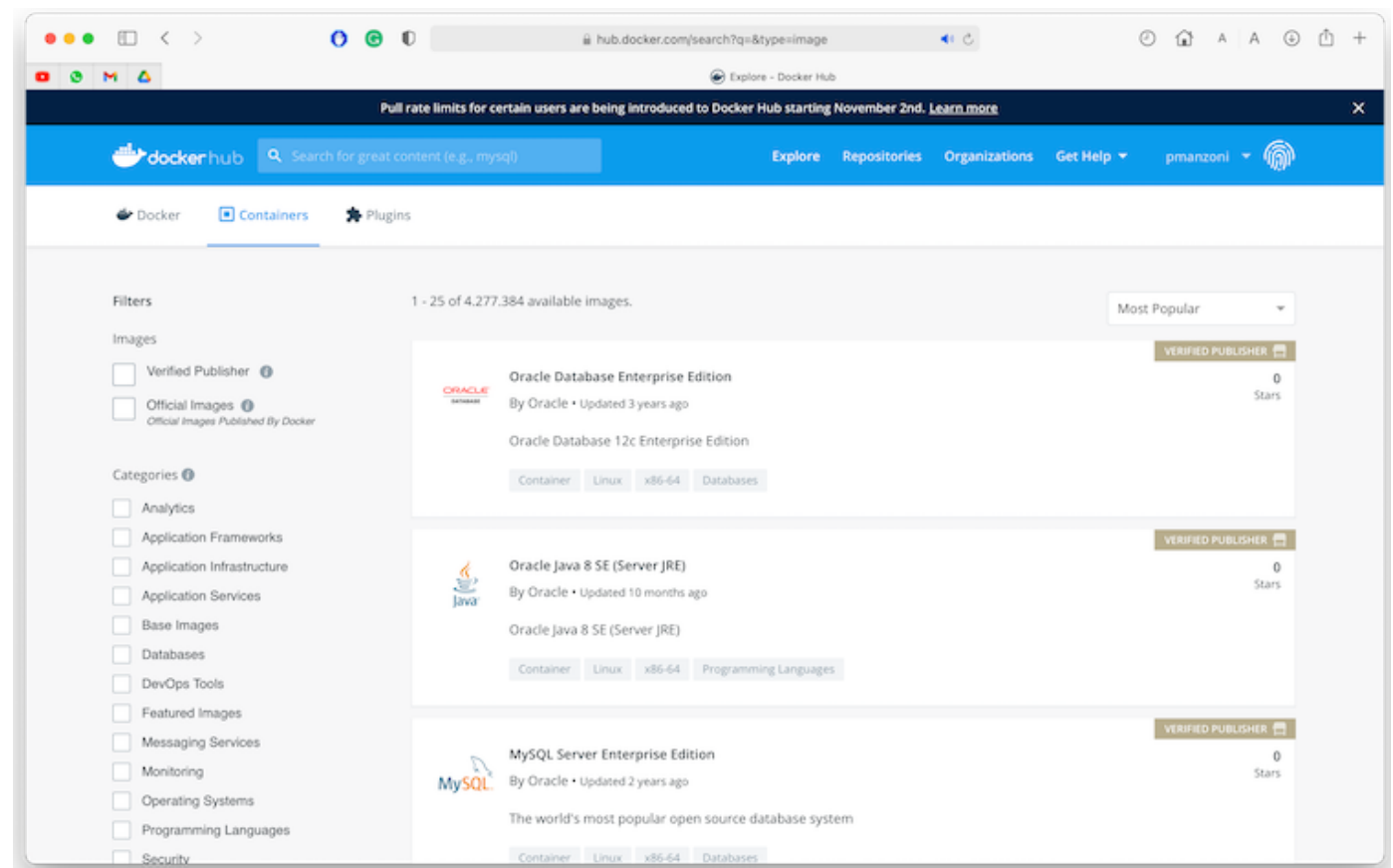


Docker registries

- A Docker registry stores Docker images. Docker Hub (<https://hub.docker.com/>) is a public registry that anyone can use, and Docker is configured to look for images on Docker Hub by default.
 - You can run your own private registry, too, e.g., <https://docs.github.com/en/packages/guides/about-github-container-registry>

But also:

- Amazon EC2 Container Service.
- Microsoft Azure Container Service.
- Google Container Engine for Docker Containers.



Images

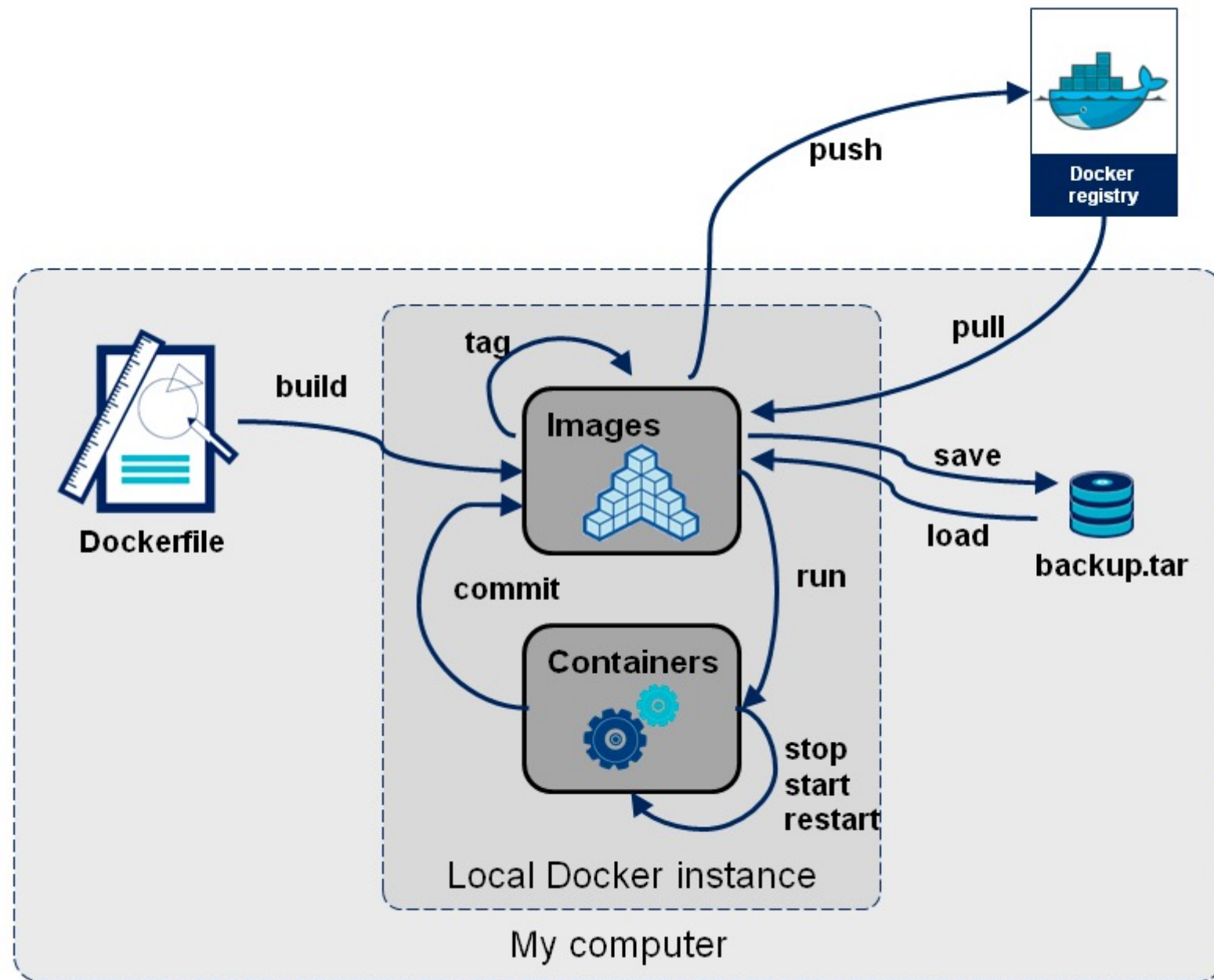
- An image is a read-only template with instructions for creating a Docker container. Often, an image is based on another image, with some additional customization.
 - For example, you may build an image which is based on the ubuntu image, but installs the Apache web server and your application, as well as the configuration details needed to make your application run.
- You might create your own images or you might only use those created by others and published in a registry.
- To build your own image, you create a Dockerfile with a simple syntax for defining the steps needed to create the image and run it.

```
FROM alpine:3.5
RUN apk add --update py2-pip
RUN pip install paho-mqtt
COPY sisub.py /home/
CMD ["python", "/home/sisub.py"]
```

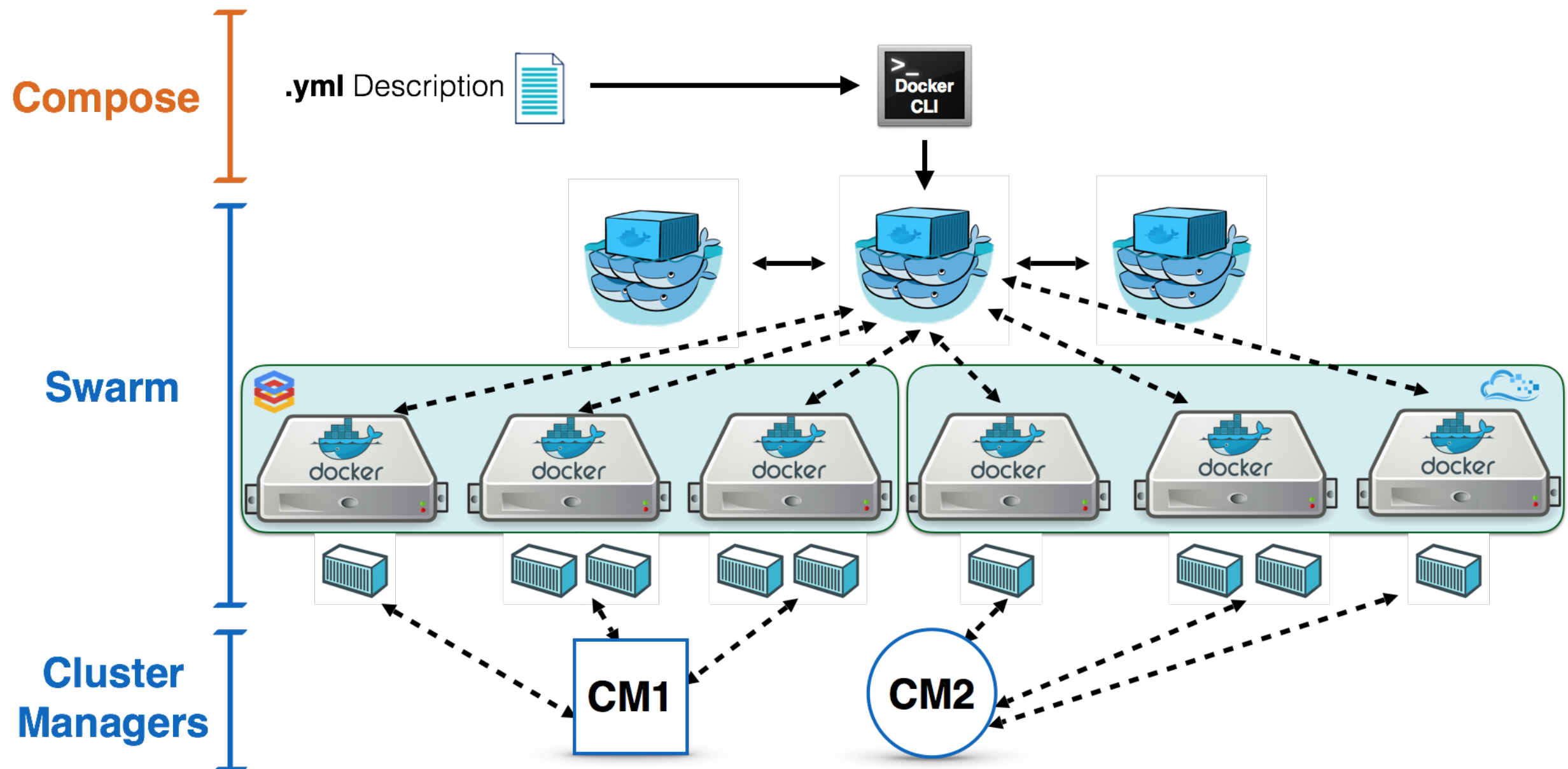

Containers

- A container is a **runnable instance of an image**. You can create, start, stop, move, or delete a container using the Docker API or CLI. You can connect a container to one or more networks, attach storage to it, or even create a new image based on its current state.
- By default, a **container is relatively well isolated from other containers and its host machine**.
- A container is defined by its image as well as any configuration options you provide to it when you create or start it. **When a container is removed, any changes to its state that are not stored in persistent storage disappear.**

Manage the lifecycle of your containers



Docker Swarm



Kubernetes

- Developed by Google and introduced in the year 2014.
- An IT management tool that has been specifically designed to simplify the scalability of workloads using containers.
- It has the ability to automate deployment, scaling, and operating application containers.

