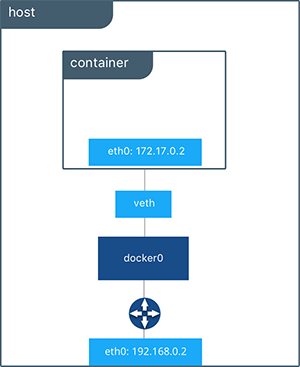
# Docker Networking

Docker containers and services are so powerful is that you can connect them together, or connect them to non-Docker workloads.

peers are also Docker workloads or not. Whether your Docker hosts run Linux, Windows, or a mix of the two, you can use Docker to manage them in a platform-agnostic way.

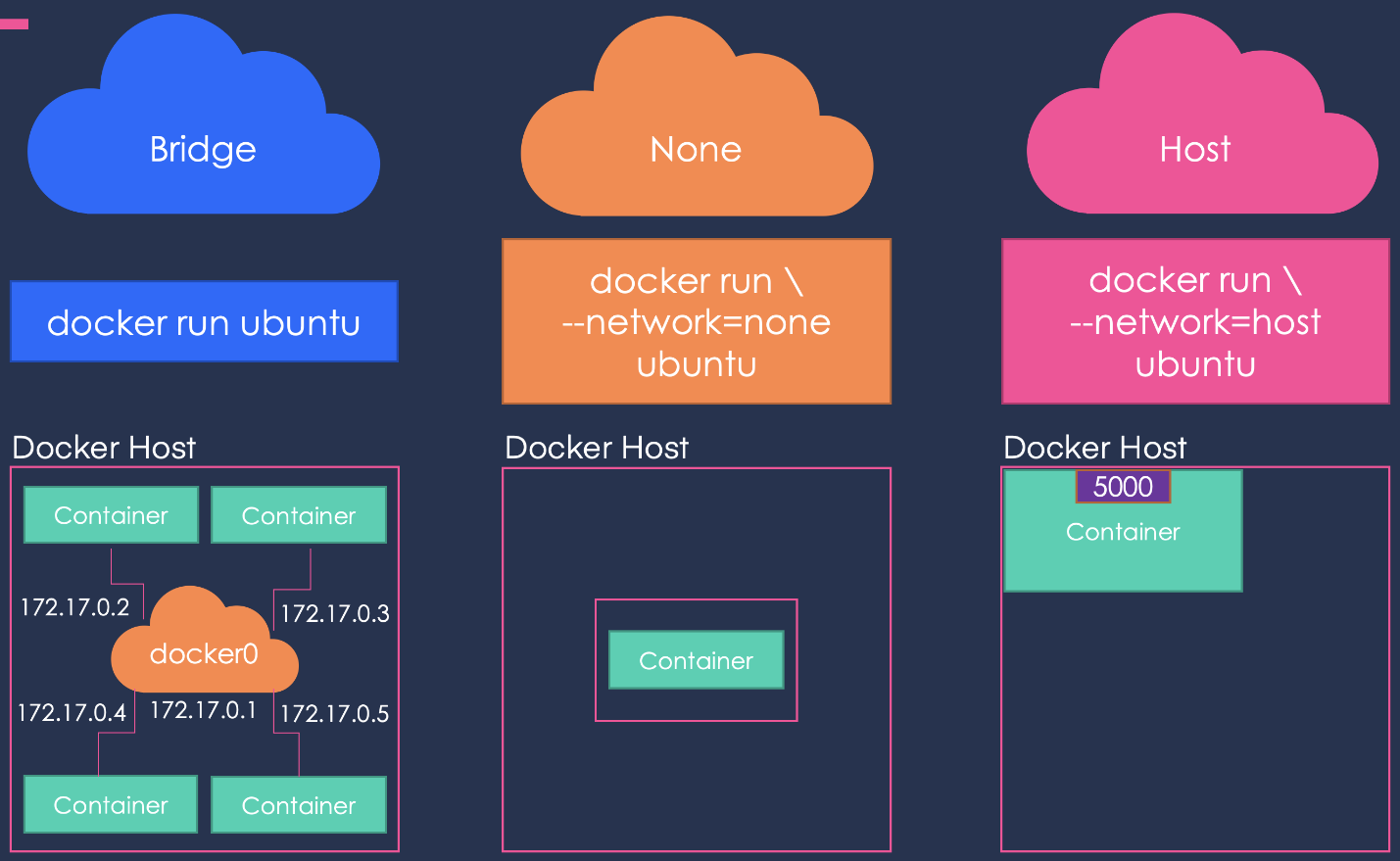


bridge:

* The default network driver. Bridge networks are usually used when your applications run in standalone containers that need to communicate.
* In terms of networking, a bridge network is a Link Layer device which forwards traffic between network segments.
* A bridge can be a hardware device or a software device running within a host machine’s kernel.

## host:

For standalone containers, remove network isolation between the container and the Docker host, and use the host’s networking directly. See use the host network.



overlay:

Overlay networks connect multiple Docker daemons together and enable swarm services to communicate with each other. You can also use overlay networks to facilitate communication between a swarm service and a standalone container, or between two standalone containers on different Docker daemons. This strategy removes the need to do OS-level routing between these containers. See overlay networks.

ipvlan:

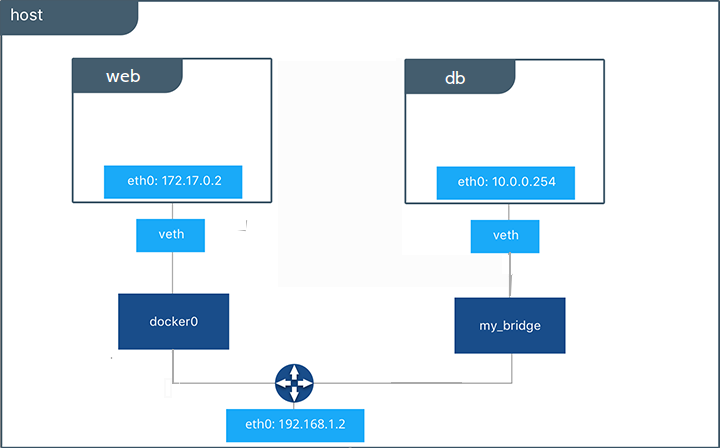
IPvlan networks give users total control over both IPv4 and IPv6 addressing. The VLAN driver builds on top of that in giving operators complete control of layer 2 VLAN tagging and even IPvlan L3 routing for users interested in underlay network integration. See IPvlan networks.

## macvlan:

Macvlan networks allow you to assign a MAC address to a container, making it appear as a physical device on your network. The Docker daemon routes traffic to containers by their MAC addresses. Using the macvlan driver is sometimes the best choice when dealing with legacy applications that expect to be directly connected to the physical network, rather than routed through the Docker host’s network stack. See Macvlan networks.

## none:

For this container, disable all networking. Usually used in conjunction with a custom network driver. none is not available for swarm services. See disable container networking.



## Network driver summary

* User-defined bridge networks are best when you need multiple containers to communicate on the same Docker host.
* Host networks are best when the network stack should not be isolated from the Docker host, but you want other aspects of the container to be isolated.
* Overlay networks are best when you need containers running on different Docker hosts to communicate, or when multiple applications work together using swarm services.
* Macvlan networks are best when you are migrating from a VM setup or need your containers to look like physical hosts on your network, each with a unique MAC address.
* Third-party network plugins allow you to integrate Docker with specialized network stacks.