# Vagrant vs Docker

**Docker** is an open platform for building, shipping, and running distributed applications.

**Vagrant** creates and configures, reproducible, and portable development environments.

Vagrant VM’s have to be launched upon the existing host OS’s, so they cost a certain amount of resources to sustain themselves. Docker negates the need for the hypervisor and the whole virtualization level, as containers can be run directly atop 3 most popular OS’s — Linux, Windows, and Mac OS. Where Docker relies on the host operating system, Vagrant includes the operating system within itself as part of the package.

Docker containers hold an application’s components, including binaries, libraries, and configurations, as well as the application’s dependencies. Vagrant is a workflow for development projects. Vagrant provisions typically a virtual machine [1].

As part of the DevOps course, my personal experience with Vagrant, and its ease of use, was more pleasant than with Dockers. For this reason alone, at this stage of my knowledge, I will prefer using Vagrant over Docker.

[1] https://medium.com/faun/vagrant-vs-docker-which-one-to-use-for-development