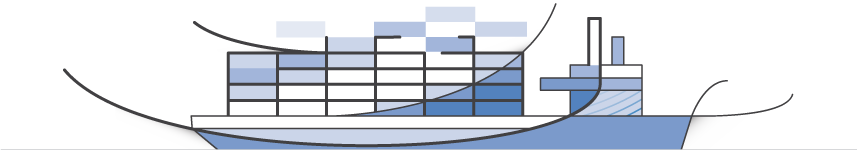
<https://www.quora.com/What-is-the-difference-between-Docker-and-Vagrant-When-should-you-use-each-one>

Vagrant and Docker are different beasts.

Vagrant abstracts the machine where as Docker abstracts the application.

1. Docker builds an image with necessary tools/frameworks/softwares (Ex: JDK, tomcat, maven repositories, Jenkins all installed and application deployed on tomcat. Starting image will start tomcat) to run our enterprise applications.
2. The image can be deployed on any container.
3. Docker also provides container to run the image.
4. A container acts as a virtual machine that contains or hosts your images. A container is just a service that runs on top of a machine. **You can run many images in one container and you can run many containers on one machine.**

**Practical Docker Use Case**

Docker on AWS provides a highly reliable, low-cost way to quickly build, run, test, and deploy distributed applications at any scale. In below image ship and containers are provided by AWS, we can place anything inside a container. 

1. Vagrant uses virtualbox to boot a virtual machine. Then you can set up your own environment and install everything you need on that machine mostly through some provisioning scripts like Chef.
2. Its most common use case is to test your application on different environments/test servers/machines where your teammates can ssh to and test their work on where everyone else can see and use.
3. We have used Vagrant in Shaklee to test our application on different servers.
4. Say goodbye to the "works on my machine" excuse as Vagrant creates identical development environments for everyone on your team.