# What is Docker and what is it used for?

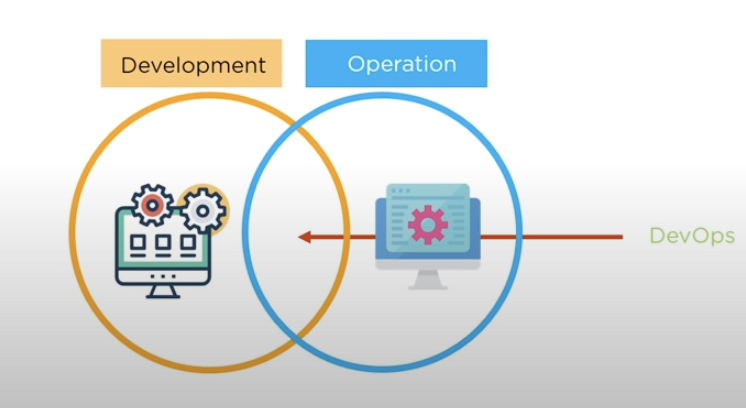
Docker is a virtual environment used to run programs.

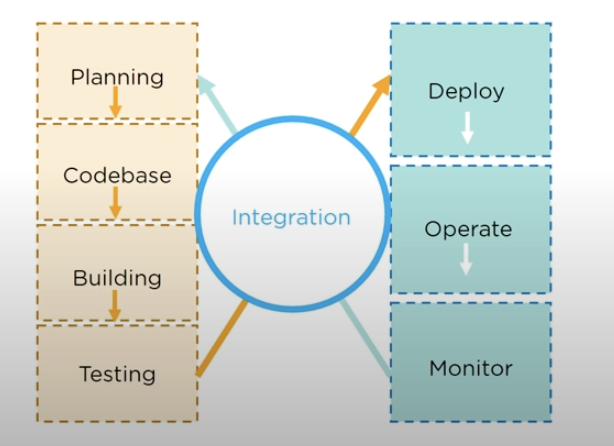
Docker vs Virtual Machine

|  |  |  |
| --- | --- | --- |
| Criteria | Virtual Machine | Docker |
| OS Support | Occupies a lot of memory space | Docker Containers occupy less space |
| Boot-up time | Long boot-up time | Short boot-up time |
| Performance | Running multiple virtual machines leads to unstable performance | Containers have a better performance as they are hosted in a single Docker engine |
| Scaling | Difficult to scale up | Easy to scale up |
| Efficiency | Low efficiency | High efficiency |
| Portability | Compatibility issues while porting across different platforms | Easily portable across different platforms |
| Space allocation | Data volumes cannot be shared | Data volumes can be shared and reused among multiple containers |

## What is DevOps and its tools

DevOps is a collaboration between development and operation teams which enables continuous delivery of applications and services to our end users.





For planning and Codebase we use tools like Jira & Git.

For Building we use Maven and Gradle

Testing Selenium

Deploy and operation Docker & Chef

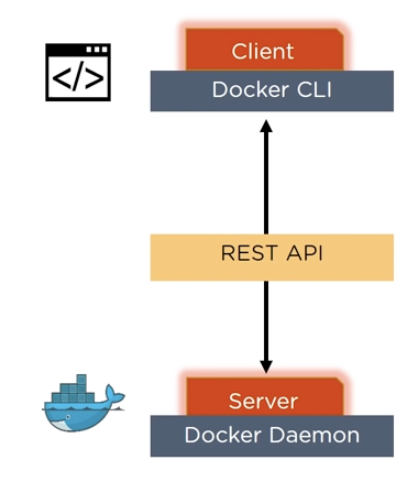
Monitor -> Nagios

## What is Docker?

Docker is a tool which is used to automate the deployment of applications in lightweight containers so that applications can work efficiently in different environments.

Container is a software package that consists of all the dependencies required to run an application.

* Multiple applications can run on same hardware.
* Maintains isolated applications
* High productivity
* Quick and easy configuration



Docker Engine is the base engine installed on host machine to build and run containers using Docker components and services.

It uses a client-server architecture

Docker Client and Server communicate using Rest API.

Docker is a service which runs a command, which is translated using REST API and is sent to the Docker Daemon (server).

Then the server checks the client request and interacts with the operating system in order to create or manage containers.

## Components of Docker

