**What is Docker?**

* Docker is a popular open-source project written in **go** and developed by **DotCloud** (A PaaS Company).
* It is basically a container engine that uses the **Linux Kernel** features like **namespaces** and **control groups** to create containers on top of an operating system.

**NOTE:**

1. Namespaces are responsible for containers to have their own mount points, user, IP address, process management, etc., Essentially it sets boundaries for the containers.
2. When we start a service, we don’t specify any memory or CPU limit. We leave it to the kernel to prioritize and allocate resources for the services. However, you can explicitly set CPU, memory limits for your services using a Linux kernel feature called CGroups. It is not a straight forward approach; you need to make some extra configurations and tweaks to make it work.

* Docker has well-defined wrapper components that make **packaging applications easy**. Before the Docker, it was not easy to run containers. Meaning, it does all the work to decouple your application from the infrastructure by packing all application system requirements into a container.

**For example**, if you have a Java jar file, you can run it on any server which has java installed. Same way, once you package a container with required applications using Docker, you can run it on any other host which has docker installed.

**Difference between Docker and Container:**

Docker is a technology or a tool developed to manage container implementations efficiently.

So, can I run a container without Docker?

Yes! of course. you can use LXC technology to run containers on Linux servers.

Things you should know about Docker:

1. Docker is not LXC
2. Docker is not a Virtual Machine Solution.
3. Docker is not a configuration management system and is not a replacement for chef, puppet, Ansible, etc.
4. Docker is not a platform as a service technology.

Docker Architecture:



