

前提知识，linux 命令， Git

Docker是什么：

出现原因：传统的是开发奖代码开发好后， 传给运维然后运维进行部署，此时就会出现问題，在开发这代码可以运行，但是在运维那边那就不能运行了，而产生开发和运维的“相爱相杀”，产生这个问题的原因就是二者环境和配置的不同。

解决的方法，开发将自己的代码、开发环境、配置、数据依赖等全部传给运维，也就是，整体环境，开发环境和运维环境全部一样，以前只给运维发代码，而现在全部一整套全部都发给运维。

而这些整套的东西称为是Docker中的“镜像”。

Docker还有一个应用就是，在安装集群的时候，每一个机器上安装的东西都是一样的，这里也将这些一样的一整套东西打包为“镜像”，然后在每一台机器上安装。

跨平台无缝承接。

Docker:

解决了运行环境和配置问题**软件容器**，方便做持续集成并有助于整体发布的容器虚拟化技术。

Docker 能干什么：

虚拟机就是带环境安装的一种解决方案，虚拟机模拟整套操作系统，可以解决上面的问题，但是这里虚拟机的问题有，启动时间，会比较长。

虚拟机还有的其他问题：

- 占内存(模拟了整套系统，包括硬件系统)
- 冗余步骤多
- 启动时间慢

基于以上的问题，Linux提出另外一种虚拟化技术，Linux 容器，这并不是模拟一个完整的操作系统。用到什么加载什么，按需取数据，只需要软件工作时候需要的资源和设置，系统变得轻量，容器内没有自己的内核，而且没有进行硬核的虚拟，更加的便捷，容器之间相互隔离。

基于以上，Docker 将传统虚拟机的各个问题解决。Docker 是缩小版，精简版的虚拟机。

DevOps: 开发自运维，一次构建，随处运行，跟快速交付和部署，便捷的升级和扩缩，更简单的系统运维，更高效的计算资源利用。

Docker Hub 存储镜像(相当于 github)

Client 客户端

对Docker 操作的命令终端，

Image 镜像

Container：容器，每一个镜像的实例

Registry：注册仓库：镜像的存储地方。

Docker 的基本组成，三大块， 镜像、容器、仓库

镜像：就是模板。

容器：模板的实例。可以对容器进行添加、删除，可以将容器看做是简易版的Linux 环境，容器和镜像几乎一样，唯一的区别在于，容器的最上一层是只读的

仓库：存储模板的地方。仓库 Repository 和 仓库注册服务器 Registry 是有区别的，Registry 上往往存放着多个仓库，每个仓库包含着多个镜像，每个镜像有着不同的标签(Tag 可以理解为 版本号)

Docker 和外部的客户端有一个通信的接口 Docker Daemon，也就是Docker 后台的一个进程，大管家。