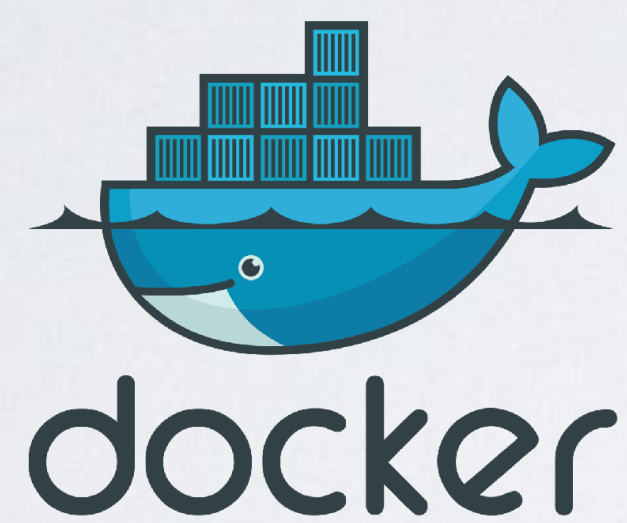


docker



DOCKER从入门到放弃

钱龙

目录



Docker是什么



为什么要学习Docker



Docker常见概念

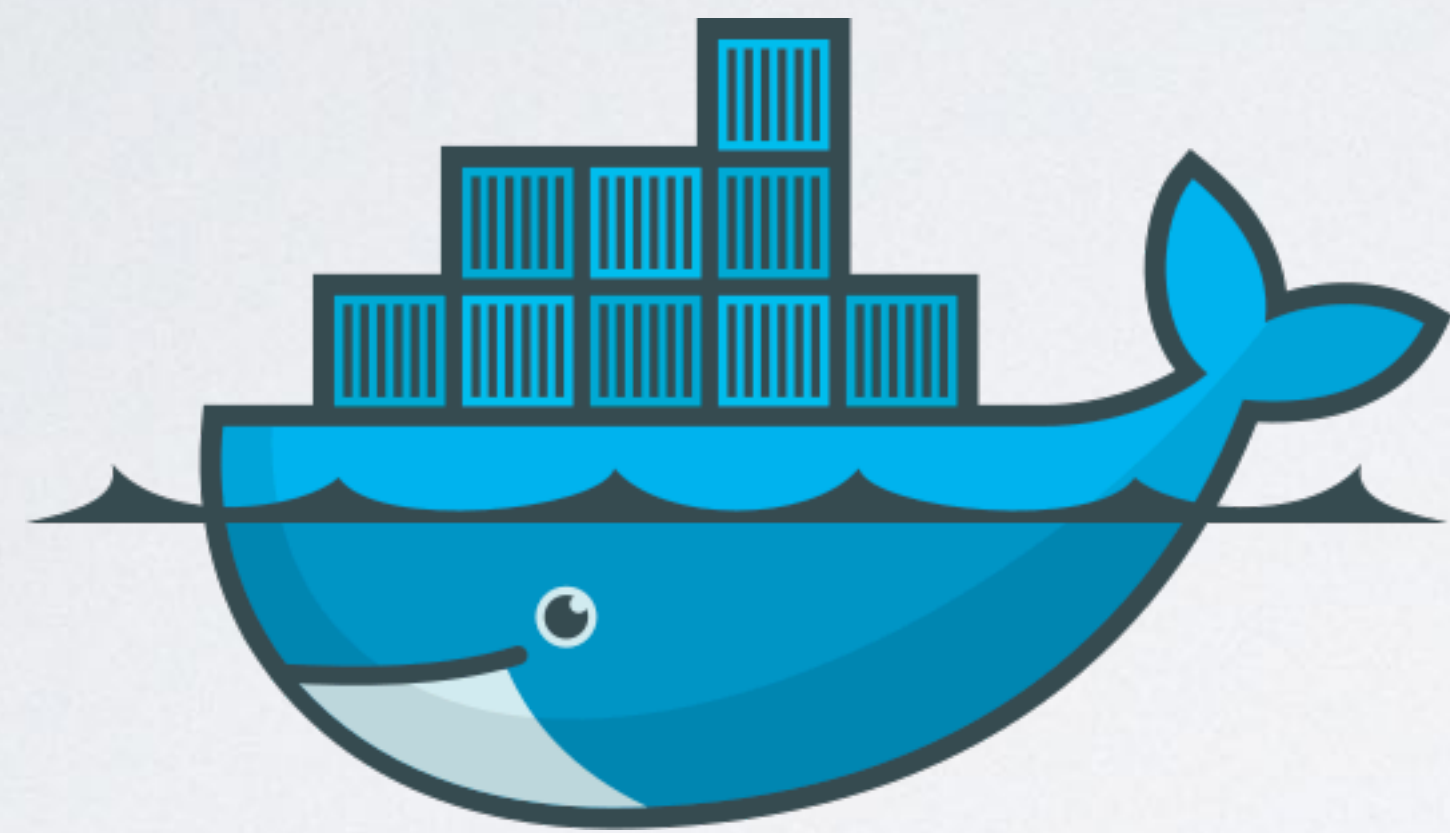


20分钟写一个docker App



容器编排

DOCKER究竟是个啥



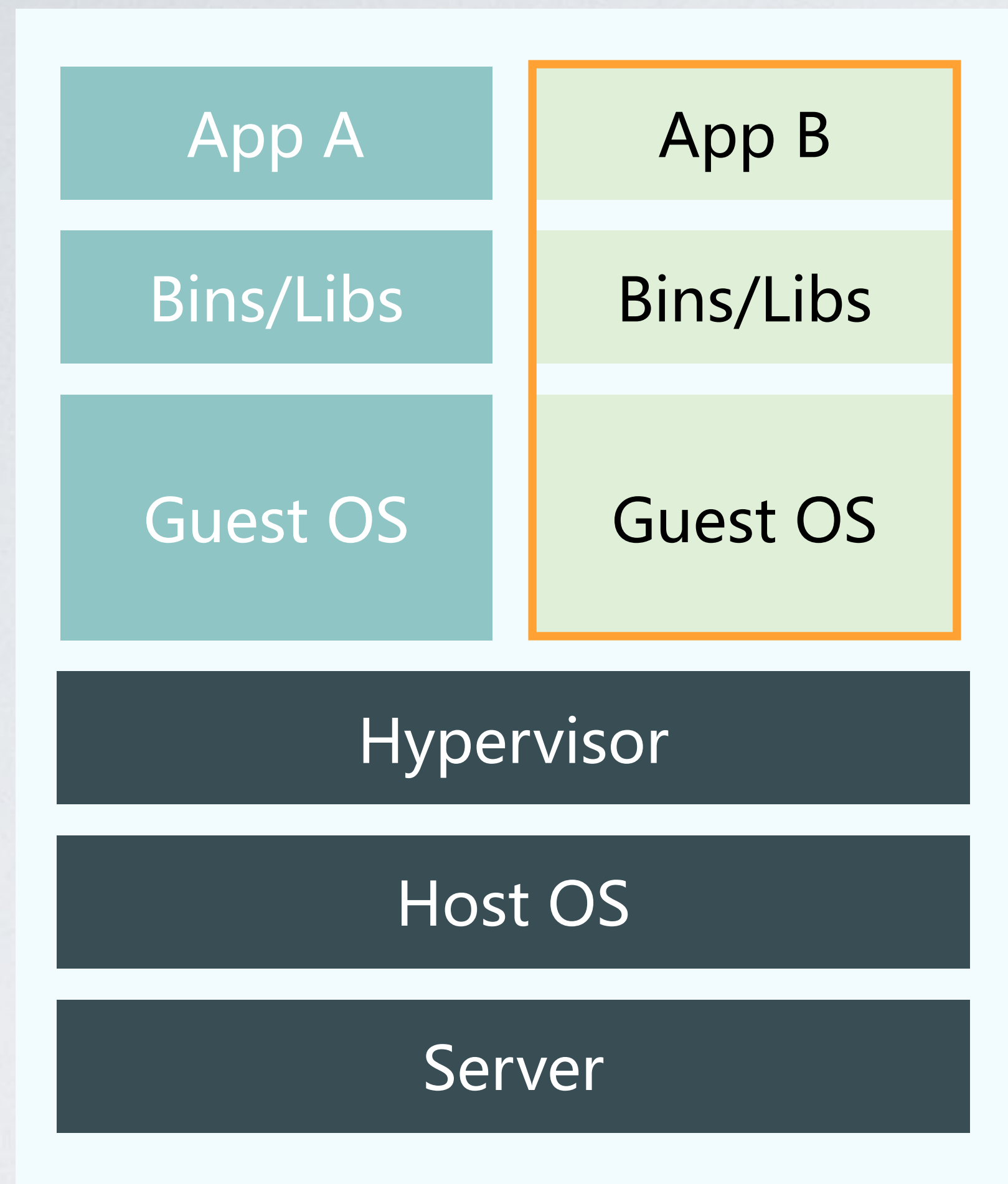
docker



DOCKER是一种虚拟化技术

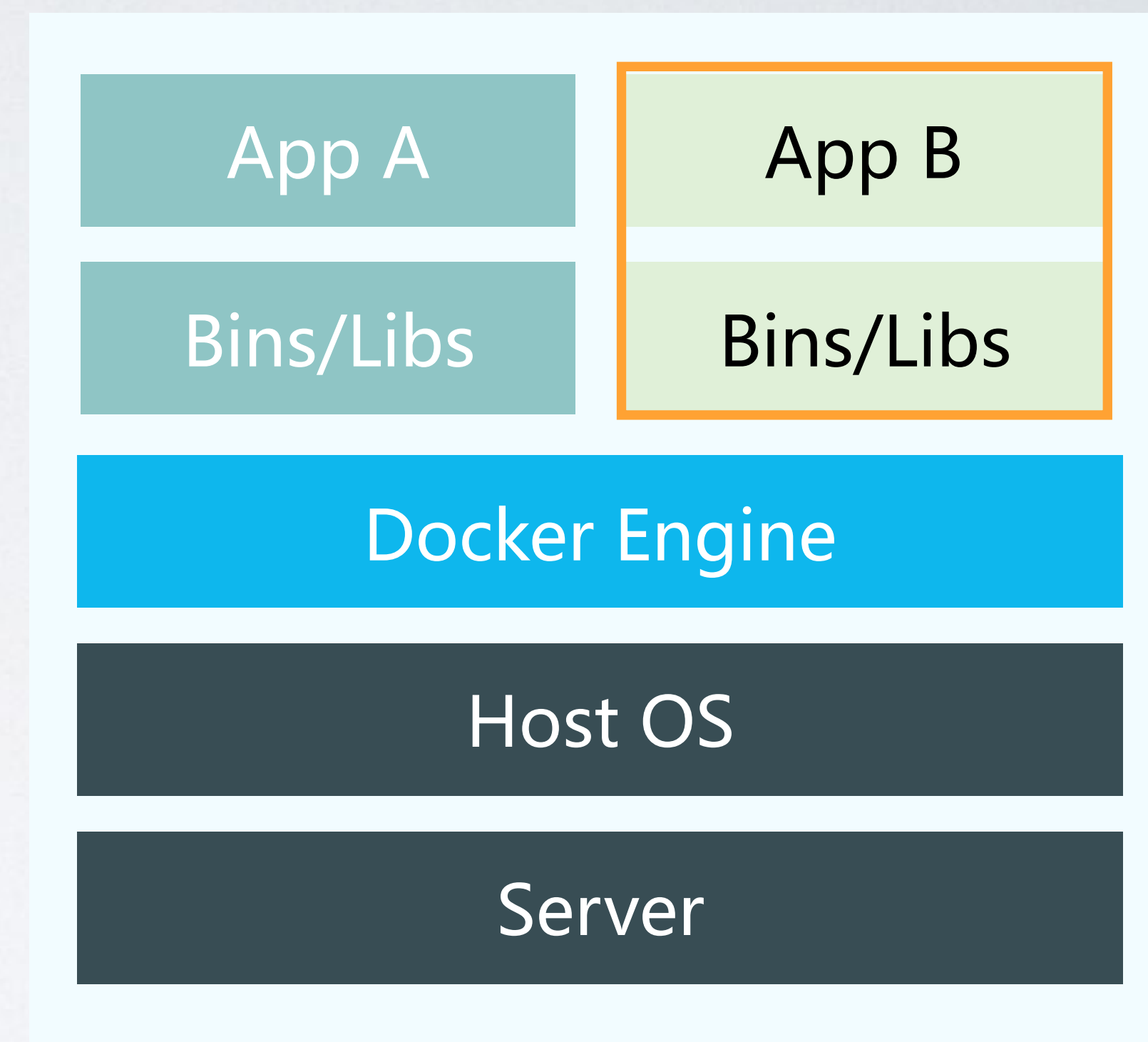
DOCKER是一种虚拟化技术

DOCKER和虚拟机有什么区别



虚拟机

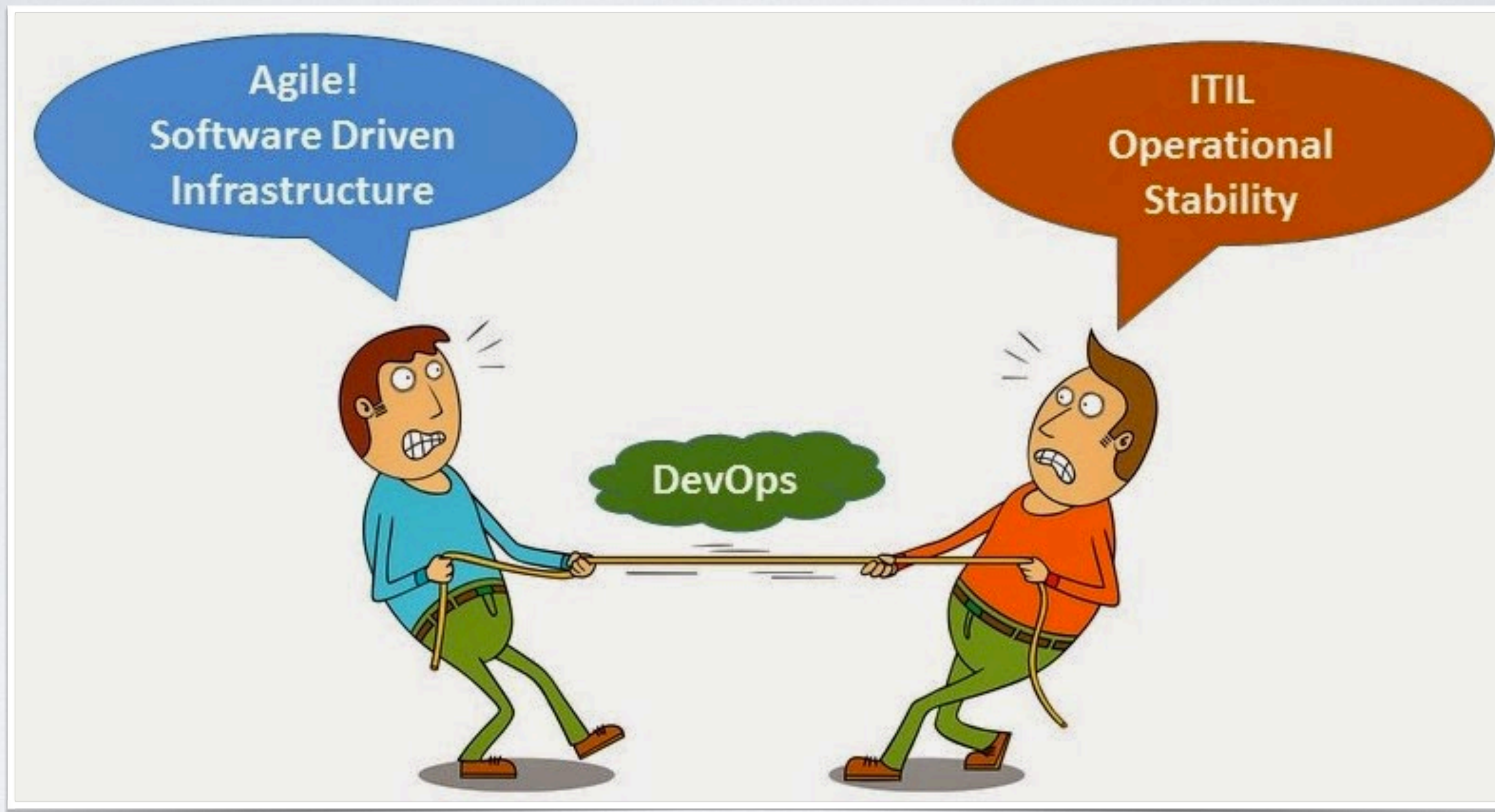
VS



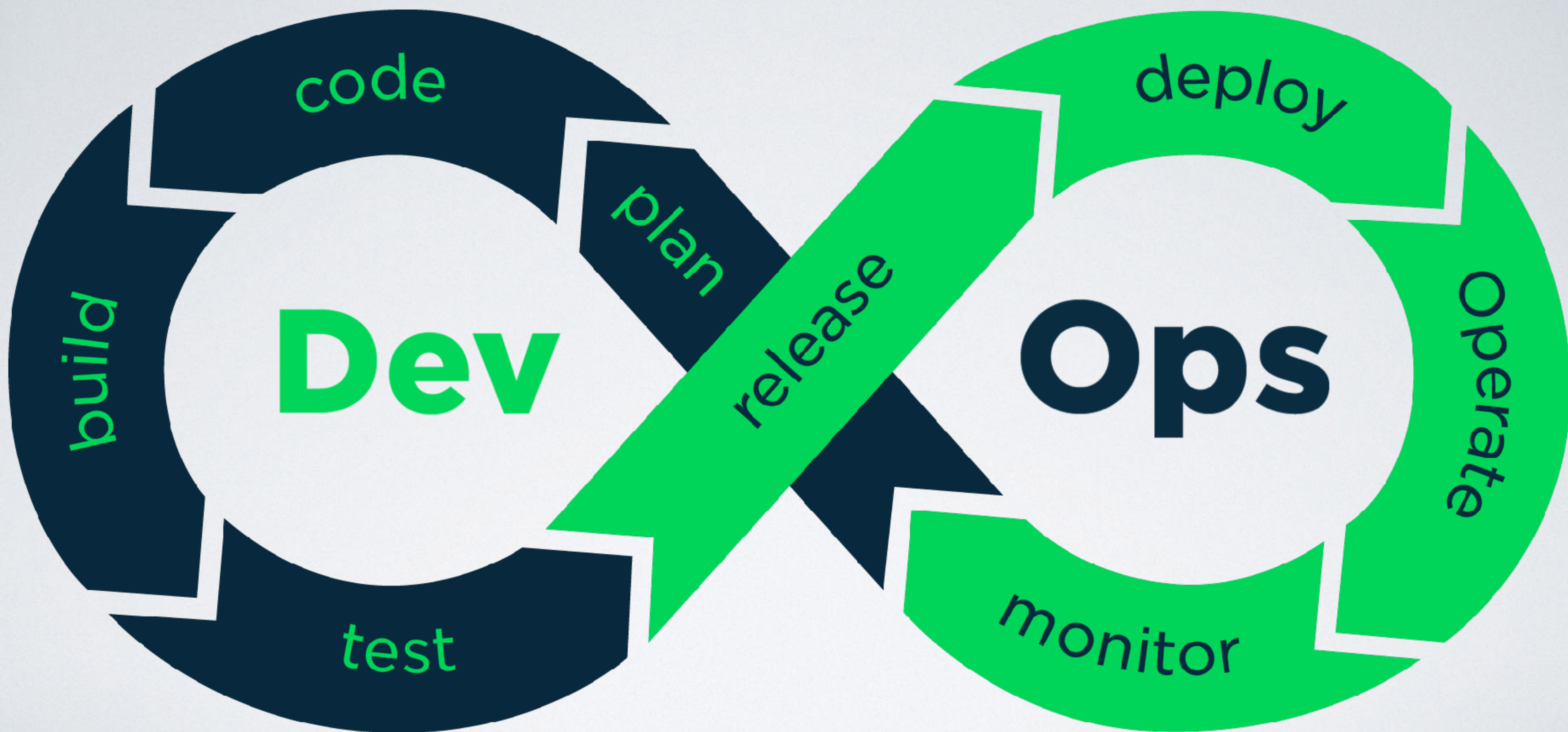
Docker

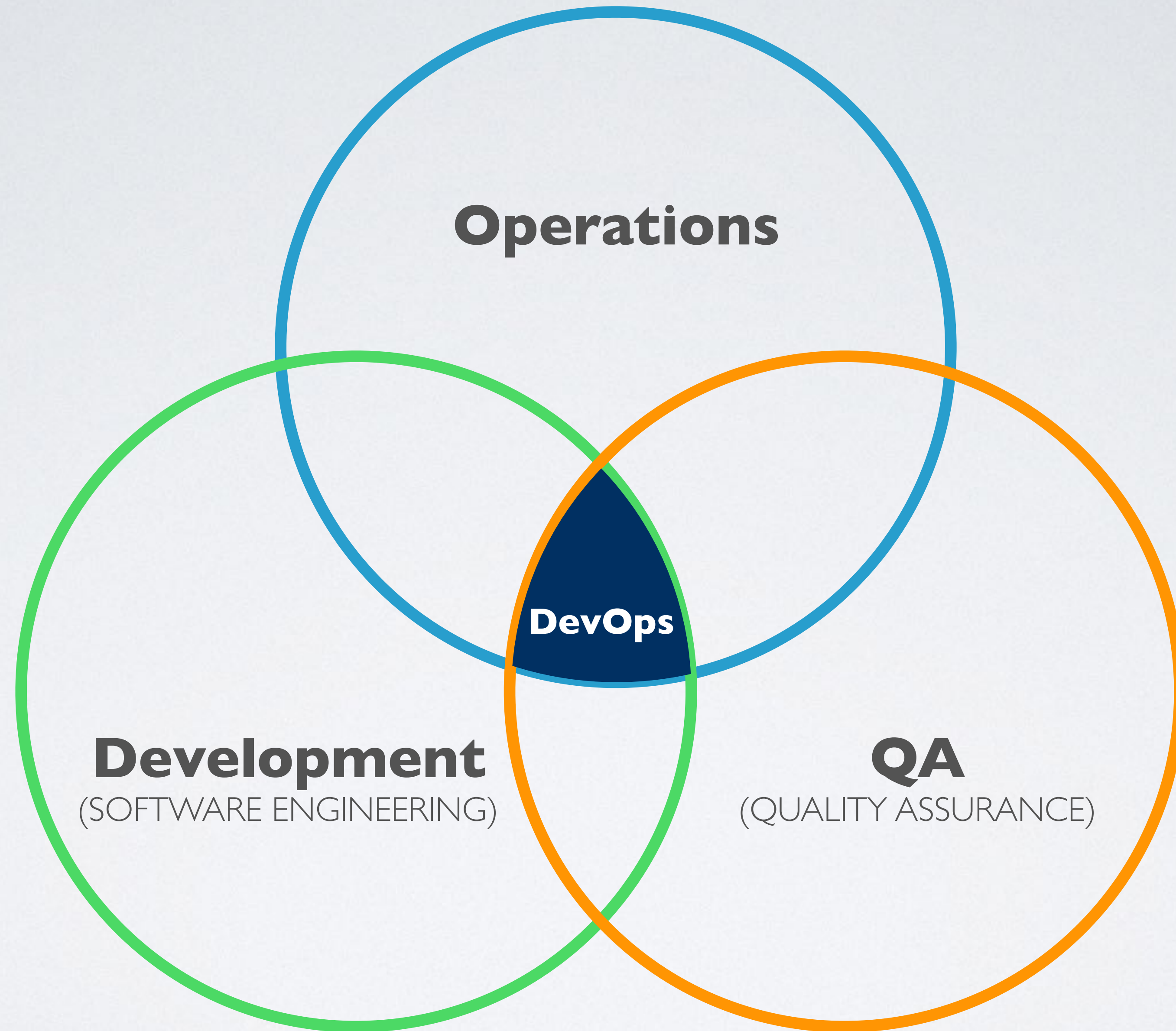
特性	Docker	虚拟机
启动	秒级	分钟级
硬盘使用	一般为MB	一般为GB
性能	99%	90%
系统支持量	单机支持上千个	单机一般几十个

为什么要学习DOCKER

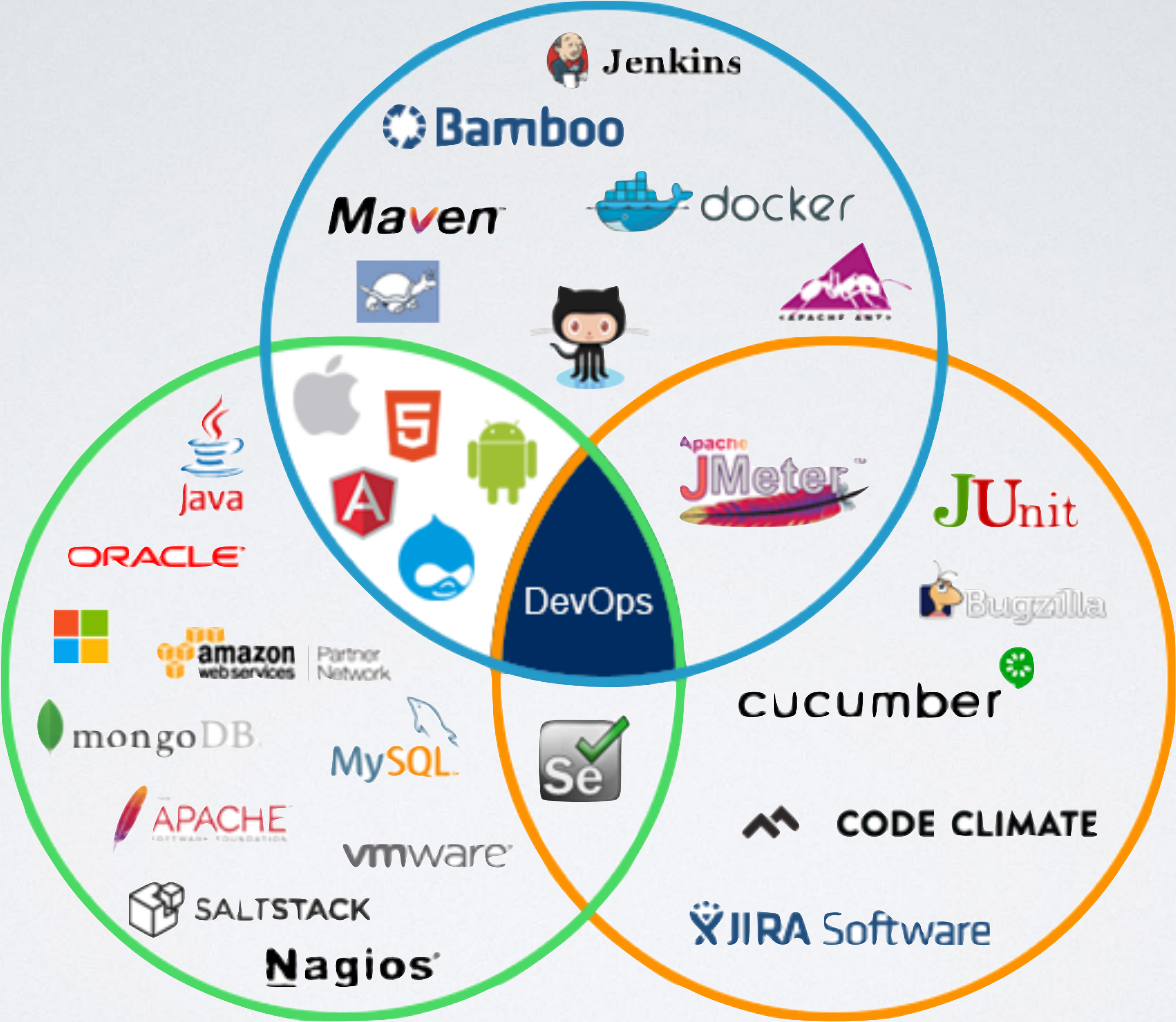


Dev Ops





Operations



Development

QA



Communication



Collaboration



Automation



Monitoring

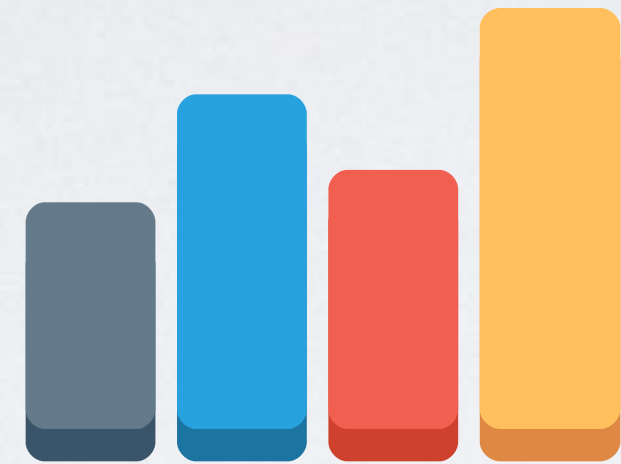


DevOps

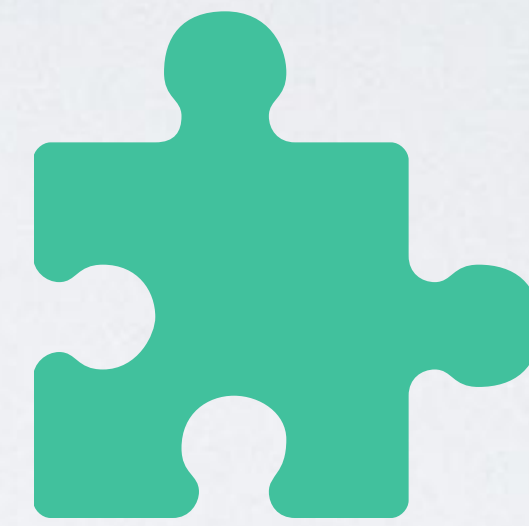
DOCKER能够帮助我们



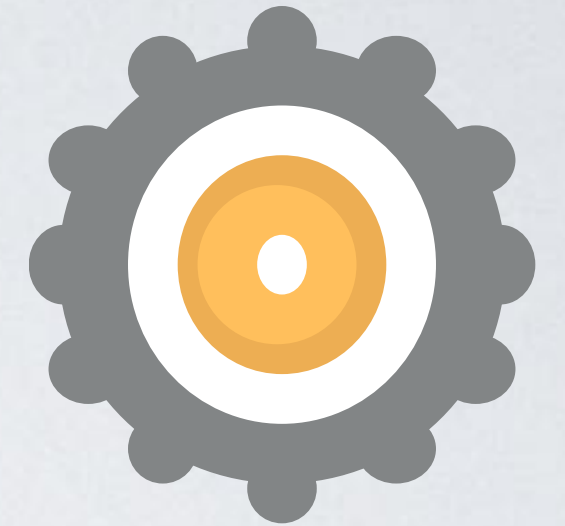
更快速的交付
和部署



更高效的虚拟化



更轻松的迁移
和扩展



更简单的管理

DOCKER拥有强大的生态圈



APCERA®



EMC²



MESOSPHERE



Microsoft

Pivotal®

vmware®

Tencent 腾讯



阿里云
aliyun.com



openstack™
CLOUD SOFTWARE



RANCHER®

我们最终目的是



DOCKER常用概念

DOCKER常用概念

Image

Container

Daemon

Registry/
Hub

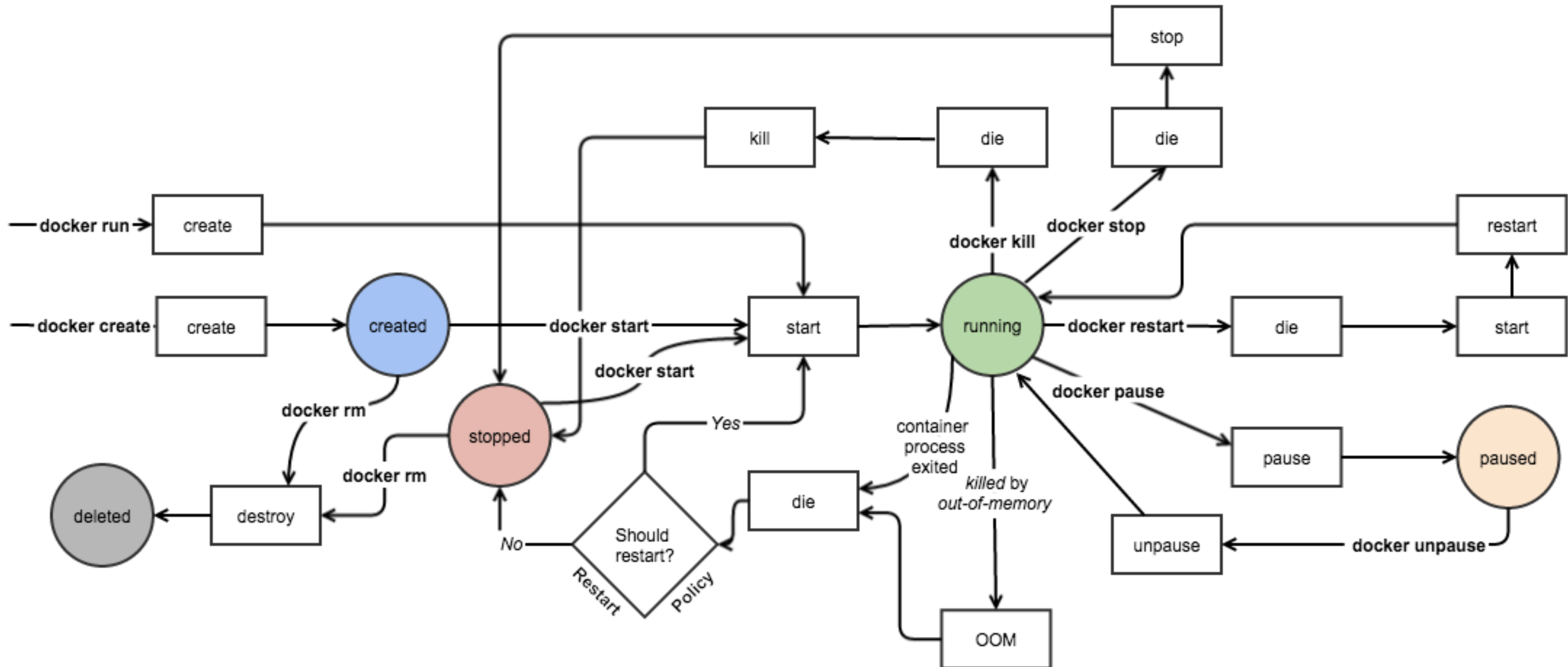
DOCKER IMAGE

- Docker Image是一个极度精简版的Linux程序运行环境，比如vi这种基本的工具没有，官网的Java镜像包括的东西更少，除非是镜像叠加方式的，比如Centos+Java7
- Docker Image是需要定制化Build的一个“安装包”，包括基础镜像+应用的二进制部署包
- Docker Image内不建议有运行期需要修改的配置文件
- Dockerfile用来创建一个自定义的image,包含了用户指定的软件依赖等
- 当前目录下包含Dockerfile,使用命令build来创建新的image
- Docker Image的最佳实践之一是尽量重用和使用网上公开的基础镜像

DOCKER CONTAINER

- Docker Container是Image的实例，共享内核
- Docker Container里可以运行不同Os的Image，比如Ubuntu的或者CentOS
- Docker Container不建议内部开启一个SSHD服务，1.3版本后新增了docker exec命令进入容器排查问题。
- Docker Container没有IP地址，通常不会有服务端口暴露，是一个封闭的“盒子/沙箱”

LIFECYCLE OF DOCKER CONTAINER



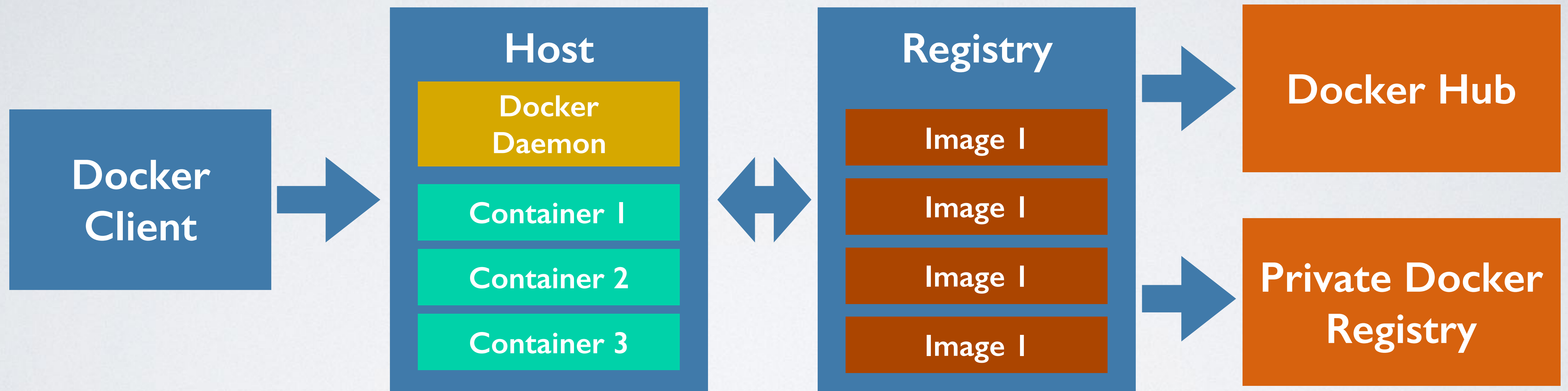
DOCKER DAEMON

- Docker Daemon是创建和运行Container的Linux守护进程，也是Docker 最主要的核心组件
- Docker Daemon 可以理解为Docker Container的Container
- Docker Daemon可以绑定本地端口并提供Rest API服务，用来远程访问和控制

DOCKER REGISTRY/HUB

- 仓库是集中存放镜像文件的场所。有时候会把仓库和仓库注册服务器（Registry）混为一谈，并不严格区分。实际上，仓库注册服务器上往往存放着多个仓库，每个仓库中又包含了多个镜像，每个镜像有不同的标签（tag）。
- Docker 仓库的概念跟 Git 类似，注册服务器可以理解为 GitHub 这样的托管服务。
- 仓库分为公开仓库（Public）和私有仓库（Private）两种形式。
- 最大的公开仓库是 Docker Hub，存放了数量庞大的镜像供用户下载。国内的公开仓库包括 DaoCloud 等，可以提供大陆用户更稳定快速的访问。当然，我们也可以在本机网络内创建一个私有仓库。
- 当用户创建了自己的镜像之后就可以使用 push 命令将它上传到公有或者私有仓库，这样下次在另外一台机器上使用这个镜像时候，只需要从仓库上 pull 下来就可以了。

DOCKER核心组件关系

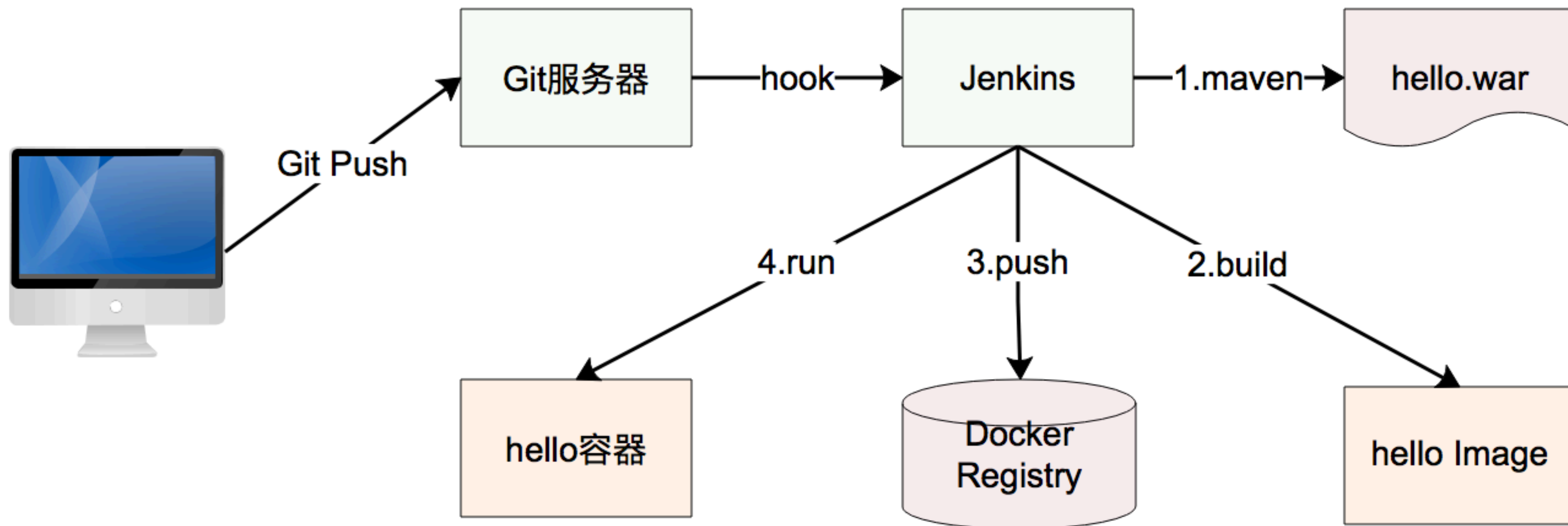


DOCKER学习资源

- 官网: <https://www.docker.com/>
- Docker Hub: <https://hub.docker.com/>
- Docker中文社区: <http://www.docker.org.cn/index.html>
- DockerOne问答社区: <http://dockone.io/>
- 我的个人站点
 - GitHub: <https://github.com/bluecoffee1013/docker-library>
 - 码云: <https://git.oschina.net/bluecoffee1013/docker-library>

现在， 让我们撸起袖子写代码吧

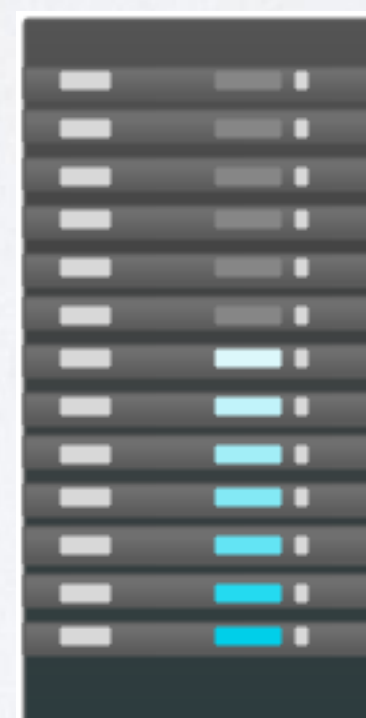
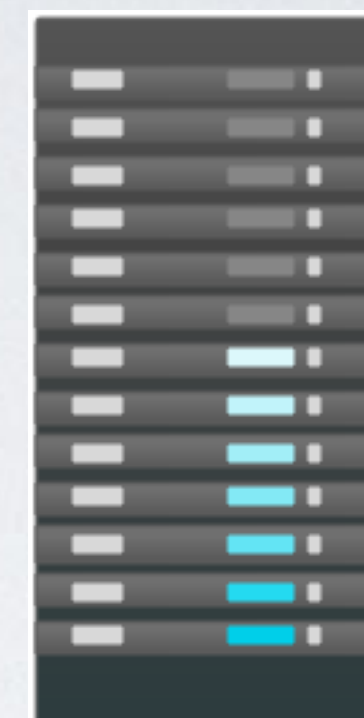
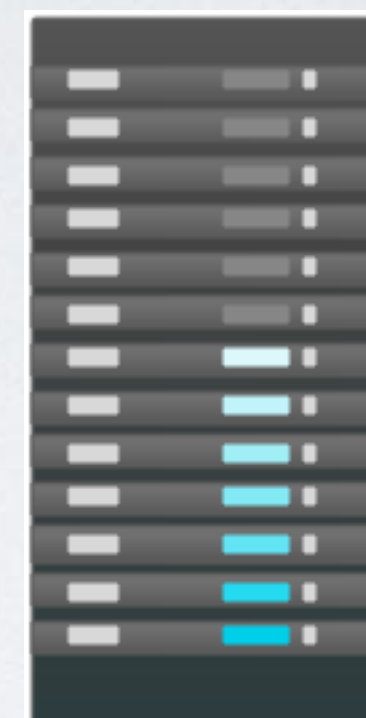
结合DOCKER的CI/CD流程



终于，到了让人  的时候了...

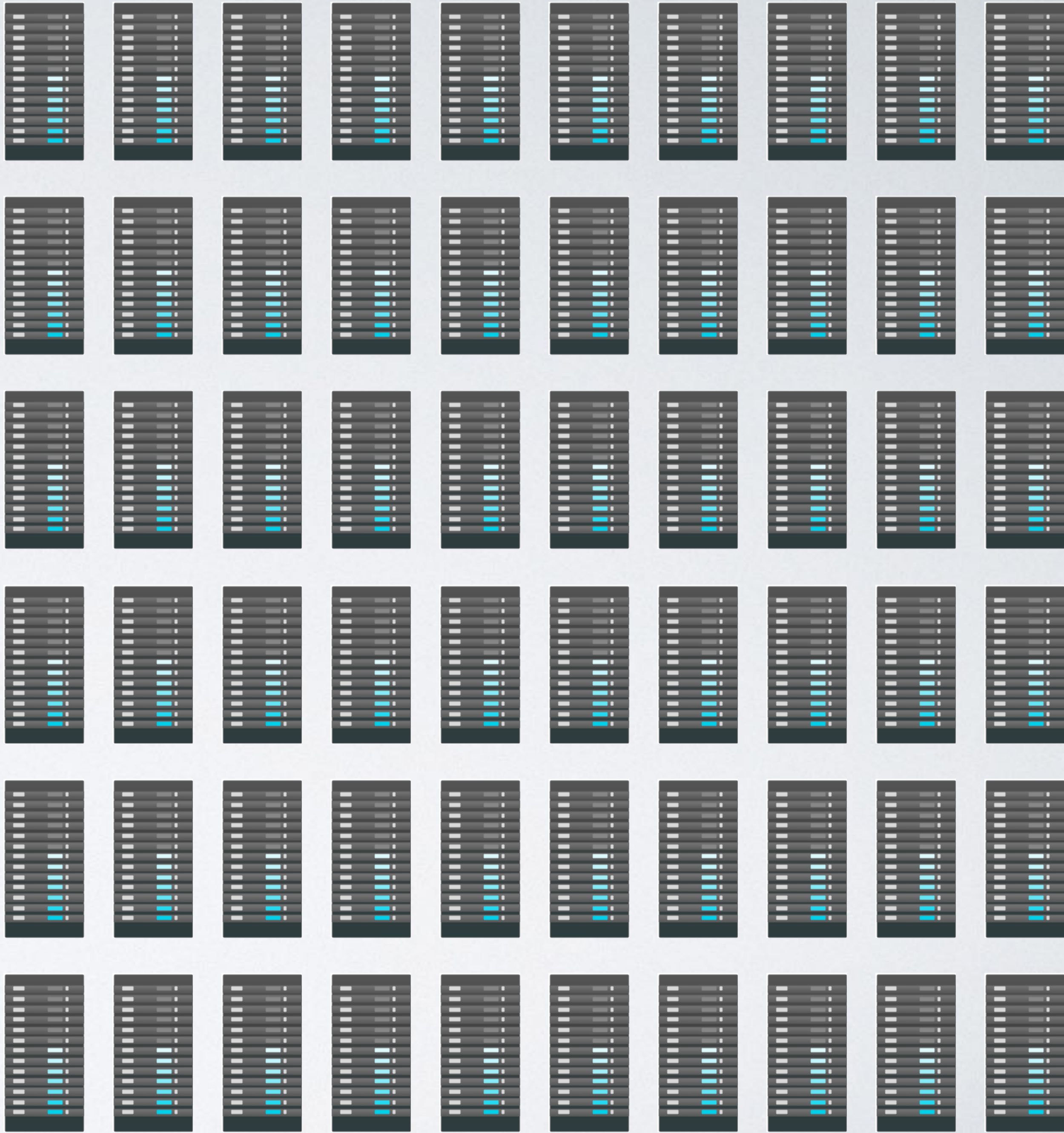


老子今天不用加班了！！！！





这么多服务，可怎么管理啊？？？

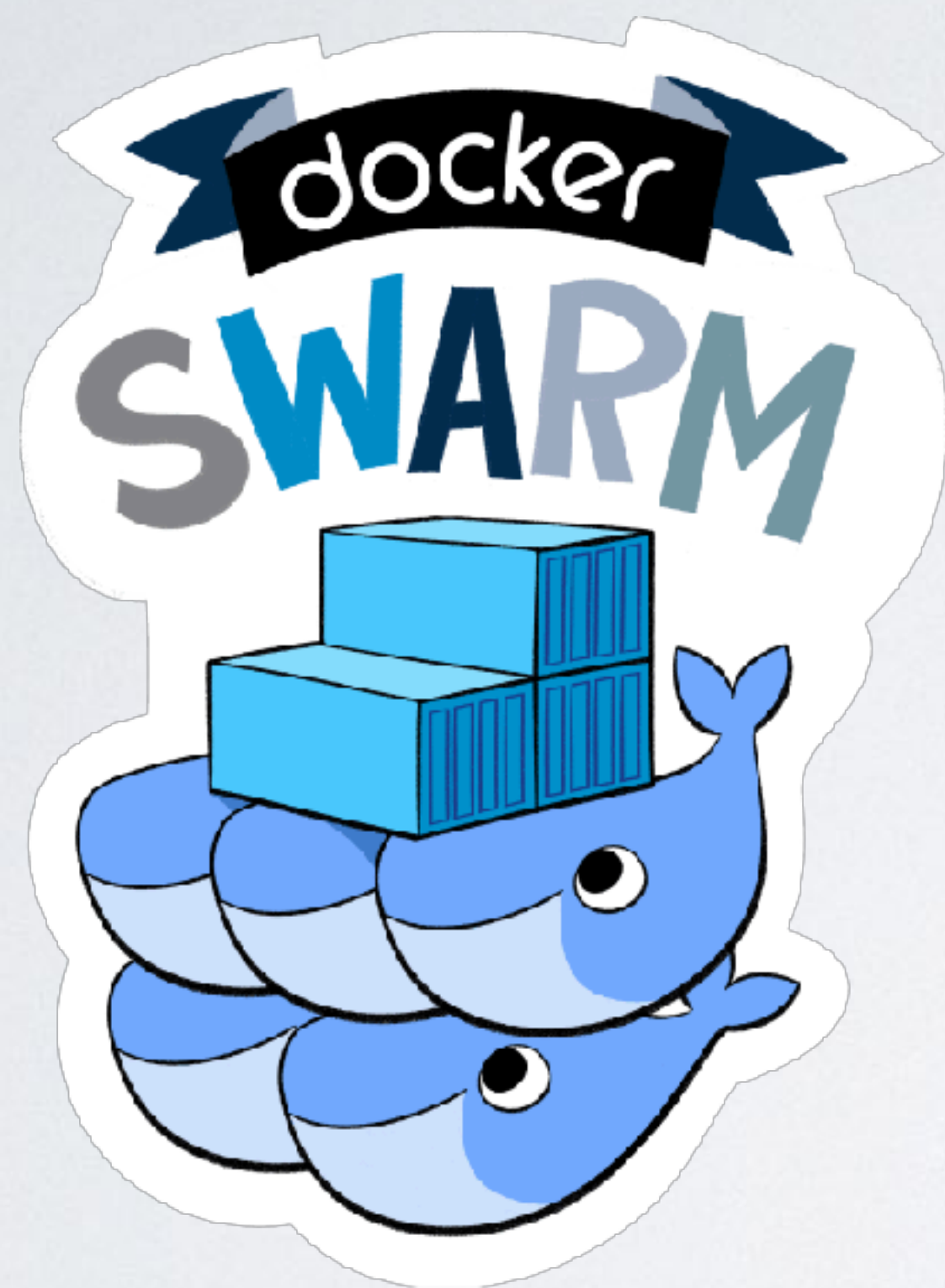


我们需要一个容器编排工具



约束
优先级
负载均衡
健康检查
服务发现

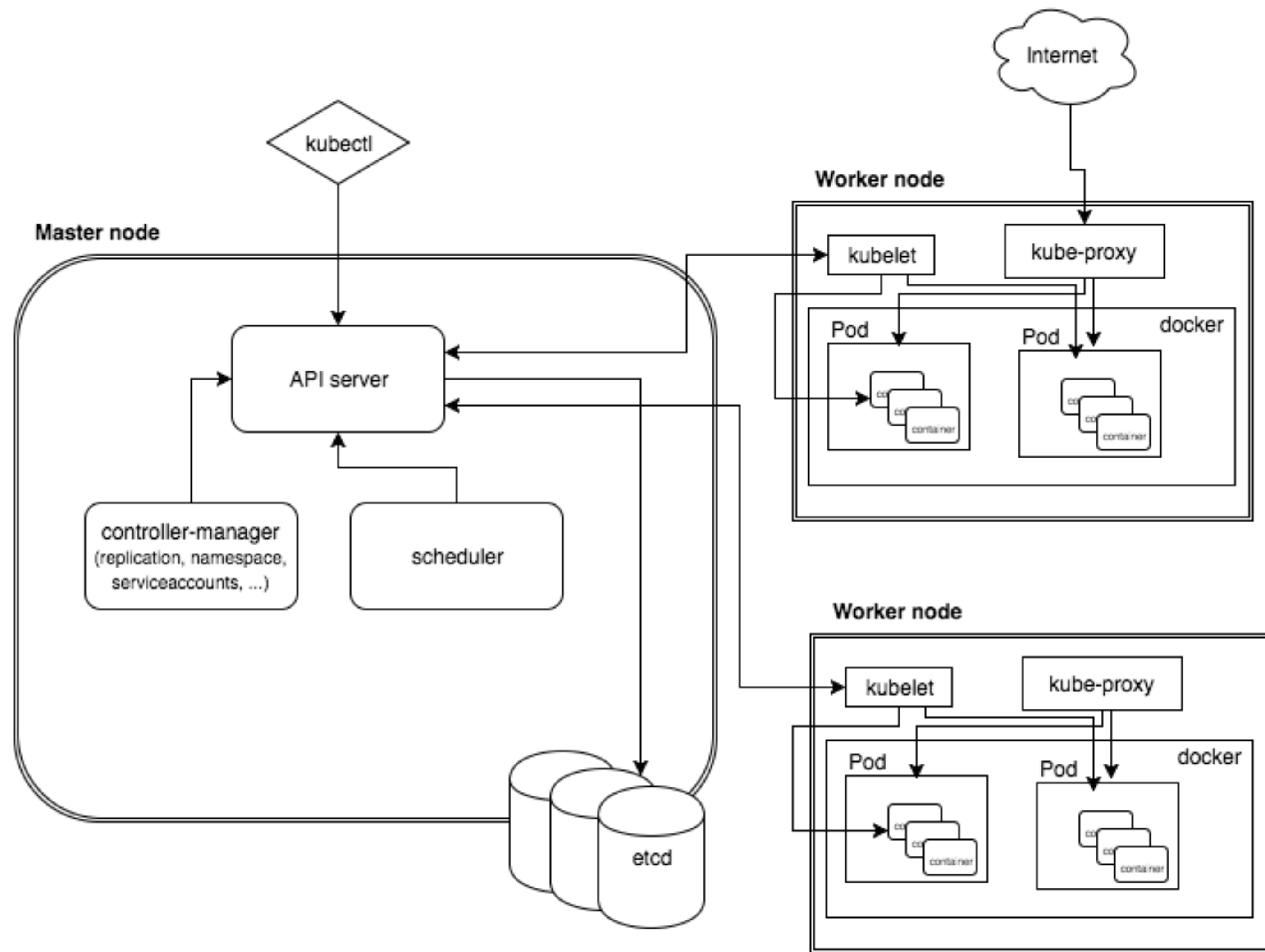
容器编排三驾马车



MESOS



kubernetes



- etcd保存了整个集群的状态；
- apiserver提供了资源操作的唯一入口，并提供认证、授权、访问控制、API注册和发现等机制；
- controller manager负责维护集群的状态，比如故障检测、自动扩展、滚动更新等；
- scheduler负责资源的调度，按照预定的调度策略将Pod调度到相应的机器上；
- kubelet负责维护容器的生命周期，同时也负责Volume（CVI）和网络（CNI）的管理；
- Container runtime负责镜像管理以及Pod和容器的真正运行（CRI）；
- kube-proxy负责为Service提供cluster内部的服务发现和负载均衡；

THANK YOU