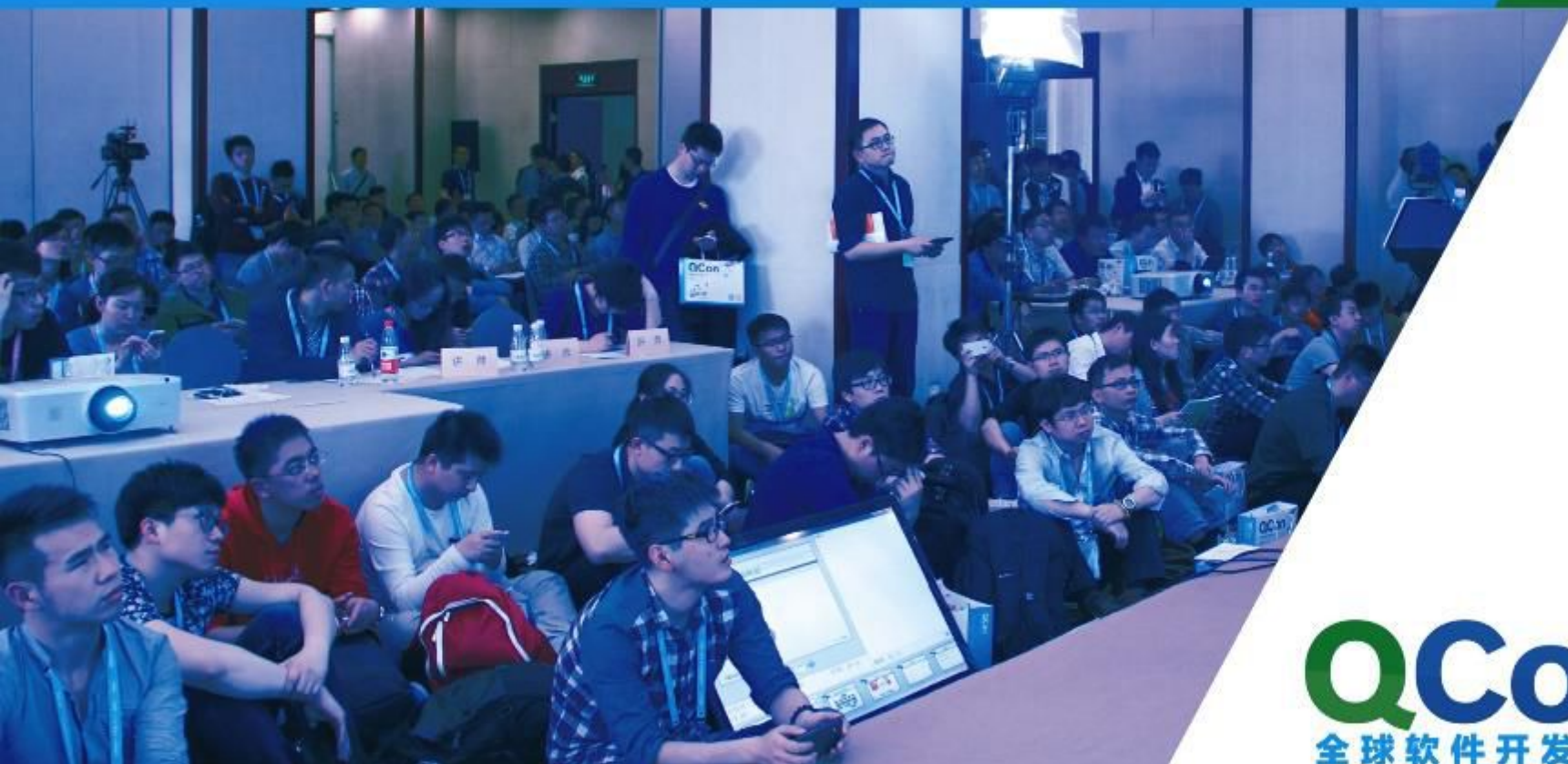


QCon全球软件开发大会

International Software Development Conference



QCon
全球软件开发大会

Geekbang >

极客邦科技

全球领先的技术人学习和交流平台

扫我，码上开启新世界



Geekbang >

InfoQ | EGO NETWORKS | StuQ

InfoQ
InfoQ

专注中高端技术
人员的社区媒体

EGO EXTRA GEEKS' ORGANIZATION
NETWORKS

高端技术人员
学习型社交网络

StuQ
StuQ

实践驱动的IT职业
学习和服务平台

InfoQ^{League}

促进软件开发领域知识与创新的传播

ArchSummit
全球架构师峰会

实践第一 案例为主

时间：2015年12月18-19日 / 地点：北京·国际会议中心

欢迎您参加ArchSummit北京2015，技术因你而不同



ArchSummit北京二维码

QCon
全球软件开发大会

【北京站】

2016年04月21日-23日



关注InfoQ官方信息
及时获取QCon演讲视频信息

汽车之家*Docker*应用实践

王盛

我们对Docker的认识

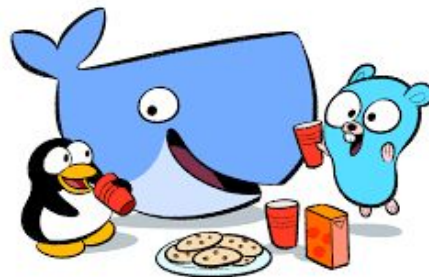
Docker是利用现有成熟的技术基础上进行的更好封装出来的产品
轻量，占用很少cpu和内存

Docker可以让开发和测试对于环境部署和使用更加简单

Docker实现了大部分程序都可以在一个安全隔离的环境中运行

可以很方便的运行中间件和应用

开源可以定制开发



我们对Docker的认识-核心技术栈

命名空间*Namespace*

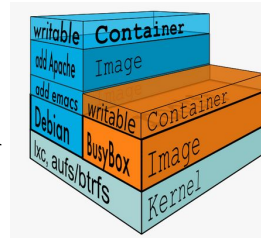
-通过*Namespace*将容器的网络、进程、用户等等隔离

控制组*Control Groups*

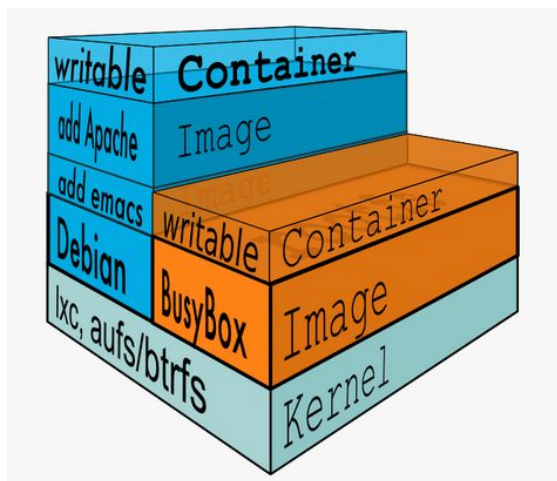
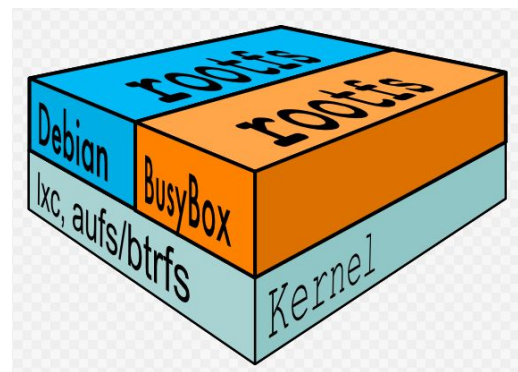
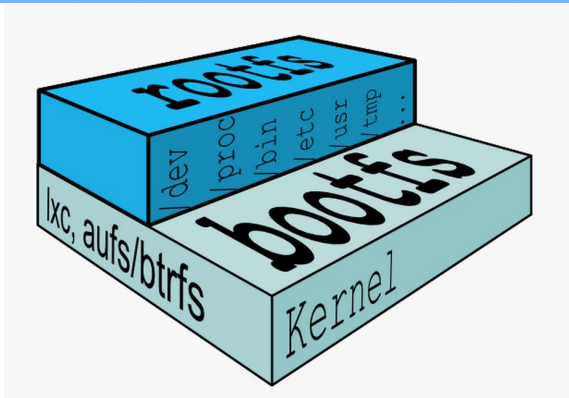
-对共享资源进行隔离、限制，避免多个容器同时运行时对资源进行抢占

联合文件系统 (*DeviceMapper/Aufs/Vfs*)

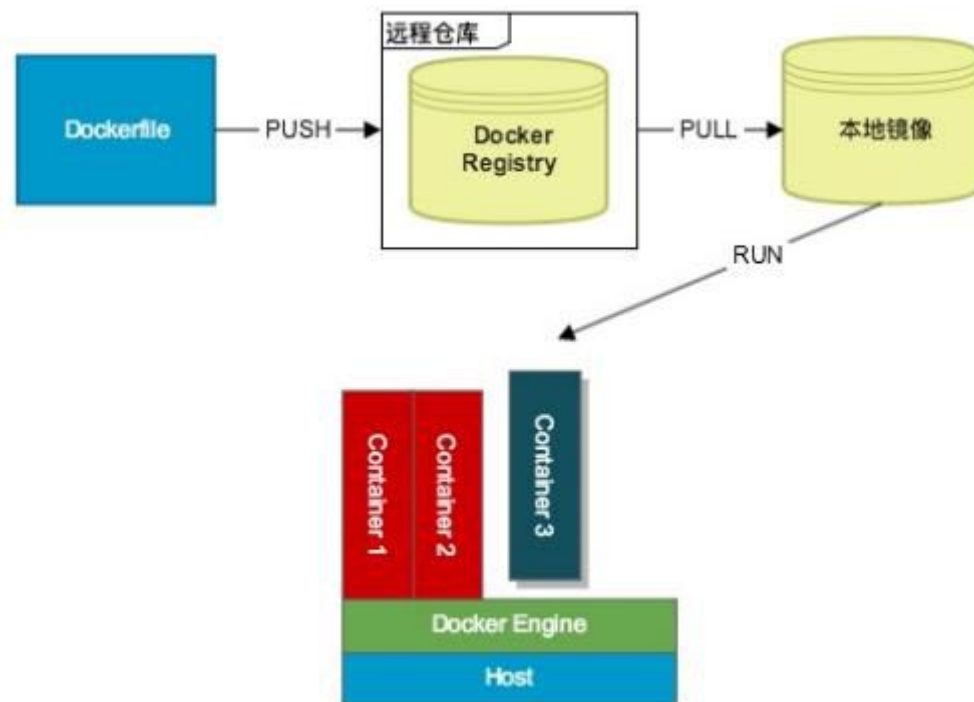
-这是一种分层、轻量级的文件系统，通过分层来进行制作镜像



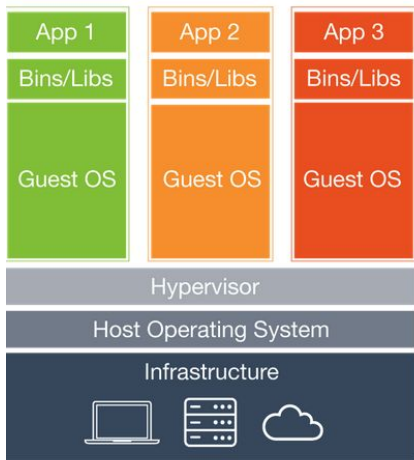
我们对Docker的认识-核心技术栈



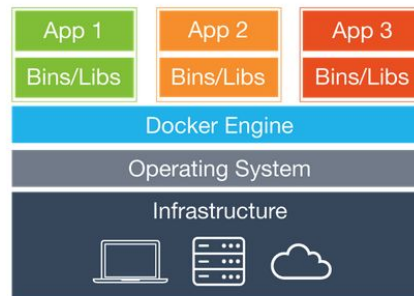
Docker生命周期



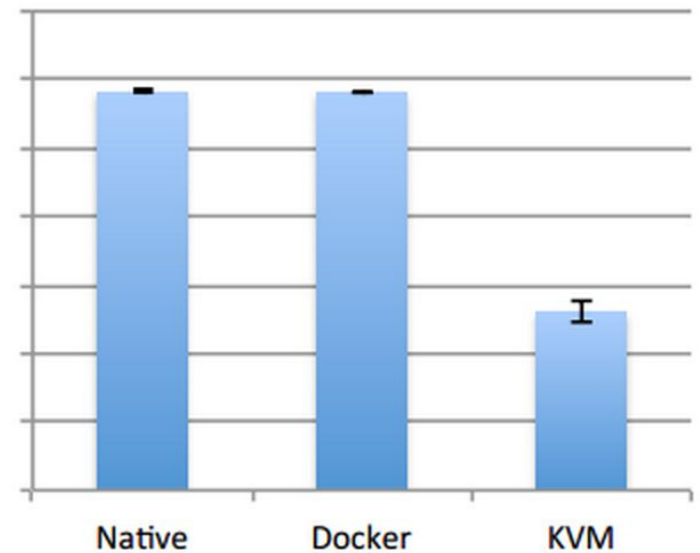
Container vs VM



Virtual Machines



Containers



Container vs VM

- 节省虚拟Guest OS需要的大量空间
- 硬盘空间以GB为单位， Guest OS越多越体现Docker优势
- 性能接近原生系统
- 轻量级,快速创建和启动

Docker仓库

Official Repositories



特点

能够快速让你的应用测试、上线

测试、生产环境相对一致，不会出现调环境而耽误大量时间

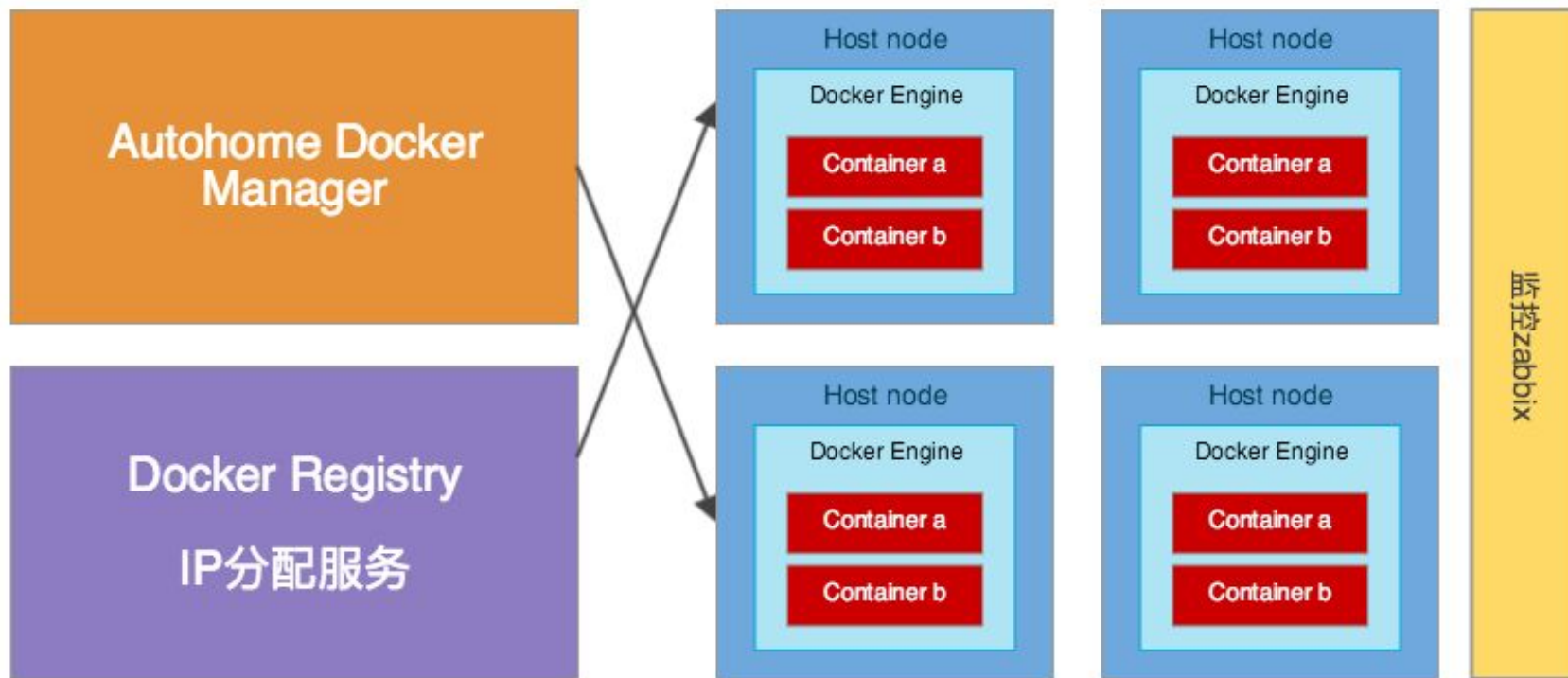
横向更加简单快速，利用有效的管理工具可以更高密度的扩展，

而省去大量的人工成本

可以更好利用现有机器资源，提高利用率

可以智能实现横向扩展

汽车之家应用Docker现状



Docker应用-现状

*Linux*发行版 *Centos 6.4* *Kernel 3.10*和*4.0*

*Docker*版本 *1.6.2*

快速部署、环境统一、轻量、更快

开发自研群管理和容器调度

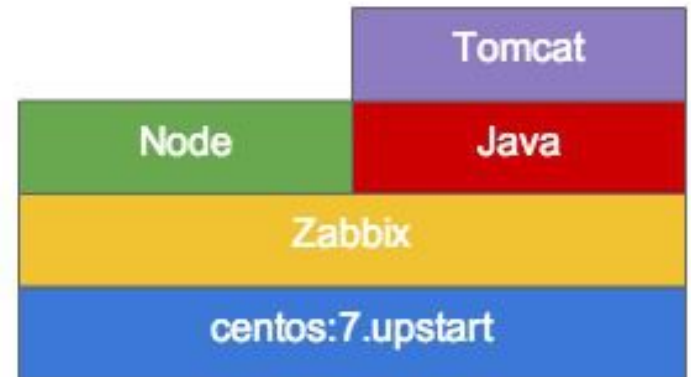
目标是让*Docker*更像物理机器

能够利用现有仓库和编排系统快速搭建开发环境

为了适应当前的网络架构更改了*Docker*网络部分

Docker应用-构建

- 有单独机器专门做构建
- 构建版本镜像版本有分层
- 基于*Centos:7+systemd+zabbix*
- 对构建后的镜像有统一的管理



Docker应用-Docker Hub

- 版本的是 *Docker Registry 2.0*
- *Registry* 本身不能够高可用，*Nginx* 负载多个 *Registry*
- 使用网络存储共享镜像
- 配置 *Mirroring*，获取官网镜像

```
mirroring:  
source: "https://registry-1.docker.io"  
source_index: "https://index.docker.io"  
tags_cache_ttl: 172800
```

Docker应用-Daemon

- 更改镜像位置（`--graph="/data/docker/lib"`）
- 删除docker0网桥，桥连到自定义网桥

```
ip link set dev docker0 down  
brctl delbr docker0
```

```
brctl addbr br100  
ip addr add 192.168.4.233/19 dev br100  
ip link set dev br100 up
```

- 增私服地址（`--insecure-registry="r1.d.xxxx.com"`）

Docker应用-网络问题

对比集中网络模式

container模式：只在单机多个容器之间

nat模式：配置映射繁琐麻烦，网络性能低

host模式：复用宿主机端口，问题是无法端口重用，容易冲突

Docker应用-网络问题

Docker目前提供的网络模式不适合业务环境

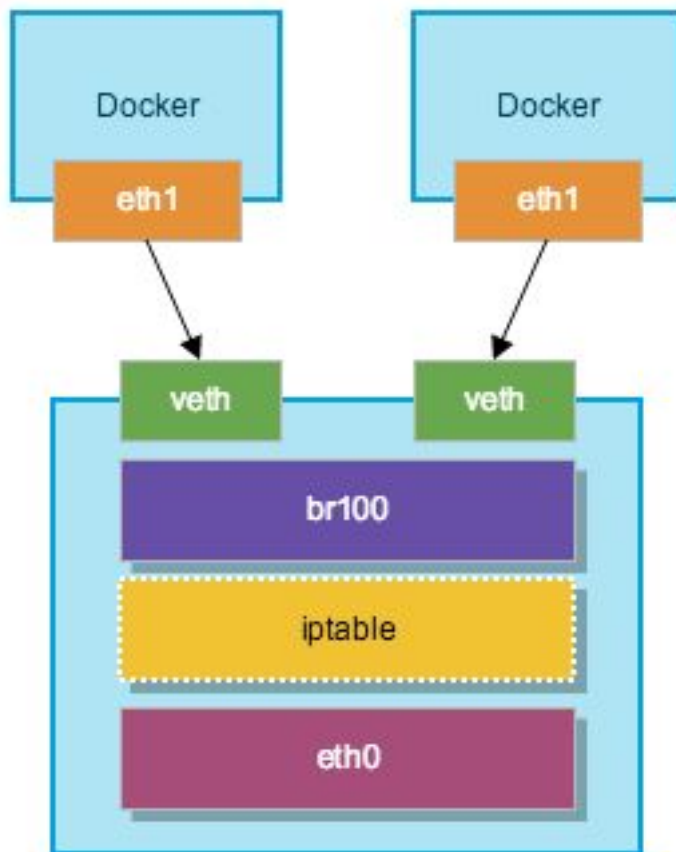
--iptables =false

修改Docker内核改成静态IP模式

```
ip, err = ipallocator.RequestIPWithNF(job)
if err != nil {
    log.Errorf("Allocator: RequestIP v4: %v", err)
    return job.Error(err)
}
```

IP是通过容器名字为标识从IP Pool获取

Docker应用-网络问题



Docker应用-双11方案

往年双11遇到的问题

测试、开发、生产环境部署问题

遇到高峰值流量无法支撑

降级方案不理想，无法利用现有资源去平衡补充

可能因为某一个业务压力有问题，导成整个流程无法
继续

Docker集群管理和容器调度

可以像使用一个*Docker engine*一样管理一个*docker engine*集群

有固定*IP*池分配服务，全部采用静态*IP*的方式

按照业务线来进行划分和管理

可以指定业务到*IP*或者不同机器随机分配容器

集成*Zabbix*进行监控、报警

将来要做的

Docker 跨服务器容灾迁移

集成新版Windows *docker*容器

Docker daemon 单例问题

Docker daemon 挂掉容器也挂掉的问题

将来要做的

Q & A

THANKS

Brought by **InfoQ**

International Software Development Conference