



51CTO 传媒

2013 云计算架构师峰会

Cloud Computing Architects Summit





轻量虚拟化技术

-- docker实战分享

- 百度BAE
- 陈轶飞
- 2013-10.25

纲要

- 一、轻量虚拟化技术
 - 是什么？
 - 涉及到哪些技术和知识？
 - 应用领域
- 二、BAE与docker
 - docker是什么？
 - BAE为什么选择docker？
 - BAE是如何使用docker的？
- 三、docker的发展和展望

轻量虚拟化技术

--是什么？

• 以Linux进程的视角来看

- 进程运行的时候，周边的环境和资源有哪些？
 - Linux内核
 - 文件系统
 - 网络系统
 - PID、UID、IPC等资源
 - 内存、CPU、磁盘等资源
 - 其它
- 每个进程的看到的周边环境都一样
- 所有进程共享这些资源

轻量虚拟化技术

--是什么？

- 技术发展过程中，产生了这样的需求
 - **资源隔离：** 不同进程，希望有自己独立的周边环境
 - **资源限制：** 限制某些进程所能使用的资源
- 能对**一组进程**进行上述隔离和限制

轻量虚拟化技术

--是什么？

- 需求总结为：
 - 为一组进程
 - 分配独立的运行环境
 - 文件系统
 - 网络系统
 - PID、UID、UTS、mount、IPC空间
 - 并能对它们能使用的资源从总体上进行限制
 - 内存
 - CPU
 - 网络流量
 - 磁盘空间
 - 磁盘读写频率
 - 最好还能防止进程组之间互相干扰

轻量虚拟化技术

--是什么？

- 轻量虚拟化，就是用来实现上述需求的技术
- 满足上述限制条件的进程组，叫做“轻量虚拟机”，或者 Container
- Process Container的概念，最早是由google工程师在2006年提出来的
 - <http://lwn.net/Articles/199643/>
 - <http://lwn.net/Articles/236038/>
- wikipedia 定义
 - http://en.wikipedia.org/wiki/Operating_system-level_virtualization



轻量虚拟化技术 --是什么？

- 感性的认识

```
root@c5cec4b035ec:~# ps axf
  PID TTY          STAT       TIME COMMAND
    1 ?            S           0:00 /usr/sbin/sshd -D
    7 ?            Ss          0:00 sshd: root@pts/0
   19 pts/0        Ss          0:00  \_ -bash
  115 pts/0        R+          0:00      \_ ps axf
  110 ?            Ss          0:00 /usr/bin/redis-server /etc/redis/redis.conf
root@c5cec4b035ec:~#
```

```
11869 ?            S           0:00  \_ lxc-start -n c5cec4b035ec1e7a1816fadaec9432eaa16d165
11881 ?            S           0:00      \_ /usr/sbin/sshd -D
20626 ?            Ss          0:00      \_ /usr/bin/redis-server /etc/redis/redis.conf
```

```
11869 ?            S           0:00  \_ lxc-start -n c5cec4b035ec1e7a1816fadaec9432eaa16d165d5
11881 ?            S           0:00      | \_ /usr/sbin/sshd -D
20626 ?            Ss          0:00      | \_ /usr/bin/redis-server /etc/redis/redis.conf
17248 ?            S           0:00  \_ lxc-start -n a972e88e548618d6eea534782dc8b5d99e876eb84
17263 ?            S           0:00      | \_ /usr/sbin/sshd -D
```



轻量虚拟化技术

--相关技术和开源项目

- 底层技术:
 - namespace/cgroups
 - chroot/veth
 - union fs (AUFS)
 - netfilter/tc/quota
- container管理
 - LXC/libvirt
- 安全相关
 - grsec/apparmor/SELinux
- 高级
 - docker/warden/Imctfy/openVZ

轻量虚拟化技术

--相关技术和开源项目

cloudfoundry/heroku/dotcloud/appfog/openshift

docker

warden

imctfy

openvz

LXC

libvirt

grsec
apparmor
SELinux

namespace/cgroups/chroot/veth
netfilter/tc/quota/union fs

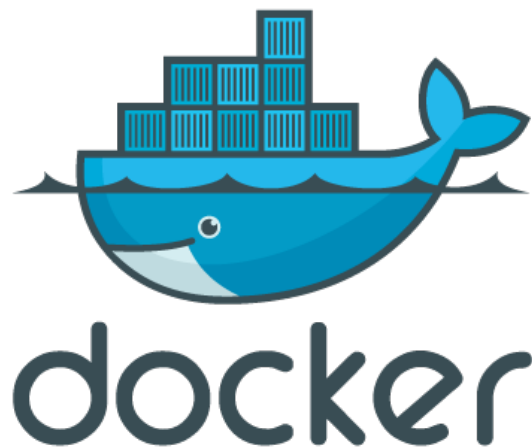
轻量虚拟化技术

--适用场景

- 在一个资源隔离、受限的环境中执行任务
- 将一个执行环境整体迁移、复制
- 运行不受信任的代码

BAE与docker

--docker是什么？



- 轻量虚拟机的完整解决方案
- dotCloud公司推出的的开源项目
- <https://www.docker.io/>
- <https://github.com/dotcloud/docker>
- 推出6个多月，Github GO语言活跃度排名第一



BAE与docker

--docker是什么？

- 基于LXC之上，但更易用
- AUFS： 创建虚拟机超快， 镜像管理超酷
- Client-Server架构
- REST API： 接口清晰
- 命令行工具： 简单应用
 - 演示

BAE与docker

--BAE是什么？

- <http://developer.baidu.com>
- 百度面向开发者的PAAS平台
- 百度轻应用的运行平台



BAE与docker

--BAE为什么选择docker?

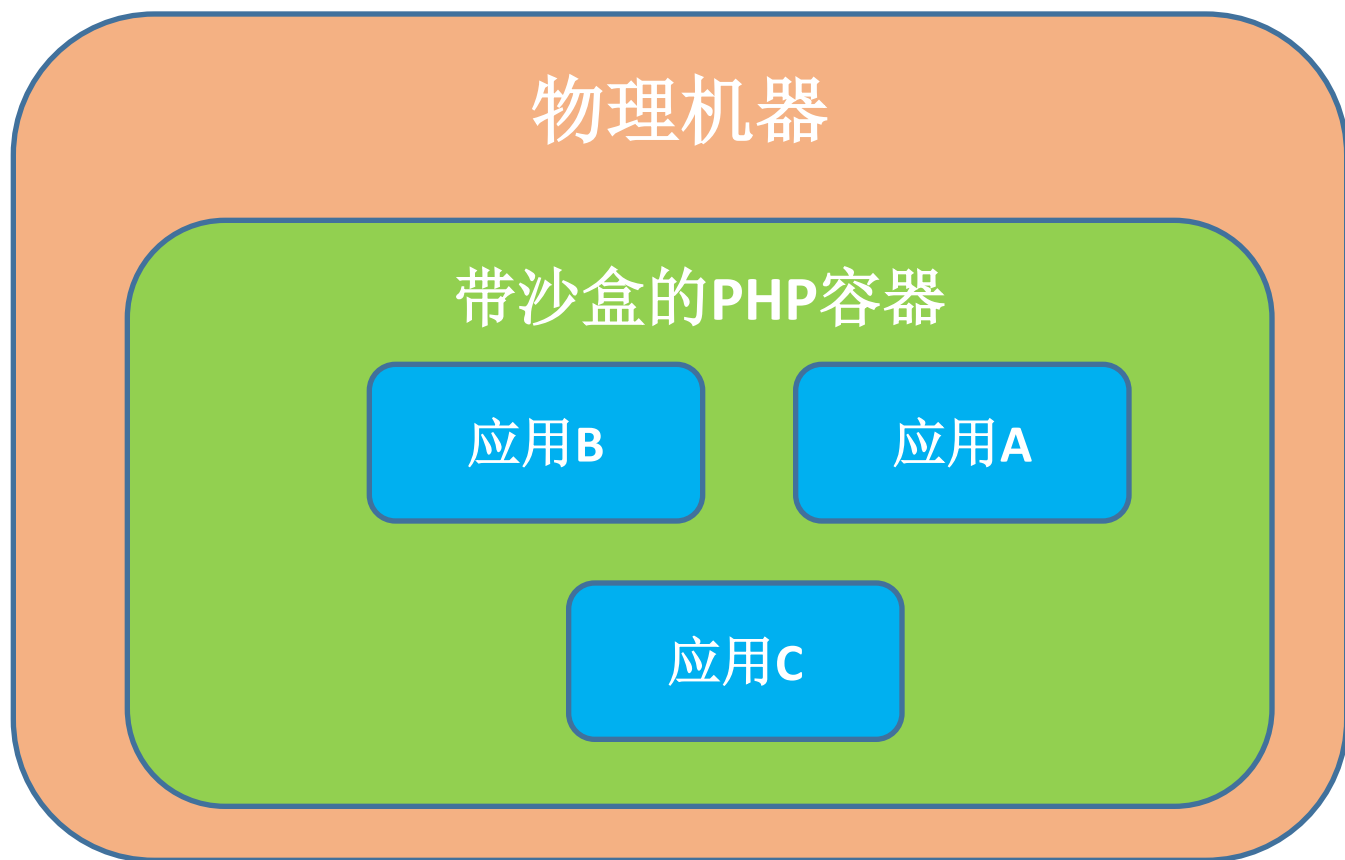
- 传统PAAS平台的困境
 - GAE是始作俑者
 - 通过沙盒技术来实现资源隔离与资源限制
 - 实现python沙盒的经历
 - 平台的开发和维护成本高
 - 诸多限制，学习成本高
 - 应用的迁移和开发成本高
 - 开发者的抱怨多，不爽



BAE与docker

--BAE为什么选择docker?

- 传统PAAS平台的困境



BAE与docker

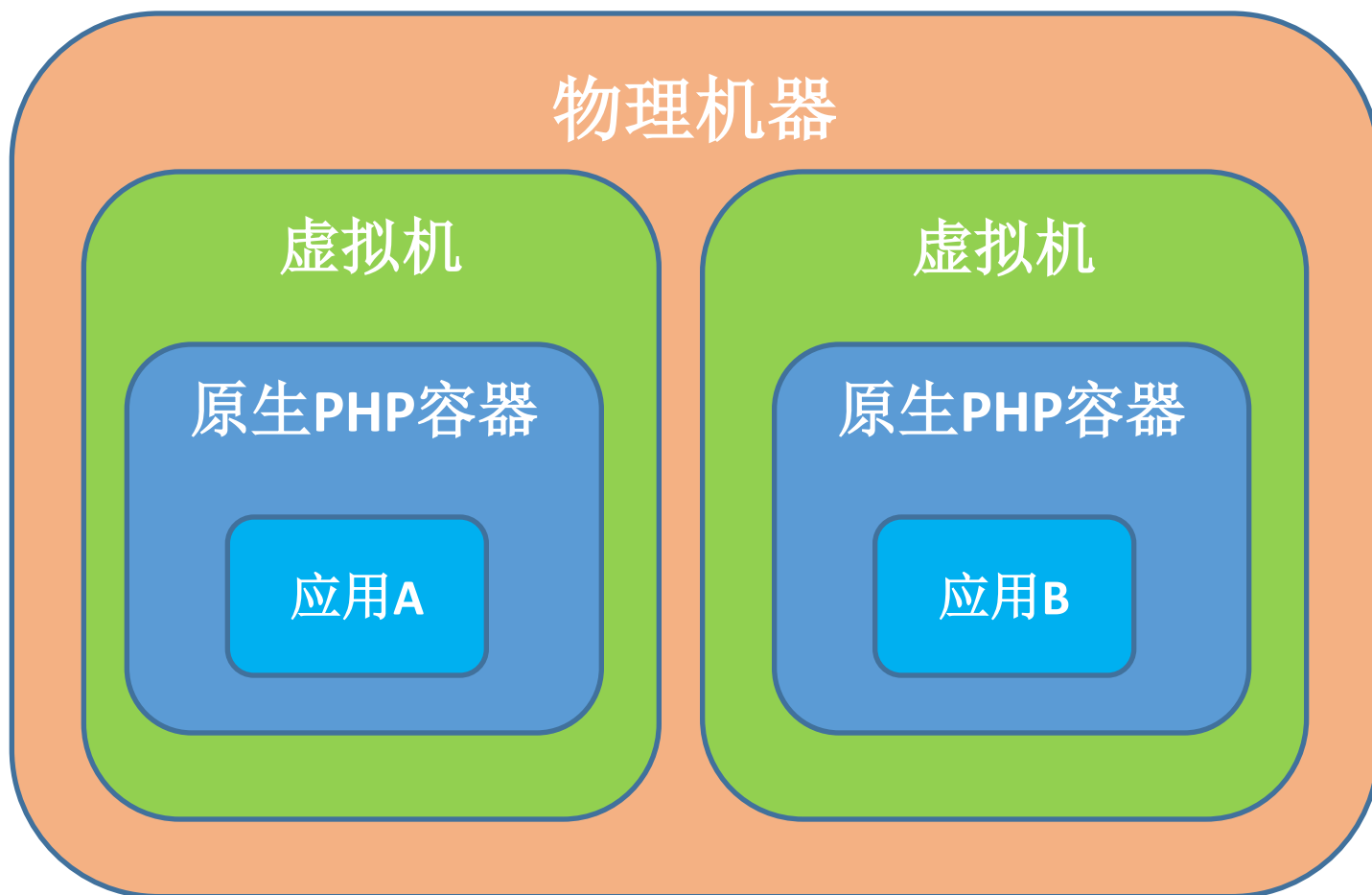
--BAE为什么选择docker?

- Container技术带来了光明
 - 干掉沙盒；通过**Container技术**在外层进行资源隔离与限制
 - 无任何语言层面的限制，开发者学习成本显著降低
 - 支持新的编程语极其简单，平台开发和维护成本降低
- 业界趋势：新兴PAAS平台纷纷选择Container
 - Cloudfoudry/openshift/heroku/dotcloud/appfog

BAE与docker

--BAE为什么选择docker?

- Container技术带来了光明



BAE与docker

--BAE为什么选择docker?

- BAE2.0平台是一种基于沙盒技术的PAAS
- 我们也深受传统PAAS平台的困扰
- 注意到Container技术，开始前期的调研和摸索

BAE与docker

--技术选型一： 自己开发

- 首先和公司内部虚拟化团队合作摸索了一套方案：
 - openstack + libvirt
 - 功能上能满足基本需求
- 问题
 - 没有考虑到最重要的一个需求：
 - 需要频繁的创建和删除Container
 - 要求创建和删除Container能在几秒内完成
 - 实际创建Container要15秒以上，无法接受
 - Openstack 画蛇添足
 - 对代码质量没有足够信心
 - 缺少后续技术支持

BAE与docker

--技术选型二: warden

- cloudfoundry的warden
 - warden 是一个比较完整的解决方案
 - 但是:
 - Ruby不熟悉, 心里没底
 - dea与warden耦合太紧
 - 社区活跃度不够



BAE与docker

--与docker的偶遇

- 一次偶然机会，与docker布道师 Jerome Petazzoni 的交流
- 对docker的评估
 - 功能上满足主要需求
 - 强悍：虚拟机的快速创建和删除能力
 - 强悍：增量式的镜像管理能力
 - 代码简单易读，出问题有信心自己可以搞定
 - 有dotcloud公司支持，社区活跃度高，看好它的未来



BAE与docker

-- 如何使用docker?

- Docker的资源隔离功能基本满足需求
- 主要的开发工作
 - 更好的资源限制
 - 更全面的安全限制
 - 公共PAAS平台，安全最重要

BAE与docker

-- 如何使用docker?

- docker私有仓库，解决镜像管理问题

docker的发展

--逐步繁荣的生态圈

- CoreOS
- Yandex – Cocaine
- Flynn -- <https://flynn.io/>
- 最新版OpenStack Havana 增加对docker的支持

docker的发展

--与redhat的合作

- Docker发展的关键一步

- 支持libvirt，充分利用 libvirt强大的网络管理能力
- 利用device-mapper技术解决对AUFS的依赖
- 利用openshift使用SELinux的成熟经验，解决安全问题
- docker将可以运行在红帽旗下的Linux发行版本上
- Openshift 集成对docker的支持

docker的发展

--待解决的主要问题

- 安全
 - user namespace来解决root权限问题
 - SELinux增强安全性
- 支持更多的Linux发行版
 - 由于AUFS的限制，目前仅支持 ubuntu
- 支持更多的架构
 - 目前仅支持x86-64

docker 未来展望

- 将成为这个领域的领头羊
- Cloudfoundry, openshift 将支持docker
- 基于docker的开源PAAS平台将诞生，且使用GO语言开发
- 越来越多的新兴开源项目将基于docker展开

docker 未来展望

- 研发、运维、架构师、学生等小伙伴们，docker 值得你去投资

参考资料

- <http://www.infoq.com/articles/docker-containers>
- http://en.wikipedia.org/wiki/Operating_system-level_virtualization
- <http://en.wikipedia.org/wiki/LXC>
- <http://en.wikipedia.org/wiki/Cgroups>
- http://en.wikipedia.org/wiki/Linux_Containers
- <http://en.wikipedia.org/wiki/Aufs>
- <http://libvirt.org/>
- <http://linuxcontainers.org/>
- <https://wiki.ubuntu.com/LxcSecurity>
- <http://lwn.net/Articles/236038/>

参考资料

- <http://blog.dotcloud.com/under-the-hood-linux-kernels-on-dotcloud-part>
- <http://blog.dotcloud.com/kernel-secrets-from-the-paas-garage-part-24-c>
- http://www.nsnam.org/wiki/index.php/HOWTO_Use_Linux_Containers_to_set_up_virtual_networks
- http://openvz.org/Main_Page
- <http://aufs.sourceforge.net/>
- <http://blog.docker.io/2013/08/containers-docker-how-secure-are-they/>

谢谢大家