

DaoCloud 的 Docker 监控之道

The Power of Native

Kay Yan
「DaoCloud」

www.daocloud.io

1990



硬件·虚拟化

新的硬件架构及指令集，支持在单一硬件上运行多个租户操作系统。

IBM/Intel



计算·虚拟化

传统的硬件架构上，Hypervisor解析执行操作系统到硬件的指令集。

VMware/KVM



2000



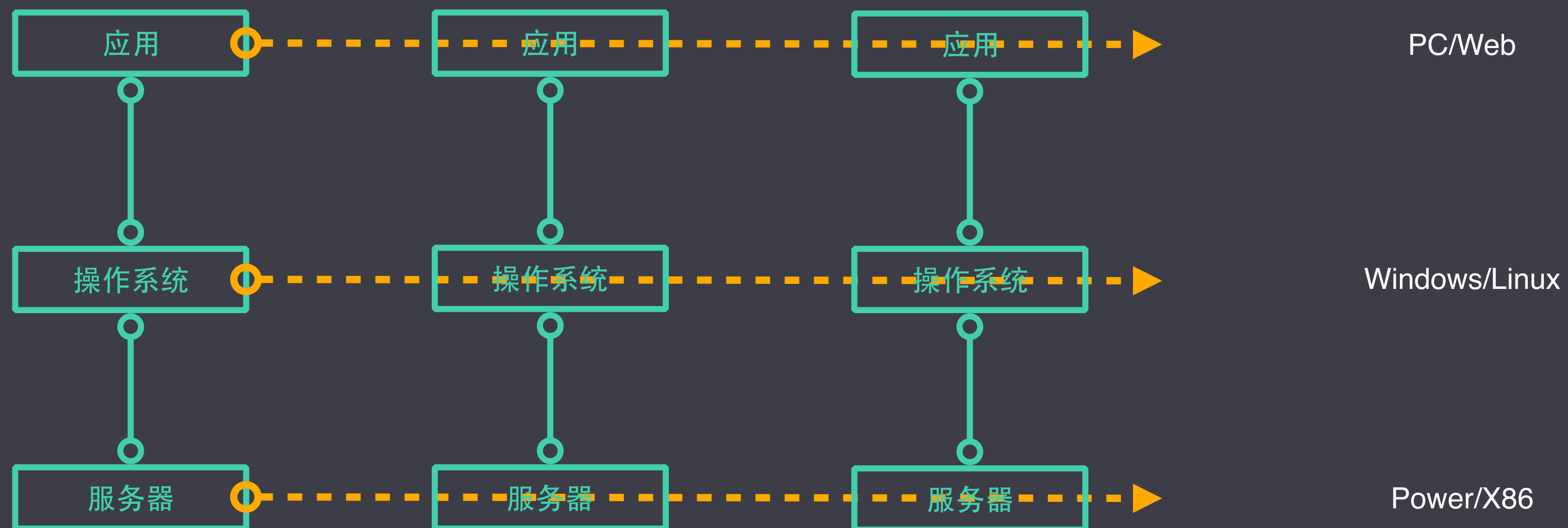
2010

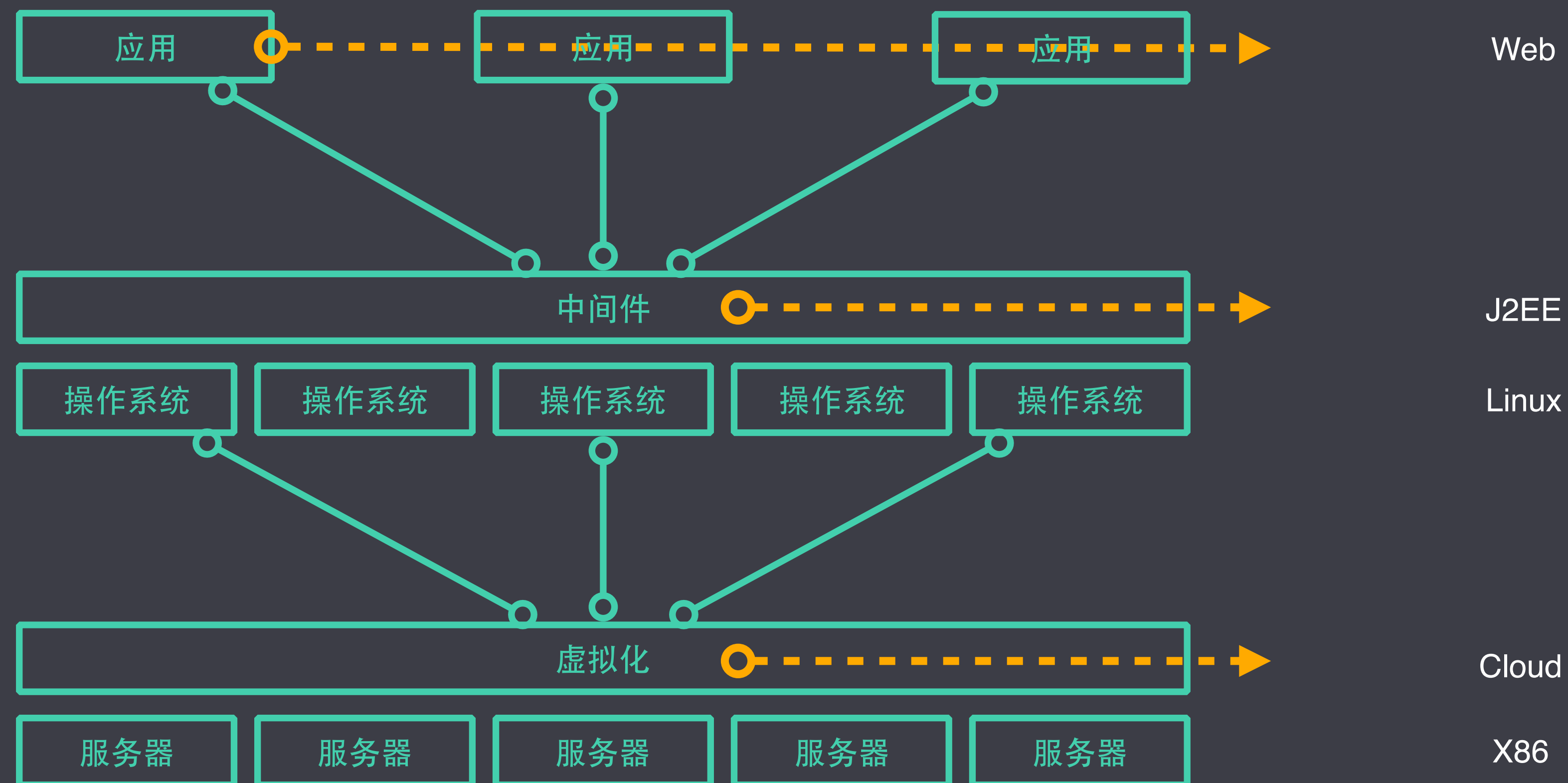


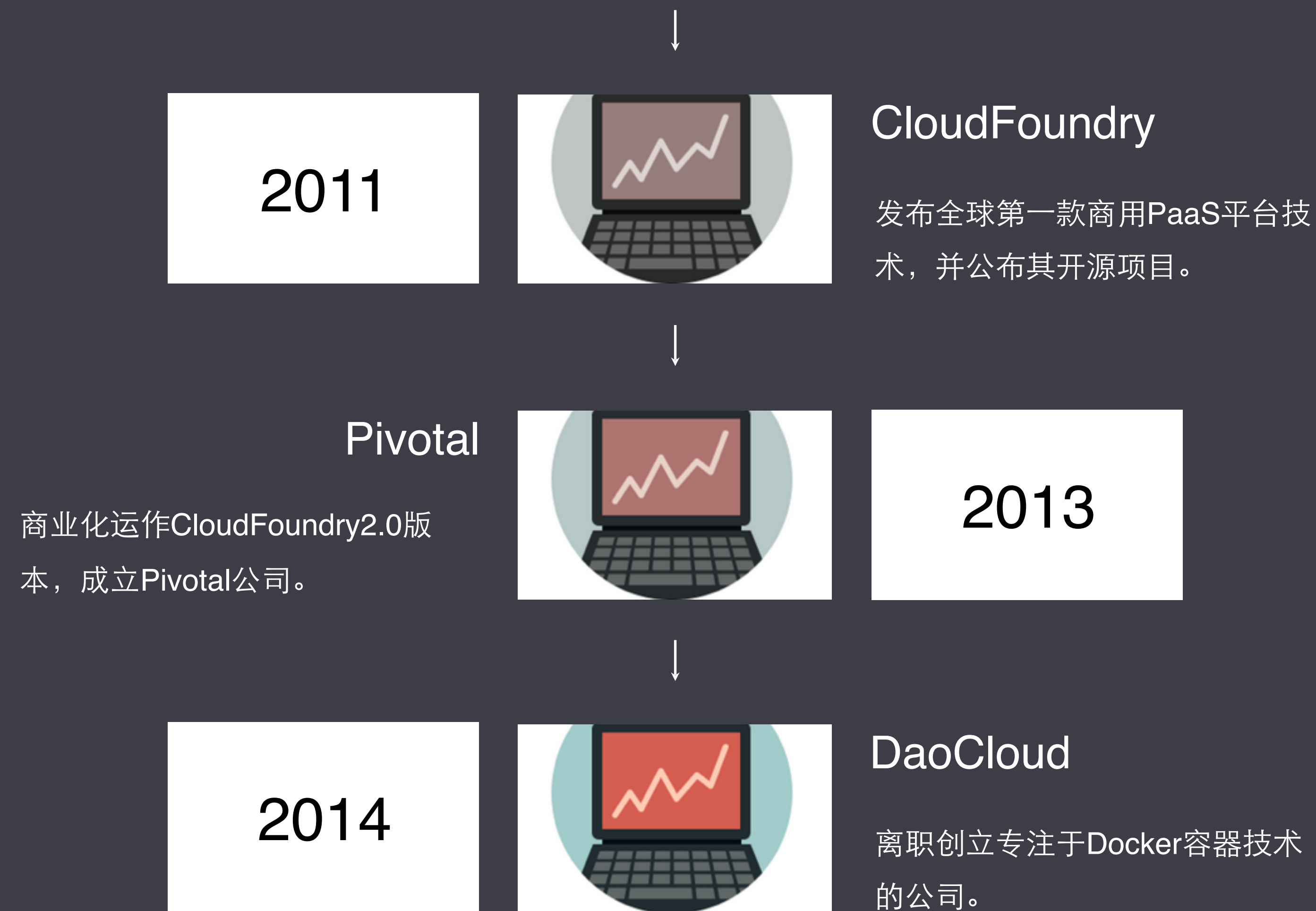
应用·虚拟化

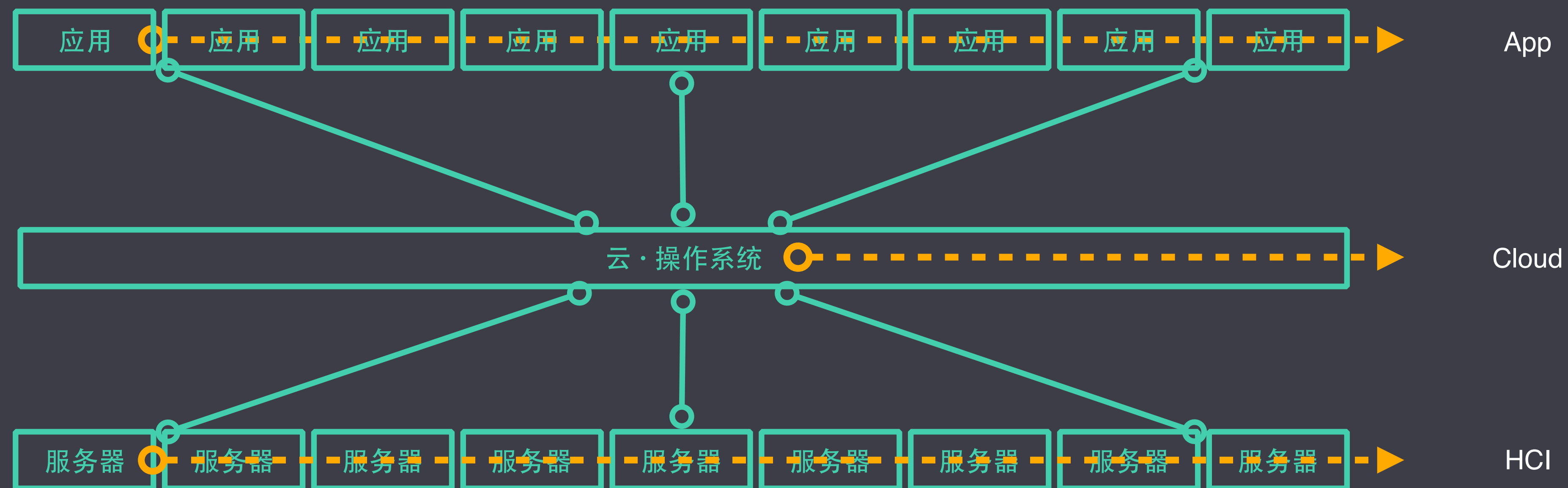
Linux系统上，通过调用内核特性，以进程的方式运行多个应用容器。

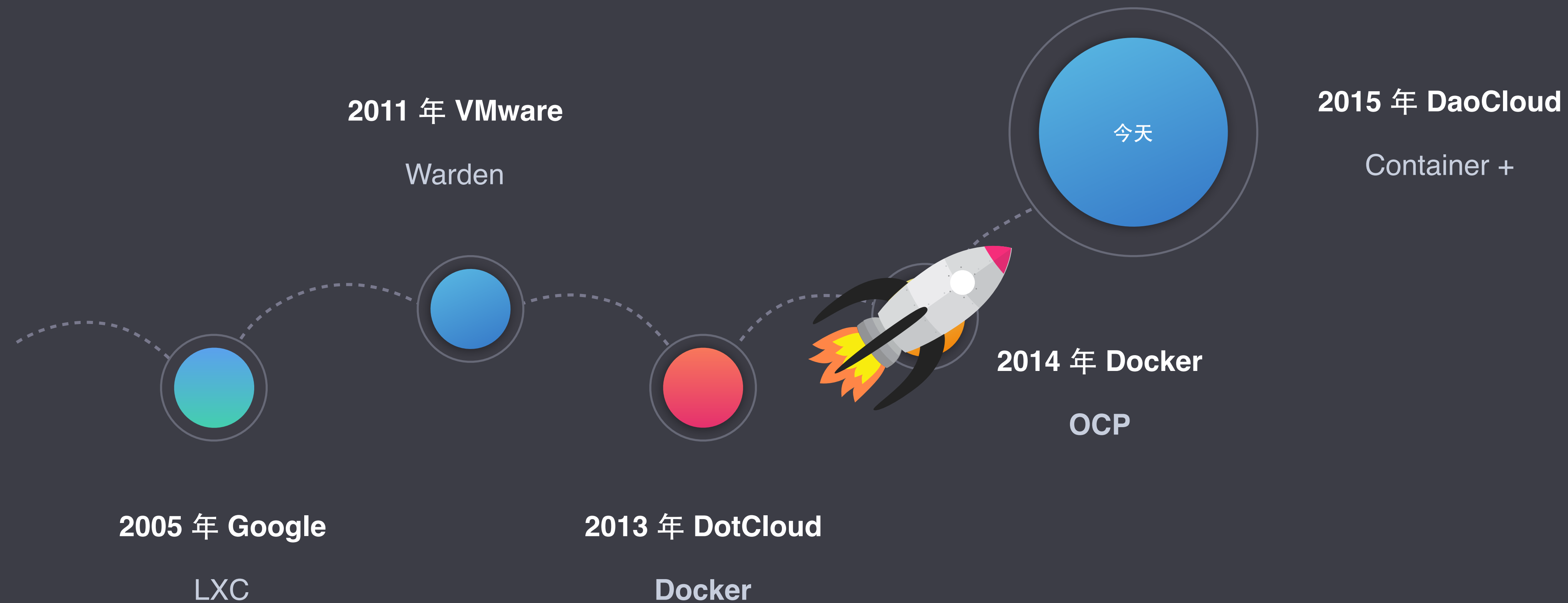
LXC/Warden













容器技术不等同于轻量级虚拟化技术，适合传统IT转互联网架构的应用场景。

特性

- 硬件接口
- 操作系统
- 运行空间
- 隔离策略
- 系统资源
- 启动时间
- 镜像存储
- 集群规模
- 高可用策略

虚拟机

- 模拟
- 广泛
- 用户级
- Hypervisor
- 5-15%
- 分钟级
- GB-TB
- 近百
- 备份，容灾，迁移

容器

- 直接
- Linux为主
- 内核级
- CGroups
- 0-5%
- 秒级
- KB-MB
- 上万
- 弹性，负载，动态



Docker in 2013

A lightweight runtime and robust tooling to build and run virtual containers on Linux to create the operating environment for your distributed applications.

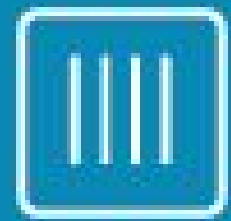


Docker Basics



Docker Image

The basis of a Docker container



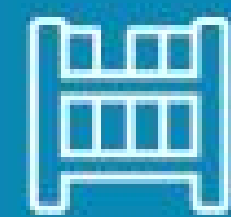
Docker Container

The standard unit in which the application service resides



Docker Engine

Creates, ships and runs Docker containers deployable on physical or virtual host locally, in a datacenter or cloud service provider



Docker Registry

On-premises registry for image storing and collaboration



特性	非容器	容器
应用列表	ps grep	docker ps
进程列表	top	docker top
端口信息	netstat	docker ps
网络流量	ifstat	docker stats
磁盘性能	iostat	docker stats
日志	tail -f /var/log	docker logs
磁盘信息	isof	docker volume

Docker in 2016

A Containers as a Service (CaaS) platform which enables IT Ops managed and secure application environment (infrastructure and content) for developer self service to build and deploy applications





Production Ready for 1.5 year

40%

Docker users already running in production

85%

using Swarm & Compose

5X Real Docker adoption last year



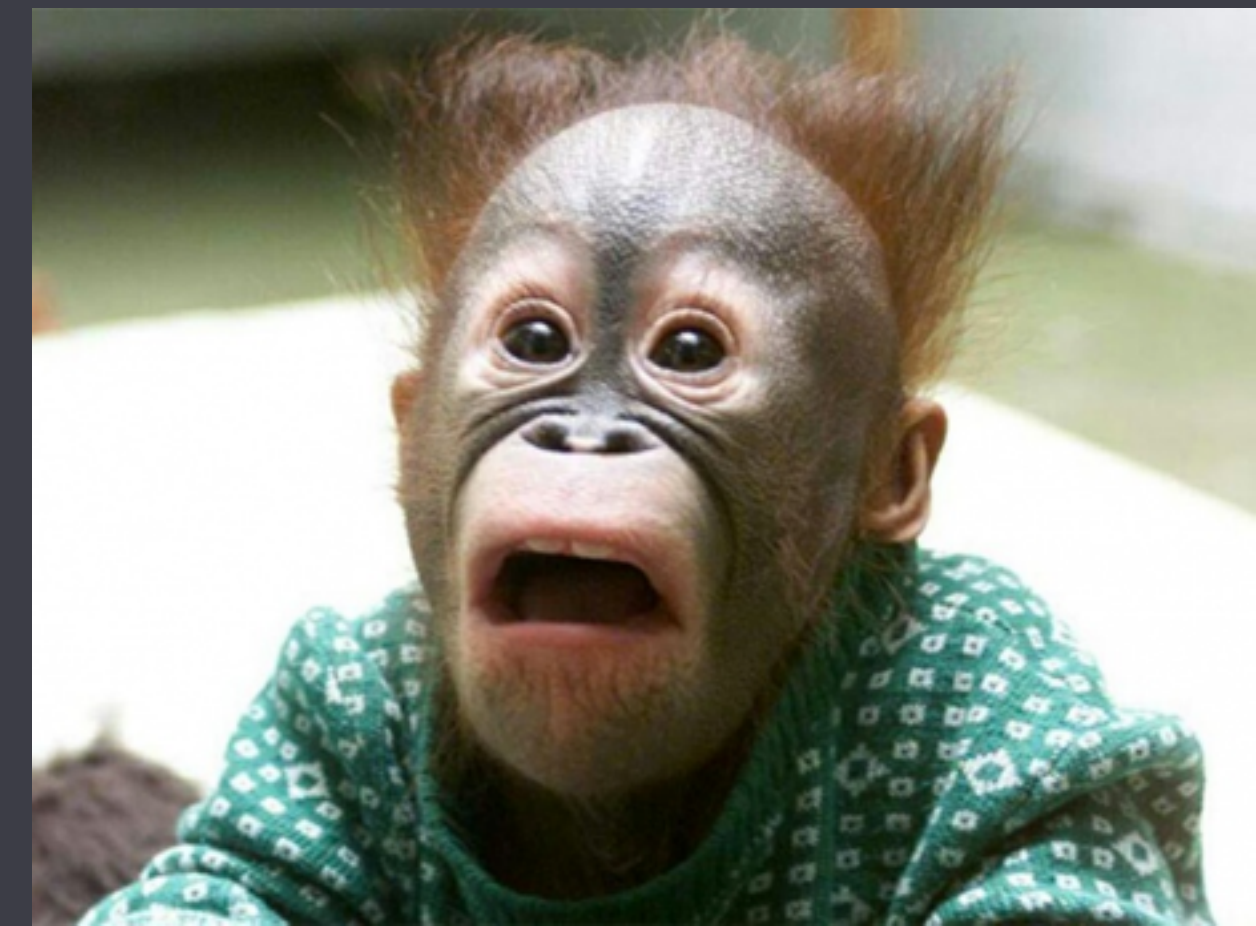
Out of Box overlay networking - SDN



Manage network as a top-level command

6

More than 6 implements today



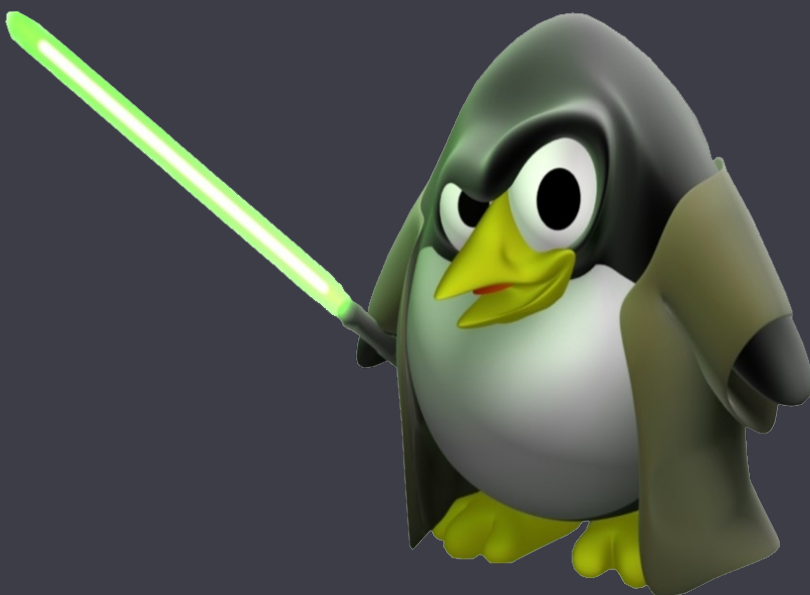
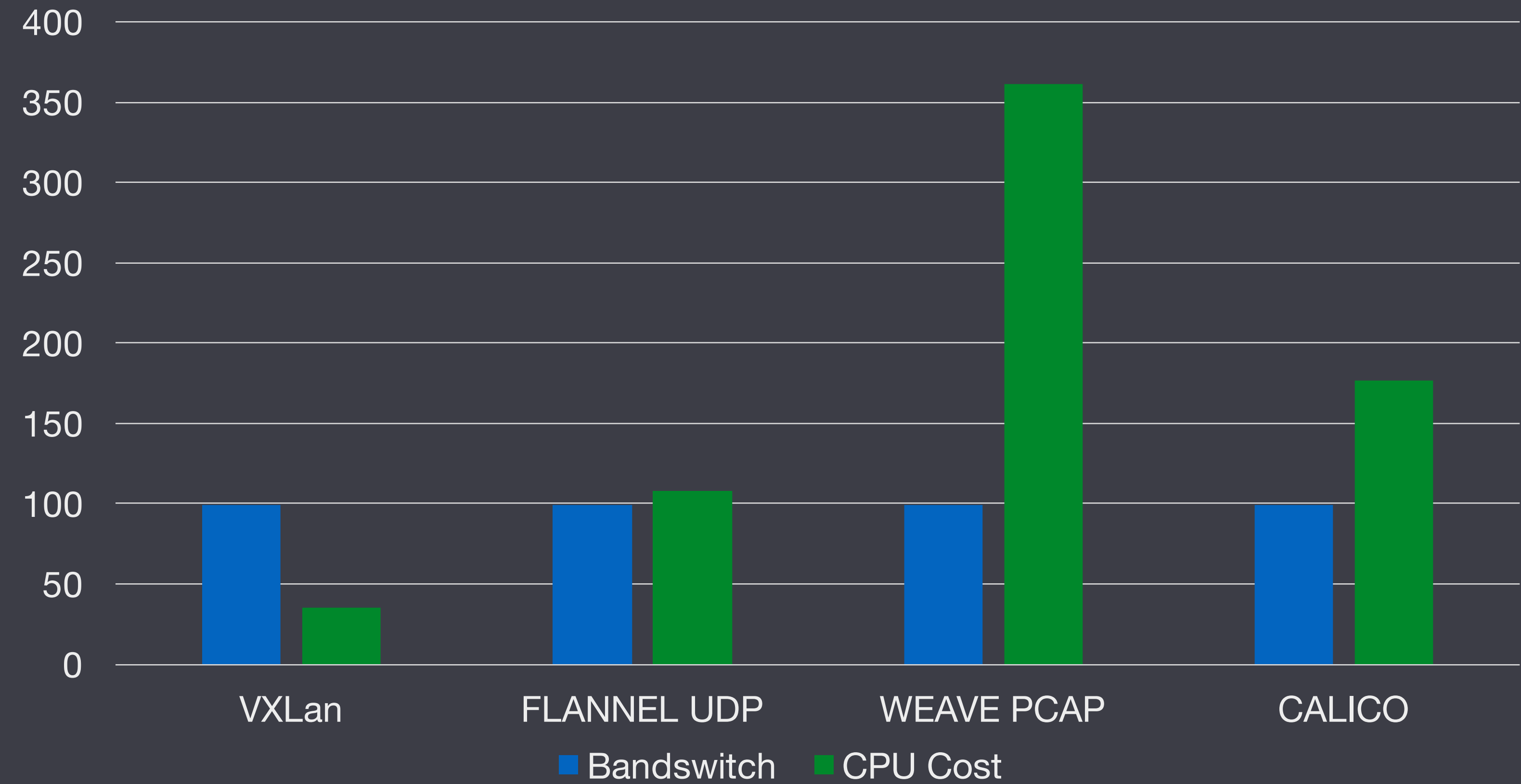
Container-Container Network

IP address per container

Seamless, does not change the application code



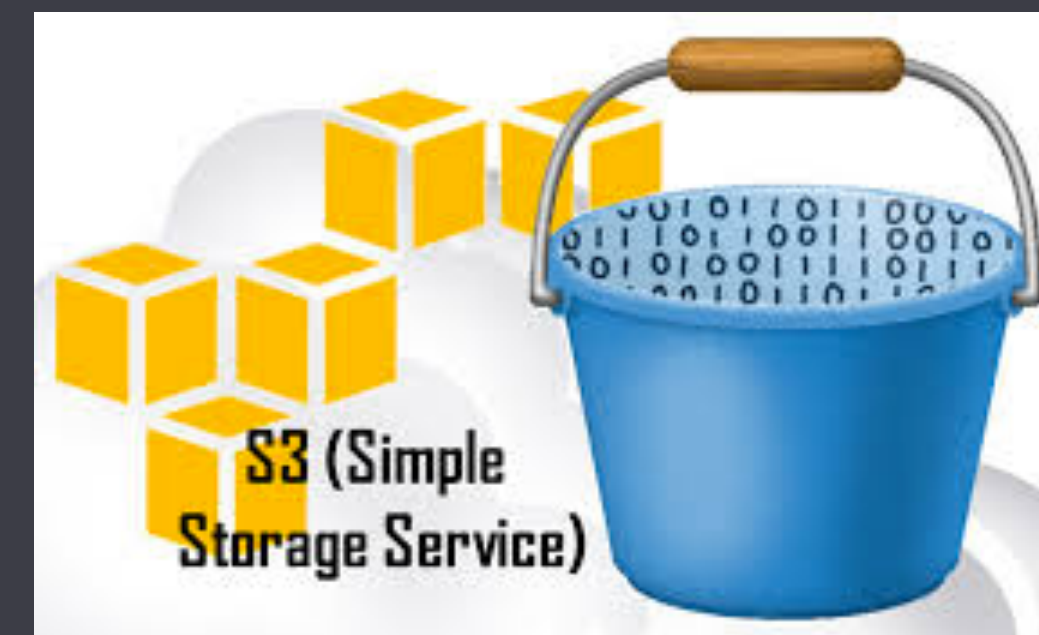
Overlay Network benchmark

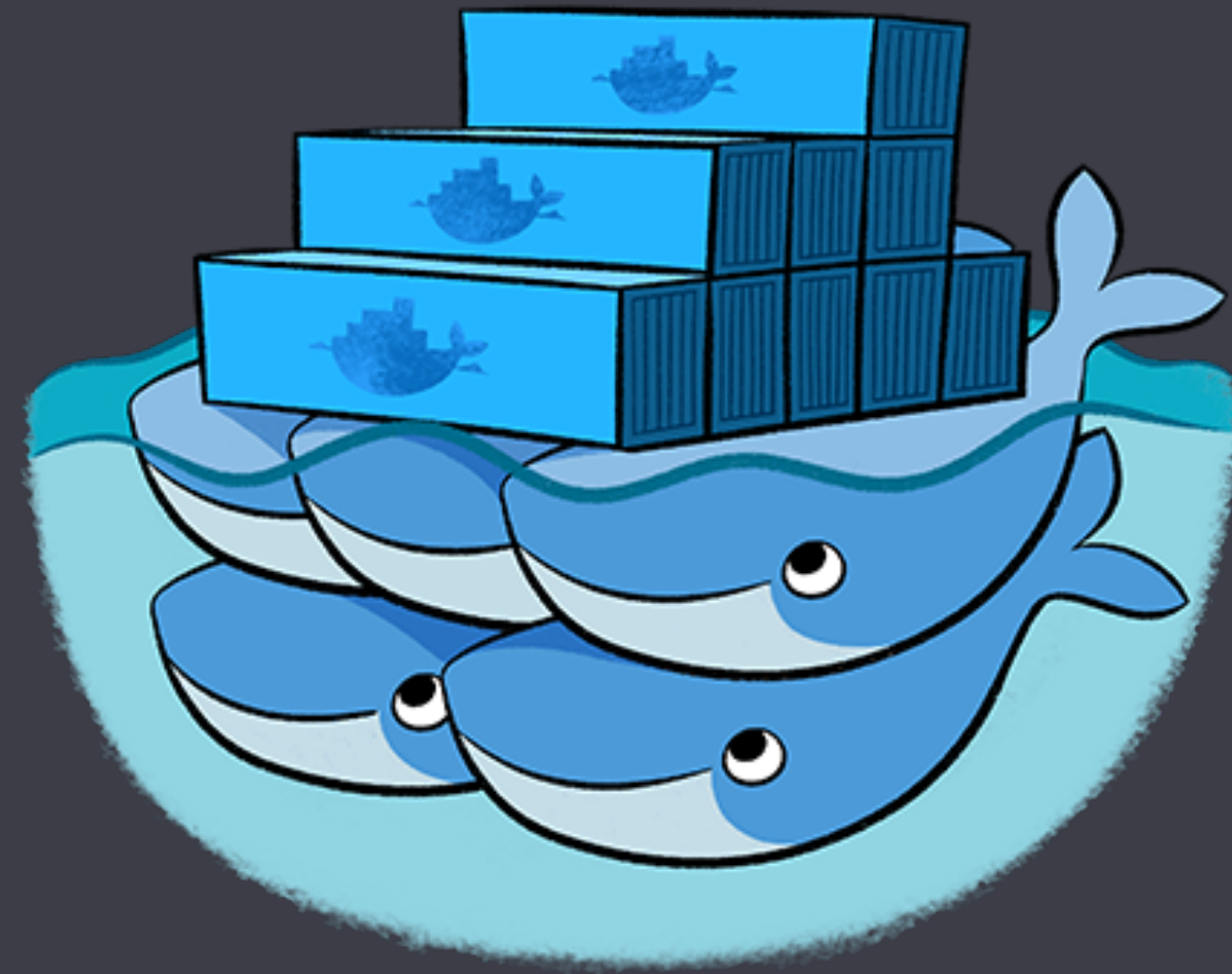


Build-in persistent Storage

Attach any volume to any container, dynamically

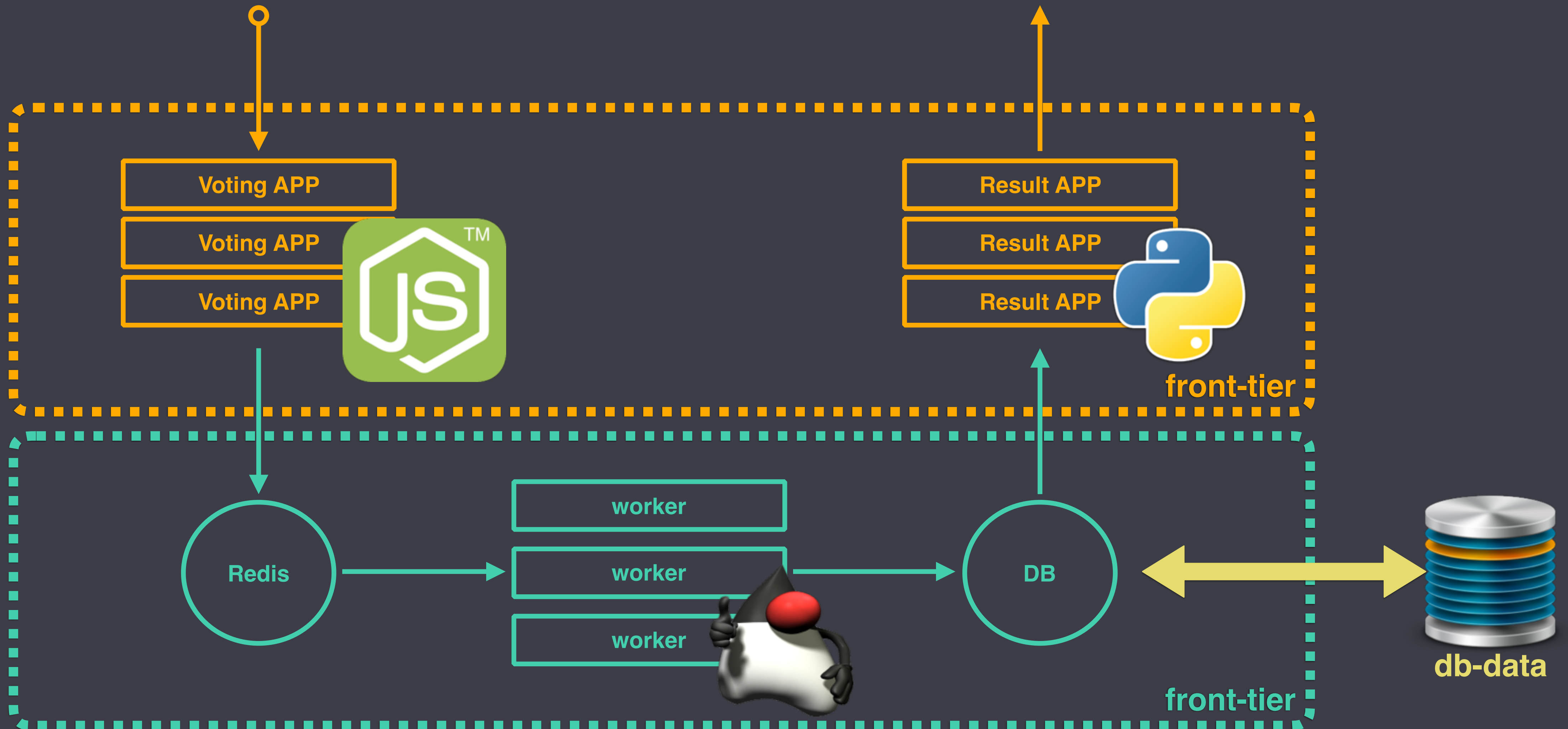
Multi backends





Ready for production

From Complex Architecture



From Easy Compose

```
version: "2"
services:
  voting-app:
    image: voting-app
    ports:
      - "80"
    networks:
      - front-tier
      - back-tier
  result-app:
    image: result-app
    ports:
      - "80"
    networks:
      - front-tier
      - back-tier
  worker:
    image: app-worker
    networks:
      - back-tier
```

接右边

接左边

```
redis:
  image: redis
  ports:
    - "6379"
  networks:
    - back-tier
  container_name: redis
db:
  image: postgres:9.4
  volumes:
    - "db-data:/data"
  networks:
    - back-tier
  container_name: db
volumes:
  db-data:
networks:
  front-tier:
  back-tier:
```


Compose UP !

请输入节点名称

容器: 0

主机: 10

这是一个10节点Docker/Swarm集群

Node 1000

Container 100000

Latency <150ms





服务 ▾

编辑 ▾

daocloud_adr

root@ip-172-31-12-56:~# docker info | grep Nodes

Nodes: 10

root@ip-172-31-12-56:~#

EC2 控制面板

事件

标签

限制

实例

实例

预留实例

映像

AMI

ELASTIC BLOCK STORE

卷

快照

网络与安全

安全组

弹性 IP

置放群组

密钥对

网络接口

负载均衡

负载均衡器

AUTO SCALING

启动配置

Auto Scaling 组

启动实例

连接

操作 ▾

按标签和属性筛选，或者按关键字搜索

1 到 50, 139

☐	Name	实例 ID	实例类型	实例
☐		i-72d49e4a	t2.micro	
☐		i-d6d49eee	t2.micro	
☐		i-0ad59f32	t2.micro	
☐		i-e8d49ed0	t2.micro	
☐		i-b2d79d8a	t2.micro	
☐		i-85d59fbd	t2.micro	
☐		i-8dd59fb5	t2.micro	
☐		i-53eba16b	t2.micro	
☐	dongpu-poc-002	i-e8e9a1d0	t2.medium	
☐	dongpu-poc-003	i-e9e9a1d1	t2.medium	
☐	dongpu-poc-004	i-ebe9a1d3	t2.medium	
☐	dockerhub-sync-prod	i-0b5b2d33	t2.small	

从以上选择一个实例

Docker/Swarm不但功能强大

Who know Ops' Life



We need Automation



DevOps + Docker + PaaS
开发-运维-云平台

规划设计



高速迭代

高可靠性



永远可用

纵向扩展



横向扩展

桌面应用

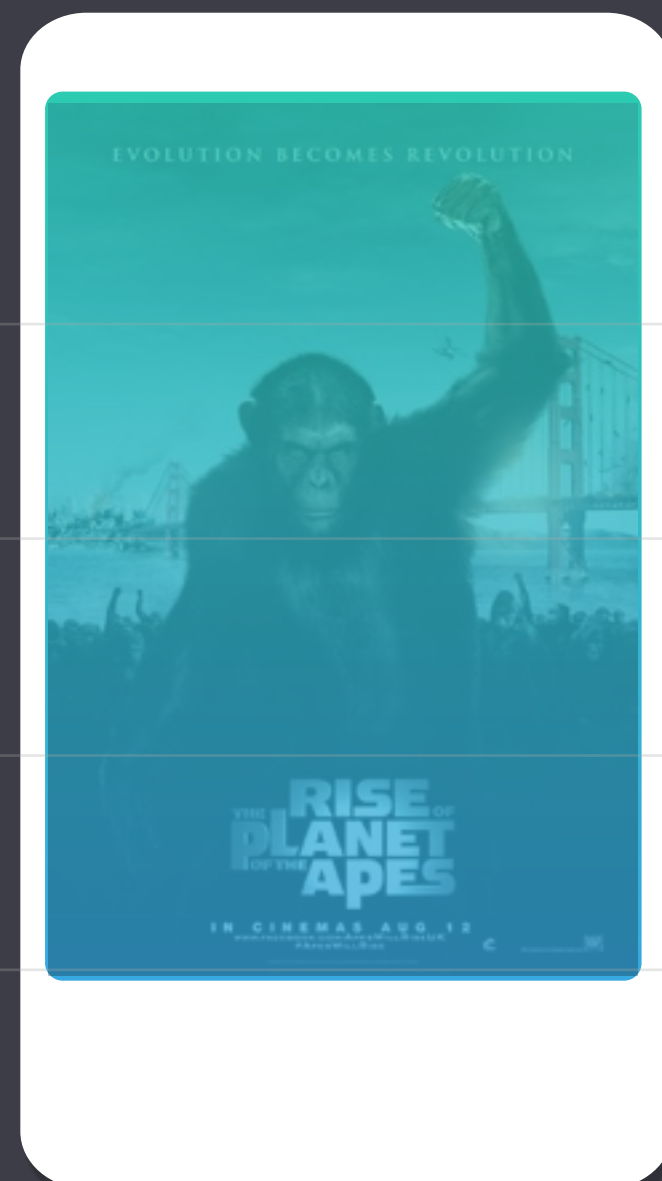


移动优先

控制互联



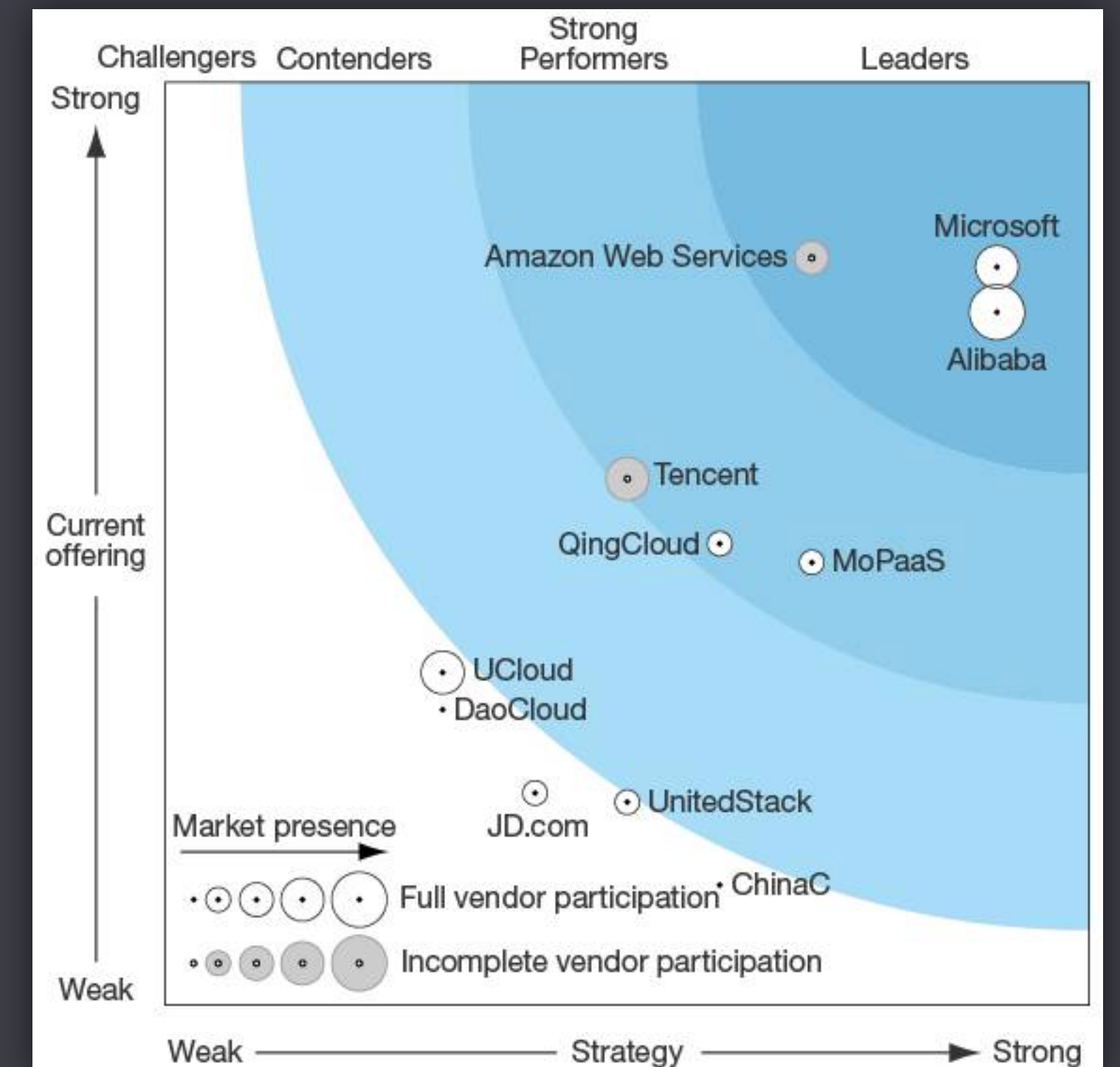
自治进化

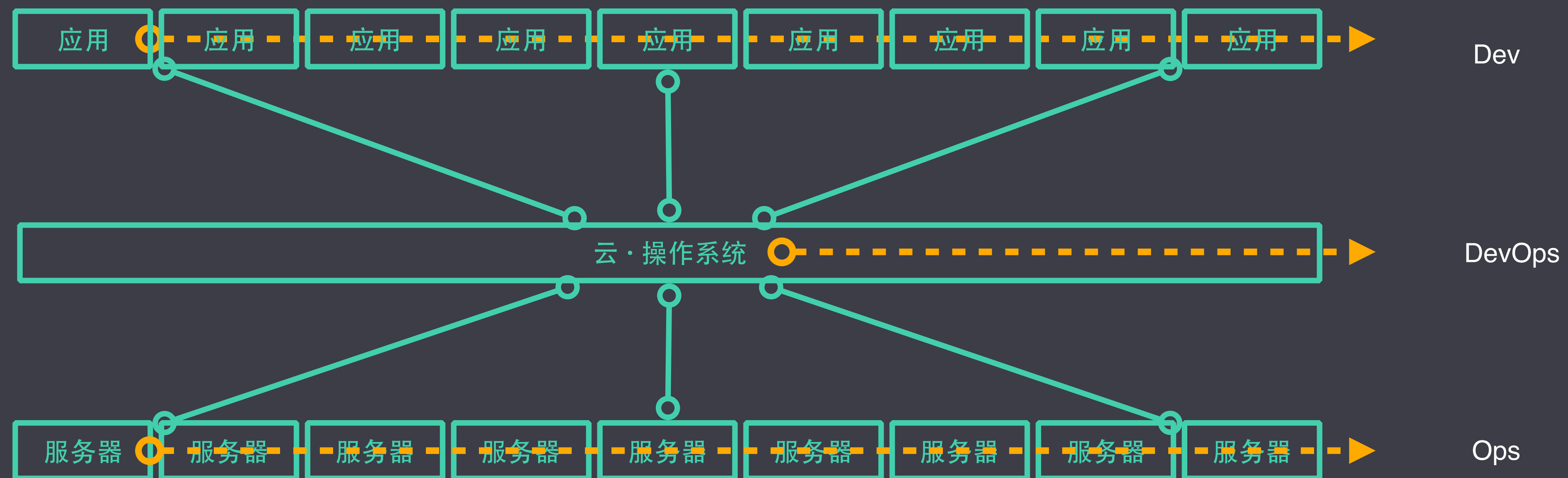


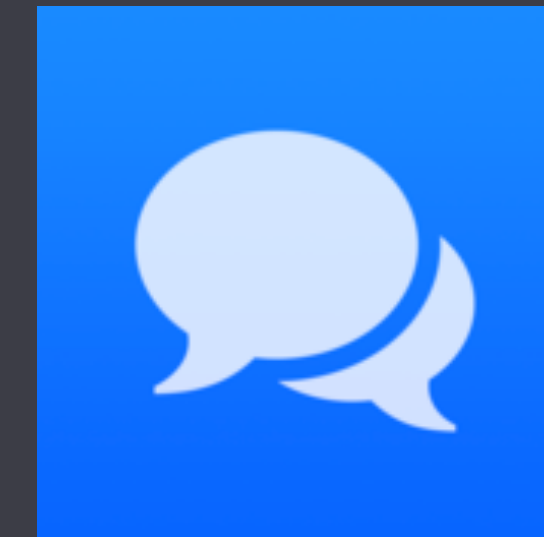
资源的云

VS.

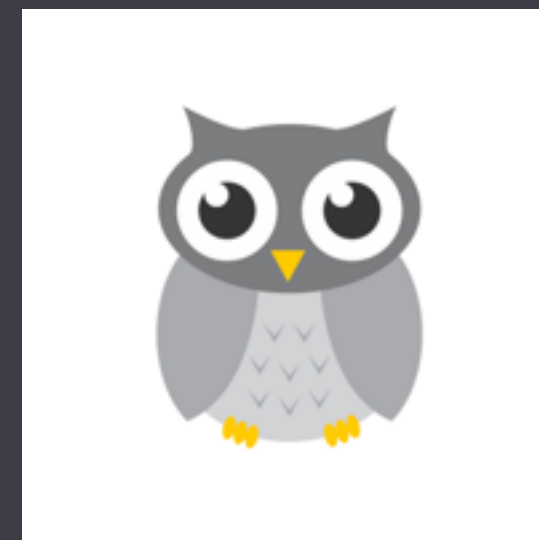
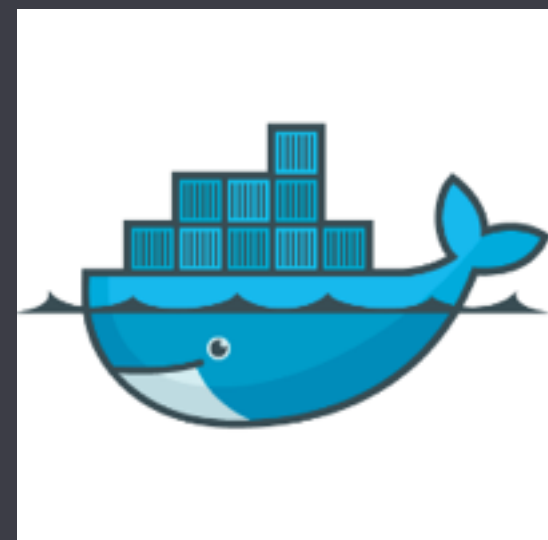
应用的云



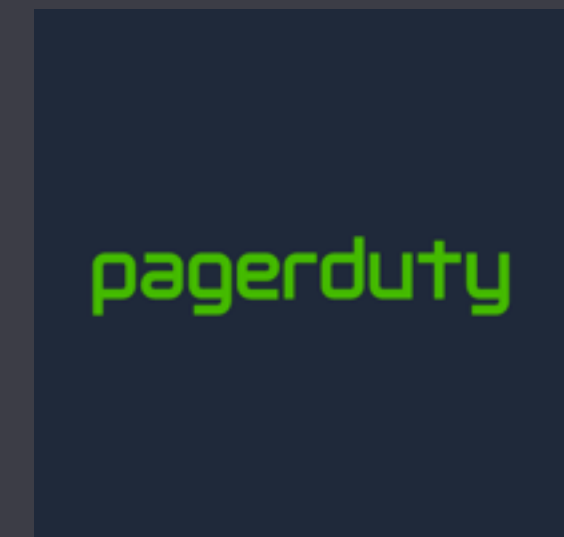




Dev



DevOps

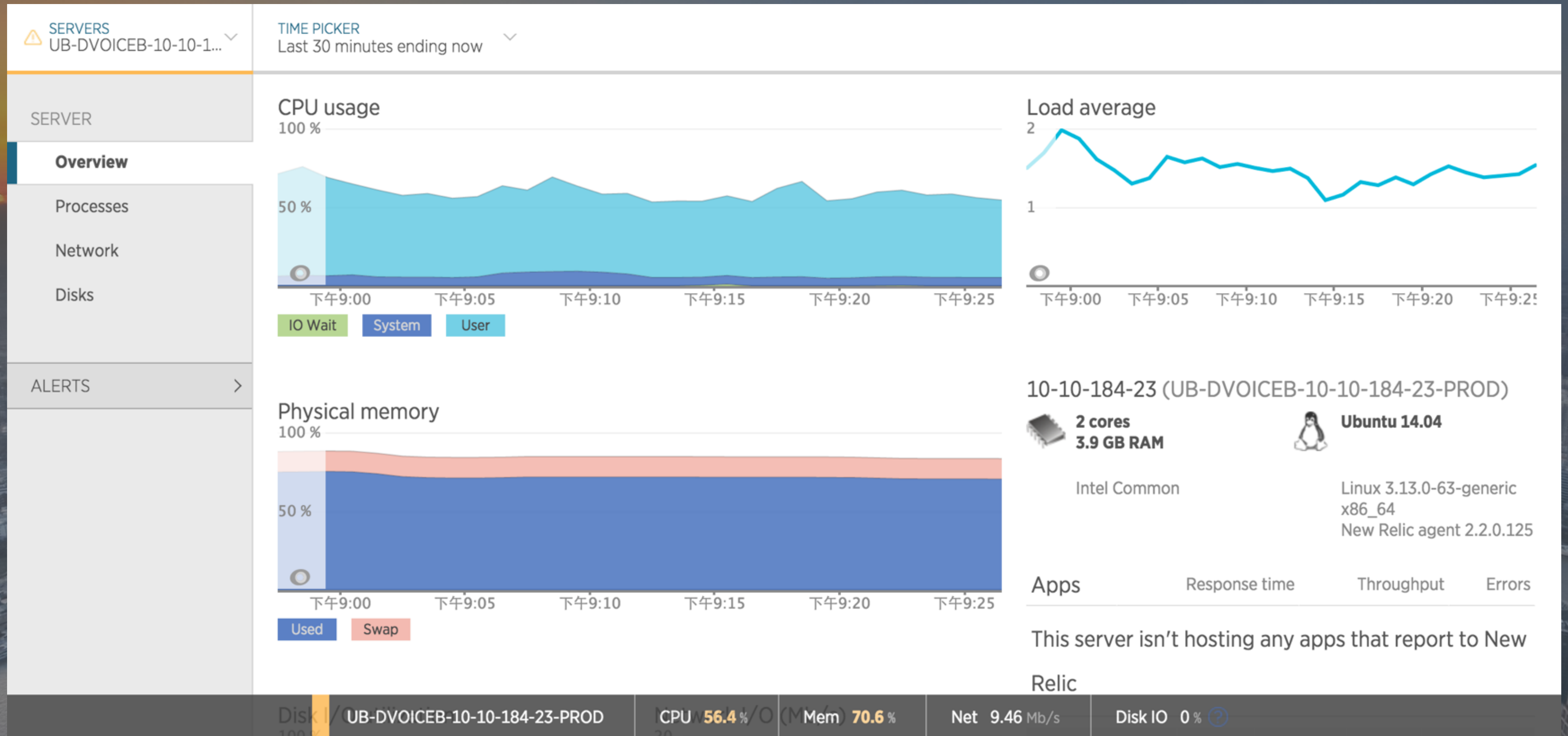


Ops



**BIG BROTHER IS
WATCHING YOU**

Infrastructure Monitor



您的系统还没有许可证，还可以继续试用 14 天，请立刻添加许可证



系统状态

健康

您当前的系统运行正常

控制器	正常
Swarm	正常
etcd	正常

告警

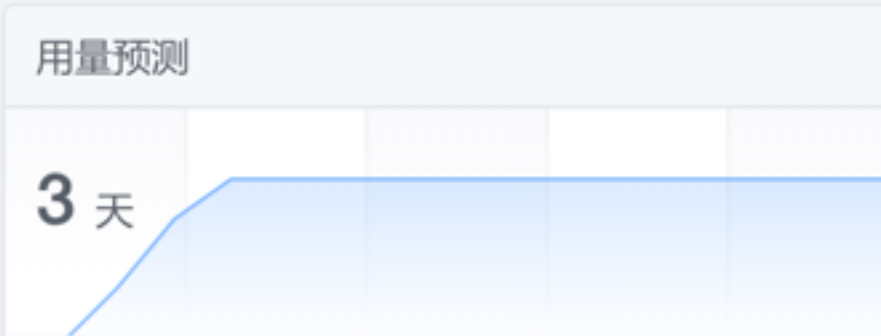
1

严重警告

您还未配置许可证，请联系客服申请试用



- ### 应用事件
- ww 部署成功
14 小时前 耗时 2 秒
 - a 扩展成功
17 小时前 耗时 6 秒
 - a 部署成功
17 小时前 耗时 182 秒



DaoCloud Enterprise 1.1.0-dev



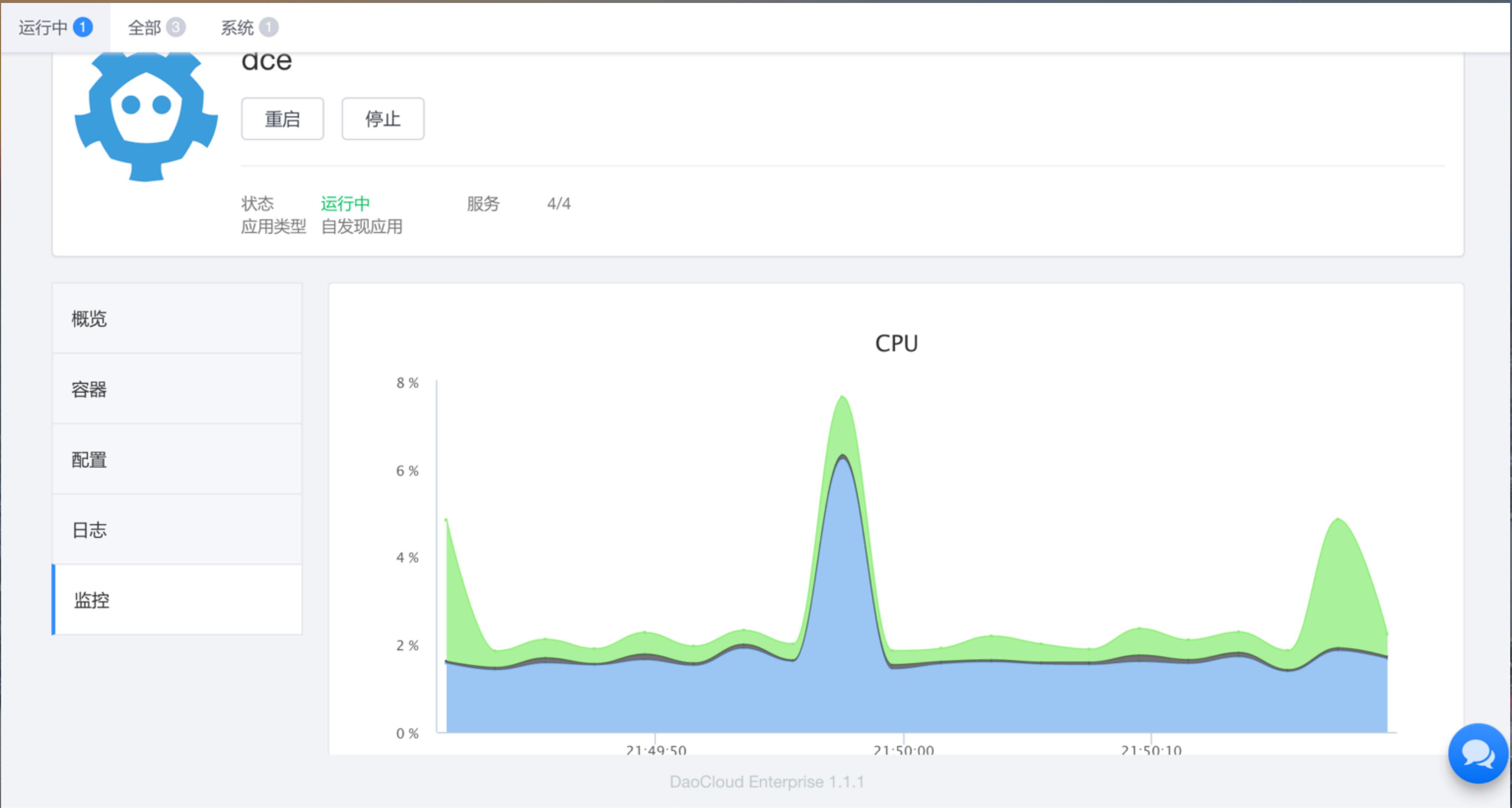
容量

性能

预测

告警

App Centric



应用状态

应用容器

应用日志

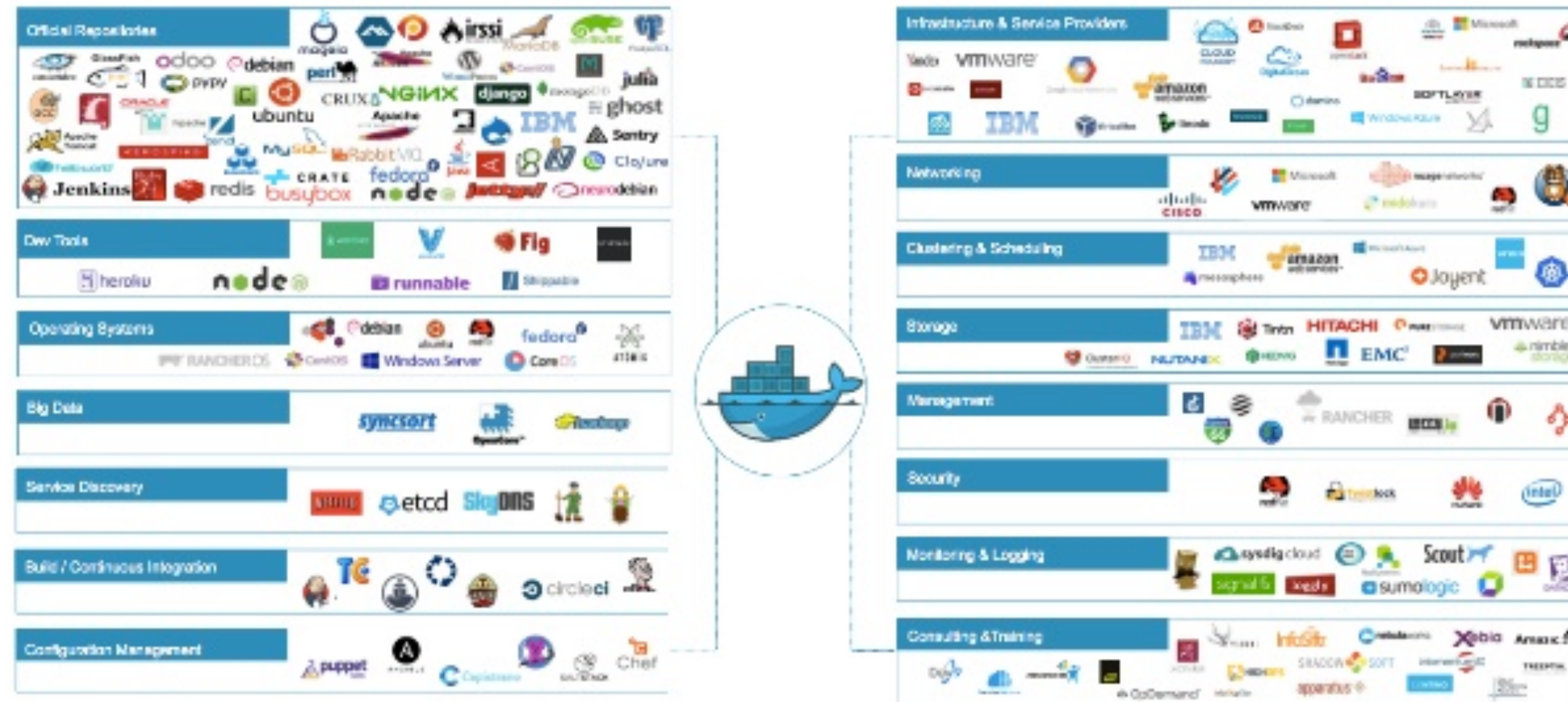
应用性能

Ecosystem

The Docker Ecosystem



Thank You to the Ecosystem



大家说好才是好





我 27

未指派 7

全部 193

Michael 6

Hypo 11

realityone 0

yeting 1

更多

 Hypo

baka

关闭对话



你好 构建通过了 为什么创建应用后 绑定的volume里面连我 git clone下来的仓库都没有？
2 小时前

您好，代码是在构建的时候clone的
1 小时前



HYPO 将此对话指派给了 HYPO 1 小时前



讲道理 clone进去了 我的 package.json 不然构建也过不去 但是部署的时候 绑定了 volume 直接报错 没找到 package.json
从 Web 发送, 1 小时前



回复 备注 发送



baka
用户
发消息

地区
中国南京

最近访问
7 分钟前

邮箱
songjiaxin2008@vip.qq.com

用户 ID
62747480-9565-4339-8589-3c

用户模型
社区版 活跃用户

用户标签
+ 添加标签...

Thank You!

