

Docker应用：超大规模容器调度系统的设计与实现

The Power of Native

Allen Sun
「DaoCloud」

www.daocloud.io

个人介绍



- DaoCloud技术合伙人，高级工程师
- 热爱golang&docker
- 《Docker源码分析》作者
- docker、swarm等项目committer

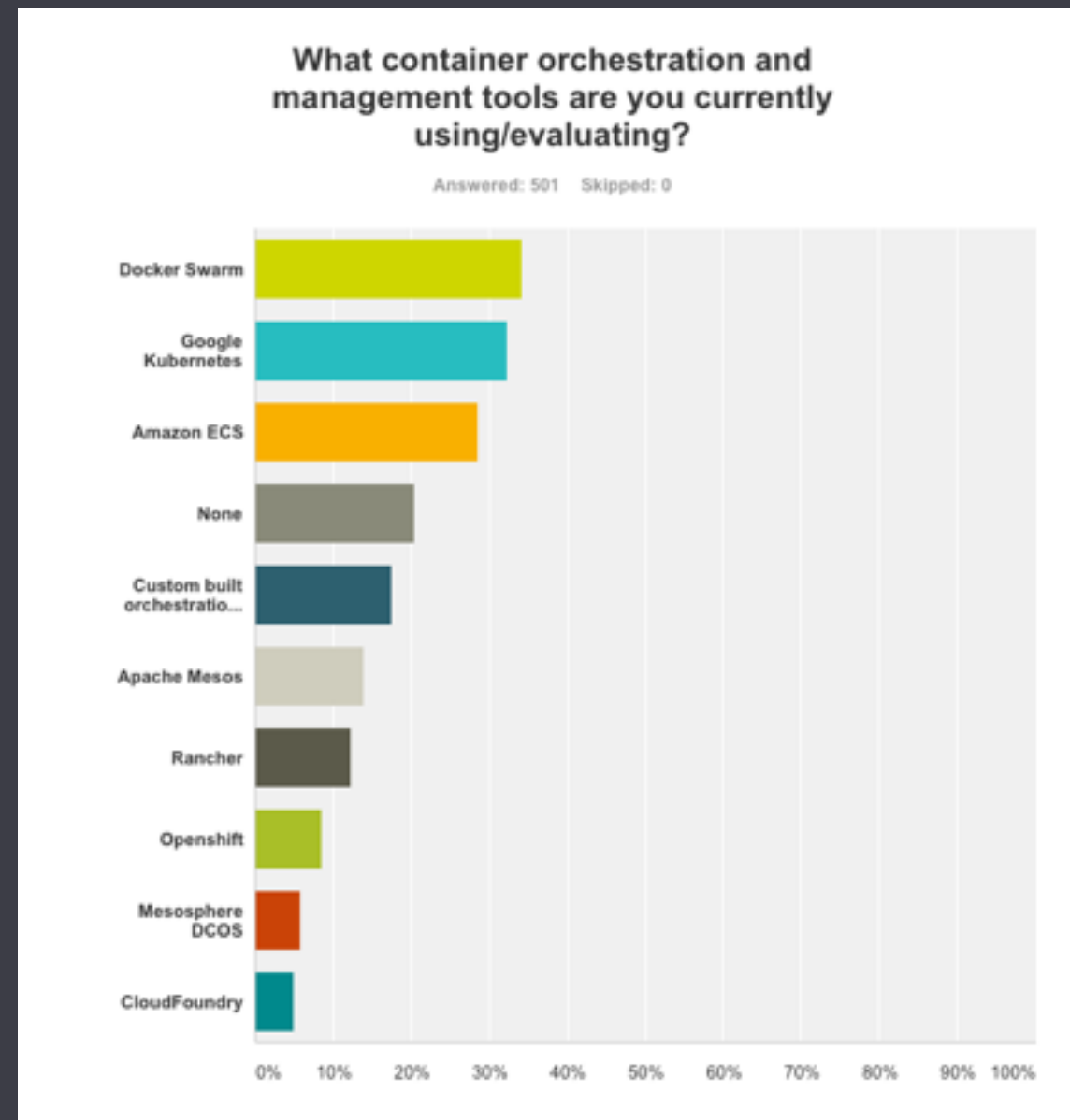


Agenda

- 什么是容器编排
- 容器编排与应用系统
- 容器编排与调度能力
- DCE架构分析

什么是容器编排

- 面向容器化应用系统
CNCF
微服务架构应用
- 容器调度能力
资源分配与调度
容器动态迁移
- 容器集群能力
容器通信能力
集群扩展能力



容器编排的评估指标

性能：

- 扩展速度
- 响应时间

灵活性：

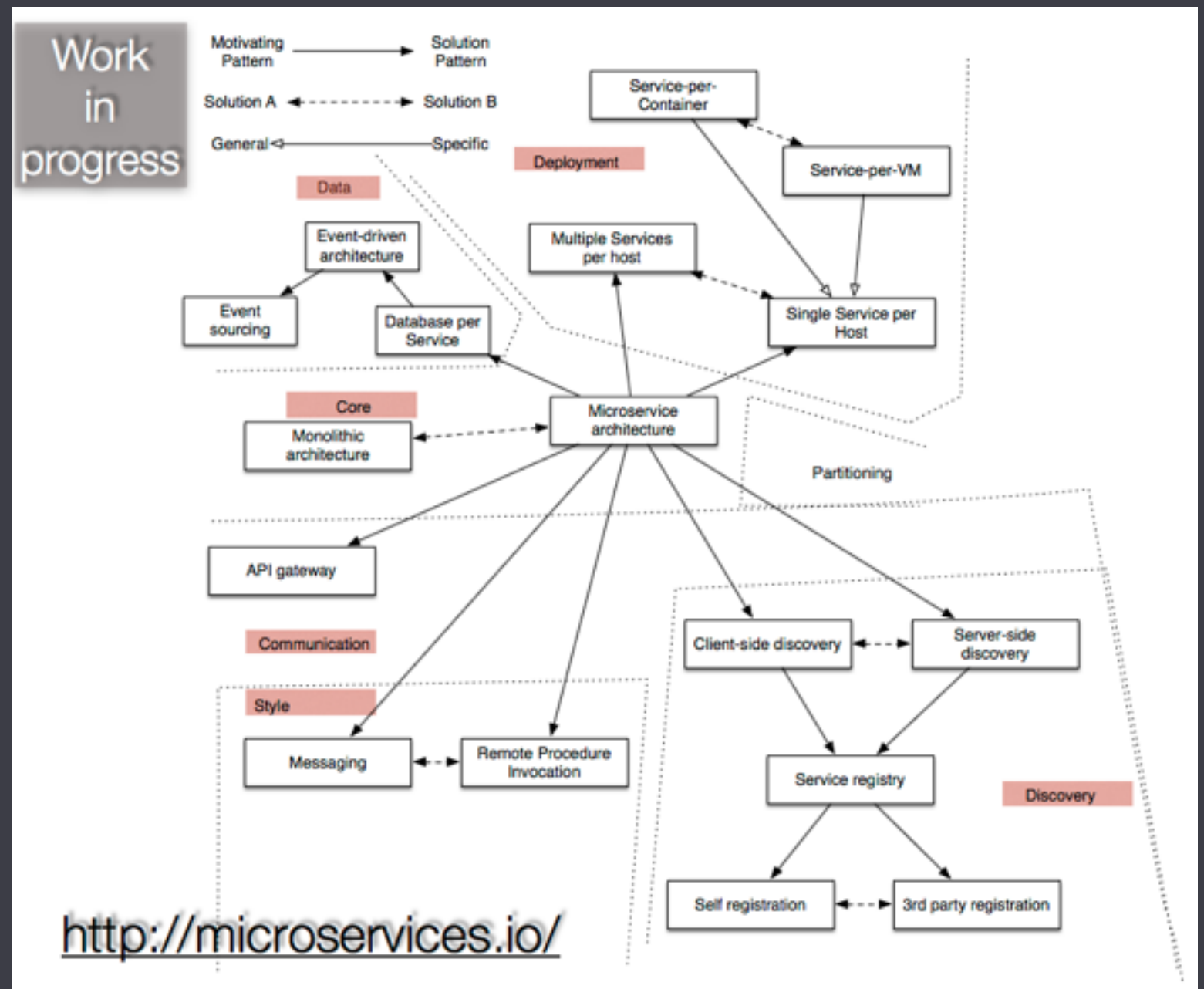
- 与现有工作流的集成程度
- 开发、测试、生产环境的平滑迁移

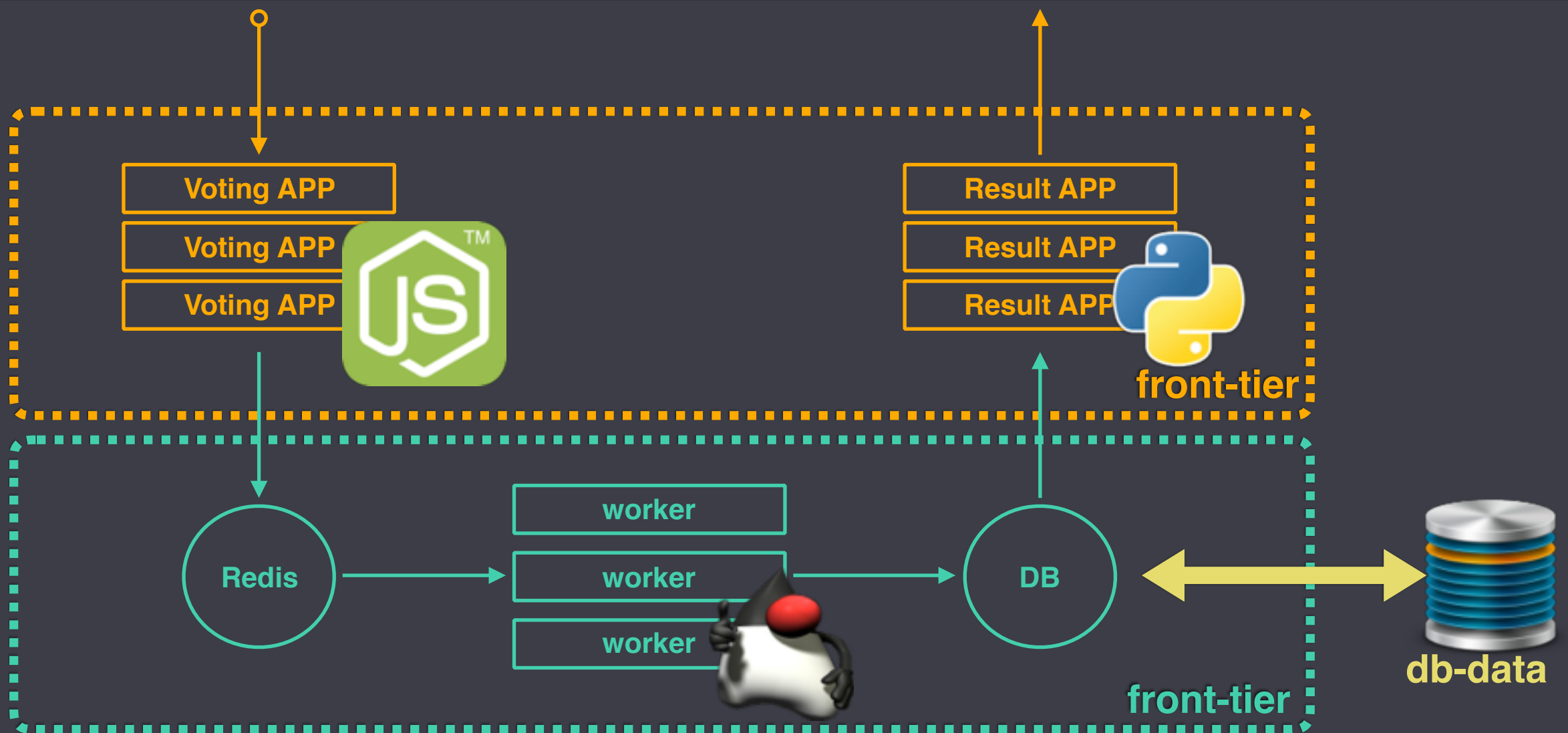
简易程度：

- 学习曲线
- 编排系统复杂度

容器编排与容器化应用系统

- Cloud Native Application
- 微服务架构
- 应用通信能力






```
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

root@ip-172-31-12-56:~/vote-app#

容器调度能力

资源分配与调度

- 资源利用率
- 用户需求限制
- 快速决策能力
- 多种公平原则

容器动态迁移

- 伸缩迁移
- 故障迁移

Node 1000
Container 100000
Latency <150ms

daocloud_admin

服务 编辑

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	实例状态
	i-72d49e4a	t2.micro	运行中
	i-d6d49eee	t2.micro	运行中
	i-0ad59f32	t2.micro	运行中
	i-e8d49ed0	t2.micro	运行中
	i-b2d79d8a	t2.micro	运行中
	i-85d59fbd	t2.micro	运行中
	i-8dd59fb5	t2.micro	运行中
	i-53eba16b	t2.micro	运行中
dongpu-poc-002	i-e8e9a1d0	t2.medium	运行中
dongpu-poc-003	i-e9e9a1d1	t2.medium	运行中
dongpu-poc-004	i-ebe9a1d3	t2.medium	运行中
dockerhub-sync-prod	i-0b5b2d33	t2.small	运行中

从以上选择一个实例

```
root@ip-172-31-12-56:~# docker info | grep Nodes
Nodes: 10
root@ip-172-31-12-56:~#
```

Docker/Swarm不但功能强大

容器调度能力

强大扩展能力背后

- 架构的精细
 - 系统复杂度低
 - workflow 组件数量精简
- 容器状态的缓存
 - 容器状态由Swarm Manager缓存
 - 减少数据的存储
 - 提高扩展速度
 - 最终一致性

DaoCloud Enterprise



一套容器集群解决方案

管理你的计算资源，包括物理机和公有云环境

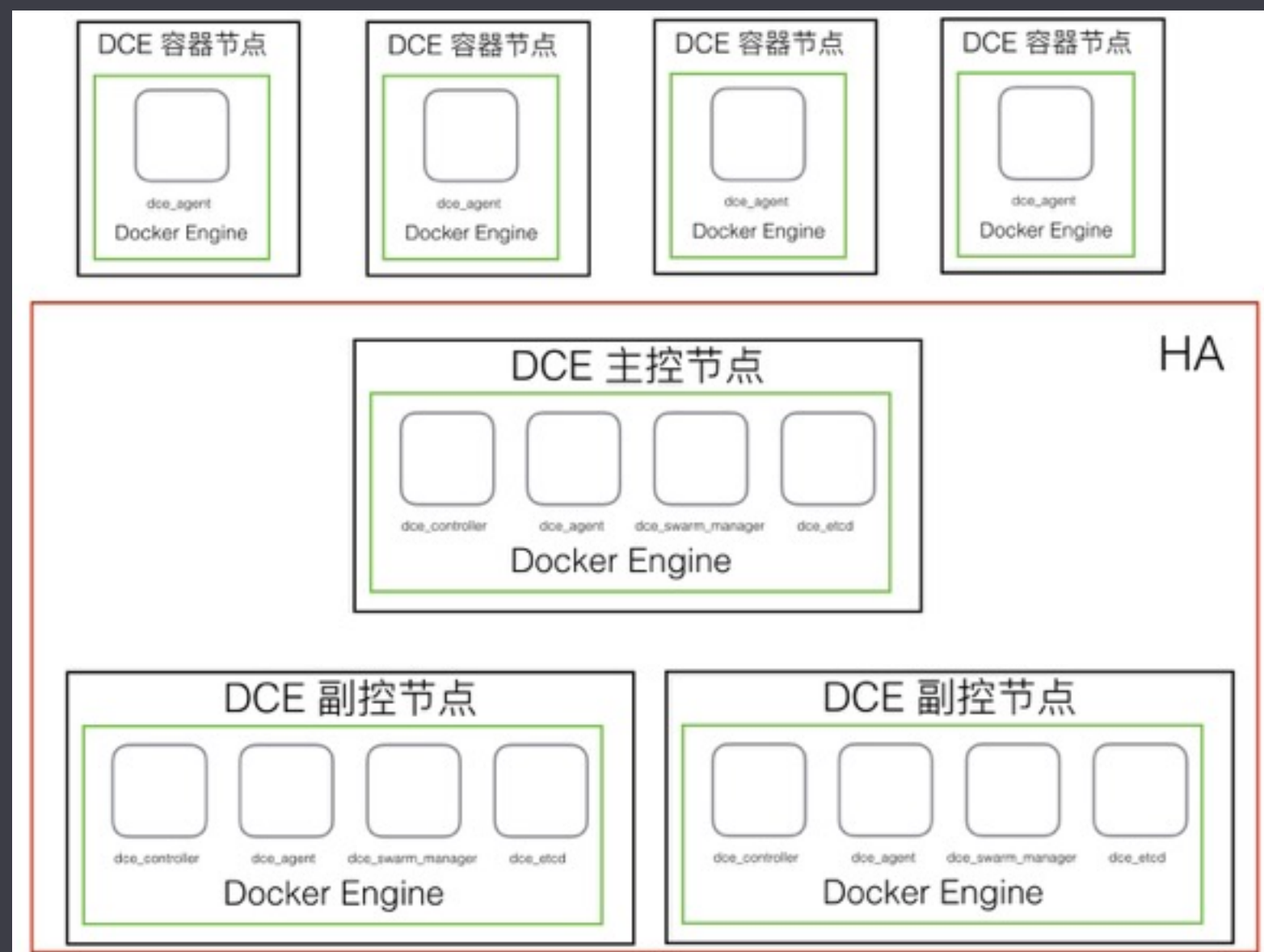
部署并管理所有的 Docker 应用

- 标准化
- 企业级
- 开放性

DaoCloud Enterprise 架构图

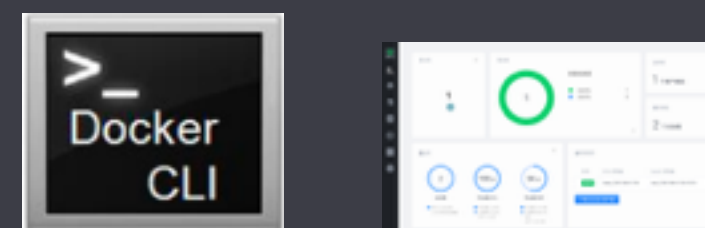


dce_agent	运行用户所有的容器和应用
dce_controller	调用 dce_swarm_manager 服务，向用户提供 WEB 控制台 UI，集群管理 API 和 CLI 运维套件服务
dce_swarm_manager	安装有 Swarm 服务的容器，负责容器节点的管理和自发现
dce_etcd	key-value 键值库，负责存储集群中的多种信息



系统高可用(HA)

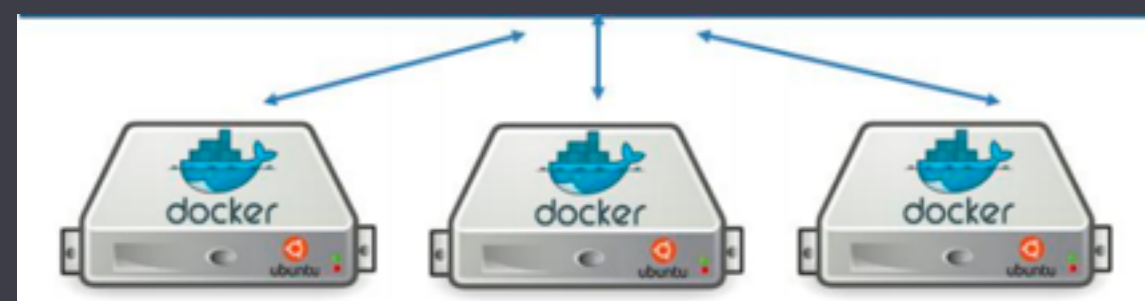
- 启动多控制节点，主备方式实现HA
- 分布式锁实现主节点选举
- 副节点实现请求转发
- 支持etcd、zookeeper等



主控节点

副控节点

副控节点



应用高可用(HA)

Docker Engine的状态设置

- 正常状态
- 下线状态
- 维护状态

应用容器高可用

- 应用多实例
- 自动负载均衡
- 节点故障迁移

Thank You!