DevOpsDays

Shanghai

— 2017.8.18-8.19 —

上海龙之梦酒店（长宁区延安西路1116号）

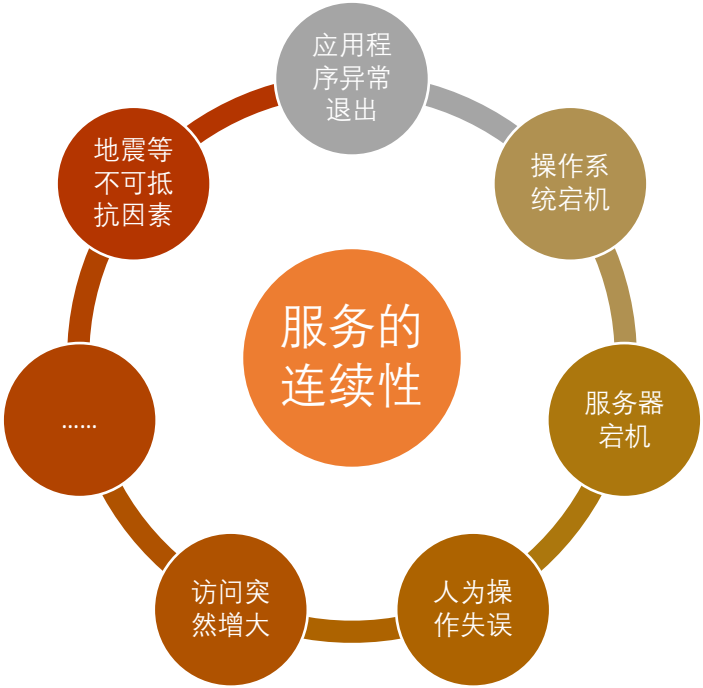
主办单位：高效运维社区 GreatOPS Community　Best Practice 最佳实践

基于DevOps、微服务以及k8s的高可用架构探索与实现

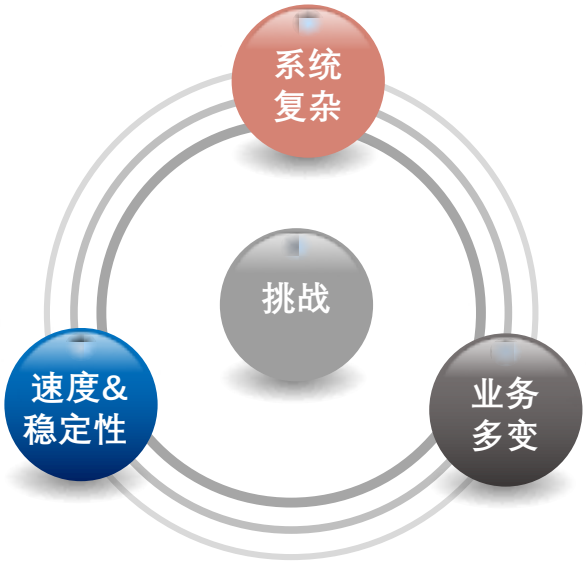
刘淼 HPE

目录

企业级高可用性架构的挑战



企业级高可用性架构的挑战



外部：多种OS/软硬件
内部：新旧系统并存

重构的困难

频度：对部署造成压力
扩展：按需伸缩的实际需求

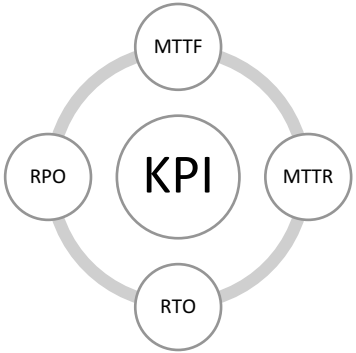
变更的挑战

速度：快速的变更对应需求
稳定：服务持续稳定性要求

速度和可靠性的压力

目录

高可用性架构目标



保证业务连续性&稳定性

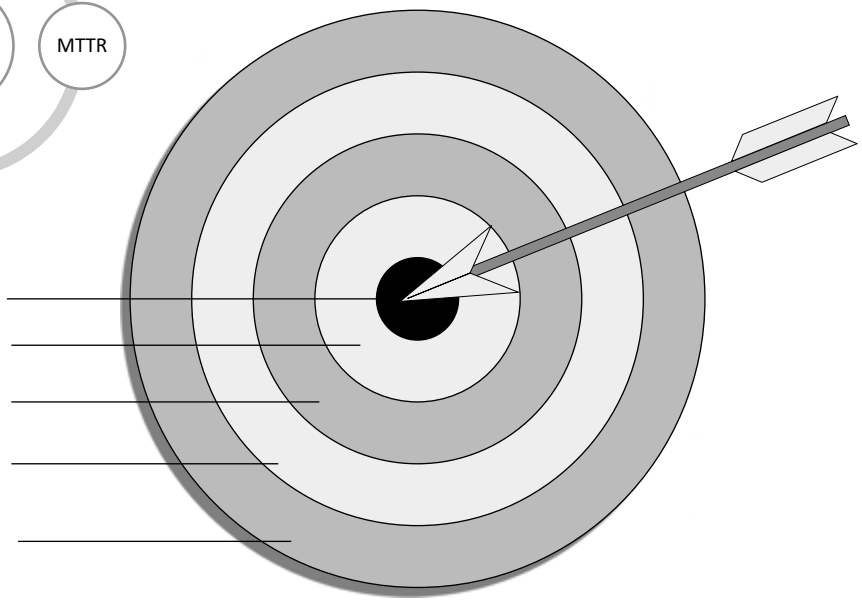
高可用指标

服务连续性

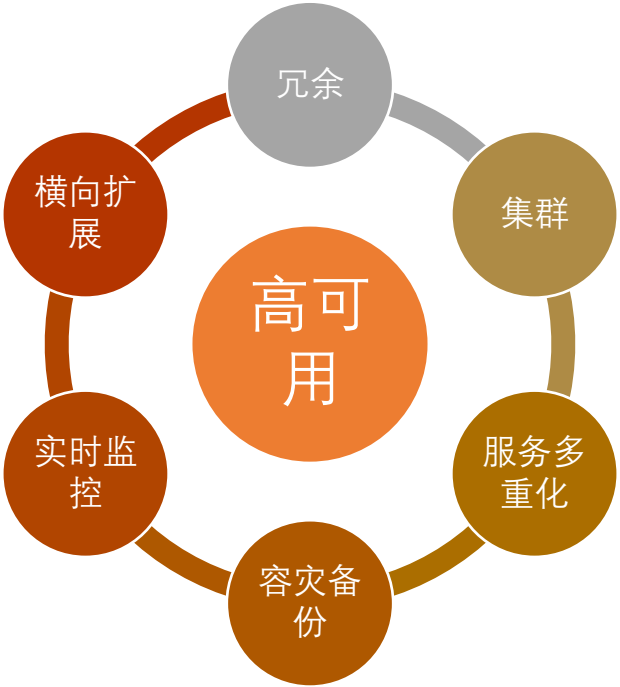
服务稳定性

...

良好扩展性



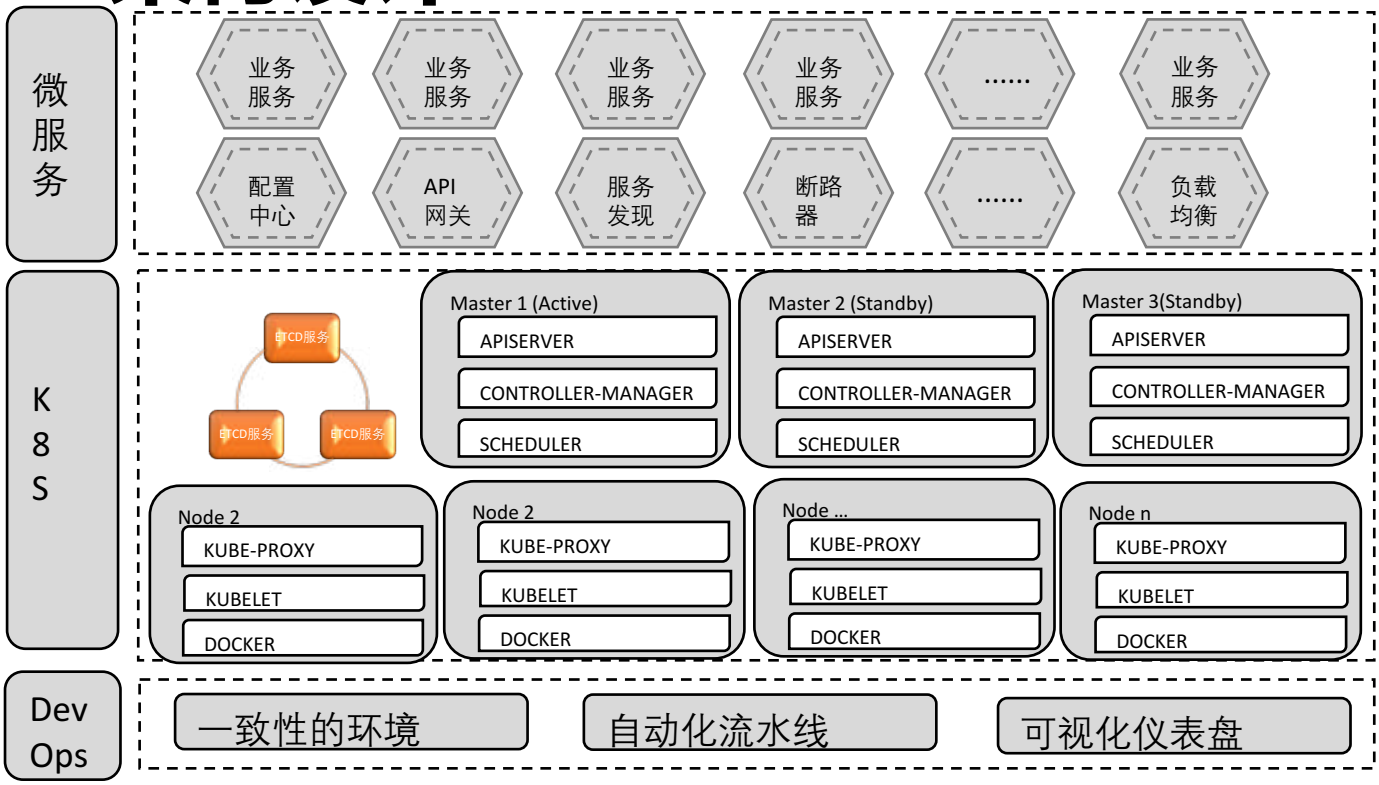
高可用策略&手段



要素&原则

服务的自愈机制	简化&解耦	环境一致性
消除ETCD的单点	独立部署	持续集成
消除MASTER单点	无状态	持续交付
可动态横向调整	可回滚	持续反馈
……	……	……

架构设计


微服务
业务服务
业务服务
业务服务
业务服务
.....
业务服务
配置中心
API网关
服务发现
断路器
.....
负载均衡
K8S
etcd服务
etcd服务
etcd服务
Master 1 (Active)
APISERVER
CONTROLLER-MANAGER
SCHEDULER
Master 2 (Standby)
APISERVER
CONTROLLER-MANAGER
SCHEDULER
Master 3(Standby)
APISERVER
CONTROLLER-MANAGER
SCHEDULER
Node 2
KUBE-PROXY
KUBELET
DOCKER
Node 2
KUBE-PROXY
KUBELET
DOCKER
Node ...
KUBE-PROXY
KUBELET
DOCKER
Node n
KUBE-PROXY
KUBELET
DOCKER
Dev Ops
一致性的环境
自动化流水线
可视化仪表盘

目录

高可用的K8S

运行服务的高可用性

1

•由K8S的RC或者DaemonSet等保证服务的自愈可用

K8S自身的高可用性

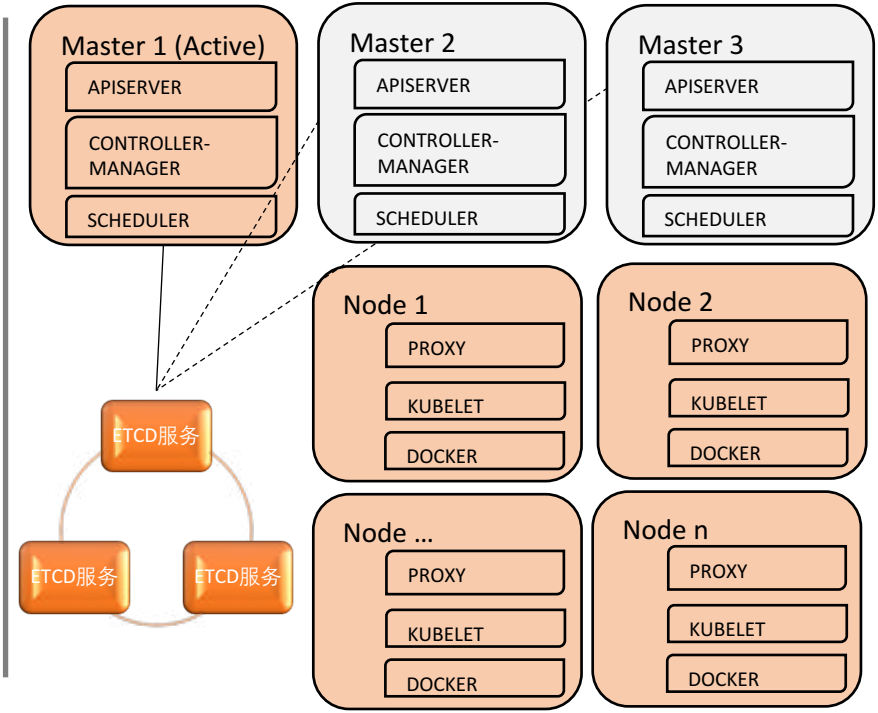
2

•冗余策略消除单点，保障ETCD和Master始终可用

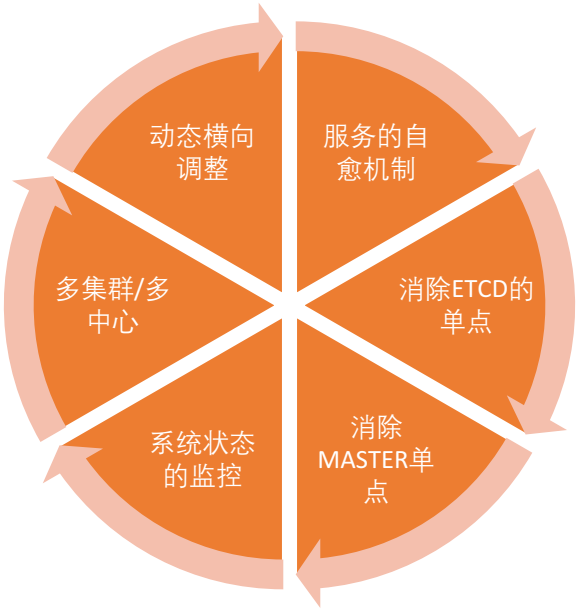
消除单点的ETCD

集群：类似**ZK**，使用奇数个数节点
存储：使用**RAID**等保证数据的稳定可靠性

消除单点的Master

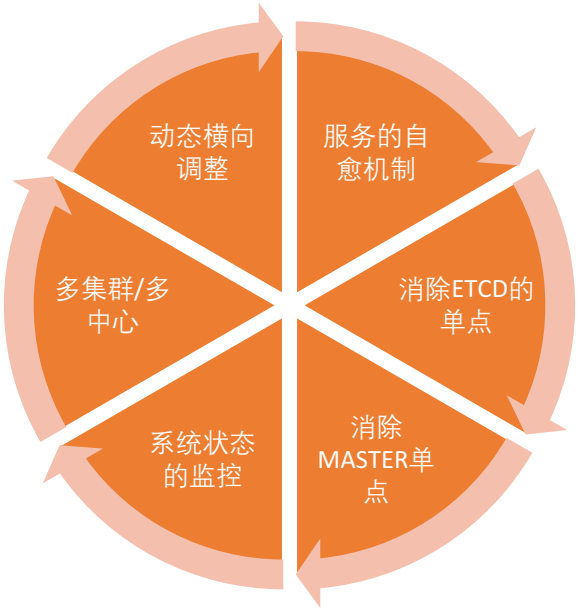


高可用的K8S：监控



通过Zabbix等监视系统相关的系统状况，结合业务日志，统一使用Grafana进行业务和系统状况的监控

高可用的K8S：动态横扩



定义指标及预定操作

1. 定义系统资源/业务等指标

采集数据

2. 采集系统资源/业务实时数据

监视业务状态

3. 确认是否达到自动触发的阈值

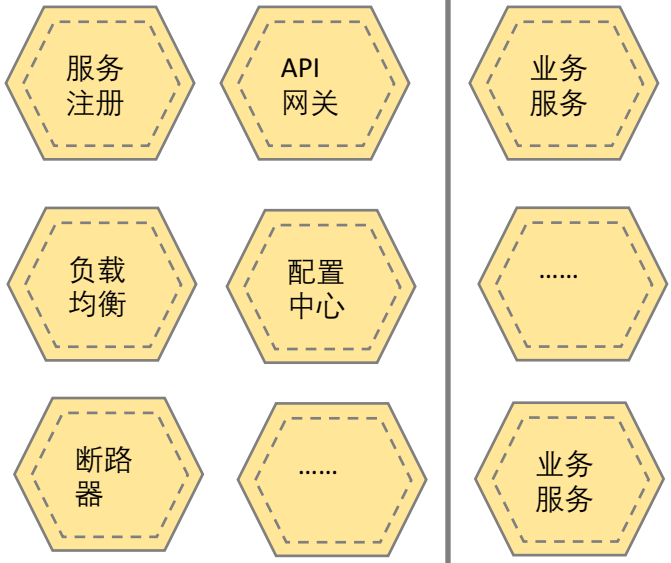
定义指标

4. 自动进行scale等预定操作

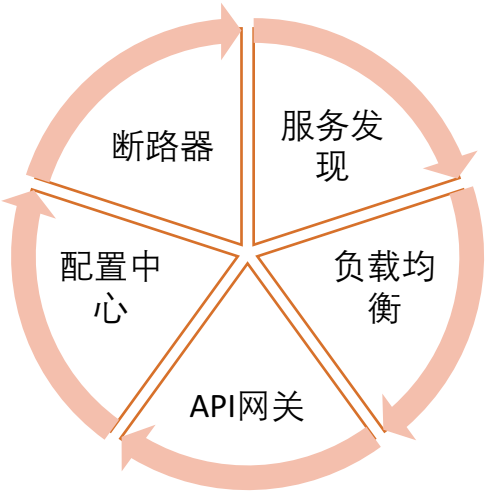
目录

专注于业务实现的微服务架构

1. •部署独立化
2. •API规范化
3. •接口轻量化
4. •功能解耦&简化
5. •规模小型化
6. •服务可置换
7. •去中心化…



微服务：服务注册



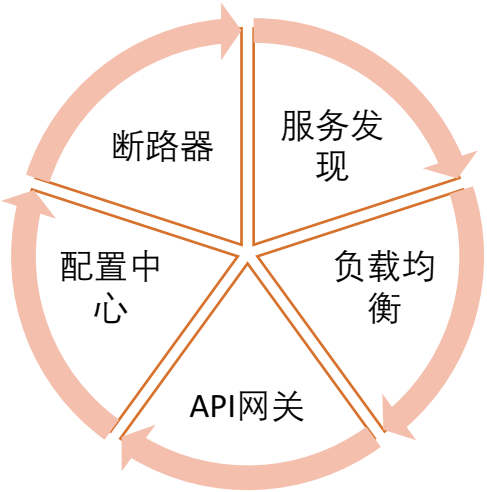
服务发现：

微服务在设计上将业务功能解耦拆解成了可独立部署小型单元，但是随着服务的增多，没有服务发现的机制将会困难重重。Spring Cloud中的Eureka组件则提供了这样的一个功能，一行注解便可开箱即用。

策略：

高可用性：适当冗余，使用多个Eureka服务端保证服务注册不会因为服务注册中心的Eureka停下而无法使用。

微服务：负载均衡



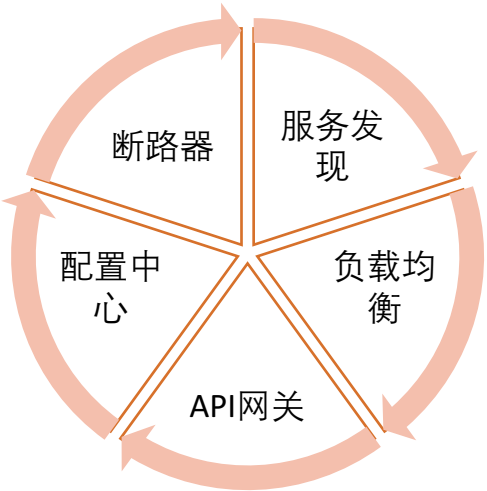
负载均衡：

使用微服务的方式进行封装的无状态的小型单元，可以非常方便地使用Kubernetes进行横向扩展，而对横向扩展之后的服务进行访问，保证其负载的均衡可以使用Ribbon组件。

策略：

- 简单轮询负
- 加权响应时间
- 最小并发请求
- 随机

微服务：API网关



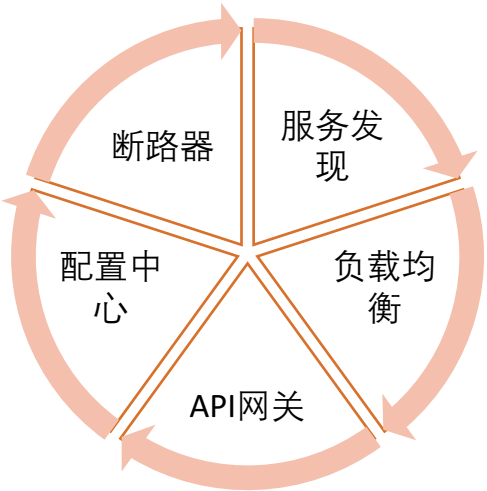
API网关：

Zuul可以为提供身份验证/动态路由/安全控制等诸多功能，使用Zuul作为统一的Api网关出口进行管控，便于集成和管理。

策略：

- 使用统一API网关出口
- 使用OPEN API标准格式定义API
- 使用Swagger进行管理

微服务：配置中心



配置中心：

为了保证一致性的开发/测试/生产环境，便于统一管理和修改，对整体的配置管理引入了Spring Cloud Config。

策略：

- 统一配置管理
- 统一依赖管理
- 结合DevOps实践方式拆分环境
- 按环境独立部署

目录

DevOps：环境一致性

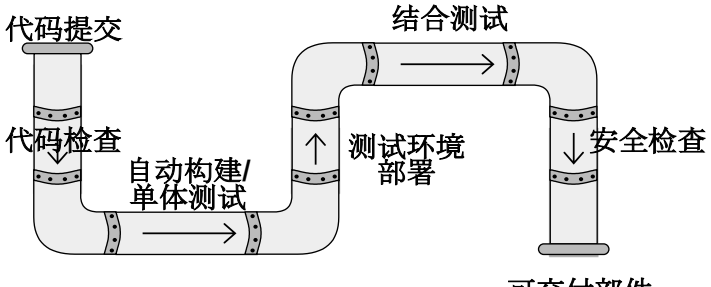
开发环境一致性：保证一致性的开发环境，确保所有成员在一致性的环境中进行开发，避免因各种版本不合导致的Rework，提前引入安全扫描机制。

测试环境一致性：一致性的测试环境避免因环境的问题出现的非缺陷性确认和对应所需时间以及缺陷的延后出现的可能性，提前引入安全扫描机制。

生产环境一致性：确保准生产环境能够得到尽可能类似生产环境的测试要素，同时避免因软硬件变动导致的各种问题。

DevOps:可定制流水线

可定制的流水线提高CI/CD的效率和稳定性



DevOps:可视化

通过对软件开发全生命周期的KPI进行管理和可视化管控。从构建到测试，从开发到部署，从构建频度到成功率，更加有效的掌握整体情况，在实际的DevOps落地实践中通过有效的可视化打通那堵看不见的”墙”。

目录

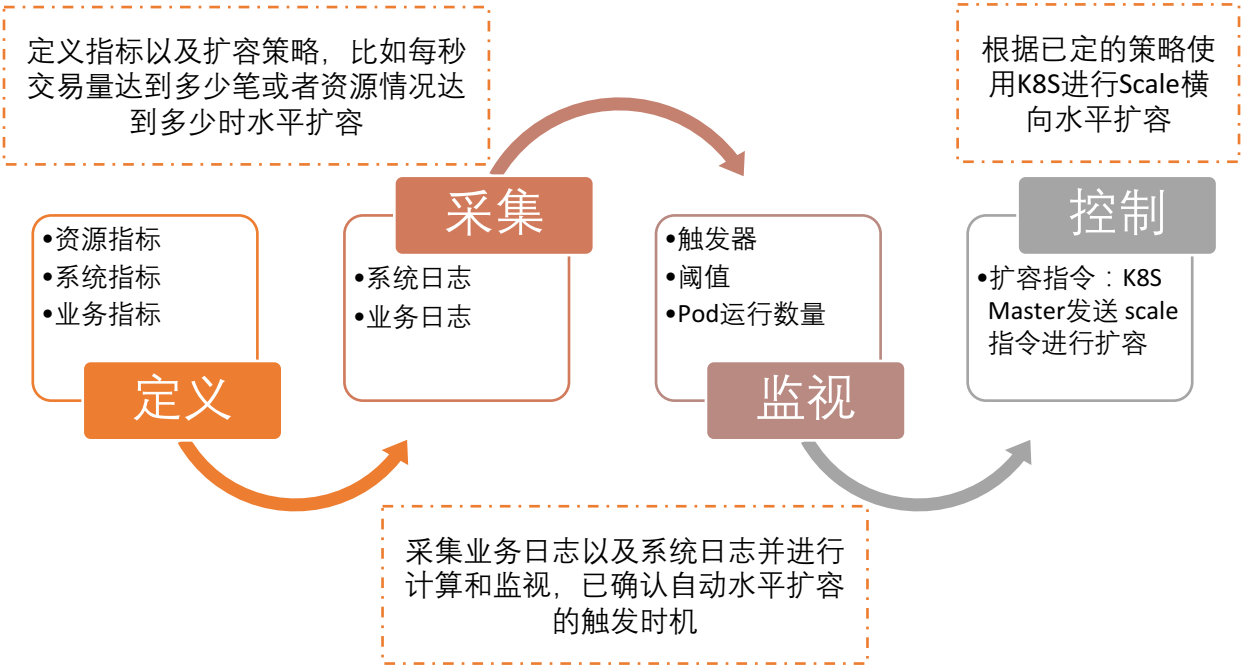
弹性扩容：策略

在容器化的基础之上，对微服务进行优化解耦，尽量除去或者减少对横向扩容产生不利影响的要素比如有状态的服务设计等。

强化业务和系统资源实时监控功能，以确保问题发生之前有可能的途径事先做出部分判断。
保证运维操作的可回滚性，以保障问题发生之后可以迅速恢复服务的提供。

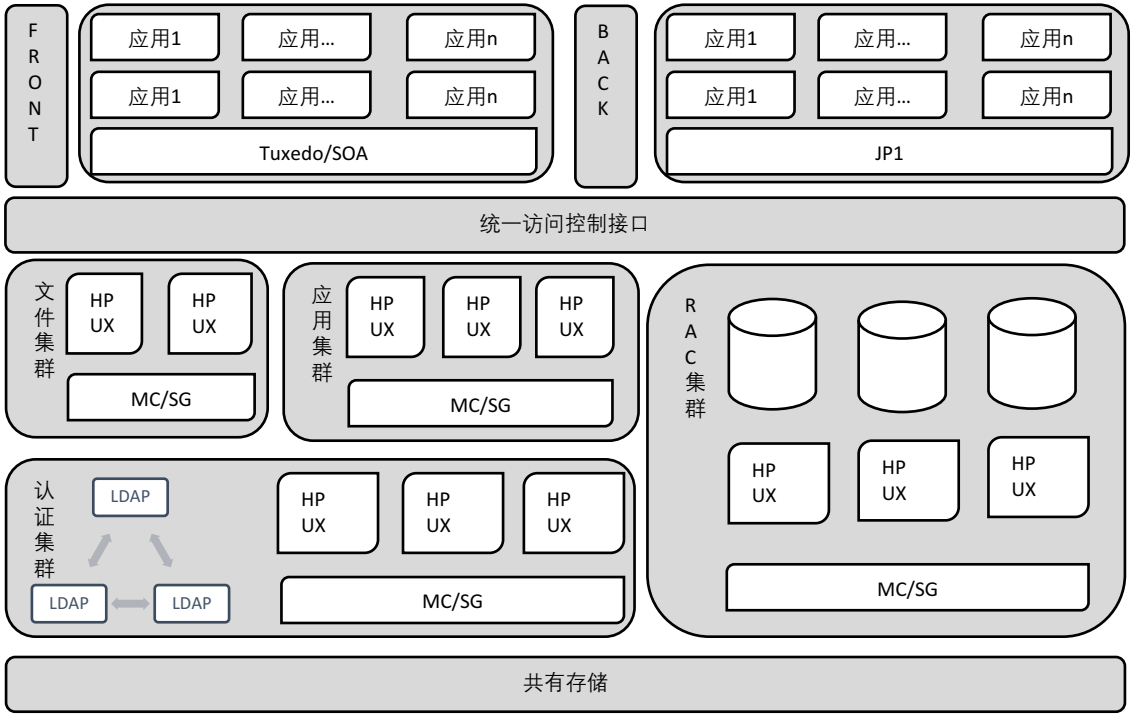
K8S对无状态服务可以进行非常容易地横向扩展，这建立在对现有状态的监控机制健全的前提和基础之上，通过设定的扩容策略，在监控达到触发条件时，进行动态弹性扩容。

弹性扩容：动态实施


定义指标以及扩容策略，比如每秒交易量达到多少笔或者资源情况达到多少时水平扩容
•资源指标
•系统指标
•业务指标
定义
采集
•系统日志
•业务日志
采集业务日志以及系统日志并进行计算和监视，已确认自动水平扩容的触发时机
•触发器
•阈值
•Pod运行数量
监视
根据已定的策略使用K8S进行Scale横向水平扩容
控制
•扩容指令：K8S Master发送 scale 指令进行扩容

目录

实践案例分享


FRONT
应用1
应用...
应用n
应用1
应用...
应用n
Tuxedo/SOA
BACK
应用1
应用...
应用n
应用1
应用...
应用n
JP1
统一访问控制接口
文件集群
HP UX
HP UX
MC/SG
应用集群
HP UX
HP UX
HP UX
MC/SG
RAC集群
HP UX
HP UX
HP UX
MC/SG
认证集群
LDAP
LDAP
LDAP
HP UX
HP UX
HP UX
MC/SG
共有存储

实践总结

应用级可用性

- 服务多重化
- Dummy机制实现热部

节点级可用性

- FailOver/Failback

数据中心级可用性

- 定时数据同步机制
- 少量实时数据同步

资源协调

- 跨级群资源协调困难

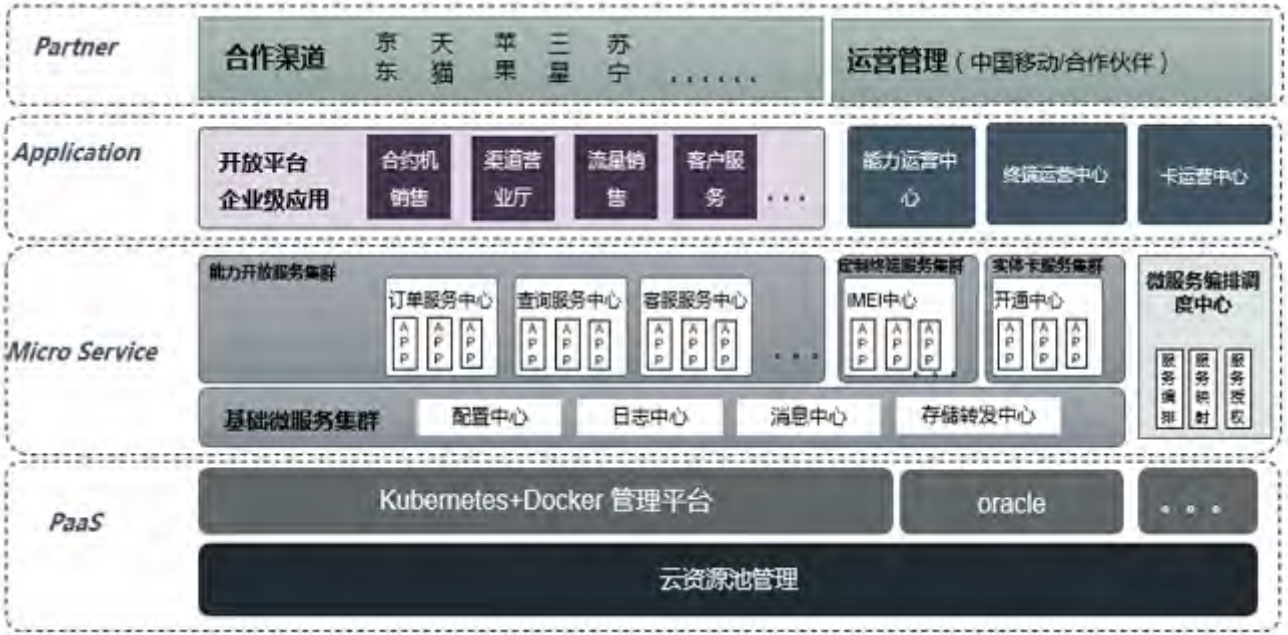
横向扩展

- 扩容困难

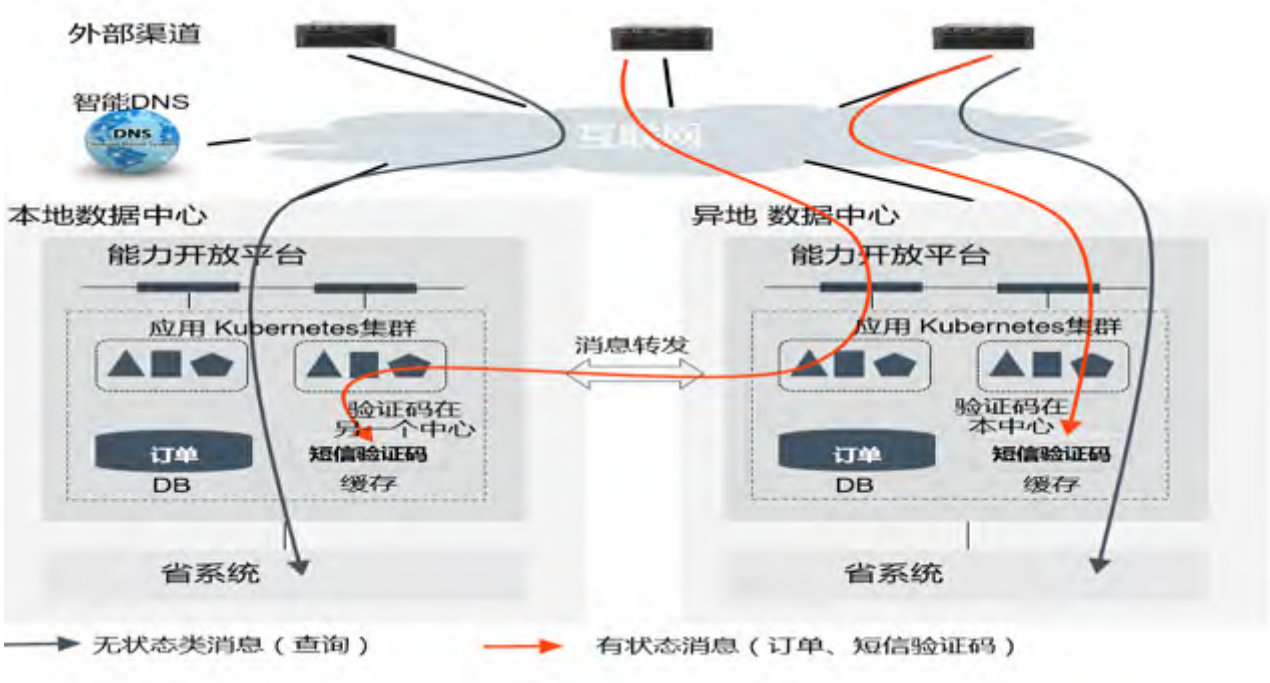
数据不一致

- 物理/逻辑节点混淆
- FailOver&Failback的影响

实践案例分享


Partner
合作渠道
京东
天猫
苹果
三星
苏宁
.....
运营管理（中国移动/合作伙伴）
Application
开放平台
企业级应用
合约机销售
渠道营业厅
流量销售
客户服务
...
能力运营中心
终端运营中心
卡运营中心
Micro Service
能力开放服务集群
订单服务中心
查询服务中心
客服服务中心
...
定制终端服务集群
IMEI中心
实体卡服务集群
开通中心
微服务编排调度中心
服务编排
服务映射
服务授权
基础微服务集群
配置中心
日志中心
消息中心
存储转发中心
PaaS
Kubernetes+Docker 管理平台
oracle
...
云资源池管理

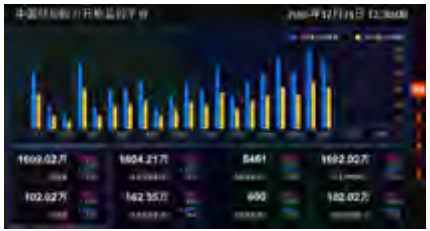
能力开放平台双中心双活

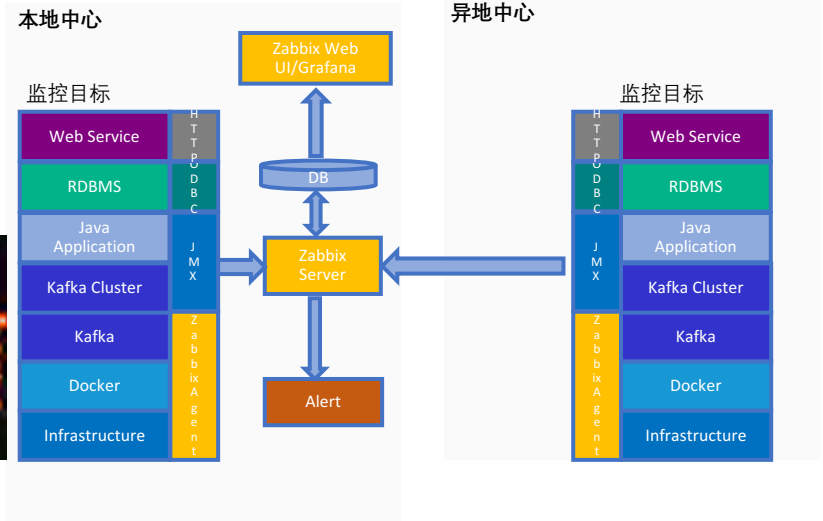


双中心业务一体化监控

监控内容：业务监控、系统监控、安全监控，侧重于业务监控。

监控方案：业务监控为主，实时收集双中心监控数据，并展现和告警。同时安全监控主要使用漏洞检测工具自检、不定期软件补丁升级、定期安全扫描。





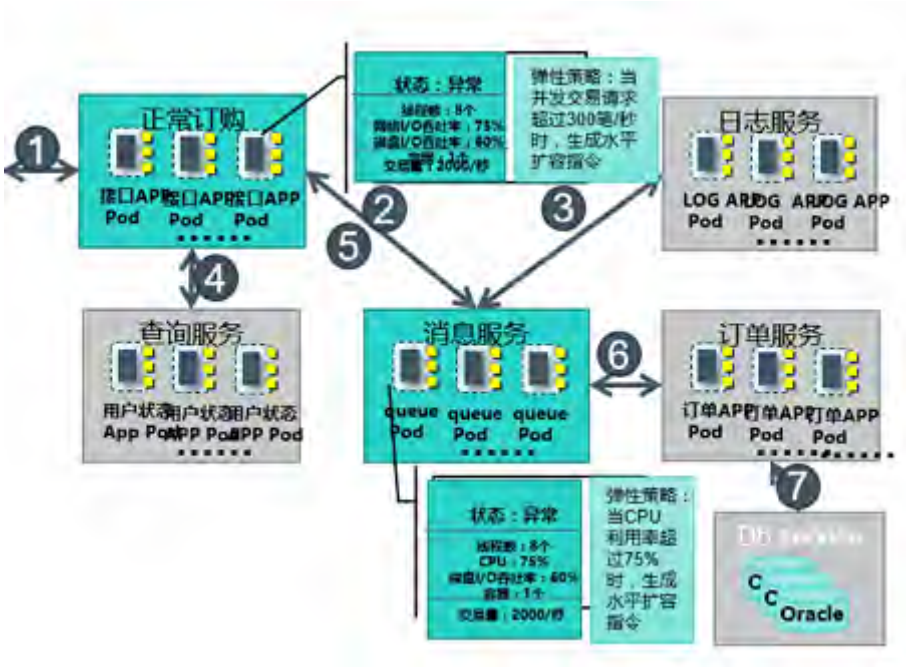
动态弹性扩容自适应业务需求

天猫双十一的订单量相当于全年其他日的订单总合；

京东618订单量相当于平日**10**倍；

苹果新机型首发当日订单也会比平日增长**20**倍以上

517电信日，订单量增加**126**倍



实践总结

应用级可用性

- 服务多重化
- RollingUpdate

节点级可用性

- 节点存在透明化

数据中心级可用性

- 双中心双活

资源协调

- 资源统一协调控制

横向扩展

- 按需动态横向扩展

一体化监控

- 系统/业务一体化监控
- 系统安全定期监控

高效运维社区
GreatOPS Community

会议

- 8月18日 DevOpsDays 上海
- 全年 DevOps China 巡回沙龙
- 11月17日 DevOps金融上海

培训

- EXIN DevOps Master 认证培训
- DevOps 企业内训
- DevOps 公开课
- 互联网运维培训

咨询

- 企业DevOps 实践咨询
- 企业运维咨询

商务经理：刘静女士
电话 / 微信：13021082989
邮箱：liujing@greatops.com

Thanks

荣誉出品

高效运维社区

国际最佳实践管理联盟