

金融级云原生 PaaS 探索与实践

王成昌 (晙曦) 蚂蚁金服技术专家





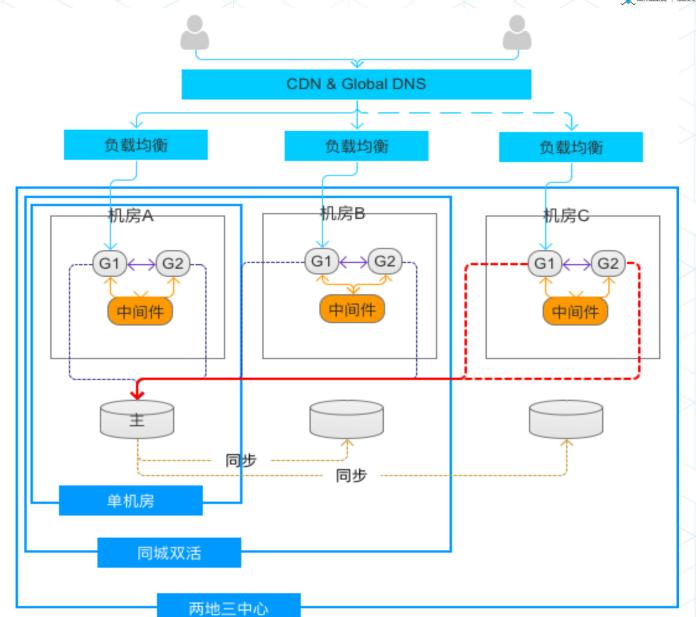




业务架构

演进

- 容量
 - ▶ 应用|数据库|机房
- 容灾
 - ▶ 机房|地域



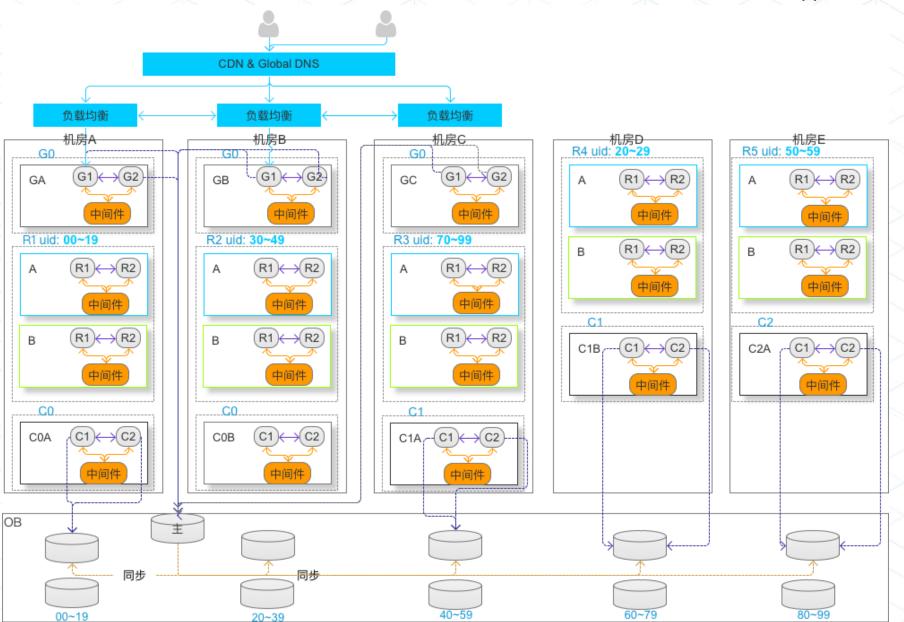




业务架构

单元化

- 高可用
- 一致性
- 可扩展
- 高性能







业务诉求

- 运维成本
 - > 突发流量应用 | 机房 生命周期
- 运维效率
 - > 大规模下基础设施稳定性
- 业务 Mesh 化
 - > 精细化流量控制
 - > 基础组件升级
- 业务可复制
 - > 业务敏捷
 - > SaaS 面向站点级别输出





PaaS 能力

- 面向多租户多环境;
- 基础资源管控;
- 应用发布运维体系;
- 业务实时监控,日志收集;
- 机房级和地域级容灾能力;





云原生 PaaS 产品架构方案

两地三中心架构 业务架构 同城双活架构 异地多活架构 跨机房和地域统一应用运维 分钟级容灾切换和恢复 全面变更风险管理 产品层 无限弹性可扩展 PaaS 核心层 核 中间件变配 蓝绿发布 Mesh流调拨和治理 灰度分组发布 心 (DRM/Scheduler/Message) 流 容灾切换和恢复 应急预案管理 弹性建站/下站 容器腾挪/迁移 基础发布运维 单元化能力 跨集群管理 跨集群状态 分发/汇聚 跨集群应用 批次发布 原生资源管理 变更管控 接入层流程调拨 应用层流量调拨 资源管理 压测/灰度 集群伸缩管理 Pod伸缩管理 配额管理 数据层流量调拨 跨集群发布策略 跨集群网络 流量管理 单元化 容器镜像管理 运维原子操作 精细化调度 弹性流量管理 多集群管理 跨集群镜像管理

容器层

K8S API Server

CAFÉ API Server Aggregation Layer

容器运行时 (Docker/Pouch/安全容器) CNI Plugins (VLAN/VXLAN/VPC Router/ENI)

CSI Plugins (NAS/OSS/Cloud Disk/Ceph)

网络接入 (SLB/ALB)

IaaS层(Aliyun/OpenStack/VMWare/Bare Metal)







为什么要有集群联邦

- 异构屏蔽:
 - > 底层集群变化;
- 统一管控:
 - > 业务弹性建站管控统一;
- 可扩展:
 - > 多租硬隔离;
 - ➤ 体量(单集群内节点数 1w+, Pod 10w+),集群数量多;





联邦核心能力

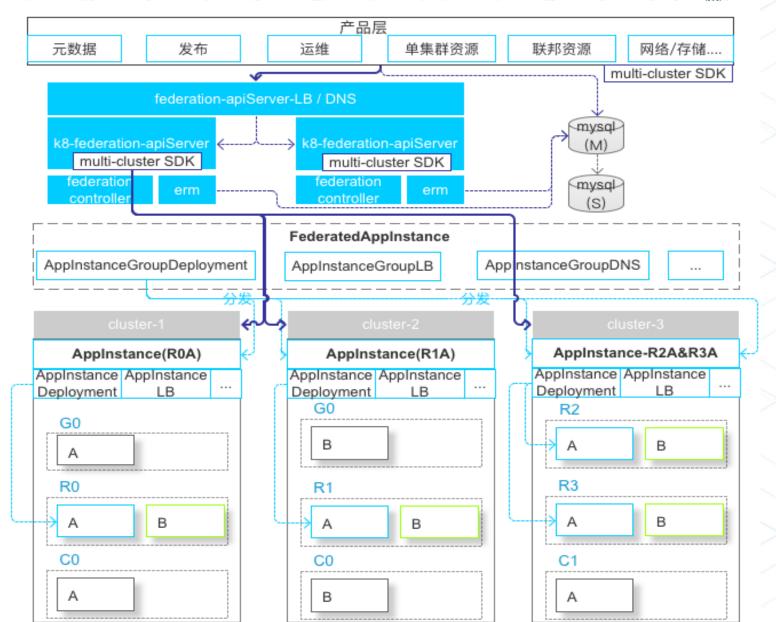
- 跨集群资源同步
 - ➤ Template,Override,Placement 模型;
 - > 状态回流;
 - ➤ 扩展 CRD;
- 跨集群发现





联邦架构

- 关系型存储;
 - > 数据量
 - > 容灾
- 基于部署单元分发









应用管理&交付

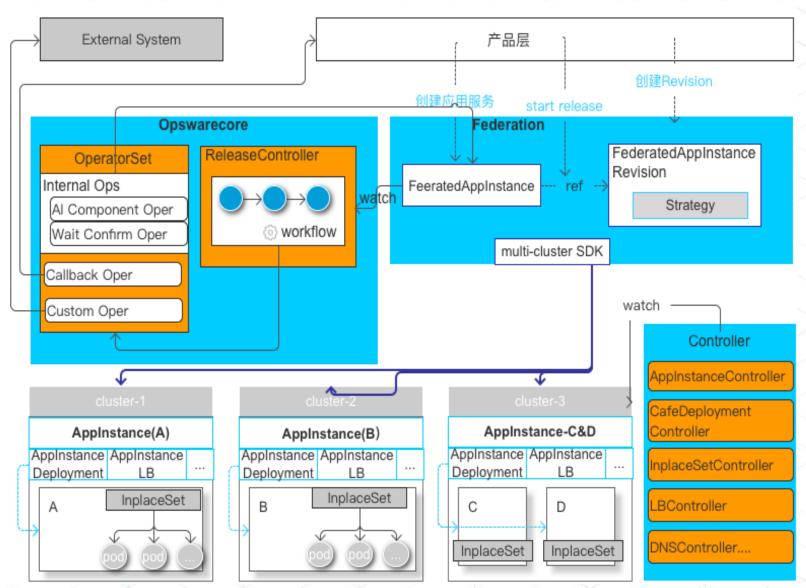
- 基于统一管控背景下的 Dockerfile 管理和生成;
- 基于组件关联的 FedAppInstance + revision 版本控制;
- 快速构建能力 binary2Image 能力;





发布运维

- 基础运维能力下沉;
 - ▶ 原地升级, 分组驱动;
 - > 流量控制;
- 多应用有序发布;
- 应用 ReleasePipeline 管理
 - > 应用依赖项顺序;
 - > 发布顺序;
 - > Beta 发布
 - > 分组发布;
- 变更管控能力;







发布流程

无损发布流程控制;

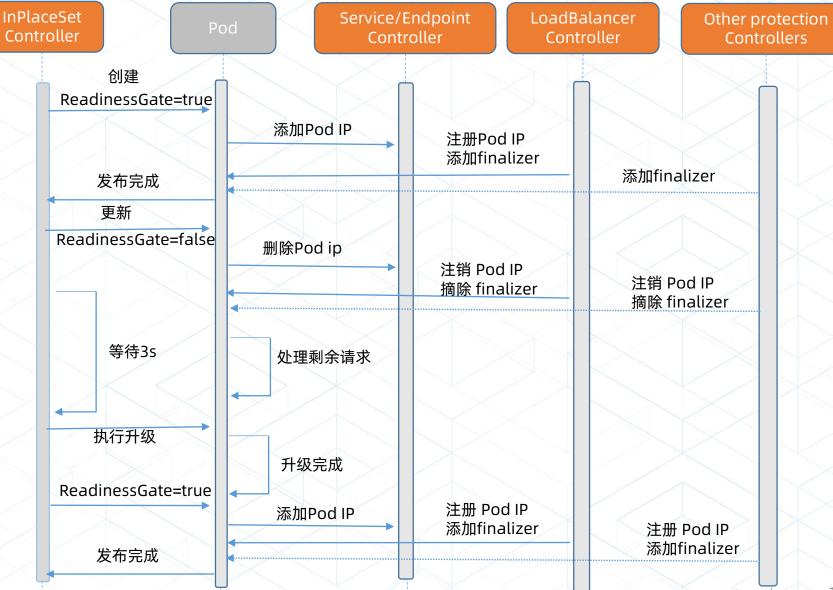
内部流量:

RPC

外部流量:

SLB (ALB)

DNS







安全风险保障

- 审计追踪;
- 用户安全 基于 RBAC 体系和 PaaS 账号体系打通;
- 租户安全 租户隔离|环境隔离|集群隔离;
- 容器运行时 配额|隔离控制(磁盘, CPUSET);





技术风险保障

业务变更三板斧

可灰度:

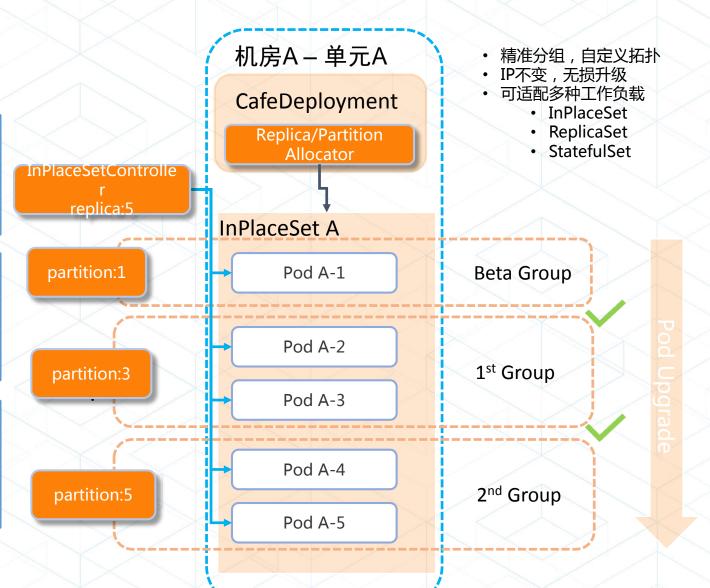
应用发布进程可控,允许灰度、 分组和 Beta 验证;

可回滚:

• 随时暂停、回滚,任何变更有据可查;

可监控:

• 接入监控告警体系,全程保证可观测性







技术风险管控

Operator变更三板斧

可灰度:

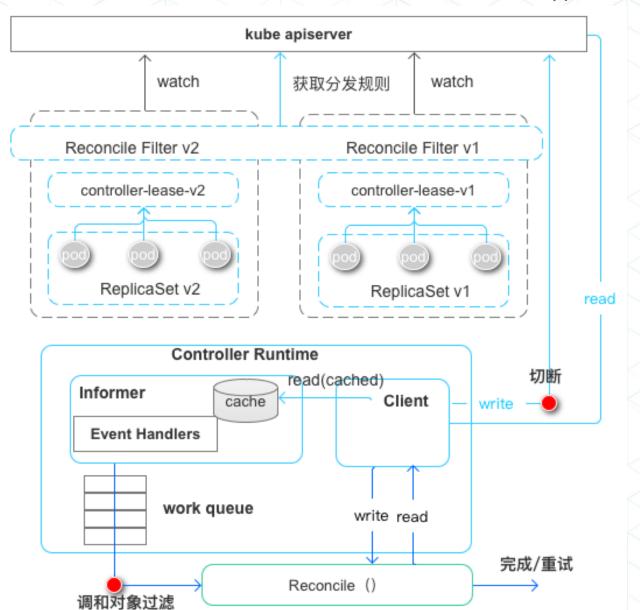
Controller 发布避免 0-1;

可回滚:

- 回滚到基线版本;
- 快速止血, 停止新版本调和;

可监控:

• metrics 指标(任务队列|消费&重试情况|吞吐量&错误);





欢迎关注 SOFAStack 公众号 获取分布式架构干货



使用钉钉扫码入群 第一时间获取活动信息

