

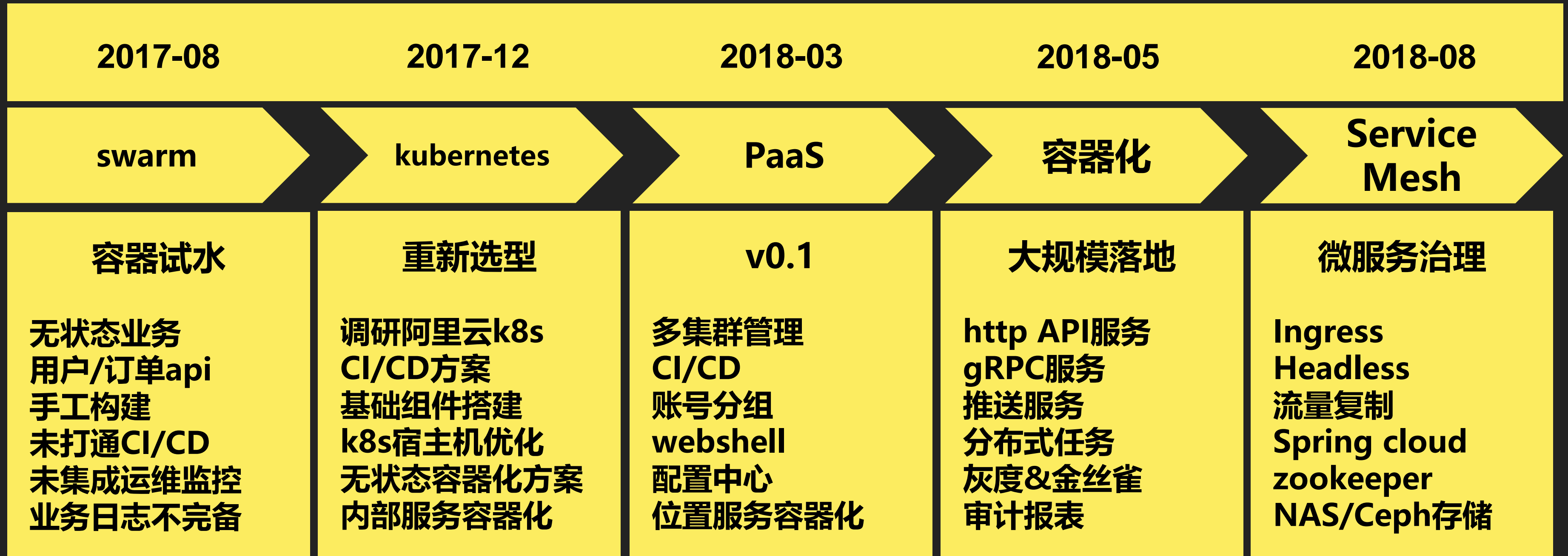
ofo业务容器化落地实践

系统技术部 王强



- 容器化历程
- 容器化现状
- PaaS容器平台
- 容器化业务架构
- 优化实践

容器化历程



容器化现状

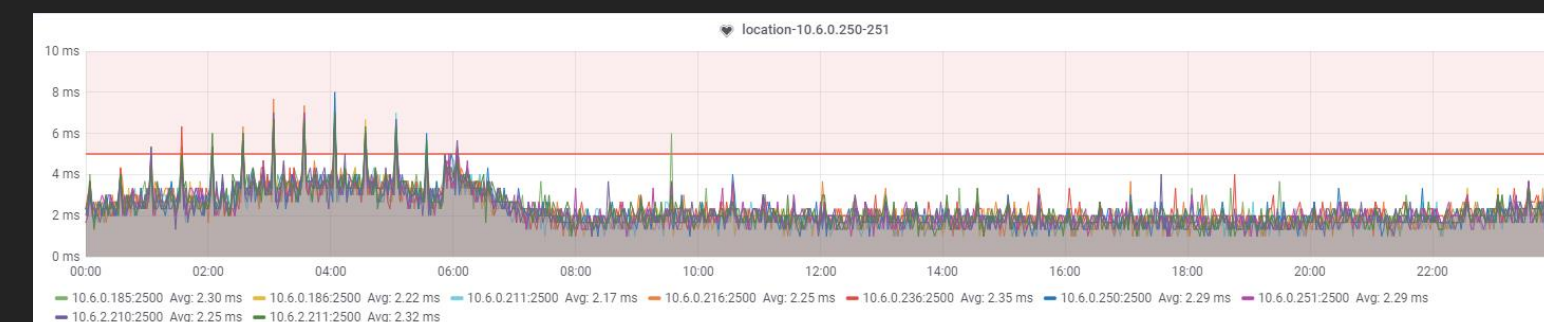
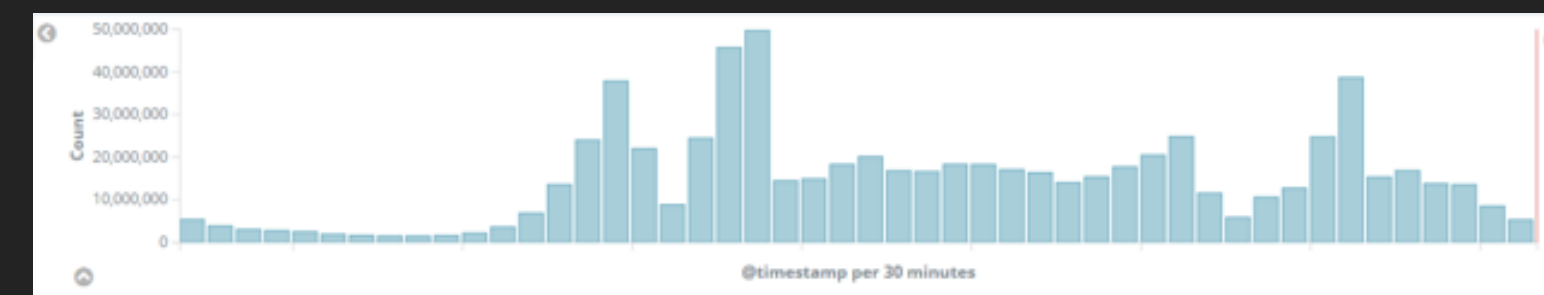
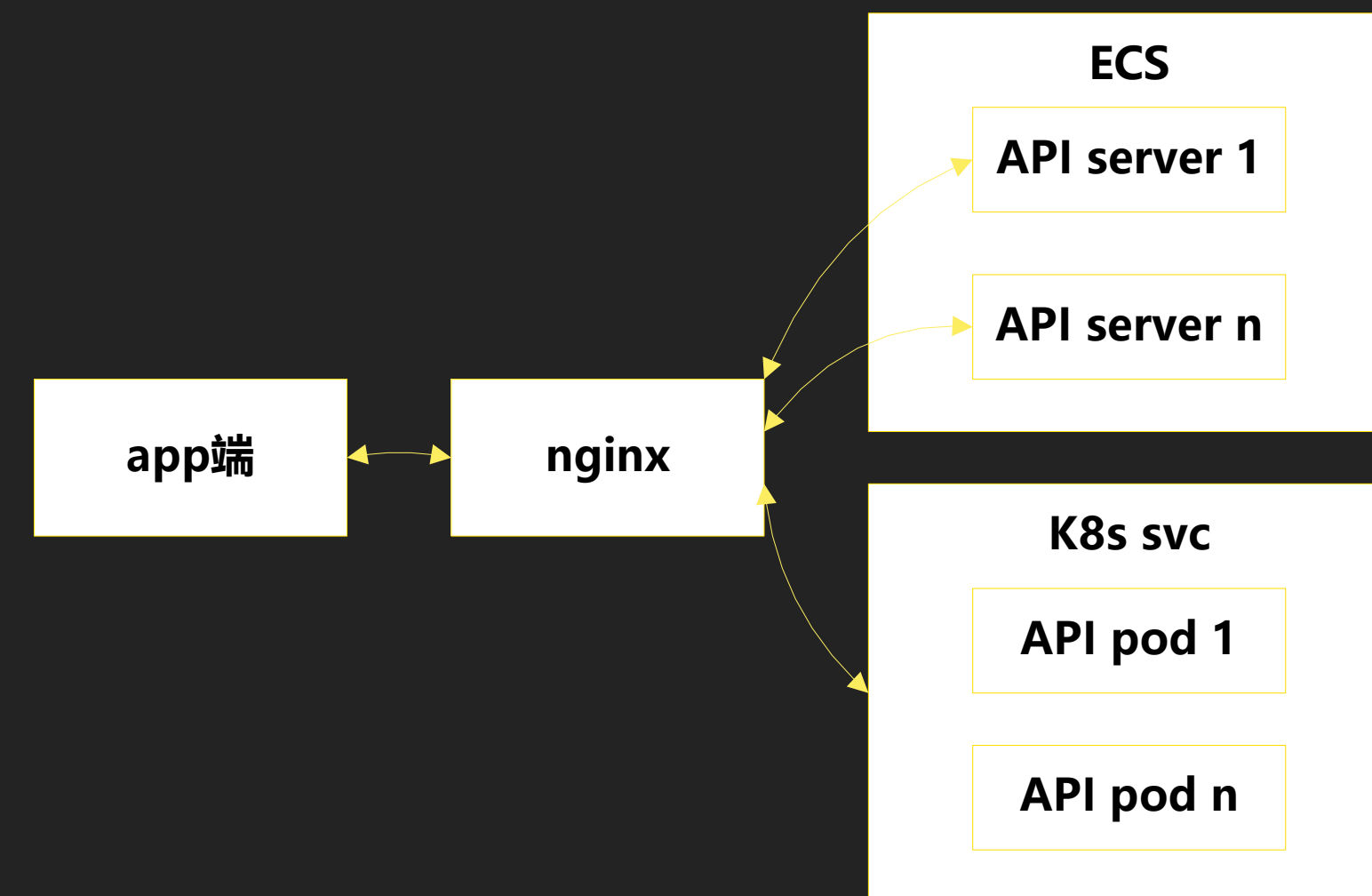
- **4个集群**：国内及海外测试/生产k8s集群
- **300+nodes**：32核64G，16核32G，8核16G虚机，按应用类型对节点分组
- **1500+ pods**：4核8G容器为主，**5%**有状态业务
- **80/50**：容器化覆盖**80%** c端业务，承接**50%+**线上流量
- **30/10**：资源利用率提升**30%**以上；性能损失不超**10%**，个别优于原部署方式
- 新业务全量容器化：推送服务（**百万**并发），iot长连接锁网关，iot数据分析
- 多种语言类型业务：**go, nodejs, java, python, php, c/c++**

PaaS容器平台



容器化业务架构-C端

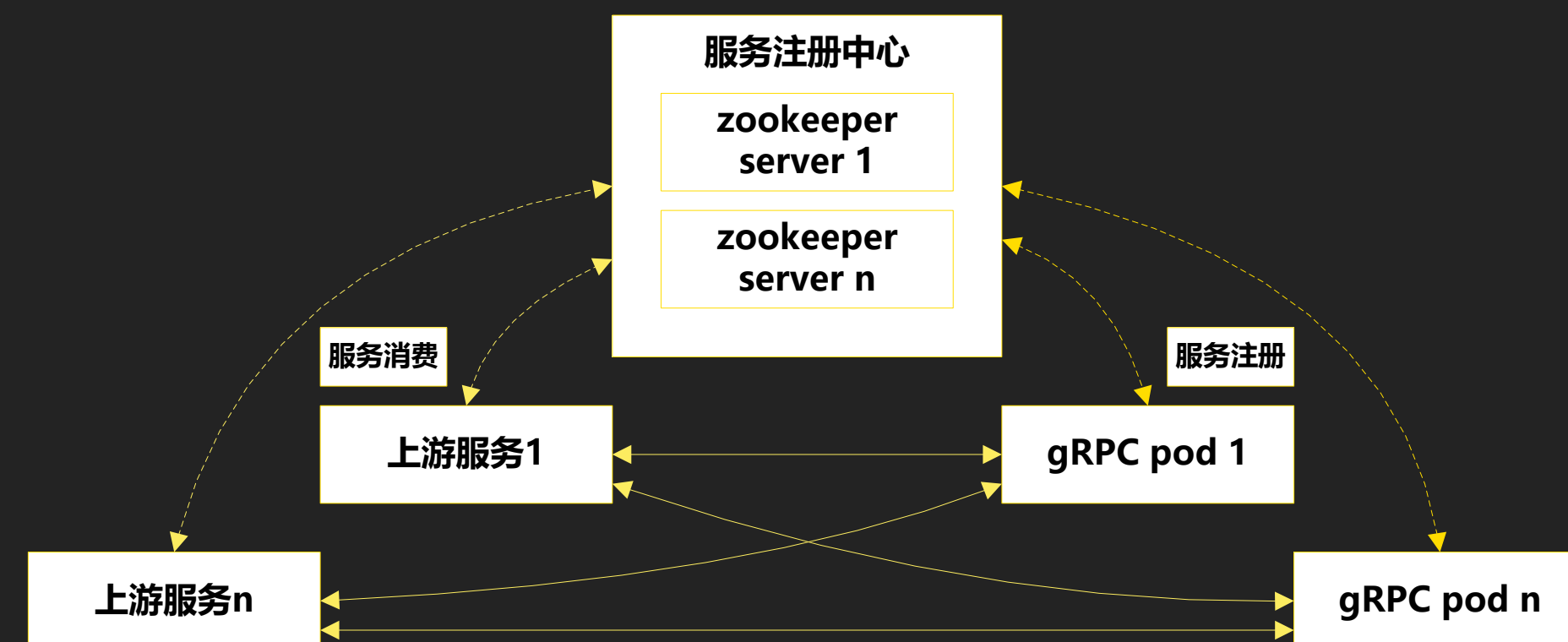
- REST API
- nodejs, go
- 无状态业务 (k8s deploy)
- K8s svc ip:port作为nginx upstream
- 业务举例：位置服务 (QPS 3万, RT 2.5ms)





容器化业务架构-gRPC服务

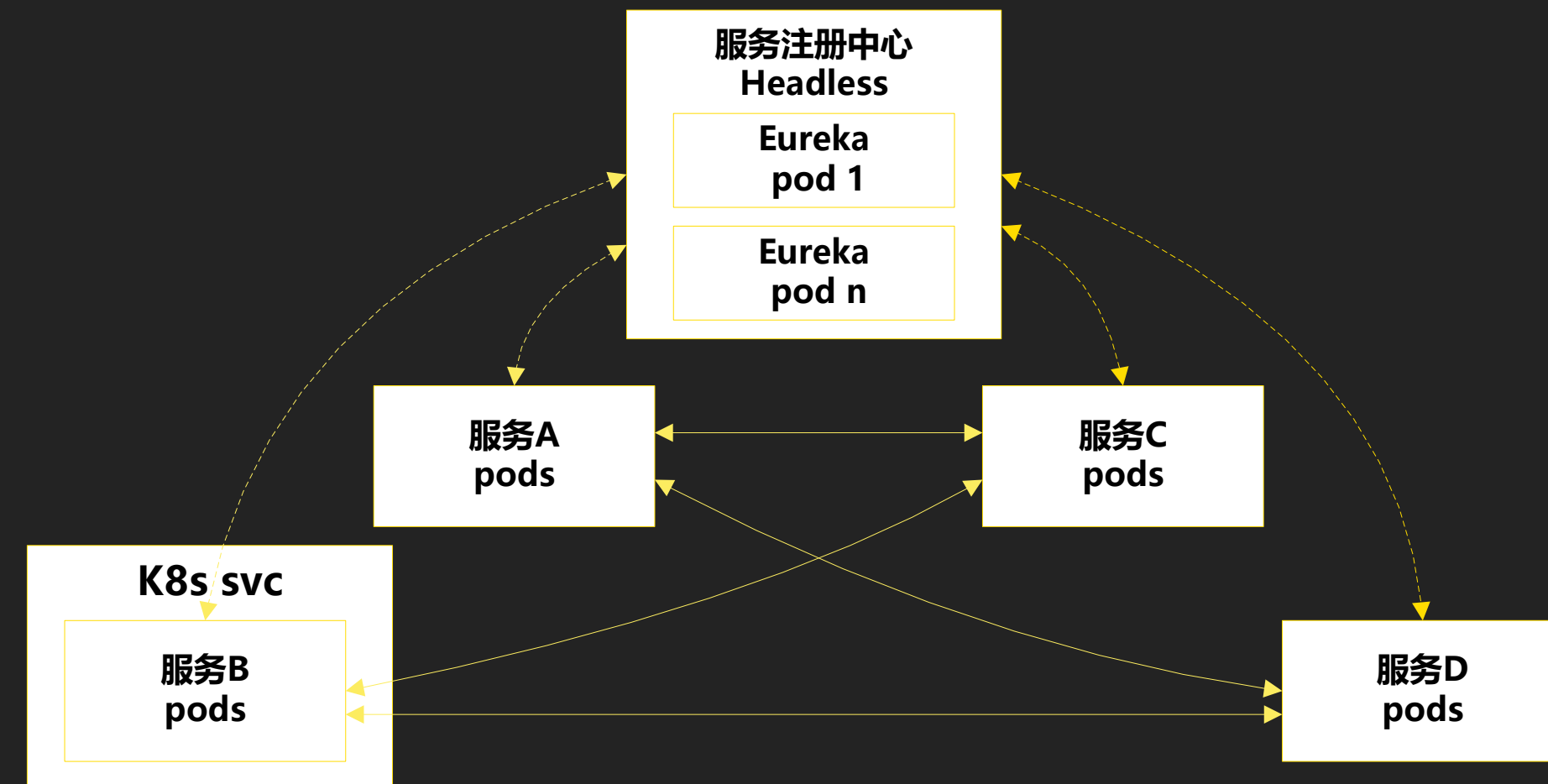
- gRPC API, 基于zookeeper服务发现
- nodejs, go
- 无状态业务 (k8s deploy)
- 主机网络: 供非容器的上游服务访问
- 随机端口: 单节点注册多个实例
- 容器退出: 向zk反注册





容器化业务架构-spring cloud

- 全部组件容器化部署
- Eureka: 有状态部署
- Headless: eureka-0, eureka-n
- configmap: eureka.client.serviceUrl***
- K8s svc: 外部接口, 供容器外部访问



优化实践

宿主机优化：单节点百万TCP并发，`/etc/sysctl.conf`，`/etc/security/limits.conf`

容器优化：`initContainer`内执行优化脚本，`sysctl -w **`

容器日志自动清理：通过环境变量指定待清理日志路径，日志最长保留时长

监控conntrack：`conntrack -S`中的`insert_failed`值，`nf_conntrack table full`

扩缩策略：k8s HPA，手动扩缩pods或宿主机节点

调度策略：基于节点label+nodeAffinity对特殊应用分组调度

业务优化：GOMAXPROCS与申请的CPU资源相匹配，jdk8+感知cgroup CPU/memory限制

优化实践-kubedns

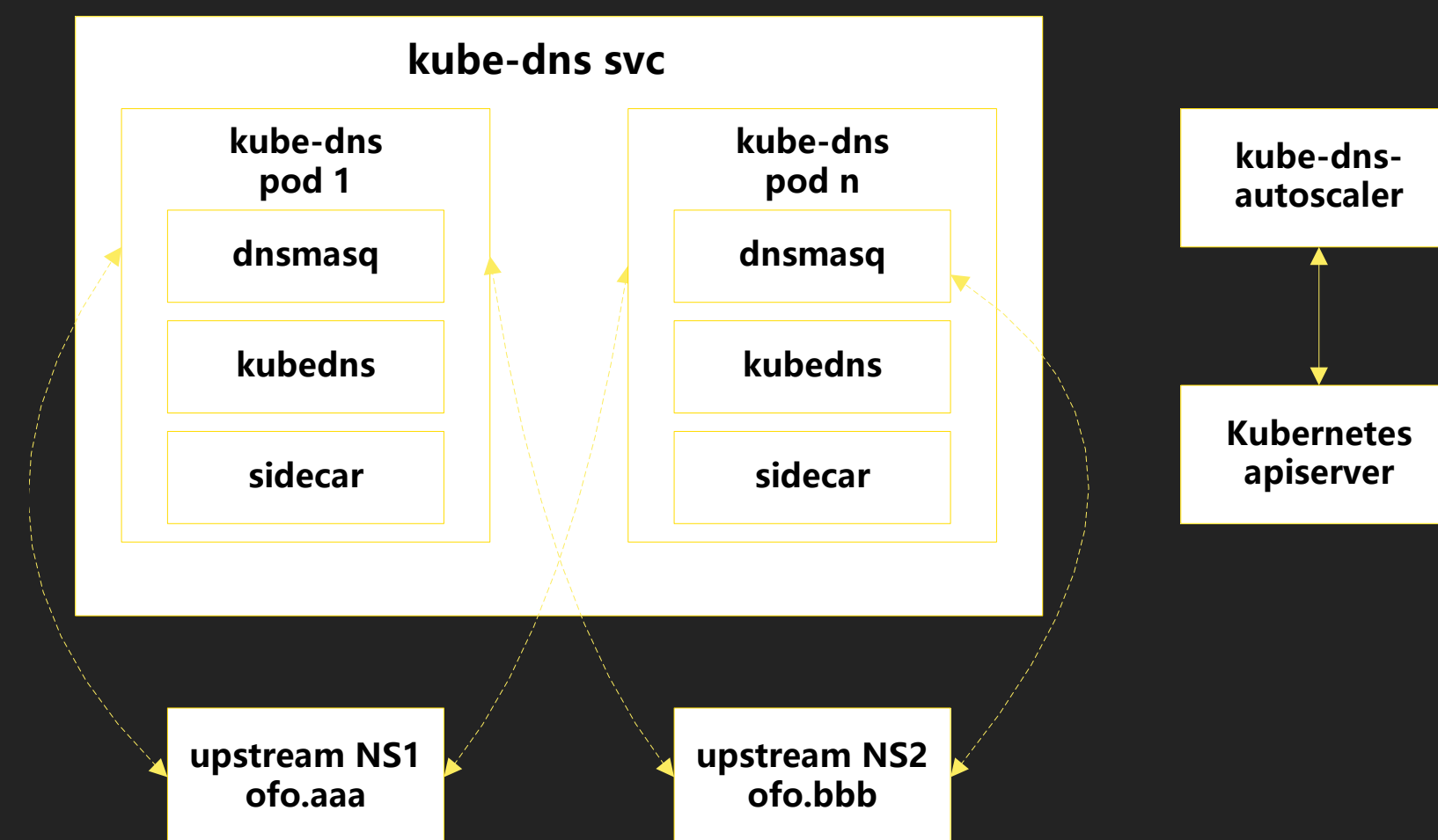
自动扩缩：基于kube-dns-autoscaler

调整参数：--cache-size, --dns-forward-max, --neg-ttl

增大cpu/memory

兼容本地域：kube-dns cm配置upstream NS

添加默认搜索域



优化实践-Alpine Docker Image

go/nodejs/java业务的基础镜像

TLS证书： 提取ca-certificates apk，仅保留ca-certificates.crt

优化脚本： sysctl -w ***, net.core.somaxconn等

普通账号： www权限启动容器进程，特殊命令添加至sudoers

本地化： 东八区时区，国内apk源，nodejs源

DNS解析超时： 修改musl libc源码，禁止默认并行ipv6请求

```
//musl/src/network/lookup_name.c, function 'name_from_dns'
static const struct { int af; int rr; } afrr[2] = {
    { .af = AF_INET6, .rr = RR_A },
    { .af = AF_INET, .rr = RR_AAAA },
};

for (i=0; i<2; i++) {
    if (family != afrr[i].af) {
        qlens[nq] = __res_mkquery(0, name, 1, afrr[i].rr,
            0, 0, 0, qbuf[nq], sizeof *qbuf);
        if (qlens[nq] == -1)
            return EAI_NONAME;
        nq++;
    }

    //hack: if set the AF_UNSPEC family, just return ipv4 result
    if (family == AF_UNSPEC) break; the fix
}
```

优化实践-GOMAXPROCS

问题现象：同配置（8核16G）的容器go业务比ECS方式性能损耗一半以上

问题分析：容器宿主机iowait是ECS下的3倍，go业务线程数是ECS下的2倍。容器按宿主机CPU核数启动go工作线程，当容器CPU限制低于宿主机CPU核数时出现了争用，因此要为容器设置正确的**GOMAXPROCS**环境变量

优化结果：如右图


```
[... wrk]$ wrk -t10 -c100 -s ./fence.lua -d60s -T30s http://10.1.1.1:231:2500/.../school
Running 1m test @ http://10.1.1.1:231:2500/.../school
 10 threads and 100 connections
  Thread Stats   Avg      Stdev     Max   +/-  Stdev
    Latency    9.28ms    9.98ms 137.09ms   89.60%
   Req/Sec    1.38k    347.37   2.19k    61.10%
825097 requests in 1.00m, 152.65MB read
Requests/sec: 13745.32
Transfer/sec:    2.54MB

[... wrk]$ wrk -t10 -c100 -s ./fence.lua -d30s -T30s http://10.1.1.1:186:2500/.../school
Running 30s test @ http://10.1.1.1:186:2500/.../school
 10 threads and 100 connections
  Thread Stats   Avg      Stdev     Max   +/-  Stdev
    Latency   10.05ms   11.23ms 144.10ms   88.43%
   Req/Sec    1.34k    429.31   2.23k    58.27%
399973 requests in 30.01s, 74.00MB read
Requests/sec: 13326.34
Transfer/sec:    2.47MB
```




2018 杭州·云栖大会
THE COMPUTING CONFERENCE

 **Alibaba** Group
阿里巴巴集团

 ofo 小黄车

驱动数字中国

EMPOWER DIGITAL CHINA