

新智超脑PaaS云平台介绍

kubernetes在能源企业落地实践

陈晓宇

PaaS云平台相关技术介绍

kubernetes和企业应用的差距

我们做了什么

我们正在做的，还有我们踩过的

客户案例

主办：



新奥集团简介



新奥集团股份有限公司创建于1989年，从燃气业务起步，经过持续的战略升级与产业拓展，以“**提升品质生活、推动智慧生产、创建美丽生态**”为使命，构建了**生态、生活和互联网**三大事业领域，旗下拥有**新奥能源 (02688.HK)、新奥股份 (600803.SH)、北部湾旅游 (603869.SH)** 等五家上市公司，业务覆盖国内**160多个**城市，以及亚洲、欧洲、美洲、澳洲等地区。

截止2016年底，新奥总资产和销售收入均**超千亿元**，员工总数超过**4.5万人**。



新奥能源
02688.HK



新奥股份
600803.SH



北部湾旅游
603869.SH



创始人
王玉锁



生态



生活



互联网

主办： CLOUD NATIVE COMPUTING FOUNDATION



PaaS云平台相关技术概览

PaaS云平台介绍

新智云PaaS云平台是基于 **Kubernetes** 的企业级**容器**云平台，提供以应用为中心的容器云产品和解决方案，涵盖轻量级容器虚拟化、微服务、DevOps、持续交付等，最大化地帮助企业实现业务应用的快速交付与持续创新。

PaaS云平台价值



应用环境的集中
管控降低运维成
本



DevOps 全流程交
付提高产品迭代效
率



运行环境快速就绪
简化资源管理



助力微服务框架
推动人员架构和
IT流程的变革



自动水平伸缩
容错和自我修复

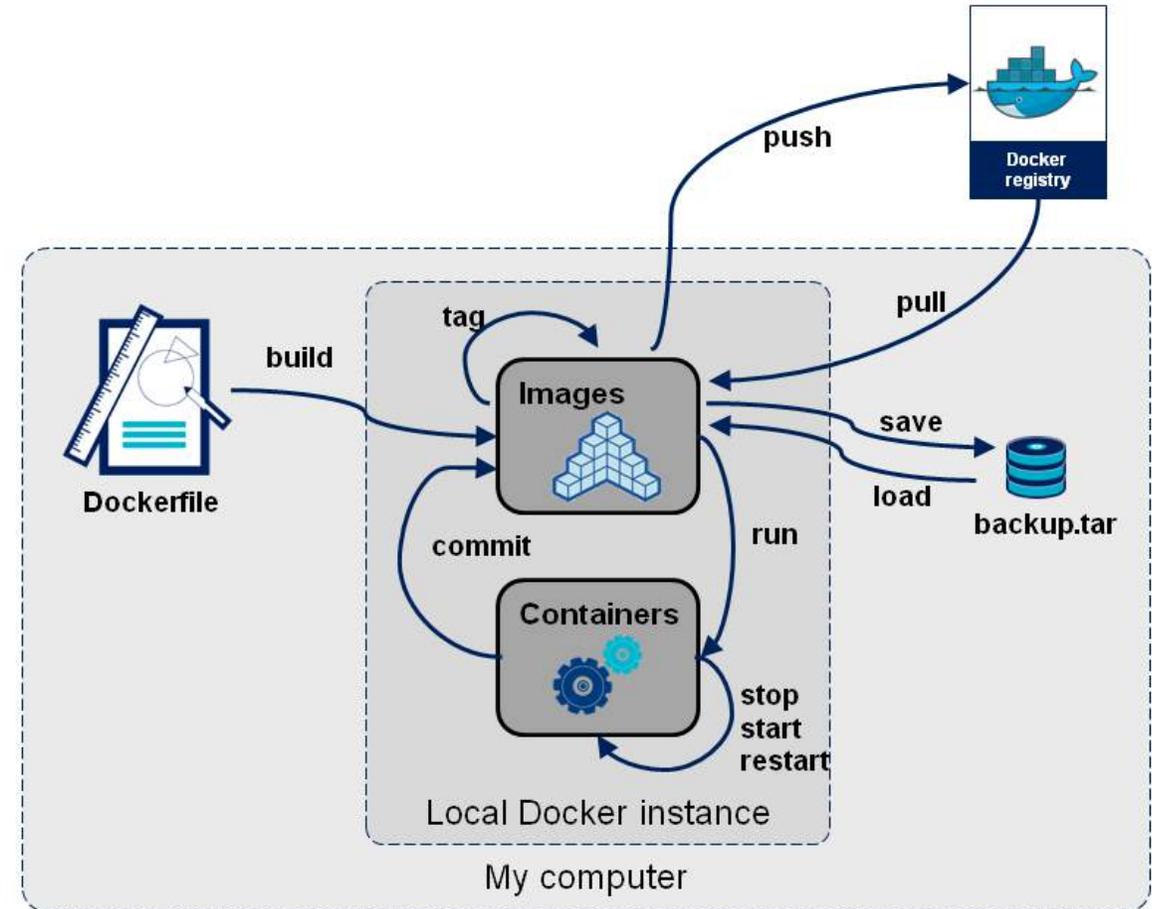
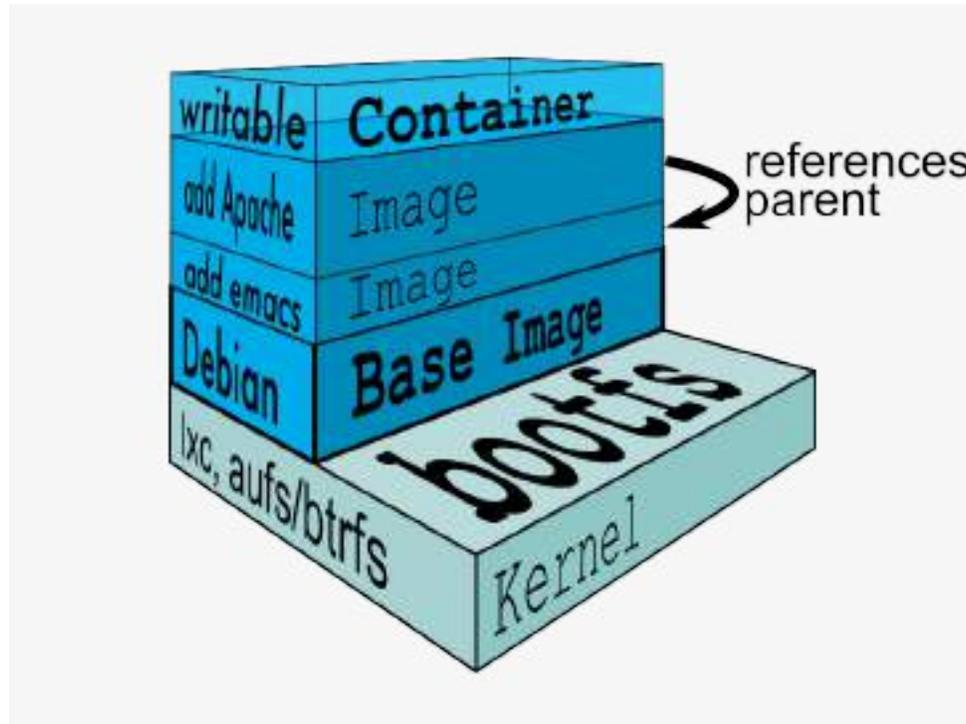
主办：



PaaS云平台相关技术概览

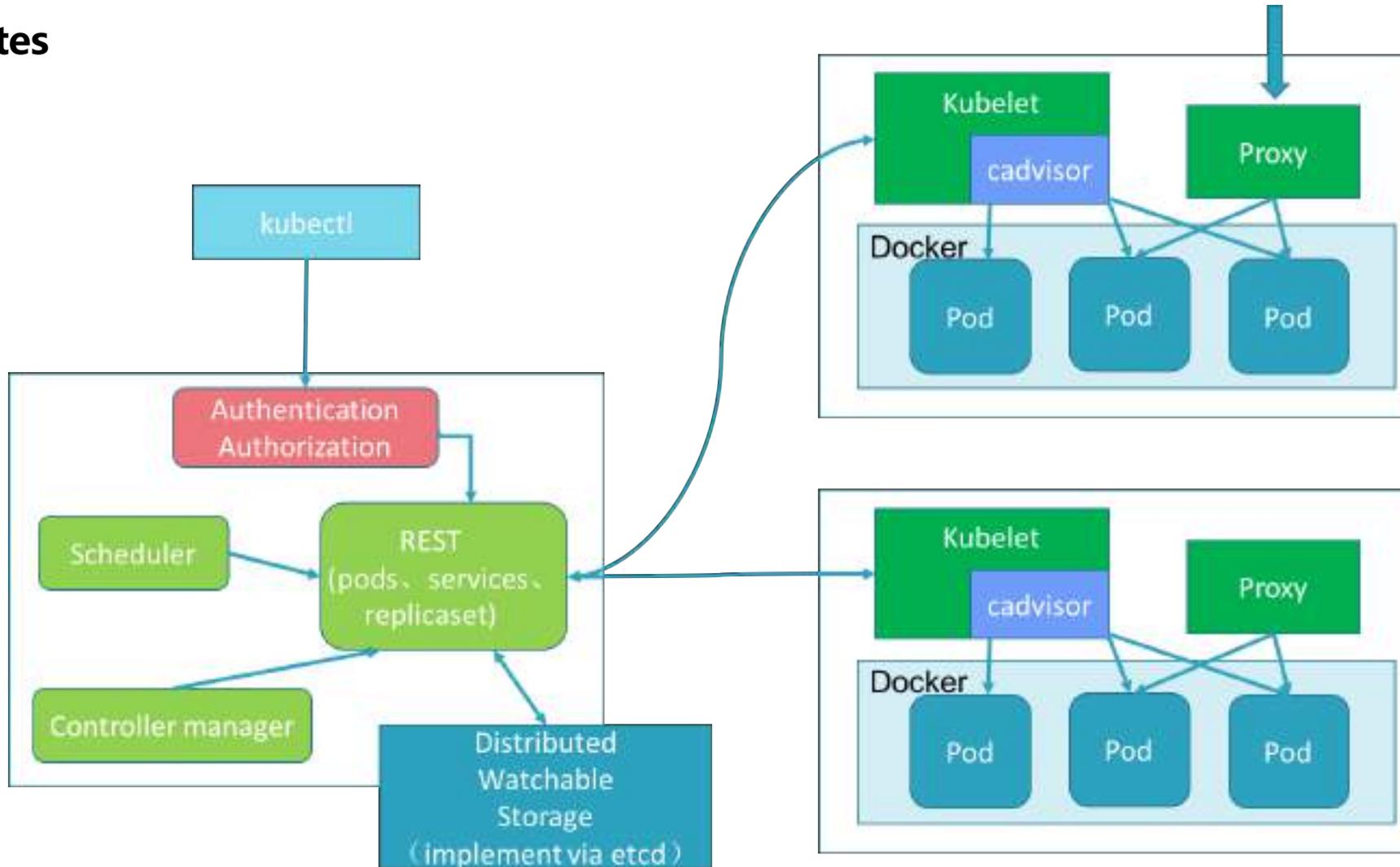


Docker



PaaS云平台相关技术概览

Kubernetes



kubernetes和企业应用的差距



1 它只负责管理容器

2 它把网络交给了“标准”

3 它把存储交给了第三方

4 环境配置、服务配置繁琐

5 应用程序有状态、集群化，难以管理和维护

为什么K8S不适合直接使用

6 传统应用如何迁移

7 服务如何暴露出去

8 如何监控容器的运行，错误如何排除

9 不懂容器技术的人如果上手

10 如何与Devops结合

我们做了什么



让K8S变得容易使用了！

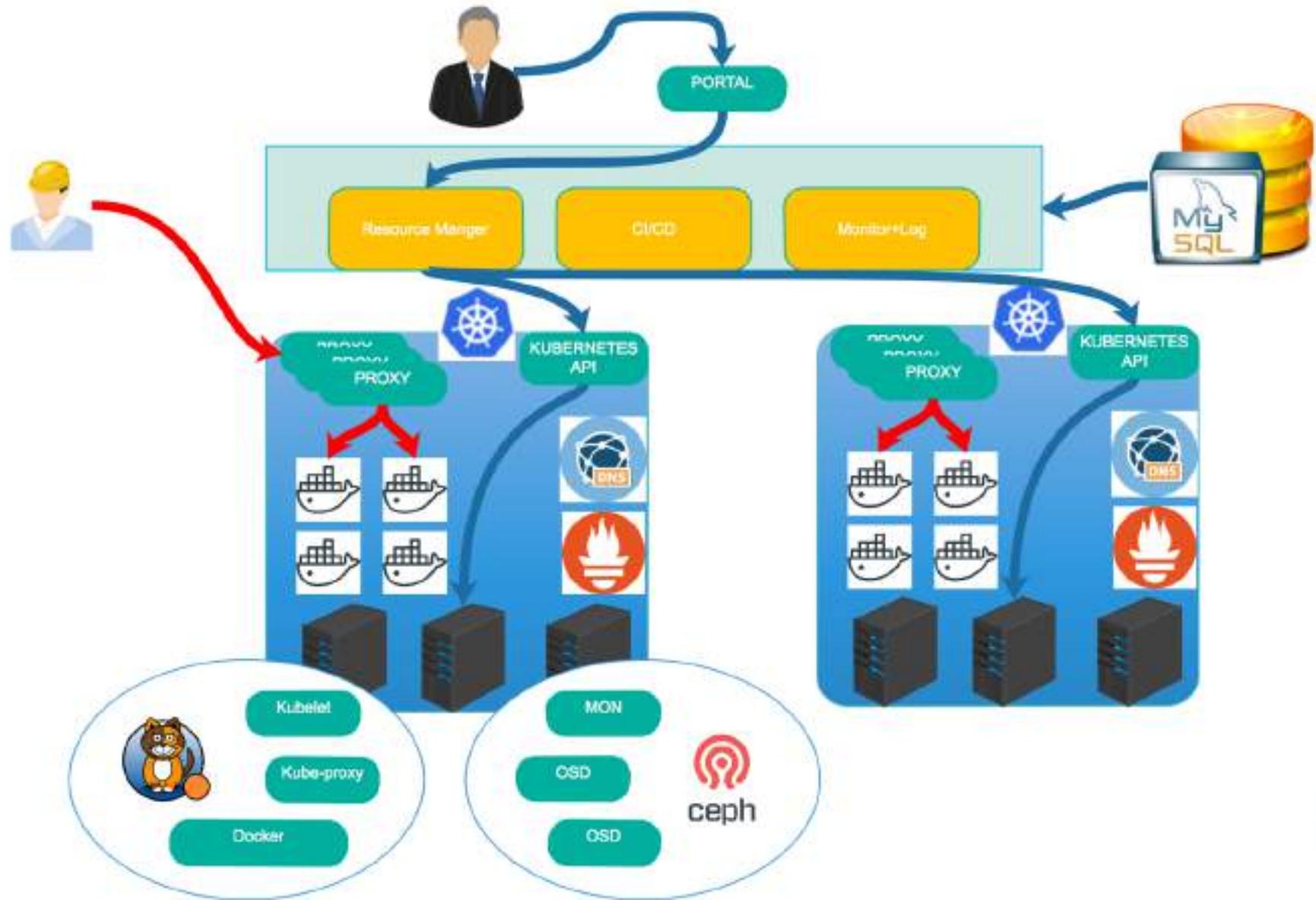
我们做了什么

- 开源方案整合
- Golang编写CI/CD
- Web管理页面
- 多租户资源管理
- 应用商店
- 工单审计
- 日志检索查询
- 网络规则配置
- 多集群管理
- Dockerfile可视化



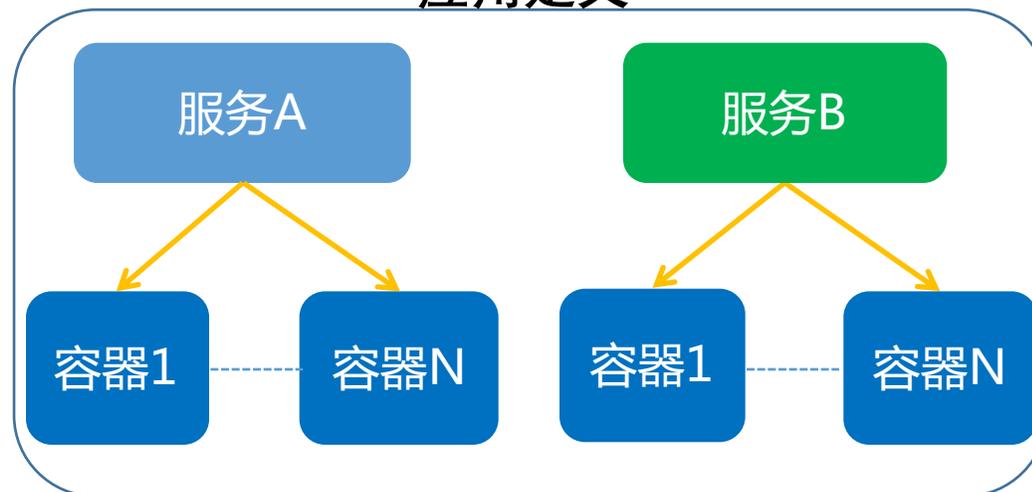
我们做了什么

多集群管理：
管理多个Kubernetes集群



我们做了什么

应用定义



服务创建流程



选择镜像名称和版本
查询公有和私有镜像

设置服务中容器个数

设置CPU和内存配额

配置存储大小
配置网络访问方式和访问出口

自定义环境变量
注入主机容器IP等

我们做了什么

```

apiVersion: v1
kind: Service
metadata:
  labels:
    app: kibana
  name: kibana
spec:
  type: NodePort
  ports:
  - name: web
    port: 5601
    protocol: TCP
  selector:
    app: kibana
  sessionAffinity: None
---
apiVersion: extensions/v1beta1
kind: Deployment
metadata:
  labels:
    app: kibana
  name: kibana
  namespace: default
spec:
  replicas: 1
  selector:
    matchLabels:
      app: kibana
  template:
    metadata:
      labels:
        app: kibana
    spec:
      containers:
      - image: kibana:5.4.3
        imagePullPolicy: IfNotPresent
        name: kibana
        env:
        - name: ELASTICSEARCH_URL
          value: "http://paasdev.enncloud.cn:9200"

```



实例数量 ↑

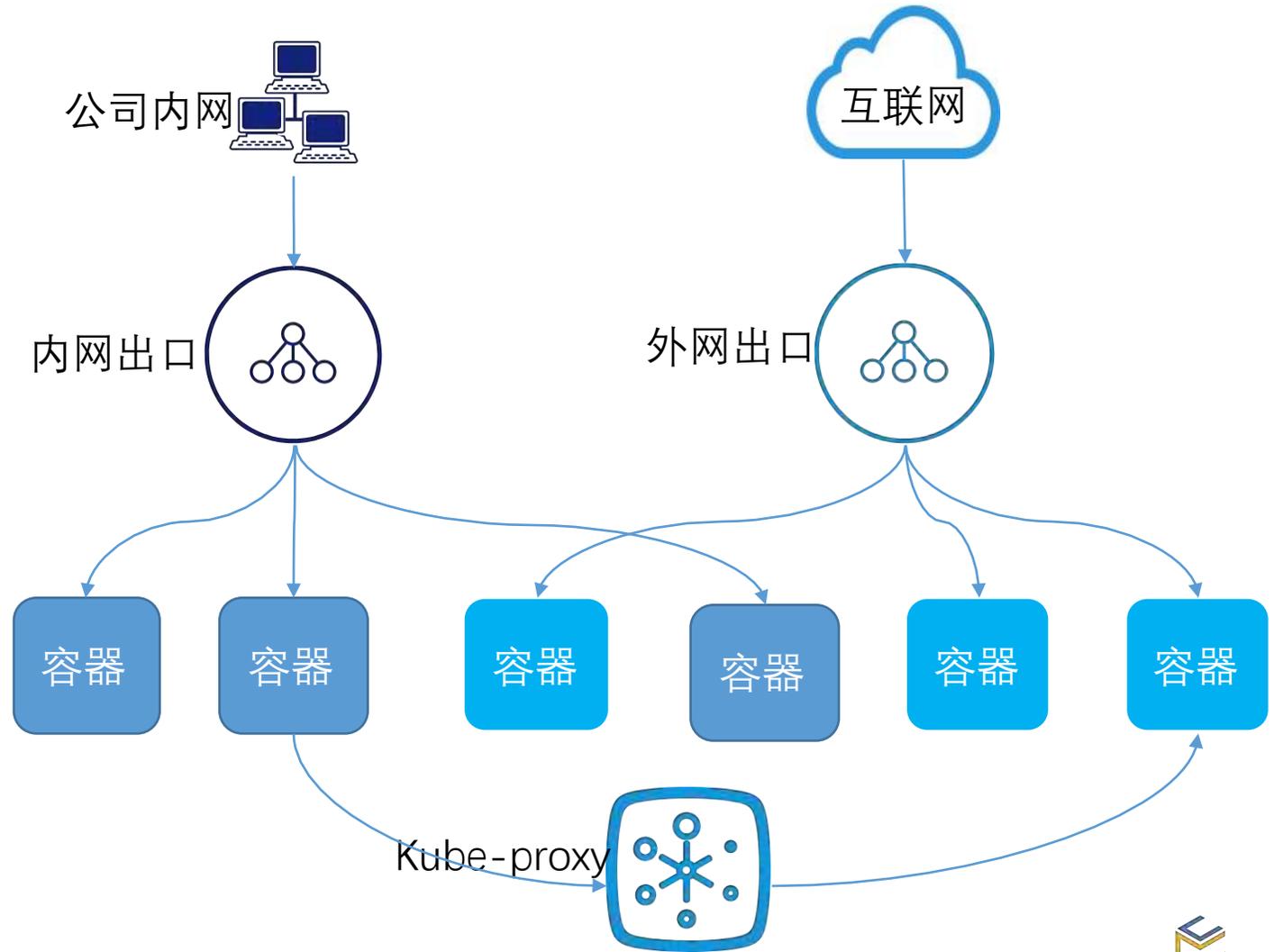
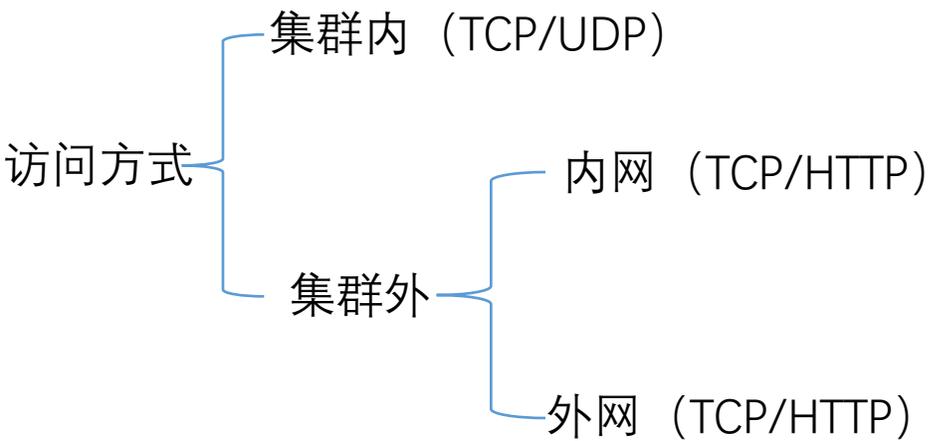
访问方式 可公网访问 内网访问 仅在集群内访问
 服务可通过内网访问，选择一个网络出口：

映射端口

容器端口	协议	映射服务端口
<input type="text" value="10120"/>	<input type="text" value="TCP"/>	<input type="text" value="动态生成"/>

我们做了什么

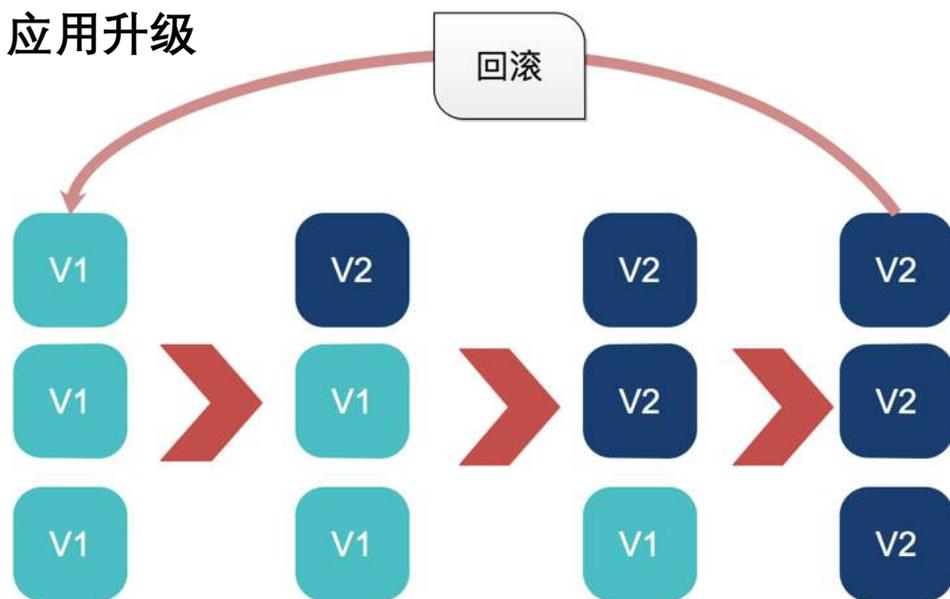
1.应用访问



- 多租户
- 回话保持

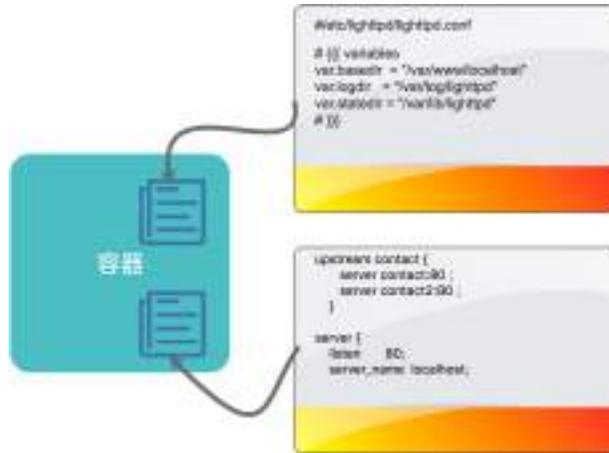
我们做了什么

应用升级



我们做了什么

服务配置/secret



定义配置文件



容器内配置文件

```

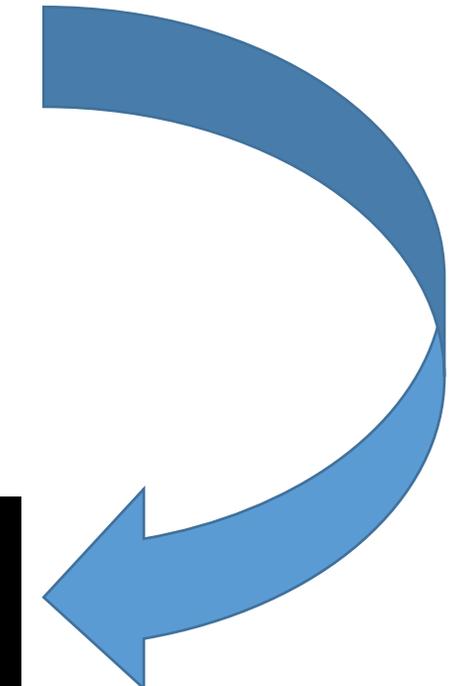
/etc/nginx/conf.d # cat default.conf
server {
    listen 80;
    server_name localhost;

    location / {
        root /usr/share/nginx/html;
        index index.html index.htm;
    }

    location /CDTPWebGateway/ {
        proxy_pass http://cdtpwebgateway-xdgc-idesign-dev;
    }

    error_page 500 502 503 504 /50x.html;
    location = /50x.html {
        root /usr/share/nginx/html;
    }
}

```



我们做了什么

应用商店

根据不同类别提供常用镜像，一键即可部署应用集群。

Web服务器	Tomcat 8 HAProxy Httpd
持续集成与部署	SonarQube 代码分析 Jenkins 持续集成 Gitlab 代码管理
开发环境	Java 开发环境
数据库与缓存	MySQL Redis PostgreSQL

模板应用名称:

选择应用分类:

镜像: 选择镜像

镜像版本: 缺省值: 镜像版本 创建时如未选择版本 提示: 请选择镜像版本

配置文件名:

定义配置文件 [查看模板](#)

```

#member
ETCD_NAME="$hostname"
ETCD_DATA_DIR="/etcd-data"
ETCD_LISTEN_PEER_URLS="http://$hostip:2380"
ETCD_LISTEN_CLIENT_URLS="http://$hostip:2379"
#cluster
ETCD_INITIAL_ADVERTISE_PEER_URLS="http://$hostname:2380"
ETCD_INITIAL_CLUSTER_STATE="new"
ETCD_INITIAL_CLUSTER_TOKEN="test-token"
ETCD_ADVERTISE_CLIENT_URLS="http://$hostip:2379"
${!AICount:=AICount}
ETCD_INITIAL_CLUSTER="{[range $index,$size - 1]}{if eq $index $AICount}{{$hostname}}=http://{{$hostname}}:2380[[else]]
{{$hostname}}=http://{{$hostname}}:2380.{{end}}[[end]]"
#!$! etcd-qinzha0-0,etcd-qinzha0-1,etcd-qinzha0-2
    
```

[跳转到帮助文档](#)

预览配置文件

数据存储路径: 缺省值: etcd

容器端口:

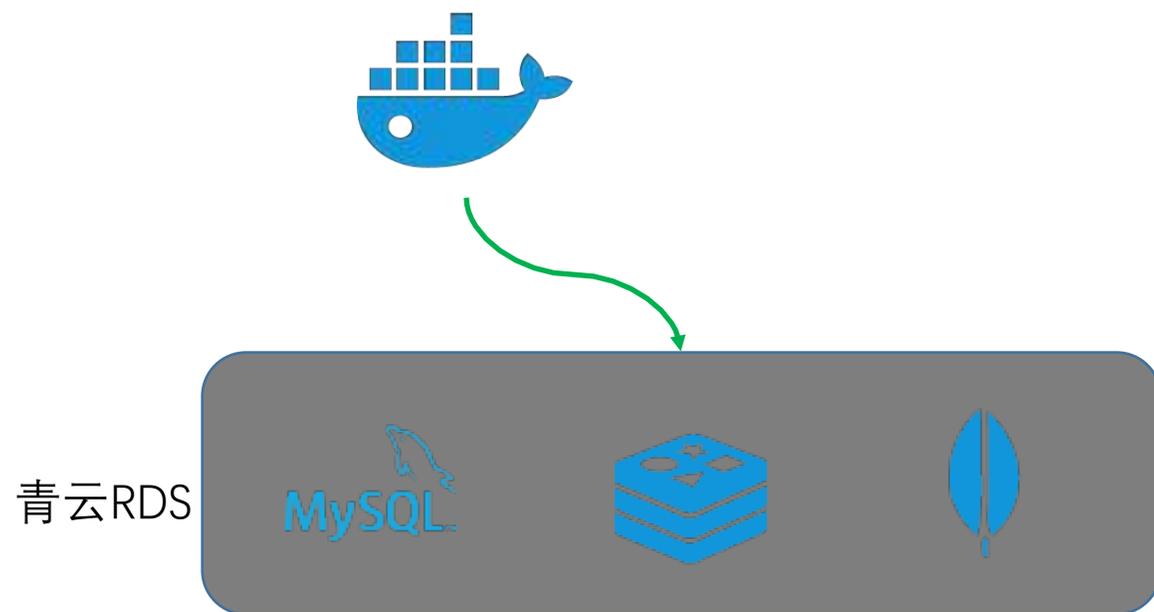
端口	端口类型	协议	操作
10120	内部端口	TCP	<input type="text"/>
	外部服务端口	TCP	<input type="text"/>

[添加容器端口](#)

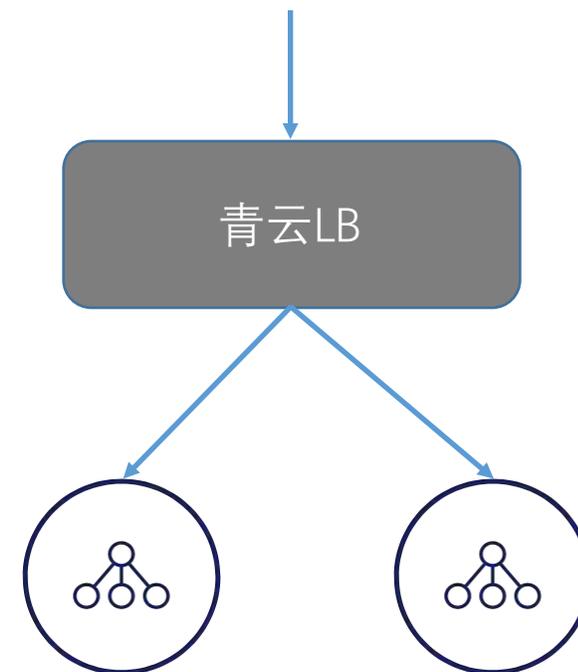
执行方式: 重启容器 加载配置 缺省值: 重启容器

我们做了什么

对接青云RDS



对接青云负载均衡



我们做了什么

镜像仓库

集成Harbor镜像管理

- 1.打通用户权限
- 2.镜像同步
- 3.ImagePullSecrets关联
- 4.镜像扫描 (harbor1.1.1已支持)



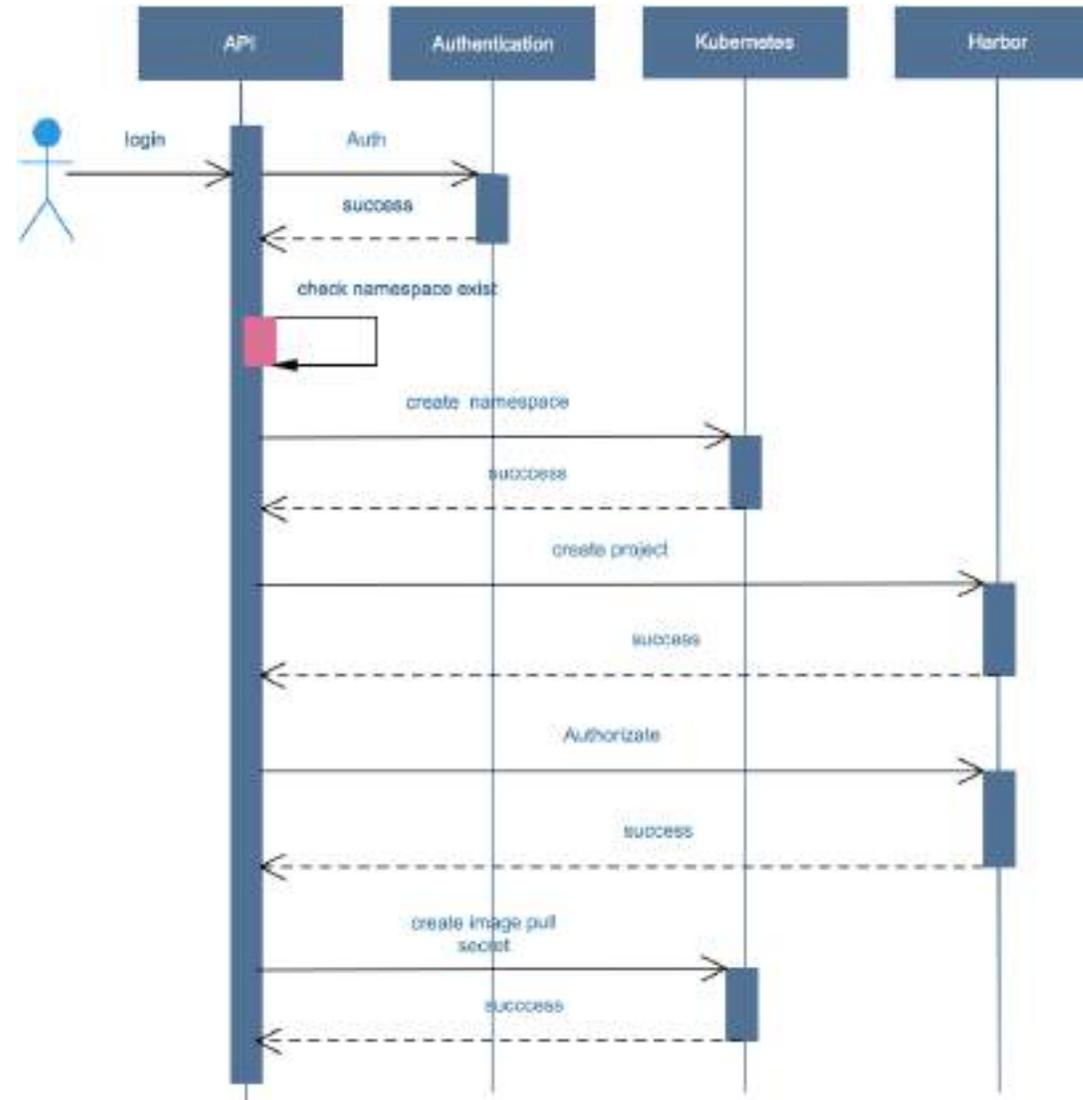
我们做了什么

创建命名空间

创建Project

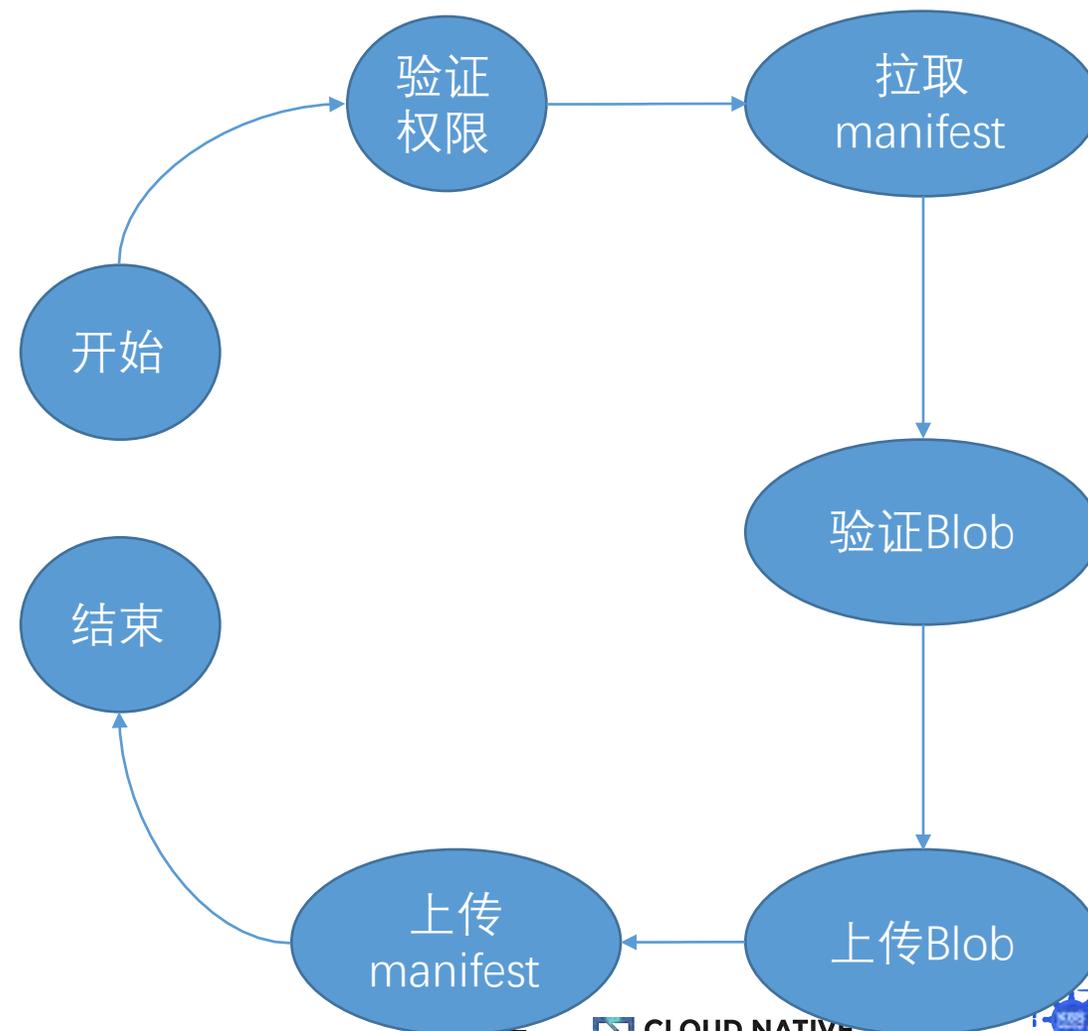
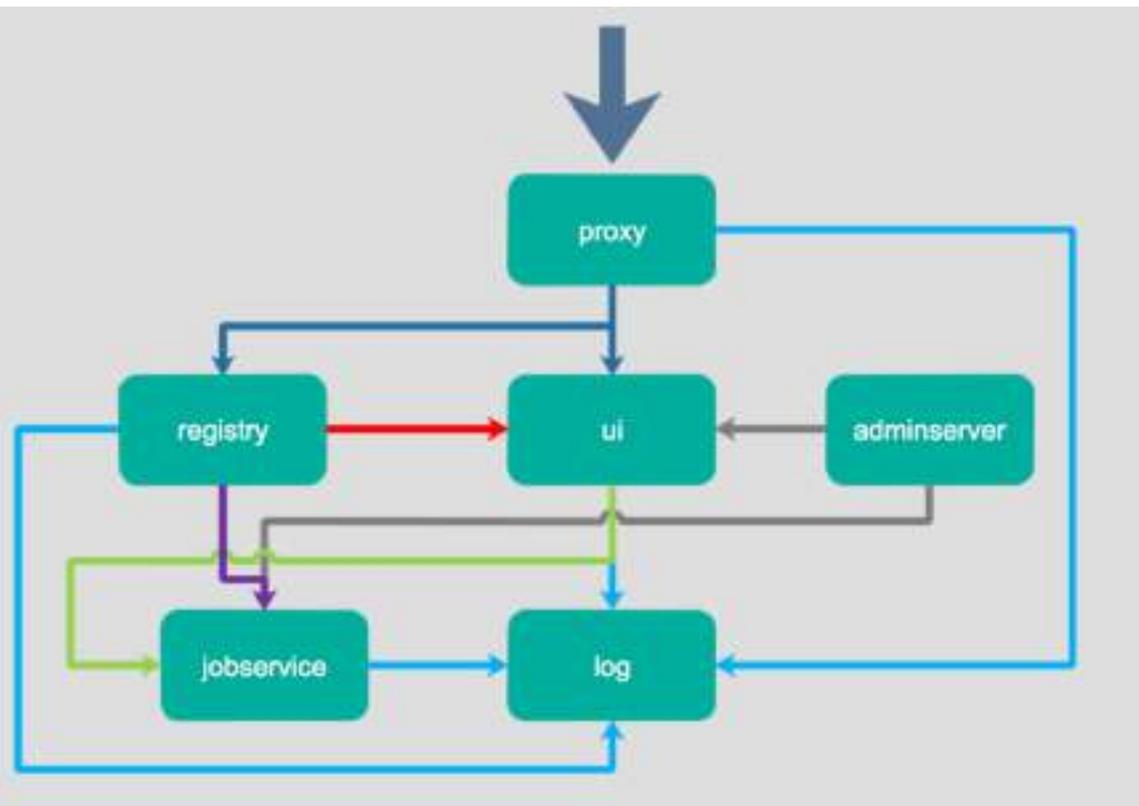
授权

创建并管理secret



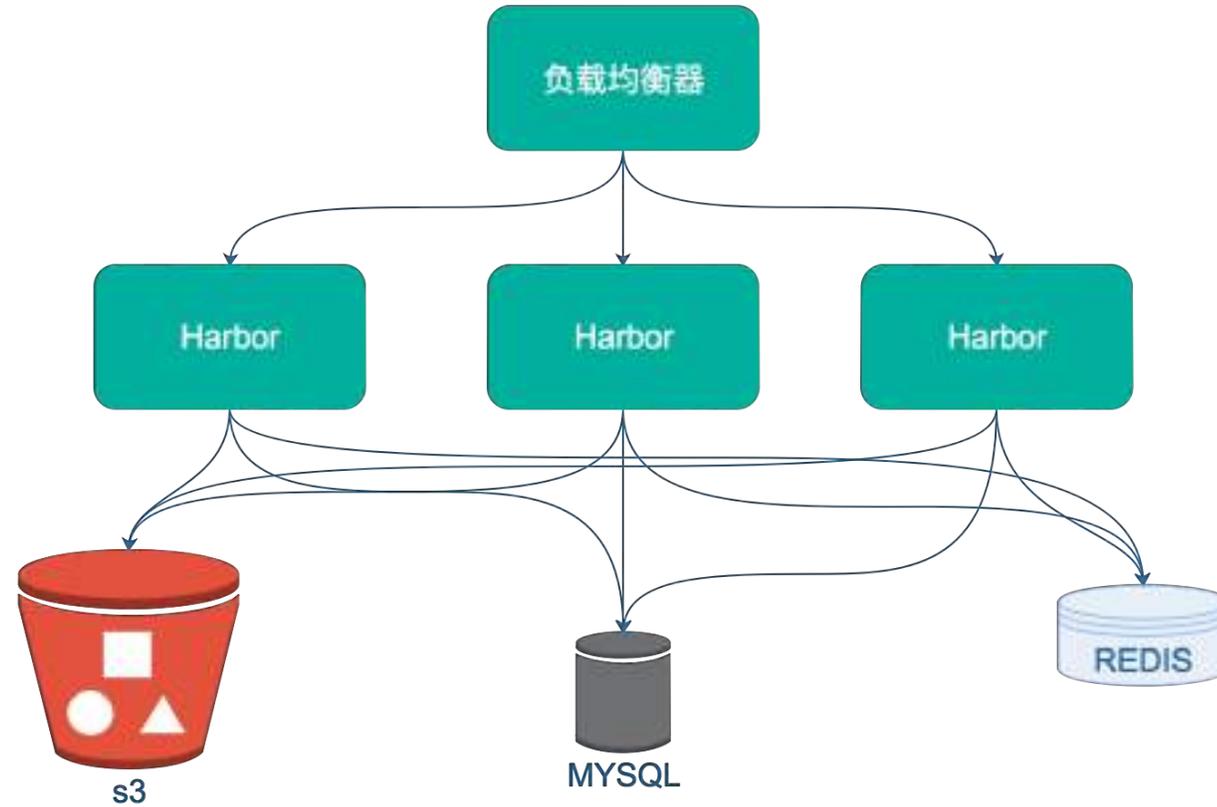
我们做了什么

5. 镜像Project之间拷贝、公网下载



我们做了什么

6.Harbor高可用部署



我们做了什么

▼ DevOps旨在通过共同目标,成就协同与效率,提升用户体验

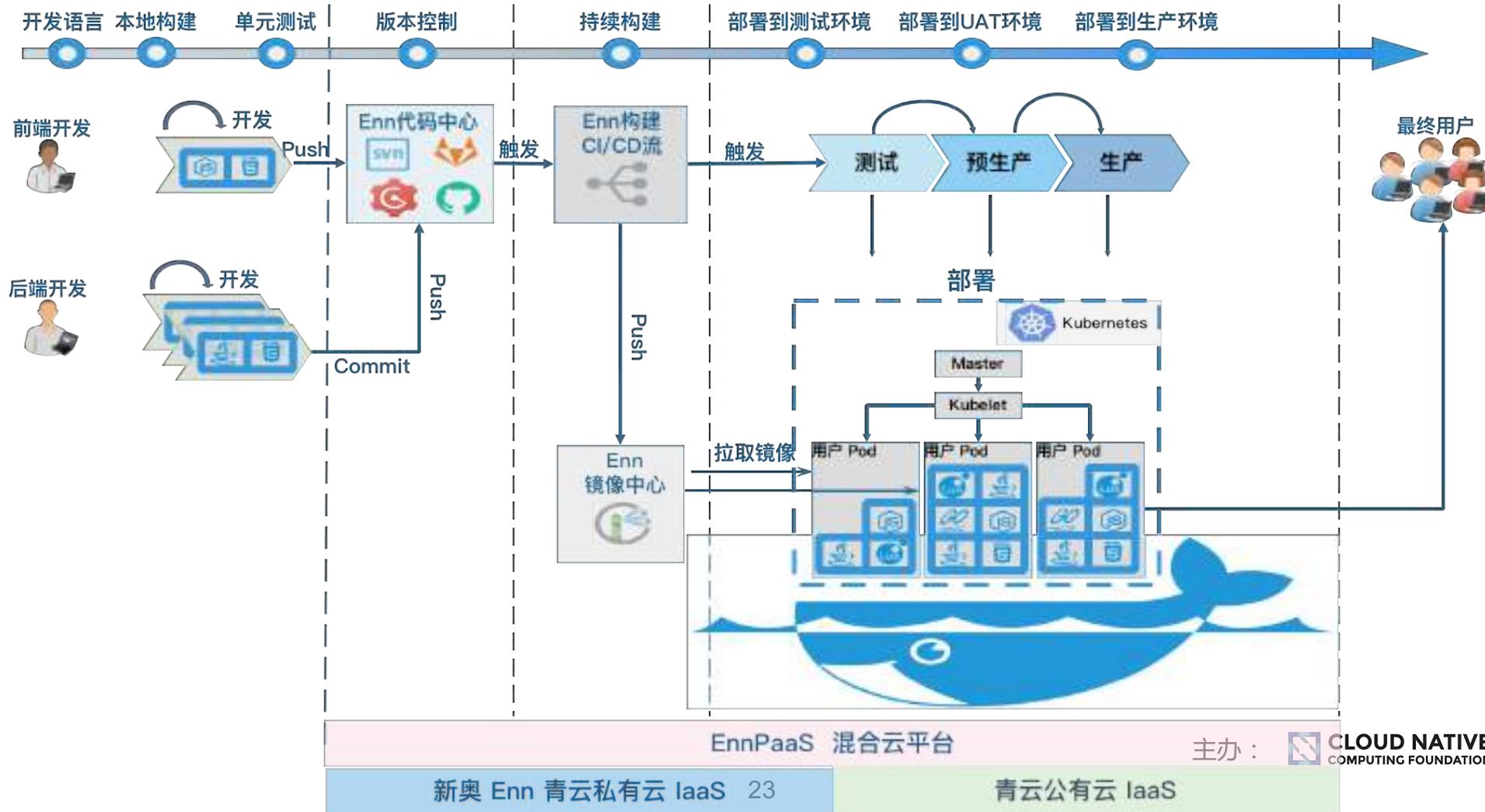


主办：



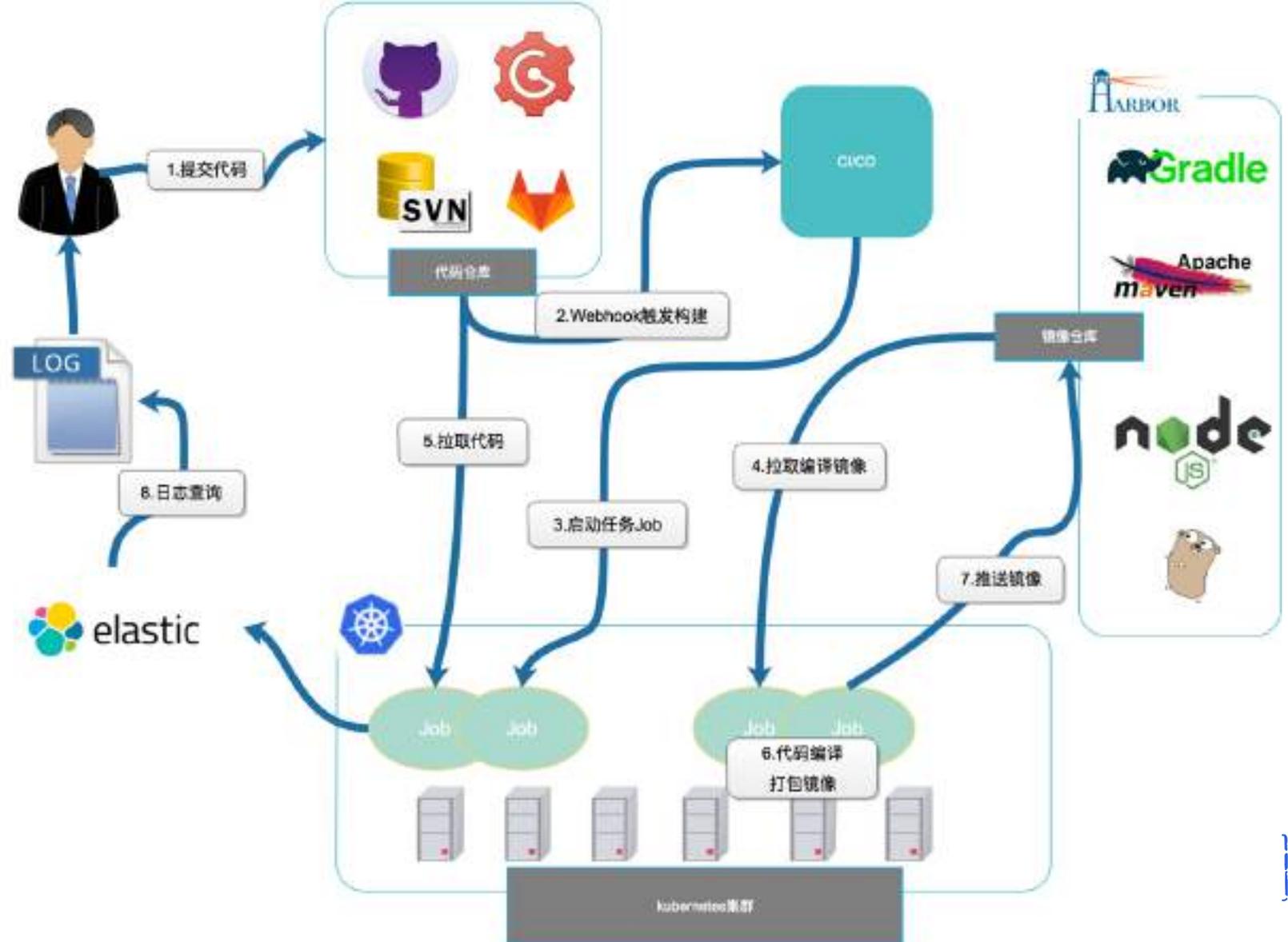
我们做了什么

EnnPaaS敏捷开发生产云：开发 → 生产 生命周期



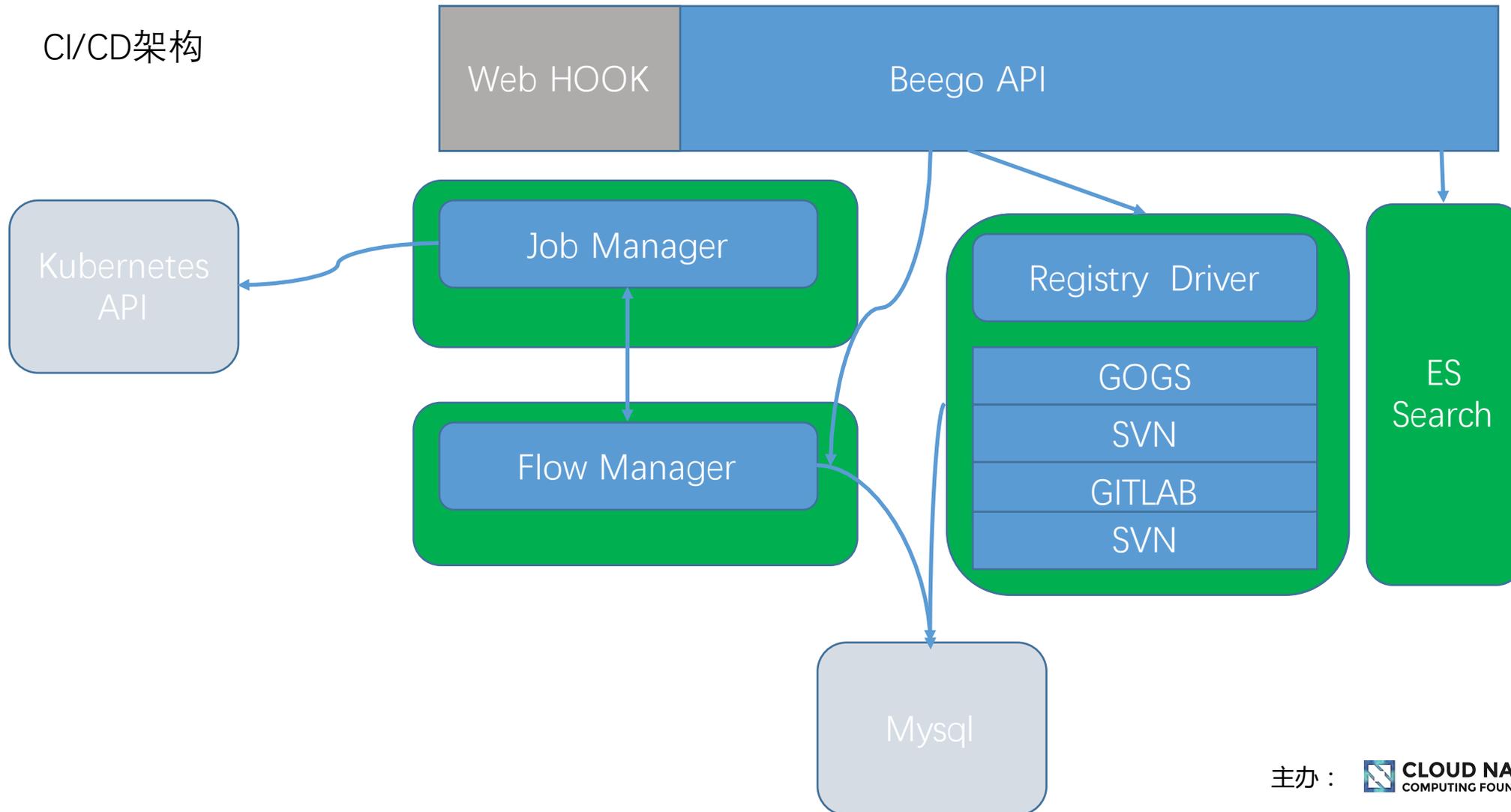
我们做了什么

以可视化 Flow 流程的方式，构建开发测试流水线，实现编译、集成、打包、测试、发布的自动化。



我们做了什么

CI/CD架构



主办：



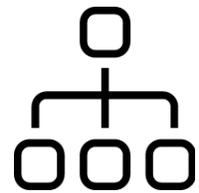
我们做了什么



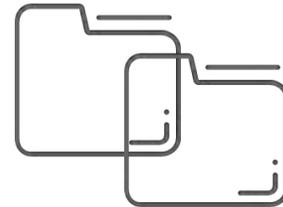
多租户



人员管理



组织管理



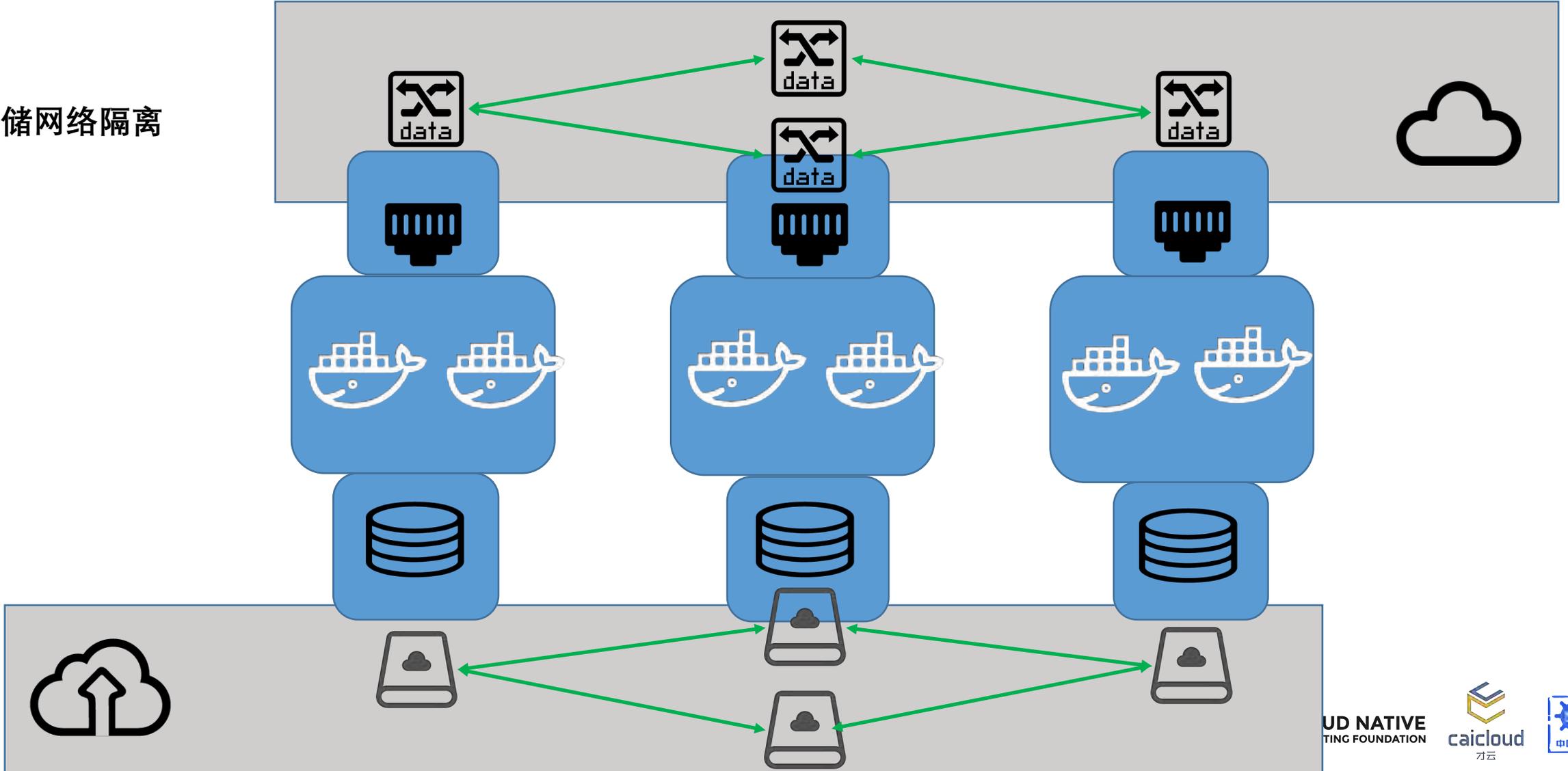
资源隔离



计费计量

我们做了什么

存储网络隔离



我们做了什么

日志



永久保留历史日志



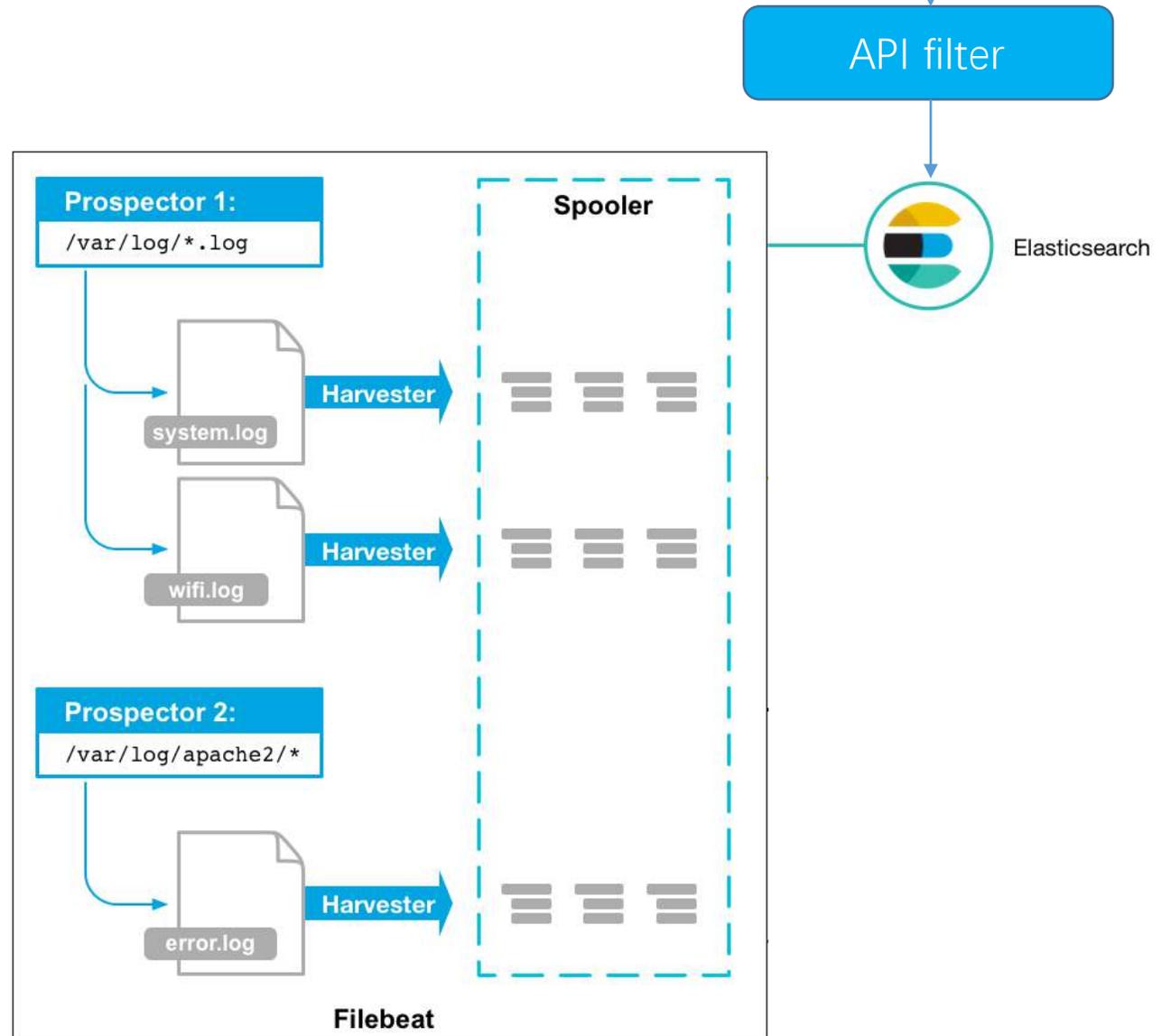
关键字或者时间维度综合查询



多租户、多集群统一日志采集

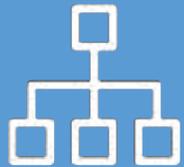


指定目录监控



我们做了什么

监控



关联自动伸缩组



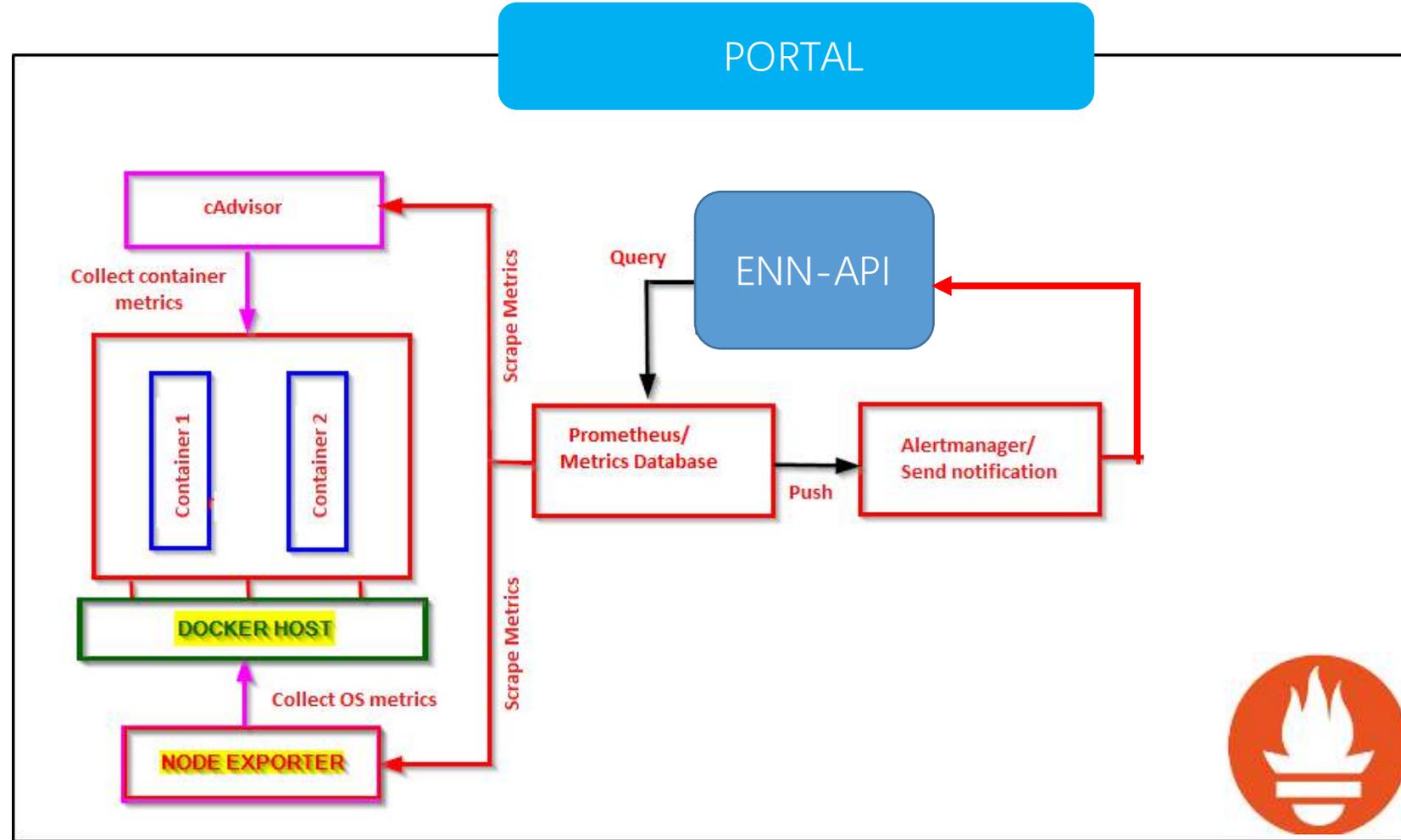
服务指标采集并能根据服务和容器粒度查询



Prometheus 集群



多维度告警去重和邮件通知

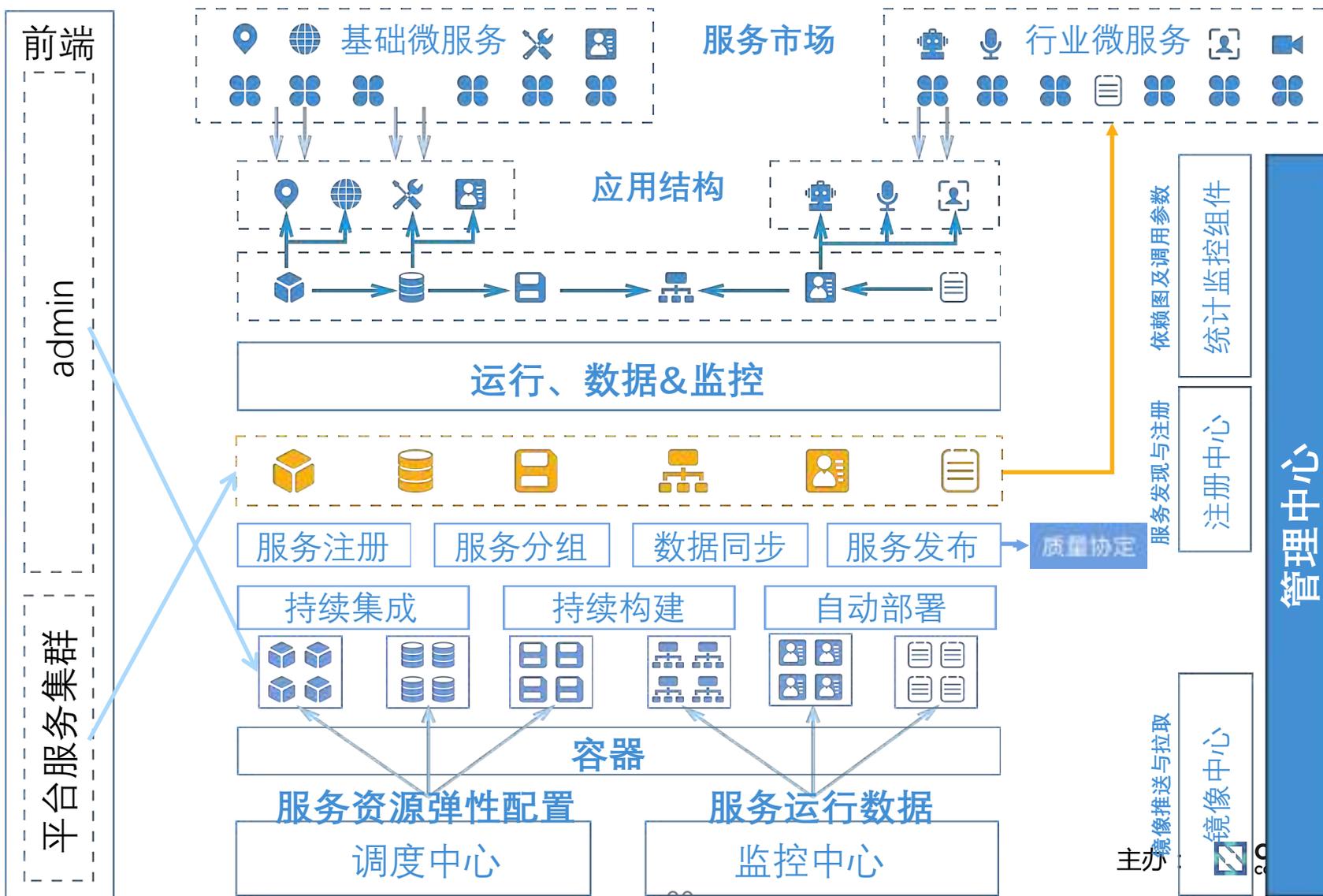


主办：



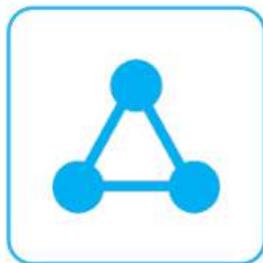
我们做了什么

微服务支撑



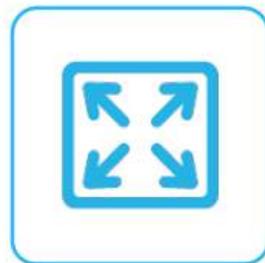
我们做了什么

应用连续性



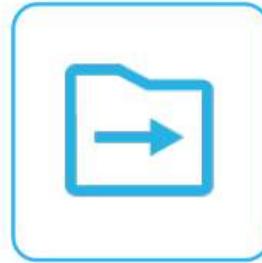
负载均衡

应用负载分流
问题节点自动隔离



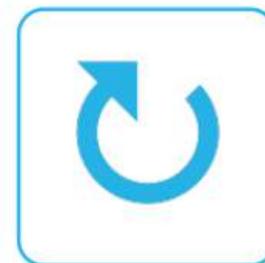
弹性伸缩

容器实例自动伸缩
灵活的管理策略



自动迁移

独特的副本控制器技术，保障实例数不变



自动重启

应用故障，自动重启

系统可靠性



全组件HA

所有系统组件均已实现容器化及HA



Master高可用

集群Master节点高可用，自动切换



存储高可用

有状态服务存储卷高可用



安全隔离

集群节点添加标记可安全隔离

主办：



我们正在做的，还有我们踩过的

我们踩过的坑

- 1 ceph存储挂载与解挂
- 2 Haproxy稳定性
- 3 Docker存储占用
- 4 并发的日志查询
- 5 kubernetes新版本升级

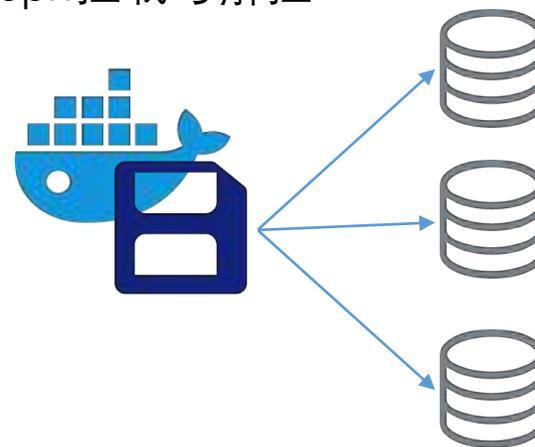
我们正在做的，还有我们踩过的

1.Prometheus挂掉



storage.local.target-heap-size

2.Ceph挂载与解挂



- locked by other nodes
- failed to mount rbd volume

我们正在做的，还有我们踩过的

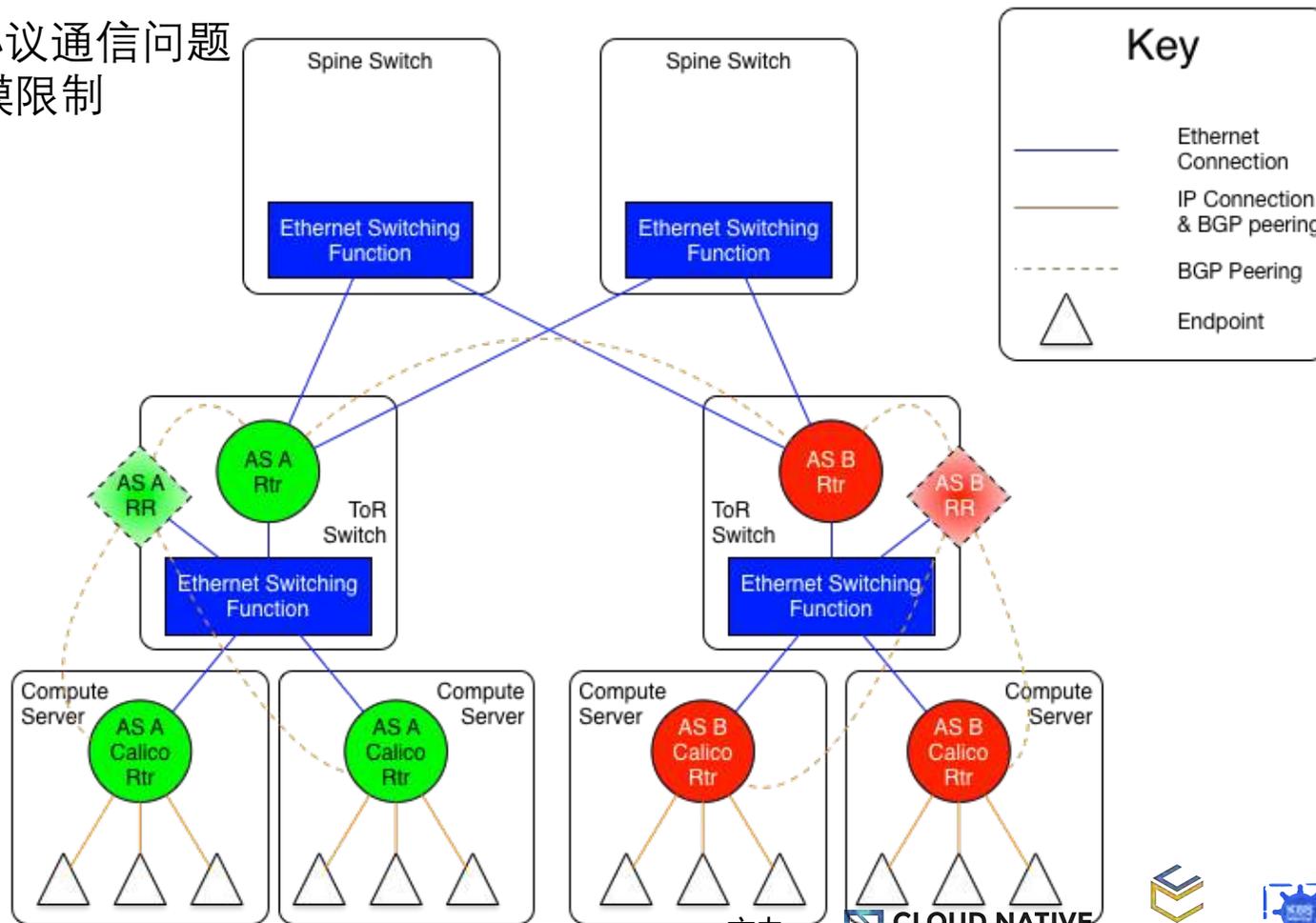
3.Kubernetes新版升级

批量导出导出成yaml
修改后导入

- apiVersion
- Kind
- initcontainer

4.calico网络规模

- 跨网段BGP协议通信问题
- 单个网段规模限制



我们正在做的，还有我们踩过的

我们正在做的

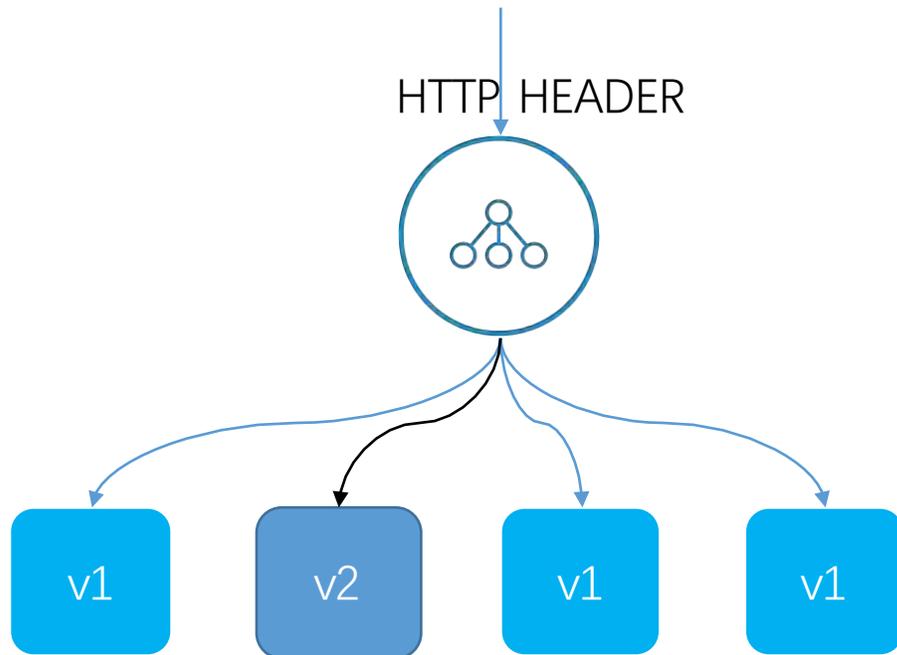
- 1 灰度发布
- 2 历史日志对象存储对接
- 3 自定义监测恢复脚本和预警
- 4 服务链路跟踪
- 5 和IAAS联动

主办：

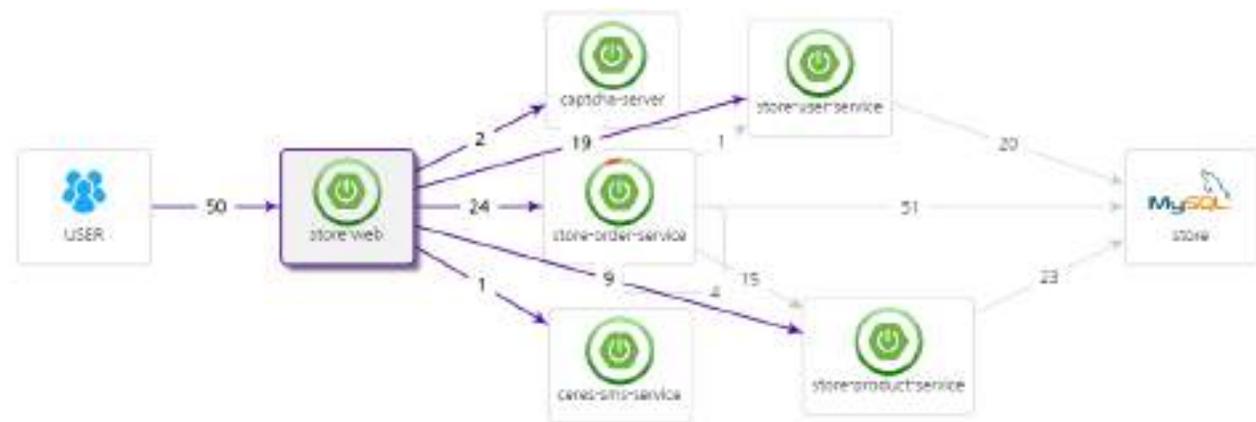


我们正在做的，还有我们踩过的

1. 灰度发布

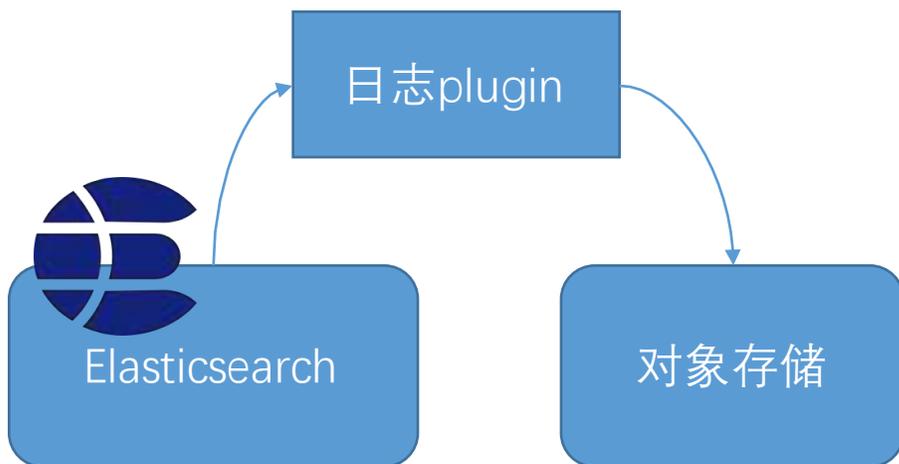


2. 服务链路跟踪

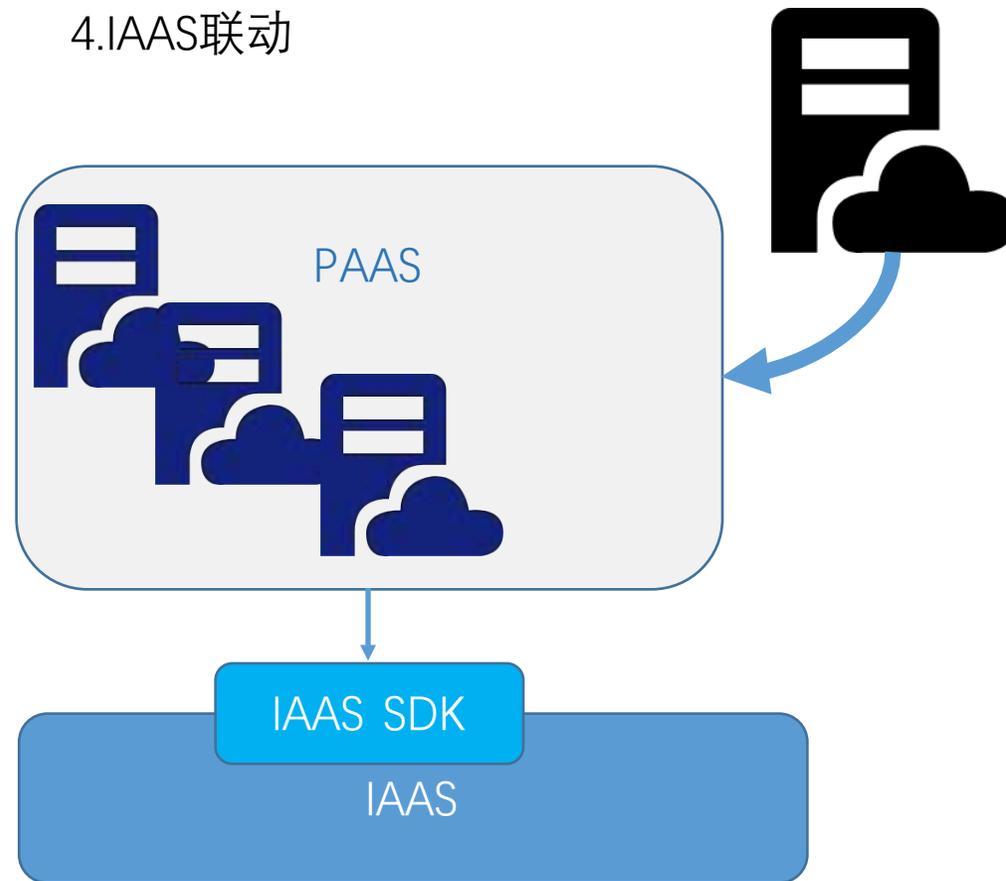


我们正在做的，还有我们踩过的

3. 日志长时间存储和下载



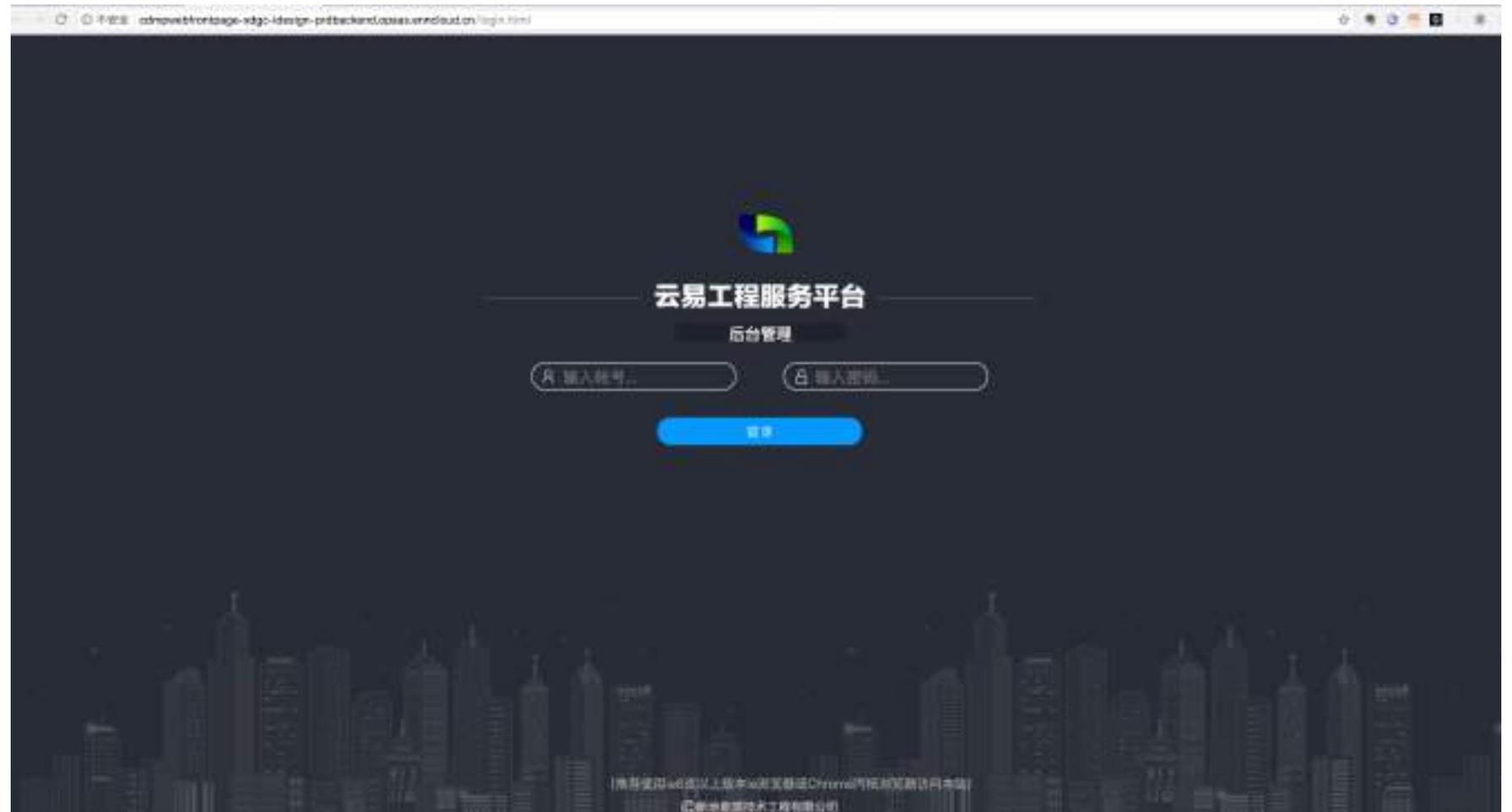
4. IAAS联动



客户案例

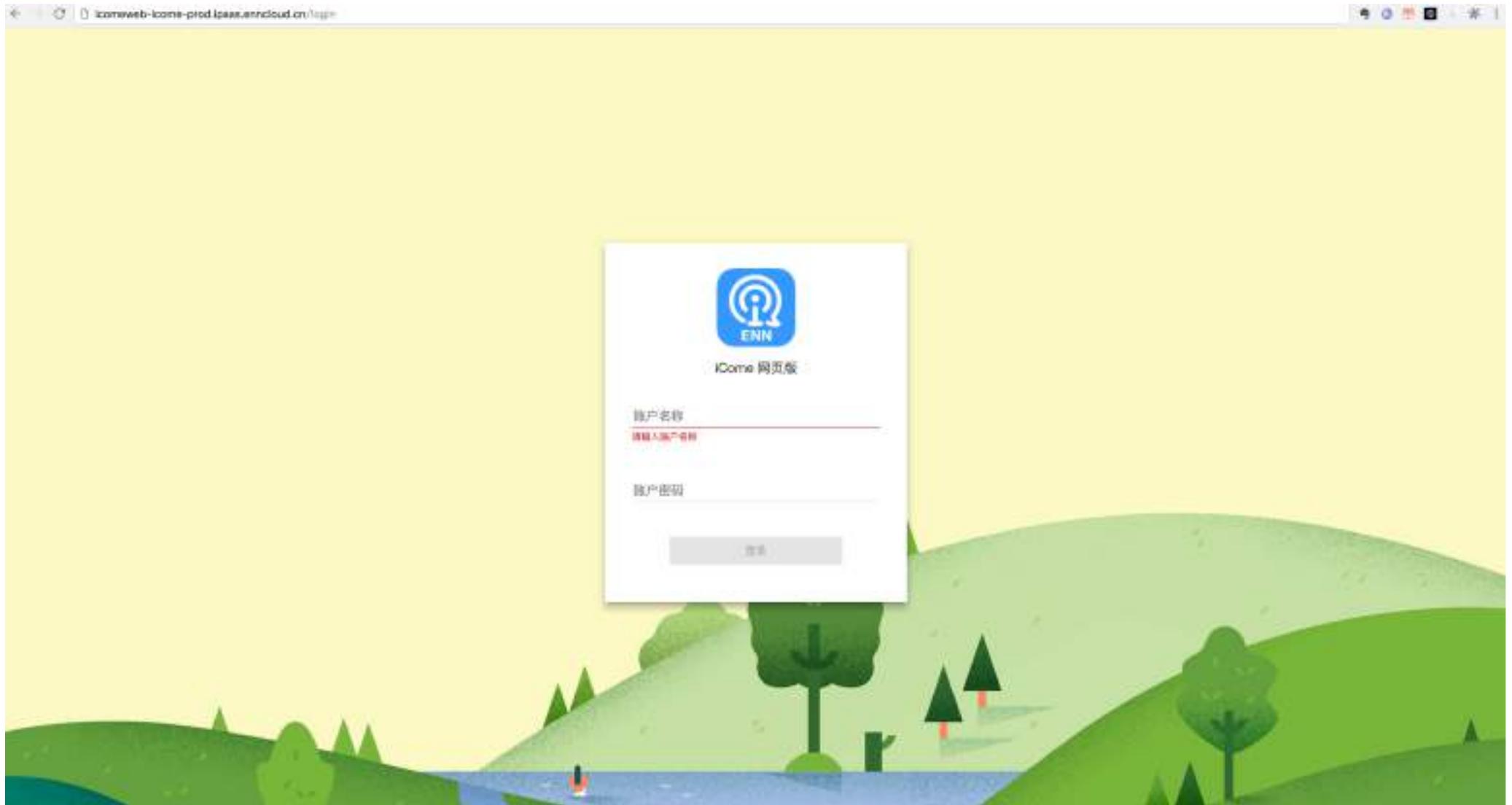


网络设计平台



客户案例

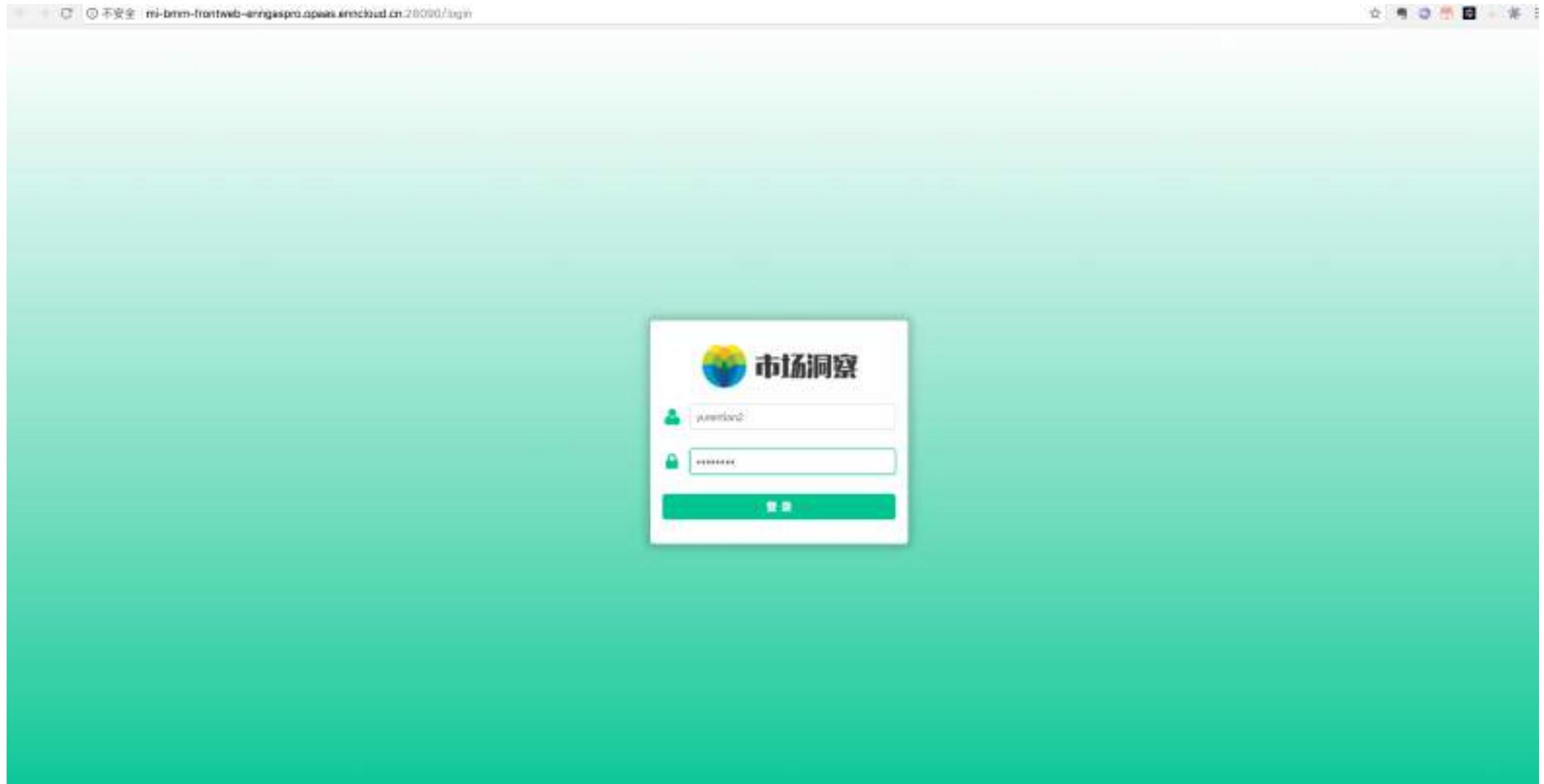
icome



客户案例



市场洞察



欢迎加入我们





Q&A

Thank you for your time

主办：

