

微博 Docker Cloud Platform(DCP) 实现分钟级服务成倍扩容

——云栖社区在线实时培训第九期

DCP整体架构

不可变基础设施

弹性调度

编排与服务发现

Docker Cloud Platform (DCP)

容器化的混合云弹性调度平台

以最低成本实现弹性能力

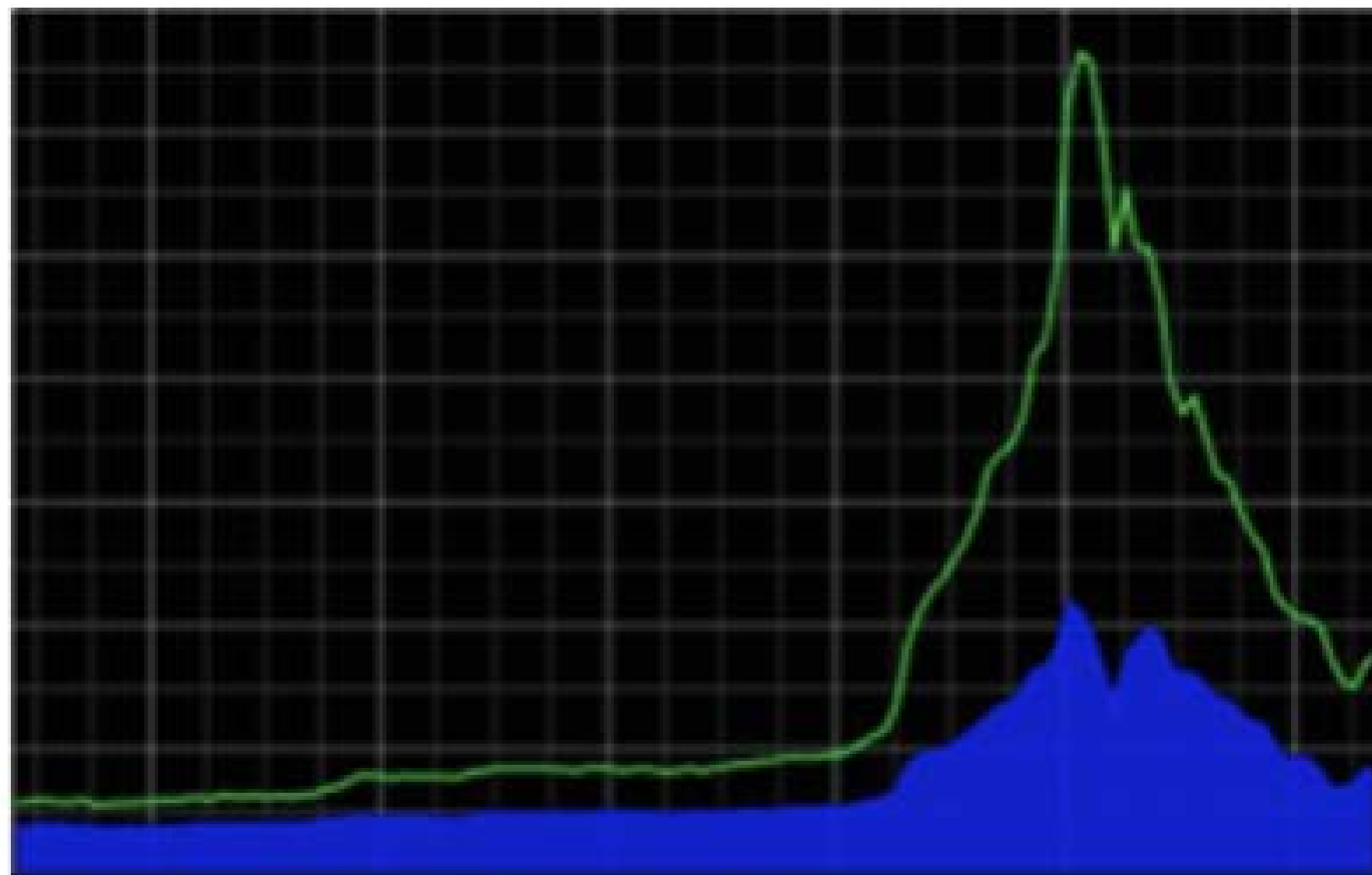


图1： 春晚feed业务

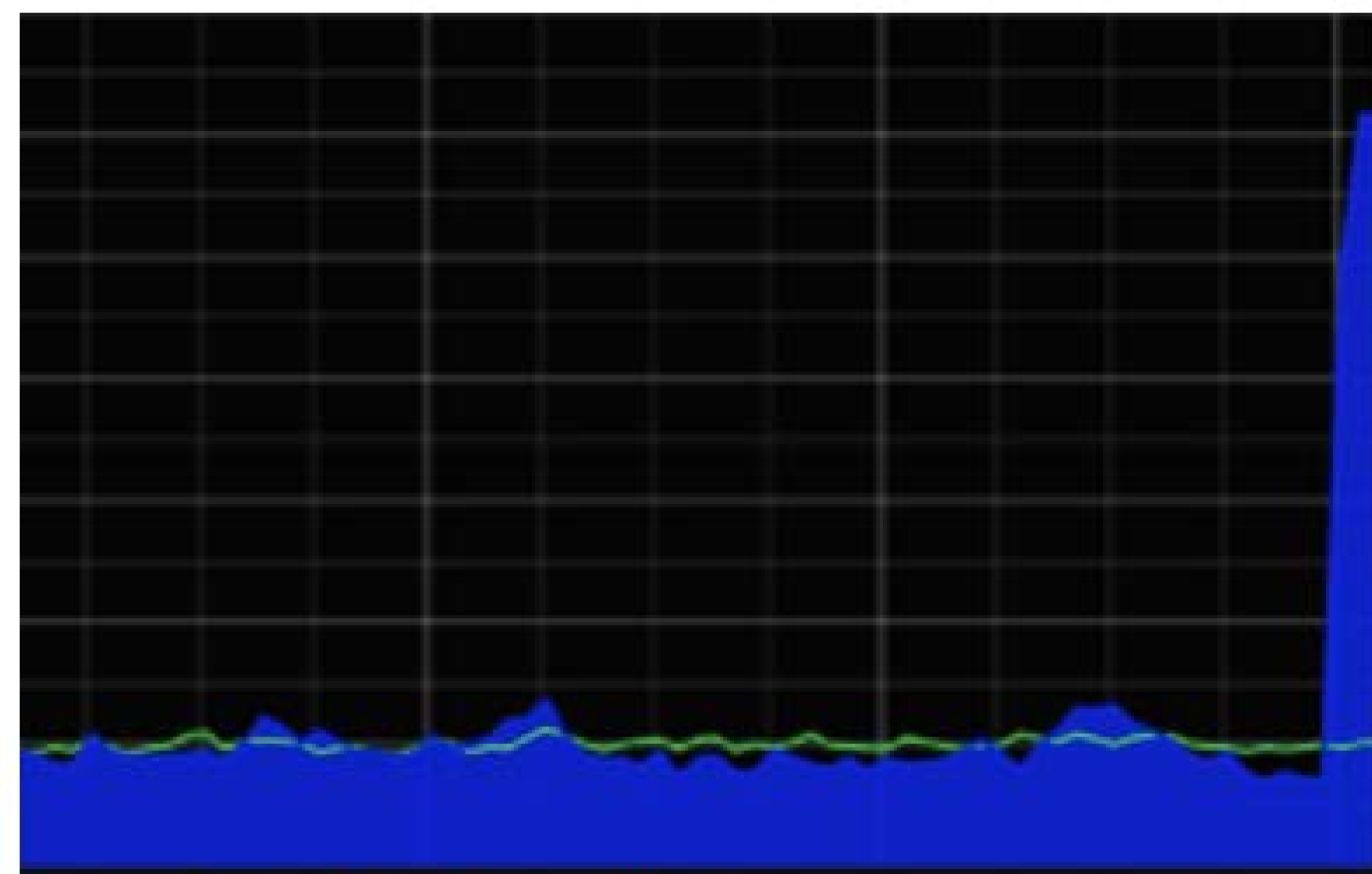
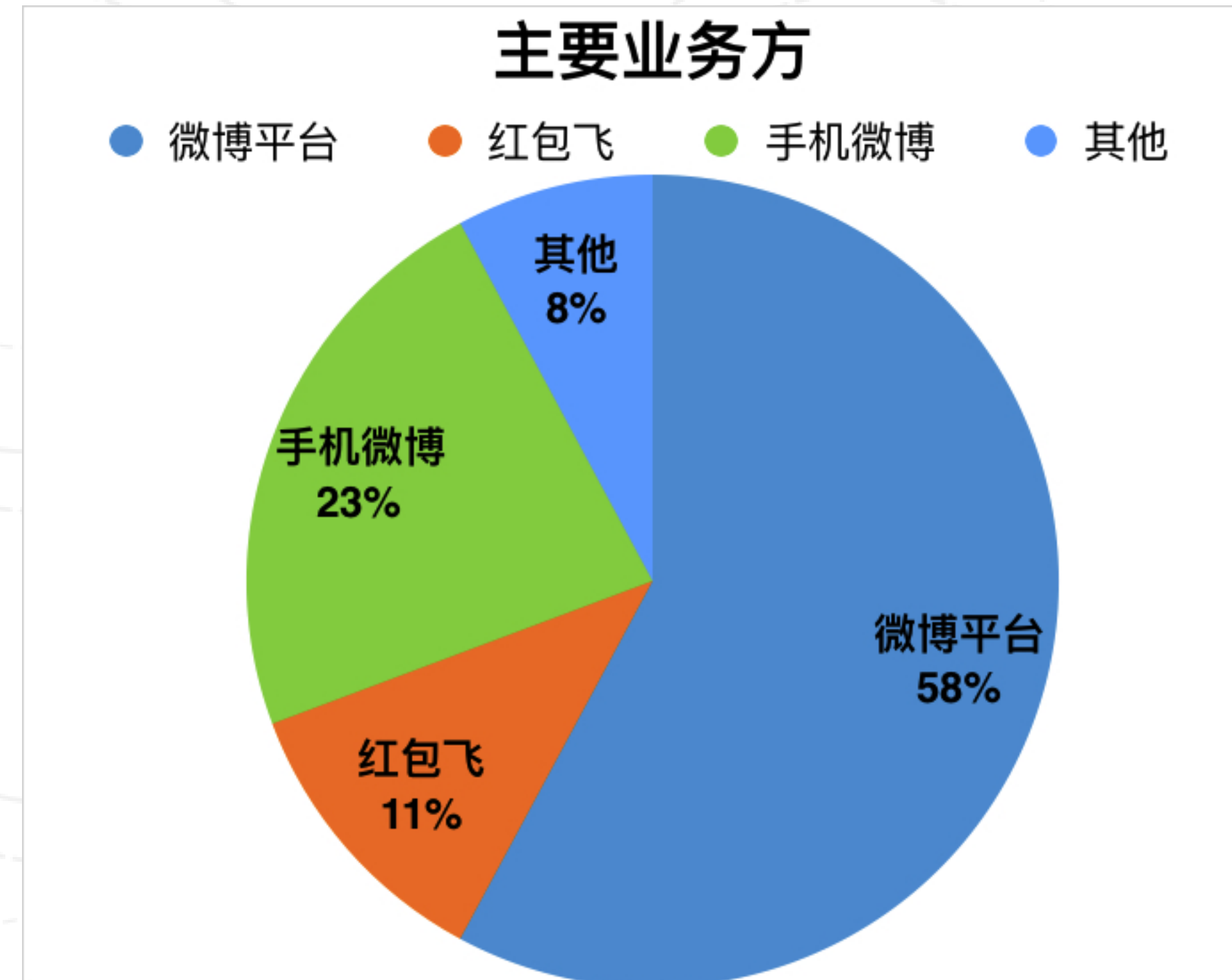


图2： 红包飞业务

DCP系统功能



Docker Cloud Platform(DCP)的“初心”

弹性能力

10分钟内16,000核、25,600GB计算资源弹性能力。

节约成本

2016年三节总体成本不超过2015年，且未来可大幅降低单位成本。

平台化

拉通不同语言、运行环境差异，向各业务系统提供标准的弹性能力。



公共云+专有云

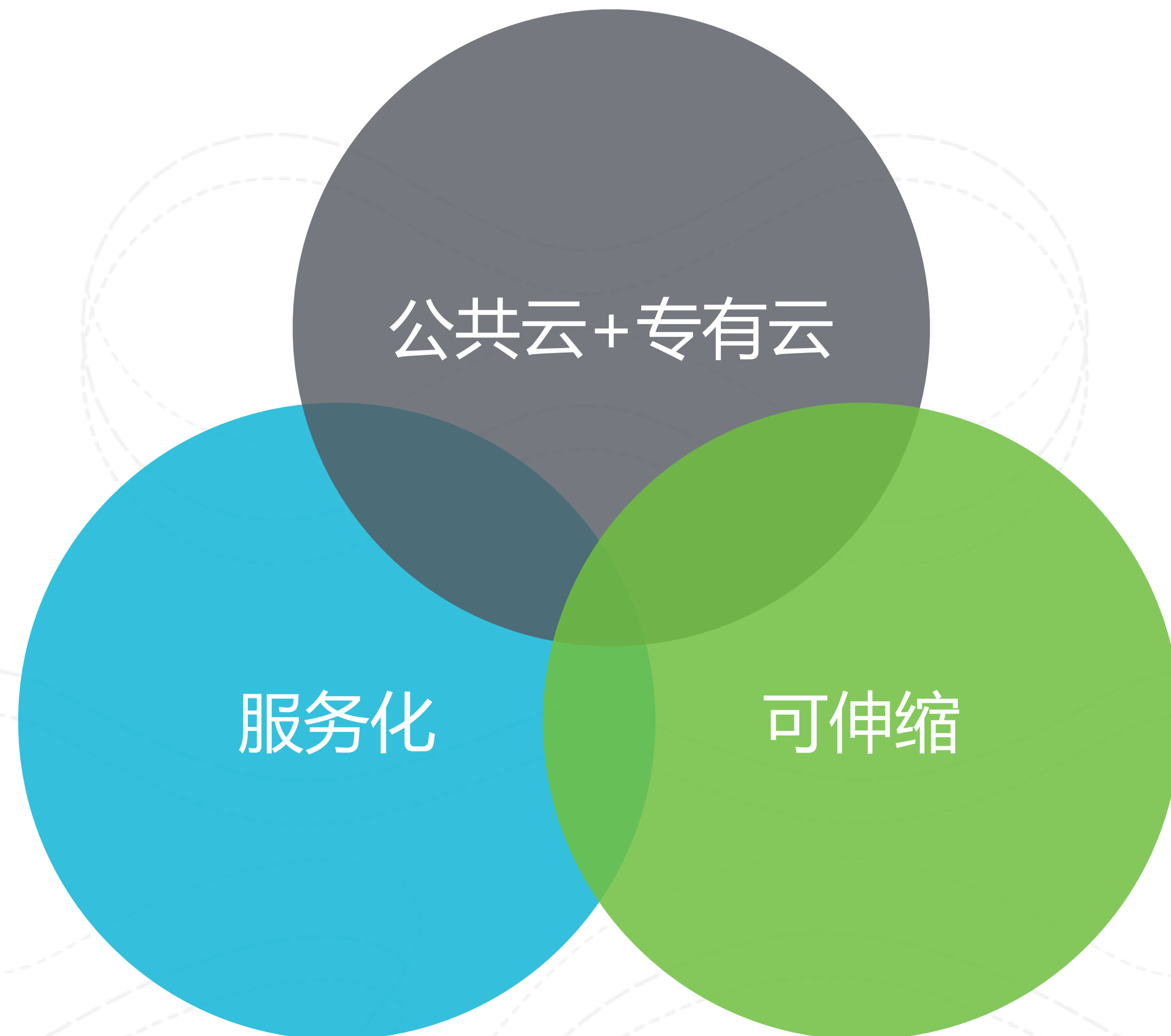
利用公有云按需付费的特点，降低单位成本。业务在公有云与专有云间无差异化部署。

服务化

系统各组件间仅通过Restful API进行通信。各组件应具有良好的扩展性，实现机制可插拔。

可伸缩

各系统组件可根据管理集群的规模，实现自身的自动伸缩。各组件应无状态、无单点，运维操作自动化。



DCP架构分层

不可变基础设施

1. 容器化
2. 系统初始化
3. 镜像分发
4. 带宽监控

弹性调度

1. 业务跨云调度
2. 调度机制
3. 容量评估

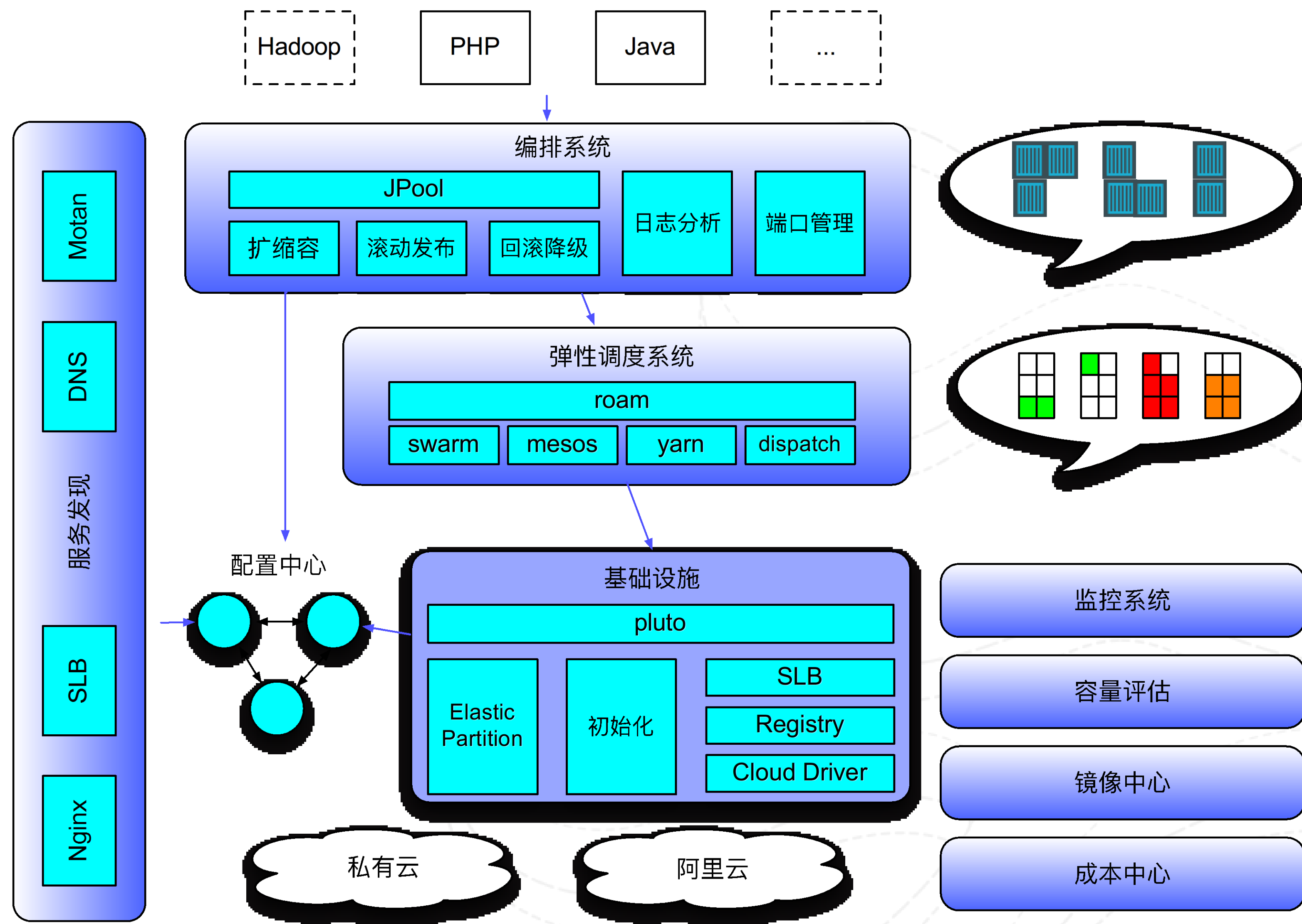
服务发现

1. 配置中心
2. 负载均衡
3. RPC
4. 资源

容器编排

1. 自动扩缩容
2. 监控报警

DCP整体架构



DCP整体架构

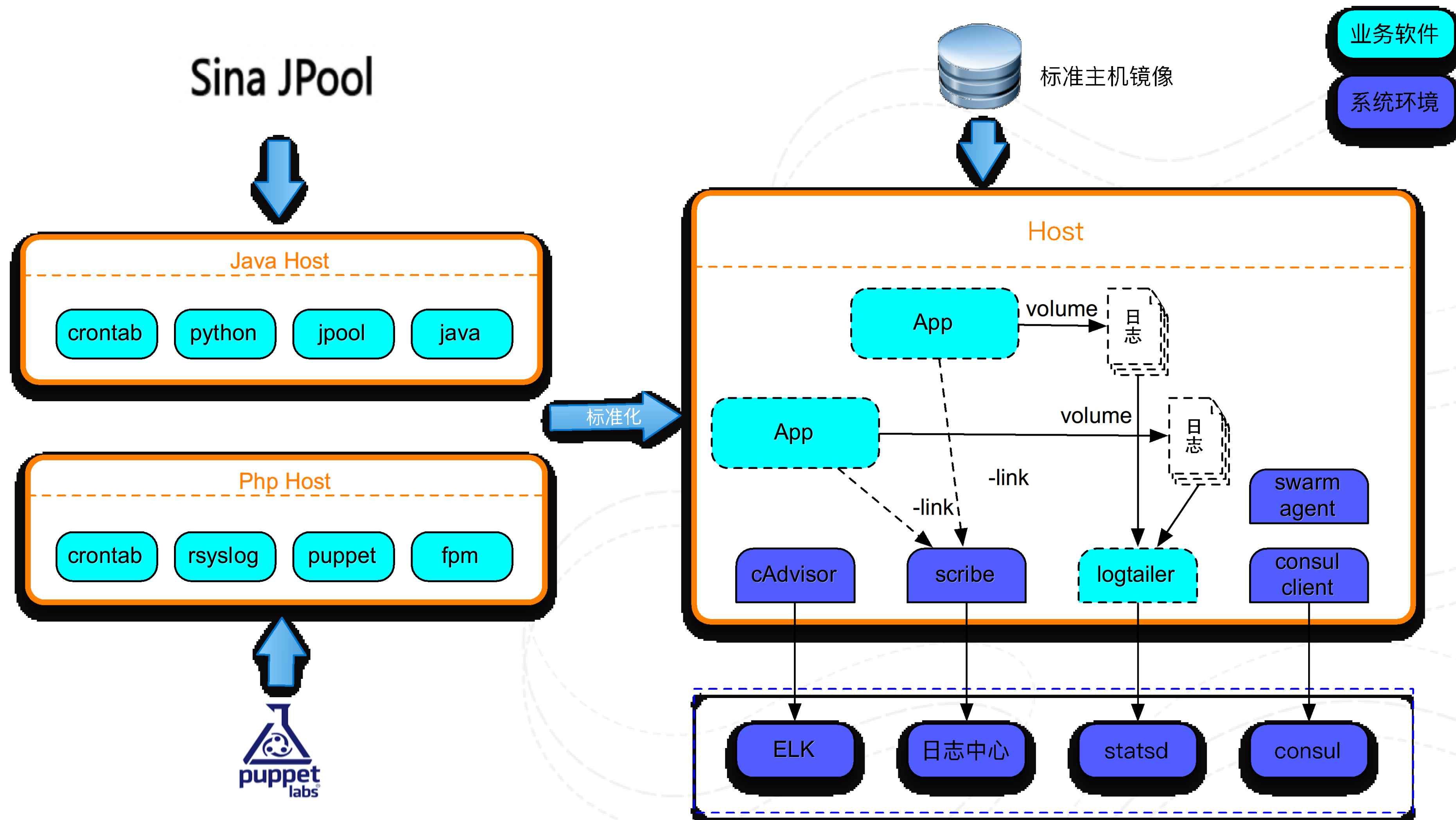
不可变基础设施

弹性调度

编排与服务发现

容器化

所有业务均以Docker镜像发布，不依赖、不修改宿主运行环境



Java		php	
cAdvisor 0.7.1.fix			
Mesos 0.25			
Swarm 1.0.0			
Consul 0.5.2			
Docker 1.6.2	Registry v2	Daemon Wrapper	Registry v1
devicemapper-direct-lvm		Docker 1.3.2	
		devicemapper-loop-lvm	
CentOS 7.1.1503/3.10.0-229.el7.x86_64		CentOS 6.6/2.6.32	

初始化

性能优化

流水线

各Stage尽可能并行化，目前可实现4分钟内初始化上百台能力。

异步化

通过Ansible的Callback机制，把耗时操作异步化。

水平扩展

实现Ansible自动伸缩功能，根据初始化需求规模自动伸缩，可支持分钟内千级别初始化能力。



初始化

参考Ansible Tower，实现任务调度及执行状态

The screenshot displays the Ansible Tower configuration management interface. A modal window titled '详细' (Details) is open, showing the configuration for a task named 'hongbao | service'. The configuration includes the following details:

- module_name:** service
- module_vars:**
 - delegate_to: null
 - failed_when: null
 - role_path: /etc/ansible/roles/hongbao
 - playbook_dir: /etc/ansible
 - role_uuid: f78d5197-87bc-4716-ae52-e48d61245e96
 - always_run: false
 - changed_when: null
 - original_file: /etc/ansible/roles/hongbao/tasks/init.yml
 - role_names: {
 - facts
 - hongbao}
 - ignore_errors: false
 - register: null
- module_args:** name=gmond state=started

The modal also shows a '关闭' (Close) button. In the background, a table lists the execution details of the task, including the action name, status, and execution time.

耗时(秒)	开始时间	结束时间
	2016-02-07 18:00:30	2016-02-07 18:00:30
16	2016-02-07 18:00:03	2016-02-07 18:00:29
	2016-02-07 18:00:02	2016-02-07 18:00:03
	2016-02-07 18:00:03	2016-02-07 18:00:03
1	2016-02-07 18:00:01	2016-02-07 18:00:02
1	2016-02-07 18:00:00	2016-02-07 18:00:01
0	2016-02-07 18:00:00	2016-02-07 18:00:00
0	2016-02-07 18:00:00	2016-02-07 18:00:00

镜像分发

解决大批量设备创建，Docker镜像分发性能问题

降低镜像大小

- 不变基础镜像
- 本地load

反向缓存

- 在IDC节点缓存
- 最新镜像自动同步

自动伸缩

- 分发网络可自动伸缩

P2P分发

- 超过千级别规模，利用P2P进行分发

混合云网络架构

VPC

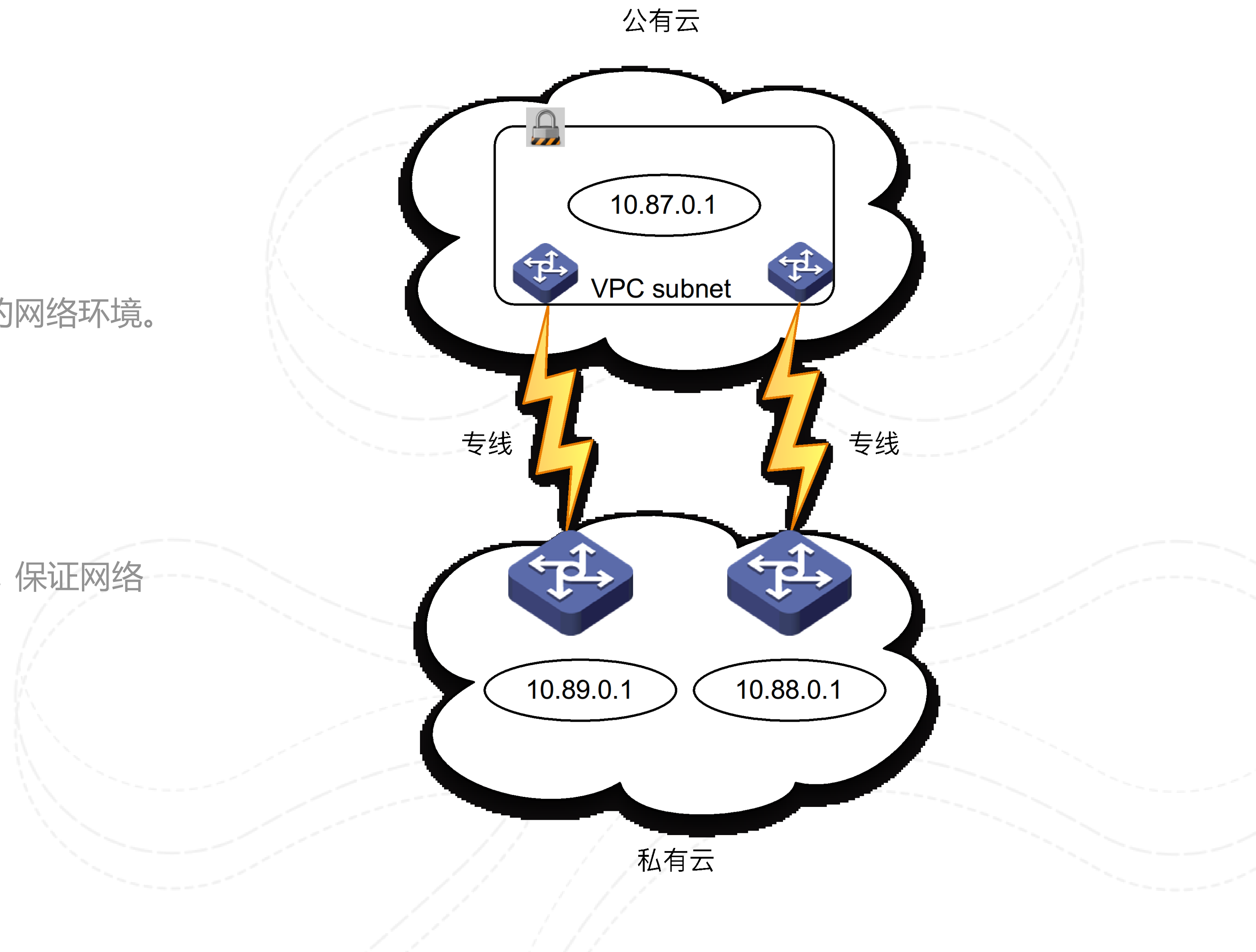
利用公有云的VPC服务，构建与专有云内网一致的网络环境。

网络高可用

采用多专线确保网络高可用

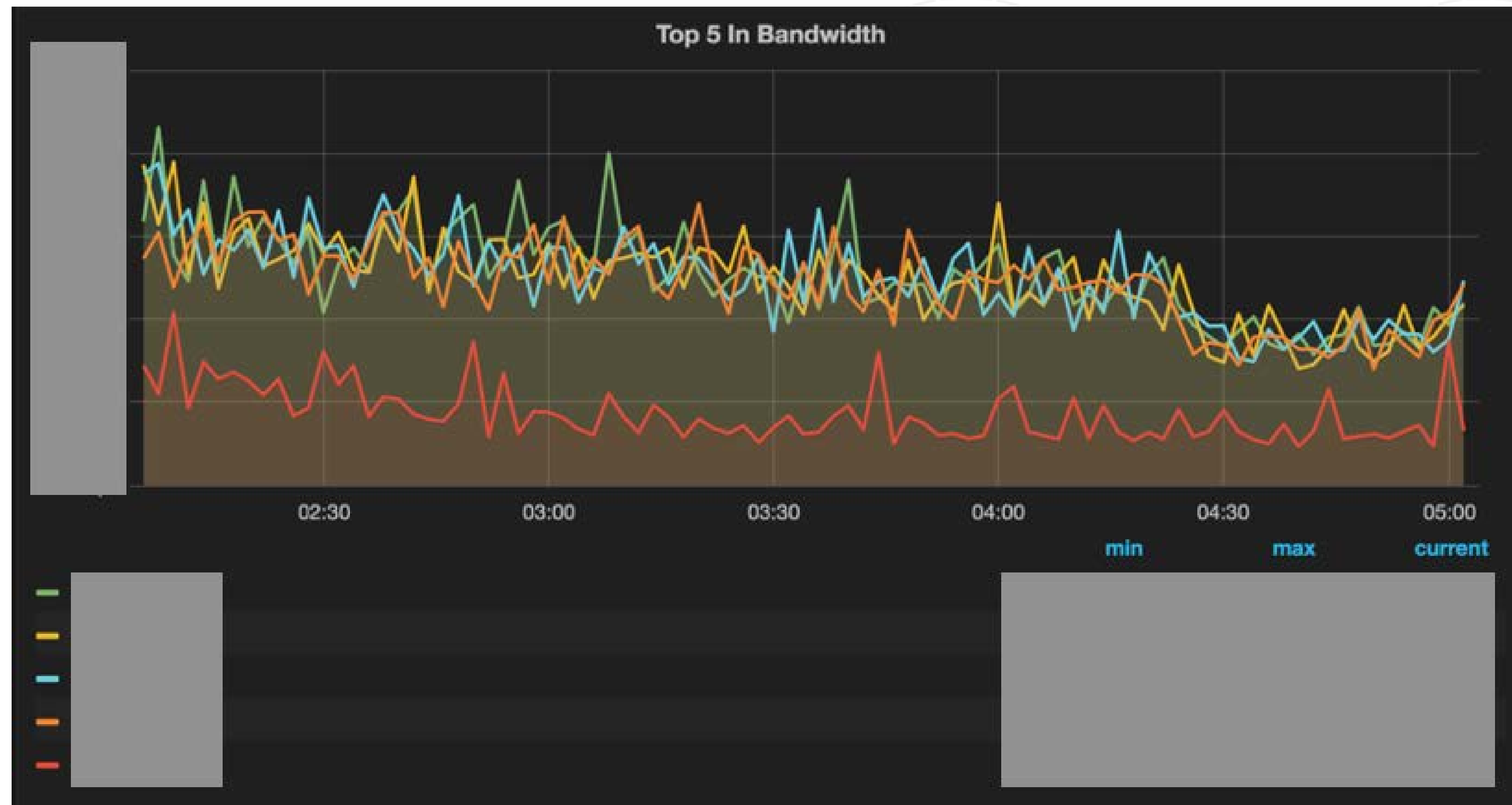
路由

在核心交换上配置路由转发规则，采用就近原则，保证网络低延迟。当网络故障时，自动启用备份路由。



带宽监控

跨云专线监控，细化到IP级别，自动聚合到产品线



DCP整体架构

不可变基础设施

弹性调度

编排与容器发现

业务跨云弹性调度

业务以最小规模在阿里云上预付费部署，所有弹性需求均采用按需方式

最小规模部署

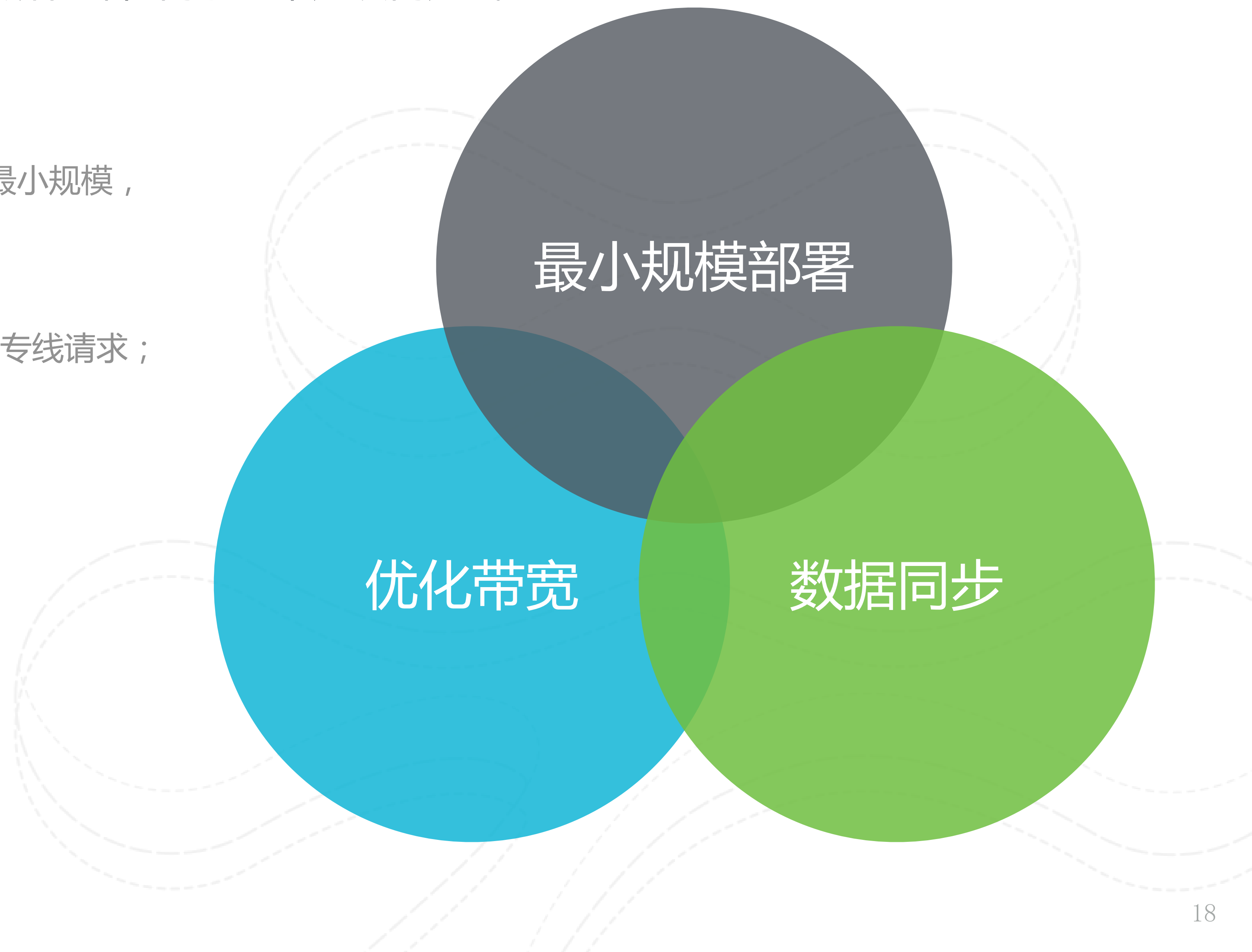
在公有云上通过预付费方式，常态化部署业务的最小规模，并提供在线服务。

优化带宽

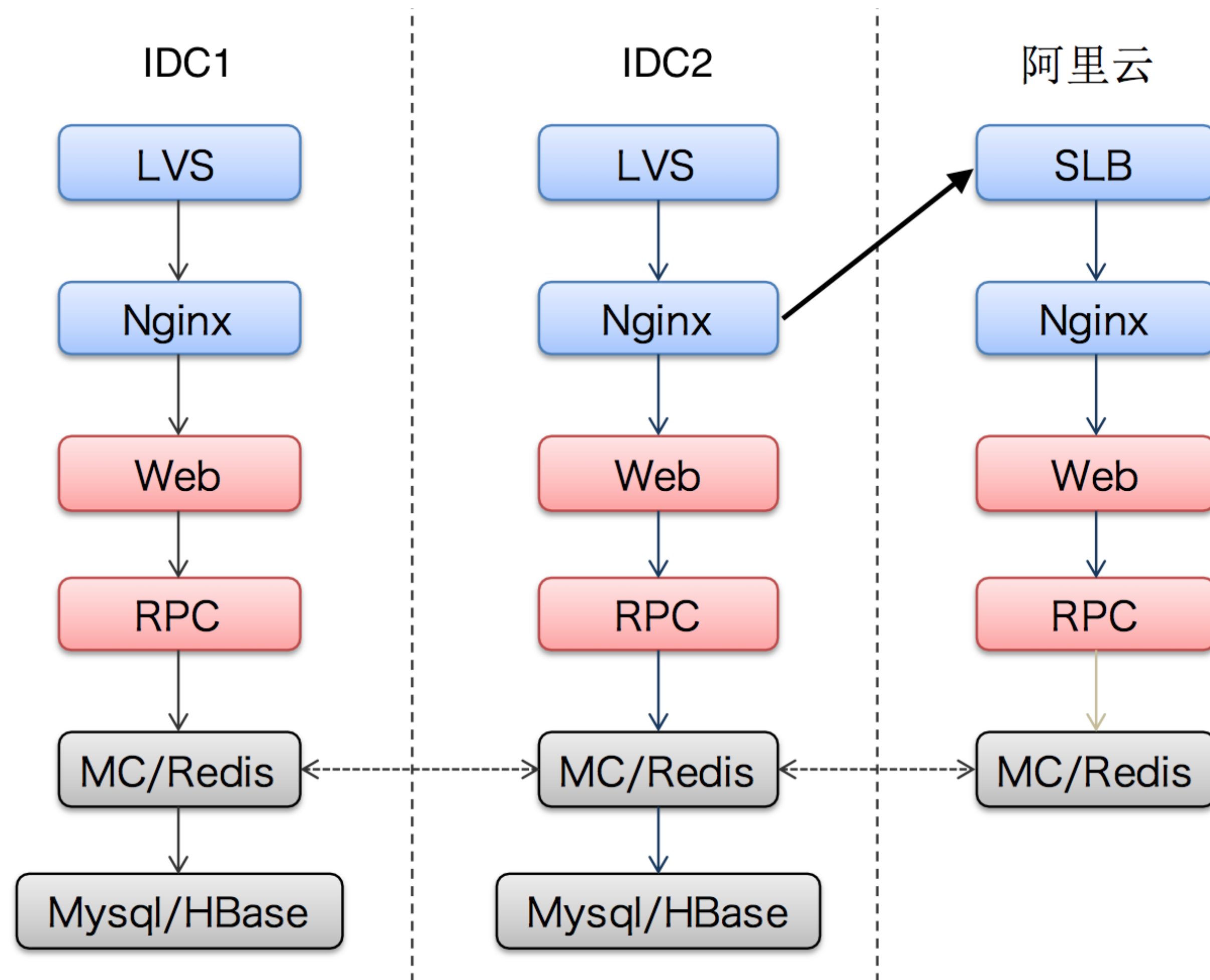
优化跨专线调用：一、资源Local化，尽量减少跨专线请求；
二、跨专线请求开启压缩；

数据同步

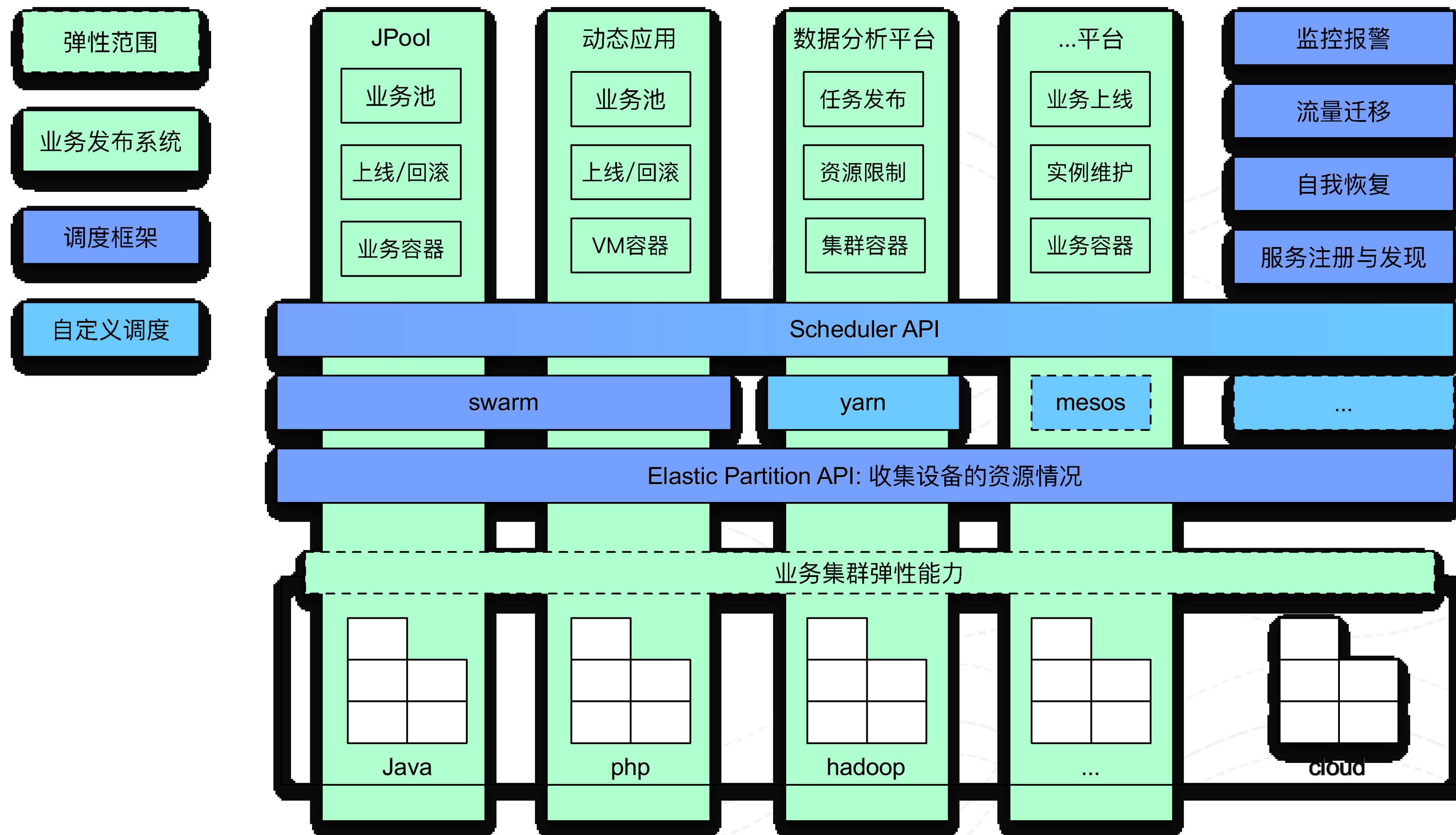
基于消息的缓存同步机制。



业务混合云部署形态



弹性调度机制



容量评估



DCP整体架构

不可变基础设施

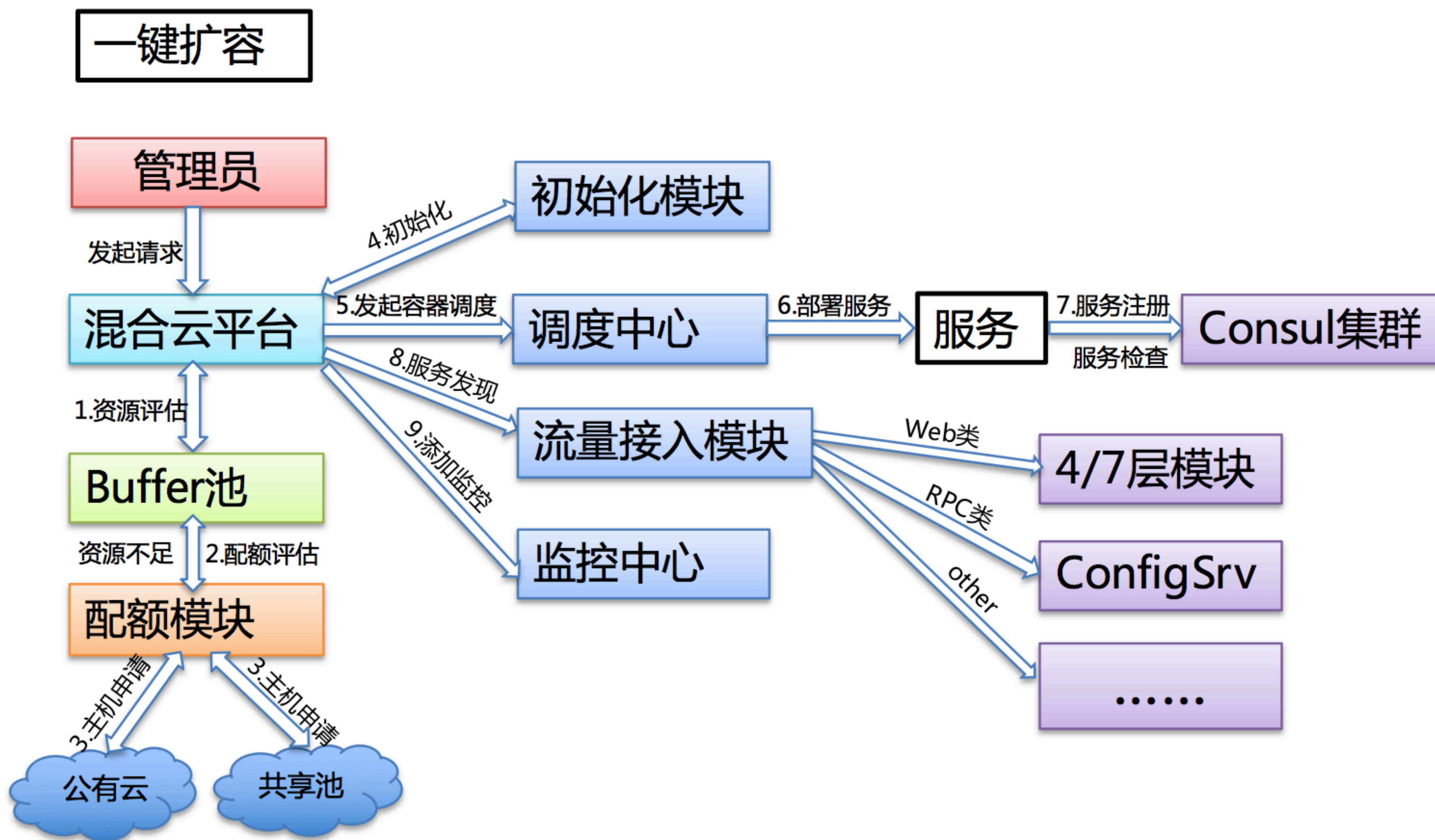
弹性调度

编排与容器发现

业务容器编排



自动扩缩容



基于Nginx服务发现的问题

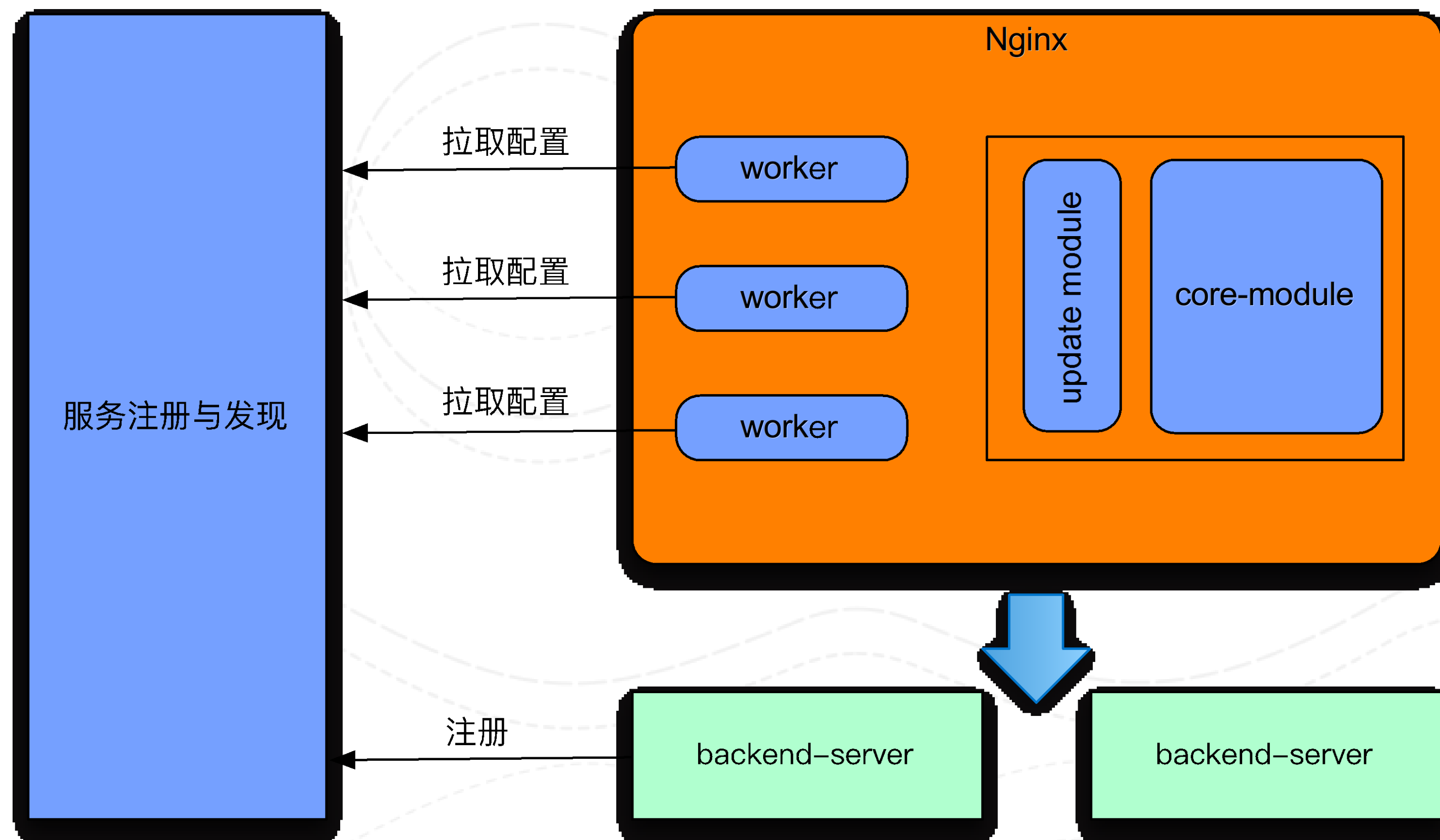
基于nginx reload来实现服务发现，管理成本高、不支持并发注册、会带来性能损耗。

Nginx Upsync

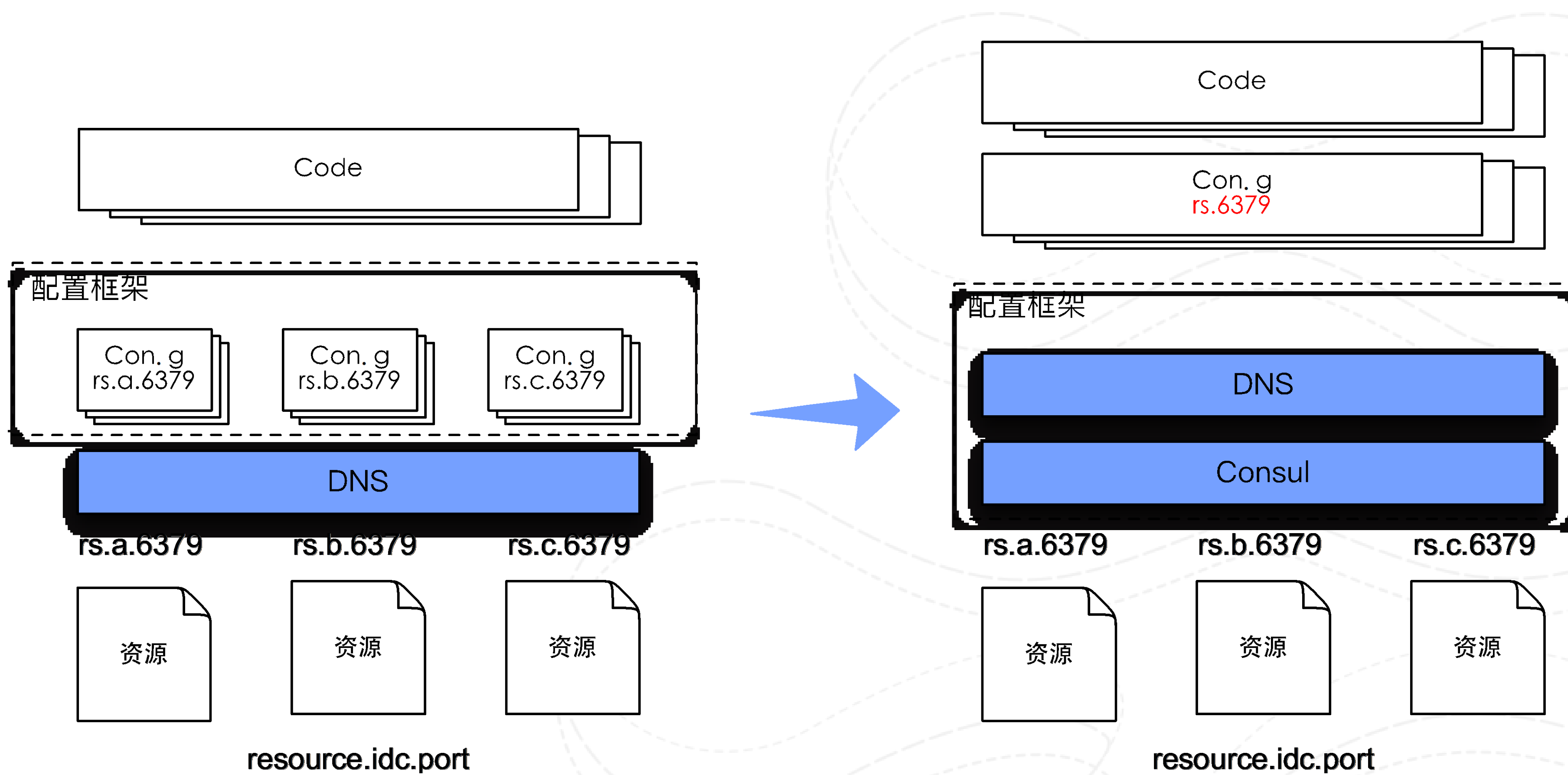
支持基于Consul自动服务发现，解决reload的性能问题

项目地址：

<https://github.com/weibocom/nginx-upsync-module>



资源的自动发现



1000+

共弹性调度超过1000个ECS实例

60% QPS < 10分钟

分流60%核心流量

以超过1亿/天的数量新增

Thanks!



yq.aliyun.com

云栖社区，我们的IT江湖