



# All on Serverless

计缘

阿里云智能云原生应用平台

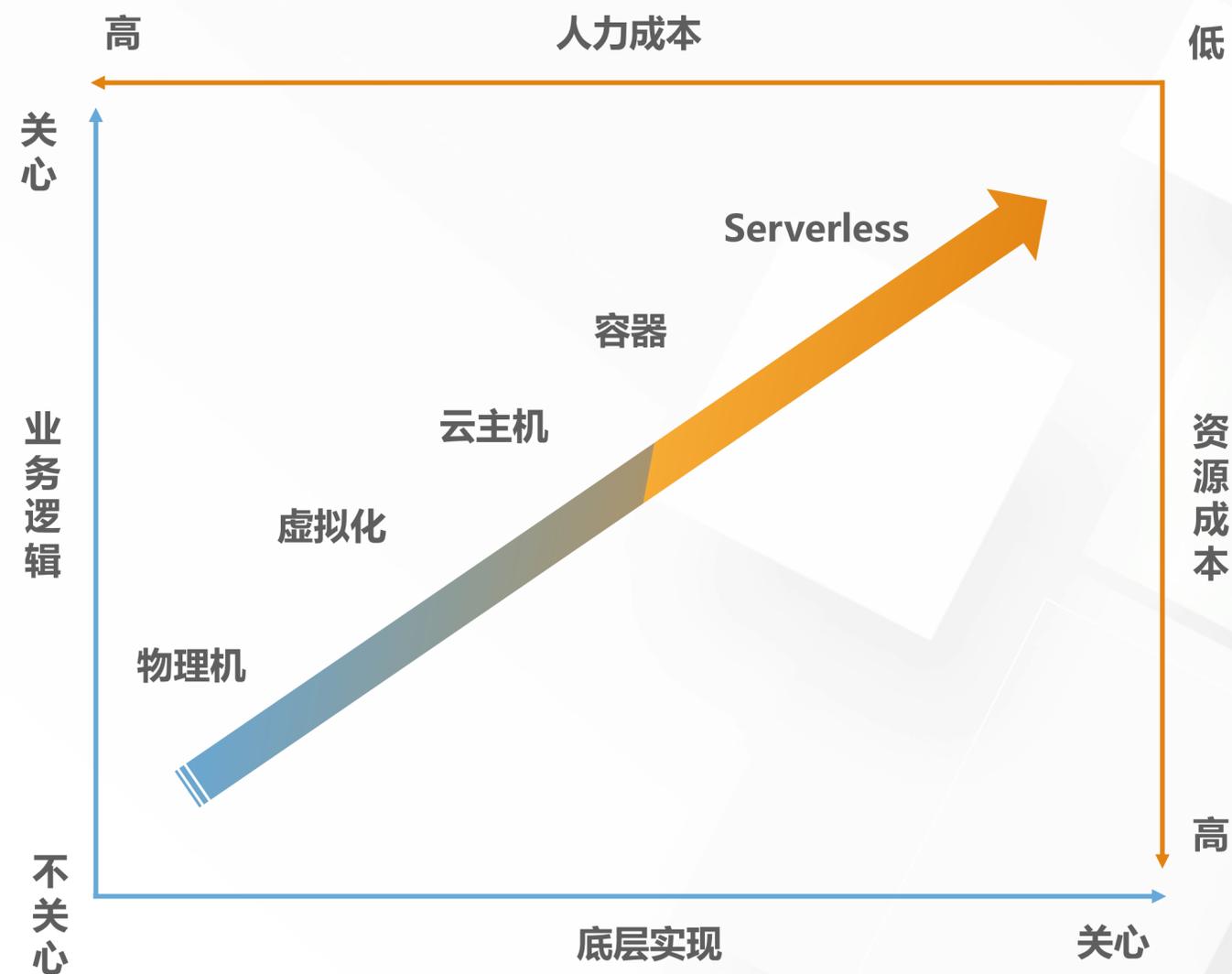


# 01 Serverless 概念解读

# 计算的进化趋势

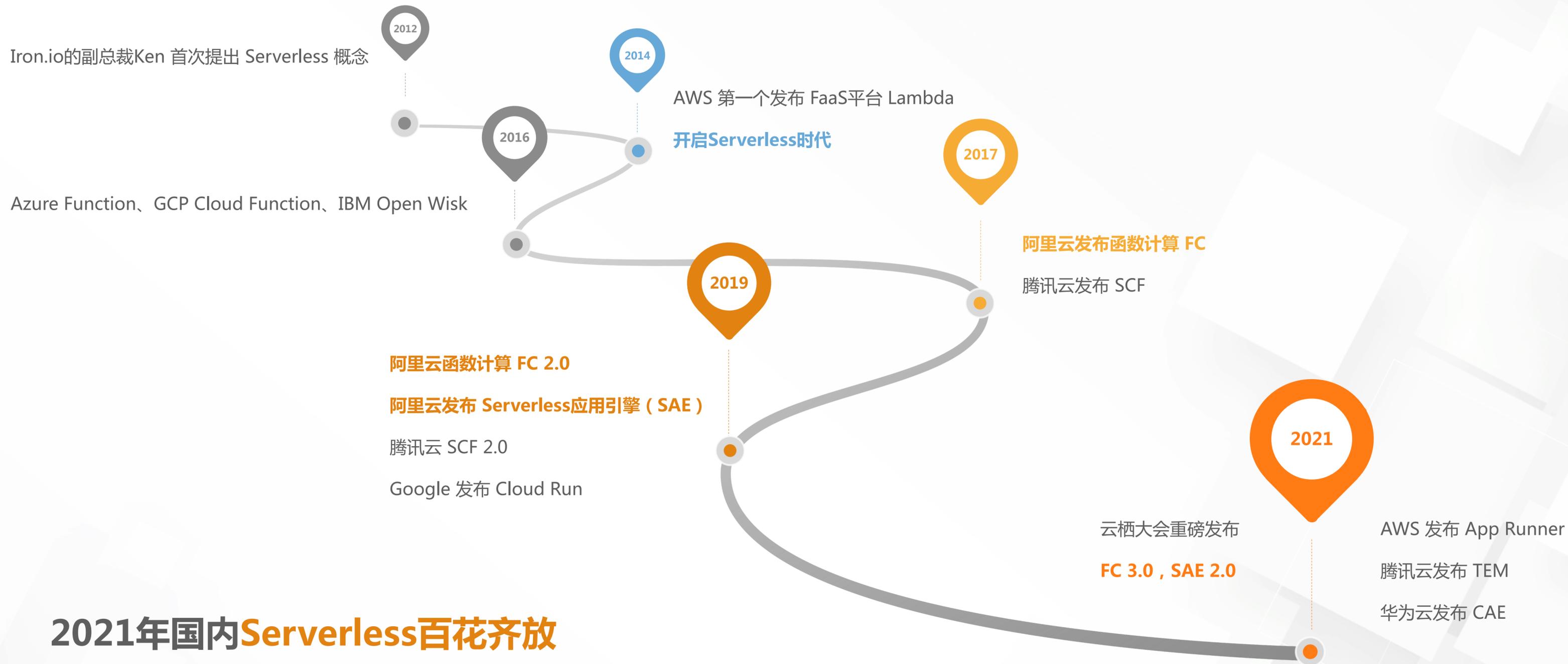


Serverless 是云计算发展的必然结果，就像人类的进化历程，代表着生产力的解放，Serverless 极大提升了客户使用云的效率。



Serverless 是云原生技术发展的高级阶段，Serverless 相对于 Serverful，对用户强调 No Server（Serverless 并不是说没有服务器，只是业务人员无需关注服务器，代码仍然是运行在真实存在的服务器之上），计算资源的维护交给了云厂商，用户只需要聚焦业务逻辑代码、按实际使用付费。

# Serverless发展历程

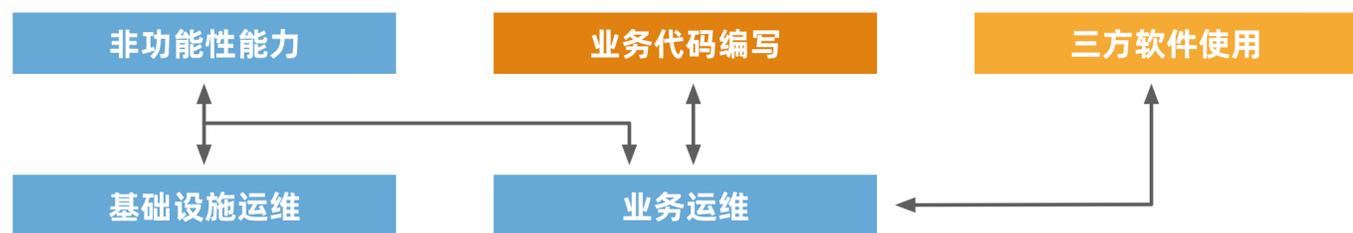
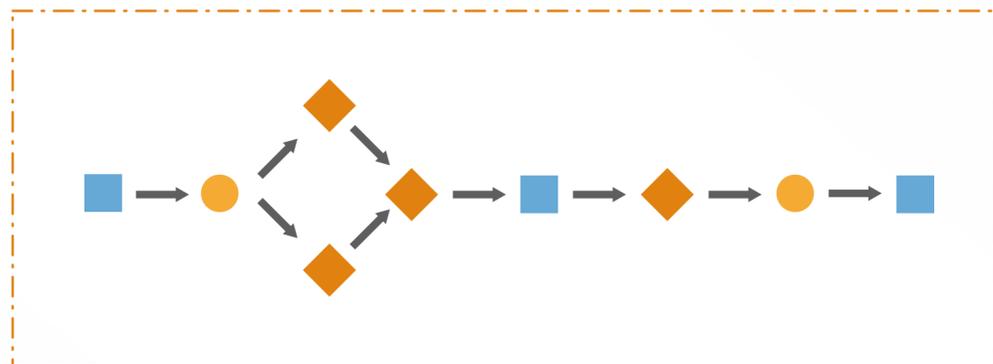


2021年国内**Serverless**百花齐放

# Serverless 架构优势



## 传统架构



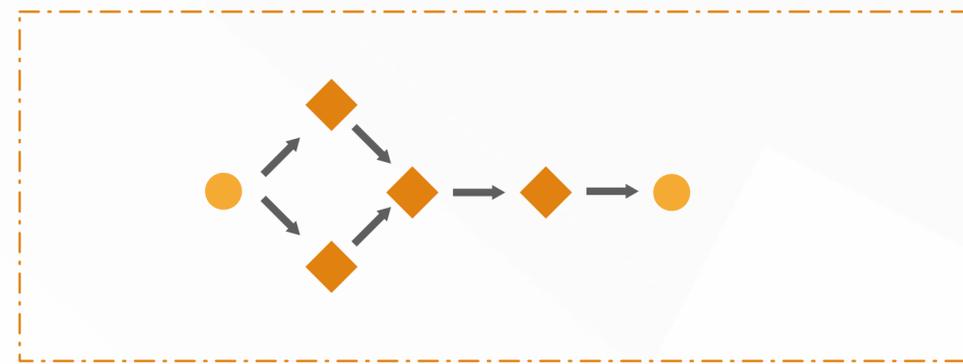
开发人员



运维人员



## Serverless 架构



开发人员



# Serverless 带给客户的价值

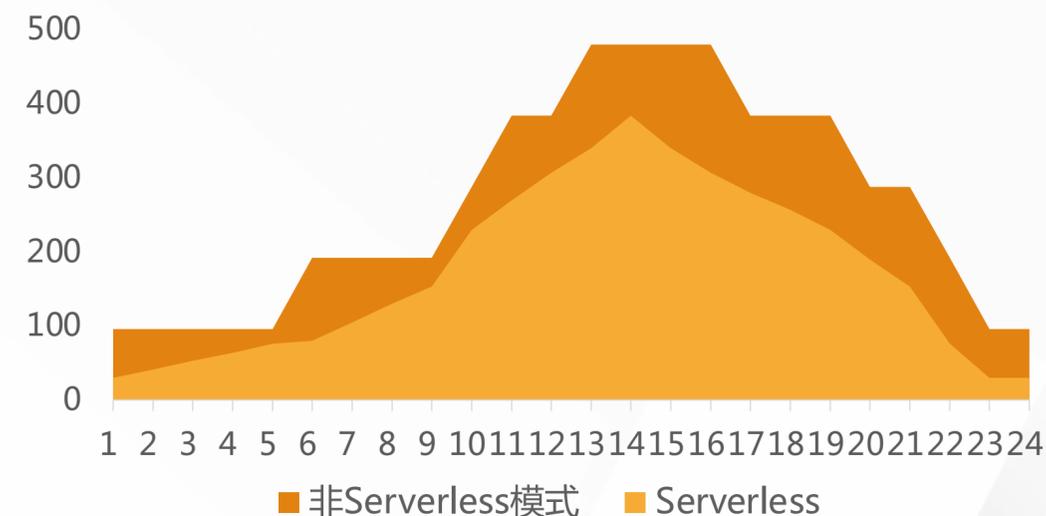
## 要算力，也要弹性

Serverless VS 非 Serverless 模式弹性效率对比



## 提高资源利用率

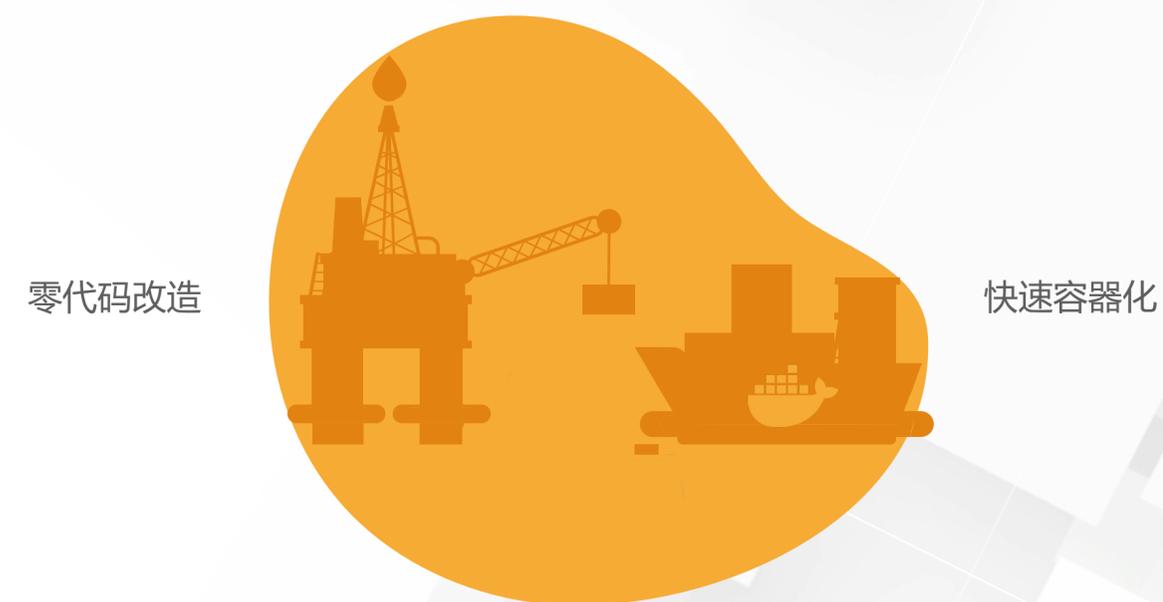
Serverless VS 非 Serverless 模式弹性成本对比



## 要人少，也要高效



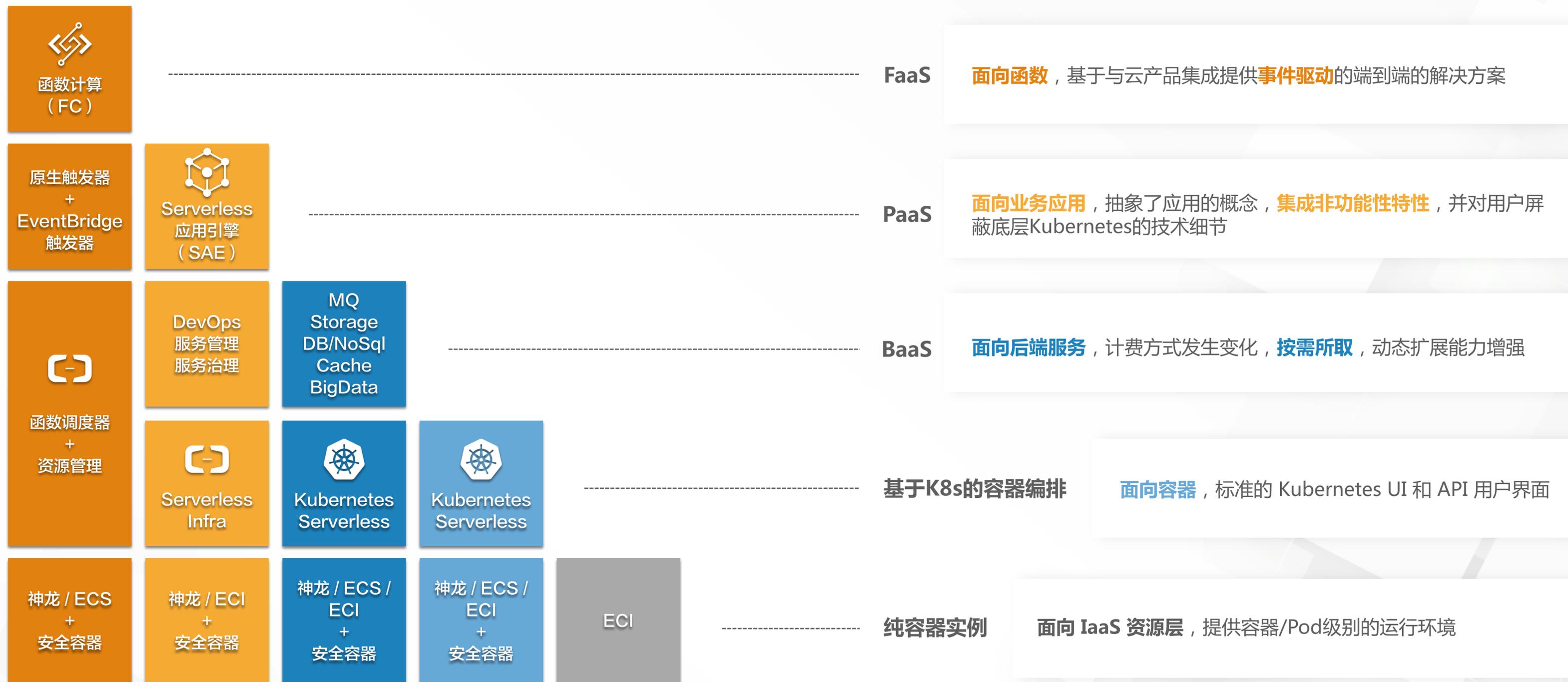
## 要简单，也要容器化



# Serverless 的不同形态



Serverless 有FaaS、PaaS、BaaS三种基本的形态，分别应对不同的场景，也有不同的实现方式。



# 02 Serverless FaaS 函数计算 FC

# 函数计算 FC 是什么



函数计算是**事件驱动的全托管计算服务**。使用函数计算，客户**无需采购与管理**服务器等基础设施，只需编写并上传代码。函数计算为客户提供了从128MB到3GB内存规格的**弹性实例**和从4GB到32GB内存规格的**性能实例**，可根据业务需求灵活选择资源规格。支持弹性实例**百毫秒级弹缩**，性能实例**秒级弹缩**的弹性能力。支持**近20+**触发器类型满足各类业务场景，并且提供完善的**日志、可观测、告警**能力。为客户提供**高可靠、高可用、高性价比**的计算服务。



# 函数计算 FC 是云产品的连接器



集成云生态能力：SDK/API

**函数计算 + 阿里云 = Serverless 新生态**

被云生态所集成：事件驱动callback（独有）

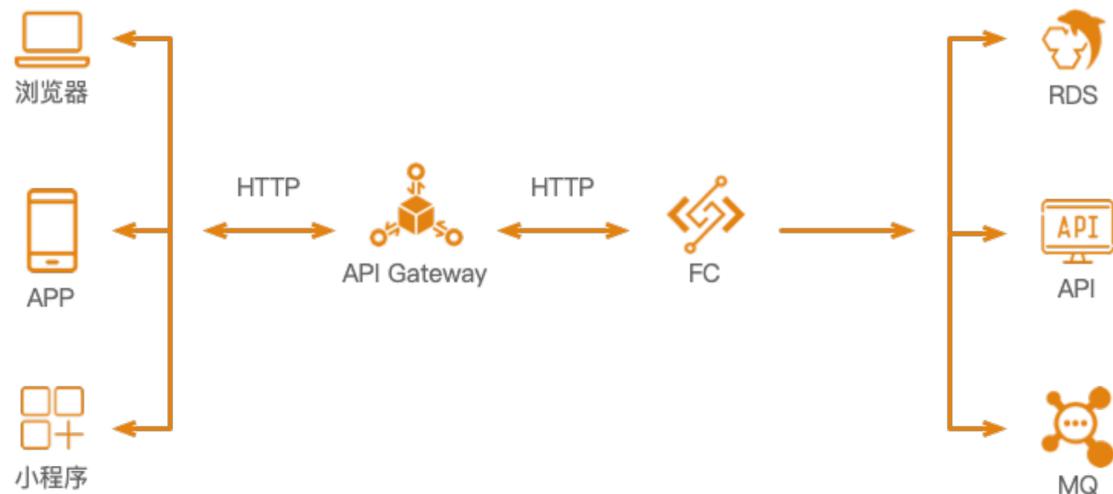
# 函数计算 FC 应用场景



HTTP应用通用方案

## Serverless HTTP应用场景

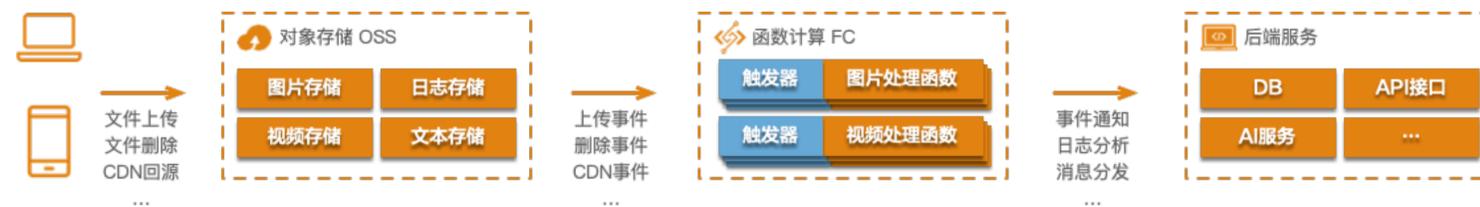
结合 API 网关，提供小程序，H5，WEB服务，REST API 等通用 HTTP 解决方案



## Serverless 音视频、图片、文本处理场景

- 支持一键配置对象存储事件监听，即可实时消费 OSS 事件，如自定义图片处理、ZIP文件解压缩、CDN自动刷新、数据实时同步等
- 自定义处理逻辑，支持多种编程语言，实现诸如文本分析、MapReduce等
- 毫秒级弹性伸缩，轻松应对负载的波峰波谷，实际案例支持10W+级QPS

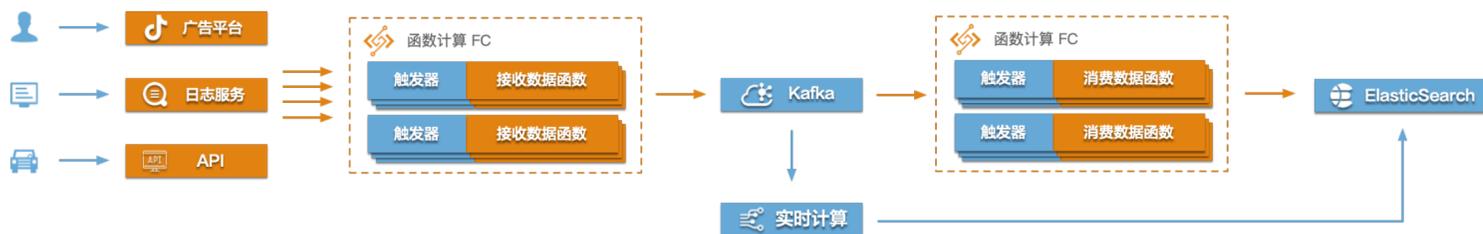
内容处理通用方案



大数据场景通用方案

## Serverless 大数据ETL场景

- 支持一键配置消息中间件数据消费，无需额外构建消费者，如 Kafka 消息备份、日志清洗、聊天消息分发等
- 支持自定义处理逻辑，承载多种多样的业务需求
- 支持高可用，支持配置错误重试



## Serverless 游戏场景

- 将游戏场景中通用的业务能力进行抽象，从游戏主服中进行剥离，沉淀到统一服务层，最大化的提升资源利用率，减少系统维护工作量和成本
- 函数计算接管强 CPU 密集型的战斗结算场景，稳定应对开服、运营活动时的流量脉冲

Serverless游戏场景

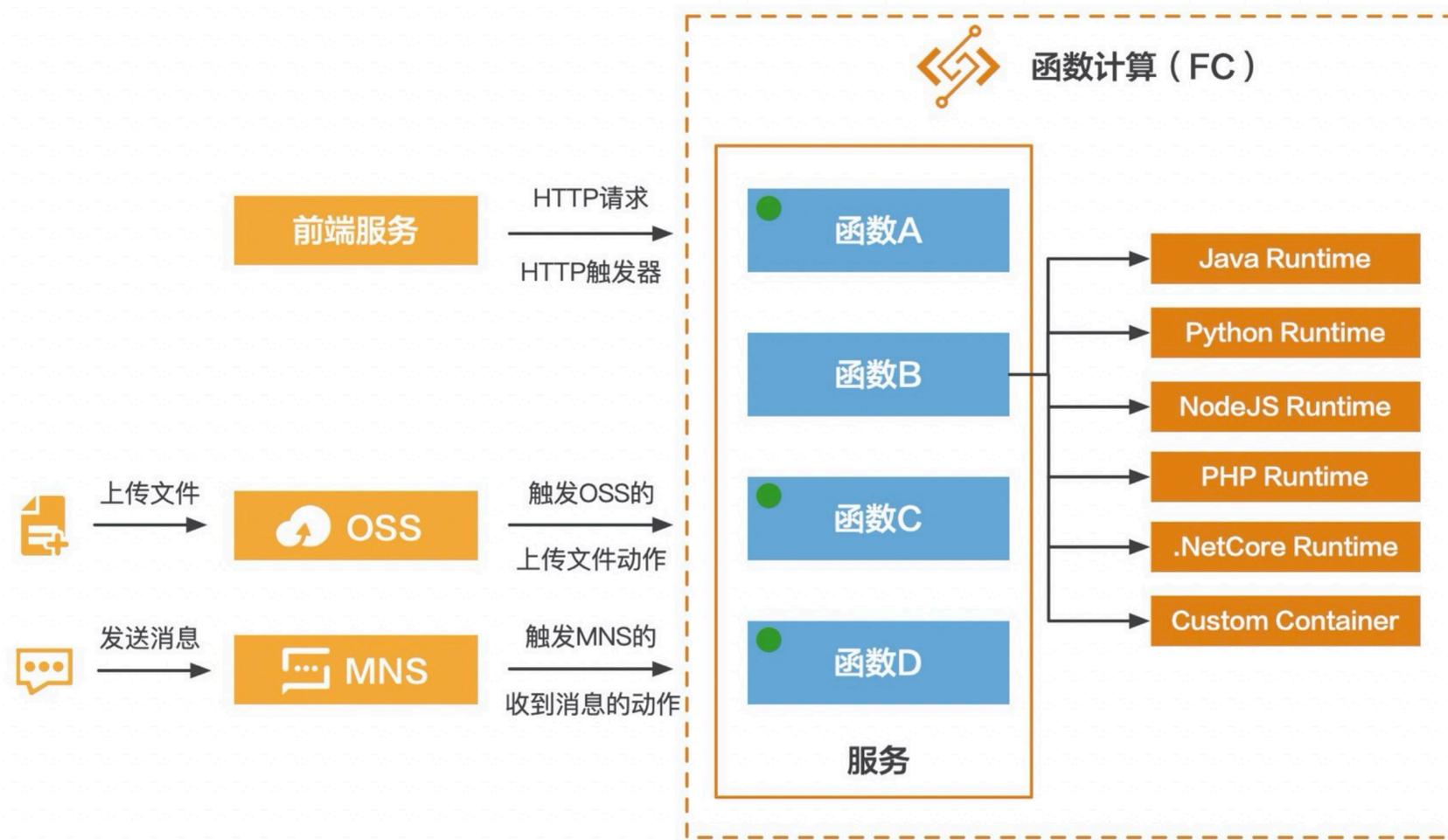


# 多触发器/多语言执行环境

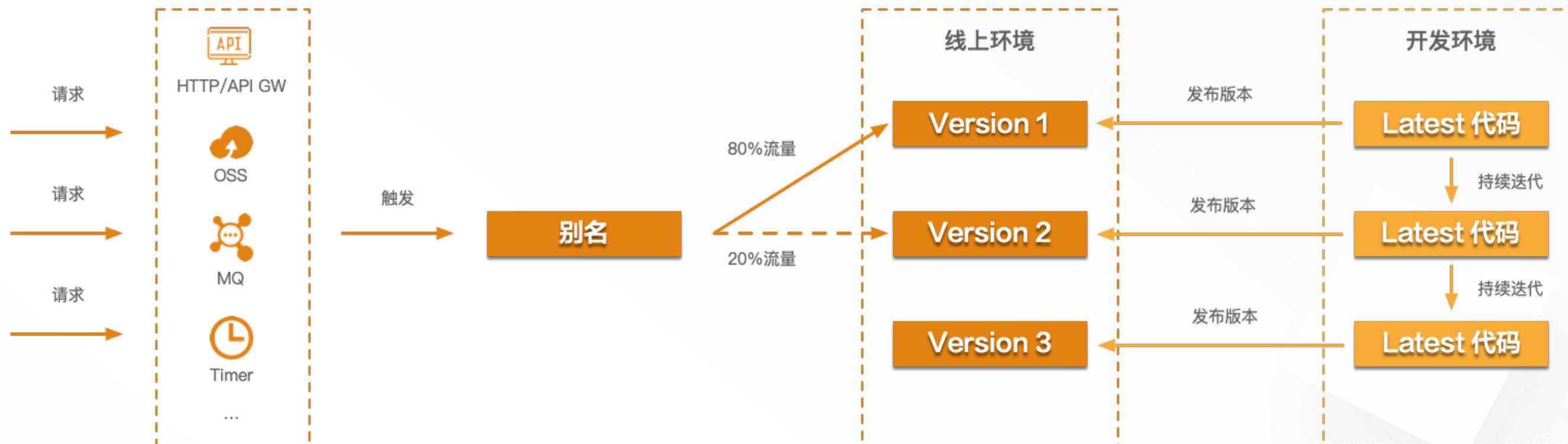


由提供**20+**触发器，涵盖**所有主流开发语言**的Runtime满足各种**现有业务场景**及**创新业务场景**

- 我们最常使用的HTTP请求协议其实就是一种类型的触发器，在FC里称为HTTP触发器，除了HTTP触发器以外，还提供了OSS（对象存储）触发器、SLS（日志服务）触发器、定时触发器、MNS触发器、CDN触发器等等，多大20+个原生触发器。再结合EventBridge，触发器扩展性得到进一步增强。
- FC中的函数可以设置不同开发语言的Runtime来运行不同的编程语言，为了能更加灵活适应多种场景，还提供了使用客户自定义镜像作为Runtime的能力。



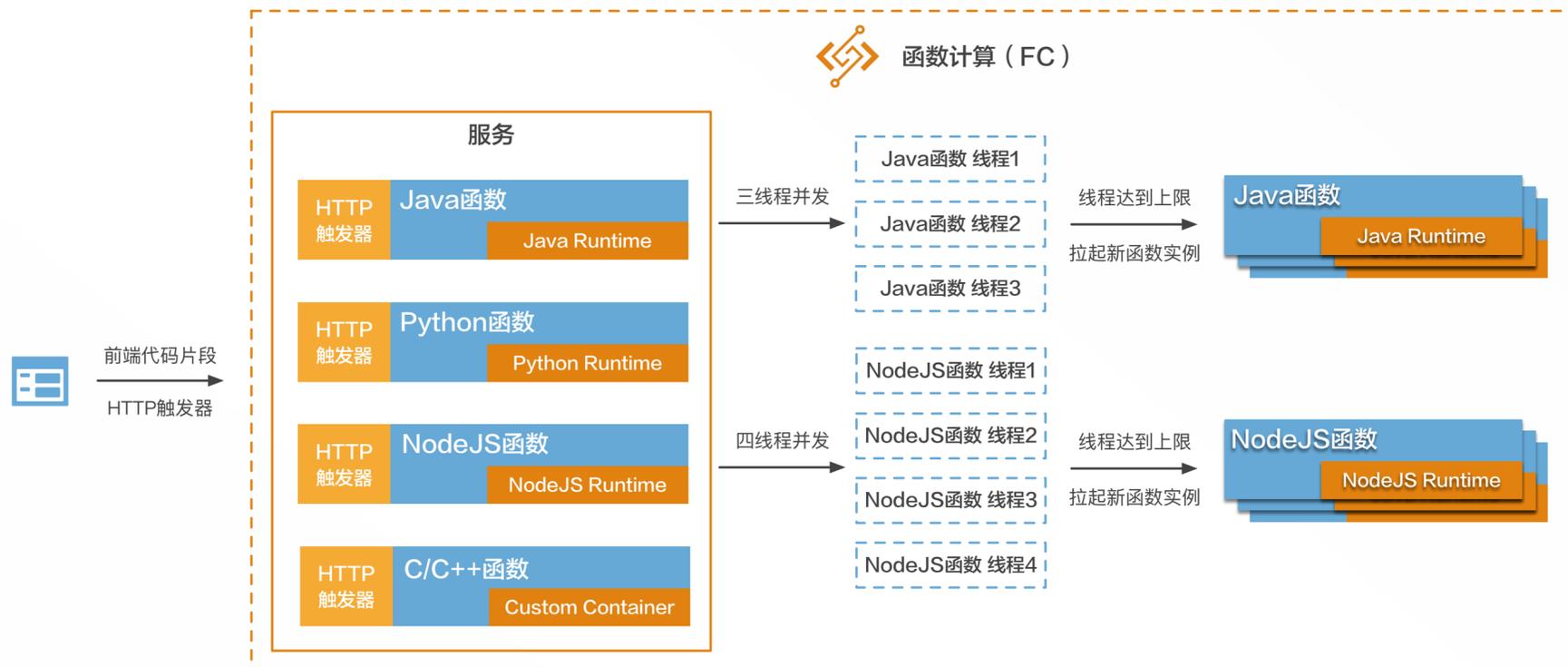
# 灰度发布能力



通过线上**共存新老版本**的灰度发布方式小范围验证新版本的能力，并且可以**动态、快速地切换**主版本和灰度版本，实现**增长黑客**

- 版本相当于服务的快照，包括服务的配置、服务内的函数代码及函数配置，不包括触发器。版本发布后不能更改，顺序递增。
- 别名可以理解为指向特定服务版本的指针，无法脱离服务或版本单独存在。使用别名访问服务或函数时，函数计算会将别名解析为其指向的版本。
- 在别名上可以设置主版本和灰度版本，并且可以对齐分配流量比例，从而实现一个业务有两个版本在线上共存。
- 当有新版本发布，需要切换线上版本时，只需要动态变更别名指向的版本号，即可快速实现线上版本切换。

# 单实例多并发



函数中有较多时间在等待下游服务的响应	适用	等待响应一般不消耗资源，在一个实例内并发处理可以节省费用。
函数中有共享状态且不能并发访问	不适用	例如全局变量，多请求并发执行修改共享状态可能会导致错误。
单个请求的执行要消耗大量CPU及内存资源	不适用	多请求并发执行会造成资源争抢，可能会导致内存不足（OOM）或者延时增加。

**用更少的资源承载更多的请求，进一步减少实例数，保护下游资源，优化资源成本。同时进一步降低冷启动概率，提高请求性能**

- 部分请求之间可以共享状态，如多个请求可以在一个实例内共用数据库连接池，避免大流量击穿下游服务。
- 降低冷启动率，进一步增加弹性速度。由于多个请求可以在一个实例内处理，创建新实例的次数会变少，冷启动概率降低。
- 减少占用VPC IP。在相同负载下，单实例多并发可以降低总的实例数，从而减少VPC IP的占用。
- 减少偏 I/O 场景下的总执行时长，进一步节省费用。

# 异步高可用/异步有状态调用



由用户**灵活控制**函数生命周期，实现**更多的业务场景**，比如可以灵活控制只录制长视频中的若干段

- 无状态异步调用的适用场景：
  - 函数执行时间短。
  - 不需要经常查看每个执行结果。
  - 不需要停止正在执行的函数。
- 有状态异步调用的适用场景：
  - 函数执行时间长（如2小时）。
  - 需要查看每次执行的结果。
  - 在执行函数的过程中需要中断函数的执行。

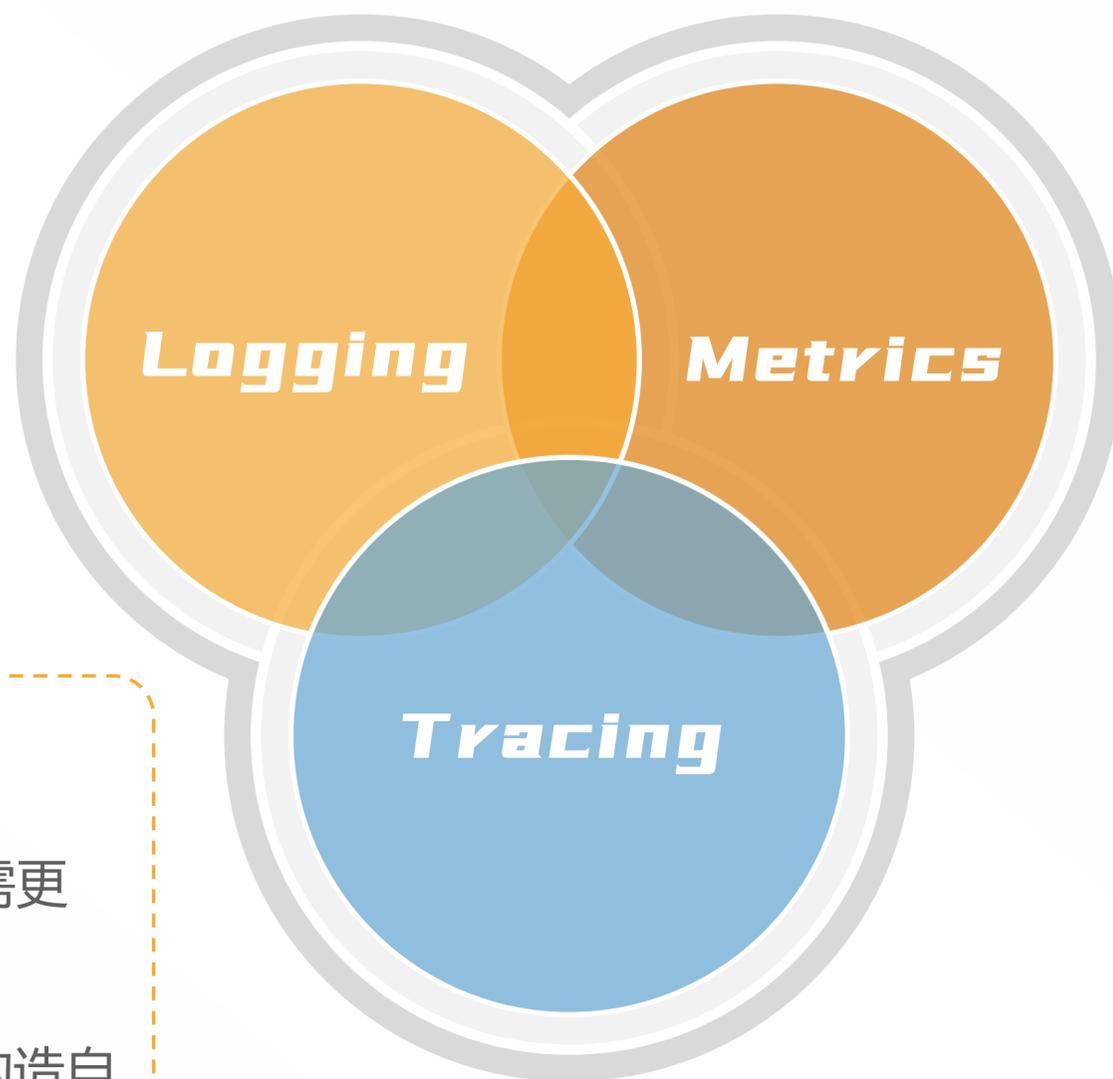
## 异步高可用

- 将函数的执行状态信息，例如执行失败或执行成功等通知到下游目标服务。
- 支持设置异步调用的重试次数和消息存活时长。

## 有状态异步调用

- 所有异步调用，会展示调用状态，如执行中、堆积、成功、失败等。
- 支持重试、随时终止、查看任务堆积情况等。

# 可观测能力



## Logging

- 支持runtime 原生语法，无需更改业务代码
- 有对日志分析处理能力，可构造自定义监控大盘
- 提供指标日志，分析请求调用详情

## Metrics

- 监控中心
  - 多层次，多维度
  - 指标丰富，层层递进
  - 错误排查，一镜到底
- Runtime指标
  - 集成APM平台
  - ARMS、博睿
- 自定义指标
  - 结合函数日志自定义业务监控大盘

## Tracing

- 拥抱开源，基于 OpenTracing 协议
- 分析全链路耗时
- 监测函数计算冷启动情况

# 权限控制和环境管理

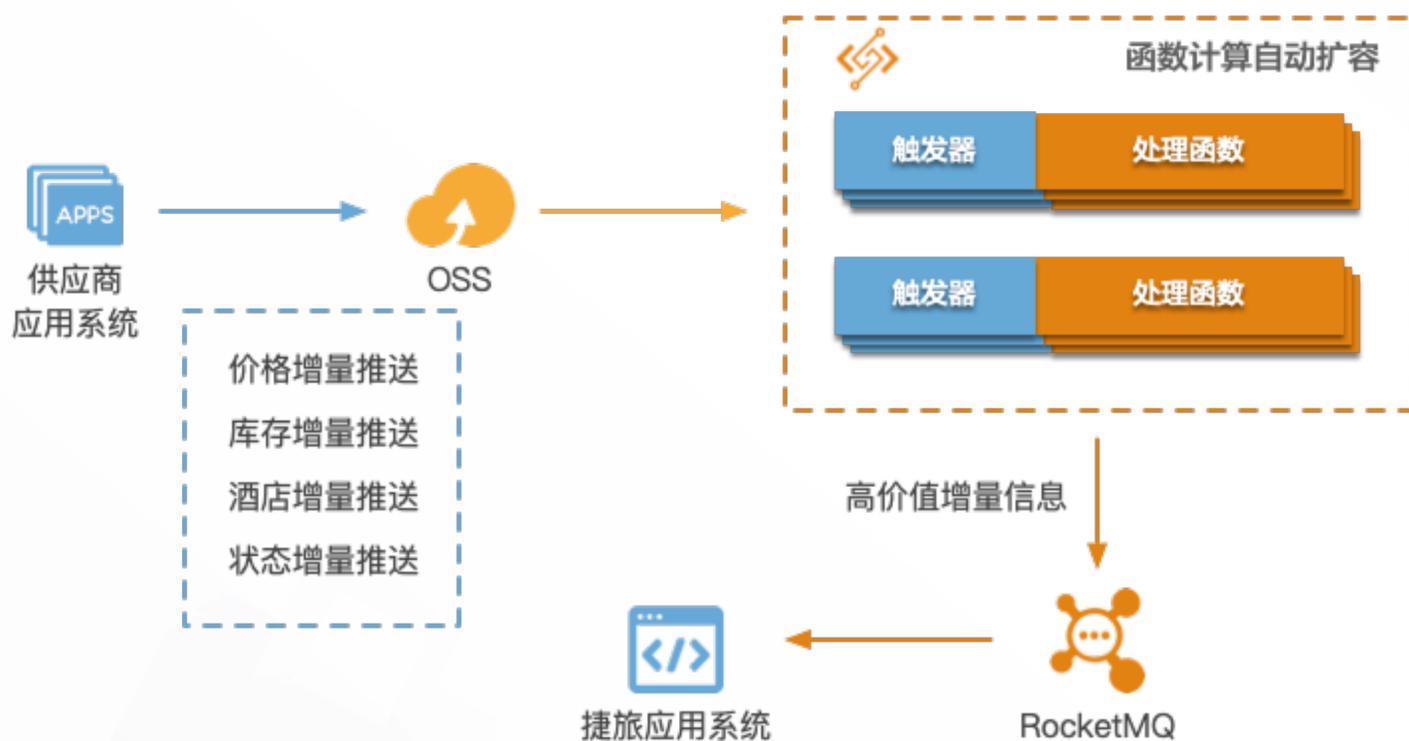


不同业务/环境负责人	负责人 A	负责人 B
RAM角色权限	仅授权对应服务RAM 权限	仅授权对应服务RAM 权限
不同服务设置不同标签	服务A 标签：Team:Dev	服务B 标签：Team:Ops
函数	函数A-A	函数B-A
	函数A-B	函数B-B
	函数A-C	函数B-C

通过标签功能将服务进行分组，可以实现不同角色对不同分组的服务拥有不同的操作权限

假设创建了10个函数计算的服务，其中5个授权给dev团队，另外5个授权给ops团队。每个团队只能查看被授权的服务，未被授权的不允许查看和编辑。

## 深捷旅弹性高可用 API 方案



### 客户背景

深圳市捷旅国际旅行社有限公司(简称“深捷旅”)，成立18年来专注于酒店行业的客房分销业务以及酒店预订信息化水平建设。通过品牌“捷旅假期”打造了业内领先的酒店B2B分销平台，合作的国内外旅游分销渠道超过1万家，拥有全球60万余家星级酒店资源，覆盖了全球300多个城市。

### 业务痛点

捷旅对接的酒店超过60万+，各个酒店的每天需要更新的信息都有几千万级的规模，而且这些消息的并发脉冲性强，时效周期短，瞬时并发处理压力大。需要同时处理的峰值消息量超过10W+，需要根据消息量的多少进行自动化的扩缩容操作，并且可以毫秒级进行快速的扩缩容，需要支持OSS，消息等多种数据源触发方式，需要支持Python, Go, Java 等各种编程语言等。

### 产品方案

函数计算可以监听多种数据源，通过毫秒级的扩容可以获得线性增长的业务处理能力。函数计算支持多种编程语言，无需客户改变编程习惯。函数计算支持便捷的部署，资源使用的实时监控，日志分析集成与报警能力。

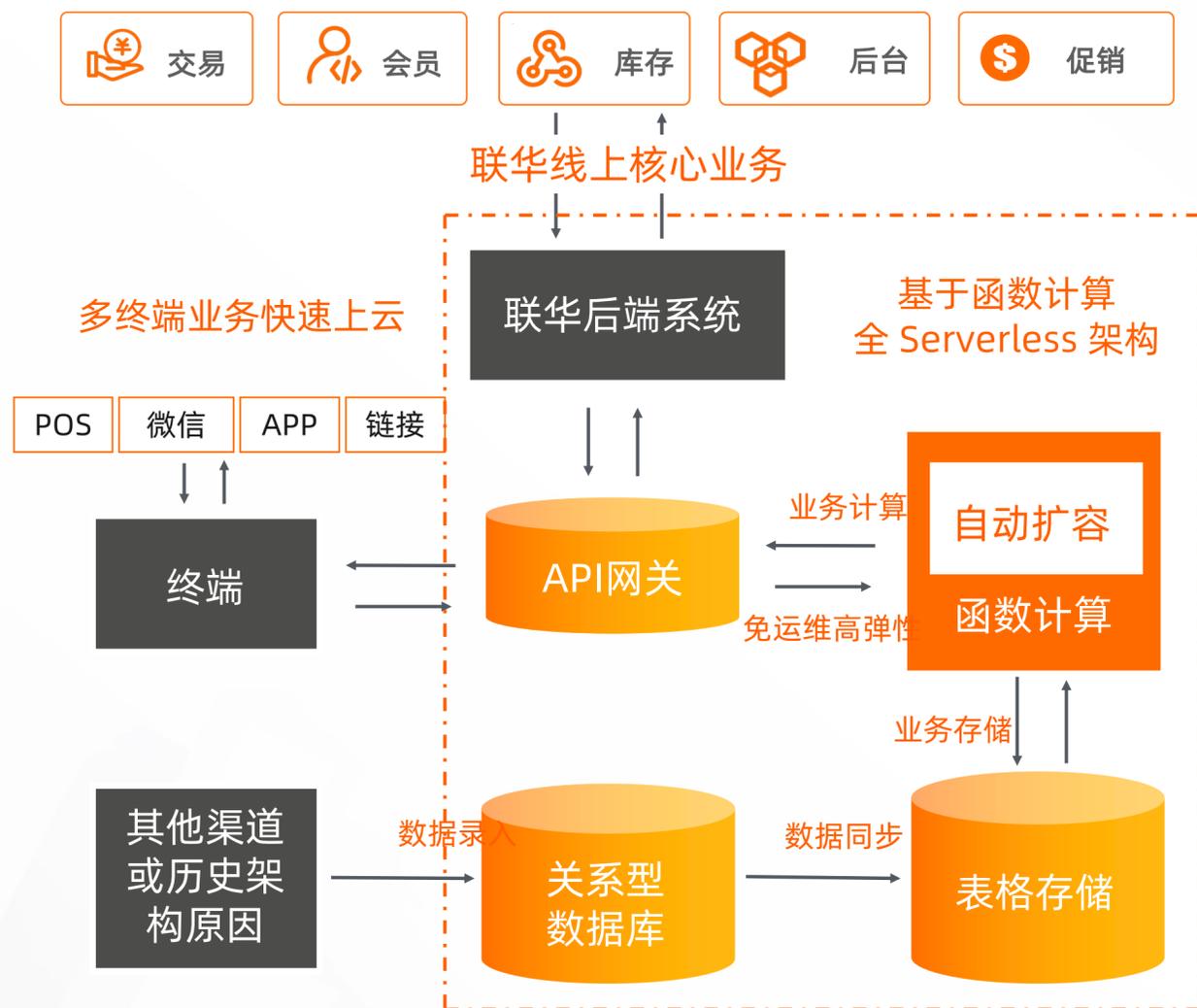
### 达到效果

- 业务稳健：函数计算弹性扩缩容，解决了以前按峰值资源配置方式的业务稳定性风险问题。
- 运维简单：以前最困扰的扩缩容资源管理部分，在使用函数计算后，完全不再需要维护，极大释放了捷旅开发运维人员的生产力。
- 节省成本：利用函数计算的按需付费的特性，在合适的规格运行下，整个资源得到了高效的利用，不为闲置付费，降低了捷旅的总体使用成本。

### 可复制性

全国、全行业可复制。

## 世纪联华全栈 Serverless 改造



### 客户背景

世纪联华快速上云，将『线上核心业务』改造为全 Serverless 架构的中台模式，采用『函数计算 + API 网关 + OTS』作为计算网络存储核心，弹性支撑日常和大促峰谷所需资源，轻松支撑 618 / 双11 / 双12 大促。

### 业务痛点

世纪联华各个系统都具有请求量大，逻辑相对简单，并且流量潮汐明显的特性，业务逻辑和存储方面都不是很重。所以提高效率、资源使用率是其最大痛点。

### 产品方案

世纪联华将所有的线上业务向Serverless模式靠齐，采用函数计算+API网关作为访问计算核心，表格存储作为存储核心。弹性支撑每日、大促峰谷所需资源。其中函数计算作为API、请求的具体实现，弹性应对抢券的请求流量的涌入和即时数据查询，保证业务流畅顺滑。

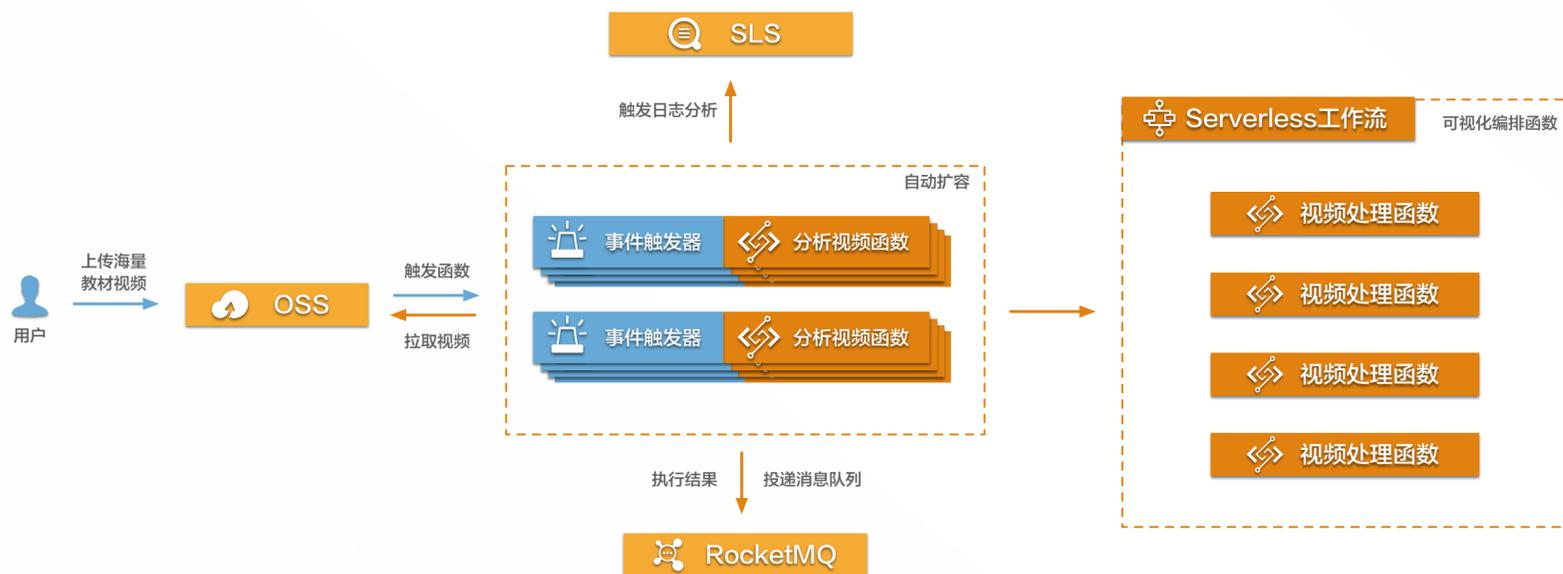
### 达到效果

- 全 Serverless 架构：FC + API 网关 + OTS Serverless 解决方案
- 弹性高可用：毫秒级弹性扩容、充足的资源池水位、跨可用区高可用
- 敏捷开发免运维：函数式极简编程可专注于业务创新，无采购和部署成本、提供监控报警等完备的可观测能力
- 业务收益：项目周期从 6 个月减少为 3 个月、4 次活动，非函数计算资源闲置超过 50 万，函数计算为闲置资源付费降低趋近于 0

### 可复制性

全国、全行业可复制。

## 蓝墨云班课 点播转码方案



### 客户背景

蓝墨是一家由美国留学生回国创业的高科技公司，专注于移动互联时代数字出版和移动学习领域的新技术研究及平台运营，依托自主研发的蓝墨移动交互式数字教材核心技术体系，为出版社、学校和教师提供移动交互式数字教材、校园移动学习平台及教师自助数字出版等解决方案的公司。

### 业务痛点

蓝墨每天都要处理大量视频教材资源，涉及视频剪辑、切分、组合、转码、分辨率调整、客户端适配等一系列复杂的技术工作。通过几年的实践，蓝墨的技术团队通过FFmpeg等技术已经建立起一整套自主可控视频处理机制，支撑了业务的快速发展。但业务增长速度是蓝墨的工程师们始料未及的，高峰期高于往年数十倍的视频处理需求让现有的架构不堪重负，严重影响了用户体验。

### 产品方案

蓝墨技术团队把代码逻辑从原有的架构无缝迁移到FC，将FC作为处理视频的计算资源基础层，结合OSS触发器和Serverless workflow构建视频上传、拉起资源、分布式处理、回传视频的自动化流程。

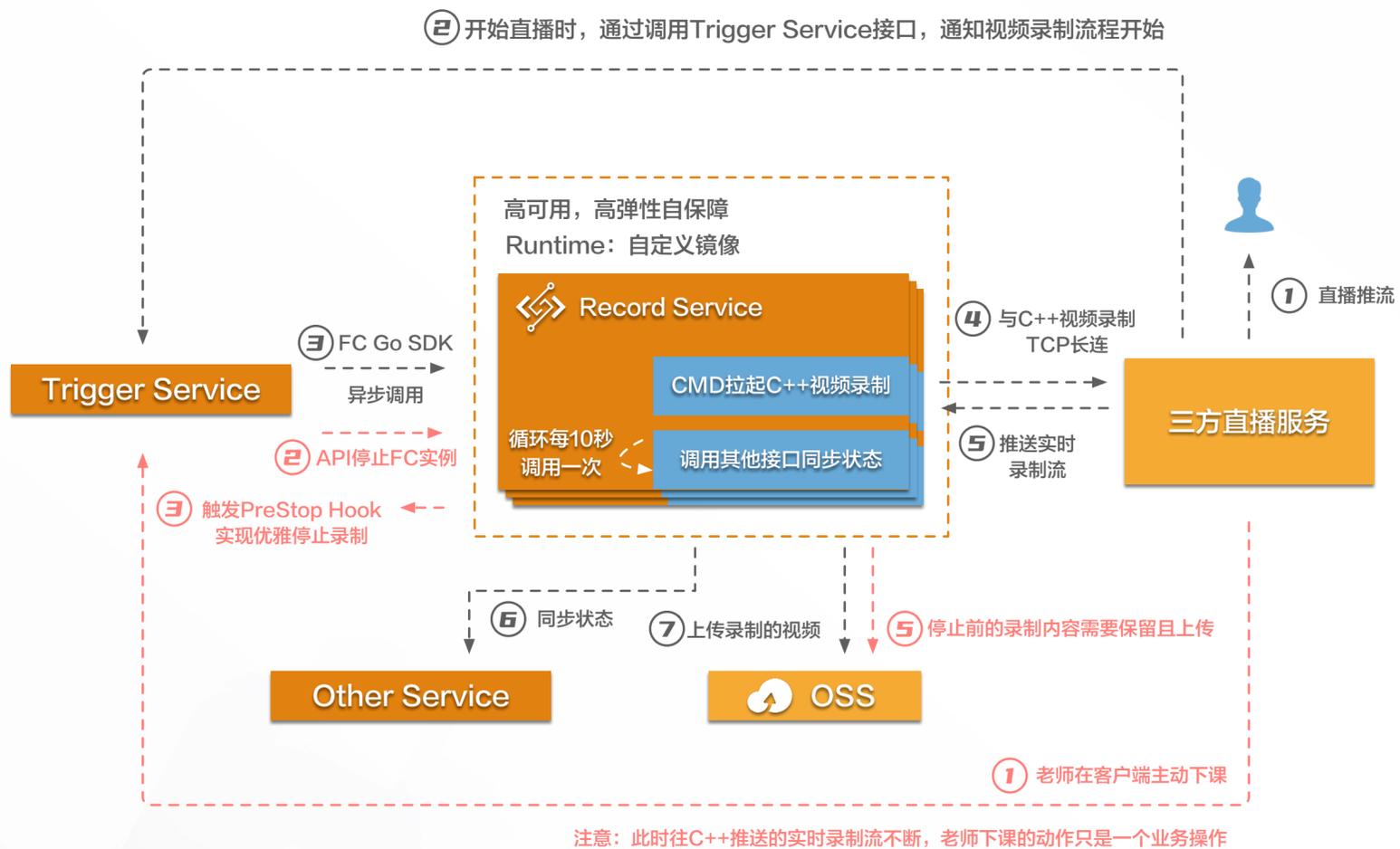
### 达到效果

通过对接OSS触发器，只要OSS上有新的视频源文件上传，就能自动拉起函数计算实例，通过整合Serverless workflow，还能对分布式任务进行统一编排，实现对于大文件切片后进行并行处理并最终合并的复杂操作，更进一步的提升了处理效率。

### 可复制性

全国、全行业可复制。

## 头部在线教育 实时录制业务



### 客户背景

某头部在线教育企业是一家集教育培训、教育产品研发、教育服务等于一体的大型综合性教育科技集团。为响应国家政策，客户率先开始启动点播配套的相关业务，其中对老师录制的授课视频进行转码是其核心之一。目前已经使用Serverless方案完美落地并验证。

### 业务痛点

上课客户端由老师控制，经常第一位老师开始上课后就会一直开启直播，无法区分不同老师的授课视频，客户需要从长视频中截取录制不同老师授课的视频，整个流程需要客户自己来控制，基于ECS架构的整套流程复杂度高，容错率较低，并且需要预置大量计算资源，资源浪费比较严重。

### 产品方案

技术团队把代码逻辑从原有的架构无缝迁移到FC，将FC作为处理视频的计算资源基础层，利用函数计算的弹性能力优化资源利用率，并通过函数计算有状态异步调用的能力，和客户的控制流程相结合，灵活控制启停录制功能。

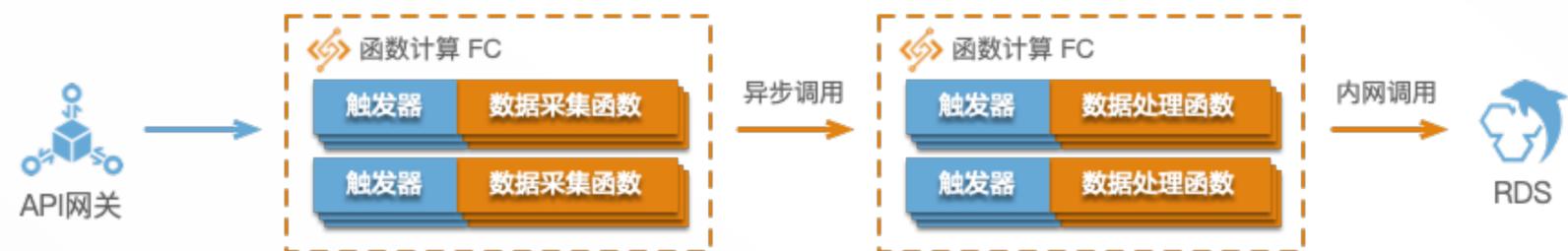
### 达到效果

合理降低了客户原有的视频实时录制方案的复杂度，避免了很多不必要的运维操作，人效有明显提升。通过预热实例、预留实例、弹性实例、性能实例配合使用，有效应对流量波峰波谷。且在保障算力的同时兼顾业务录制不同品质视频需要不同算力的需求。

### 可复制性

全国、全行业可复制。

## 头部车企 海量数据处理



### 客户背景

汽车市场的需求多样化、环保和安全标准越来越高、先进技术深化发展，汽车产业本身正迎来巨大的变革期，云服务、IoT、大数据、AI等飞速发展，期望借助信息技术使得各种数据服务得到有效利用。

### 业务痛点

客户的终端都是车辆，数据源和数据量都非常庞大，传统的数据接收层在应对大流量冲击时比较吃力，另外对数据的解析和处理相对轻量，使用Flink或Spark套件略显臃肿，而且成本也会增加。所以应对大流量冲击的数据接收层和性价比高的数据处理层是客户的痛点。

### 产品方案

在汽车内部的终端，会实时上报如汽车位置、车况、监控度等信息到云上，经过数据解析和转换，把数据进行入库操作。客户将数据接收层，数据解析转换层放在函数计算中处理，然后将处理后的数据流转下游服务。

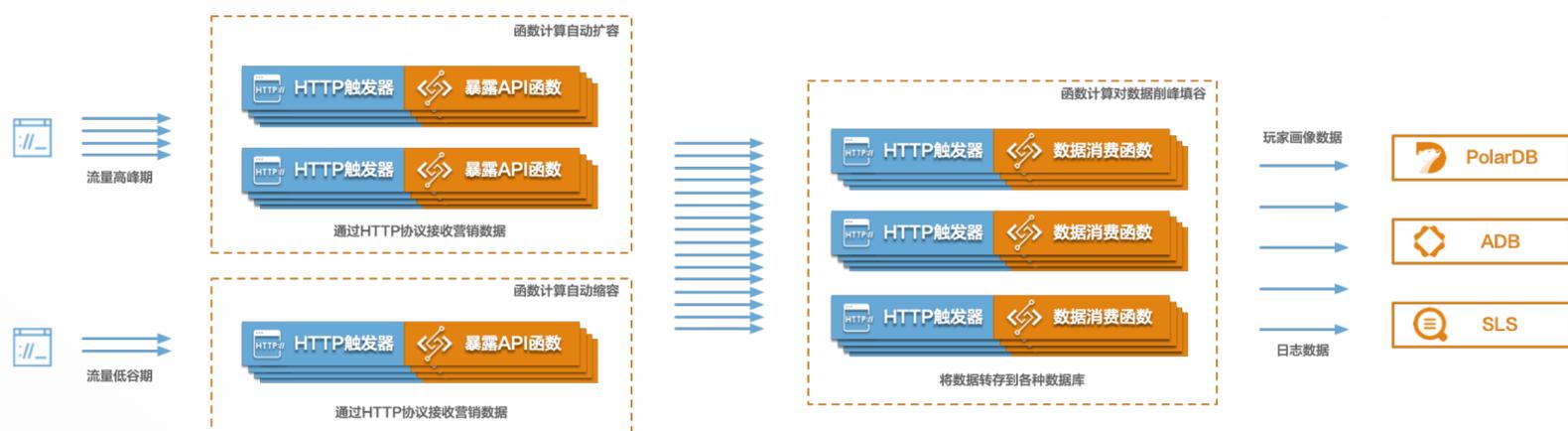
### 达到效果

- 降低成本：整个业务模型有明显波峰波谷，FC完全按量付费
- 海量并发：FC+API网关，可以应对数十万级的QPS访问
- 弹性伸缩：无论是毛刺还是稳态请求，函数计算都能支持百毫秒级的扩容

### 可复制性

全国、全行业可复制。

## 武汉游戏群 大规模游戏营销平台



### 客户背景

武汉游戏群科技有限公司，旗下有多款角色扮演和休闲卡牌游戏，在游戏新开服或者日常营销大促时，均会与其合作广告商联合投放大量互联网广告，感兴趣的玩家群体会通过广告点击被引导下载和开通游戏，且每日的数据均需要汇总分析进行玩家画像。

### 业务痛点

该业务场景无法准确预估运营流量，高并发、高弹性的后台服务对业务结果至关重要。且大量的营销数据和原始日志需要进行离线分析，驱动运营策略升级。客户用ECS做数据接收层无法承载买量时的大流量冲击。

### 产品方案

将数据接收层放在FC，依靠FC极致的弹性能力承载买量时的大流量冲击，然后异步拉起处理数据的函数做削峰填谷，保护下游业务不被流量击穿。

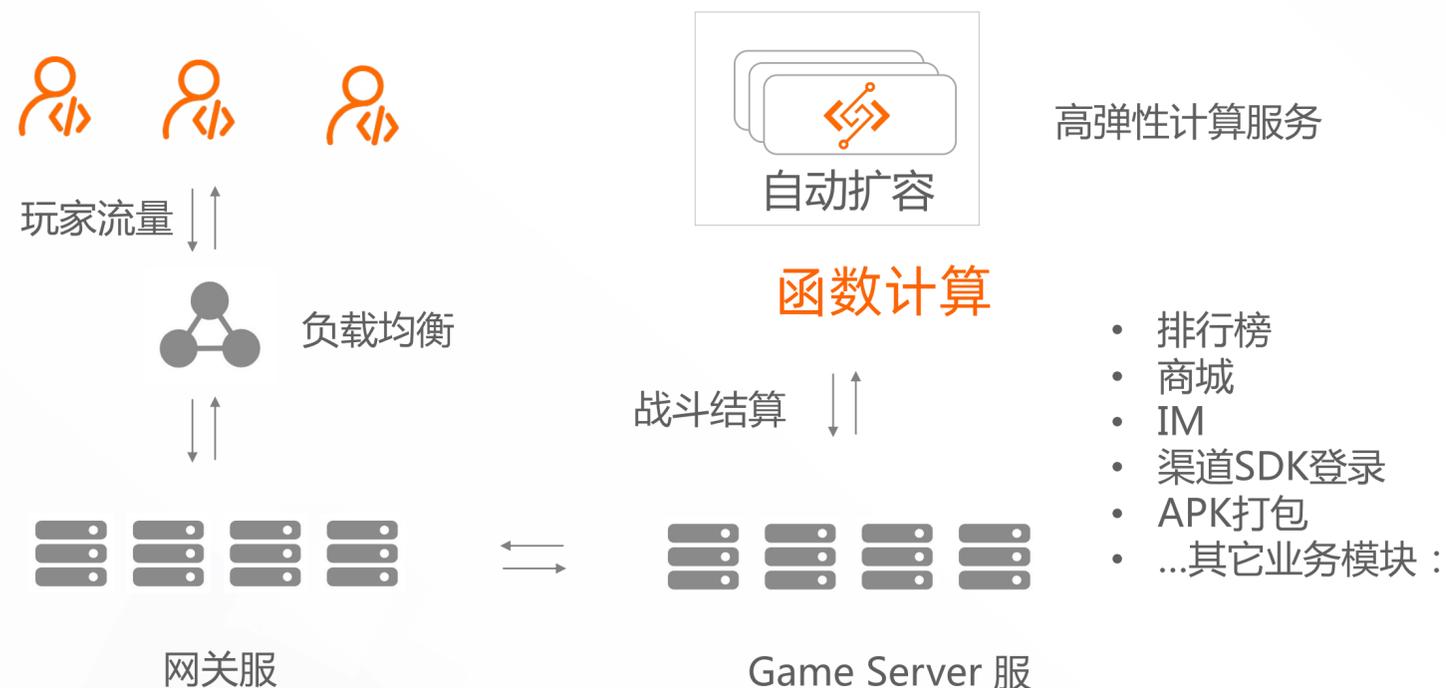
### 达到效果

- 弹性高可用：海量请求（数十万 TPS）高并发和极致弹性需求，稳定可靠的容灾和持久化存储能力，灵活的削峰填谷能力保护后端系统稳定。
- 事件驱动：独有的云产品间事件驱动能力，解放架构难点，自动化处理流程。
- 数据处理：支持定时任务，提供充足的计算资源和高度自定义数据分析策略。
- 降低成本：按量付费，客户不再需要预留闲置资源，实时反馈营销结果投入。

### 可复制性

全国、全行业可复制。

## 头部游戏客户 高弹性战斗结算业务



### 客户背景

头部游戏公司，旗下有大热的卡牌和SLG游戏作品，在该公司卡牌类明星游戏产品中，为了防止客户端作弊，在每局战斗之后，在客户端预判玩家胜利的情况下，需要服务端来进行战斗数据的结算，从而确定玩家是不是真正的胜利。

### 业务痛点

战斗结算是强 CPU 密集型，结算系统每日需要大量的计算力，尤其是开服或者活动期间突然涌入的大量玩家，导致需要的计算量瞬间几倍增长，同时需要结算系统保持稳定的延时来保证玩家的用户体验。

### 产品方案

游戏主服完成战斗后，拉起对应的函数实例对战斗结果进行二次结算，确保战斗结果无误。

### 达到效果

- 敏捷开发免运维：只需要专注业务逻辑代码的编写，工程效率大幅提高，同时云平台提供完善的监控设施。
- 弹性高可用：毫秒级伸缩扩容，计算力不再是瓶颈，保证应用在负载动态剧烈变化时仍能保持稳定的延时，平滑处理暴增的调用请求，满足在线场景。
- 降低成本：按需付费，客户不需要为峰值预留计算资源。
- 模块公共化：减轻游戏逻辑服的压力，结算系统可以快速复制到任何有类似需求的游戏。

### 可复制性

全国、全行业可复制。

# 03 Serverless PaaS Serverless应用引擎 SAE

# Serverless应用引擎 SAE 是什么



Serverless应用引擎SAE ( Serverless App Engine ) 是面向应用的Serverless PaaS平台，能够帮助PaaS层用户免运维IaaS、按需使用、按量计费，做到低门槛微服务应用上云、低门槛架构容器化转型。相对于其他Serverless产品，它将应用的概念抽象化，并提供了一整套微服务解决方案，支持Spring Cloud、Dubbo等主流的微服务开发框架。同时也支持其他多种语言和框架，实现了Serverless架构和各类架构的完美结合。



**成本更优、效率更高的一站式应用托管方案**

**微服务领域最佳的Serverless实践**

**低门槛、高效率、高可用的容器化转型方案**

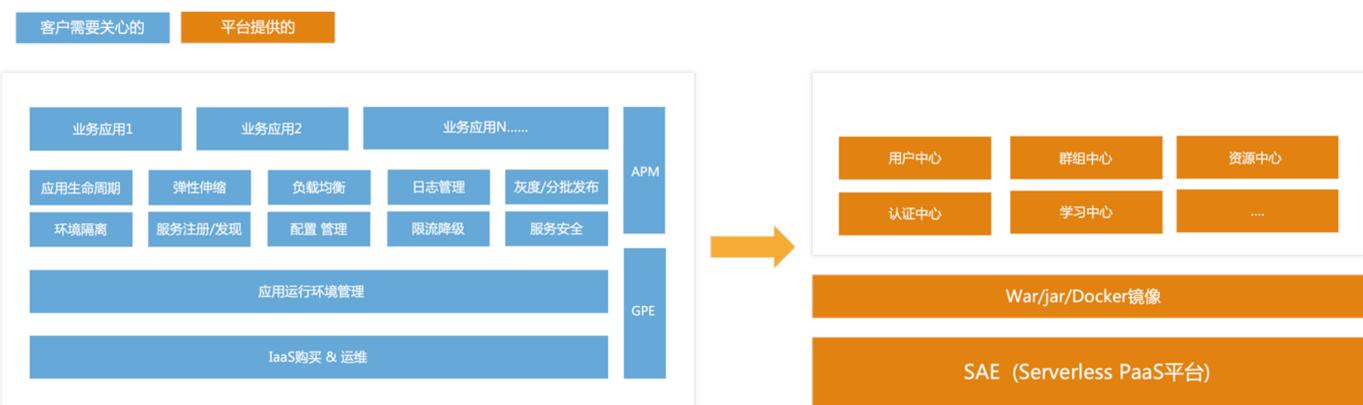
# Serverless应用引擎 SAE 应用场景



微服务架构转型

## 底门槛微服务架构转型

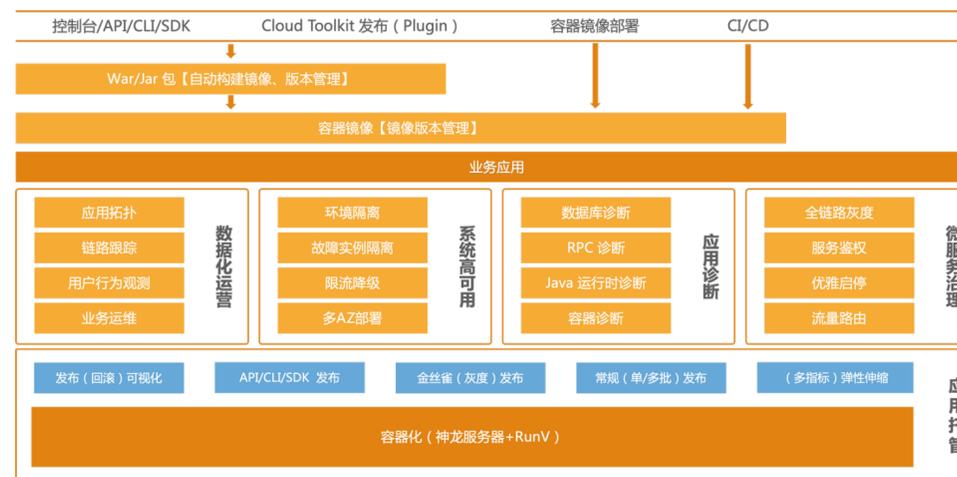
- SpringBoot、SpringCloud应用零代码改造即可迁移
- 微服务套件天然集成，无需额外搭建。比如服务注册/配置中心、APM、限流降级等
- 具备服务优雅上线下、按流量/按内容灰度发布等服务治理和服务管理的能力



低门槛容器化转型

## 低门槛容器化转型

- 不需要用户有K8s相关经验，屏蔽繁复的、有难度的K8s概念，无需破坏原有CI/CD流程
- 用户使用Jar包、War包、Zip包即可，SAE自动将代码包制作为镜像



轻松应对流量潮汐、流量脉冲场景

## 轻松应对流量潮汐、流量脉冲场景

- 使用PTS、ARMS、AHAS、SAE高可用四套件优化系统，评估系统容量基线
- 根据系统容量基线，配合定时弹性策略与指标弹性策略，从容应对流量脉冲



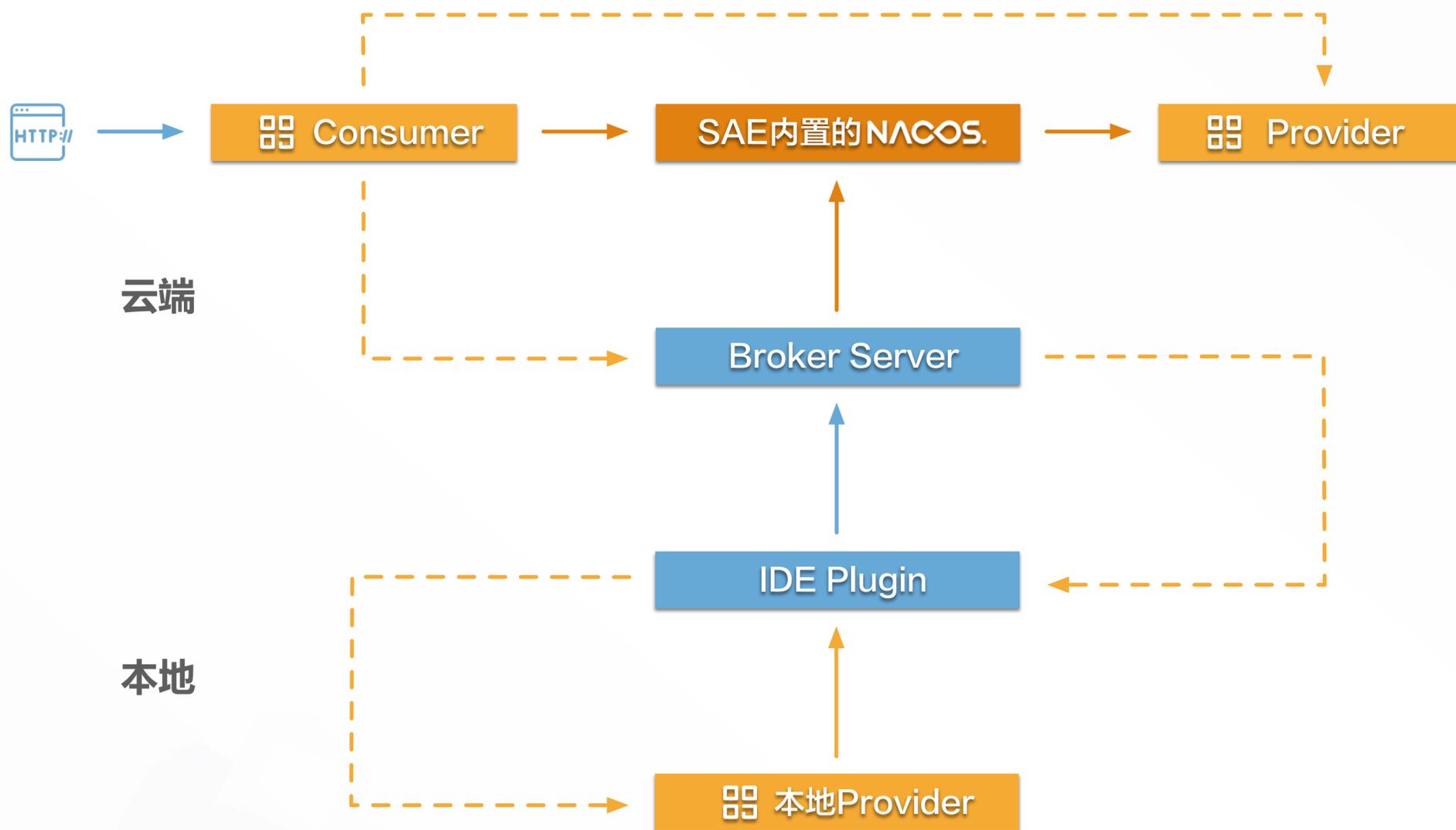
一键启停内部环境降本增效

## 一键启停内部环境降本增效

中大型企业多套环境，测试环境一般晚上都不使用，长期保有应用实例，闲置浪费高。使用 SAE 一键启停，节省2/3硬件成本



# 端云联调



## 解决Serverless架构本地开发联调的痛点

基于SAE Cloud Toolkit 插件 + 跳板机实现：

- 本地服务订阅并注册到云端SAE内置的Nacos注册中心
- 本地服务可以和云端SAE上部署的服务互相调用

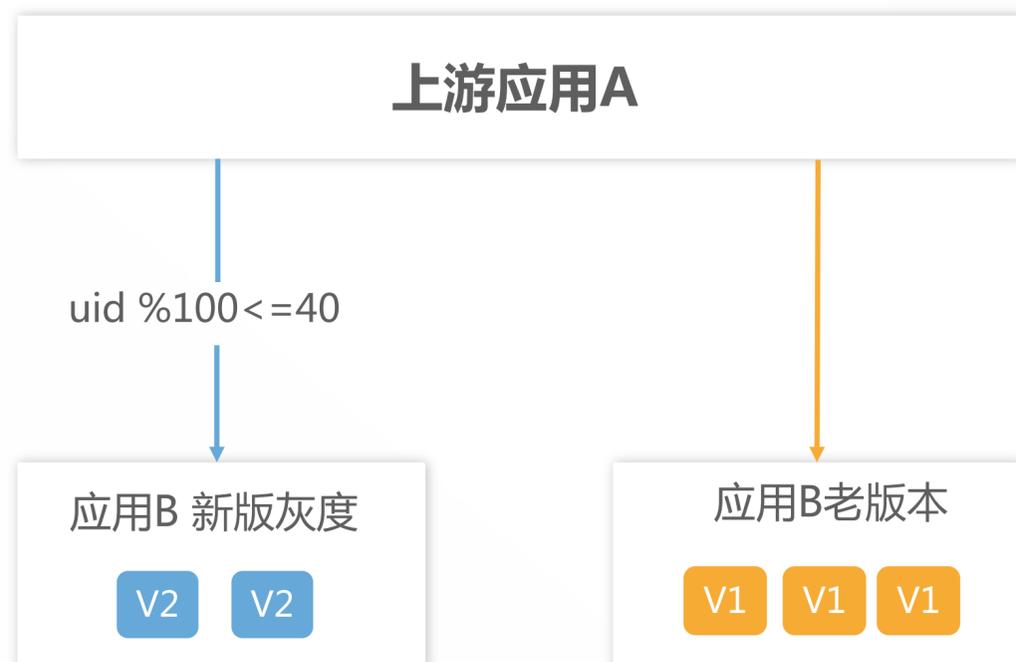
适用场景：

- 本地IDC的服务迁移到云端SAE，迁移过程中的开发联调
- 本地开发测试验证



## 类型一：按请求内容灰度

把请求内容符合指定条件的流量作为灰度流量

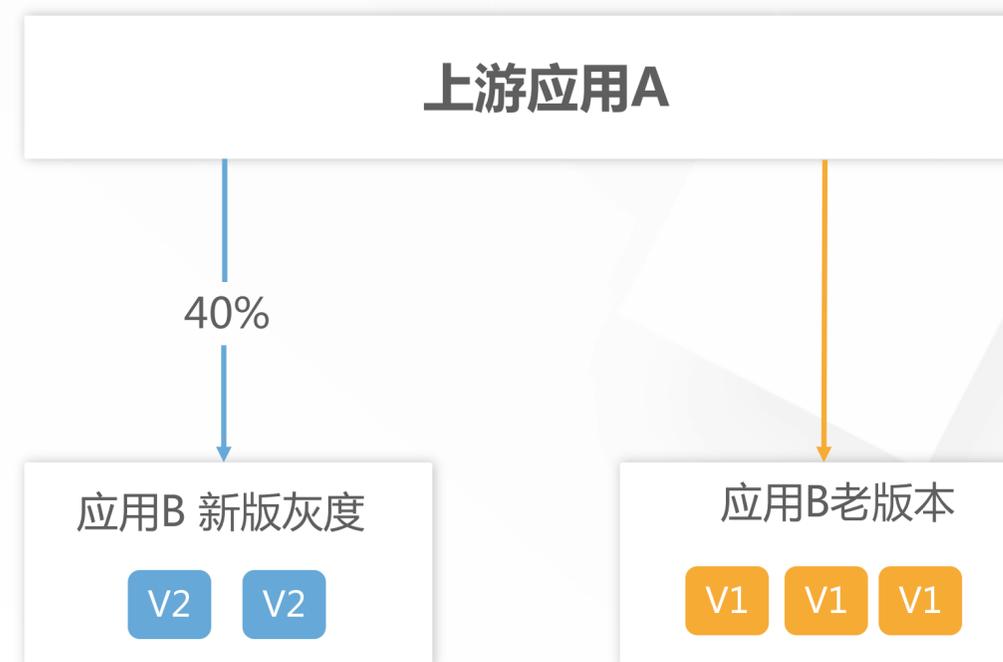


### 适用场景：

新版本发布，为了控制最小爆炸半径，精准客户进行灰度。  
如优先验证XX地域的某类客户。

## 类型二：按流量比例灰度

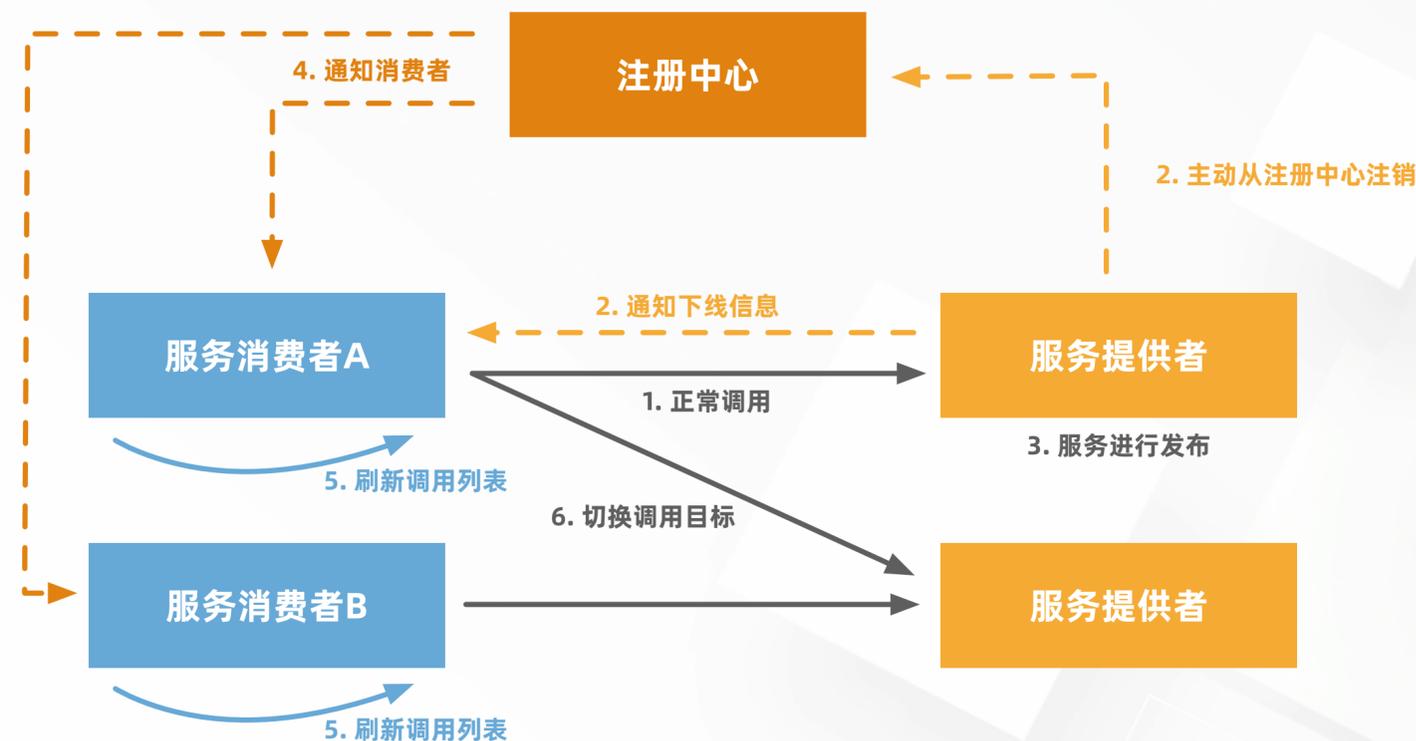
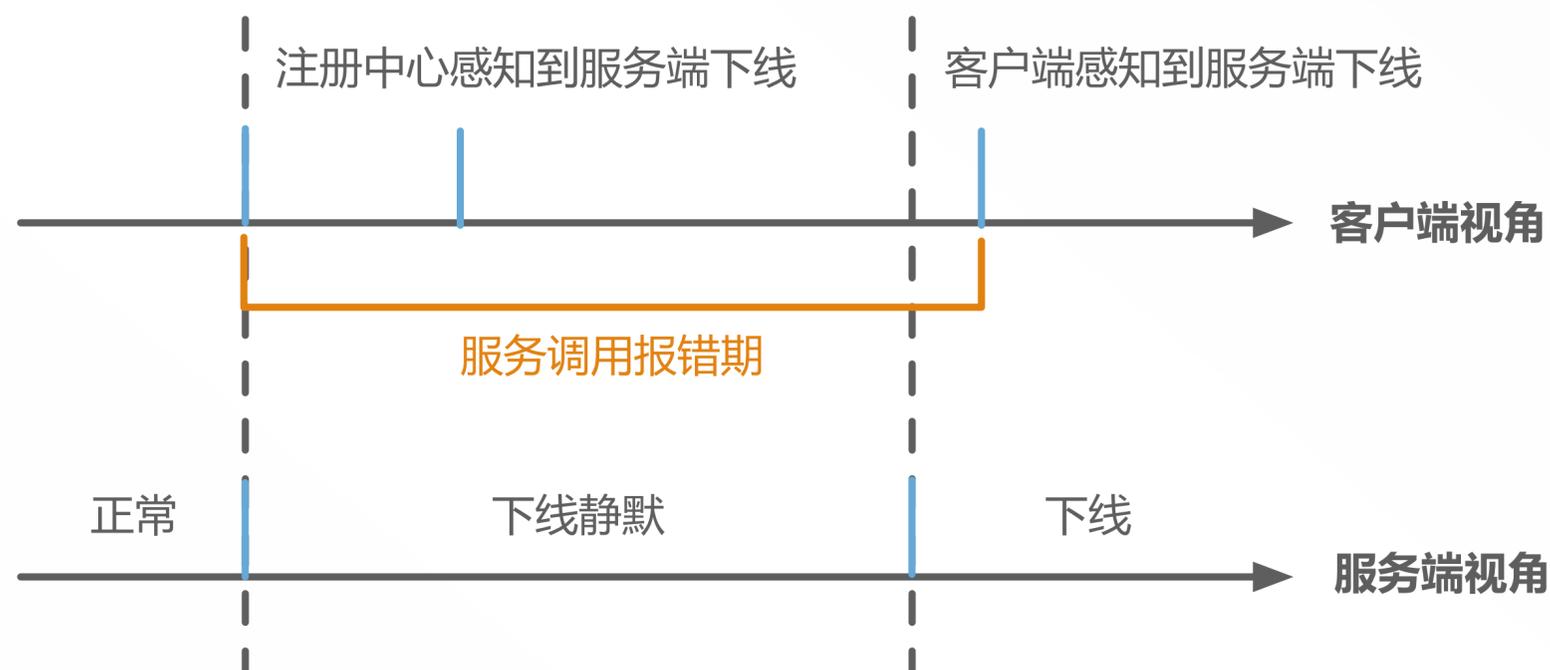
把指定比例的请求流量作为灰度流量



### 适用场景：

直接通过流量比例进行灰度，保证在安全可控范围内。  
或者在精准客户验证完后，需要持续引流增加灰度比例。

# 微服务优雅上下线



## 通过无损下线，实现在白天大流量发布也能丝般顺滑

- 发布一个服务端应用的时候，都会有一个下线静默期，从正常到下线，中间有一段时间注册中心是不知道的。这段时间内客户端去调服务端，会有一个延迟，有一个服务调用报错期。
- 在SAE里面，默认帮每个应用挂载的agent会做一个主动通知，在Provider下线的时候，会主动刷新注册中心服务列表，主动通知到消费端。消费端主动刷新，直接调用正常的Provider，保证你在应用发布过程中流量无损，丝般顺滑。

# 丰富的弹性指标，灵活的弹性策略

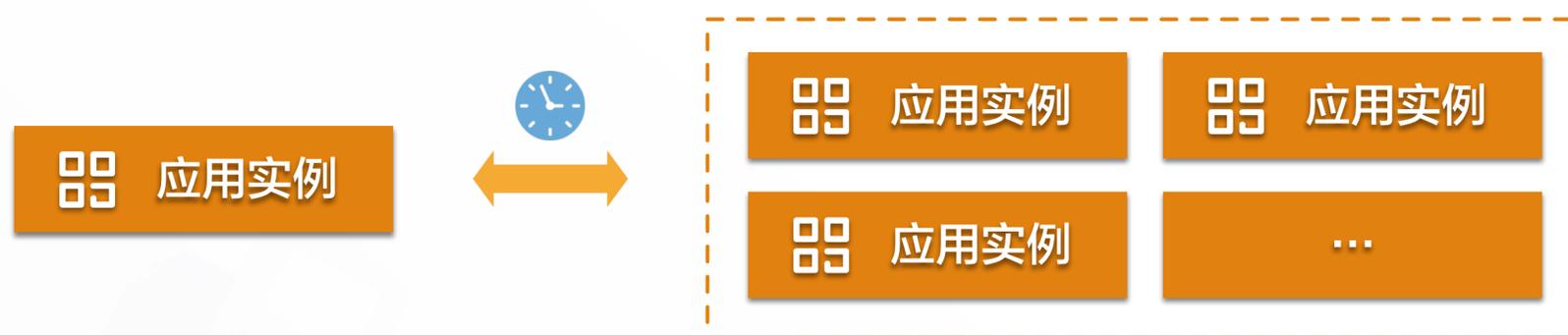


基于基础监控指标（CPU、Mem等）和业务监控指标（QPS、RT、TCP等）触发的水平伸缩



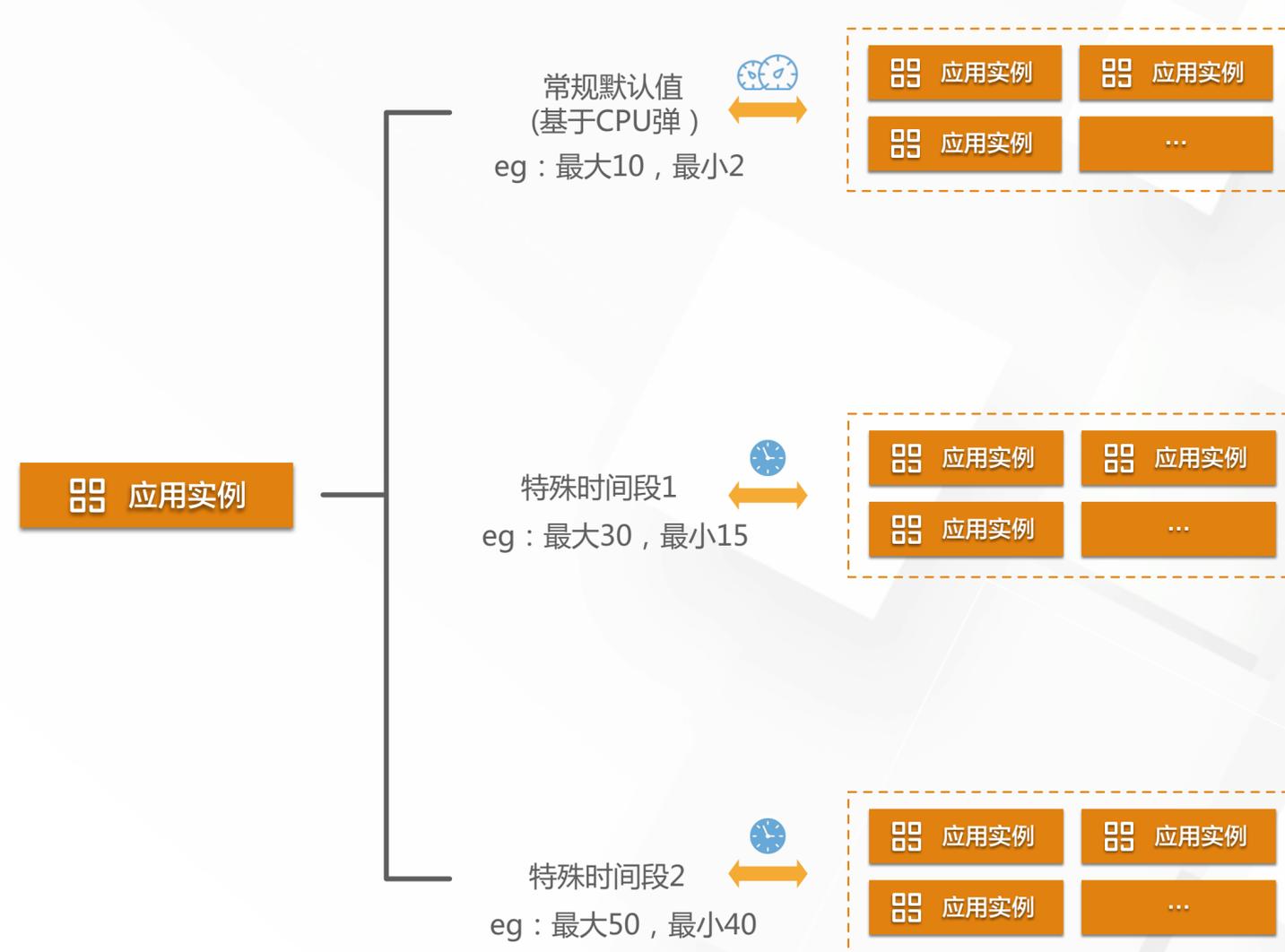
- 优势：比开源K8s HPA 指标丰富，且可以自定义指标。
- 适用场景：适用于有突发流量、典型脉冲的应用场景，多用于互娱 / 游戏 / 社交平台 / 电商等行业。

基于定时时间段触发的水平伸缩



- 优势：操作简单，易用，定时时间段粒度精细。
- 适用场景：适用于资源画像存在周期性的应用场景，多用于餐饮 / 出行 / 证券 / 医疗政府等行业。

基于定时弹性&指标弹性混用的弹性策略

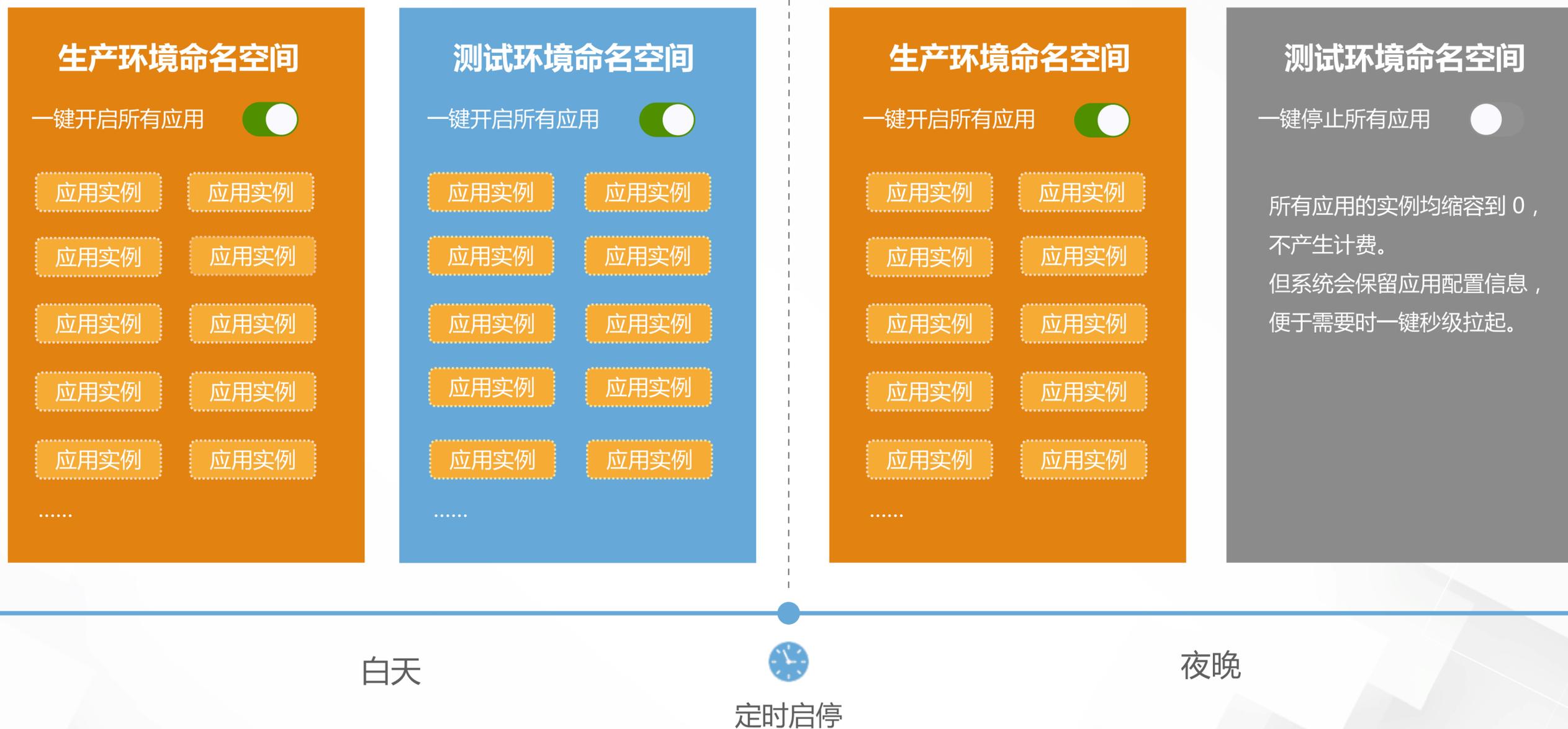


- 优势：两种策略混合使用，最大程度满足灵活性，只用设置一次规则全搞定。
- 适用场景：适用于固定时段内有突发流量、典型脉冲，常稳时段内流量波动不均的应用场景，多用于 媒体报社 / 在线教育 / 语音识别合成 等行业。

# 一键启停内部环境，降本增效



中大型企业多套环境，内部环境长期资源浪费，使用 SAE 一键启停，可以节省一部分资源成本



# 可观测能力



能够看到应用的调用关系拓扑图，可以定位到慢SQL、慢服务、方法的调用堆栈、进而定位到代码级别的问题

应用名称	日志产生时间	状态	IP地址	调用类型	服务名称	方法栈	线程剖析	时间轴 (ms)
ikf	2020-09-10 23:37:10.434	●	...	前端	35:8080/blog/selection/1			5169ms
ikf-web	2020-09-10 23:37:10.503	●	...	HTTP入口	/blog/selection/1	+	+	5006ms
ikf-blog	2020-09-10 23:37:10.504	●	...	HTTP入口	/blog/selection/1	+	+	5004ms

ikf-blog(/blog/selection/1)

提示：如果方法栈不足以定位代码问题，请继续查看线程剖析结果。

调用方法	行号	扩展信息	时间轴 (ms)
Tomcat Servlet Process			5004ms
org.apache.catalina.core.StandardHostValve.invoke(org.apache.catalina.connector.Request request, org.apache.catalina.connector.Response response)	109		5004ms
org.springframework.web.servlet.FrameworkServlet.doGet(javax.servlet.http.HttpServletRequest request, javax.servlet.http.HttpServletResponse response)	866		5003ms
com.ikeyforge.blog.controller.BlogController.findBlogByIsSelection(int isSelection)	227		5002ms
com.mysql.jdbc.ConnectionImpl.prepareStatement(java.lang.String sql)	4036	参数: select blog0 .id as...	
com.mysql.jdbc.PreparedStatement.executeQuery()	1964	参数: select blog0 .id as...	

## 线程剖析

机器IP: ...23 - 实际耗时: 5004ms - 监听耗时: 4500ms - 服务名称: /blog/selection/1

说明：只包含一个子方法时不缩进。红色表示真正消耗CPU的方法。格式：类名.方法名:行号[监听耗时]

```
java.lang.Thread.run( ):748 [4500ms]
org.apache.tomcat.util.threads.TaskThread$WrappingRunnable.run( ):61 [4500ms]
java.util.concurrent.ThreadPoolExecutor$Worker.run( ):624 [4500ms]
▼显示已隐藏的58个条目
com.ikeyforge.blog.controller.BlogController$$FastClassBySpringCGLIB$$7809e1ce.invoke( ):-1 [4500ms]
com.ikeyforge.blog.controller.BlogController.findBlogByIsSelection( ):227 [4500ms]
com.ikeyforge.blog.controller.BlogController.sleepForArms( ):233 [4500ms]
java.lang.Thread.sleep( ):-2 [4500ms]!
```

➤ 应用整体指标、应用实例指标、应用接口指标三个维度监控应用健康情况。

➤ 调用链路追溯、方法栈详情(耗时、SQL、异常信息等)、线程剖析。

# 镜像加速



原生 K8s 部署重建升级策略



重点提升效率场景

➤ 应用部署升级

➤ 应用/实例重启

SAE原地升级策略



部署效率提升42%

SAE部署升级策略



原生 K8s 部署重建升级策略



重点提升效率场景

➤ 弹性扩容

➤ 应用重启

➤ 应用部署 (更改配置)

SAE镜像预热策略



弹性效率提升30%

SAE部署升级策略



# 限流降级



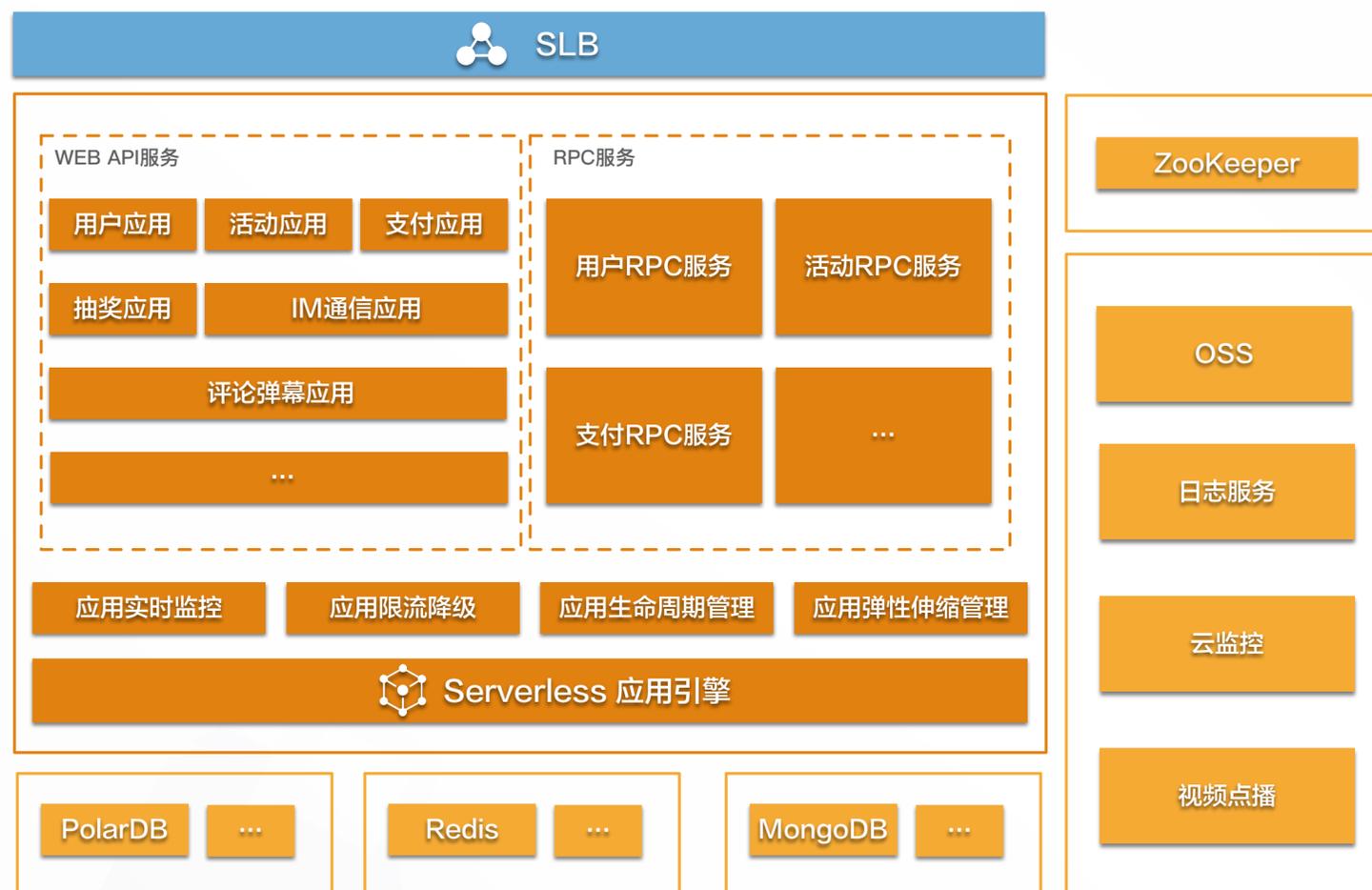
## 集成应用高可用服务AHAS，提供白屏化的限流降级配置

- 限流：针对非核心服务调用者
- 降级：针对系统需要调用的非核心服务
- 保障核心业务的正常运行

# 经典案例



## 爱奇艺体育 直播平台



### 客户背景

爱奇艺体育是中国知名的体育赛事平台，提供欧洲杯，西甲、网球、高尔夫等赛事直播、视频、新闻等超多精彩内容。

### 业务痛点

- 扩容繁琐：面对流量激增，需要临时部署应用进行应对，流程耗时，影响用户体验。
- 缺少应用级监控：运维团队期望对线上微服务架构做到有效的监控，能及时发现、定位问题。
- 资源利用率低：赛事带来周期性的流量，预估不准带来资源浪费巨大。

### 产品方案

客户平滑、快速的将赛事直播业务全部迁移到SAE中，配合SAE丰富的弹性策略轻松应对赛事直播时的不确定的流量脉冲。

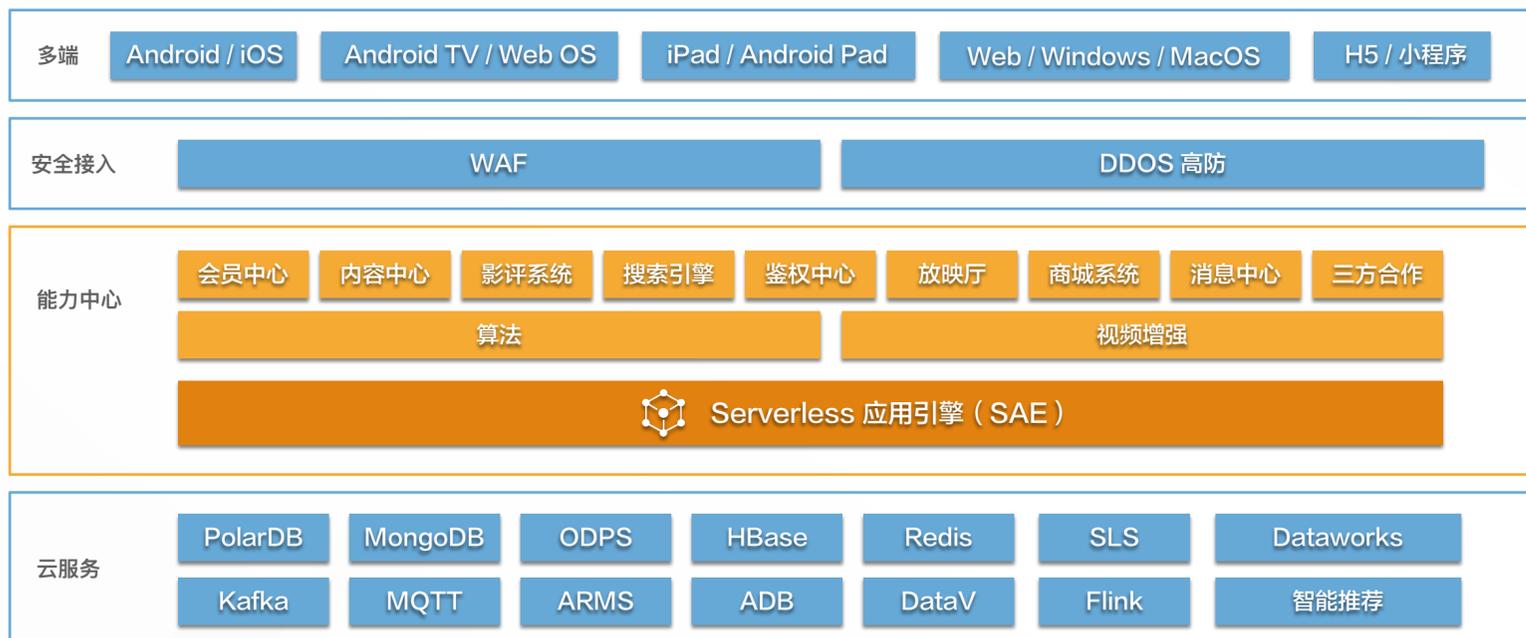
### 达到效果

- Serverless 弹性：从容应对周期的业务洪峰，省去原来预估资源、部署应用的操作，扩容效率提升40%。
- 内置应用监控：集成ARMS监控，可以快速精准定位问题，处理速度大约提升30%。
- 资源节省：释放闲置资源，整体资源利用率提升近50%。

### 可复制性

全国、全行业可复制。

## 南瓜电影 7 天内全面 Serverless



### 客户背景

南瓜电影App是国内领先的专注于影视精品化运营的垂直类视频产品，在移动互联网、OTT等客户端，为用户提供差异化内容运营服务，是国内唯一为用户提供专注于精品电影电视剧的全会员制视频App。

### 业务痛点

- 业务场景具有潮汐特性，业务高峰主要是晚间和节假日以及热门影片发布，面临快速响应和成本上的压力。
- 线上业务系统监控能力存在一些薄弱的地方，需要端到端的整体监控，保证业务系统稳定运行。

### 产品方案

客户仅用时一周便将所有核心业务从传统的ECS集群部署架构迁移到SAE Serverless架构，配合多种指标弹性策略，从容应对新片上映时的流量脉冲场景。

### 达到效果

- 效率提升：屏蔽了底层资源，不必关心底层IaaS资源管理、运维细节，加速业务迭代周期。开发运维效率提升70%。
- 成本优化：无需长期保有固定资源，按照实际业务量进行应用的扩缩容，做到按分钟计费。成本优化40%。
- 业务稳定高效：SAE提供了很好的弹性策略，同时能够做到秒级扩容，最大限度保证了整个集群的高可用。扩容效率提升10倍。

### 可复制性

全国、全行业可复制。

## 广州小迈 游戏核心微服务



### 客户背景

小迈是一家致力以数字化领先为优势，实现业务高质量自增长的移动互联网科技公司。为用户开发丰富的工具应用、休闲游戏、益智、运动等系列的移动应用。累计开发400余款产品，累计用户下载安装量破七亿。目前在SAE上部署了十几款游戏APP，比如爱上猜成语、成语最强答人、我找茬贼快、答多多、欢乐找找茬、多多短视频等。

### 业务痛点

游戏有非常大的波峰波谷，休闲游戏都是无状态的，可以拆分成不同的服务模块来提升服务性能和质量，所以对系统有以下核心诉求：

- 系统的稳定性和容灾能力
- 平台的自动弹性伸缩能力
- 对微服务架构的支持
- 便捷的发布回滚能力，甚至是不停服升级

### 产品方案

客户核心技术栈是Java Spring Cloud和Dubbo，服务拆分粒度较细，所以对于服务的管理和治理有很高的要求。SAE在微服务领域天然具备完善的服务管理和治理的能力，再加出色的资源弹性能力，完全符合客户的核心诉求。

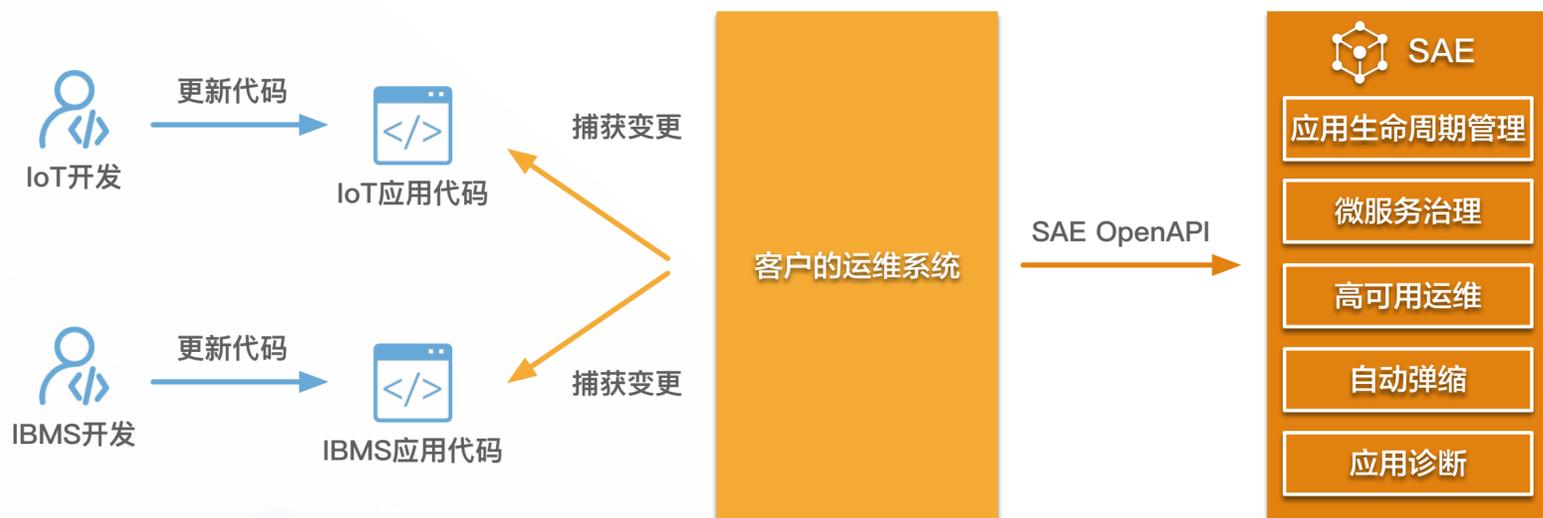
### 达到效果

- 简化运维：只用关注业务逻辑，无需关心底层资源，平台保障集群可用性。
- 成本优化：根据监控指标弹性扩缩，轻松应对流量波动，并节省闲置成本。
- 效率提升：配套的微服务治理能力和白屏化的灰度发布能力，极大提升了发布效率。

### 可复制性

全国、全行业可复制。

## 头部地产企业 基于SAE快速实现业务创新



### 客户背景

某头部地产企业，业务涵盖地产开发、商业运营、租赁住房、智慧服务、房屋租赁、房屋装修六大主航道业务，并积极试水养老、产城等创新领域。覆盖13大业态，完成一、二线核心城市及三、四线潜力城市的布局，业主满意度连续十二年超过90%，用“满意+惊喜”的高品质服务赢得客户的广泛认可。

### 业务痛点

客户 IoT、IBMS等新业务发展快，原有的开发和运维模式匹配不上业务进度。期望能降低运维成本，发布自动化，低门槛拥抱K8s+微服务+Serverless技术红利。

### 产品方案

客户有完整的运维团队，有自建的运维系统，通过SAE提供的丰富的Open API，将SAE的能力集成到客户自己的运维系统中，在不改变任何操作习惯的情况下完成了整个底层资源的切换。

### 达到效果

- 快速交付：免运维，聚焦核心业务的开发，提升交付效率。
- 节省成本：无需固定保有IaaS主机，按需使用，按分钟计费。
- 一站式体验：打通微服务生态各产品，应用监控、DevOps、其它阿里云基础产品（SLB、NAS等）。
- 零改造：客户0代码改造迁移Spring Cloud应用。

### 可复制性

全国、全行业可复制。

# 经典案例



## Timing App 免运维，低门槛微服务化



### 客户背景

Timing是一款在线学习的社交类App，帮助社区用户提升学习凝聚力，制定与达成学习目标。主要功能包括自习室、打卡工具、视频直播、学习服务，契约群、学习日记等。目前已有2000万的活跃用户，日均百万DAU。

### 业务痛点

- 迭代效率慢：传统开发模式无法支撑公司业务的快速发展。
- 系统维护成本高：均采用开源自研，但无运维人员。
- 稳定性风险：部分开源产品维护力度有限，社区不活跃。
- 技术成本高，资源使用浪费：K8s+微服务学习曲线陡峭，希望能快速上手。业务有突发流量，长期保有资源浪费大。

### 产品方案

客户初始的技术栈是单体PHP，随着业务发展单体PHP的弊端逐渐显露。之后客户结合SAE微服务的能力，用一个月时间将PHP技术栈换成Java Spring Cloud，将核心业务都迁移到SAE。另外客户没有运维团队，得益于SAE免运维的能力，成功落地运转起来DevOps。

### 达到效果

- 提效率：屏蔽IaaS+K8s运维，低门槛微服务化，基于Arms快速发现诊断问题。
- 业务稳定：基于自动弹性，无需容量规划，轻松应对洪峰流量。
- 降成本：无需自建微服务组件，灵活弹缩，节省35%硬件成本。

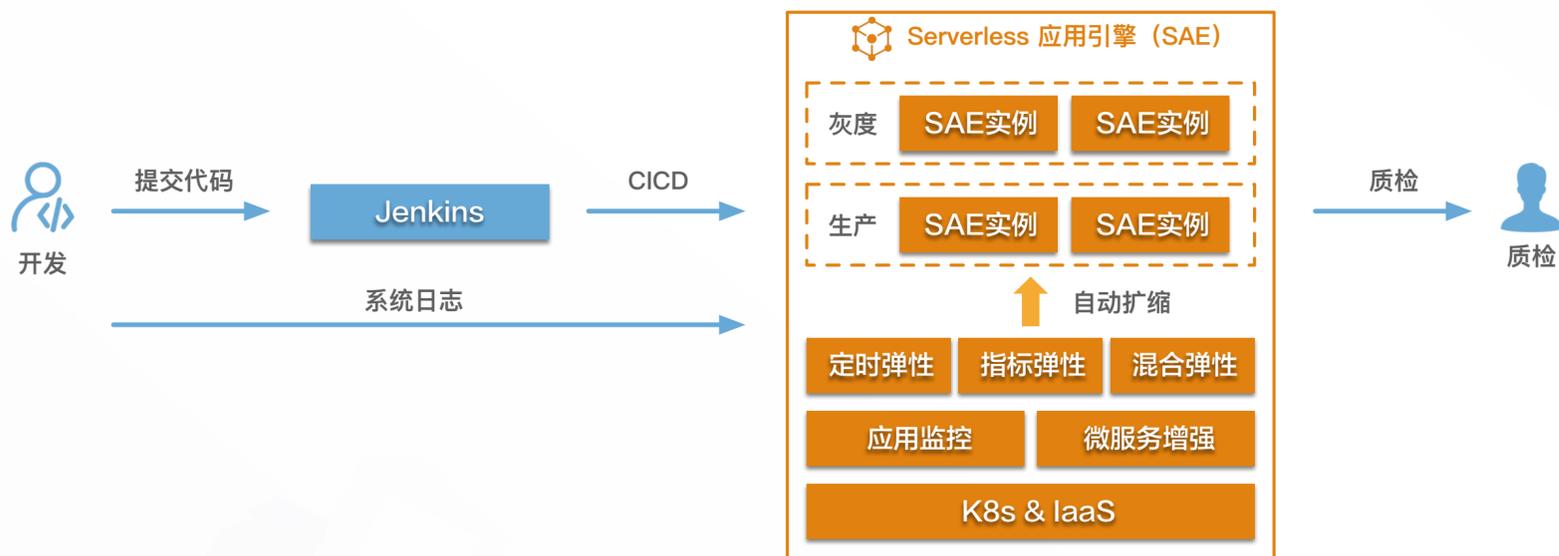
### 可复制性

全国、全行业可复制。

# 经典案例



## 视野数科 SAE + Jenkins打造云原生DevOps 运维效率提升60%



### 客户背景

视野数科是专注服务于一级市场、公司信贷、产业规划招商、面向多层次资本市场的大数据金融信息服务商。深度理解大型银行/金融机构/政府/企业集团等用户对于数据的需求，为客户呈现专业加工后的数据，提供企业/产业/财务数据整合、数据加工处理、数据中台、系统开发和大数据模型分析咨询服务的一整套解决方案。

### 业务痛点

- 开发迭代效率慢：单体烟囱式架构，代码耦合度高，开发效率慢。
- 上线流程复杂，成本高：使用SVN代码管理 + 人工部署，缺少规范化DevOps流程，每次上线前都要在灰度环境来回折腾20~30次数据校验，频繁发版测试，开发和运维幸福感差。
- 容器化自建成本高：尝试过ECS+Docker方案，但Docker Daemon升级、镜像registry管理、配置管理、调度打散等均需要自维护/研发。
- 容量预估无法自动化：每次客户侧有营销活动/重要事件（如新华财经金融排名等），需提前一周告知视野数科备容ECS，存在备容不准风险和闲置浪费问题。

### 产品方案

针对这些问题，视野数科借助阿里云Serverless应用引擎（SAE），完成两方面的升级：

- 从0到1打造Git+ Jenkins + SAE的云原生DevOps体系。
- 零门槛从ECS单体架构，升级为微服务 + K8s + Serverless架构，一步迈进云原生。

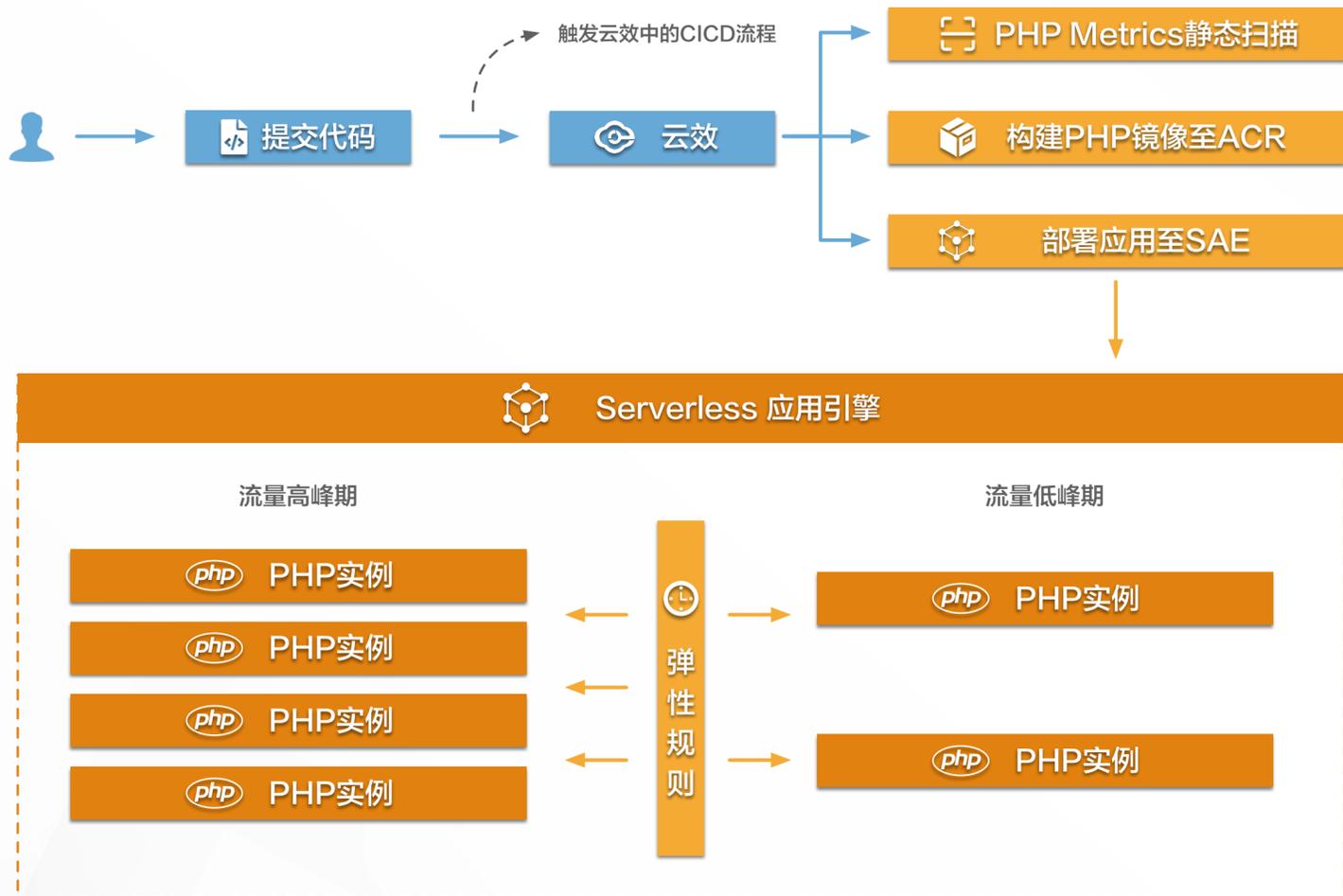
### 达到效果

- 标准化部署流程，降低运维成本：借助SAE，开发自助完成CICD，提升发版效率。运维人员得到彻底解放，从重复繁杂的垂直业务上线事务中抽离出来，专注于横向高可用体系和监控平台的建设，运维效率提升60%。
- 低门槛微服务架构转型：将业务拆成微服务之后，无需关心微服务周边配套和稳定性容灾，SAE提供了开箱即用的历经双11考验的全套微服务治理能力。
- 无需容量规划，秒级扩容：借助SAE灵活多样的弹性策略，视野数科再也无需提前备容，营销活动峰值时秒级自动扩容，峰谷时回收资源降本。
- 无感拥抱K8s，查看日志更方便：借助SAE零门槛容器化，沿用原有部署方式管理应用。不用再care堡垒机权限、机器账号密码等，直接通过SAE + SLS方便查看日志。

### 可复制性

全国、全行业可复制。

## 升学在线 从容应对流量高峰



### 客户背景

升学在线是武汉升学在线科技股份有限公司旗下品牌，拥有6000余所签约高中、1200余所合作高校的升学就业服务平台。升学在线旗下拥有以校内讲学、校外研学、平台测评等形式构建的高中升学规划教育服务体系，一站式解决高校和学生就业、企业校招难题的高校就业管理服务体系。

### 业务痛点

客户整体的流量会比较大，而且有不定期的流量潮汐现象。另外客户的整体技术栈是PHP单体应用，部署在ECS上，在流量潮汐的场景下，客户目前做不到应用横向快速扩展，从而导致应用系统在流量高峰期出现性能瓶颈，影响用户体验。所以客户的核心诉求是期望在流量潮汐的场景中，让应用系统具有应对突发流量快速横向扩展的能力，在资源侧提高应用系统性能。

### 产品方案

为了满足客户的需求，升学在线选择阿里云的Serverless App Engine (SAE) 作为应用系统的部署平台。SAE是面向应用的Serverless PaaS平台，能够帮助用户免运维IaaS、按需使用、按量计费，做到低门槛服务应用上云，并且支持多种语言和高弹性能力。

### 达到效果

- 稳定性：通过镜像加速、冷启动优化等技术，基于Serverless架构的SAE在面对大型应用时也具有极强的水平扩展和底层扩容能力，帮助客户从容应对大数据量、大流量的冲击。
- 降低成本：Serverless架构使客户无需采购和管理服务器等基础设施，降低运维成本。同时SAE分钟级别的按需收费模式，加上极致的弹性伸缩能力，使成本控制达到了最优化。
- 提高人效：客户只需专注业务逻辑的开发，不用分心考虑运维侧的问题，只需要提交业务代码，后续的CI/CD流程均由云效和SAE自动处理。同时SAE提供应用节点维度和应用接口维度的监控功能，帮助客户快速排查问题，极大程度提高了人效。

### 可复制性

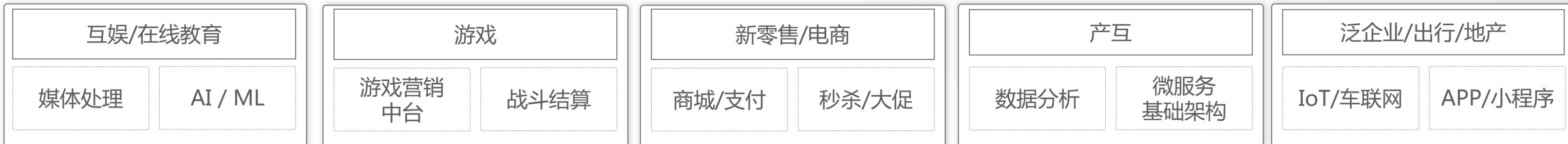
全国、全行业可复制。



# All On Serverless

## 从专用到通用，从复杂到简单

# All On Serverless 从专用到通用，从复杂到简单



## 主打方案



## 核心竞争力



## 函数计算 FC

继云主机、容器服务之后的下一代通用计算平台

提供弹性伸缩、免运维的低成本无服务器运行环境

## Serverless 应用引擎 SAE

面向业务应用架构、微服务架构的一站式应用托管平台

零门槛、零改造、零容器基础即享 Serverless+K8S 带来的技术红利

