

# 基于 Serverless 的 淘宝前端研发模式升级

Brought to you by  
阿里巴巴淘系技术部 七念/吴敏琦 with ❤️

# 极客邦科技 会议推荐2019

5月

**QCon** 北京

全球软件开发大会

大会: 5月6-8日  
培训: 5月9-10日

**QCon** 广州

全球软件开发大会

培训: 5月25-26日  
大会: 5月27-28日

6月

**GTLC**  
GLOBAL  
TECH LEADERSHIP  
CONFERENCE

上海

技术领导力峰会

时间: 6月14-15日

**GMTC** 北京

全球大前端技术大会

大会: 6月20-21日  
培训: 6月22-23日

7月

**ArchSummit** 深圳

全球架构师峰会

大会: 7月12-13日  
培训: 7月14-15日

10月

**QCon** 上海

全球软件开发大会

大会: 10月17-19日  
培训: 10月20-21日

11月

**GMTC** 深圳

全球大前端技术大会

大会: 11月8-9日  
培训: 11月10-11日

**AiCon** 北京

全球人工智能与机器学习大会

大会: 11月21-22日  
培训: 11月23-24日

12月

**ArchSummit** 北京

全球架构师峰会

大会: 12月6-7日  
培训: 12月8-9日

# InfoQ官网 全新改版上线

促进软件开发领域知识与创新的传播



关注InfoQ网站  
第一时间浏览原创IT新闻资讯



免费下载迷你书  
阅读一线开发者的技术干货

# 七念 / 吴敏琦

- 淘系技术部 Node.js 基础架构团队负责人
- 阿里巴巴前端委员会 Node.js 方向负责人
- Node.js 地下铁线下沙龙创始人



# 基于 Serverless 的 淘宝前端研发模式升级

# Agenda

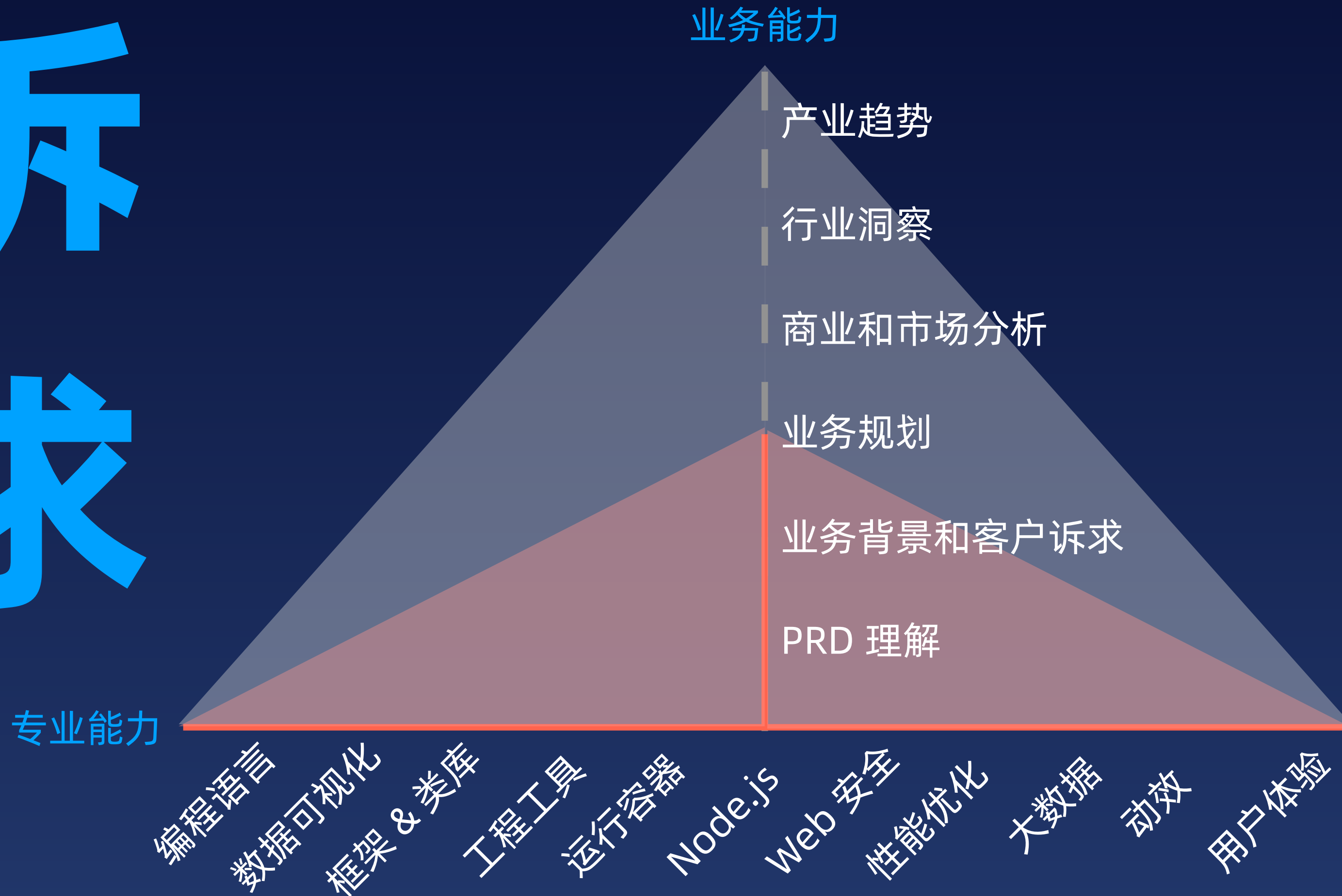
- 4 个自问自答
  - Q1: 为什么我们需要进行前端研发模式升级?
  - Q2: Why Serverless?
  - Q3: 淘系前端如何通过 Serverless 进行研发模式升级?
  - Q4: 下一步怎么搞?
- Q & A

# Q1：为什么我们需要进行前端研发模式升级？

更多参与到业务交付中

跨技术栈提升研发效能

# 诉求



API

调试

监控

排查

性能  
优化



# 研发模式升级的探索一直在进行

- Node.js
- 前后端分离
- 康威定律
- 微服务
- DevOps
- 应用治理
- 容器化
- 故障演练



Sandbox  
应用治理平台

# 业务开发的本质是交付服务(service)和功能(functionality)

Service refer to a software functionality or a set of software functionalities  
From wikipedia

# 痛点

但我们却在为  
**Reliability、Scalability、Performance、...**  
支付高昂的成本



68% 的应用 qpm < 50

# Q1: 为什么我们需要进行前端研发模式升级?

**Answer:** 前端需要更多的参与到业务交付中, 获得更大的发展空间; 需要跨技术栈进一步提升研发效能。

**同时,** 不希望引入太多额外的成本。

# Q2: Why Serverless ?

Here **Serverless** Refer to Serverless Computing

# 现象

无服务计算  
增长趋势飞快

巨头云集  
技术快速成熟

## Top Growing Cloud Services

Place	Service	Growth Rate	2017 Use	2018 Use
#1	Serverless	75%	12%	21%
#2	Container-as-a-service	36%	14%	19%
#3	DBaaS SQL	26%	35%	44%
#4	DBaaS NoSQL	22%	23%	28%
#5	DRaaS			

### General Serverless Computing 生态

	Cloud/On-Premise Service/Open Source
Amazon	Lambda, BeanStalk, FarGate, FireCracker, SAM, Chalice
Google	Cloud Functions, App Engine, FireBase, Cloud Run, Kubernetes, KNative
Microsoft	Azure Functions
阿里云	FC, ECI
腾讯云	SCF
Pivotal(Spring)	PFS, Riff
Oracle	Oracle Functions, Fn
IBM	IBM BlueMix Cloud Functions, OpenWhisk
SAP	SAP Cloud Platform Functions
TJ(大神)	Apex
其它	Fission, kubeless, OpenLambda, Serverless, ClaudiaJS, dawson-cli, gordon, ...

# 思潮



业务开发变轻、变薄，聚焦业务逻辑

基础设施变重、变厚、变标准

云原生

# What is Serverless Computing?

Serverless computing refers to the concept of building and running applications that **do not require server management**. It describes a finer-grained deployment model where applications, bundled as one or more functions, are uploaded to a platform and then **executed, scaled, and billed in response to the exact demand needed at the moment**.

## FaaS + BaaS

# 价值

## 资源利用

- 服务器成本
  - 部署策略 (多机房)
  - 部署模型 (流量均衡)
  - 成本模型

68% 的应用  $qpm < 50$

## 运维成本

- 缩扩容 (估不准)
- 治理成本
  - 机房搬迁
  - OS、基础镜像升级
  - 框架升级
  - 中间件、依赖升级

## 其它

- 人力成本
- 机会成本

都交给平台吧！

Hello, 轻量级的业务服务端研发.

## 资源利用

- 弹性缩扩容
  - 极速启动, Scale 到 0
  - 按实际用量付费

## 运维成本

- 自动伸缩
- 将服务的连接、安全、控制和观测功能下沉, 实现全局最优治理
- 多语言友好
- 自动化

## 其它

- 独立进行交付业务



# 梦想

只关心自己的业务交付

Focus on the goal

业务迭代效率高

云端一体

周、天 -> 随时随需

系统运维简单，监控告警完备

DevOps -> NoOps

性能优化、成本优化是平台的事情

甩锅平台

我的创见之外所有想要的皆是你要提供的

God mode、Mummy mode

# Q2: Why Serverless ?

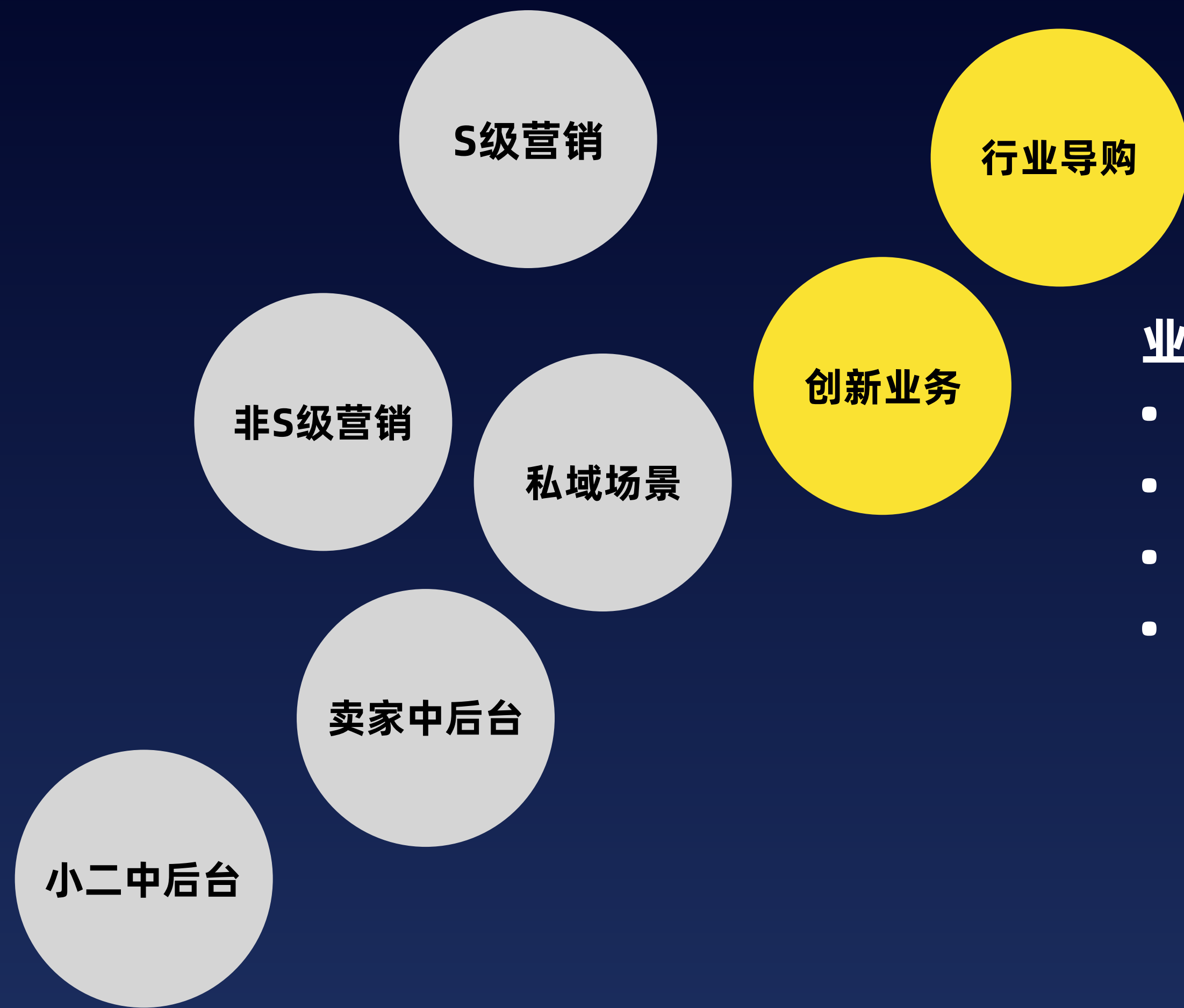
**Answer:** 通过引入 Serverless , 让轻量化的业务服务端研发成为可能, 降低前端参与业务交付的门槛。

**同时**, 也让从云端一体视角重新审视研发效率, 性能优化成为可能。

**Q3:** 淘系前端如何通过 Serverless 进行研发模式升级?

# 业务

用户体验要求



## 业务目标的多样性

- 频道效率
- 用户留存
- 拉新
- 复购

研发成本



**人和货的多元化，让场的组织形式正在变得越来越丰富。**

导购正在升级，围绕私域并不单纯是店铺、商品，围绕公域不再是搜索、首页、营销活动固定入口，而是以轻应用为主阵地，轻应用在这样的背景下百花齐放，围绕行业长尾业务**快速建场**成为整个行业的核心诉求

# 不仅仅是 BFF

# 目标



而是，独立承载前台业务的开发

# 架构

## 前端页面 (H5、Weex)



诊断能力  
链路分析 错误聚类  
函数流量回放

监控能力: 业务  
白屏化日志  
函数级别监控、报警  
业务级别监控、报警

监控能力: 平台  
Node.js 进程重启  
Sidecar 重连  
Runtime 健康检查

业务网关  
业务 CDN 容灾、业务灰度、业务降级、业务切流

FaaS 网关  
流量感知、HSF/HTTP 触发器、限流

中间件 Sidecar

用户函数代码  
Framework: Midway FaaS  
Function Runtime  
函数容器  
用户函数实例 1

用户函数实例 2

用户函数实例 3

用户函数实例 N  
根据流量和资源自动伸缩扩容

日志收集 Sidecar

容器调度

研发集成闭环  
本地开发套件  
预发集成和定位  
线上发布与管控  
线上切流、灰度、回滚

服务市场  
中台能力黄页  
中台能力巡检  
服务 Mock  
服务定义注册

线上监控/诊断

运行时

研发态

限流

中台能力 (HSF、Tair)

# 运行时降低运维成本



- 弹性缩扩容 (Elastic scaling)
- 容灾 (Disaster recovery)
- ServiceMesh
- Function Runtime/Layer
- 自动监控、报警接入 (Automatic monitoring and alerting integration)
- 业务监控聚合 (Business monitoring aggregation)
- 平台监控对业务屏蔽 (Platform monitoring shields business)
- 链路分析 (Link analysis)
- 流量回放 (Traffic replay)
- 故障演练 (Fault simulation)



# 研发态提升效率：研发闭环



## 从存量 HSF 接口快速生成

标准服务名:

market. getContents

HSF Group:

HSF

HSF Service Name:

com.alibaba.wireless.media.service.Conte

Version:

日常 1.0.0.daily 预发 1.0.0

方法签名:

VideoResultDO getContents(ContentBatch

注意: Interface 和 Abstract Class 无法正常

## market.wireless.media.getContents

服务详情 测试与Mock 使用示例 发布 编辑

### 服务名: market.wireless.media.getContents

服务质量: 监控数据未回流

类型: GenericHSF 最后修改

超时: 未设置 容量: 未设置

入参类型 出参类型 服务

```
1 [
2   {
3     "class": "
4     "hasTag":
5     "bizType":
6     "hasMark":
7     "isCache":
8     "sources":
9       "DEFAU
10        "0
11      ]
12    },
13    "hasCover"
14    "hasItems"
```

### market.wireless.media.getContents

服务详情 测试与Mock 使用示例 发布 编辑

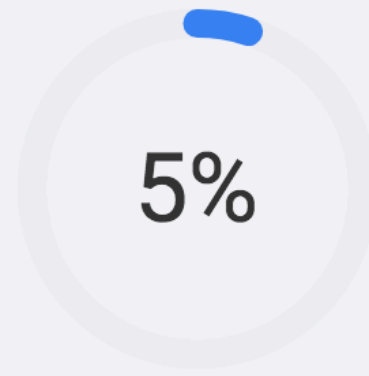
#### Midway-FaaS with Market-Layer

```
1 import { func, FunctionHandler, plugin, provide } from '@ali/midway-faas';
2 import { InvokeType } from '@std/market.wireless.media.getContents';
3
4 @provide()
5 @func('index.handler')
6 export class Fn implements FunctionHandler {
7
8   @plugin('std')
9   std: Std;
10
11   async handler(event) {
12
13     const result = await this.std.invoke<InvokeType>(
14       'market.wireless.media.getContents',
15       [
16         {
17           "class": "com.alibaba.wireless.media.request.ContentBatchRequest",
18           "hasTag": true,
19           "bizType": "",
20           "hasMark": true,
21           "isCache": true,
22           "sources": {
23             "DEFAULT": [
24               "0"
25             ]
26           },
27           "hasCover": true,
28           "hasItems": true,
29           "hasMedia": true,
```

## 灰度比例

- 0%
- 5%
- 10%
- 30%
- 50%

2019-06-11 10:33:56



下一步

## 灰度函数信息

函数名字	latest版本	beta版本
scrolls-client	0.0.2	0.0.2

## 操作

出现问题终止灰度

灰度完成, 切换到新版本

LANCER

© 2018 Theme designed by ICE

函数全部发布成功, 您可以 [切流函数上线](#)

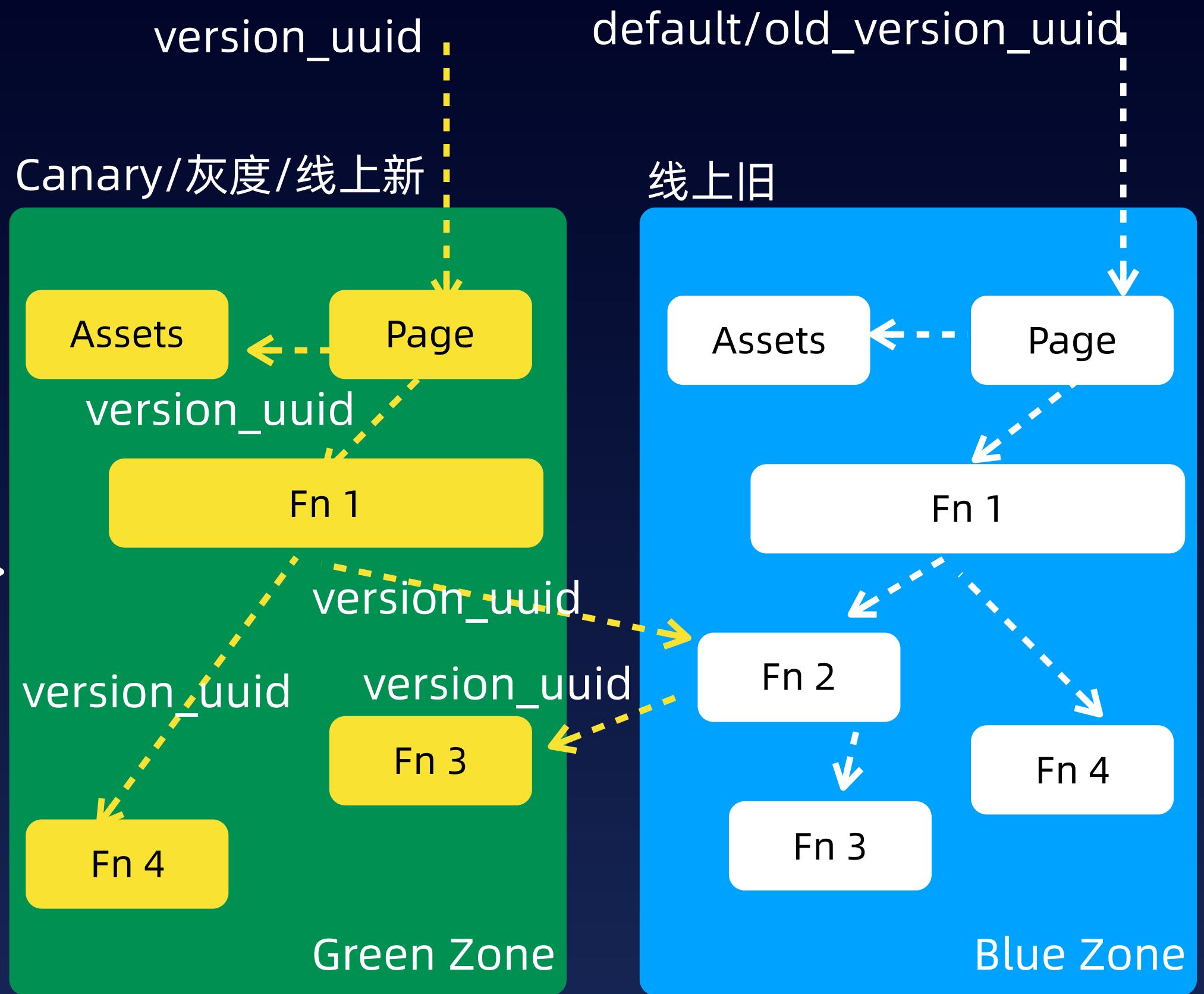
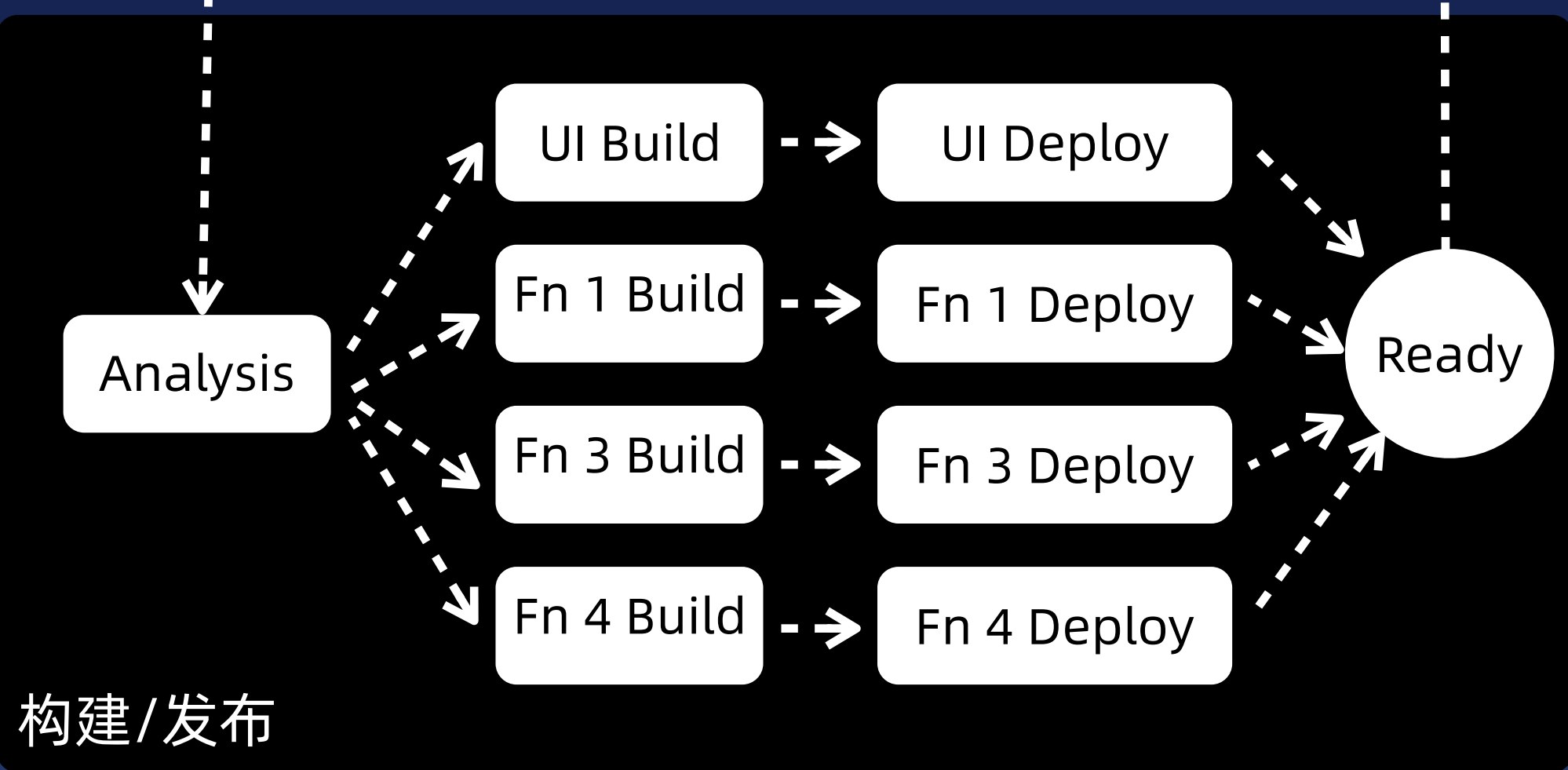
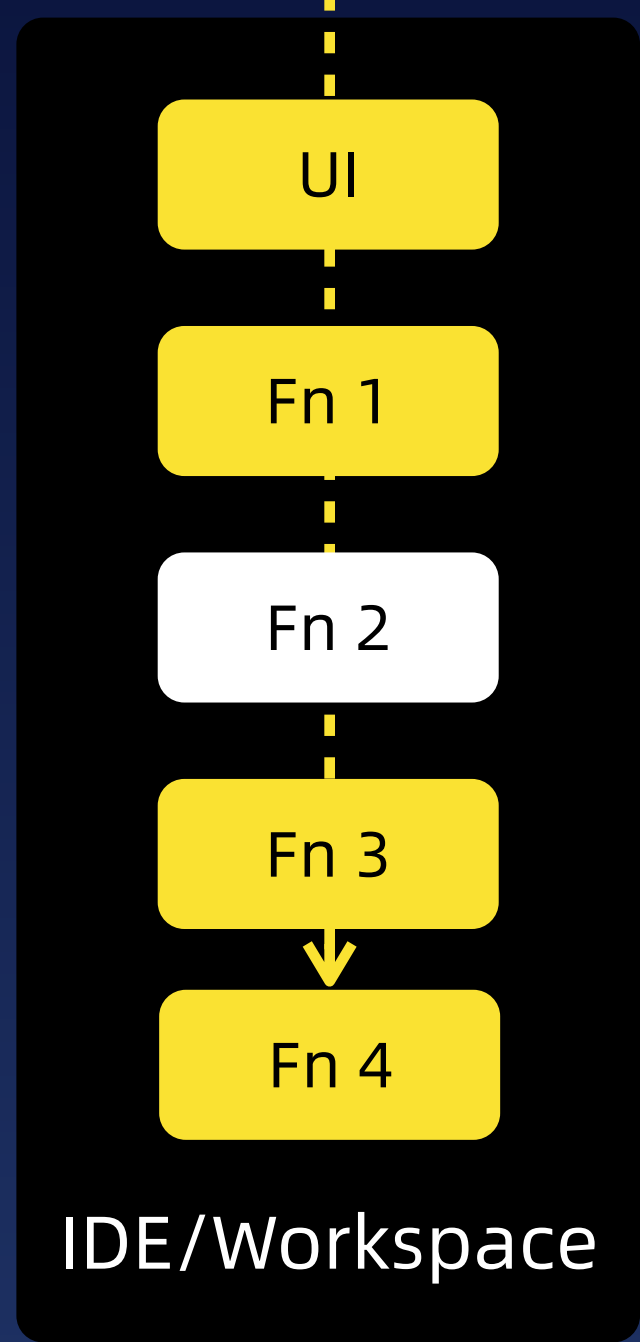
状态	函数	版本	平台函数详细信息	触发器信息	函数 Tar 包	Sandbox 监控
成功	banner	2.0.1	<a href="#">查看</a>	<a href="#">查看</a>	<a href="#">下载</a>	<a href="#">查看</a>
成功	index	2.0.1	<a href="#">查看</a>	<a href="#">查看</a>	<a href="#">下载</a>	<a href="#">查看</a>

The screenshot displays the Chrome DevTools interface. The top panel shows the Sources tab with a file explorer on the left and a code editor on the right. The code editor contains TypeScript code for a module, including imports, exports, and function definitions. The bottom panel shows the Memory tab with a heap snapshot of 35.4 MB. The snapshot table lists objects with their distance from the root, shallow size, and retained size.

Object	Distance	Shallow Size	Retained Size
[1] :: (GC roots) @3	-	0 0 %	2 200 808 6 %
[2] :: global / @1467 □	1	40 0 %	950 832 3 %
[13] :: Node / Parser @183640160 □	1	1 736 0 %	1 736 0 %
[17] :: Node / Parser @183641568 □	1	1 736 0 %	1 736 0 %
[8] :: Node / Parser @210973376 □	1	1 736 0 %	1 736 0 %
[19] :: Node / PipeWrap @183655232 □	1	416 0 %	416 0 %
[20] :: Node / PipeWrap @183533056 □	1	416 0 %	416 0 %
[11] :: Node / TCPServerWrap @210974432 □	1	400 0 %	400 0 %
[5] :: Node / TCPServerWrap @356686976 □	1	400 0 %	400 0 %
[6] :: Node / TCPServerWrap @263648416 □	1	400 0 %	400 0 %
[7] :: Node / TCPsocketWrap @210973024 □	1	400 0 %	400 0 %
<b>Retainers</b>			
Object	Distance	Shallow Size	Retained Size
[6] in Pipe @7131 □	2	32 0 %	72 0 %
[1] in Node / PipeWrap @183655232 □	1	416 0 %	416 0 %
._handle in Socket @383113 □	3	248 0 %	1 104 0 %
1907 in (Global handles) @29	-	0 0 %	730 968 2 %

# 研发态提升效率：蓝绿发布

本地开发、调试





# Q3: 淘系前端如何通过 Serverless 进行研发模式升级??

## Answer:

选择需要快速建场的导购业务进行试点。

以前端独立支撑导购前台业务为目标。

通过**函数运行时**、**SideCar**、**配套监控诊断能力**的研发降低运行时运维成本。

通过**研发集成闭环**、**蓝绿发布**、**前端服务市场**提升云端一体的研发效率

Q4: 下一步怎么搞?



## 函数运行时 和框架

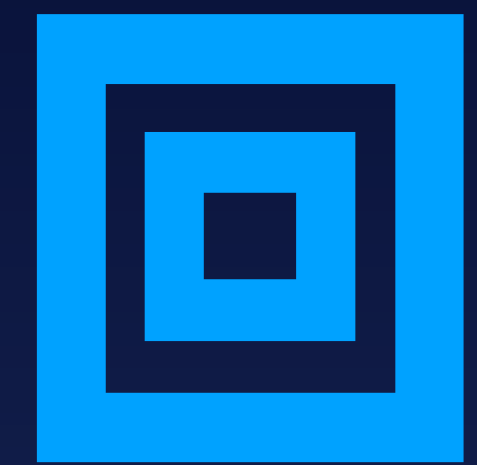
- 极速启动
- 标准化
- 研发模式沉淀为解决方案
- 杀手级 feature

## 高可用 保障

- 双促保障
- 流量录制和回放
- 故障演练

## 前端研发 工作台

- 可视化编排
- 一站式
- 场景化



# 顾

- Q1: 为什么我们需要进行前端研发模式升级?
- Q2: Why Serverless?
- Q3: 淘系前端如何通过 Serverless 进行研发模式升级?
- Q4: 下一步怎么搞?

# 前端训练营

用3个月时间，彻底学透前端开发必备技能



了解详情

- ✓ 线下线上混合式学习
- ✓ 名师手把手教学
- ✓ 一线大厂项目实操
- ✓ 毕业即享内推服务



讲师·程劭非 (winter)  
前手机淘宝前端负责人

# 重学前端

每天10分钟，重构你的前端知识体系

你将获得

告别零散技术点，搭建前端知识体系

打通JS、HTML、CSS、浏览器4大脉络

40+ 前端重难点完全解答

大厂前端工程实战演练



作者：winter (程劭非)

前手机淘宝前端负责人



扫码立即参与

到手价 **¥69** 原价~~¥99~~ (仅限 **48** 小时)

参与拼团，结算时输入【GMTC用户专享优惠口令】：**2qianduan**

THANKS

GMTC  
全球大前端技术大会