

字节跳动数据埋点验证实践

赵志荣 火山引擎大数据工程师



为什么做埋点验证

指标计算有问题？

埋点没有上报？

推荐不准确？

埋点上报不准确？



目录 CONTENT

01 埋点验证流程

02 技术架构

03 技术挑战

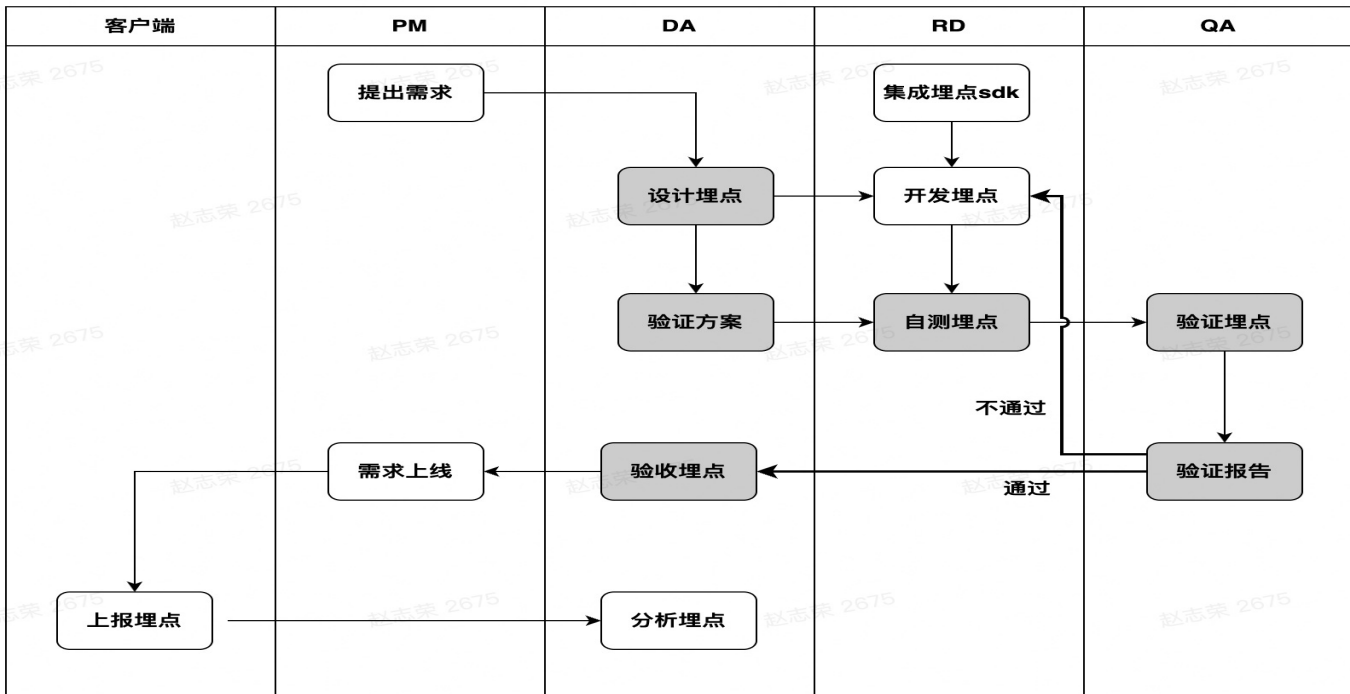
04 规划&展望

埋点验证流程

0



埋点验证流程



- 埋点生命周期：4个角色(PM、DA、RD、QA)+6个节点(提出需求、设计埋点、开发埋点、测试埋点、上报埋点、分析埋点)
- 埋点验证流程：3个角色(DA、RD、QA)+3个节点(设计埋点、测试埋点、验收埋点)+3个物料(埋点验证方案、埋点验证工具、埋点验证报告)

埋点验证流程

- 产品流程

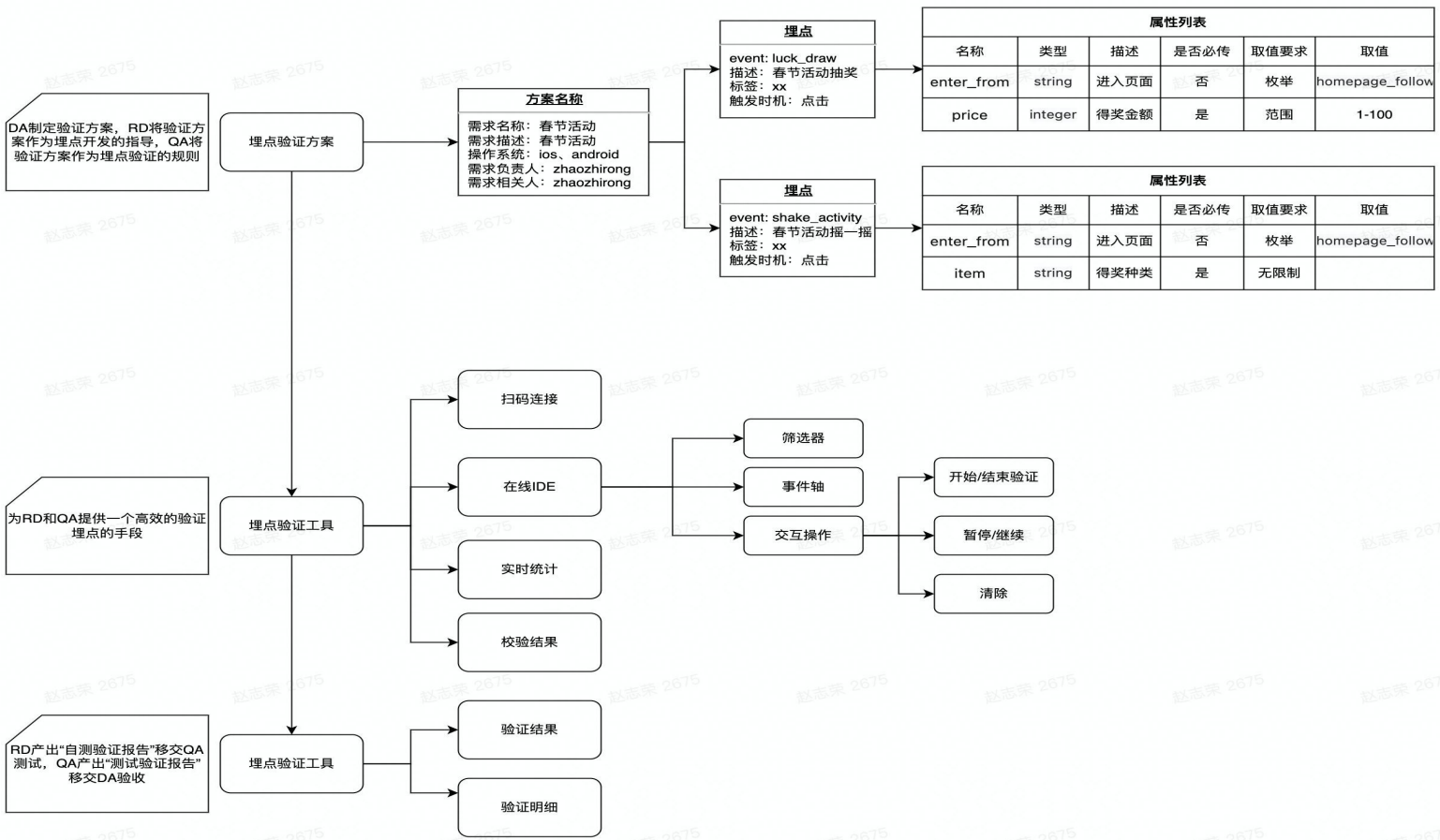
埋点验证
方案

埋点验证
工具

埋点验证
报告

埋点验证流程

- 产品流程



- 埋点验证工具

🚨 异常 2 (🚫 禁止上报 2) ✅ 成功 0 ⏳ 未验证 0

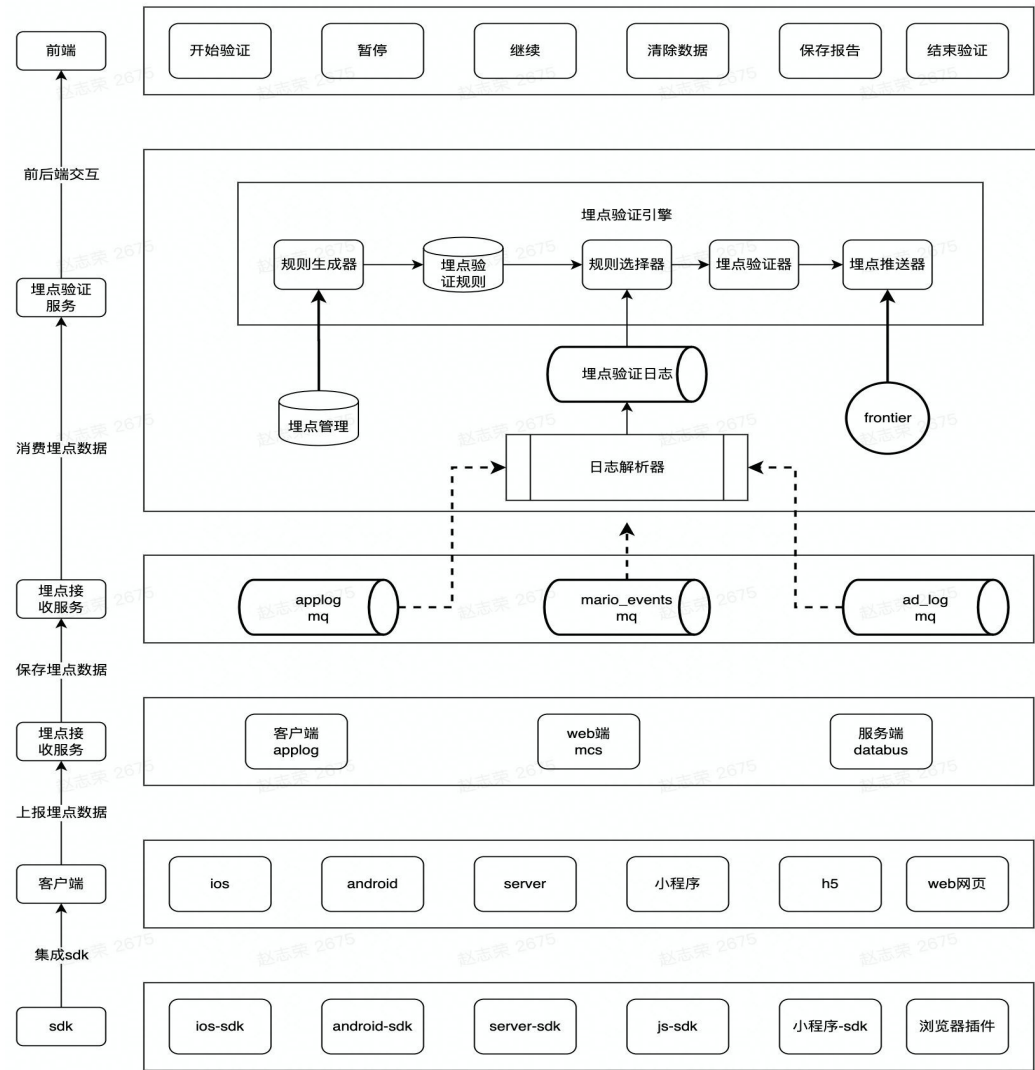
技术架构

02



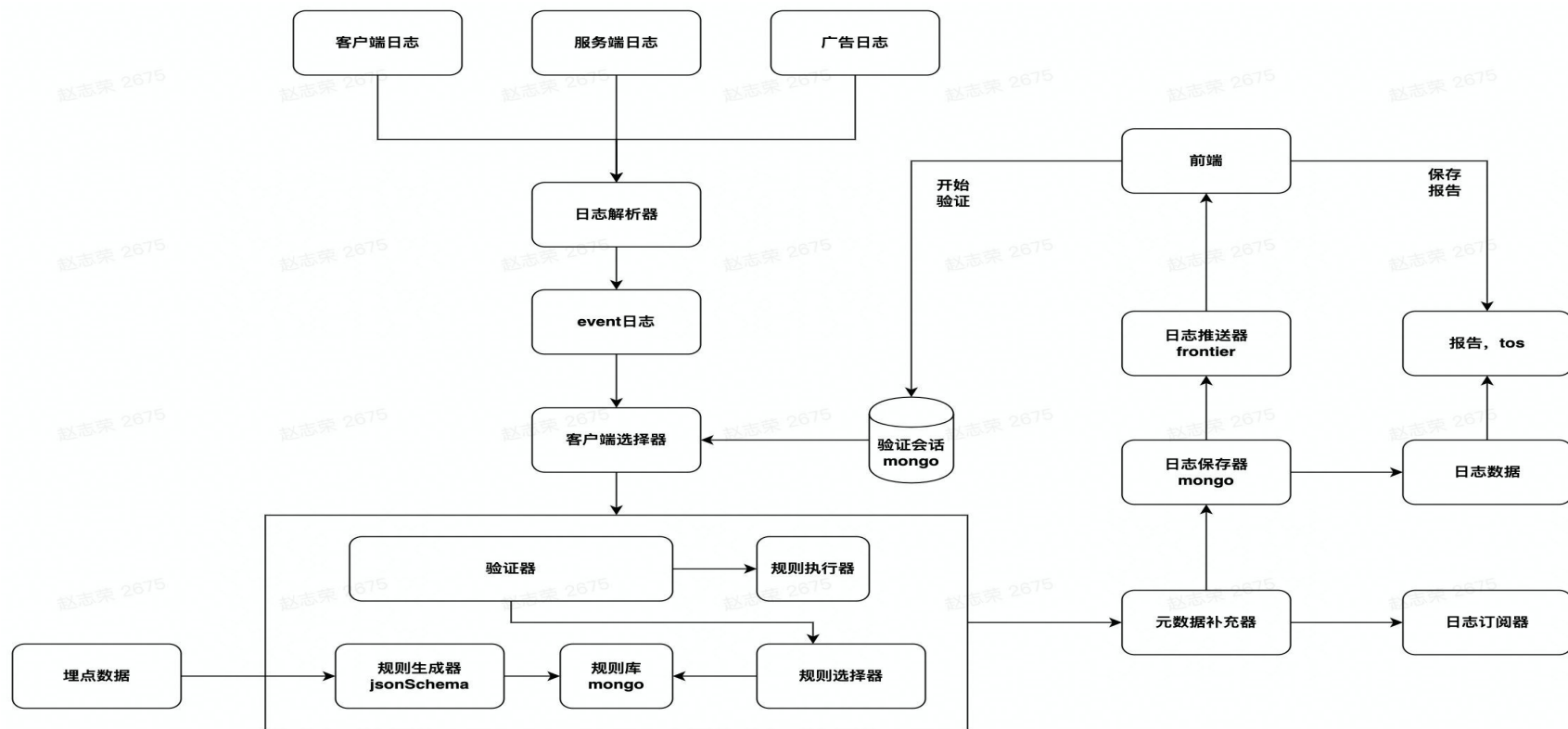
技术架构

- 技术架构图



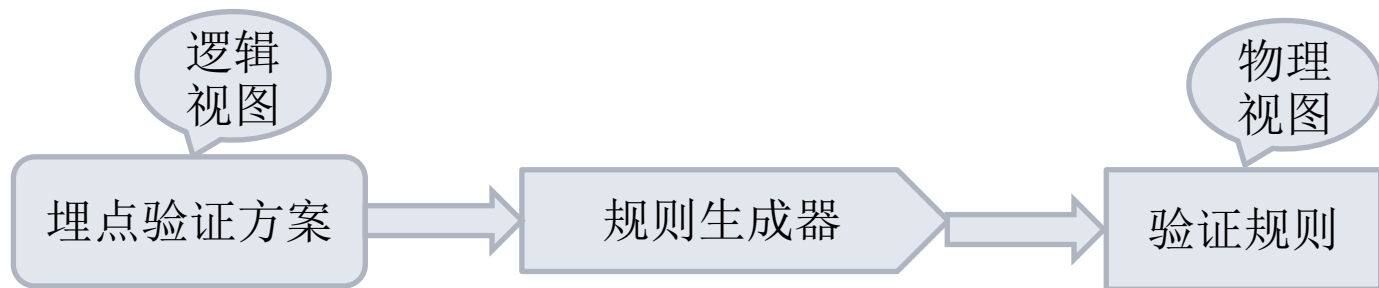
技术架构

- 埋点验证引擎



技术架构

- 埋点验证引擎-规则生成器



```
{
  "app_id": 100,
  "event": "click",
  "params": {
    "enter_from": "login",
    "duration": 1,
    "type": 3
  }
}
```

jsonSchema

```
{
  "app_id": 100,
  "event_name": "video_play",
  "logical_filter": {
    "enter_from": "login"
  },
  "meta": {
    "required_field": [
      "duration",
      "enter_from",
      "type"
    ],
    "scene": {
      "condition": "enter_from=login",
      "name": "登录页"
    },
    "validate_field": [
      "duration",
      "enter_from",
      "type"
    ]
  },
  "physical_validation": "{ \"$schema\": \"h",
  "source": "schema_scene"
}
```

技术挑战

03



易用性

快速接入埋点验证，快速开始埋点验证

准确性

埋点验证结果准确、用户可信

实时性

埋点数据实时可见

稳定性

埋点数据可靠不丢失

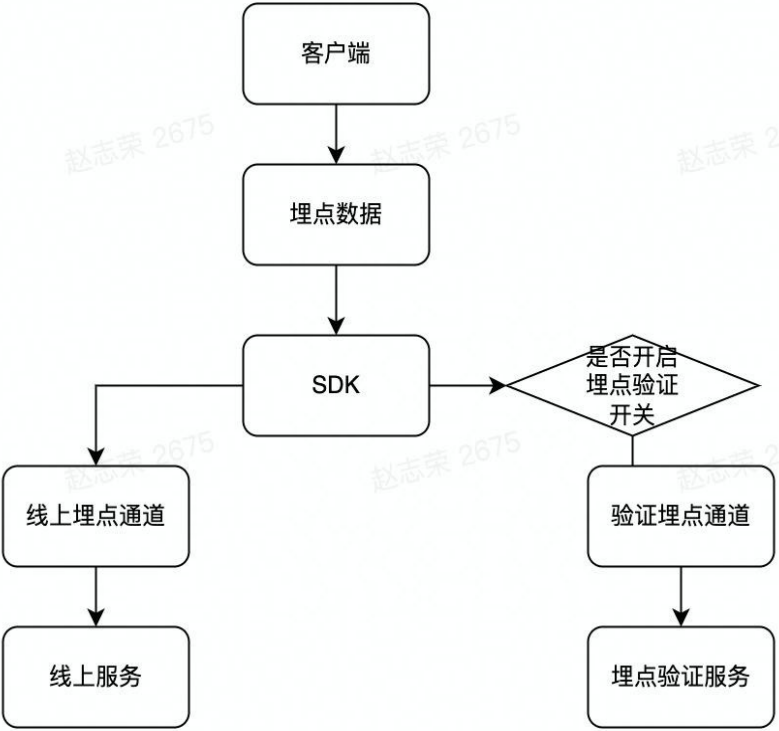
扩展性

快速接入新的埋点数据格式

技术挑战

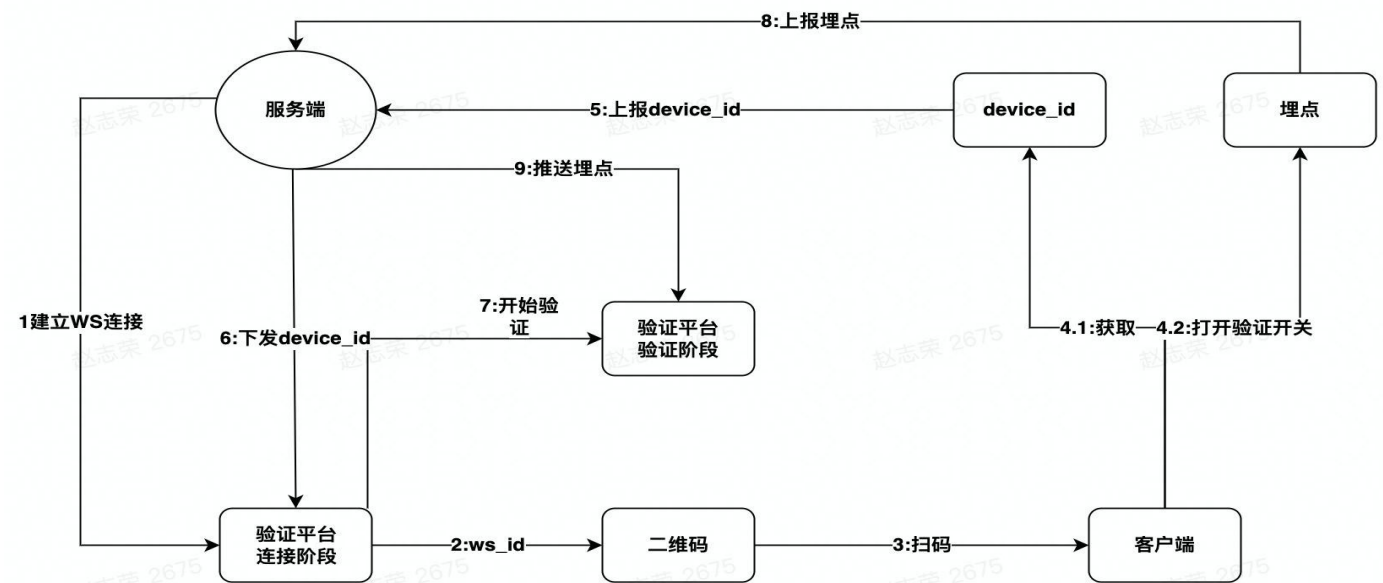
易用性-sdk

端	SDK	如何开启埋点验证开关
客户端	Android 、 IOS SDK	Android、IOS提供API，开关默认是关闭的，业务侧集成 的时候可选择在“域内测试包”打开此开关
服务端	Go 、 Java 、 Python	服务端会自行判断是否是非线上环境，如果是非线上环 境，会默认开启“埋点验证开关”
web端	JS SDK、 浏览器插件	JS SDK采用和客户端SDK一样的逻辑 为了使用方便，我们也提供了浏览器插件，用户只需打 开此插件即可，无需关注“埋点验证开关”



技术挑战

易用性-扫码链接



1、建立WS连接：服务端和验证平台建立长连接，用于通信	2、ws_id：验证平台根据ws_id生成二维码
3、扫码：客户端扫描二维码	4、获取并打开验证开关：客户端获取设备信息并且打开埋点验证开关
5、上报device_id：客户端将长连接信息和设备信息上报至服务端	6、下发device_id：服务端将设备信息推送到验证平台
7、开始验证：埋点验证平台进入验证阶段	8、上报埋点：客户端开始上报埋点
9、推送埋点：服务端将埋点推送到验证平台	

技术挑战

准确性

An online, interactive JSON Schema validator. Supports JSON Schema Draft 3, Draft 4, Draft 6, Draft 7 and Draft 2019-09. [View source code](#)

Select schema: Custom

```
1 {
2   "$schema": "https://json-schema.org/draft/2019-09/schema",
3   "type": "object",
4   "properties": {
5     "duration": {
6       "type": "integer"
7     },
8     "type": {
9       "type": "integer",
10      "enum": [
11        1,
12        2,
13        3
14      ]
15    }
16  },
17  "required": [
18    "duration",
19    "type"
20  ]
21 }
```

Input JSON: ✖ Found 1 error(s)

```
1 {
2   "duration": 100,
3   "type": "v3"
4 }
```

✖ Found 1 error(s)

Message: Invalid type. Expected Integer but got String.

Schema path: #/properties/type/type

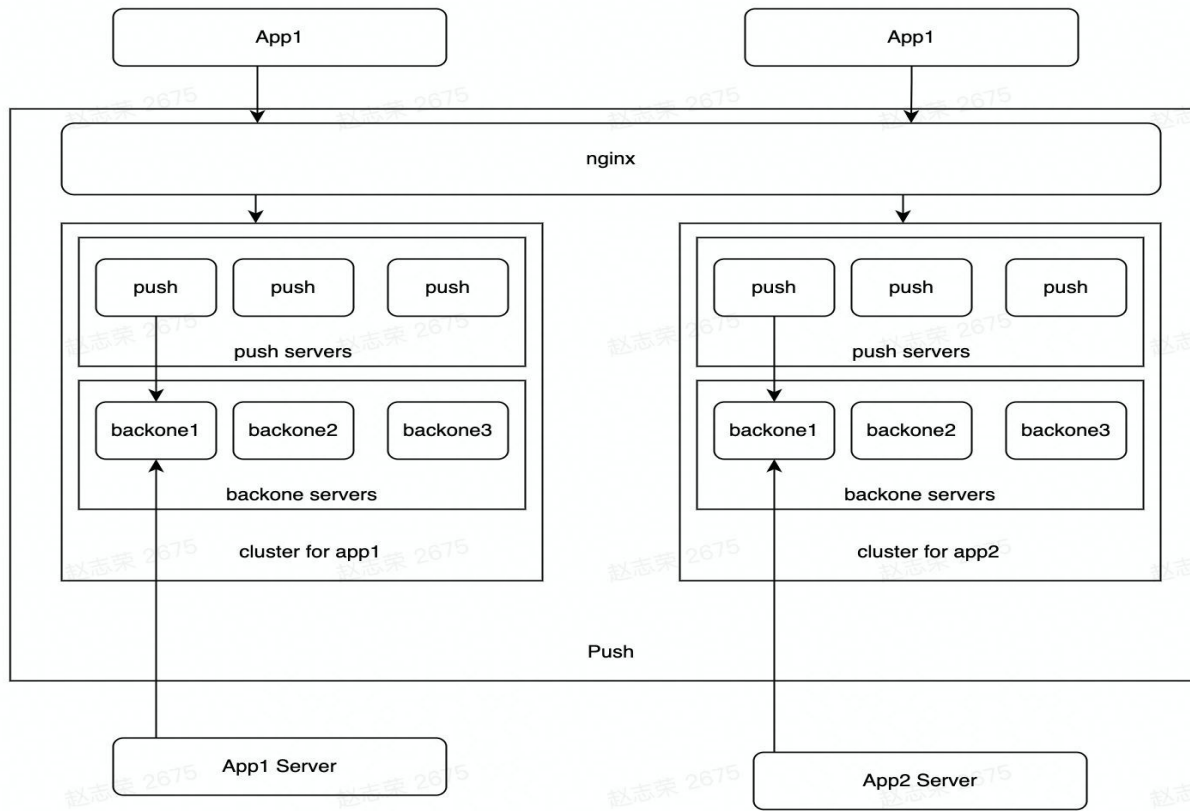
埋点验证引擎必须保证埋点验证结果的准确性，才能降低验证成本。针对埋点数据本身的格式验证，我们采用了JsonSchema作为验证手段，以支持完善的验证规则、可信的验证结果。测试地址：
<https://www.jsonschemavalidator.net/>

技术挑战

实时性

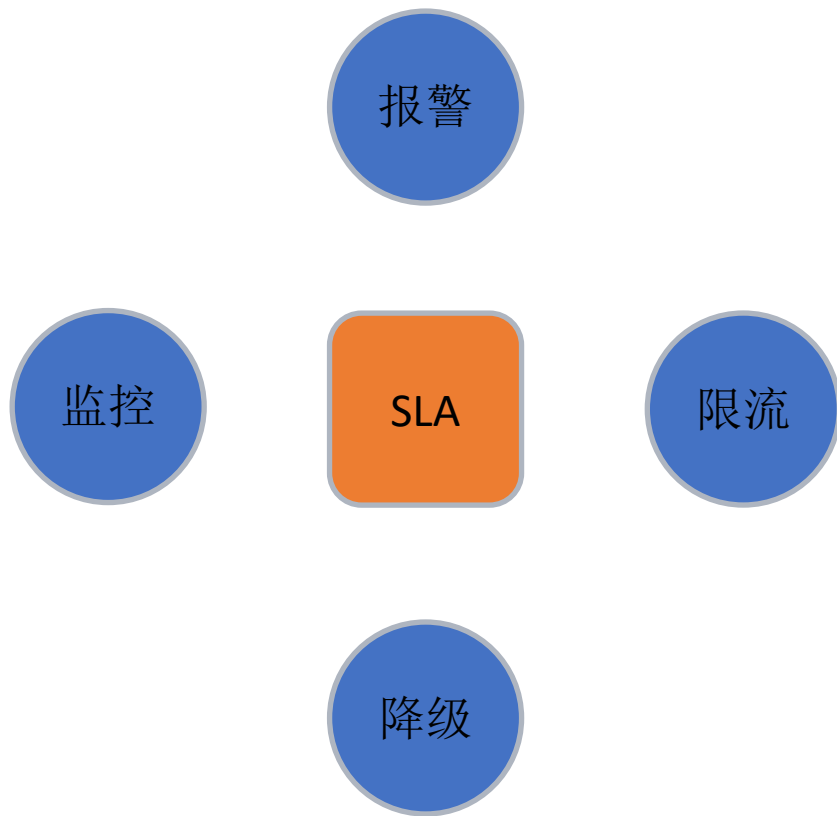
优势

- u 连接稳定性：Push服务分为两个组件Push和Backone，实现了业务和推送解耦。push面向客户端连接，设计尽可能简单，需保持大量客户端活跃连接，避免了业务服务更新时不影响客户端连接
- u 服务隔离性：不同的业务服务接入push服务，会根据接入信息做集群隔离，避免业务之间互相影响
- u 横向扩展性：当业务服务不断增多时，只需对push服务做横向扩容即可支持



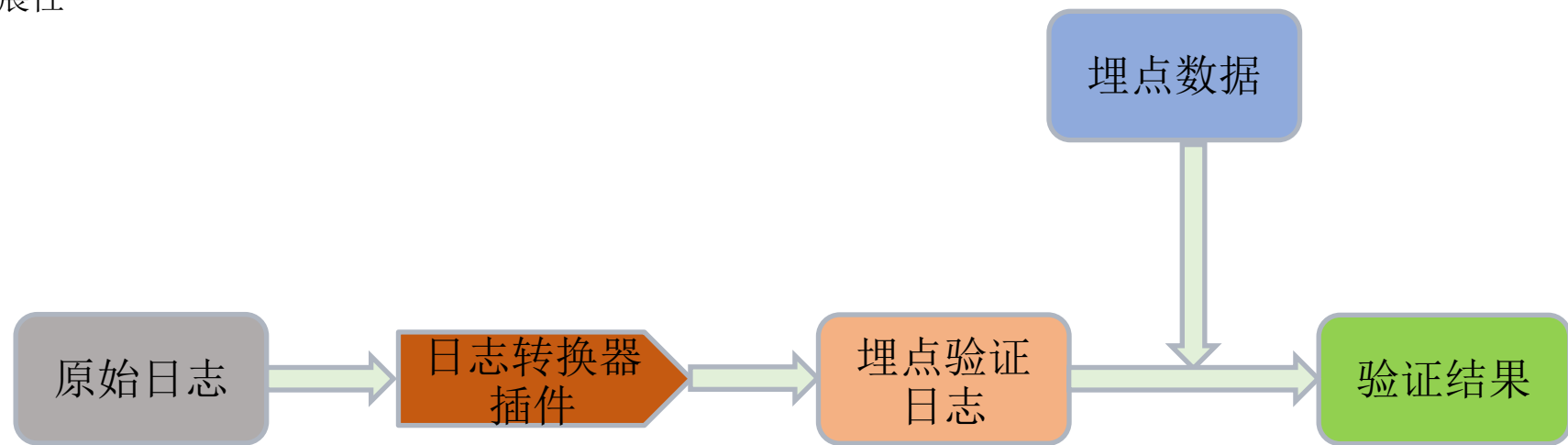
技术挑战

稳定性



技术挑战

扩展性



规划&展望

04



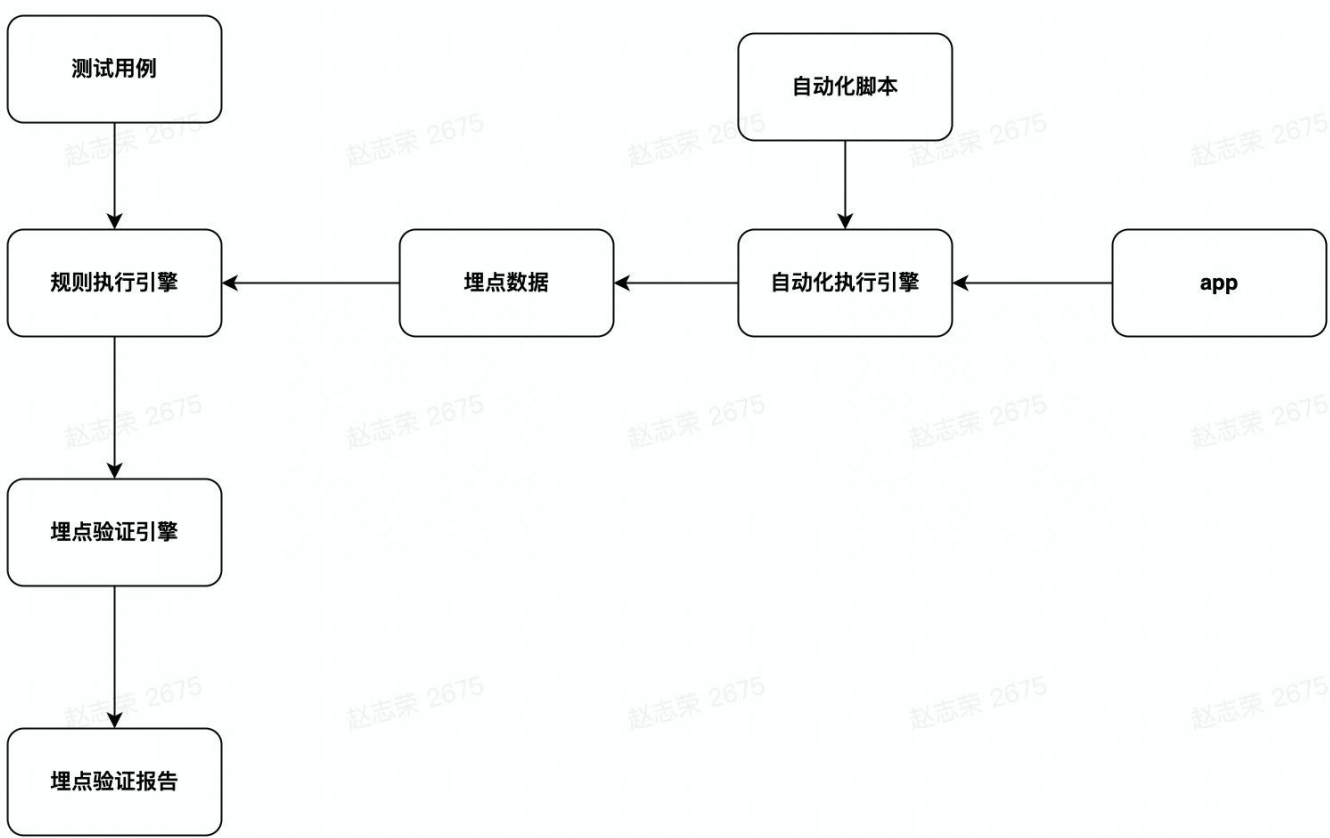
规划&展望

全链路埋点质量保障



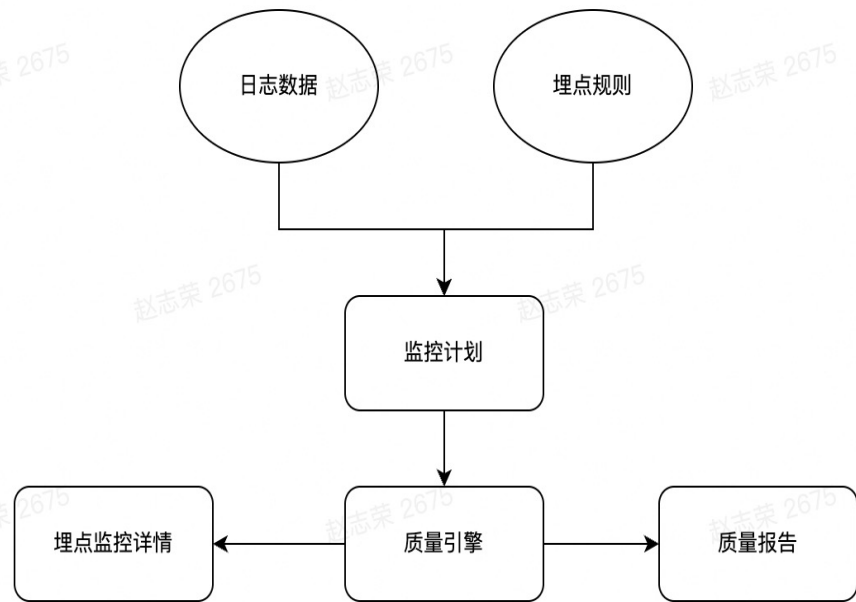
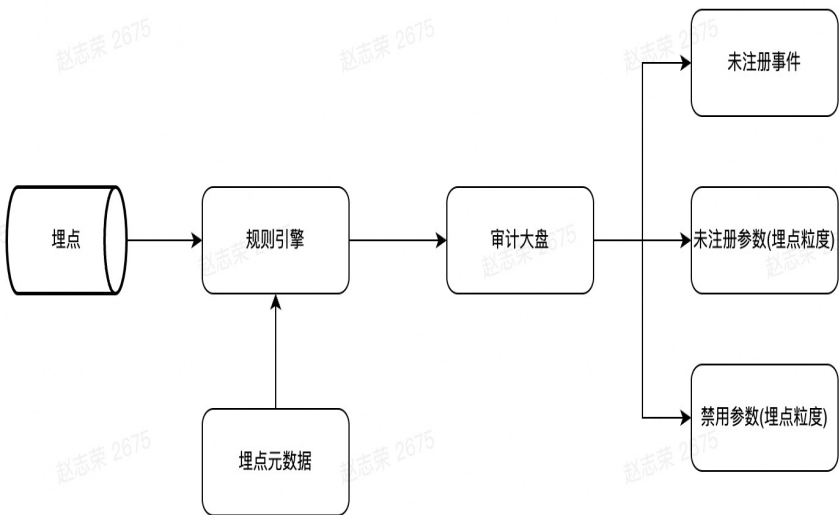
规划&展望

自动化验证



规划&展望

质量监控



欢迎联系我们



扫码关注

“字节跳动数据平台” 微信公众号



扫码添加小助手

进入 “字节跳动数据平台” 官方交流群

非常感谢您的观看



火山引擎



| DataFun.

