# 用户洞察平台介绍

魅族 — 黄振贤

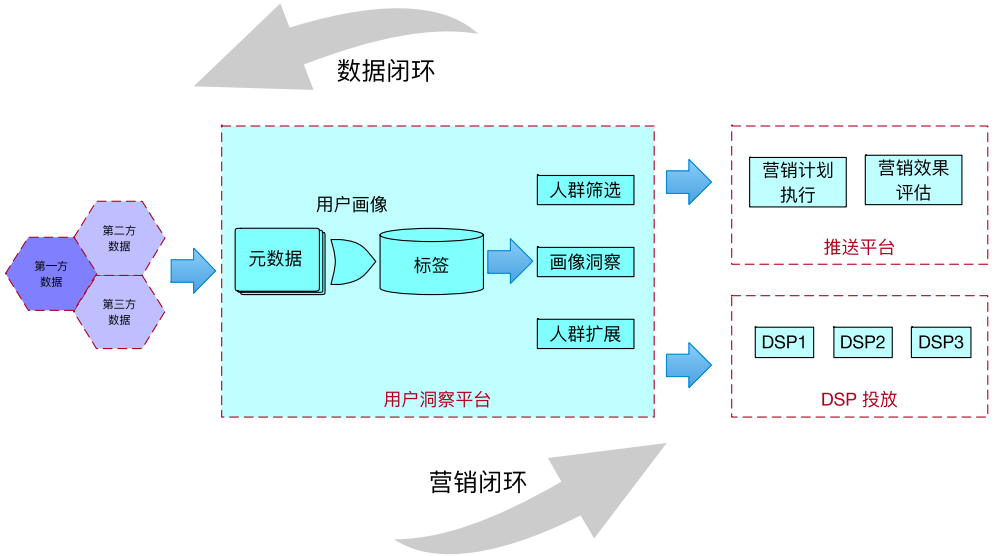
# 目录

# 1.1 用户洞察平台的定位

1. 通过对三方受众数据的汇聚、清洗、智能运算，构建了庞大的精准人群数据中心，提供丰富的用户画像数据以及实时的场景识别力。

1. 无缝对接各类业务平台的数据应用，如广告平台，PUSH推送，个性化推荐之间建立了数据通道，支持公司级的精准营销，消息及时送达服务等场

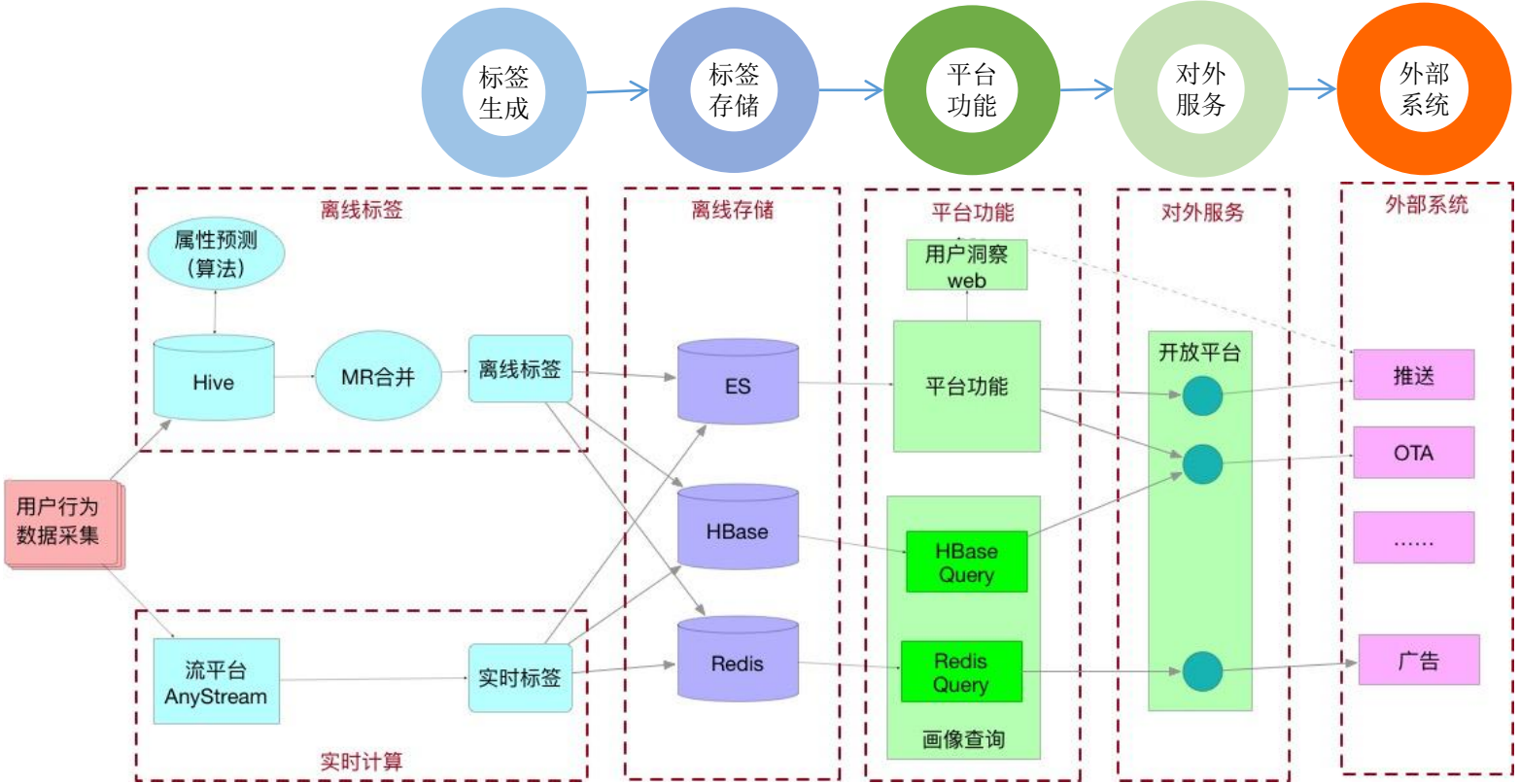
1. 营销效果评估，反馈数据可进一步加工，用于提升画像标签质量



# 1.2 核心需求

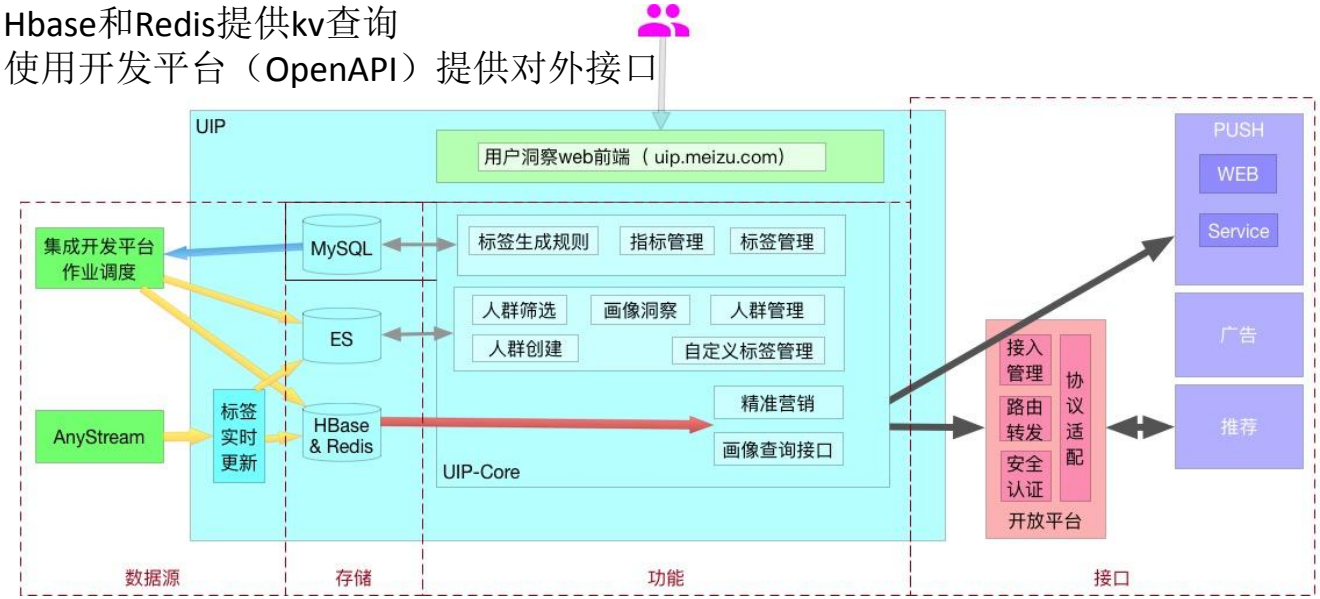

标签生成
互联网业务变化快，标签需求变更频繁
要求系统快速响应标签需求
人群洞察
对全量用户任意标签进行过滤以及聚合计算
查询1-2秒内响应。
受众分发
在无缝对接各类业务系统
实现高效实时的精确营销
标签查询
根据用户ID查询用户画像详情
对于广告业务的查询需要在更苛刻的 50ms 内返回

# 1.3数据流视图


标签生成
标签存储
平台功能
对外服务
外部系统
离线标签
离线存储
平台功能
对外服务
外部系统
属性预测（算法）
Hive
MR合并
离线标签
用户洞察web
平台功能
开放平台
推送
OTA
.....
广告
ES
HBase
Redis
HBase Query
Redis Query
画像查询
用户行为数据采集
流平台AnyStream
实时标签
实时计算

# 1.4 总体架构

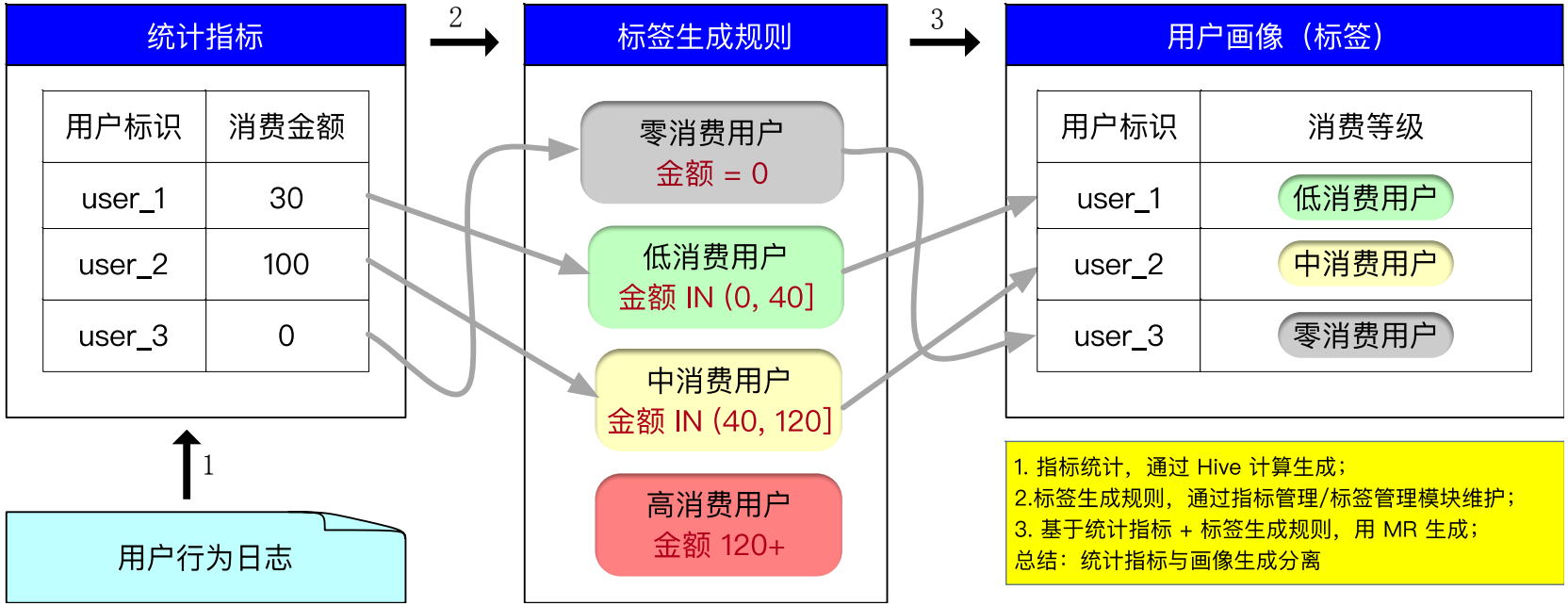
- 集成开发平台之作业调度系统上，配置和运行离线计算任务（Hive&MR)
- 流平台(AnyStream)负责实时标签计算
- 管理模块生成的相关规则，存储在MySQL，供标签生成任务(Hive/MR/流平台）使用
- 用户画像（标签）宽表保存在ES上
- Hbase和Redis提供kv查询
- 使用开发平台（OpenAPI）提供对外接口



# 目录

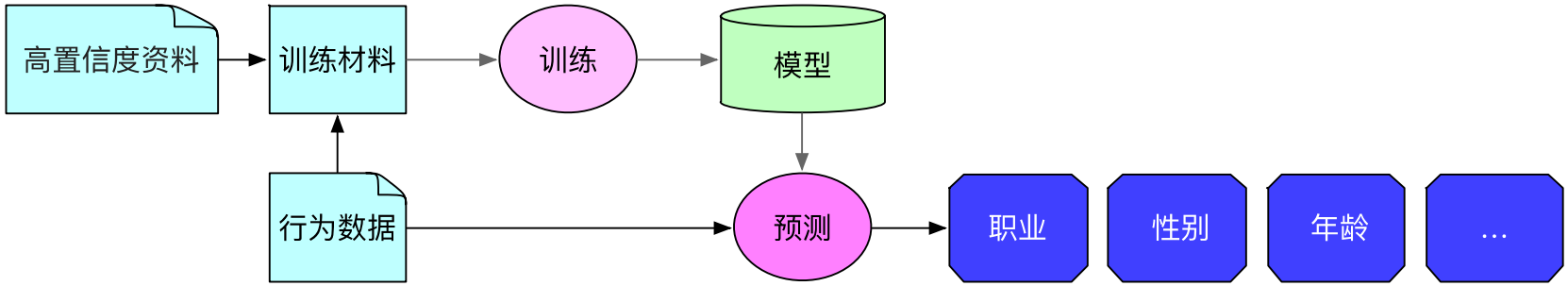
# 2.1 统计类标签

计算模式 行为日志 → 统计指标 → 生成标签



# 2.2 算法类标签计算

- 模型训练：选取高置信度资料（如用户注册信息）+用户行为数据作输入进行模型训练
- 属性预测：使用训练好的模型进行预测



# 2.3 单值标签与多值标签


- 单值标签：

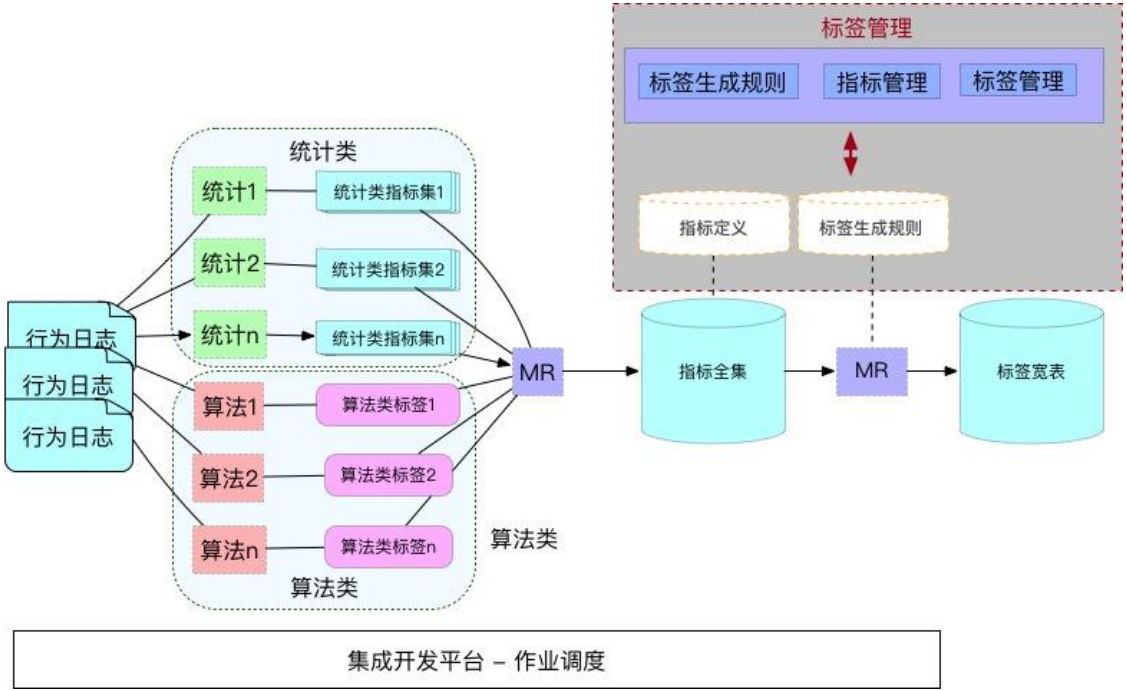
用户在该标签下只能取一个值，不能多选。

- 多值标签：

用户可以取该标签下的多个取值组合。比如用户可以有多个兴趣爱好。

多值标签的存在，会影响存储查询引擎的选型和存储结构设计

# 2.4 标签生成过程




■ 优点：

1、配置化管理，提供 Web UI 管理标签的生命周期

2、基于配置生成标签，标签宽表数据与元数据100%一致

■ 尚存不足

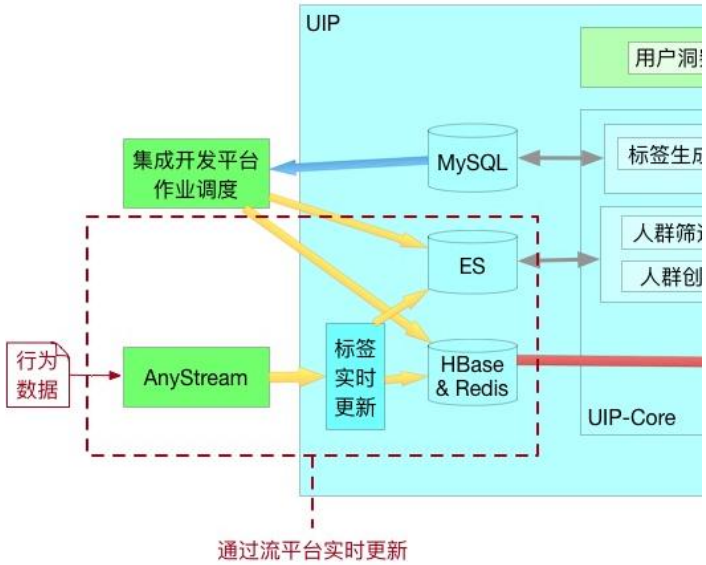
目前配置化管理只涵盖到最终的标签宽表生成。与上游的指标统计和算法预有脱节。

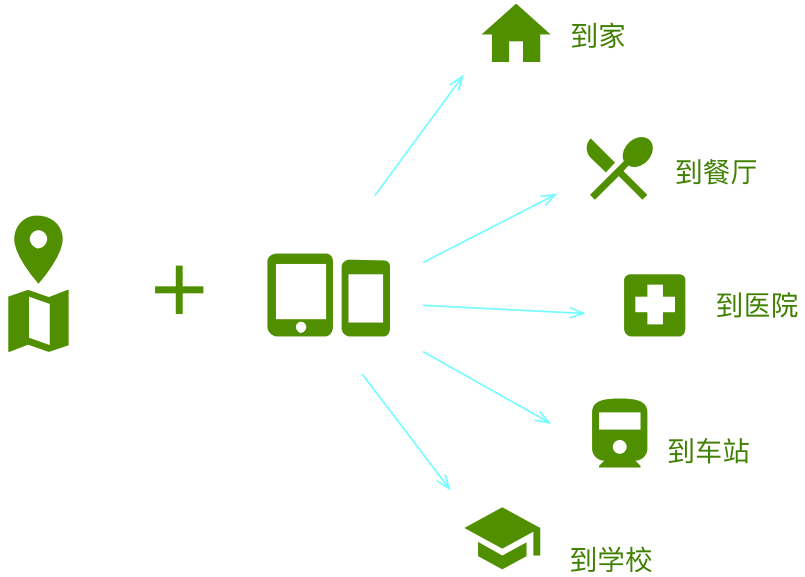
1、上游计算过程是单独开发，指标定义只是另外配置的数据描述（可能存在不一致）

2、一些标签下线（废除）后，相应的上游任务的依赖需要另外废除，否则会遗留无用的作业浪费计算资源

# 2.5 实时标签

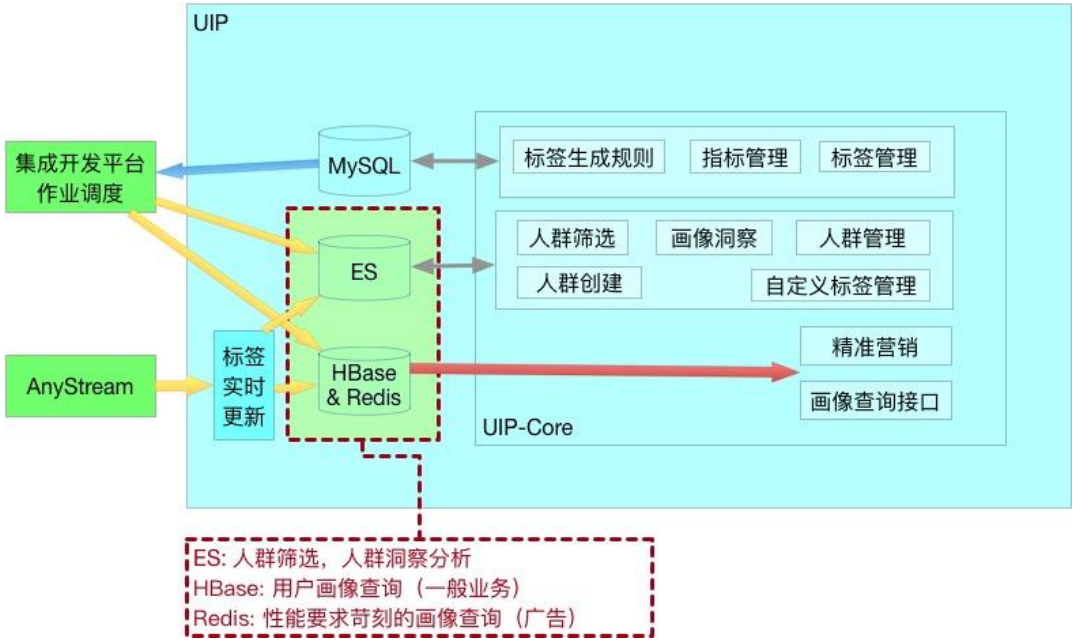
- 实时位置标签，实时分析用户所处的场景（图例）
- 其它实时标签：搜索、支付等



# 目录

# 3.1 标签存储总览




- ElasticSearch（ES）是一个基于Lucene构建的开源、分布式、RESTful搜索引擎。能够达到实时搜索，稳定，可靠，快速。
- 基于 ES 实现对全量用户任意标签进行在线筛选和聚合分析，秒及响应。
- Hbase 提供大吞吐量的 key/value 查询
- 性能要求更为苛刻的 key/value 查询（广告平台）通过使用 Redid 来实现

# 3.2 Why ElasticSearch (ES)

| 历史 | | 现状 | |
|---|---|---|---|
| Vertica | 社区版有 3 个节点和 1T 存储容量限制 | ES | 能够达到实时搜索，稳定，可靠，快速。 |
| | 随着数据规模和调用数暴增，性能出现瓶颈 | | 在线更新（实时/准实时更新） |
| | 对于多值标签，只能采用csv方式保存在varchar字段，性能低下 | | 水平扩展能力强 |
| | 多值标签检索使用字符串 LIKE 操作；聚合虽能通过一些 trick 来支持，但性能太差。 | | array type完美支持多值标签存储和分析场景 |

# 3.3 HBase 与 Redis

## Hbase

- 提供低成本，高吞吐量的 kv 查询
- 满足一般业务的查询
- 缺点是查询响应时间不太理想（针对广告业务而言）

## Redis

- 广告业务提出 50ms 内的查询延迟，这种苛刻要求需用 Redis 实现。
- Redis存储目前只服务于广告平台的查询调用。

- 考虑成本因素，主要使用 Hbase 来提供 KV 查询
- 部分要求苛刻的业务，使用 Redis 作为补充

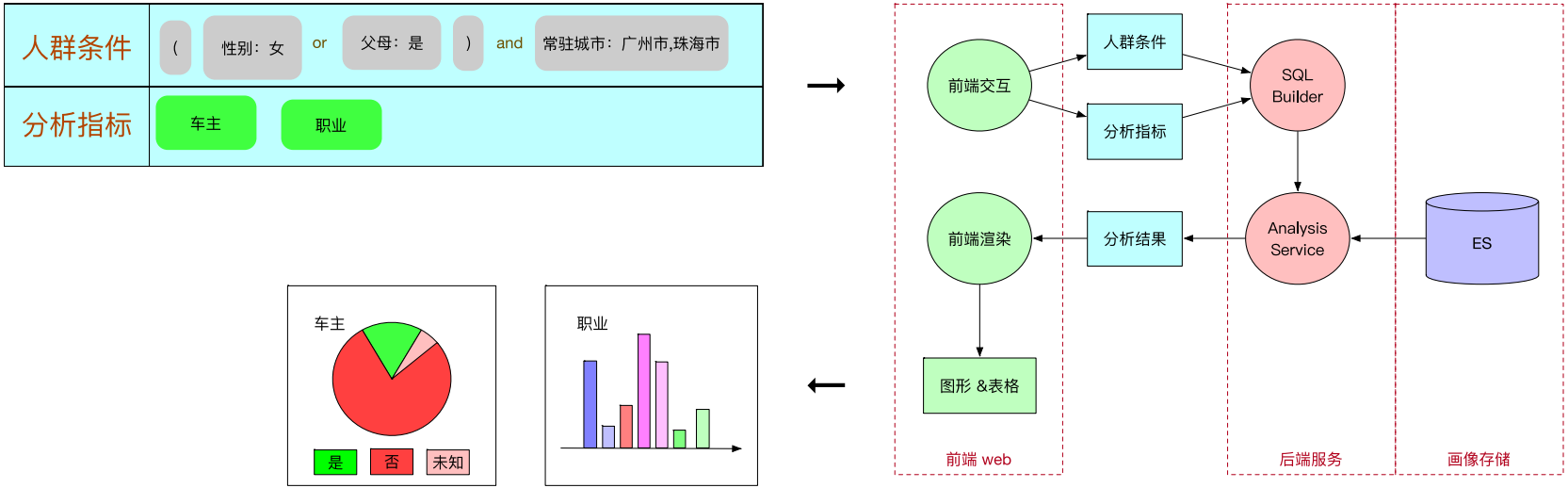
# 目录

# 4.1主要功能列表

| | |
|---|---|
| 人群管理 | 可通过两种方式创建：1、指定标签条件；2、导入imei列表<br>对人群进行修改、删除等操作 |
| 人群筛选 | 指定标签条件选项，查询满足条件的用户数 |
| 画像洞察 | Step 1. 指定标签条件选项选出用户群体<br>Step 2. 指定要分析的标签，通过聚合运算，分析用户特征。 |
| 受众分发 | 采取一定的技术手段，把指定人群推至下游的营销渠道（广告平台、推送平台、OTA等） |
| 画像查询 | 对下游系统提供查询接口，调用方指定用户标识（imei）查询该用户的画像标签 |

# 画像洞察

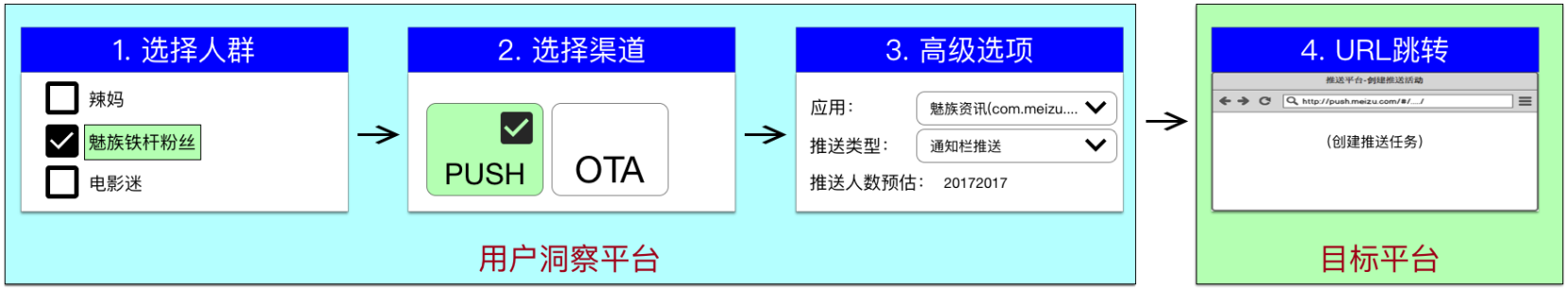
Step 1. 指定标签条件选项选出用户群体
Step 2. 指定要分析的标签，通过聚合运算，分析用户特征。



# 受众分发

交互过程：
1. 选择人群 → 2. 选择分发渠道 → 3. 高级选项 → 4. 平台跳转

无缝对接
3. 高级选项因所选渠道而异，由服务器端动态生成，前端动态渲染。
3. 根据当前用户在目标平台拥有的权限来确定选项列表。
4. 跳转 URL 由目标平台动态生成，降低平台间的耦合，同时利于实现“无缝”对接



# Q&A

Thanks