

China · Beijing

小米品牌广告引擎与算法实践

宋强

MIUI商业产品部



[北京站]

主办方 **Geekbang** & **InfoQ**
极客邦科技



促进软件开发领域知识与创新的传播



关注InfoQ官方微信
及时获取ArchSummit
大会演讲视频信息



全球软件开发大会 [北京站]

2017年4月16-18日 北京·国家会议中心

咨询热线: 010-64738142



全球架构师峰会 2016 [深圳站]

2017年7月7-8日 深圳·华侨城洲际酒店

咨询热线: 010-89880682

个人简介



2014-现在

移动广告平台

系统架构
算法策略
数据平台



Microsoft

2011-2014

搜索广告

大数据分析
实时数据计算
反作弊



2005-2011

数据库

查询优化



大纲

小米品牌广告业务简介

小米品牌广告系统架构

小米品牌广告引擎与算法实践

- 流量预估
- 库存分配
- 在线投放

小米品牌广告业务简介

投放媒体：小米手机和电视全系资源



浏览器



视频



音乐



新闻资讯



天气



日历



电视/盒子

广告样式

- 开屏、锁屏、电视画报
- 信息流、横幅
- 贴片、换肤等

业务规模

- 日曝光近百亿，年收入几十亿

小米品牌广告业务特点

售卖方式

- CPT/CPM，提前下单
- 合约式，如违约需补量赔偿

定向

- 用户属性：地域，年龄，性别等
- 设备型号：手机，电视，盒子等
- 人群包：包含，排除
- 时间：小时级
- 内容：视频分类，剧集，CP等
- 特殊定向：天气状况等

频控

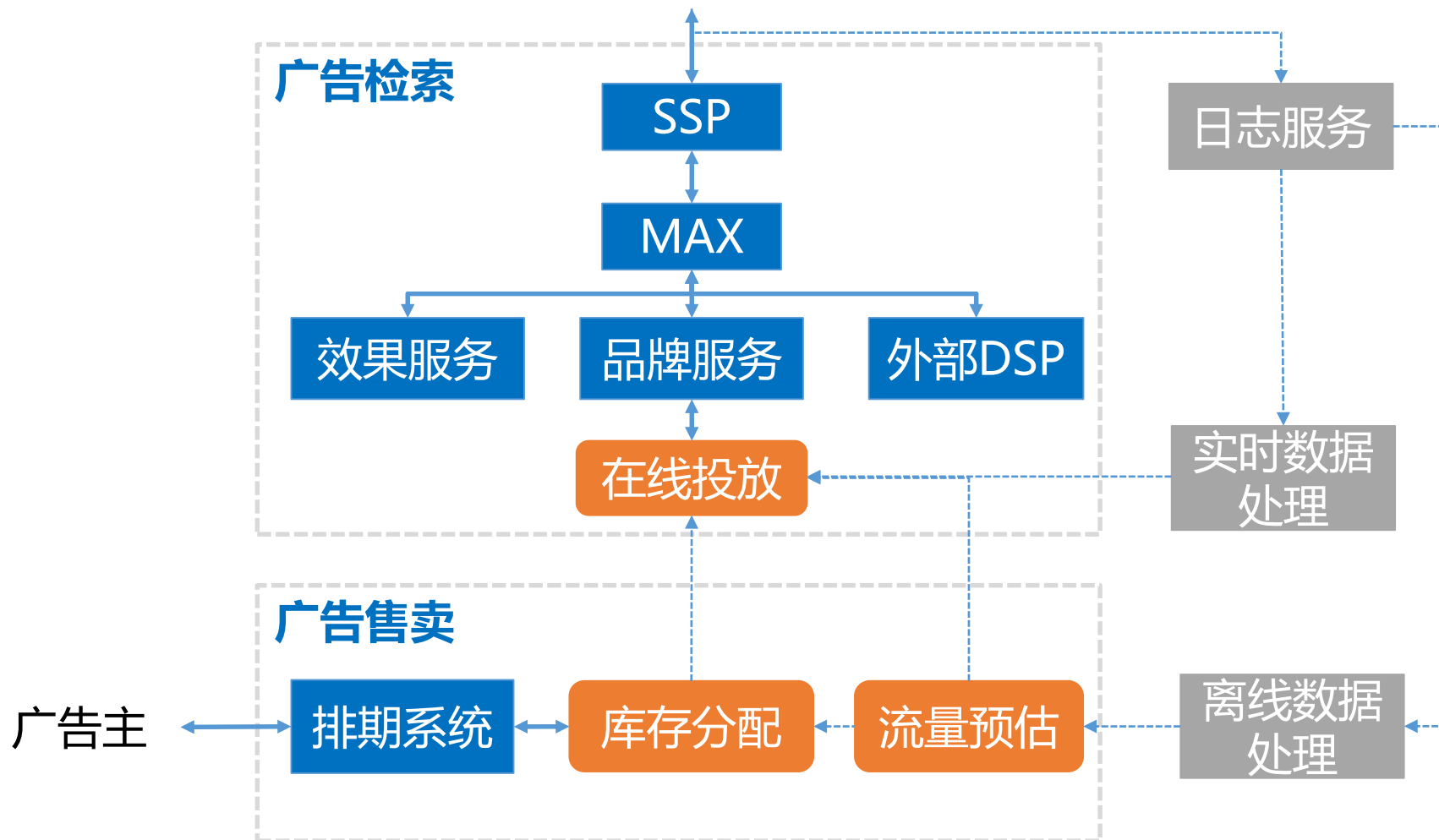
- 小时，日，周，月

第三方监测

- 秒针，AdMaster，DoubleClick

小米品牌广告系统架构

手机/电视



流量预估

问题描述

- 广告主希望购买 “2016/12/25 小米视频首页焦点图 北京 男性” 的流量
 - “日期+广告位+定向条件”

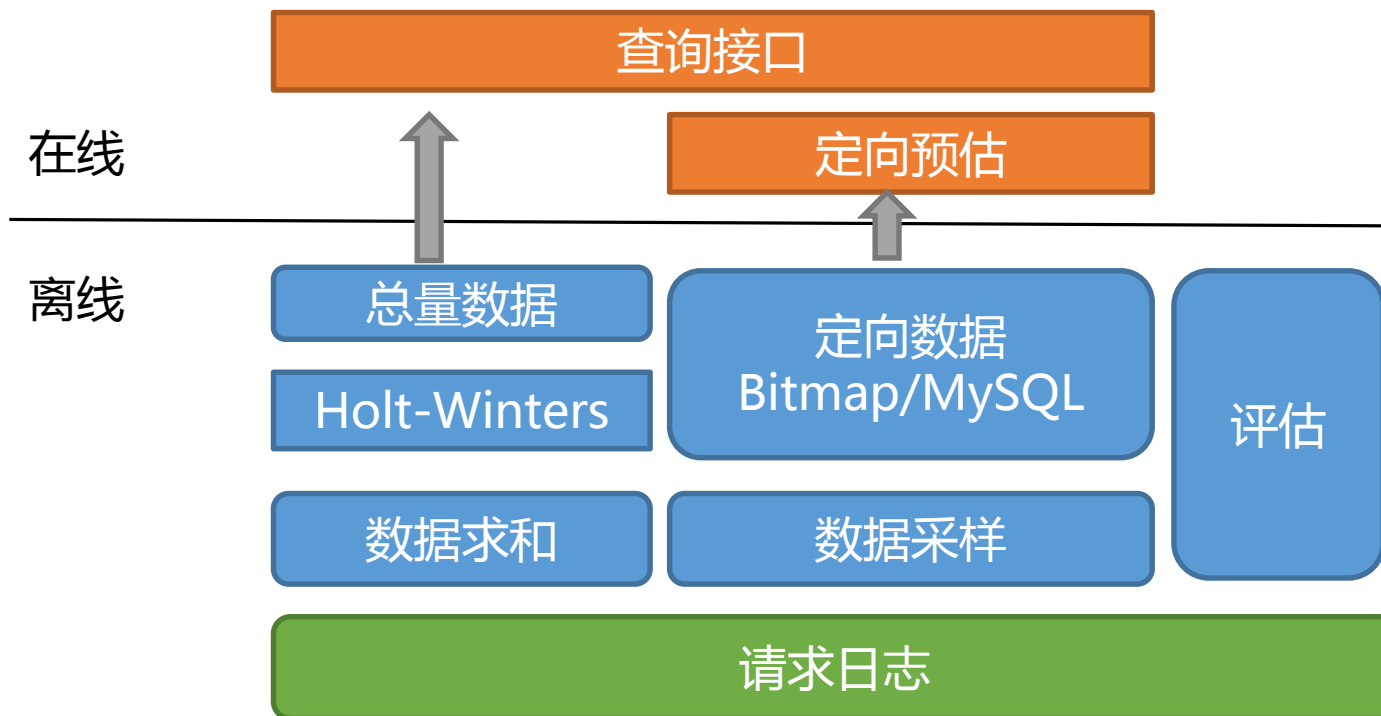
技术挑战

- 定向预估：定向条件多样，组合维度庞大
- 总量预估：流量随时间和季节变化

衡量指标

- 流量预估的准确性
- 流量查询的效率

流量预估 - 系统架构



流量预估 – 算法描述

正交算法

总量预估

- Holt-winters
- 10万CPM

单维度分布

- 地域：北京5%，上海3%，...
- 性别：男75%，女25%，...

正交求解

- 预估量（北京，男）
 $= 10万 * 5% * 75% = 3750CPM$

位图算法

X轴：请求

Y轴：

定向维度

	pv_1	pv_2	pv_3	pv_4	...	pv_n
北京	0	1	0	1		1
上海	1	0	1	0		0
男	1	0	1	0		0
女	0	1	0	1		1
体育	1	1	0	1		1
财经	0	0	1	0		0

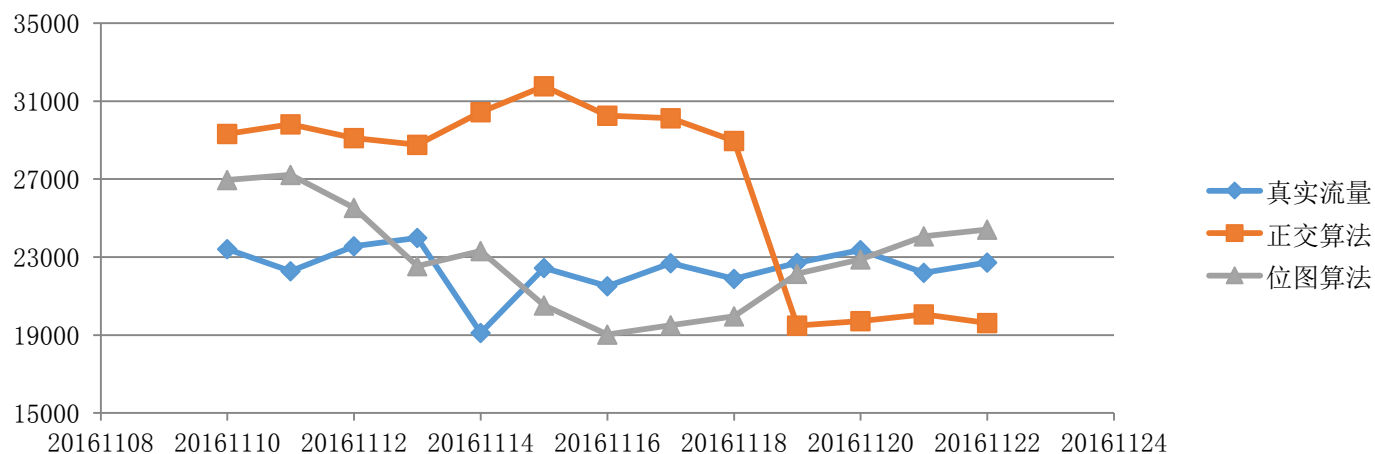
流量预估 – 算法评估

正交算法

- 优点：简单，单维度定向预估准确
- 缺点：组合维度定向预估不够精确

位图算法

- 优点：计算简单（按位与或），组合维度定向预估准确
- 缺点：内存占用较大（需进行数据采样）



库存分配

问题描述

库存

	男	女
北京	1CPM	1CPM
上海		1CPM

订单

订单1：北京，2CPM

订单2：女，2CPM

分配方案1：订单2无法满足

	男	女
北京		订单1
上海		

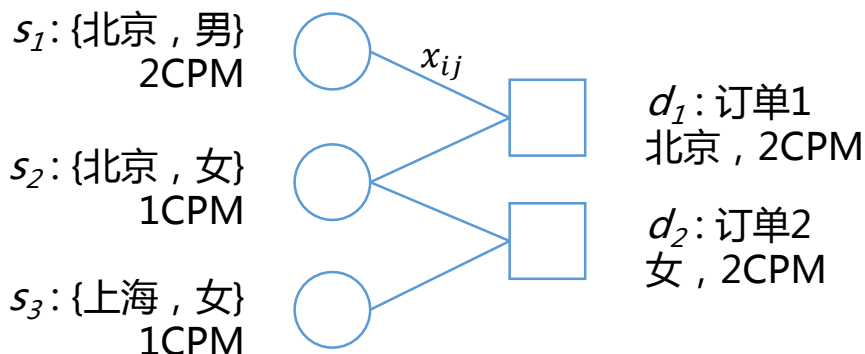
分配方案2：都可以满足

	男	女
北京	订单1	订单1
上海		订单2

技术挑战：定向维度多样，订单数据量大，库存分配的最优化求解复杂

衡量指标：库存的利用率，库存分配算法的求解速度

库存分配 – 问题建模



优化目标：播放的平滑程度和缺量损失

$$\min \frac{1}{2} \sum_{i \in \Gamma(j)} s_i \frac{V_j}{\theta_{ij}} (x_{ij} - \theta_{ij})^2 + \sum_j p_j u_j$$

约束条件

需求约束 $\forall_j \sum_{i \in \Gamma(j)} x_{ij} s_i + u_j \geq d_j$

供给约束 $\forall_i \sum_{j \in \Gamma(i)} x_{ij} \leq 1$

非负约束 $\forall_{(i,j) \in E} x_{ij}, u_j \geq 0$

s_i 表示supply的量

d_j 表示demand的量

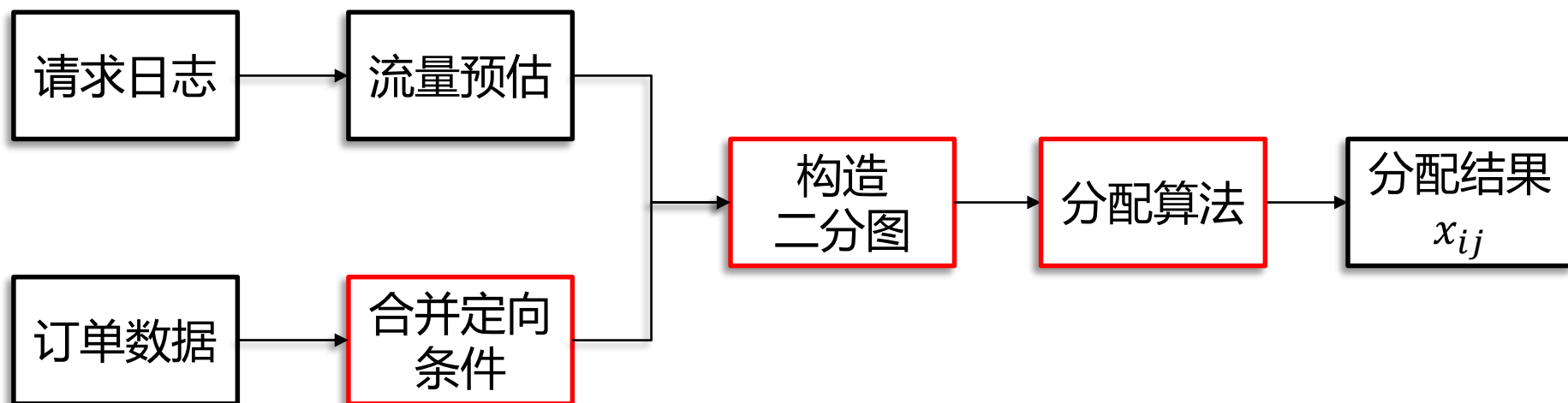
x_{ij} 表示第*i*个流量节点对第*j*个订单的分配比例

u_j 是订单的未完成量

p_j 是订单的缺量损失系数

数学建模是库存分配的核心，问题的求解采用业界成熟的算法（HWM, SHALE）

库存分配 - 流程图



库存分配 - 合并定向条件

- **目标**：生成最小规模的supply节点，简化分配算法
- **算法**：为每个维度生成**最小互斥**散列集合，再进行维度组合

订单数据

订单1：{北京, 上海, 深圳}

订单2：{北京, 广州, 深圳}, {女}

订单3：{女}

最小互斥集合

地域维度：{北京, 深圳}, {上海}, {广州}

性别维度：{男}, {女}

生成候选节点

{北京, 深圳, 男}, {北京, 深圳, 女}

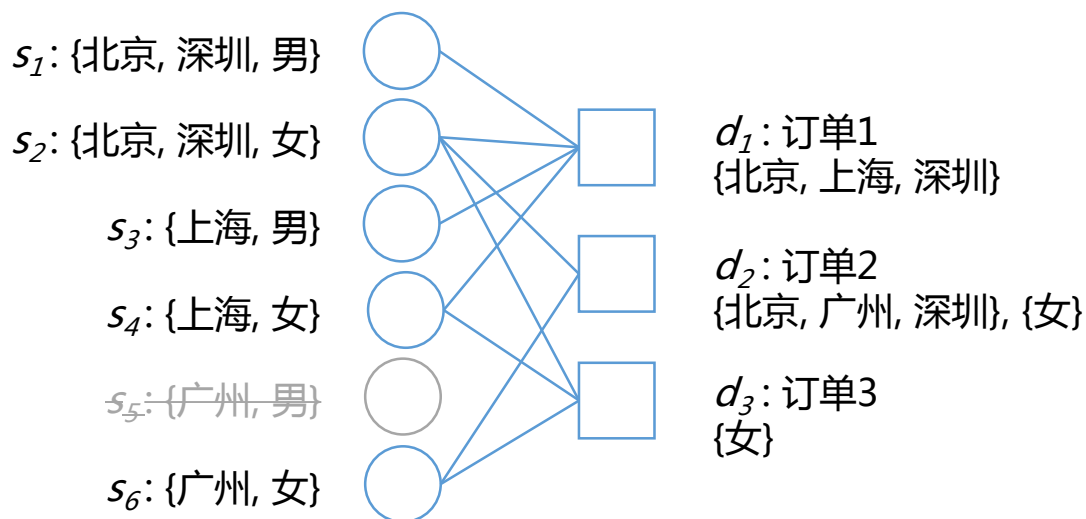
{上海, 男}, {上海, 女}

{广州, 男}, {广州, 女}

库存分配 - 构造二分图

- 遍历每一个候选supply节点，检查是否被某demand节点包含
 - 若不被任何demand包含，则丢弃
 - 若被某demand包含，记录该supply与demand的关系

候选节点



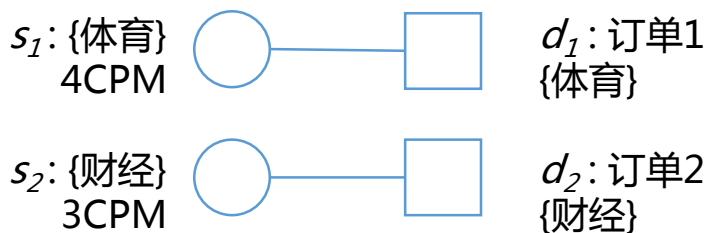
二分图的密度 = 9 (边的总数) / 15 (流量节点数 * 订单节点数)

库存分配 - 维度正交

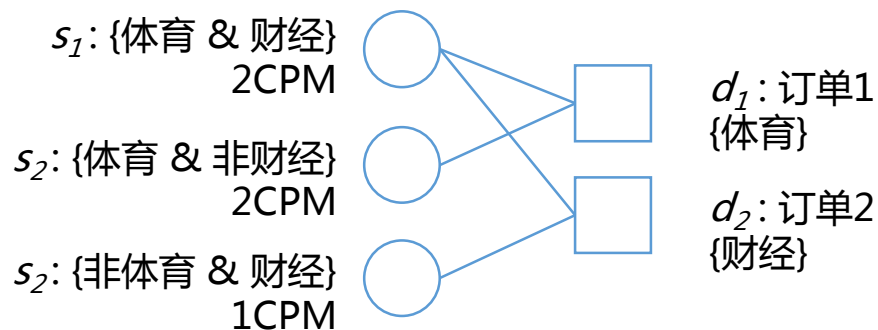
	pv_1	pv_2	pv_3	pv_4	...	pv_n
北京	0	1	0	1		1
上海	1	0	1	0		0
男	1	0	1	0		0
女	0	1	0	1		1
体育	1	1	0	1		1
财经	0	0	1	1		1

兴趣属性的取值不唯一

错误的拆分：维度不正交

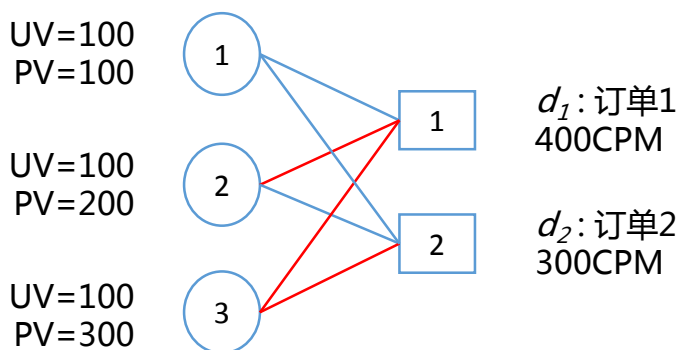


正确的拆分：维度正交



库存分配 - 频控

问题建模



关键算法

- 按照频次对supply节点进行拆分
- 在问题建模中引入频控约束

需求约束 $\forall_j \sum_{i \in \Gamma(j)} x_{ij} s_i + u_j \geq d_j$

供给约束 $\forall_i \sum_{j \in \Gamma(i)} x_{ij} \leq 1$

频控约束 $\forall i, j \quad s_i x_{ij} \leq f_j u v_i$

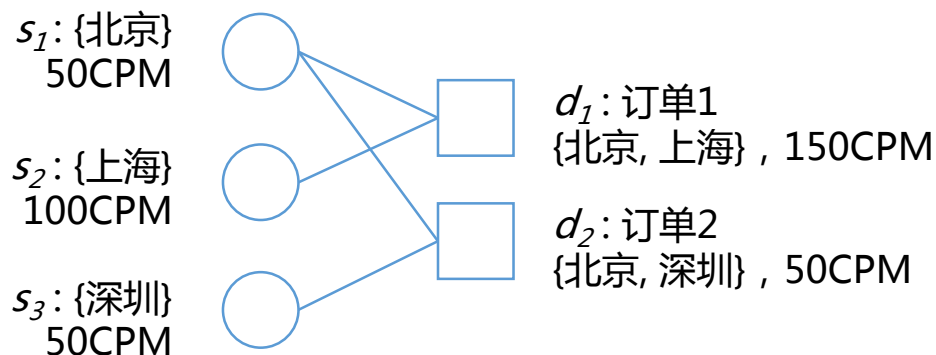
非负约束 $\forall (i, j) \in E \quad x_{ij}, u_j \geq 0$

s_i 表示supply的量, d_j 表示demand的量
 x_{ij} 表示第*i*个流量节点对第*j*个订单的分配比例
 f_j 表示第*j*个广告限制的频次
 $u v_i$ 表示第*i*个流量节点的UV量

频控约束的含义：每个流量节点可以提供的量，不能超过广告主限制的频次乘以这个流量节点的UV

库存分配 – 算法描述

- 离线分配算法的优化目标是库存的利用率，分配算法依赖于订单优先级
- **多队列优先级**：订单可用流量，订单时间等



订单优先级：订单可用流量 ($S_j = \sum_{i \in \Gamma(j)} s_i$)

- 订单2优先级高于订单1 ($S_2=100$, $S_1=150$)
- 订单2可以满足，订单1无法满足

订单优先级：订单时间

- 订单1优先级高于订单2
- 订单1和2都可以满足

在线投放

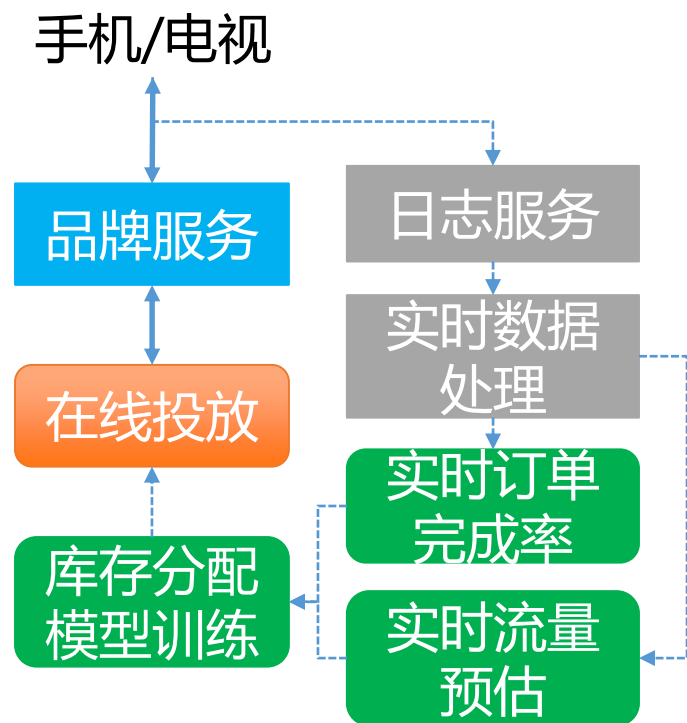
目标：提高订单的完成率和投放的平滑程度

挑战：实际流量和订单完成率偏离预期时，如何快速修正

实时反馈是在线投放的核心

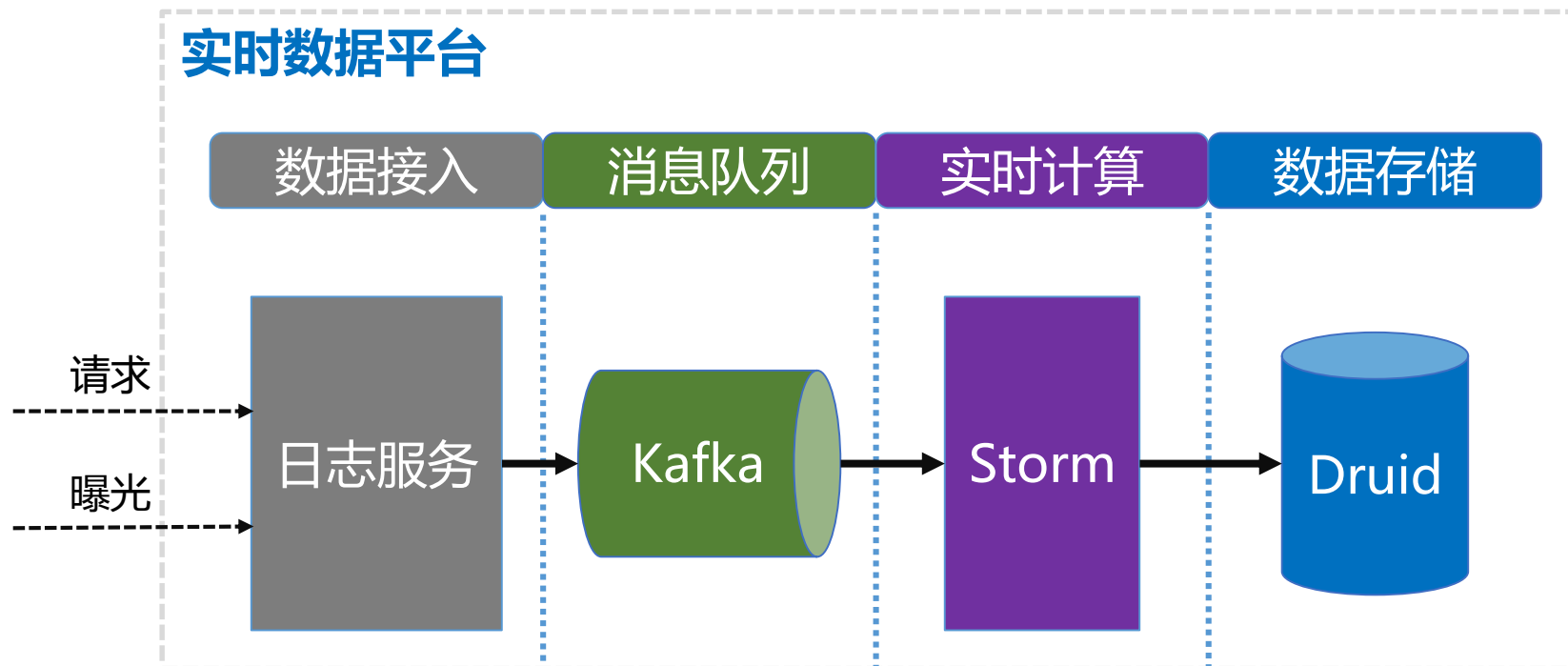
- 实时流量预估修正
- 实时订单完成率反馈

小时级模型训练更新



在线投放 – 实时反馈

实时数据平台



Druid

- 实时多维数据分析 (OLAP) 工具
- 交互式查询，低延迟高可用

在线投放 – A/B实验

品牌广告A/B实验的特点

- 在线分配按照 “订单_X_实验” 进行保量控制
- 离线分配算法训练全流量进行，不需要分流量
- **局限**：实验需要按天进行，无法动态调整流量

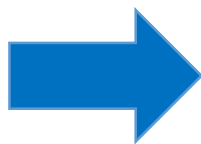
排期广告-每日实时曝光



在线投放 – 分析平台

订单投放量未完成，如何快速定位问题

- 流量预估问题
- 分配算法问题
- 其他问题



排期广告分析平台

- 支持实时和历史问题排查
- 模拟广告请求，获取广告投放关键步骤信息
- 分阶段详细的counting数据

Ad Analysis									
历史问题诊断									
AdId: 99792		TagId: 1.3.c.1		日期: 2016-11-22					
adId	tagId	hour	request	in_index	in_candidates	in_resultAdList	filterBy-Selection	filterBy-Smoothing	filterBy-Freq
99792	1.3.c.1	0	690377	690377	159372	2813	4083	152476	0
99792	1.3.c.1	1	347000	347000	81549	0	0	81549	0
99792	1.3.c.1	2	240516	240516	58452	0	0	58452	0
99792	1.3.c.1	3	213578	213578	54457	405	382	53670	0
99792	1.3.c.1	4	270789	270789	71825	2498	3391	65936	0
99792	1.3.c.1	5	642714	642714	179935	4039	5627	170269	0
99792	1.3.c.1	6	1527407	1527407	419498	3308	4514	411676	0
99792	1.3.c.1	7	2269092	2269092	601417	1747	2608	597062	0
99792	1.3.c.1	8	2218341	2218341	578908	2611	2996	573301	0

THANKS



[北京站]

主办方 **Geekbang** > **InfoQ**
极客邦科技