

—— World Of Tech 2017 ——

全球架构与运维技术峰会

2017年4月14日-15日 北京富力万丽酒店

ARCHITECTURE



出品人及主持人：

邓钦华 蘑菇街 技术总监

电商大促背后的技术挑战

蘑菇街搜索推荐架构的探索之路

——小宝





小宝

蘑菇街
搜索技术团队负责人

分享主题：
蘑菇街搜索推荐架构的探索之路

个人简介

- ▶ 丁小明（花名：小宝）
- ▶ 2003~2010：浙大
- ▶ 2010~2011：网易
- ▶ 2011~now：蘑菇街
 - ~2013：导购业务
 - 2013~now：搜索技术团队

解答两个疑问

- ▶ 1.跟分会场主题有点跑调?
- ▶ 2.搜索推荐一起讲什么鬼?

提纲

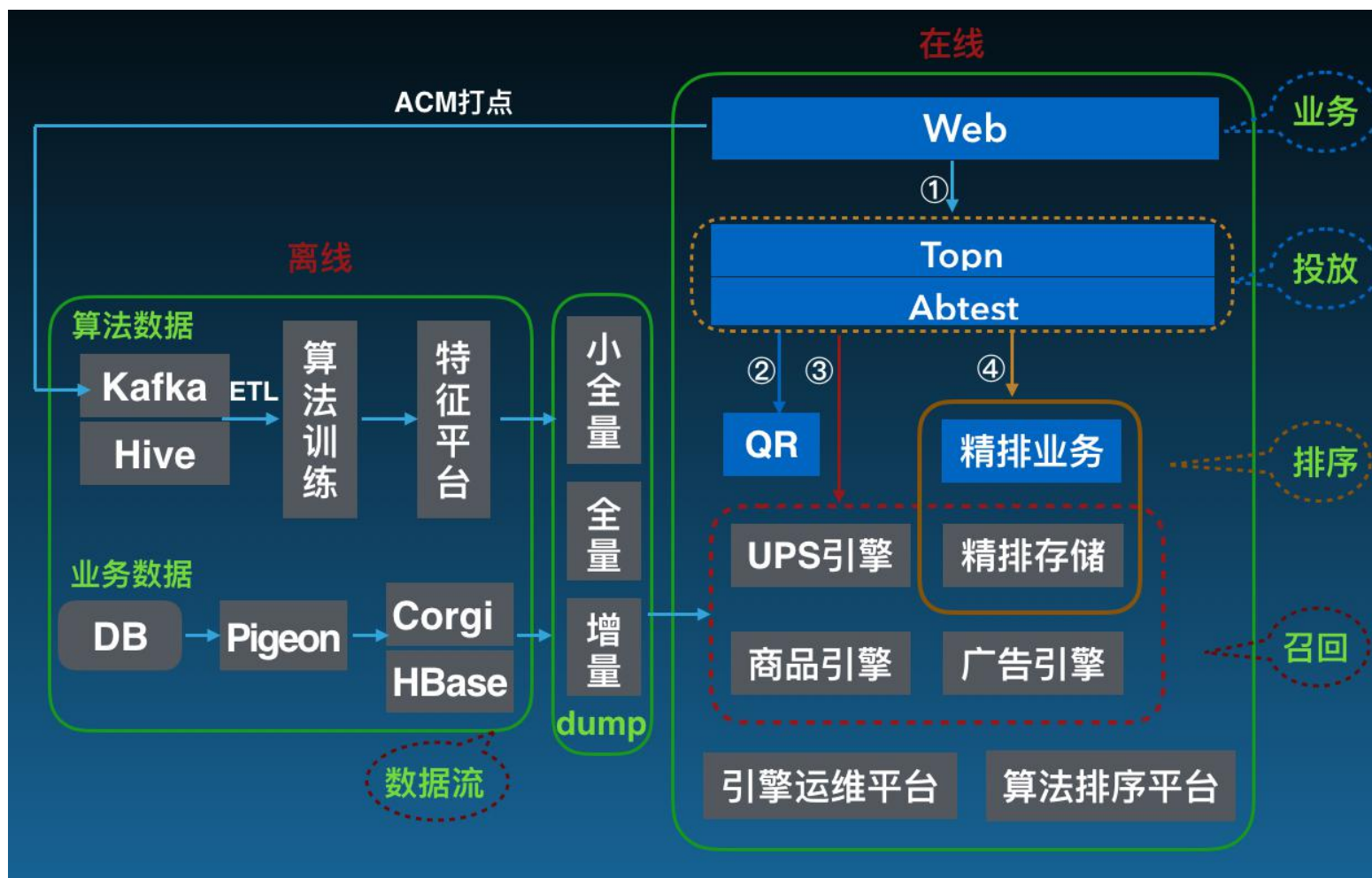
- ▶ 搜索架构的探索
- ▶ 推荐架构的探索
- ▶ Q&A

一、搜索架构的探索

搜索架构的探索阶段

- ▶ 导购时期 (~13.11)
- ▶ 电商初期 (13.11~14.11)
- ▶ solr主搜阶段 (15.4 ~16.3)
- ▶ C++主搜阶段 (15.8 ~16.11)
 - ▶ 数据流的挑战
- ▶ 平台化阶段 (17.1 ~now)

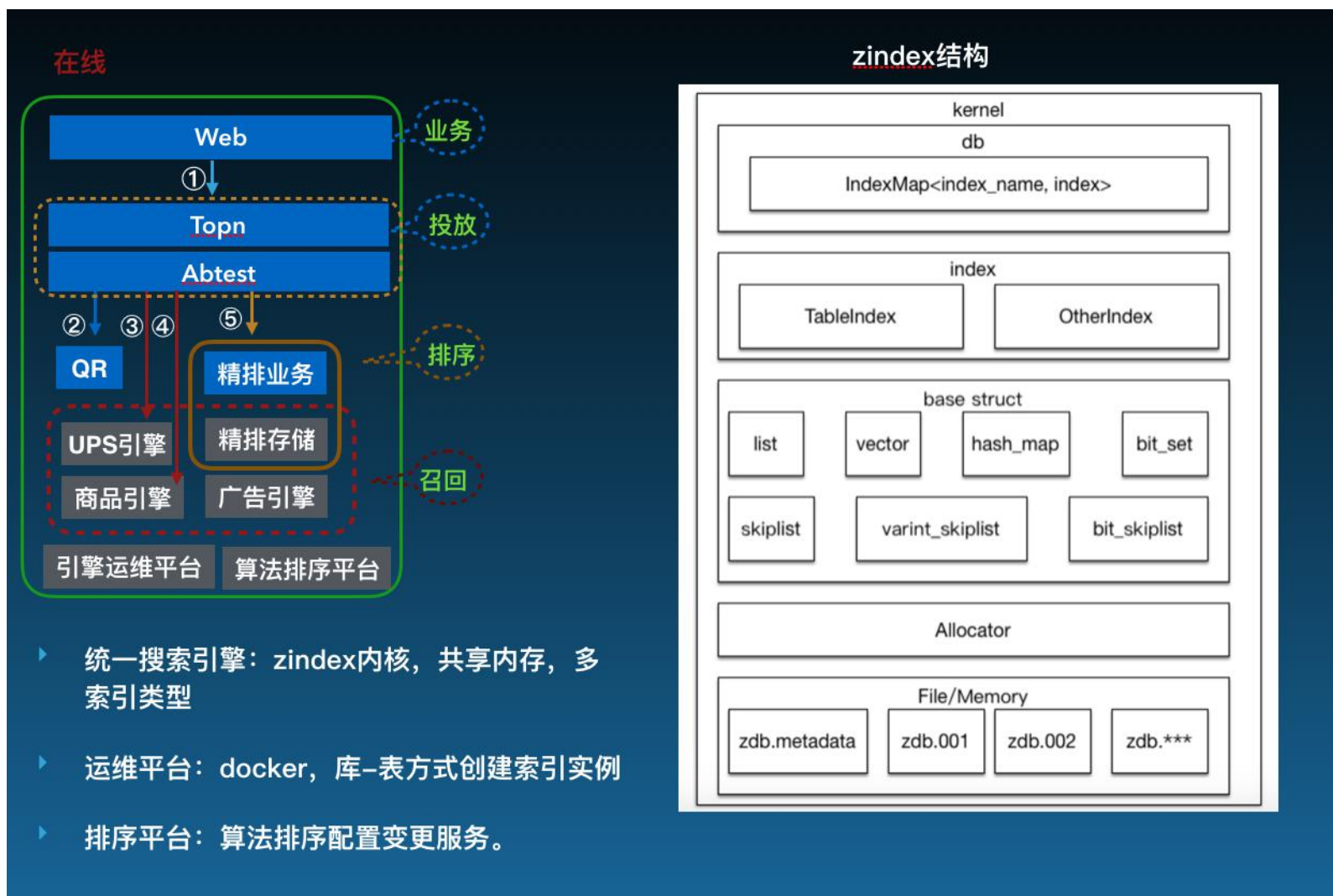
现在的架构



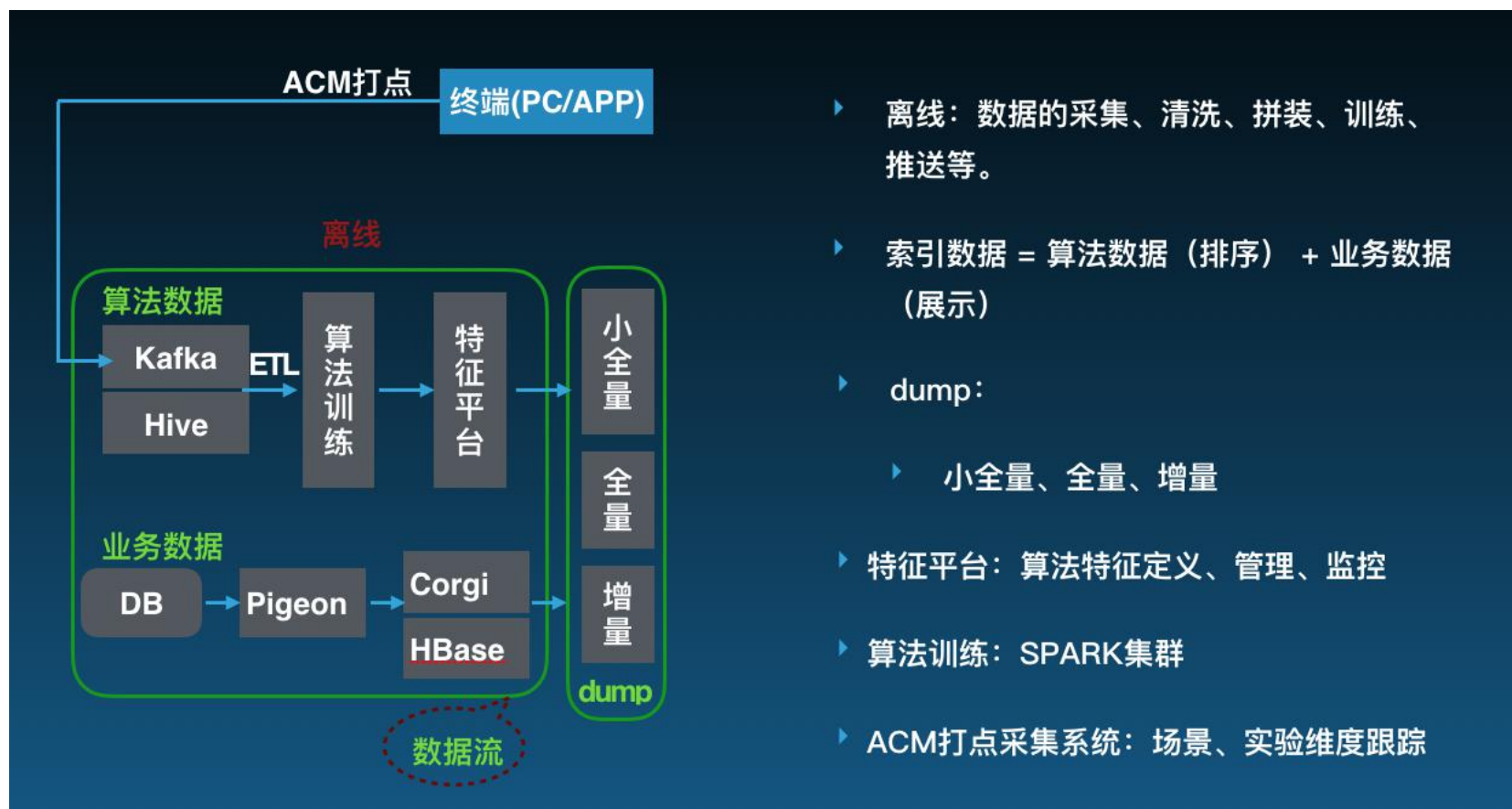
架构细节—在线部分1



架构细节—在线部分2—引擎

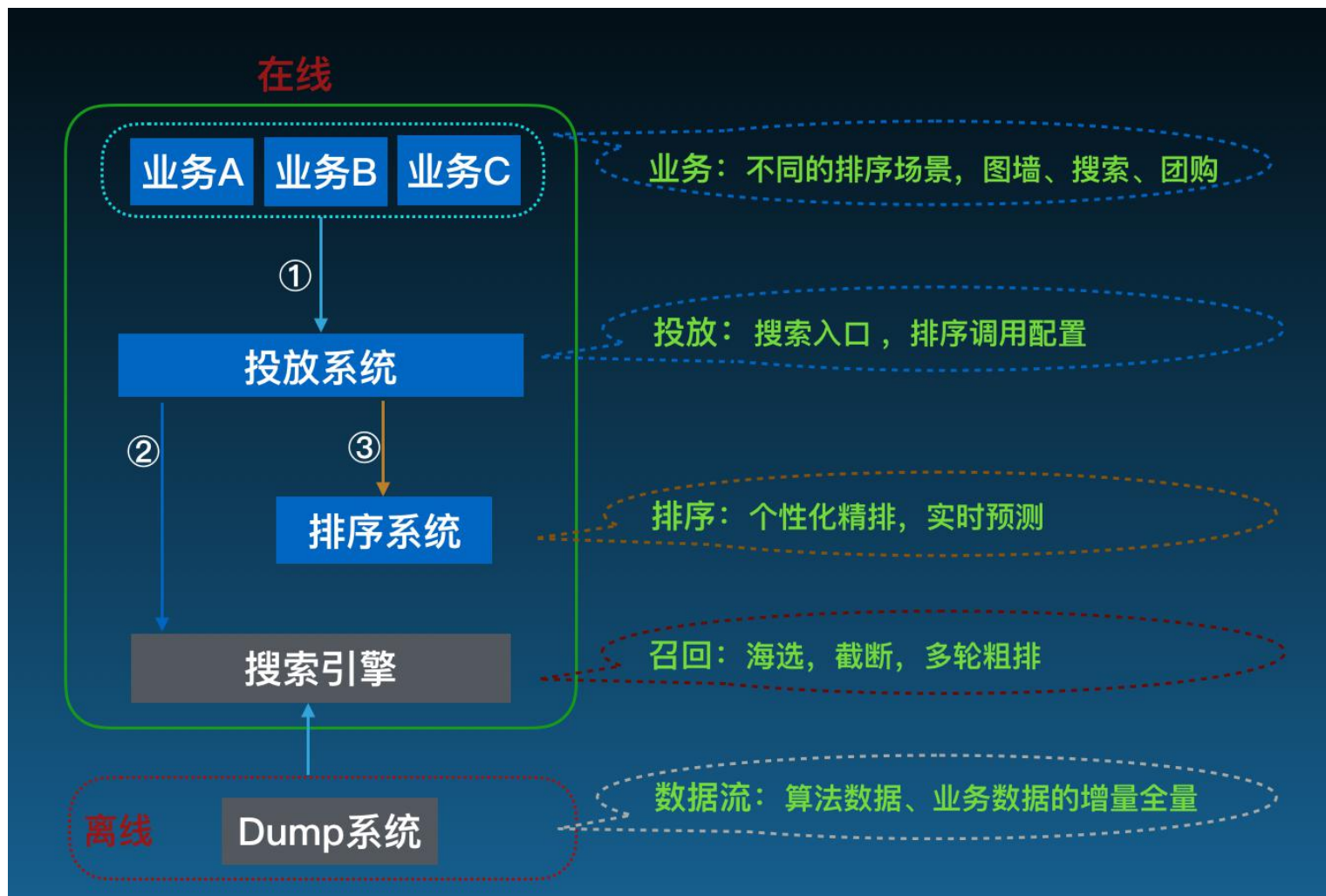


架构细节—离线部分

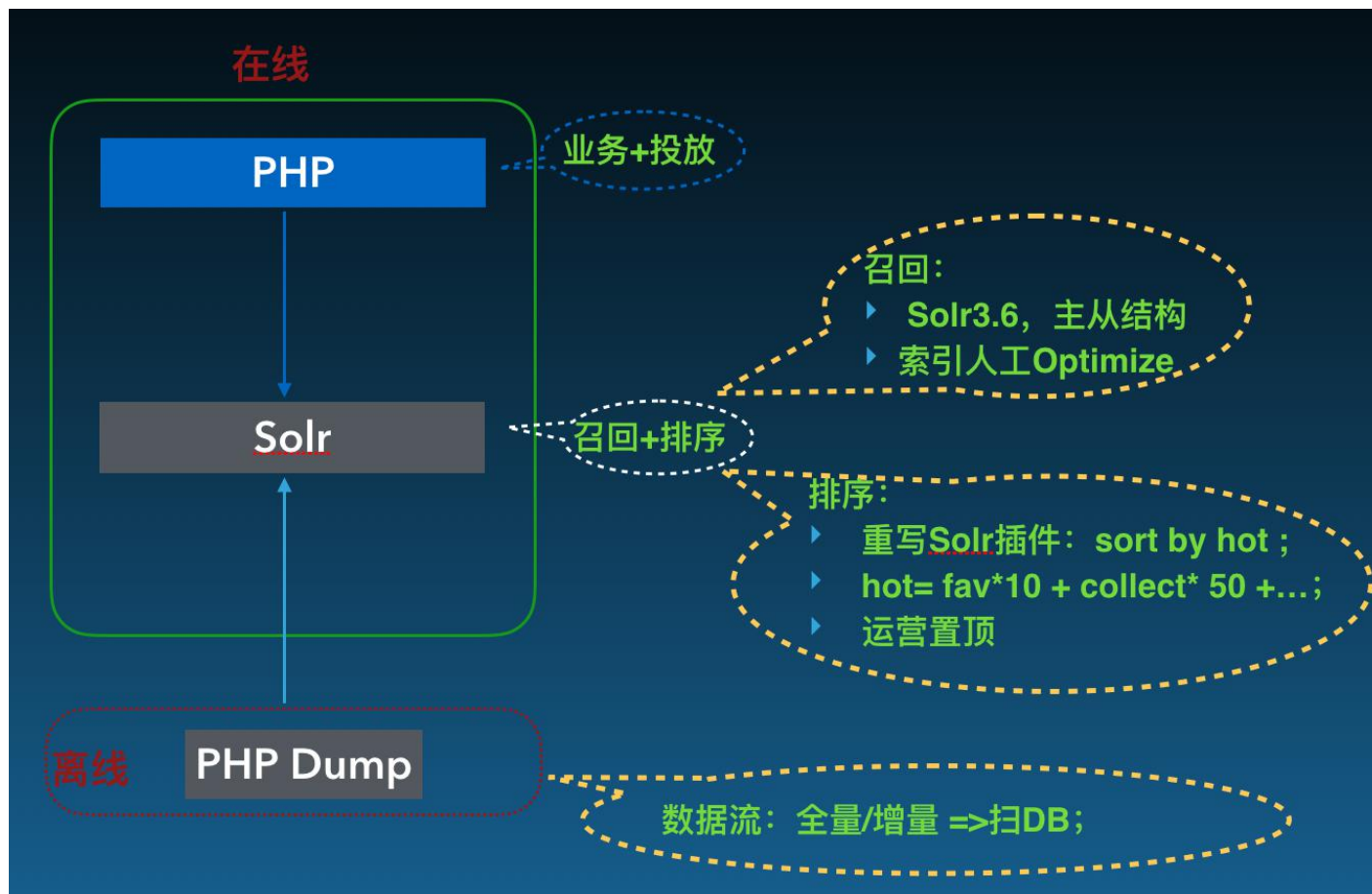


- ▶ 离线：数据的采集、清洗、拼装、训练、推送等。
- ▶ 索引数据 = 算法数据（排序） + 业务数据（展示）
- ▶ dump:
 - ▶ 小全量、全量、增量
- ▶ 特征平台：算法特征定义、管理、监控
- ▶ 算法训练：SPARK集群
- ▶ ACM打点采集系统：场景、实验维度跟踪

架构简化一下



导购时期(2013.11之前)



电商转型后的挑战(2013.11)

▶ 排序需求:

- 流量值钱
- 更多特征和数据
- 算法数据膨胀
- 排序调整频繁

▶ 增量/全量

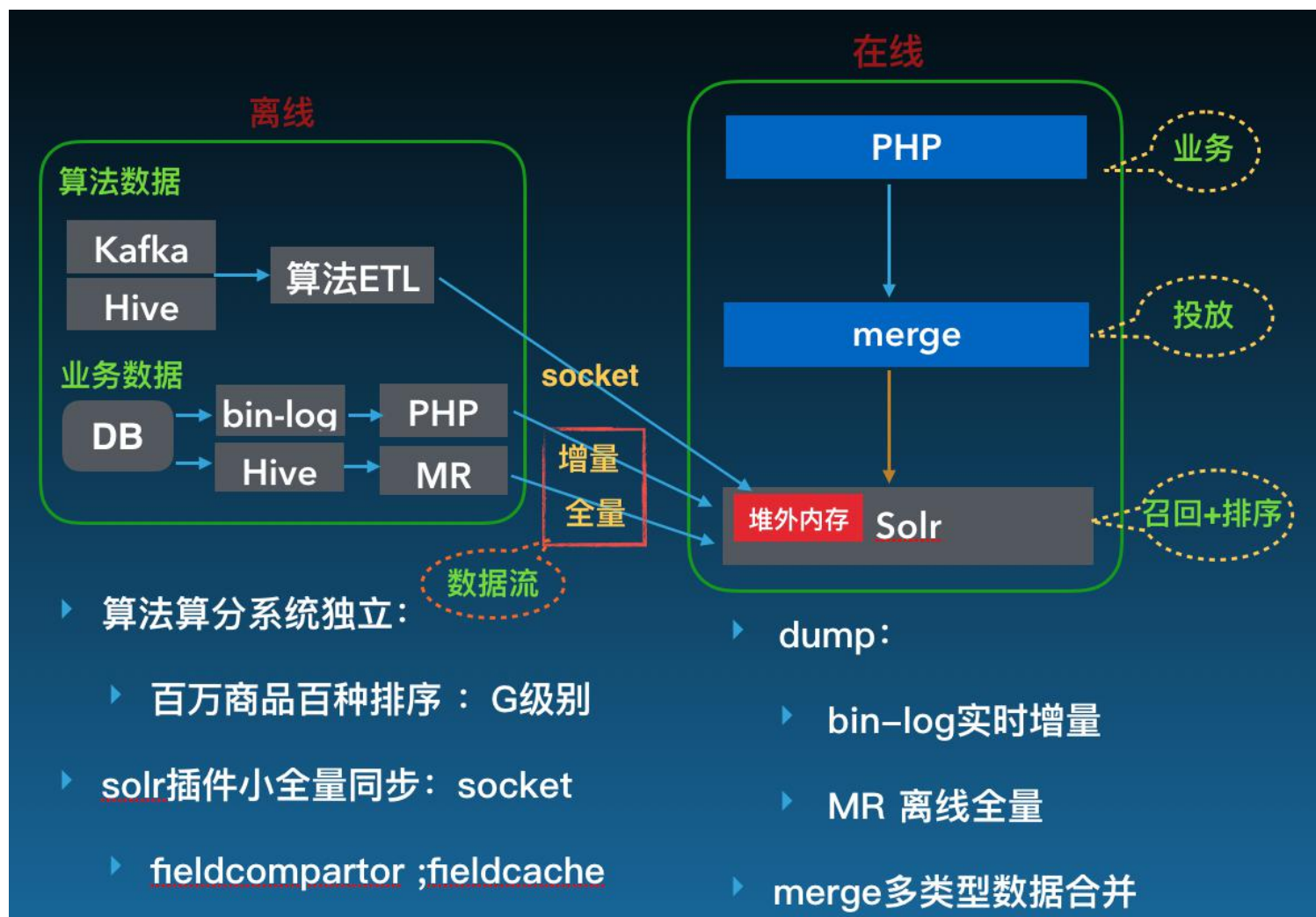
- ▶ 增量管理复杂
- 数据量大
- optimize风险大

▶ 多类型商品透出:

- ▶ 导购、广告、搭配

挑战: 排序特征变多 + 排序数据变大 + 排序变更变频繁

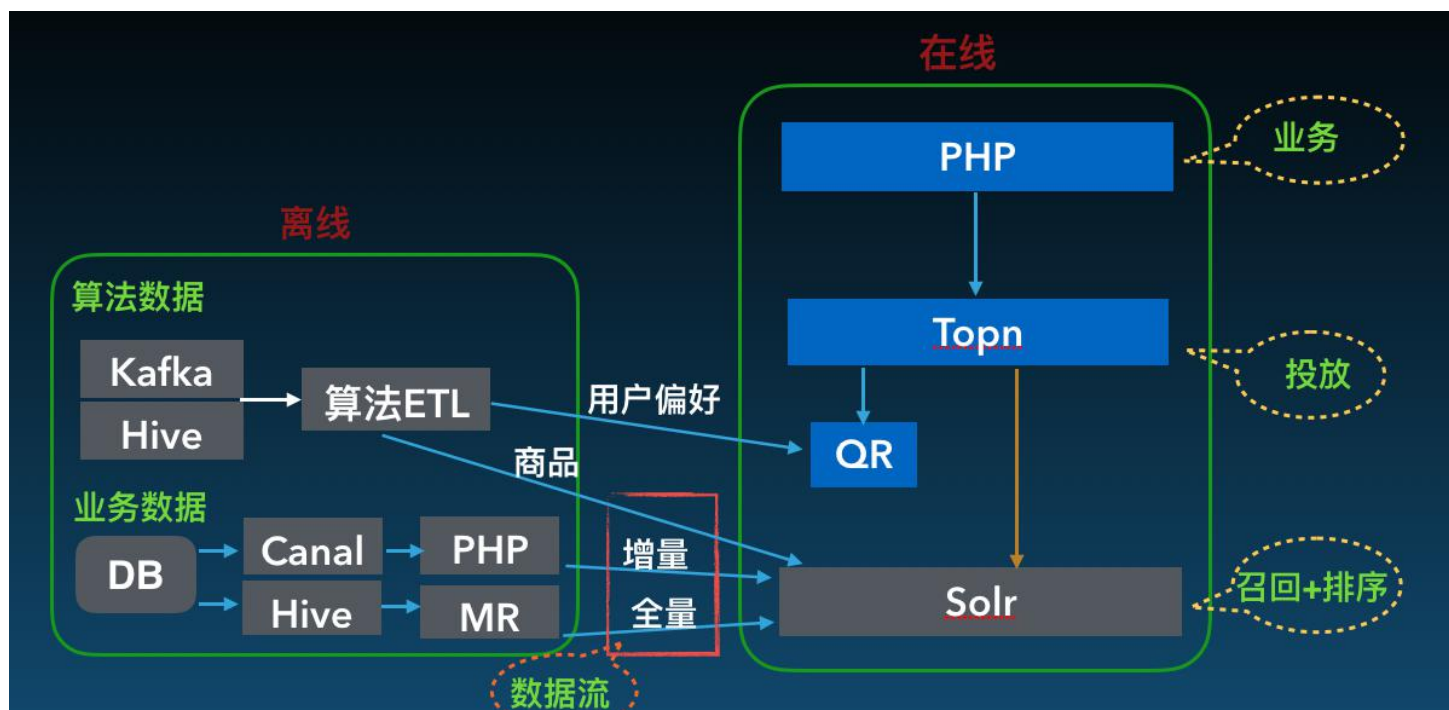
转型初期架构(2013.11)



架构挑战（2014.11）

- ▶ 排序：
 - ▶ 规则排序→LTR，算法排序需求多；
 - ▶ 排序灵活性制约：计算好的分数离线推送到solr；
 - ▶ solr内存压力：GC/段合并；
 - ▶ 静态分，相关性差；
- ▶ 大促问题：
 - ▶ 相关性：搜“雨伞”，雨伞图案的连衣裙排前面

solr主搜架构(2015.4)



- ▶ solr改造, 支持rank插件 (ranker->scorer), 配置化+动态加载
- ▶ 新增QR系统 => 相关性, 部分个性化
- ▶ 内存压力: solr升级(docvalues), 算法分走动态字段增量
- ▶ 投放层: merge->topn, 对接不同的场景

挑战

▶ 排序：

- ▶ 爆款模型 -> 个性化模型
- ▶ 重排、打散等复杂需求
- ▶ 单轮排序：靠公式融合解决

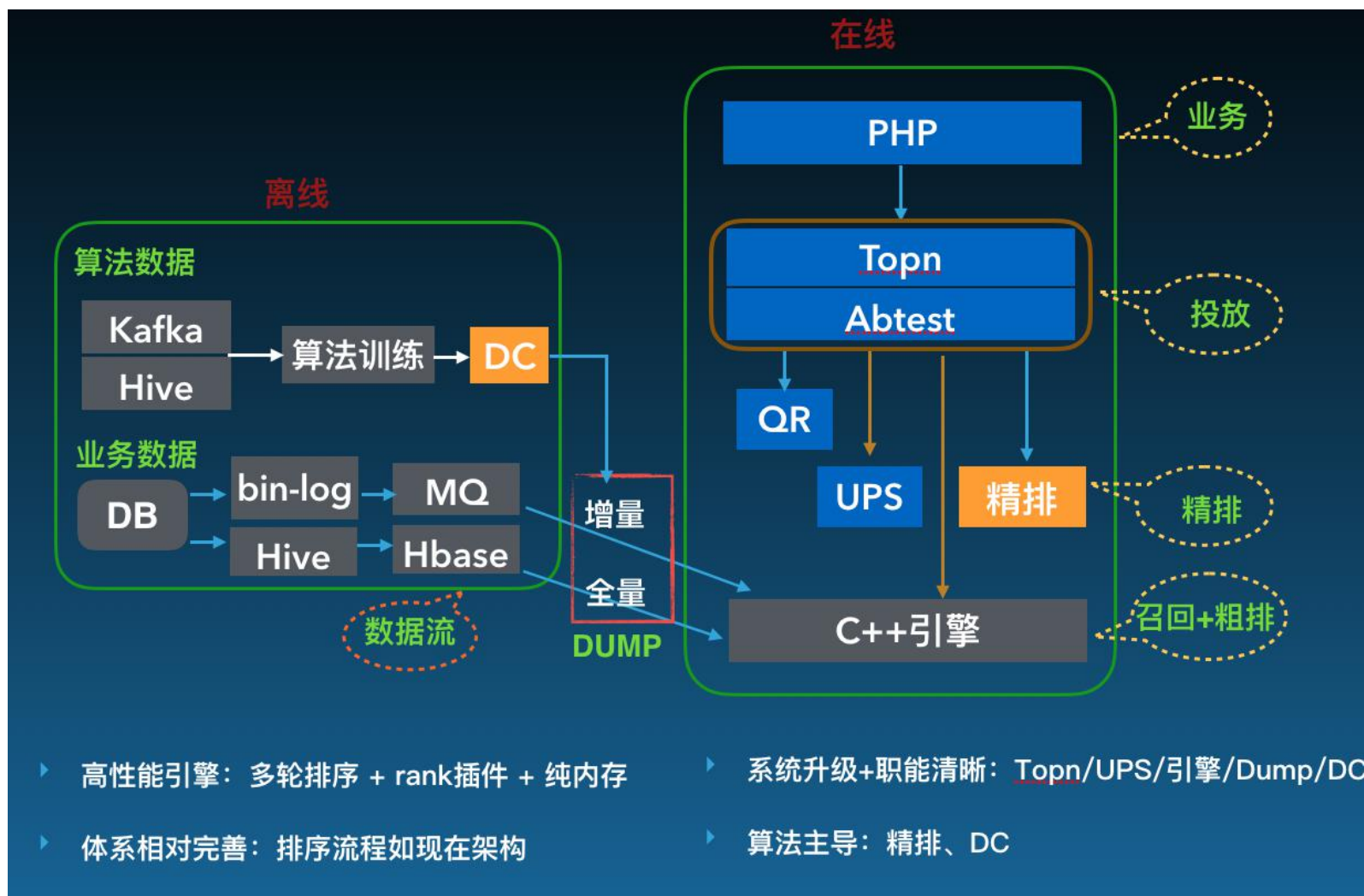
▶ bad case：

- ▶ 相关性之后，做品牌加权，再做类目打散

▶ solr限制

- ▶ solr算分性能压力
- ▶ 段合并的抖动
- ▶ 二次开发成本高
- ▶ 绕不过的GC

C++主搜架构(2015.7—2016.3)



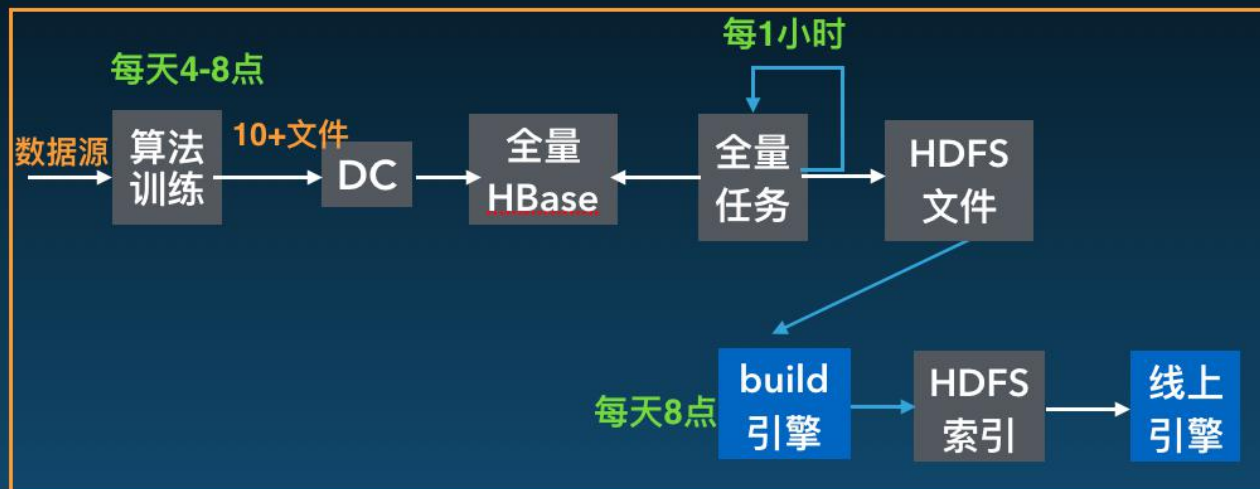
数据流出问题了

- ▶ 全量无调度，流程靠约定
- ▶ 增量带来算法分数不可比
- ▶ 业务数据增量对服务接口压力过大（促销故障）

数据流的问题-1（2016.3~9）

- ▶ 问题：全量索引靠约定 + 数据容灾机制弱；

- ▶ 场景：



- ▶ 大促故障：

- ▶ 前置任务延迟导致全量做不了，线上内存几乎撑爆；经常性全量延迟+手动处理；
- ▶ 算法误导排序分，导致线上错乱；增量恢复时间长；

数据流解决-1

- ▶ 解决：调度系统 => 流程自动化+智能化
 - ▶ 基于zk的调度系统，驱动全流程；
 - ▶ 失败重试； 错误报警； 并行化；
- ▶ 解决：dump容灾完善
 - ▶ 排序SOS字段；
 - ▶ 全量定期快照（基于Hbase）；
 - ▶ 单算法字段修复； => 配合后面的小全量

数据流的问题-2

- ▶ 问题：两次算法分数增量不可比；增量生效慢；
- ▶ 原因：
 - ▶ LR二分类模型，算法分基于预估的概率值；
 - ▶ 两次训练和预估的数据分布不同（数据不同：不同时段成交）
- ▶ 大促问题：
 - ▶ 0点时分，排序错乱；
 - ▶ 排序增量修复慢，排序错乱；



数据流的解决-2

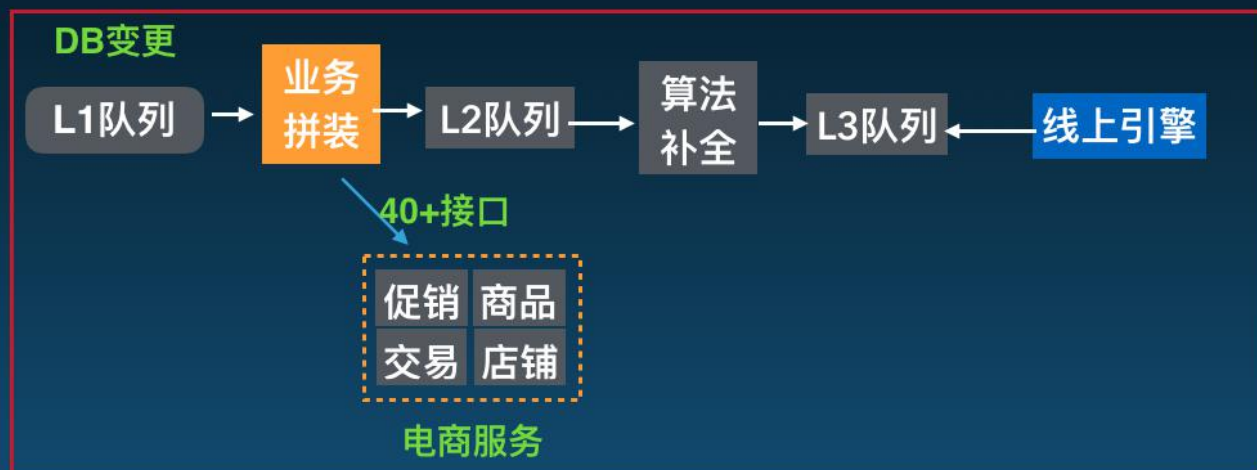
- ▶ 解决：引擎支持小全量更新（单次全量生效）
- ▶ 算法分数单独更新
- ▶ 引擎开放口（存本地文件，内存切换）



- ▶ 收益：算法分数独立；快速切换；快速恢复；

数据流的问题-3

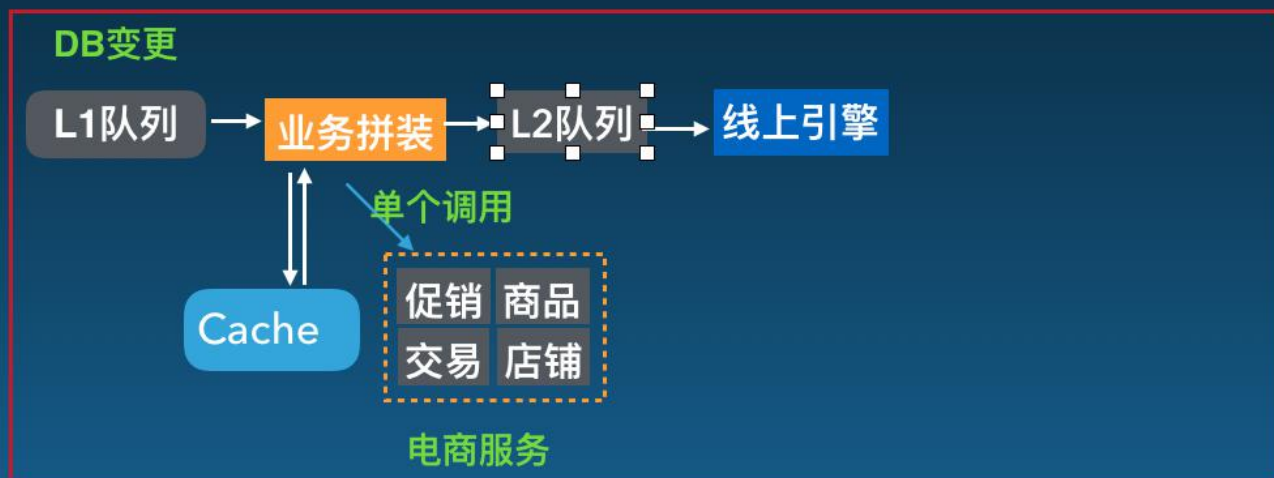
- ▶ 问题：业务增量压力
- ▶ 原因：变更doc级更新 + 大促变更



- ▶ 大促问题：=> 2000+qps
 - ▶ 促销接口故障；数据修复时间长；
 - ▶ 业务接口被迫降级；

数据流的解决-3

- ▶ 解决：字段级更新+ 中间存储
- ▶ 只拼装变更字段(id+price)
- ▶ 引擎字段级更新支持(倒排+正排)
- ▶ 中间存储解决字段间依赖调用（促销价格依赖原价）

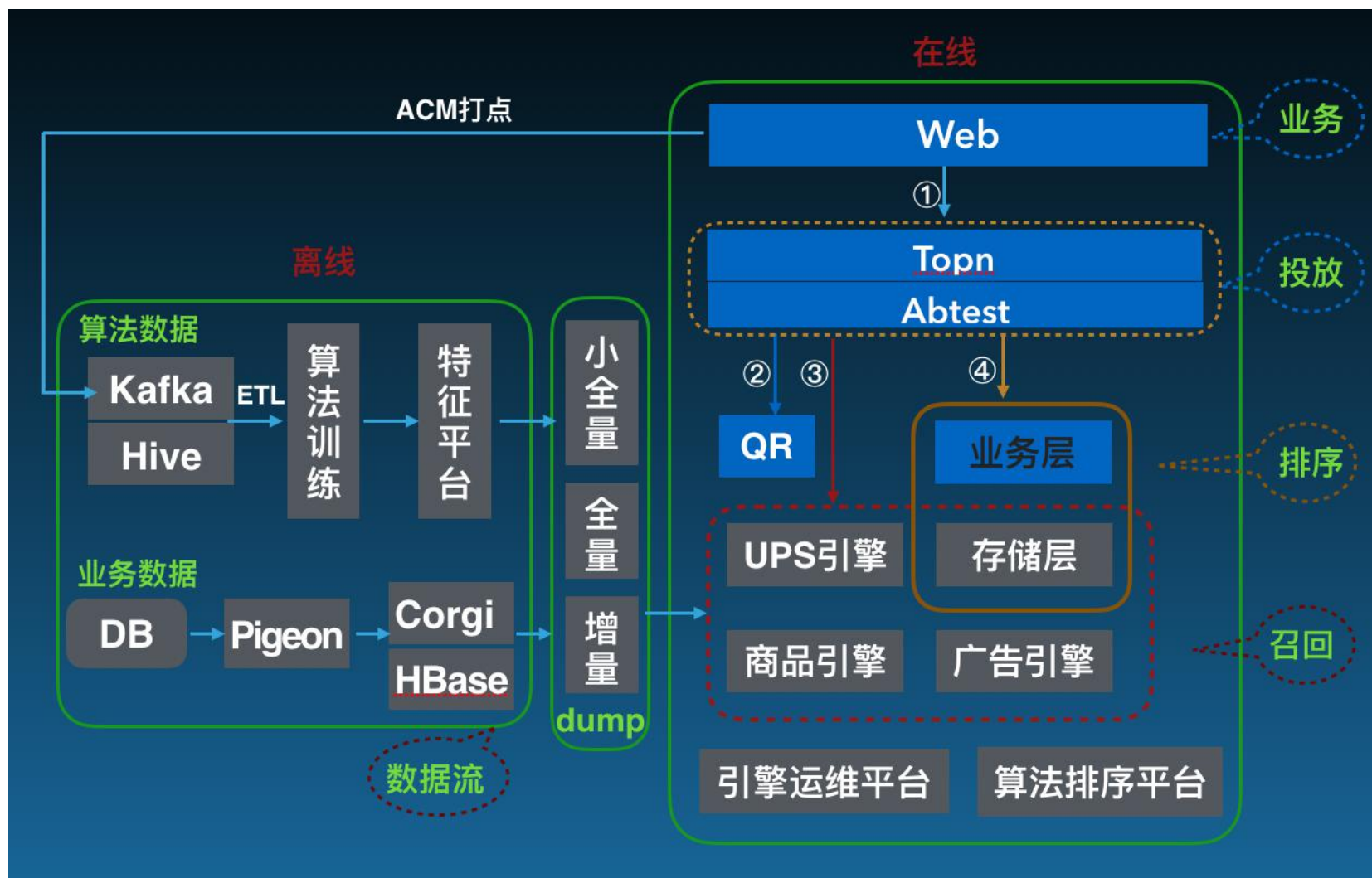


- ▶ 收益：RT**150ms-70ms**；QPS关键接口减少**80%**以上；引擎端减少压力

新挑战（2017）

- ▶ 团队融合：
 - ▶ 多套引擎：ups/广告/商品
 - ▶ 多套投放：广告/搜索
- ▶ 职责明晰：
 - ▶ 工程接管精排、特征平台
 - ▶ 排序平台
- ▶ 复杂排序计算：
 - ▶ 精排排序性能
 - ▶ 非线性模型
- ▶ 系统效率优化：
 - ▶ 运维平台
 - ▶ 数据流统一

新架构(2017)



探索总结

- ▶ 快速支持业务
 - ▶ 满足算法需求（算法跑很快，工程跟不上）

提升效率，降低成本

- ▶ 不同阶段的选择
- ▶ 体系化的建设

后续规划

0. 平台化体系深入

1. Online Learning&DNN & 强化学习

2. 深度学习框架(TensorFlow)

3. 图搜工程体系

二、推荐架构的探索

推荐场景@蘑菇街

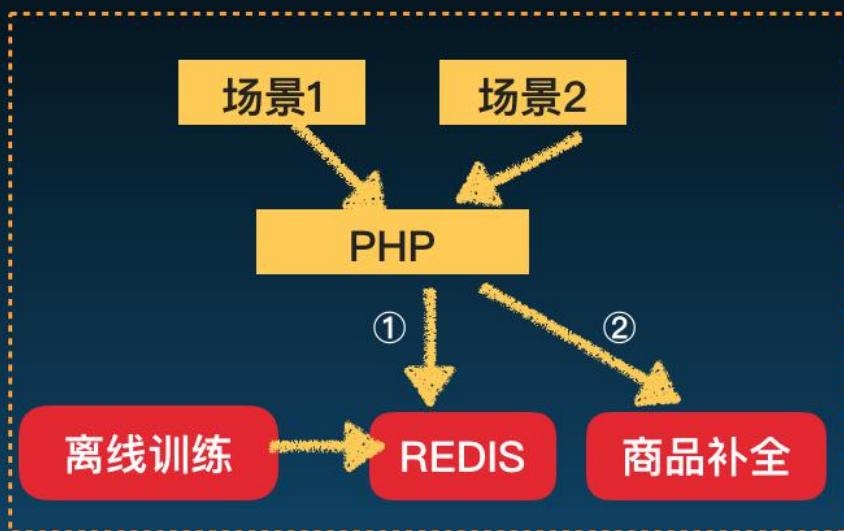
- ▶ 用户行为路径：
- ▶ 推荐实体：商品、店铺、标签、运营内容



推荐架构—发展阶段

- ▶ 发展早期(~15.6)
- ▶ 1.0 时期：从0到1(15.6~16.3)
- ▶ 2.0：投放+个性化(16.3~16.12)
- ▶ 3.0：平台化(16.12~now)

早期（2013.11~2015.6）

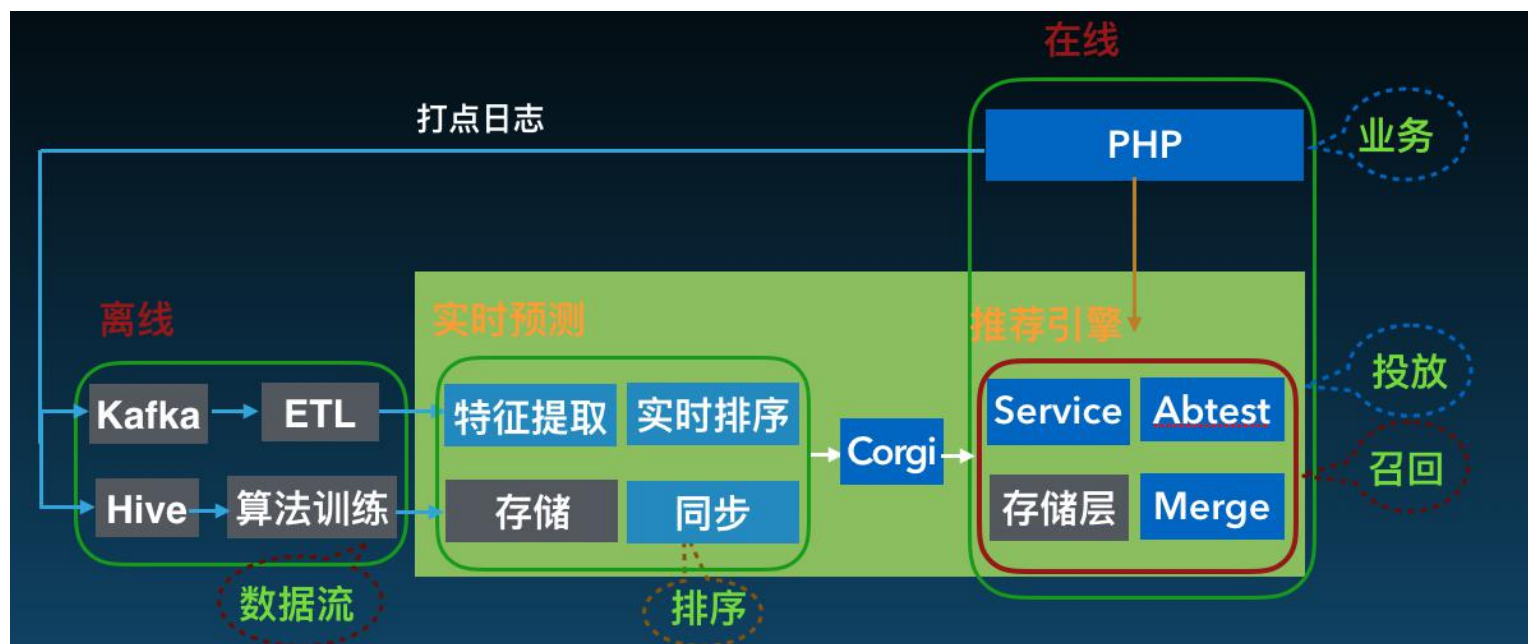


- ▶ 推荐场景少，需求简单
- ▶ 数据天级别离线更新到redis

问题

- ▶ 没有专门的推荐系统
- ▶ 效果跟踪差
- ▶ 场景对接效率低；数据导入效率低
- ▶ 无个性化；

架构1.0（2015.11）



从0到1:

- ▶ service对接场景
- ▶ 自写k-v推荐存储

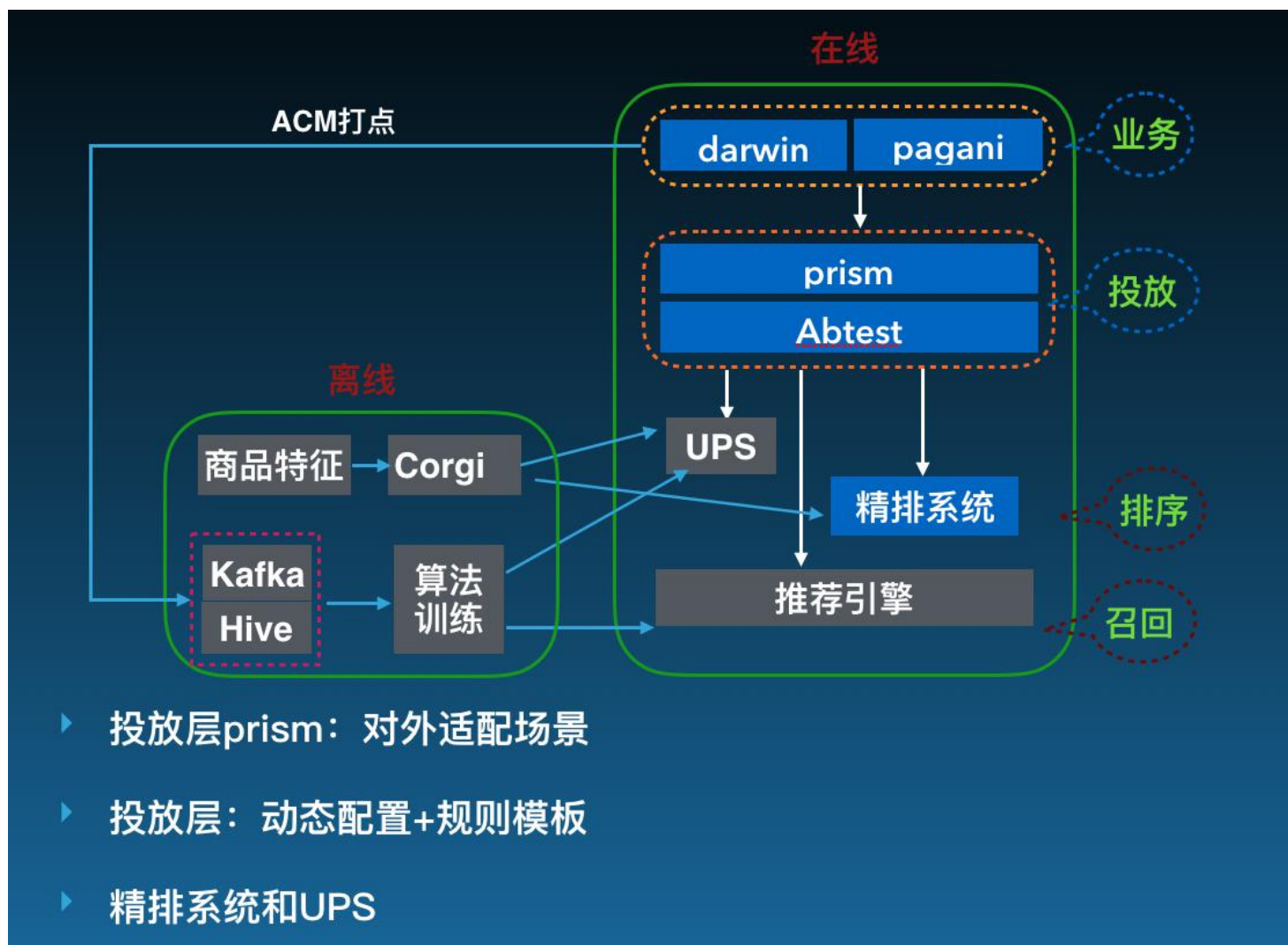
坑:

- ▶ 实时预测模块 离线化

挑战

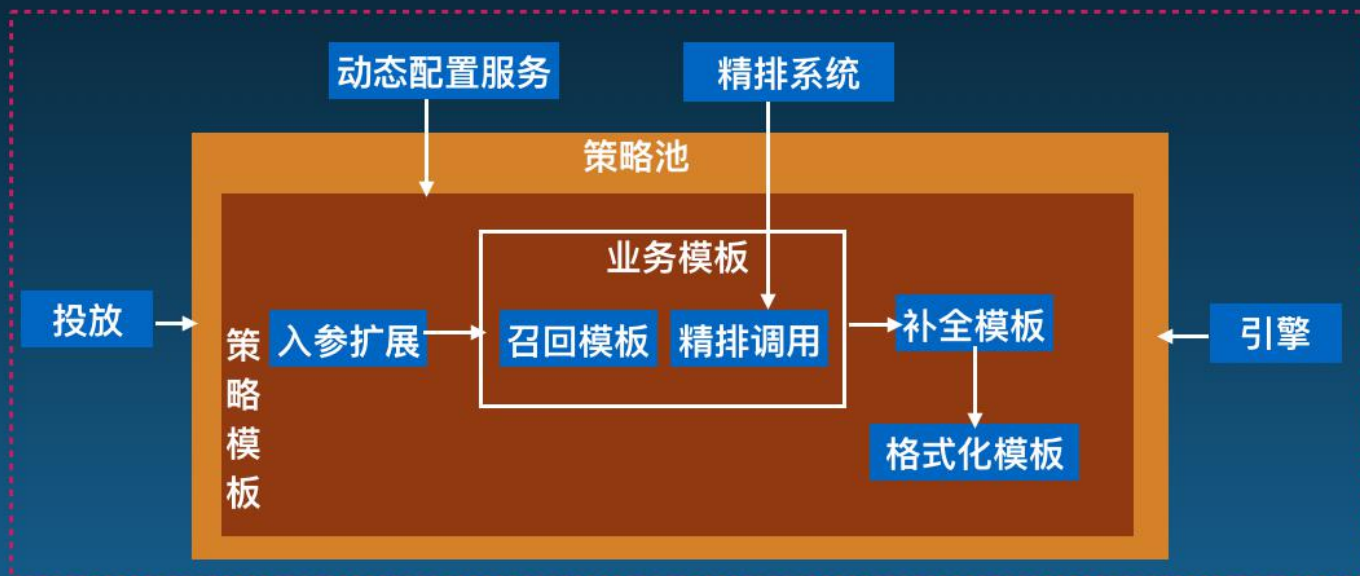
- ▶ 场景增加：
 - ▶ 多类型：猜你喜欢，搜相似，店铺内；
 - ▶ 相似场景：首页/购物车/详情页/...猜你喜欢；
- ▶ 实时排序：实时点击、加购等；
- ▶ 个性化排序：店铺、类目、离线偏好；
- ▶ 问题和挑战：
 - ▶ 多类型多场景：上游系统不一，缺统一对接层，成本高
 - ▶ 场景配置化：场景算法一对一，重复代码拷贝，维护难
 - ▶ 个性化 + 实时：缺系统支持；

架构2.0 (2016.3)



架构2.0—投放层配置化

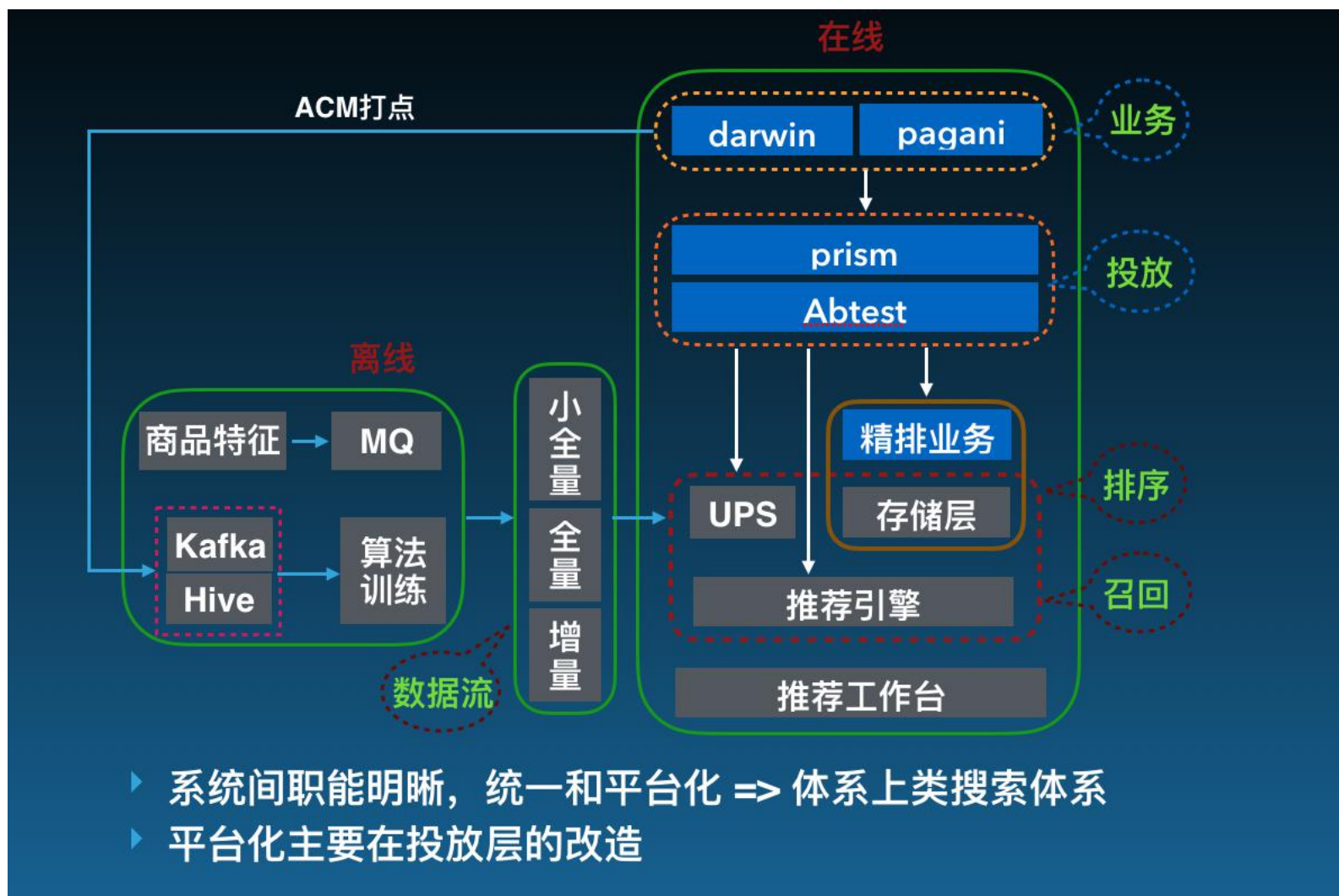
- ▶ 动态配置+规则模板：=> 不变部分模板化，可变部分配置化
 - ▶ 召回组件模板；数据补全模板；格式化模板
 - ▶ 模板方法模式



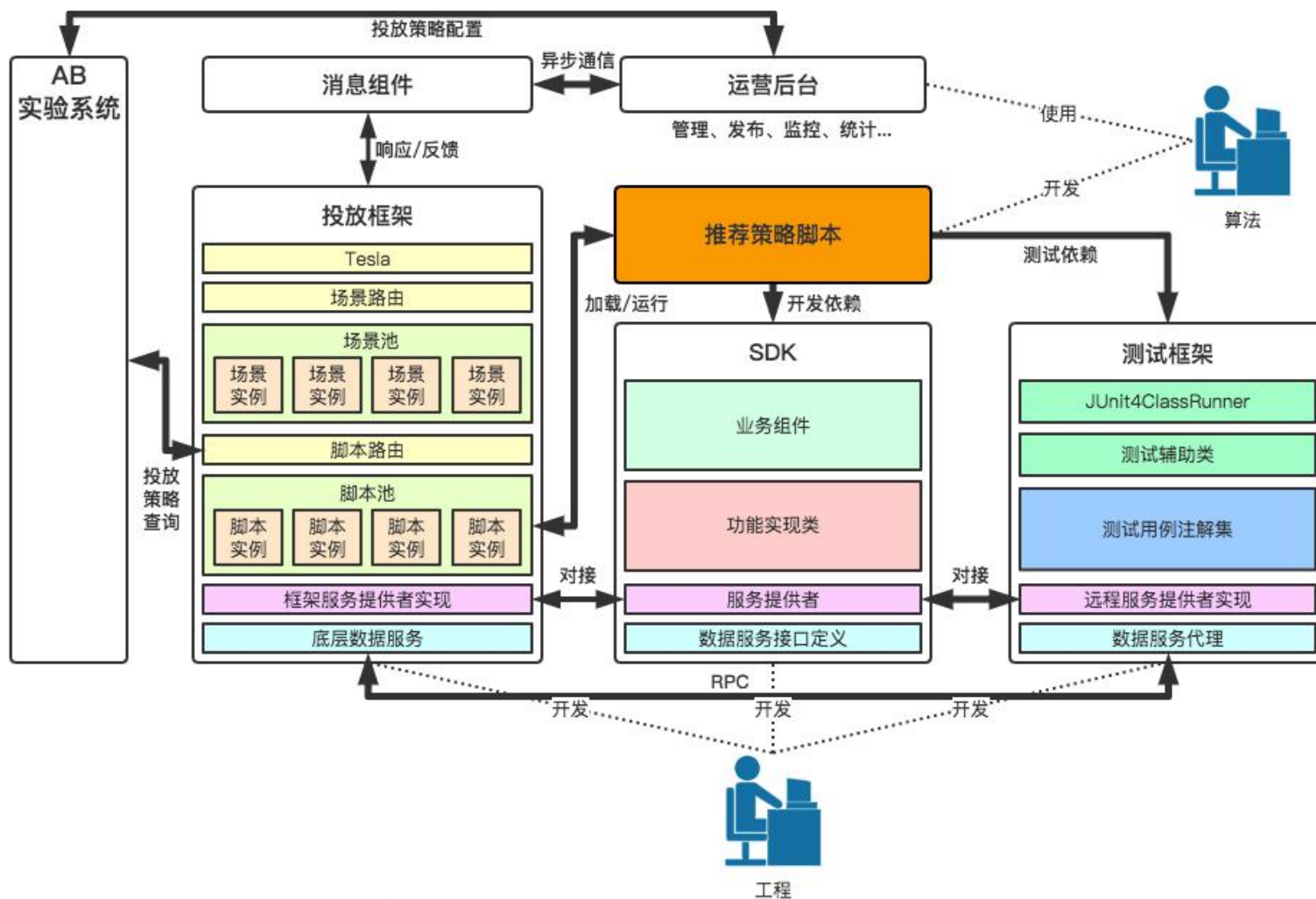
架构2.0—突破和挑战（2016.11）

- ▶ 大促突破：
 - ▶ 321大促运营位置个性化效果提升20%+
 - ▶ 双11大促：会场楼层个性化提升100%+
- ▶ 需求和挑战：
 - ▶ 日益增加的场景和类型：资源位、直播、图像等
 - ▶ 跨团队跨地域
 - ▶ 工程算法一套代码，策略开发调试复杂 => 职责不清
 - ▶ 配置化复杂：简单场景复杂化
- ▶ 方向：
 - ▶ 通用化：推荐解决方案
 - ▶ 平台化：自动化核心业务流程；职责明晰；提升效率；

平台化架构3.0 (16.12~now)



架构3.0-投放层细节



探索总结

- ▶ 快速支持业务
 - ▶ 满足算法需求
 - ▶ 提升效率，降低成本
- ▶ 平台化：面向业务 => 面向算法

后续规划

- ▶ 平台化深入：
 - ▶ 算法策略的评测和压测工具
 - ▶ 全场景智能监控报警&容灾
- ▶ 算法支持：
 - ▶ OnlineLearning & 强化学习
 - ▶ 算法驱动产品：根据算法效果来设计新产品

Thank you !

