



高可用数据服务交易系统架构实践

主讲人：TalkingData 研发总监 何坤



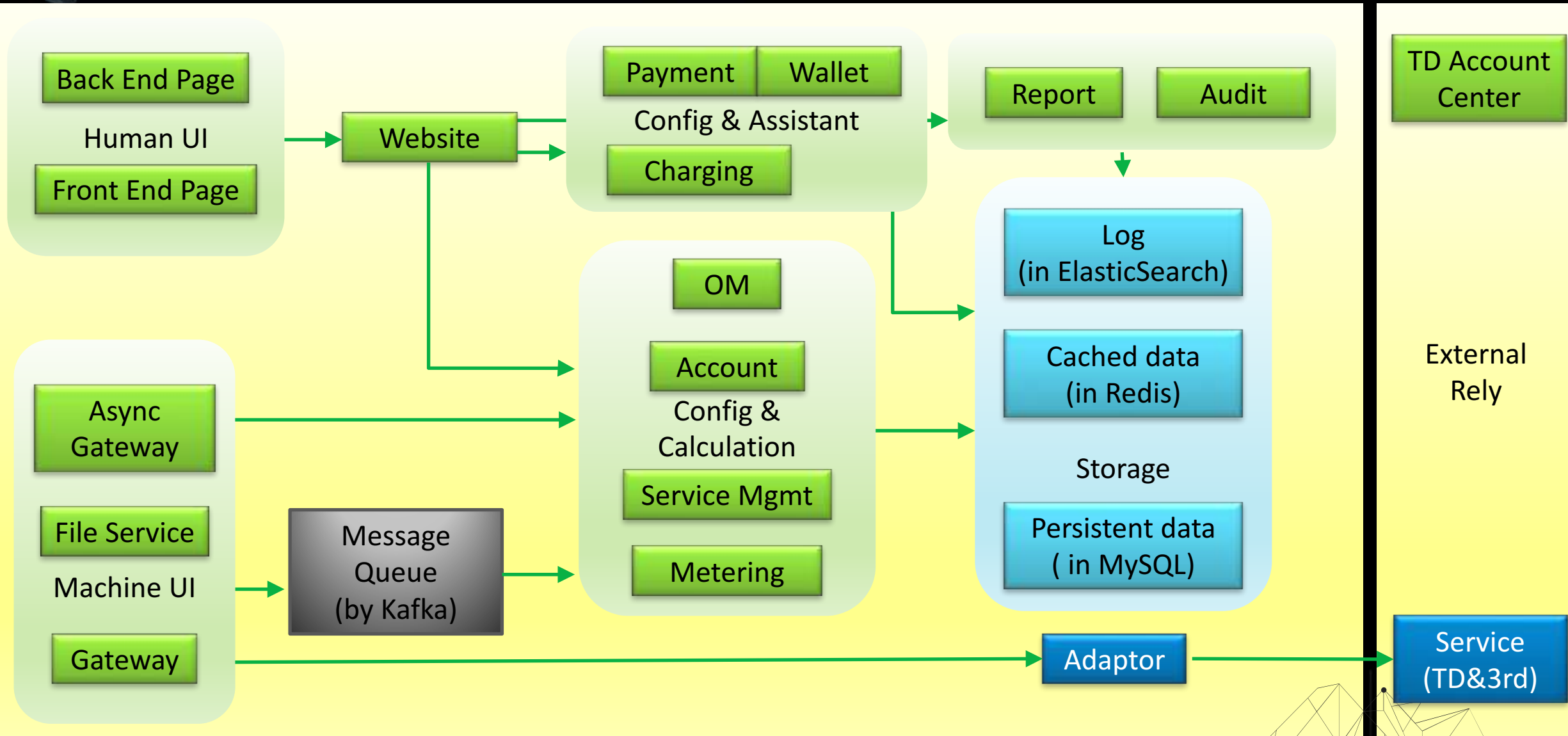
SDMK = Smart Data Market

SDMK提供了API服务，人群数据服务，异步服务等内容；还开放了Lookalike，情景感知，预测引擎，推荐引擎等人工智能服务；降低数据应用场景的难度，帮助企业发现数据的深层价值。





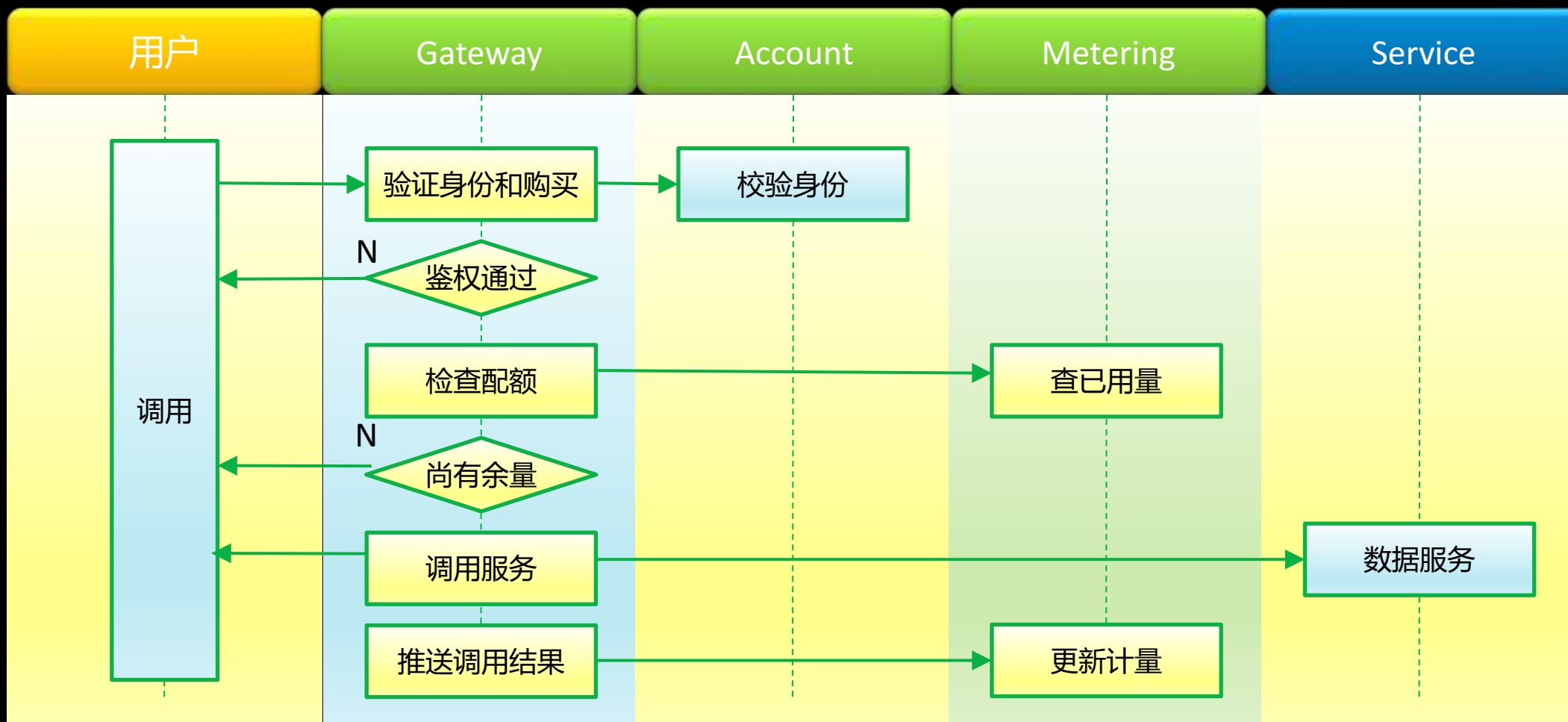
SDMK的功能模块





业务流程

➤ 服务调用基本业务逻辑

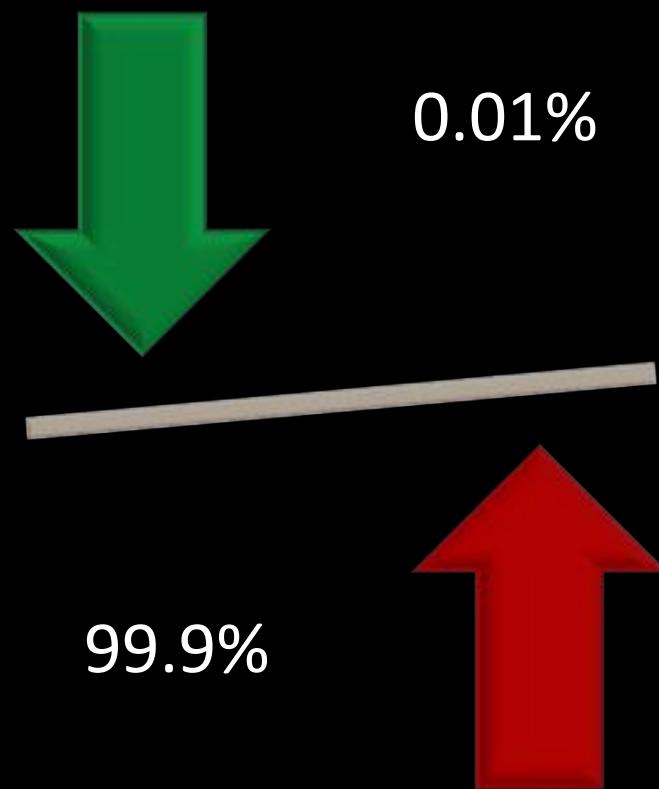




可用性挑战

➤ 要求

- 计量最终误差要求 不高于 0.01%
- 交易系统可用性要求 不低于 99.9%





准确计量

➤ 要求

1. 计量准确无误
2. 交易-计量闭环，要求高并发下实时计量
3. 容错性

➤ 异步计量

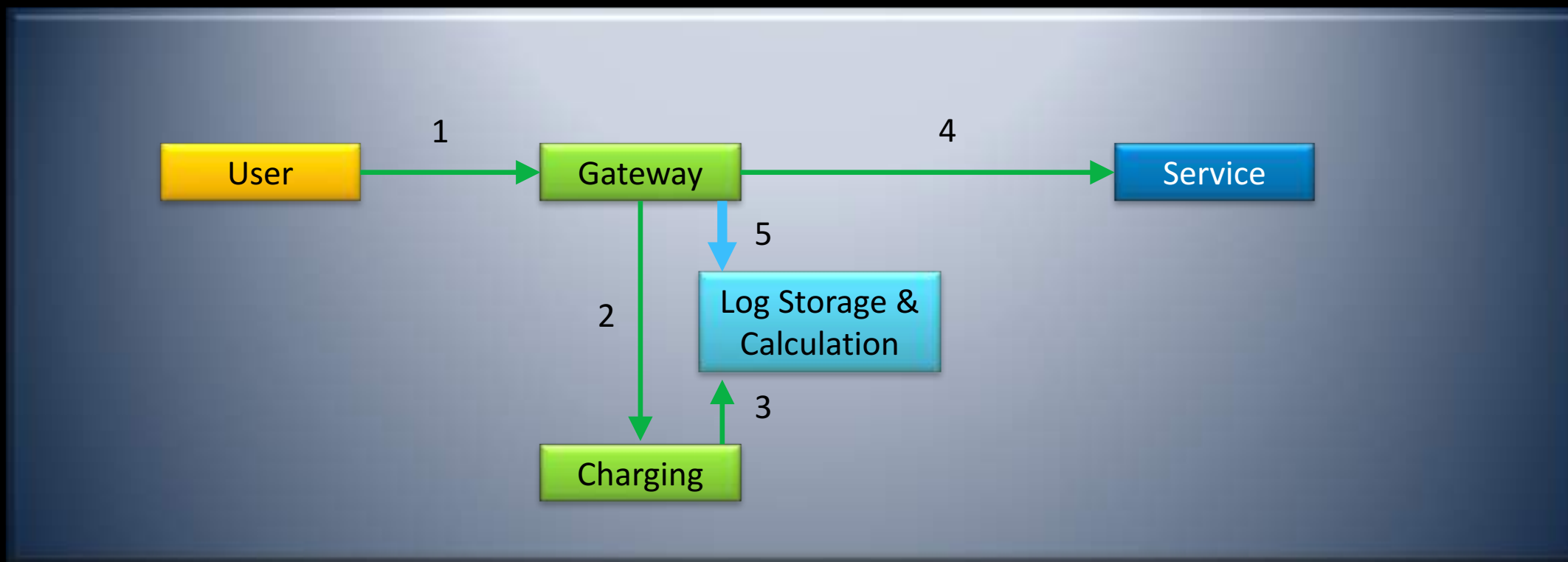
1. 减少系统耦合
2. 降低数据库压力
3. 应对高并发
4. 易于扩展





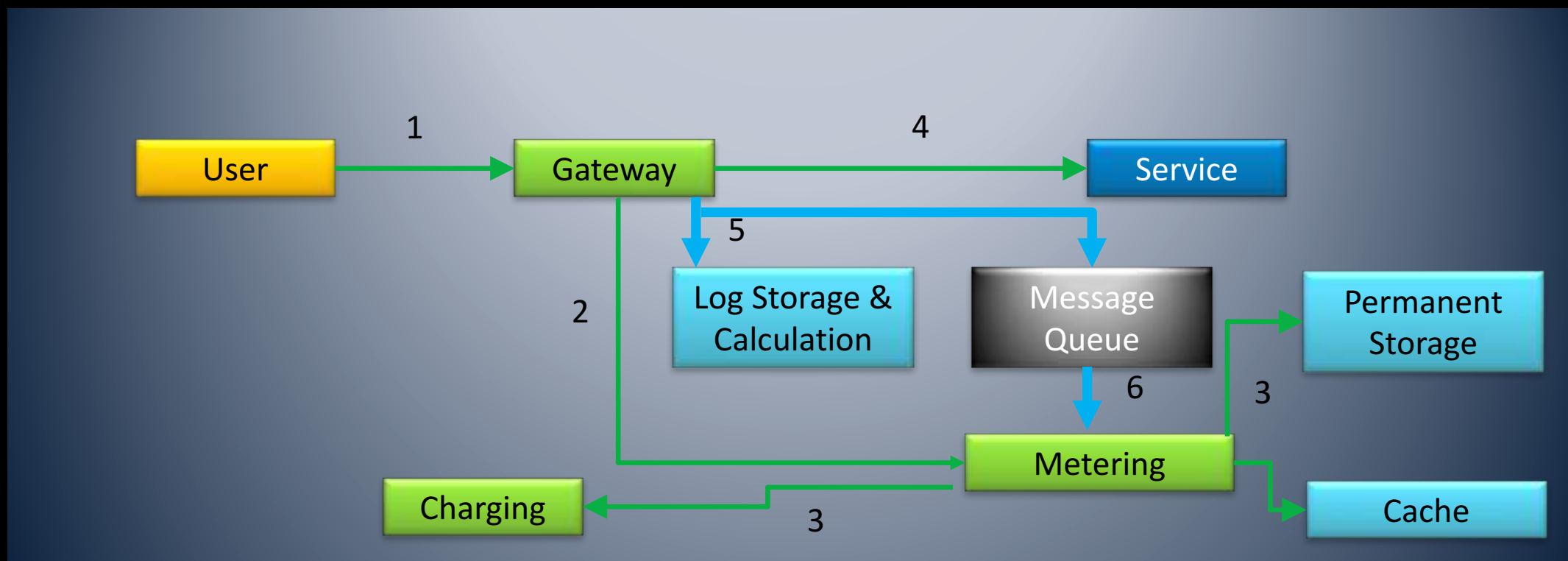
准确计量

➤ 初始架构：实现功能



准确计量

➤ 架构演进：提高效率





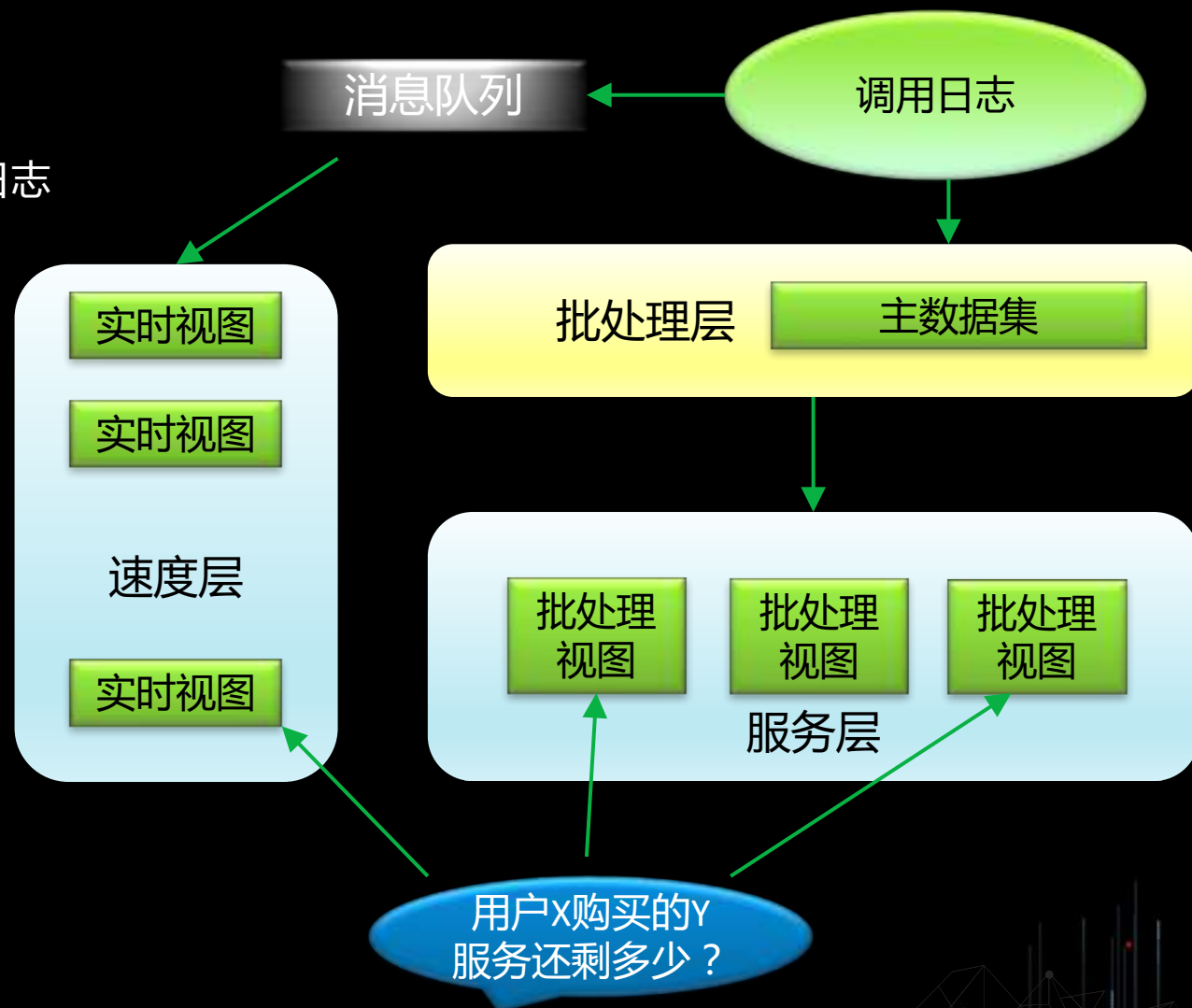
准确计量

➤ Lambda

- 主数据集（不变层）：Elastic Search中的日志
- 批处理层结果：MySQL中按天存储的用量
- 速度层：Redis中的当天和昨天结果
- 服务层：按天进行预计算
- 查询服务：Metering模块

➤ 计算结果的开-闭原则

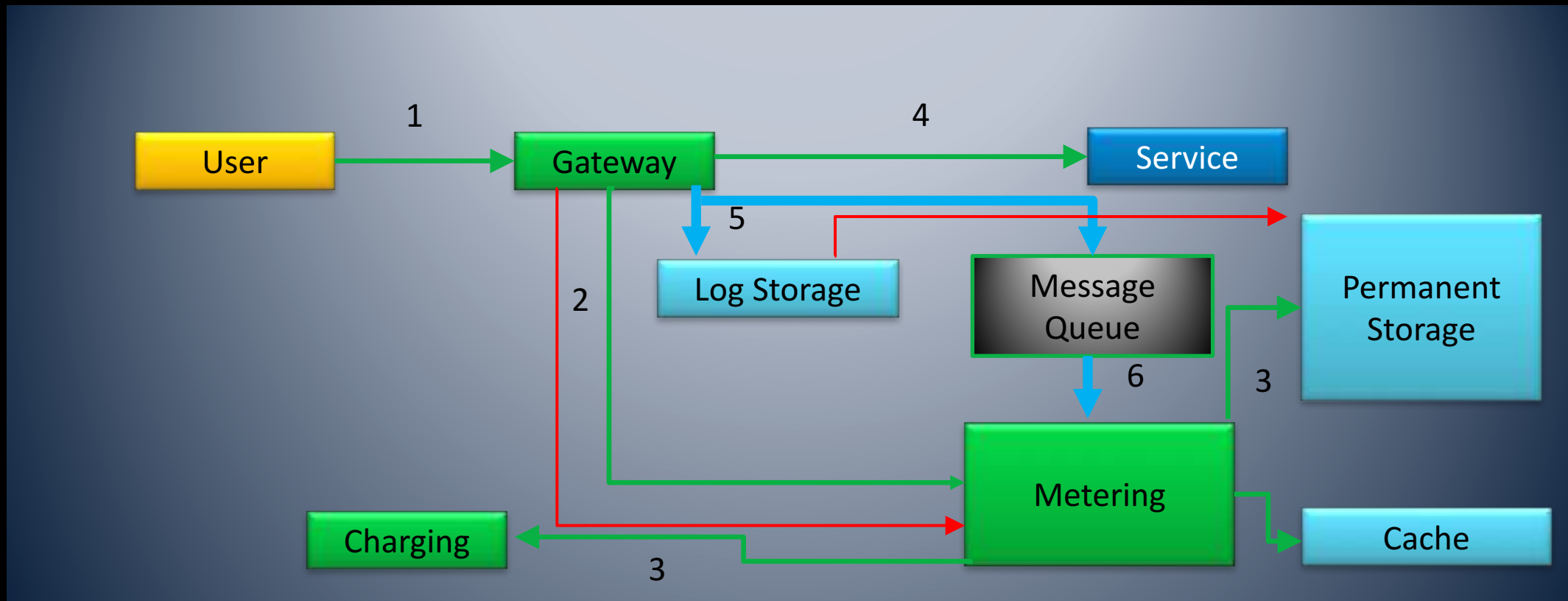
多种计量指标同时计算，按需取用





准确计量

➤ 架构演进：服务降级与重算



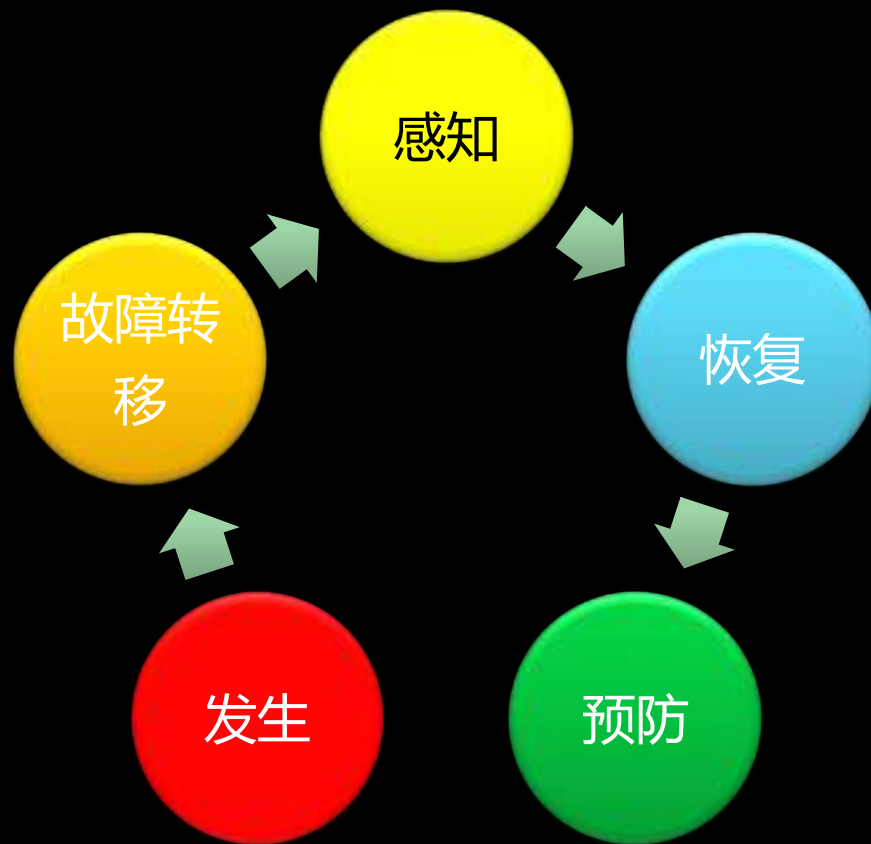


高可用性

➤ 挑战：可用性目标1级

整体服务的高可用性, 99.9% (不可用时间每年低于9个小时, 每月低于1小时)

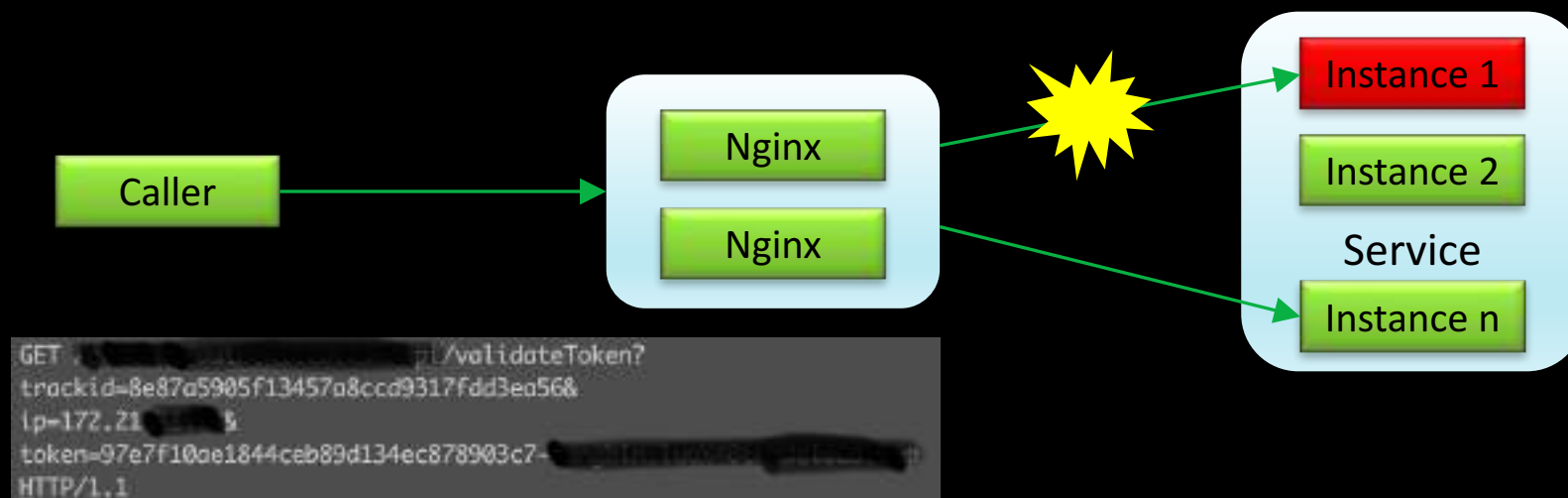
1. 【事前】预防
2. 【事中】自动化故障转移
3. 【事中】故障感知
4. 【事后】故障恢复



高可用性

➤ 分布式部署，无状态设计

1. 所有服务通过nginx调用，多upstream，轮询机制
2. 服务无状态，必要的状态保存在中央存储中(MySQL/Redis等)
3. 所有调用必须带trackid，以便定位问题，缩短故障恢复时间





高可用性

➤ 降低关键路径复杂性与负载

对于SDMK来说，关键路径就是通过Gateway进行的服务调用

1. 专注于核心业务，尽量少加入无关的复杂逻辑与数据依赖
2. 调用其它服务均需设置超时，避免被外部服务故障影响

➤ 适时拆分和合并功能模块

1. 降低模块复杂度
2. 清晰部署边界

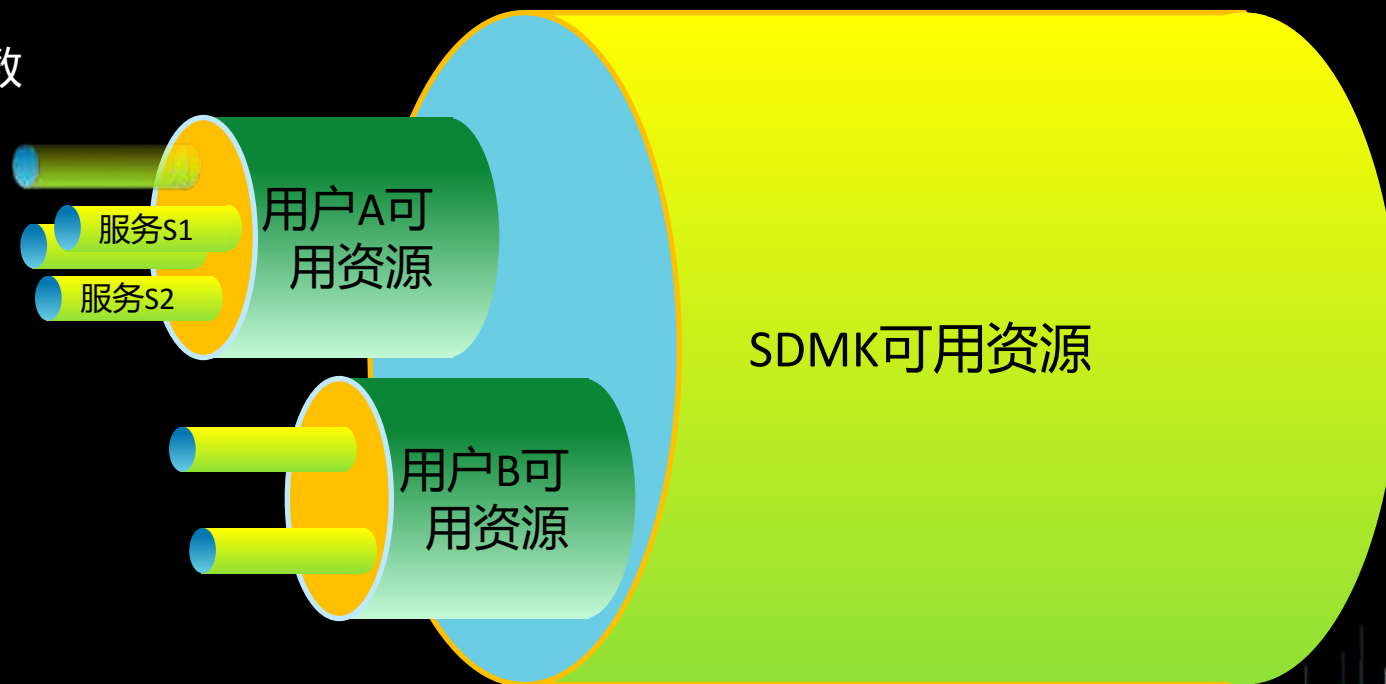


高可用性

➤ 资源限制

对资源的使用进行限制，避免无效或者故障调用耗尽资源

1. 熔断机制
2. 限制用户处于pending状态的请求数
3. 分服务SLA
4. 独立适配器



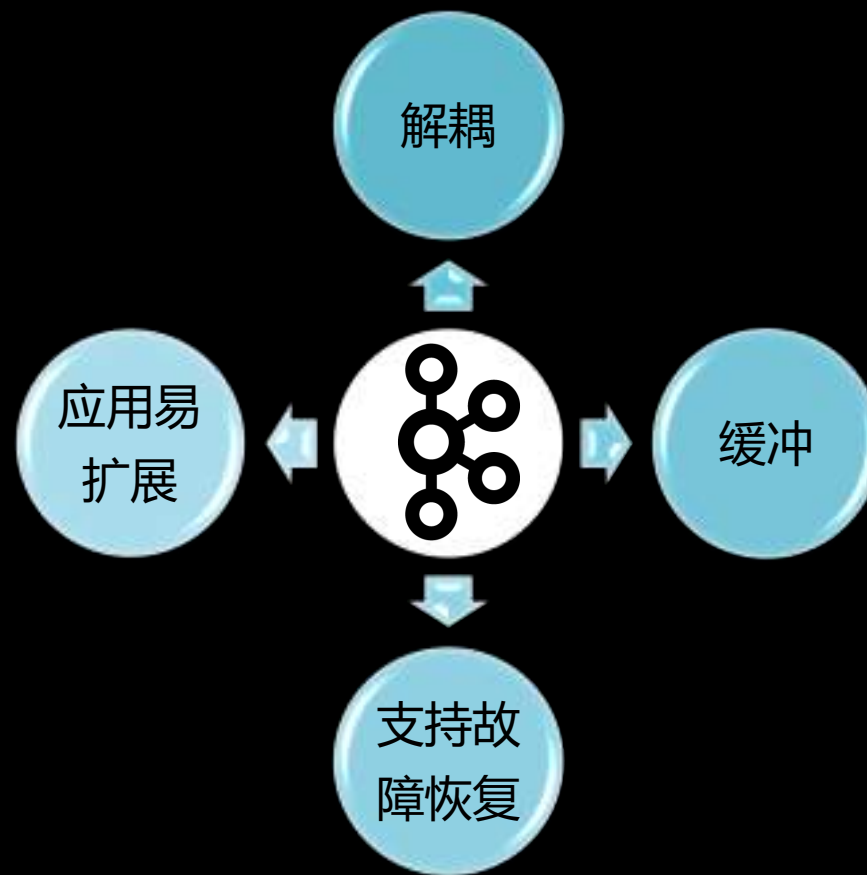


高可用性

➤ 使用消息系统

消息系统的选择

1. 数据可持久化
2. 支持订阅和队列两种方式
3. 高性能
4. 具有水平扩展性





高可用性

➤ 监控与报警

白盒

1. 所有服务上线之前必须有基本监控与报警
2. 基础组件监控与报警
3. 业务指标监控与报警
4. 调用追踪系统



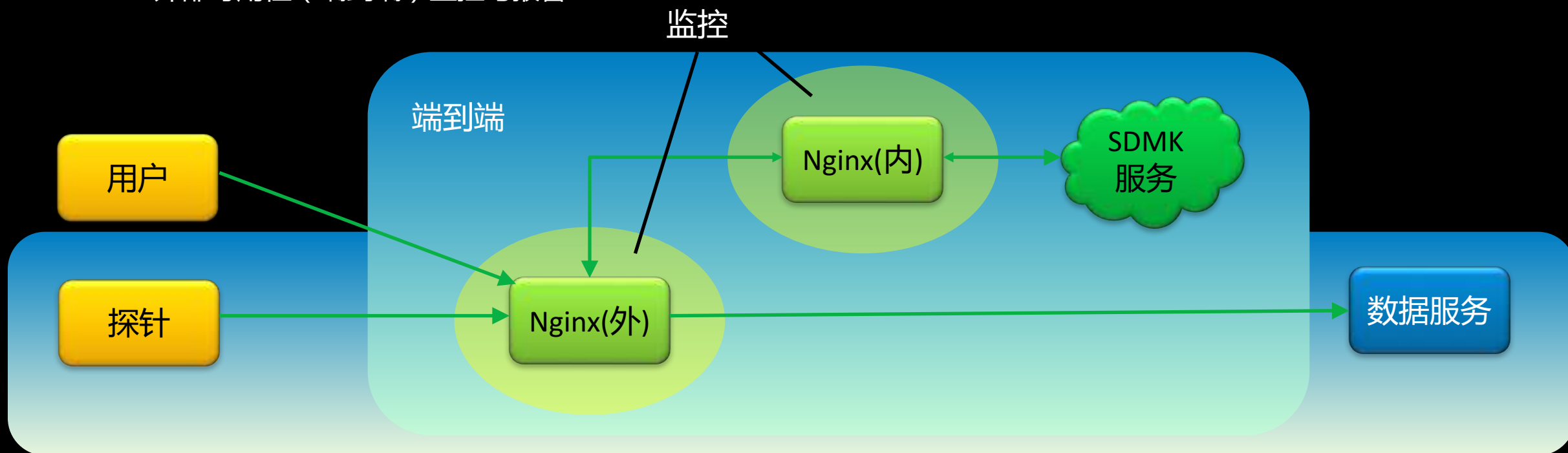


高可用性

➤ 监控与报警

黑盒

1. Nginx监控与报警
2. 探针和心跳监控
3. 外部可用性（端到端）监控与报警



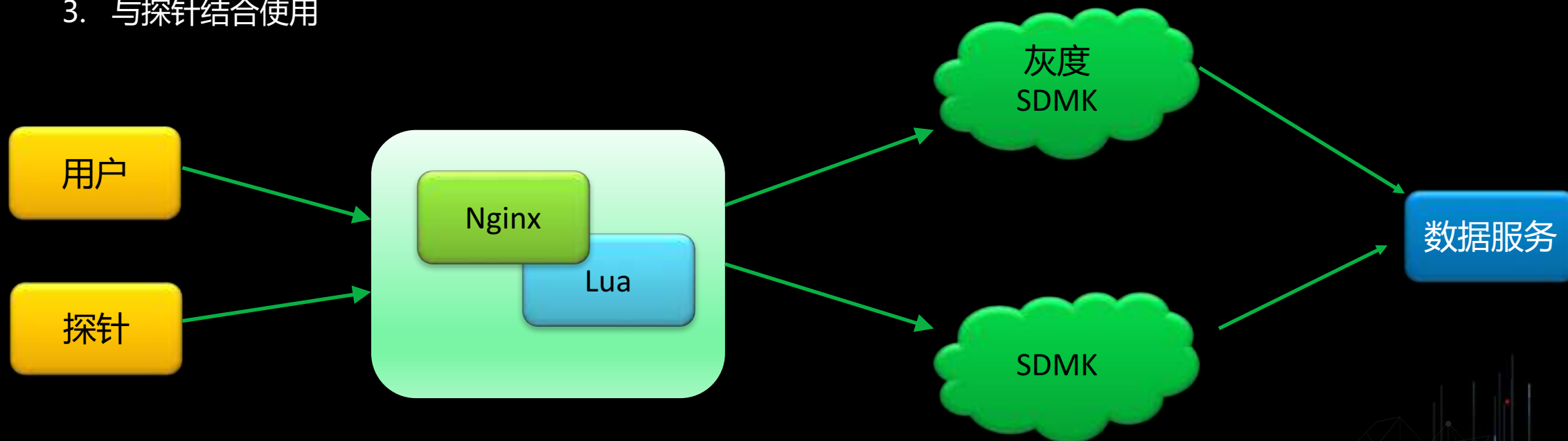


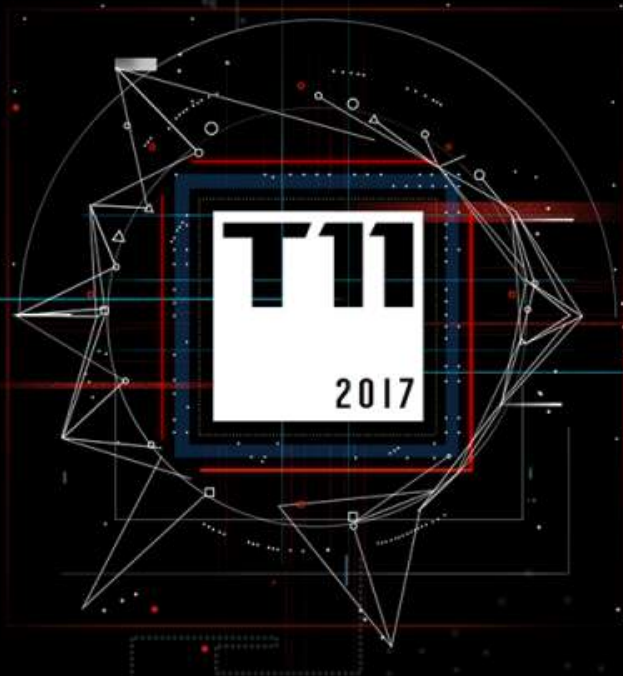
高可用性

➤ 减少更新带来的故障

灰度系统的使用

1. 基于用户标识和Lua的Nginx分流
2. SCM系统的配合
3. 与探针结合使用





THANKS