



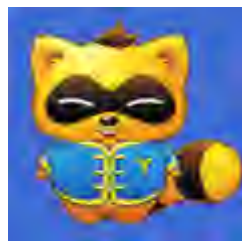
阿里直播平台的架构演进

陈康贤

2015-04

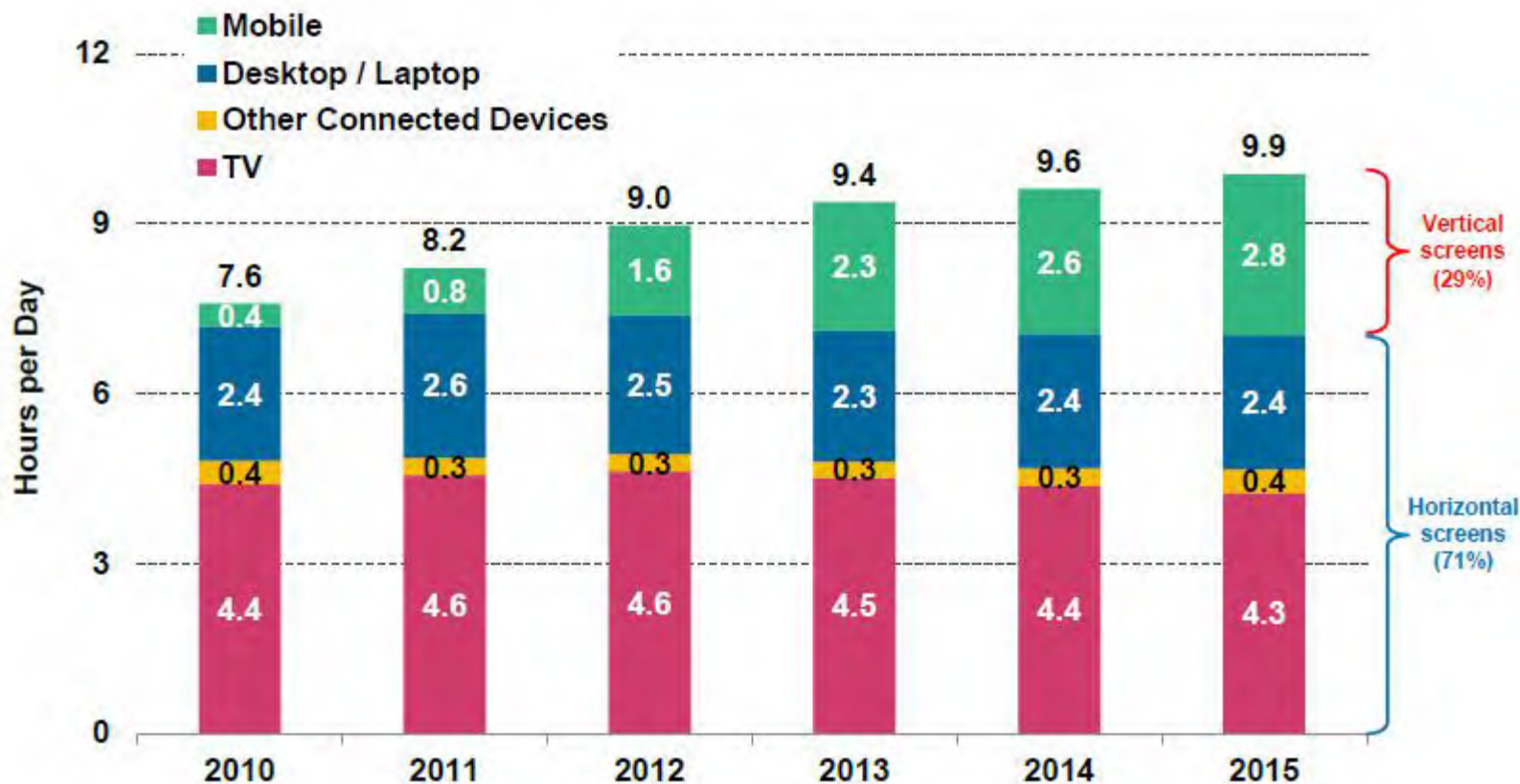


2015直播行业的发展



用户每天花费在不同终端时间占比

Time Spent on Screens by Orientation (Hours / Day), USA, 2010 – 2015



source: INTERNET TRENDS 2015 – CODE CONFERENCE by Mary Meeker



以前的直播是这样



现在的直播是这样？



互联网技术的发展极大地降低了直播的门槛，由庙堂之高走向普罗大众



直播平台架构的演变

从功能到平台

平台

功能



双十一所面临的挑战

淘宝海量的用户基础
对于可靠性近乎苛刻的要求



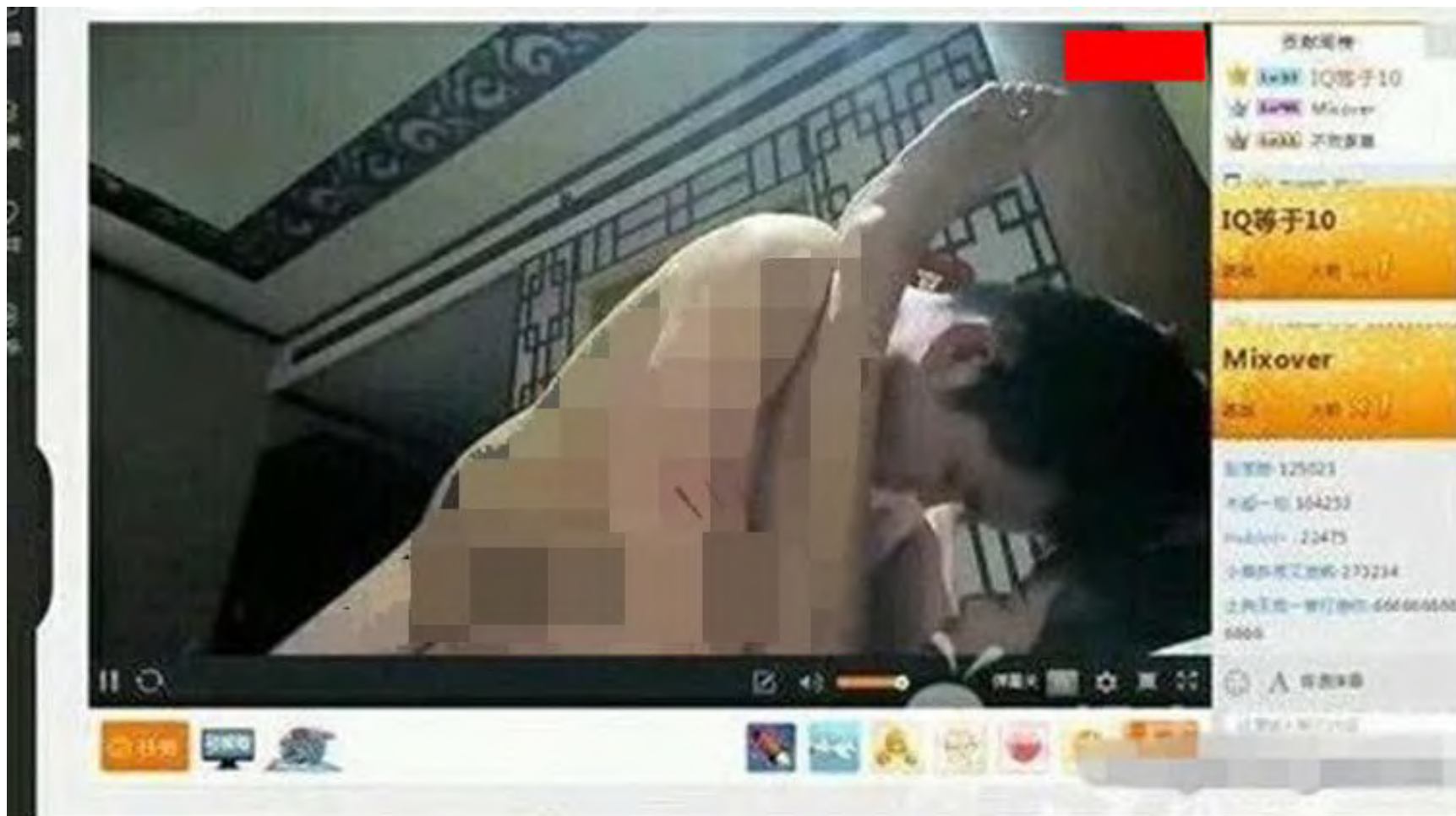
- ✓ 超高并发的消息下行
- ✓ 弹幕内容的过滤
- ✓ 视频内容自动审核
- ✓ 视频延时的控制
- ✓ 包大小的控制
- ✓ 可用性监控
- ✓ 客户端兼容性
- ✓ . . .



弹幕内容监控的重要性



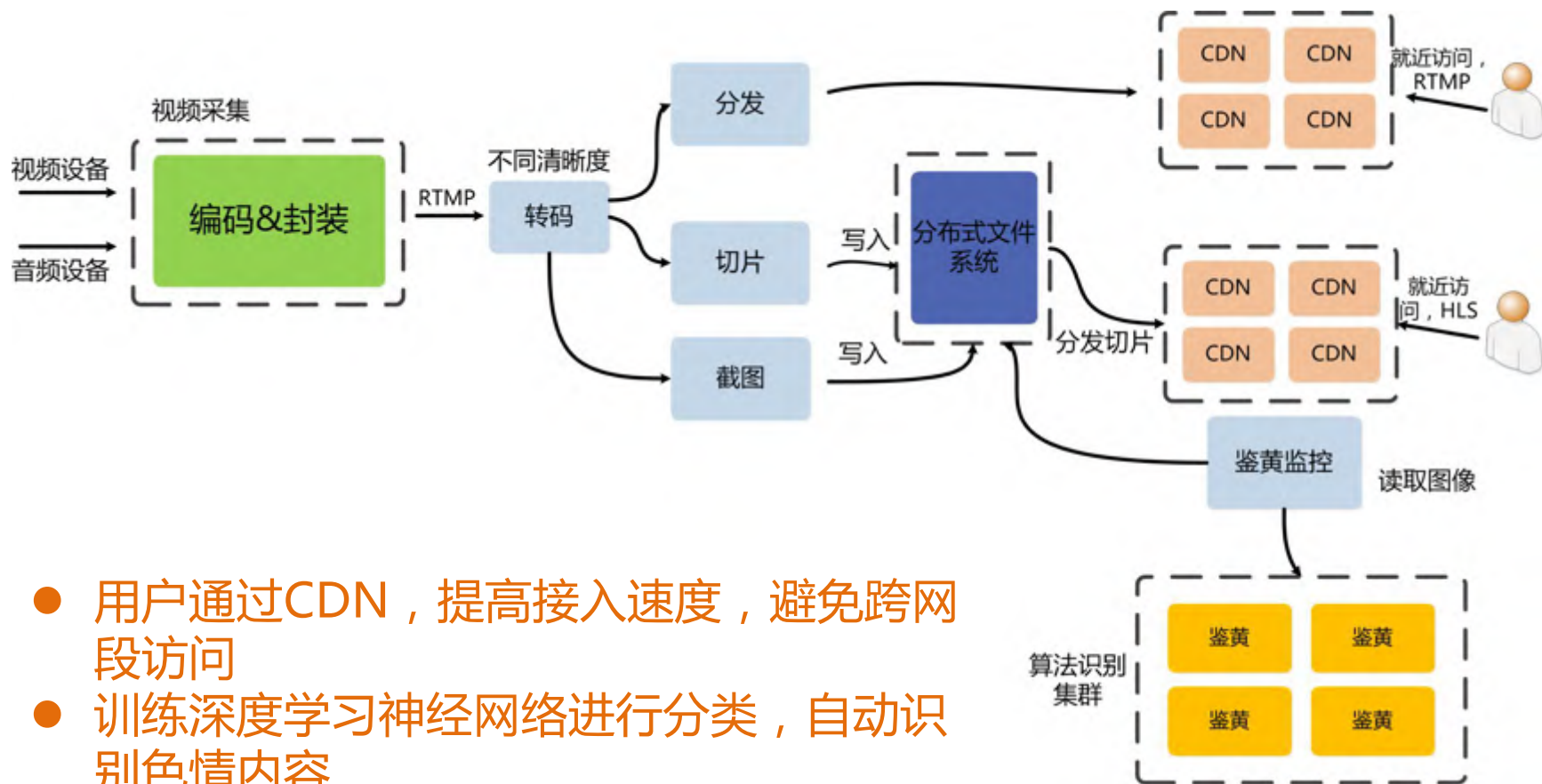
视频内容监控的重要性



当前直播平台整体架构



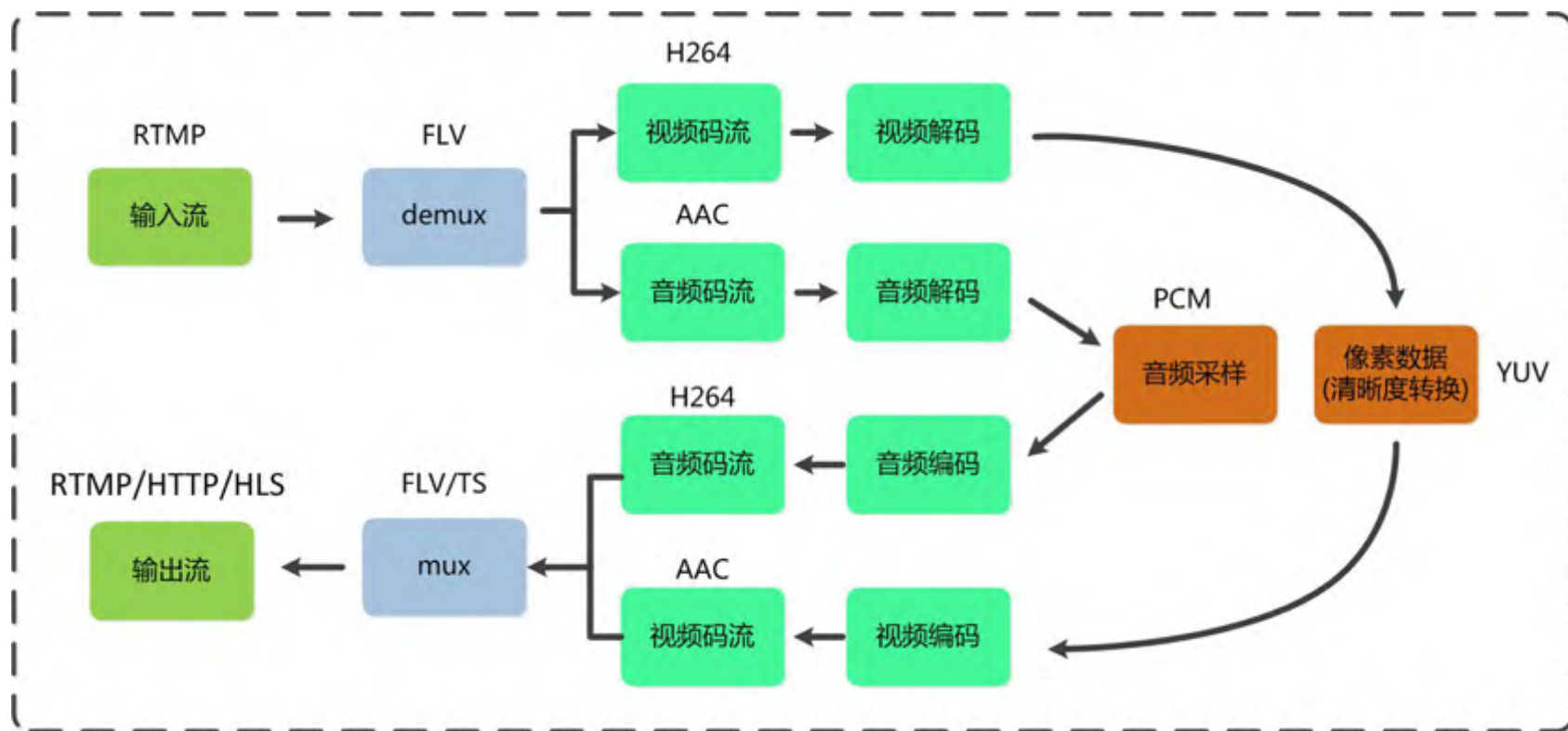
视频直播链路



- 用户通过CDN，提高接入速度，避免跨网段访问
- 训练深度学习神经网络进行分类，自动识别色情内容
- 支持不同协议访问，以满足不同延时需求



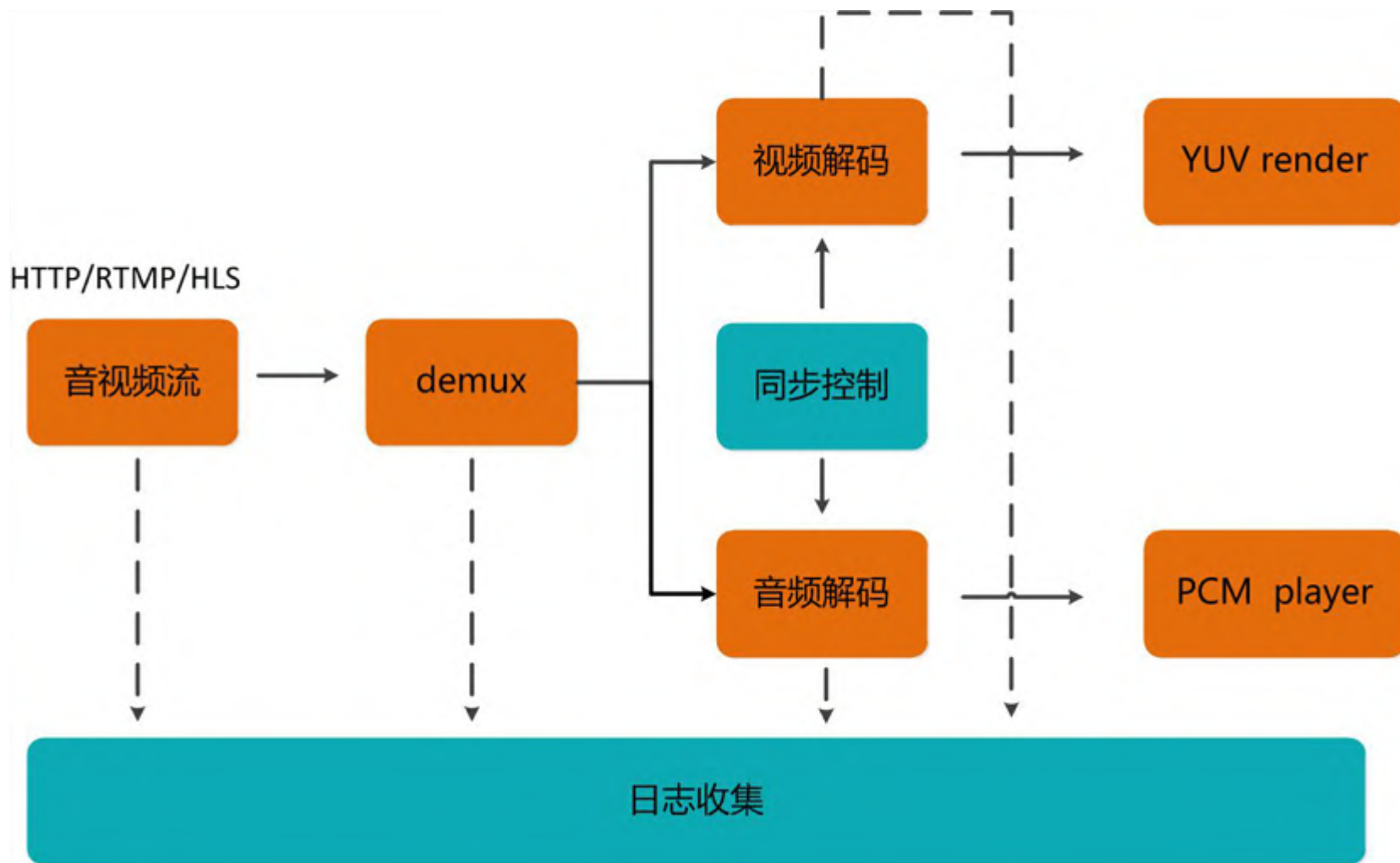
视频流转换成不同清晰度



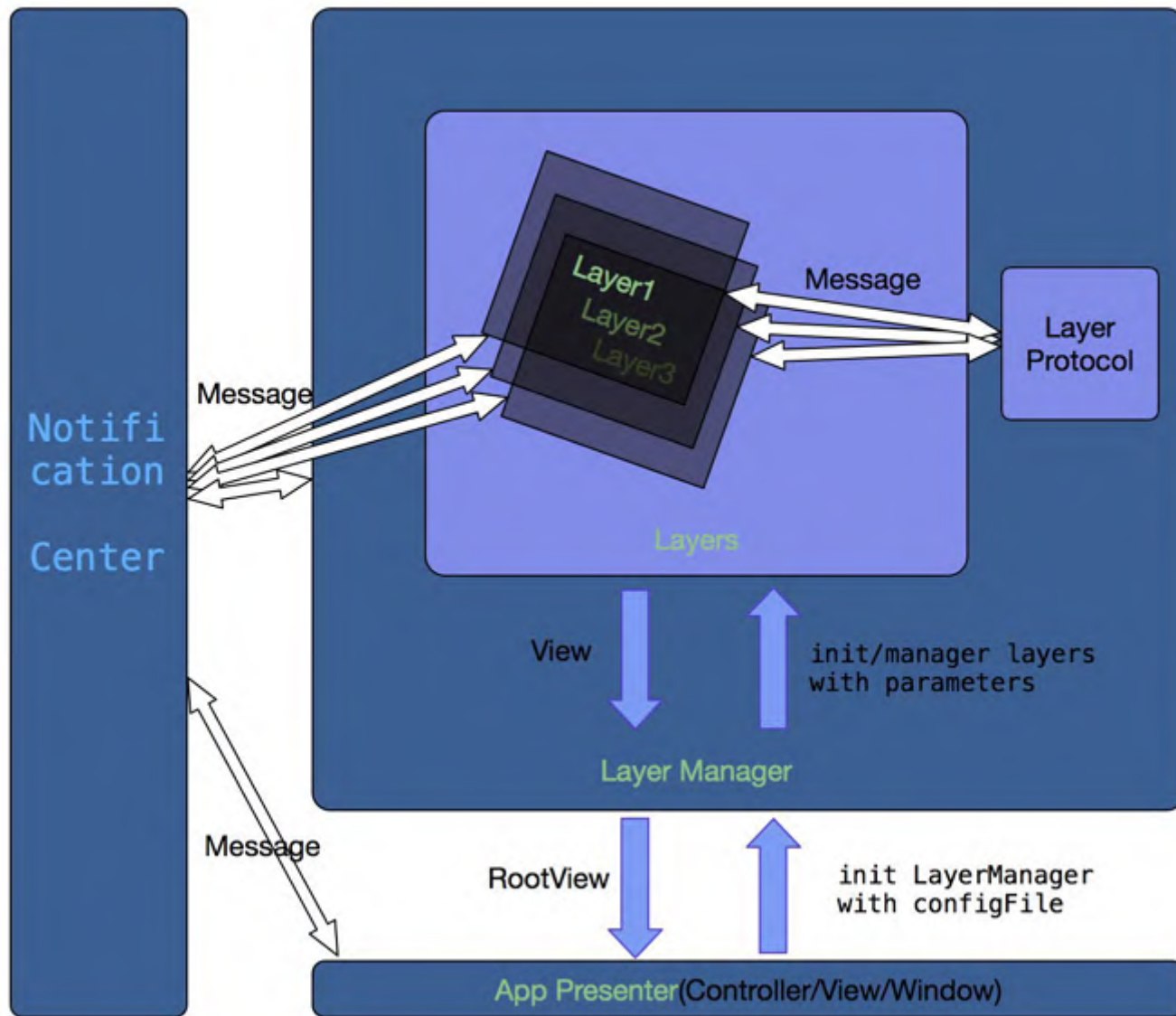
- 不同的端，不同的网络环境，需要不同码率，以保流畅
- 编码的过程非常耗资源，只能靠堆机器



播放器的基本实现



SDK如何在播放器上做层管理



视频相关技术细节



采集

- 码率、帧率根据网速自适应
- 支持不同端、不同硬件设备
- 需传输高品质画质



转码

- 转不同码率，适应不同网络和机型
- CDN分发，就近访问
- HTTP、RTMP、HLS多种格式，支持不同延时的需求

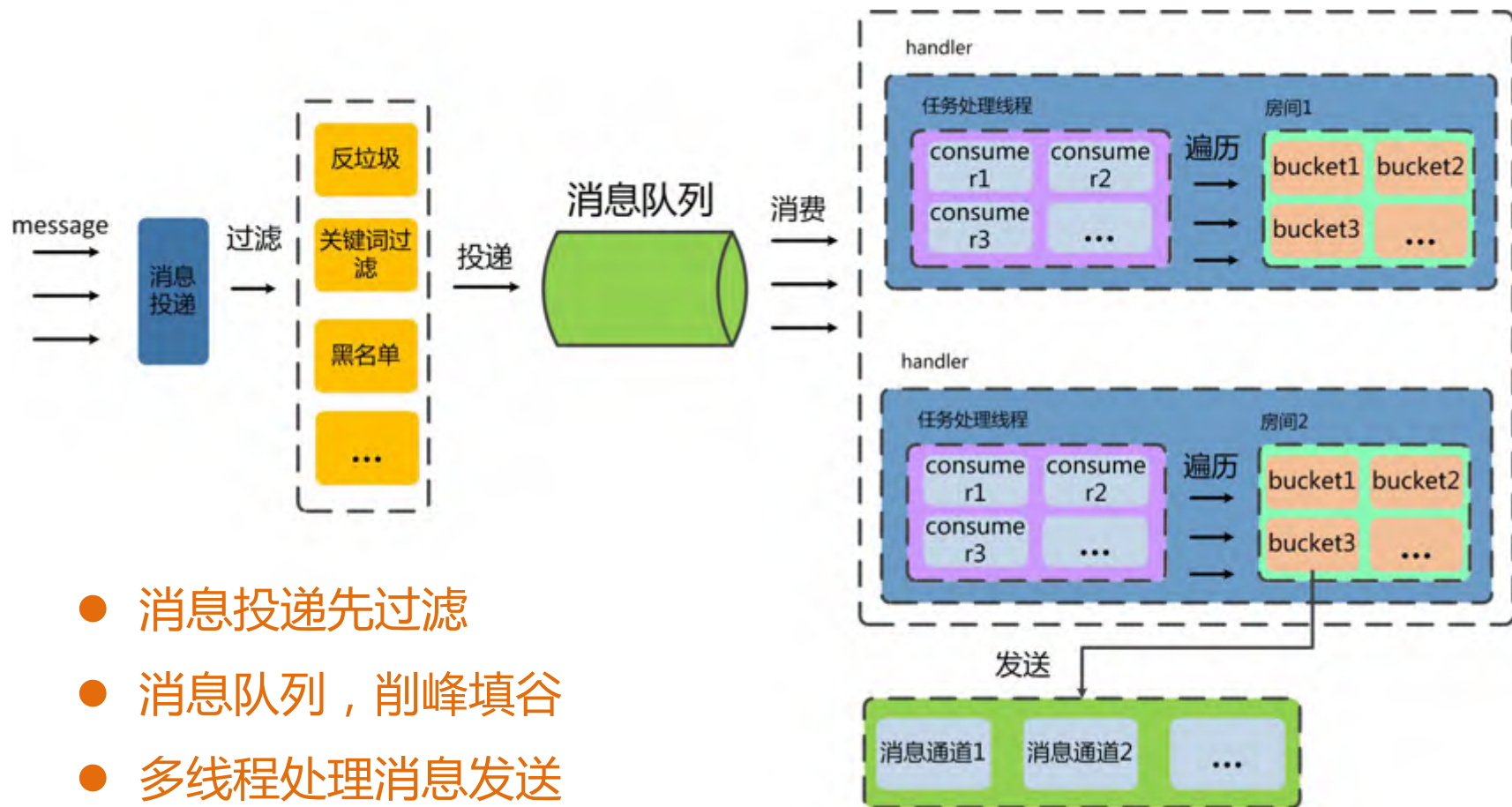


播放

- 硬解软解结合，系统播放器、flash播放器、定制播放器结合
- 播放质量数据收集
- 协议定制，控制程序包大小



消息发送流程



- 消息投递先过滤
- 消息队列，削峰填谷
- 多线程处理消息发送
- 房间分桶，降低下行压力
- 不同的消息通道，对消息发送方透明

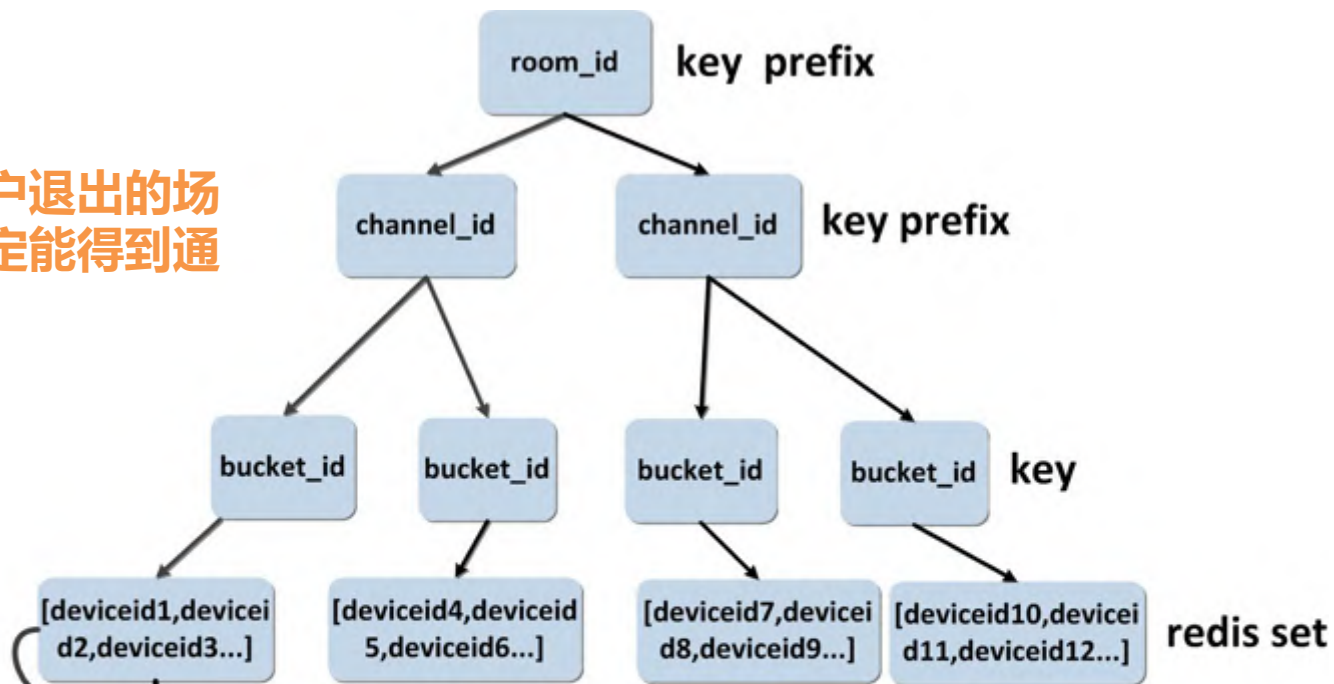


不同消息通道的优劣对比



心跳及房间结构

为何需要心跳：
维持房间结构，而用户退出的场景却各不相同，不一定能得到通知



人员列表

room_id+channel_id+bucket_id [deviceid1,deviceid2,deviceid3...]

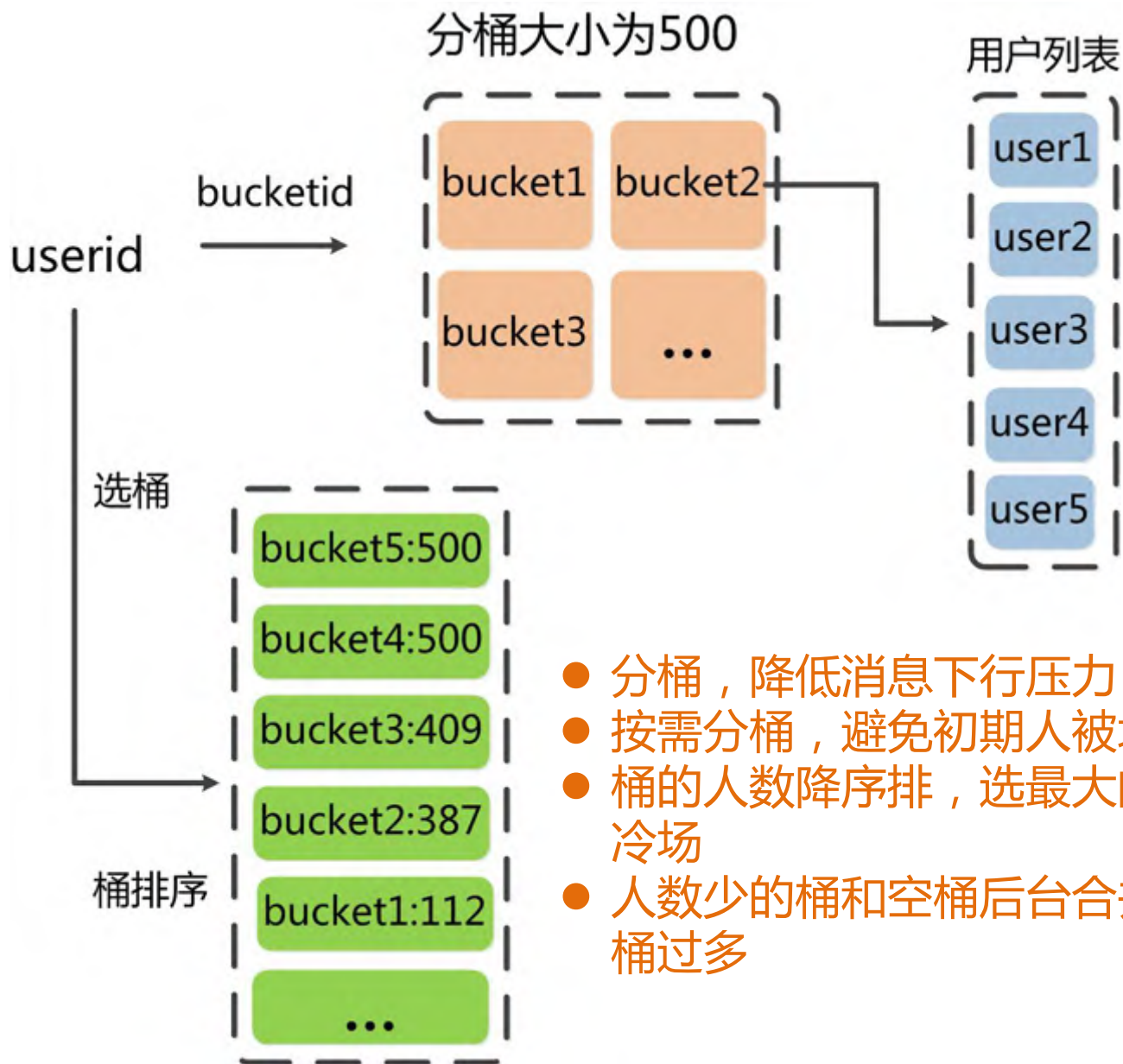
失效单元

room_id+channel_id+bucket_id+device_id value expiretime

- 心跳维持房间的树形结构，保持用户在线状态
- 受redis的限制，set的单个元素无法被动失效，需要另维护一个单元，主动失效



用户按需分桶



- 分桶，降低消息下行压力
- 按需分桶，避免初期人被均分
- 桶的人数降序排，选最大的桶，避免冷场
- 人数少的桶和空桶后台合并，避免空桶过多



固定分桶与按需分桶对比



固定分桶

• 优点

1. 分桶固定，不需要考虑桶的回收，算法实现简单

• 缺点

1. 用户刚进入会比较冷清，影响房间活跃度
2. 前期很难预估活动热度，分桶数确定后，调整分桶会比较麻烦，需要进行内存迁移



按需分桶

• 优点

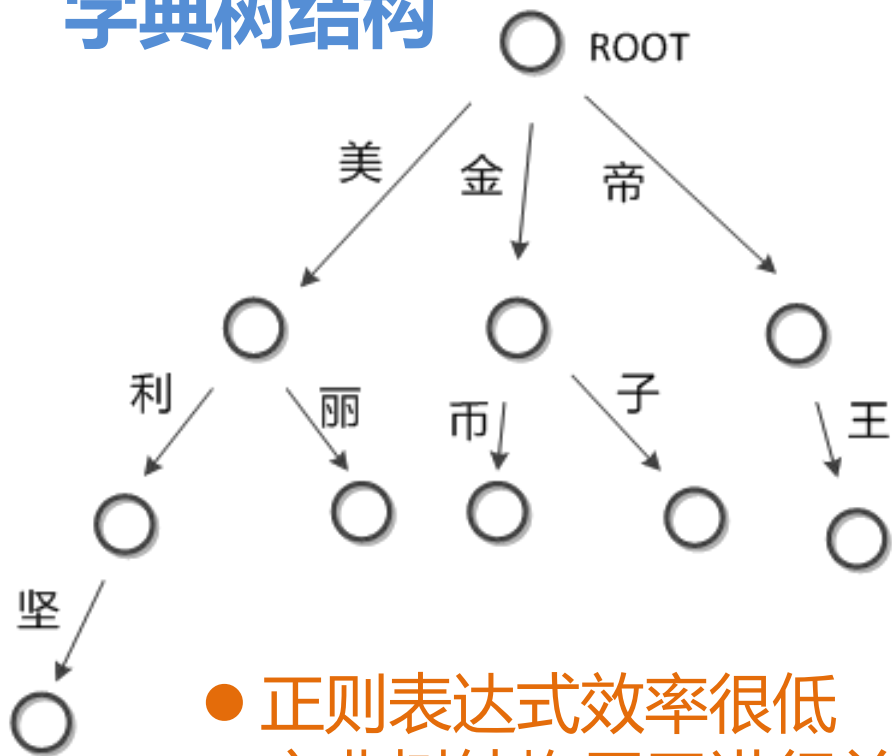
1. 根据人数来自动生成分桶，满了开辟新桶，每个桶的活跃度可尽量保持一致
2. 省去了固定分桶一旦分桶数调整，内存迁移的麻烦

• 缺点

1. 直播的过程中用户会离开，需要进行桶整理，算法实现复杂度堪比JVM内存回收算法



字典树结构



贝叶斯公式

$$p(c|x) = \frac{p(x|c)p(c)}{p(x)}$$

- 正则表达式效率很低
- 字典树结构用于进行关键字过滤
- 贝叶斯分类进行反垃圾过滤
- 高并发时避免远程调用，词库和语料库本地缓存



稳定性

- 优雅降级预案，提前演练
- 开关、脚本提前准备
- 多机房容灾部署
- 重点直播间双路备份

监控

- 在线人数、系统压力可视化
- 直播间、消息通道全链路可用性监控
- 内容监控，黄色内容、违法信息

压测

- 基本压力测试，找出瓶颈
- 全链路压测，摸清容量
- 梳理链路，评估压力，增加机器

数据

- qps、load、在线人数、消息量记录，来年便是基准经验值



平台化的挑战



- 复杂的交互抽象成SDK
- 模块化可配置
- 文档清晰
- 支持多终端(Android/iOS/h5...)
- 多端统一功能
- 统一权限和角色
- H5的兼容性
- 跨终端与时延的矛盾
- 音视频可用性、质量
- 业务资源隔离
- 内容审核
- 流量控制





Thank you!

