

软硬兼施 分布式缓存和流式计算架构

洪倍 @精硕科技 CTO 联合创始人

- 分布式缓存场景和需求

- 缓存架构演进之路

- 架构设计瓶颈剖析

- 软硬兼施 对症下药

- 分布式缓存场景和需求

- 缓存架构演进之路

- 架构设计瓶颈剖析

- 软硬兼施 对症下药

缓存 - 避免慢存储弱索引重复查询

CAP三特性中，可以牺牲C数据一致性

100ms的千人千面 = 20ms请求 + 60ms查询 + 20ms回应

论坛

今日: 875 | 昨日: 1016 | 帖子: 4933702 | 会员: 416839 | 欢迎新会员: xuzhoukill

我的帖子 | 最新回复

最新主题

- 太阳岛客运大电菜们 0/0
- 为什么有些地铁站每天买单 5/55
- 新图 2/384
- 【未知原因】部分车站大字 2/162
- 龙阳路枢纽地块：137万平方 13/559
- 龙阳路枢纽地块：137万平方 1/75
- 新村路真华路，晚上7点还在 0/202
- 2018，让我们告别旧的巴士 7/483
- 预定十二黄金圣斗士地铁纪 9/292
- 快18年了 白沙公园开了没? 4/278

最新回复

- 龙阳路枢纽地块：137万平方 1/75
- 2018，让我们告别旧的巴士 7/483
- 龙阳路枢纽地块：137万平方 13/559
- 汇金路站往虹桥火车站和往 13/668
- 车墩枢纽这样的，大气啊！ 30/2253
- 金山石化房价一目了然呀 12/725
- 东长治路公交即将再减两条 4/451
- 这年头最可怕的就是无知加 13/888
- 预定十二黄金圣斗士地铁纪 9/292
- 为什么有些地铁站每天买单 5/55

今日热门

- 车墩枢纽这样的，大气啊！ 30/2253
- 九/龙仓里有两栋楼是公租房 17/1148
- 浦江镇有几条线路早晚高峰 15/557
- 这年头最可怕的就是无知加 13/888
- 汇金路站往虹桥火车站和往 13/668
- 金山石化房价一目了然呀 12/725
- 龙阳路枢纽地块：137万平方 12/483
- 婺源站 10/634
- 15号了，航头龙华医院年底 10/784
- 新桥铁路南一半什么时候治 9/304

论坛焦点



新图



上海到札幌机票携程1折起

携程国际机票往返同行底价，
中转国内 加段0元起，支持多
段航程同时预订

广告 X



关注我们

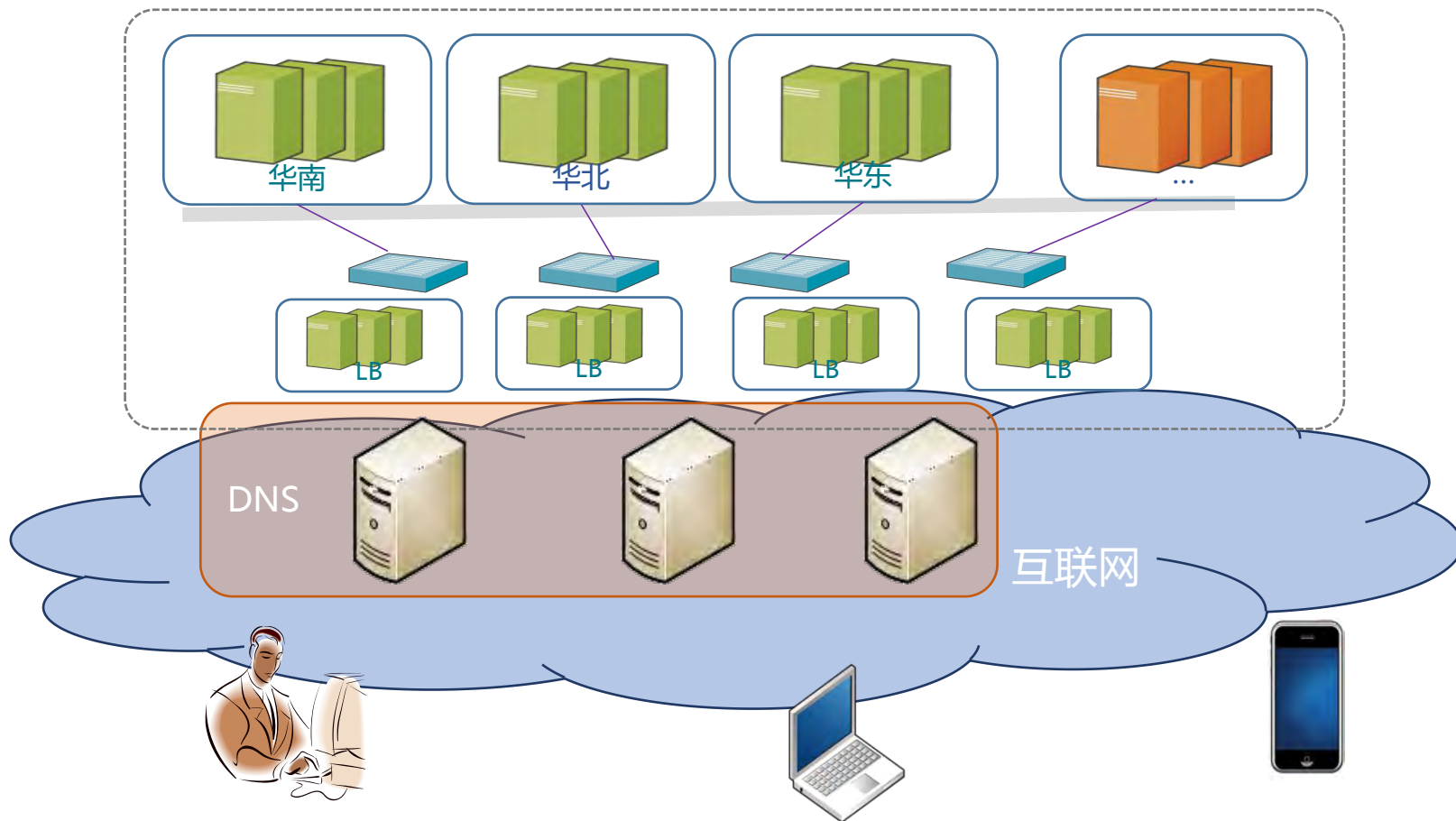


新浪微博
weibo.com/metrofans

+ 加关注

分布式 - 幅员辽阔的中国和不安全感

CAP三特性中，解决A&P的问题



分布式缓存解决的核心问题



- 分布式缓存场景和需求

- 缓存架构演进之路

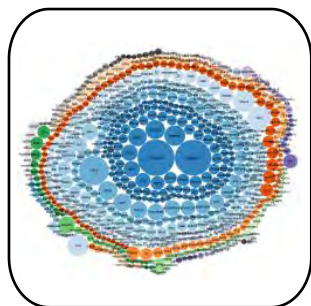
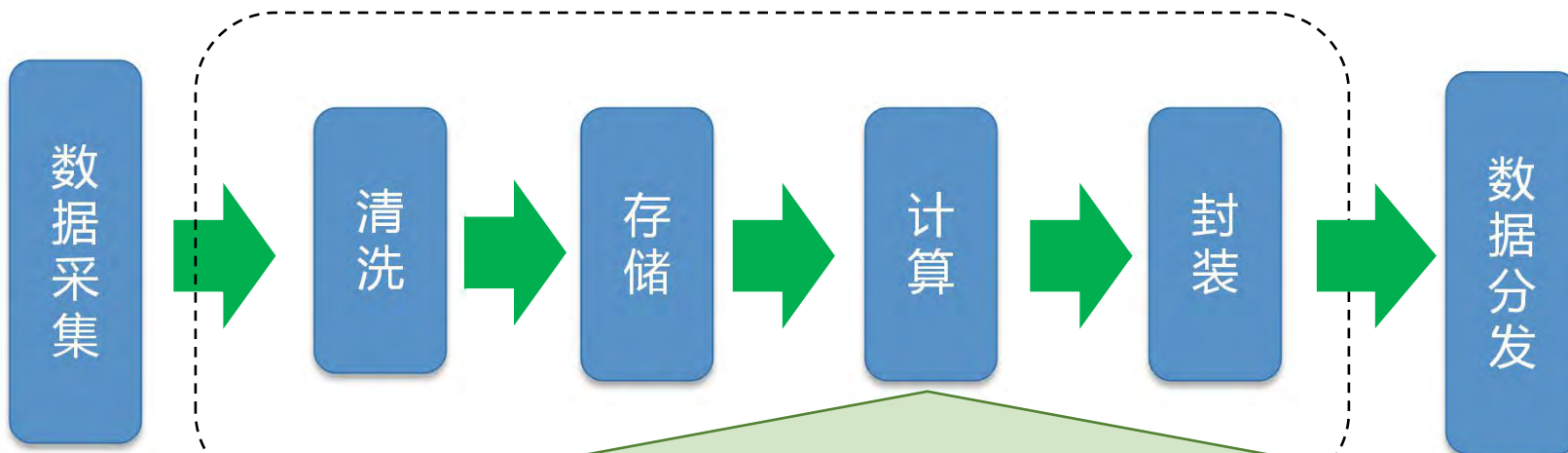
- 架构设计瓶颈剖析

- 软硬兼施 对症下药

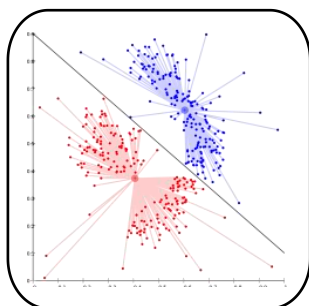
数据缓存的架构演进方向



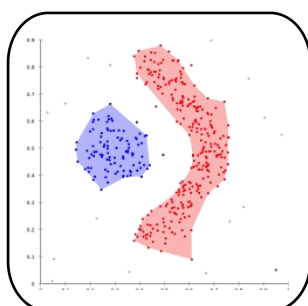
频繁更新的计算式检索



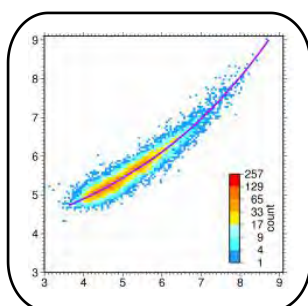
自然语言处理



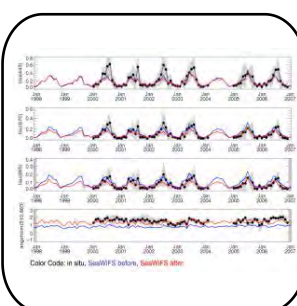
分类问题



聚类问题

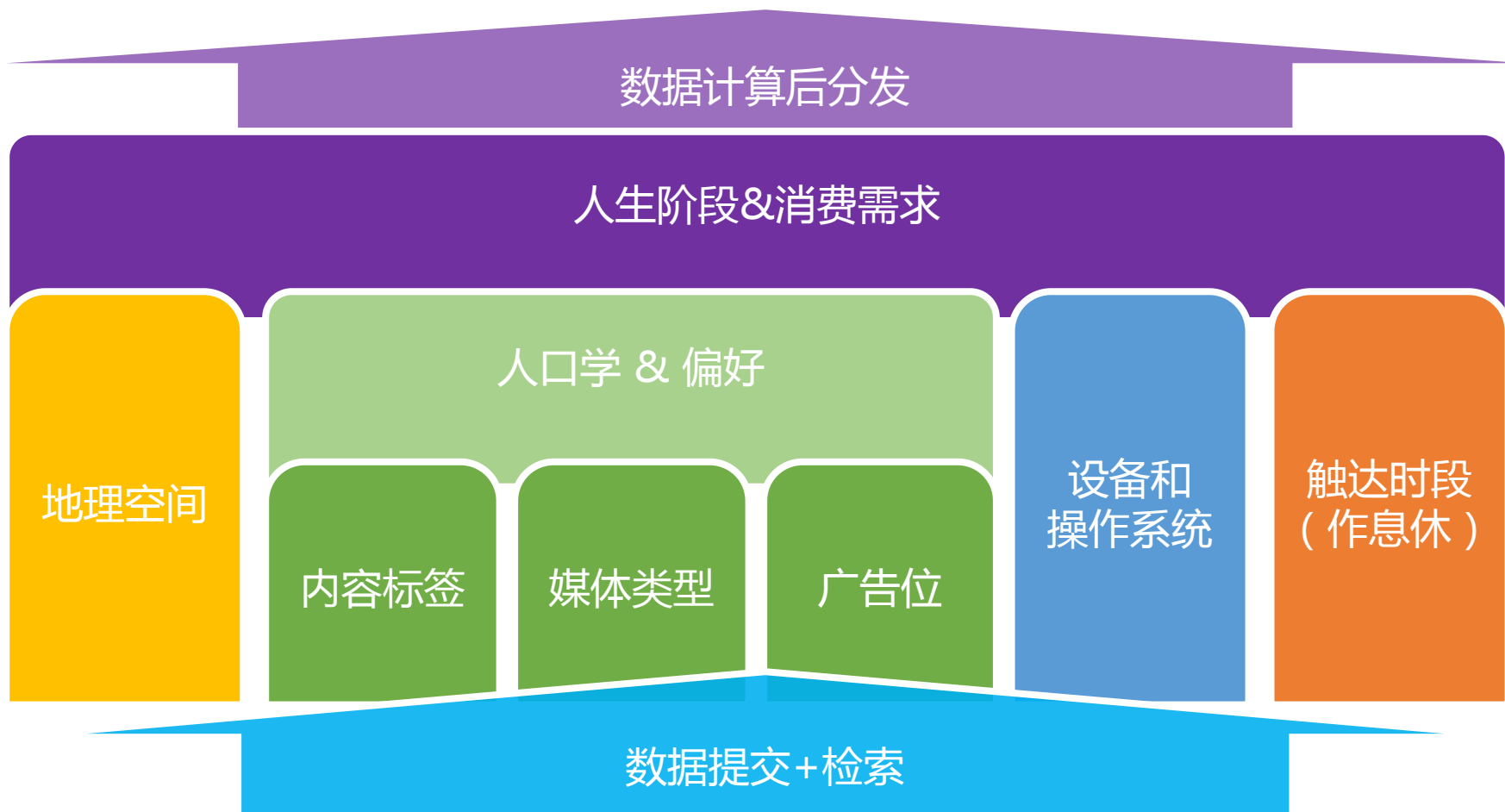


回归分析

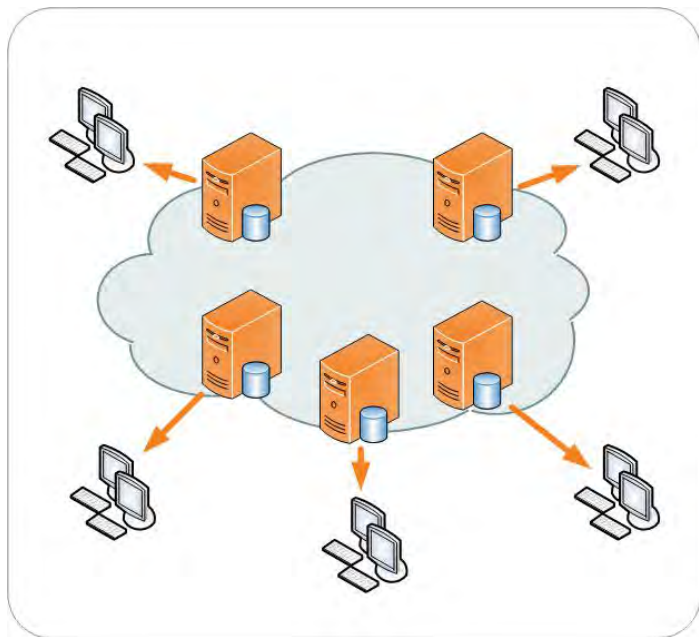


时间序列分析

以千人千面的受众标签体系为例



多机房乃至混合云带来的优势和挑战



综合成本更优

- 可以选择性价比更高的带宽资源
- 可以利用云作为弹性错峰

服务体验更好

- 容灾冗余提供服务高可用
- 网络延迟降低，请求响应更快

架构设计的挑战

- 全节点数据更新和同步机制
- 跨节点资源调度机制（负载平衡&应急切换）

- 分布式缓存场景和需求

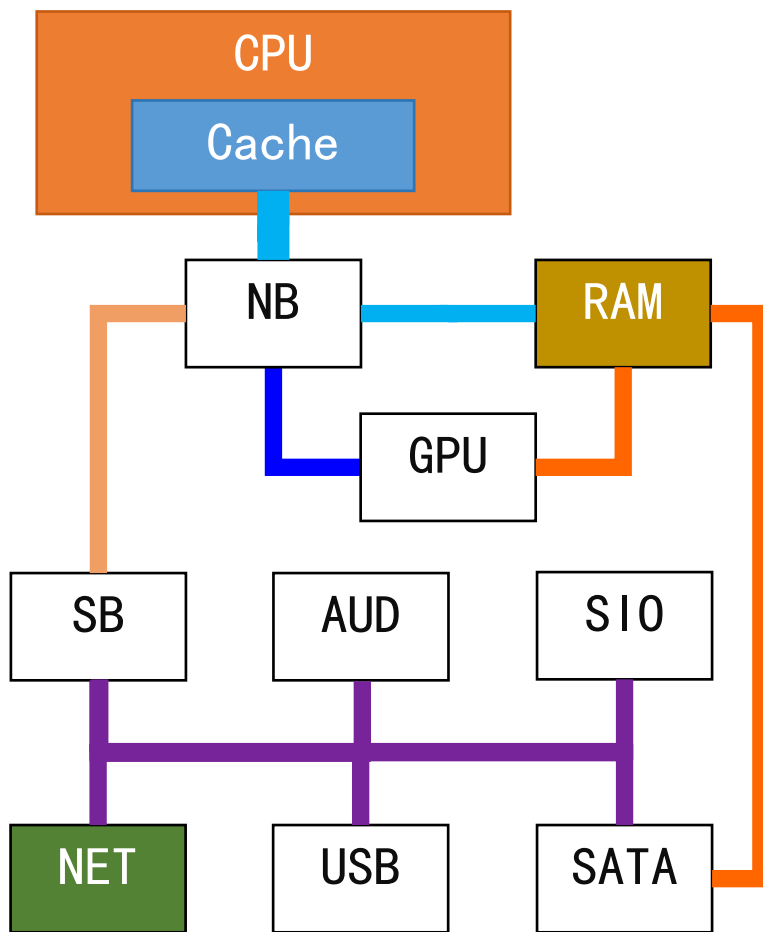
- 缓存架构演进之路

- 架构设计瓶颈剖析

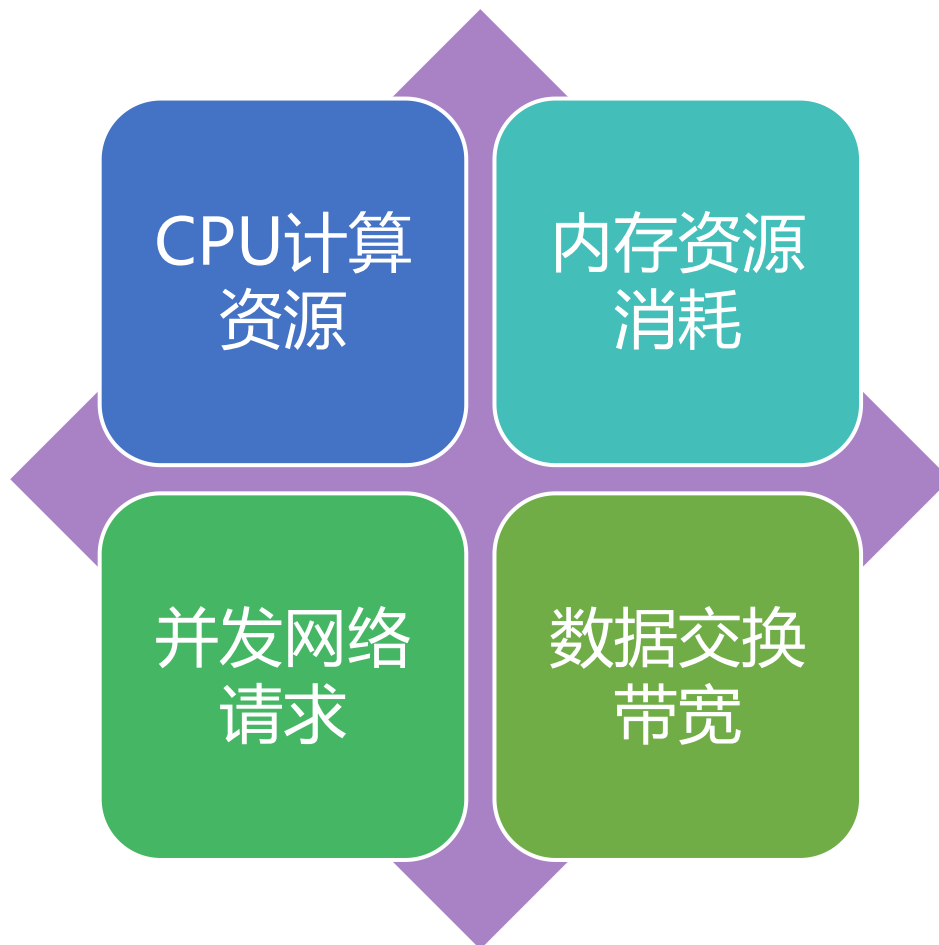
- 软硬兼施 对症下药

瓶颈的核心成因：流速 流量 资源的不匹配

- 瓶颈一：物理瓶颈——突然收窄的通道
 - 硬盘-内存数据交换
 - 内存-CPU数据交换
 - 分布式集群内的数据交换带宽
- 瓶颈二：业务瓶颈——突然增加的流量
 - 网络请求并发突增（网卡吞吐）
 - 复杂计算占比突增（CPU）
 - 缓存数据暴涨（内存）



缓存系统架构的四类瓶颈总结



四类瓶颈的通常解决方案

CPU计算资源

- 算法性能调优
- 添加 CPU 乃至 云节点
- 用 GPU 或 ASIC 辅助

内存资源消耗

- 设计合理的回收机制
- 冷热数据分离优化命中率
- 优化的数据结构设计

网络并发

- 使用压缩传输
- 使用负载均衡 乃至 云节点
- 利用客户端缓存

数据交换

- 数据压缩存储
- 使用SSD作为二级缓存
- 使用万兆网卡

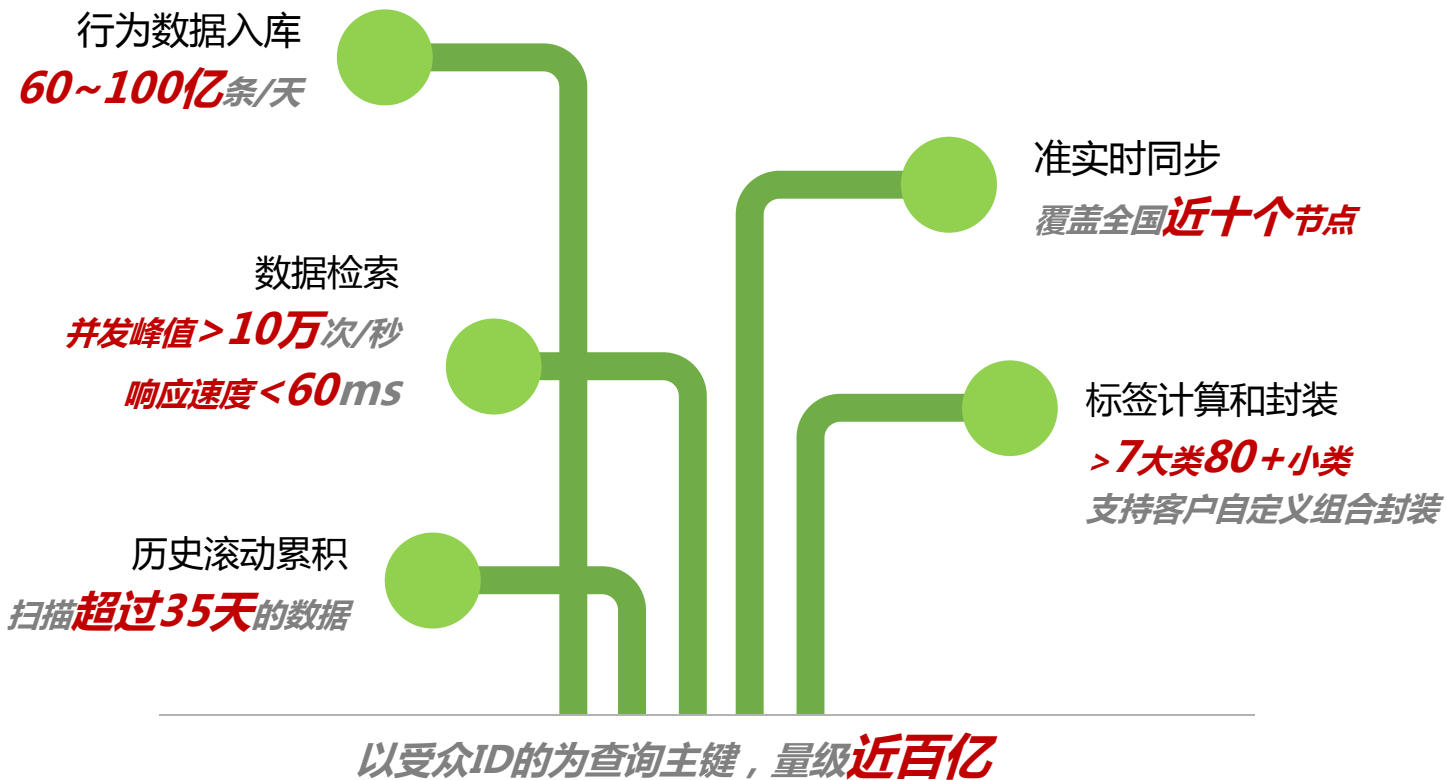
- 分布式缓存场景和需求

- 缓存架构演进之路

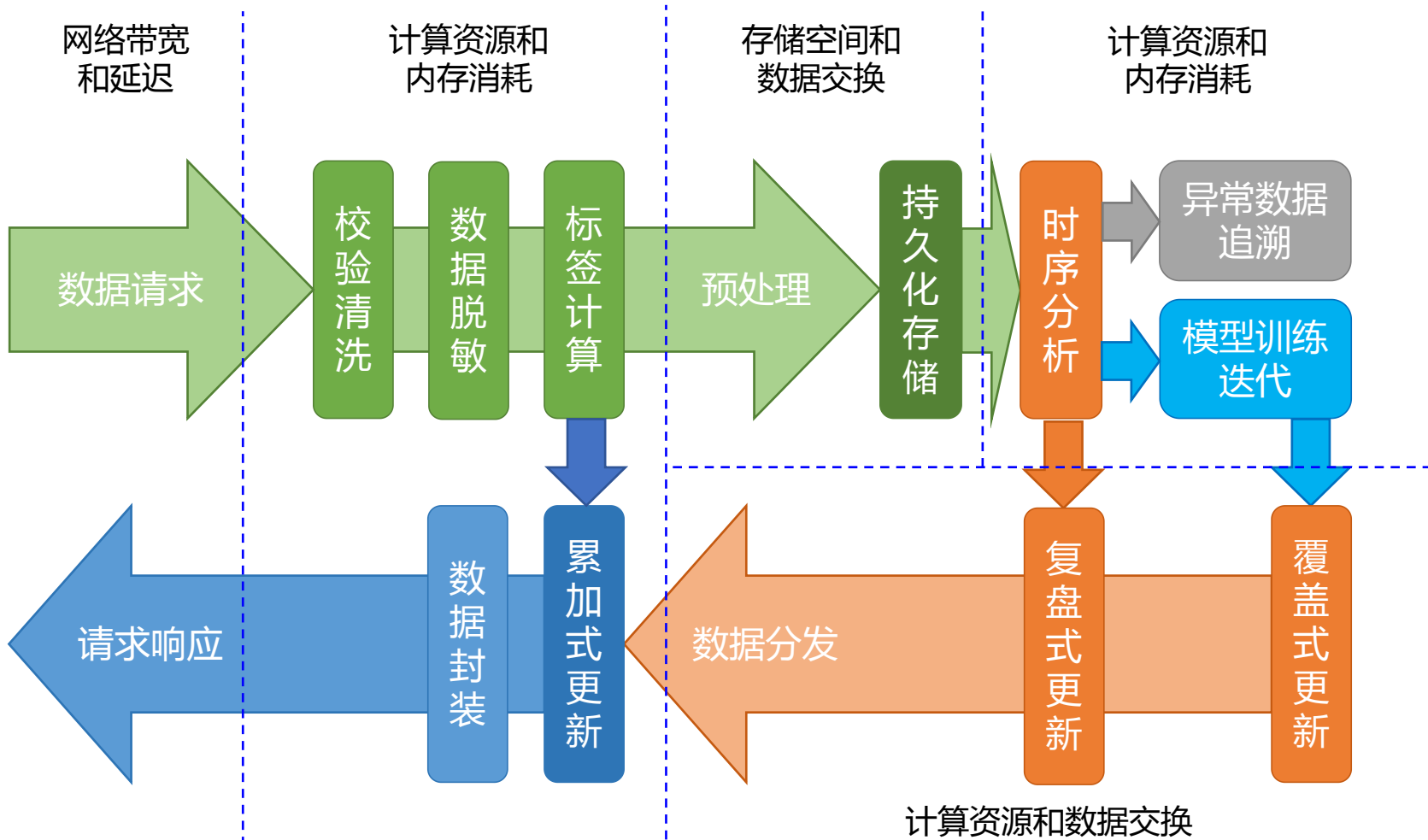
- 架构设计瓶颈剖析

- 软硬兼施 对症下药

严苛的业务场景需求

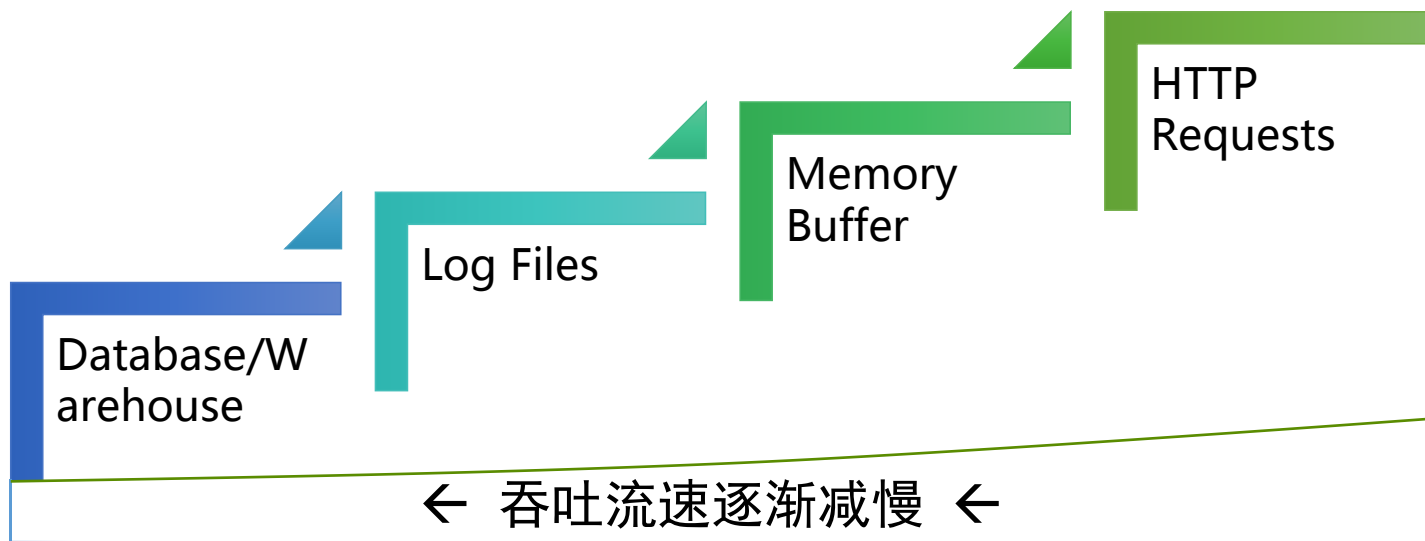


实际业务场景瓶颈分析



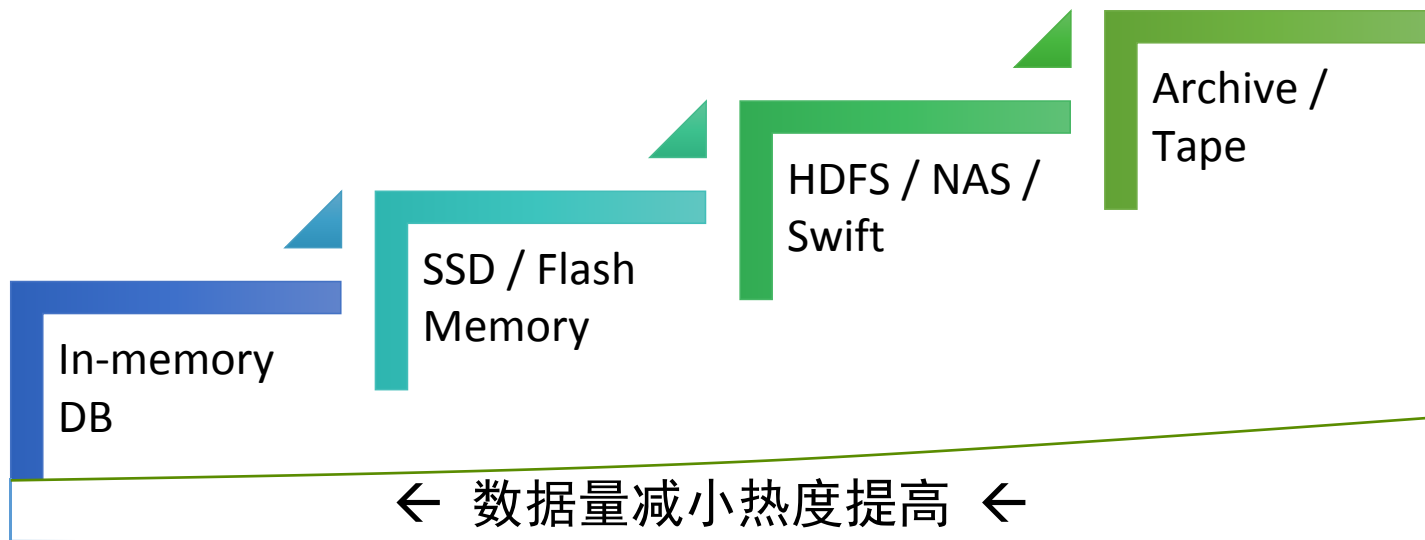
数据入库瓶颈解决对策 – 存储读写分离

- 单次请求一般有原子性和邻域性，因此实时全网同步要求不高
- 如果缓存数据依赖于长期累积，那么单次误差的波动会被平滑
- 单次请求提交的数据可以先缓存后批量延迟入库

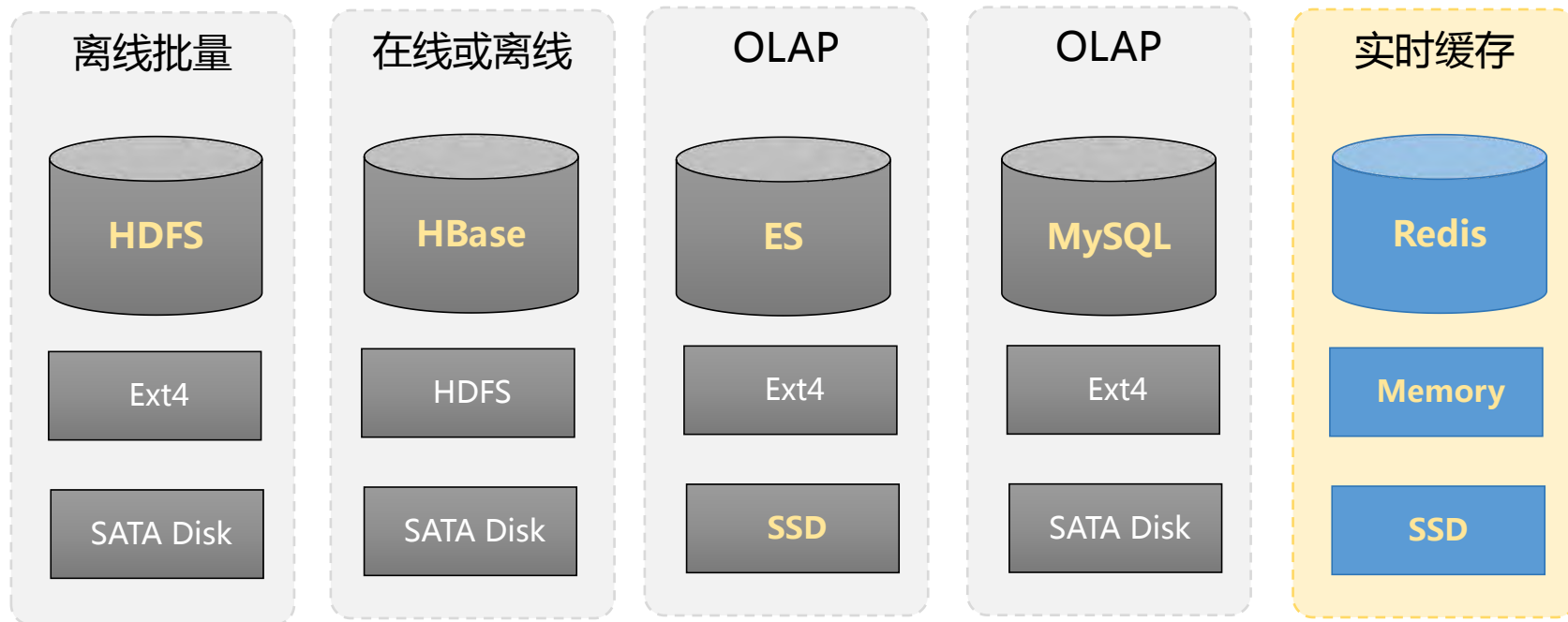


数据按热度分级 – 提升命中减少交换

- 热数据放内存，做实时计算和响应，确保命中率
- 温数据放在SSD和分布式文件系统，提升数据交换和模型迭代效率
- 冷数据高压缩率后存储，很少使用

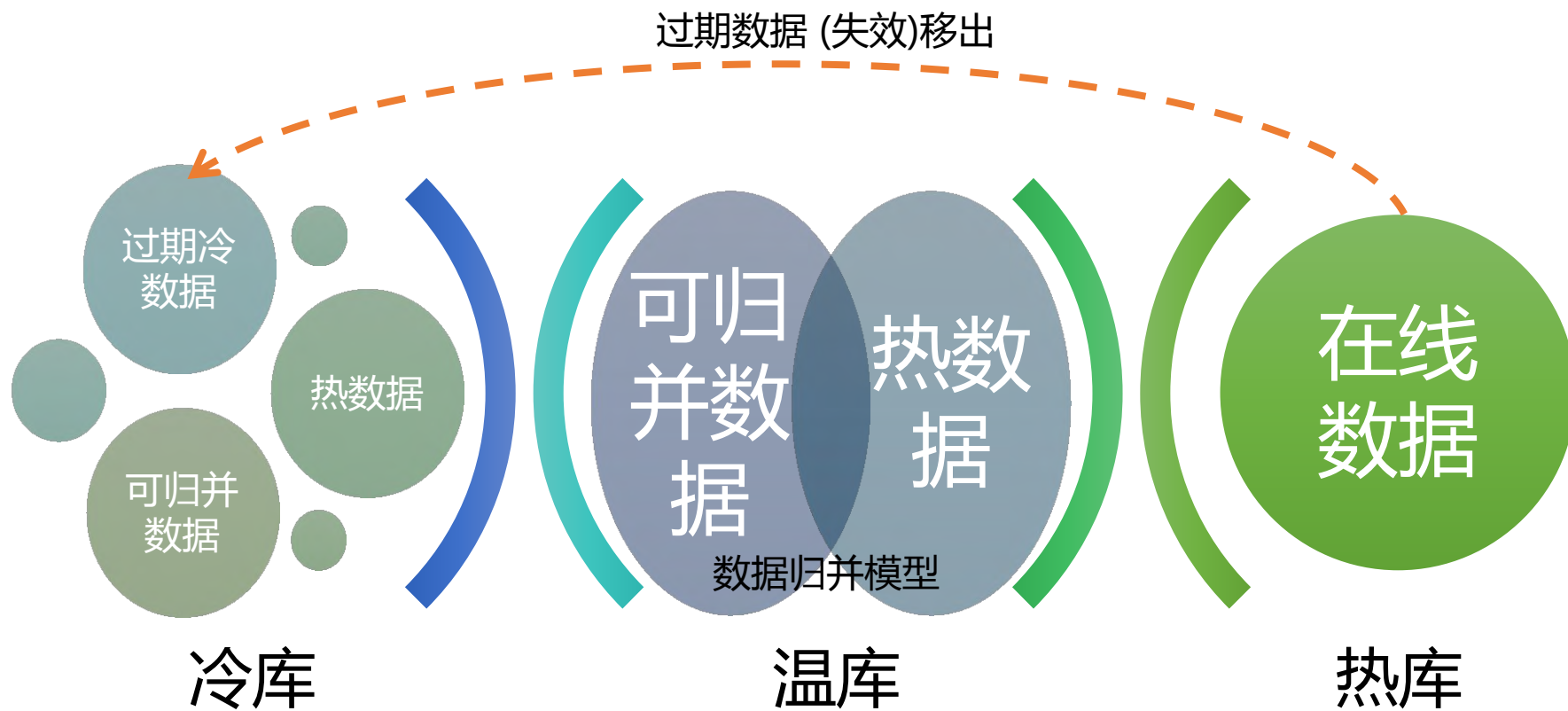


不同应用场景对应的热度分级



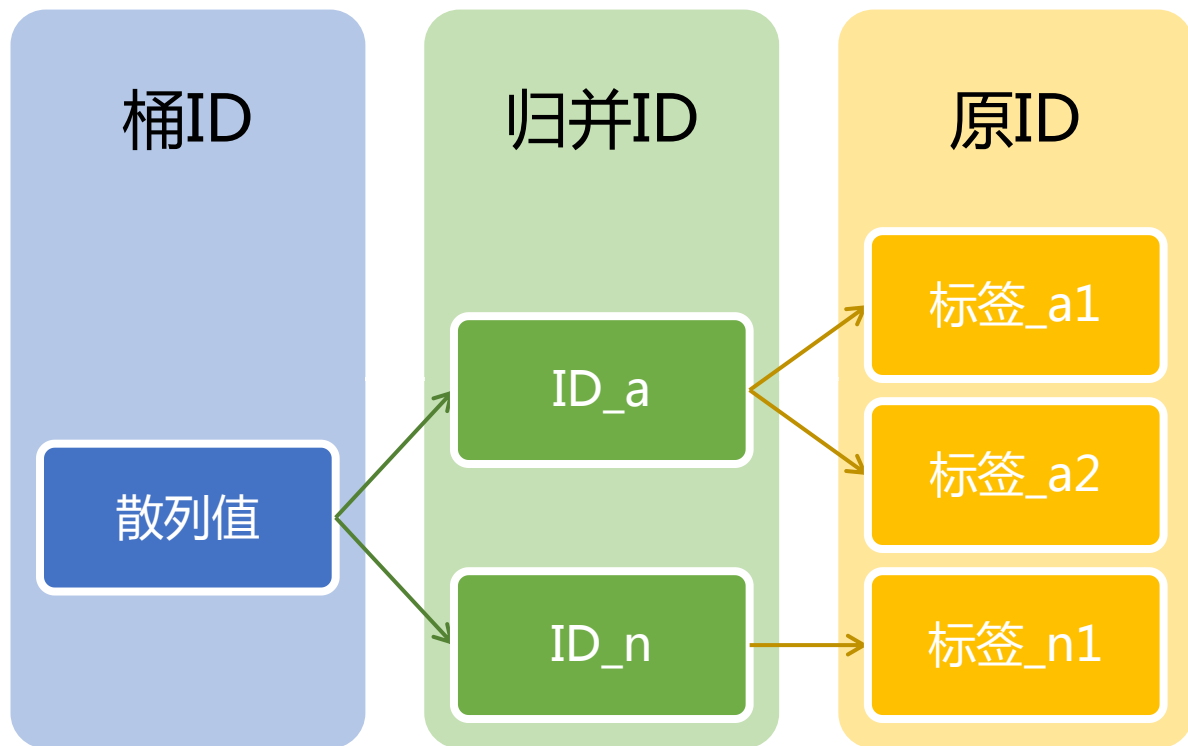
连接池 / 消息队列 **Memory+SSD**

合理设计空间回收 – 控制内存消耗



巧妙设计数据结构 – 控制内存消耗

- ID散列化，本质是一种编码同余归并（利用冲突率，合并同余项）
- 选择合理的散列编码方法，可以节约70~80%的内存



复习：四类瓶颈的通常解决方案

CPU计算资源

- 算法性能调优
- 添加 CPU 乃至 云节点
- 用 GPU 或 ASIC 辅助

内存资源消耗

- 设计合理的回收机制
- 冷热数据分离优化命中率
- 优化的数据结构设计

网络并发

- 使用压缩传输
- 使用负载均衡 乃至 云节点
- 利用客户端缓存

数据交换

- 数据压缩存储
- 使用SSD作为二级缓存
- 使用万兆网卡