

课时29

从容应对级 QPS 访问，Redis 还缺少什么

1. 功能扩展
2. 完全增量复制
3. 集群管理

功能扩展

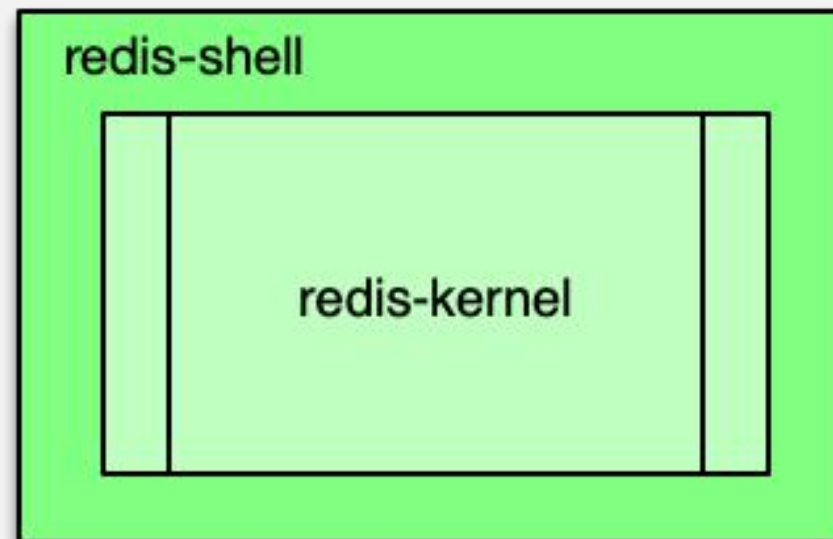
- 问题
 - 升级/bug修复需重启，耗时长，降低可用性
- 扩展
 - 热升级



功能扩展

- 热升级方案
 - 构建一个壳程序，保存全局的 redisServer 信息
 - 将 Redis 处理逻辑封装到动态连接库 So 文件
 - 第一次正常启动，从磁盘加载数据
 - 后续升级，重新加载 Redis 的 so 文件
 - 毫秒级完成版本升级

redis



redis-4.so

redis-5.so

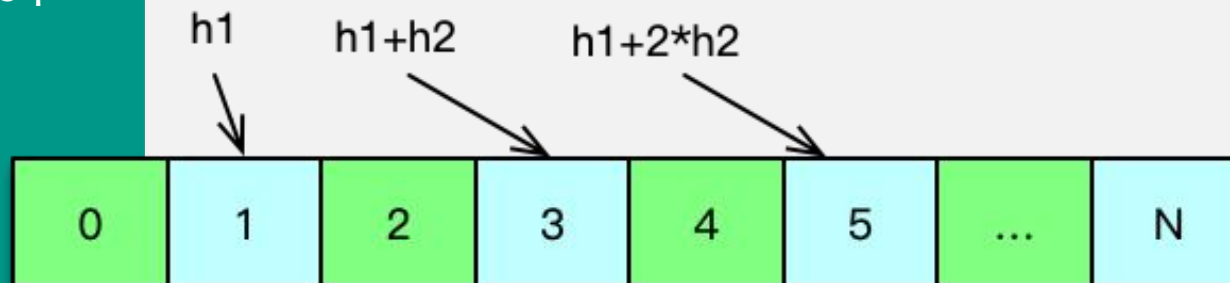
功能扩展

- 问题
 - 特殊业务场景，Redis 无法满足业务需要
- 定制化扩展
 - 扩展Redis新的数据结构
 - 扩展Redis的存储模型



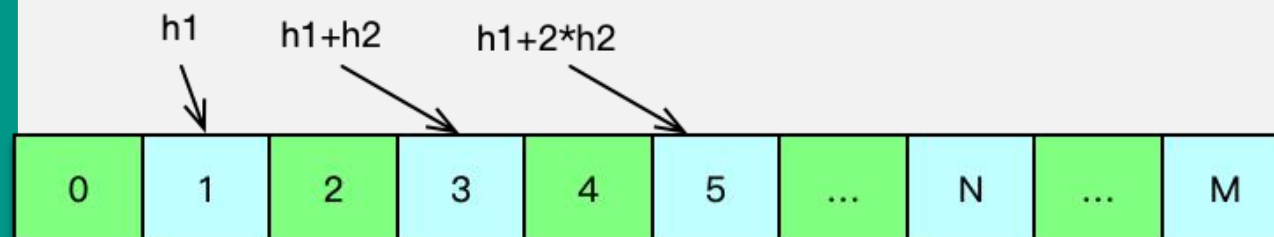
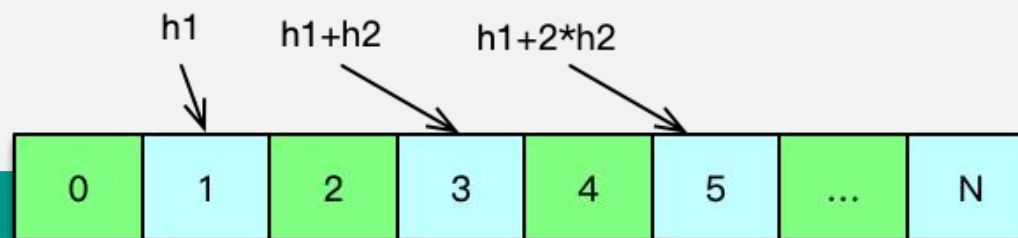
功能扩展

- 关注列表
 - Redis作为缓存使用，元素全部是long
 - Miss 重新加载，数万个元素导致redis卡顿
 - 内存效率不高
- 改造方案
 - 扩展longset数据结构
 - Longset 是一个long型一维开放数组



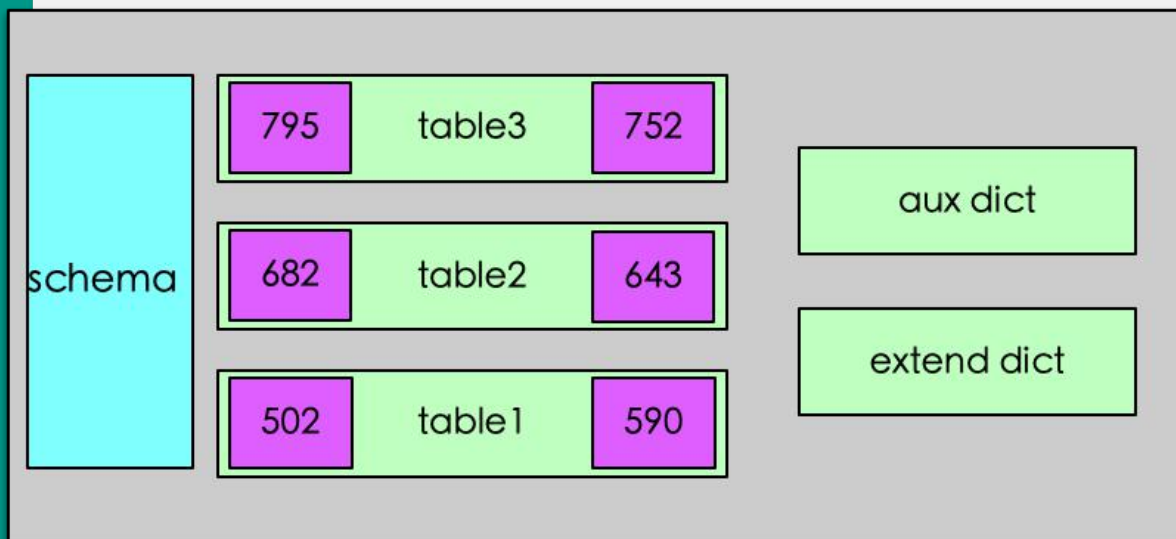
功能扩展

- 改造方案
 - client将关注uid列表，构建成long数组的二进制数据，然后lset回redis
 - Redis直接存储二进制数据作为value值
 - 采用double-hash寻址及解决hash冲突
 - 填充率超过阈值，redis返回异常，由client构建更大的long型数组，一次性写入



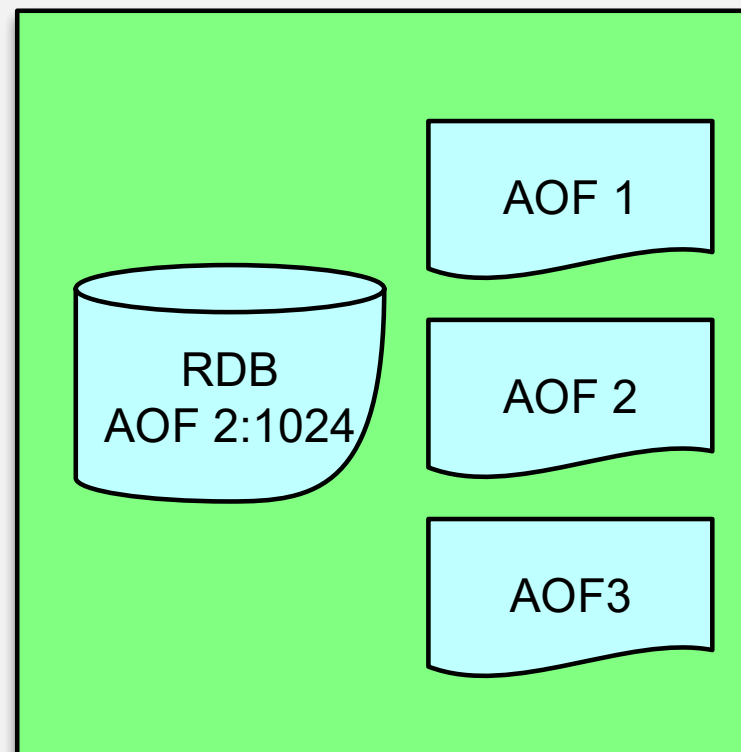
功能扩展

- 计数器
 - 一个key有多种计数列
 - 海量计数，key 递增
 - 单次查询一个key的任意多列
- 改造方案
 - 改造redisDb，采用cdb分段存储计数
 - 采用schema，指定计数的数量和类型
 - 冷热分离，热数据存内存，冷数据存磁盘



功能扩展

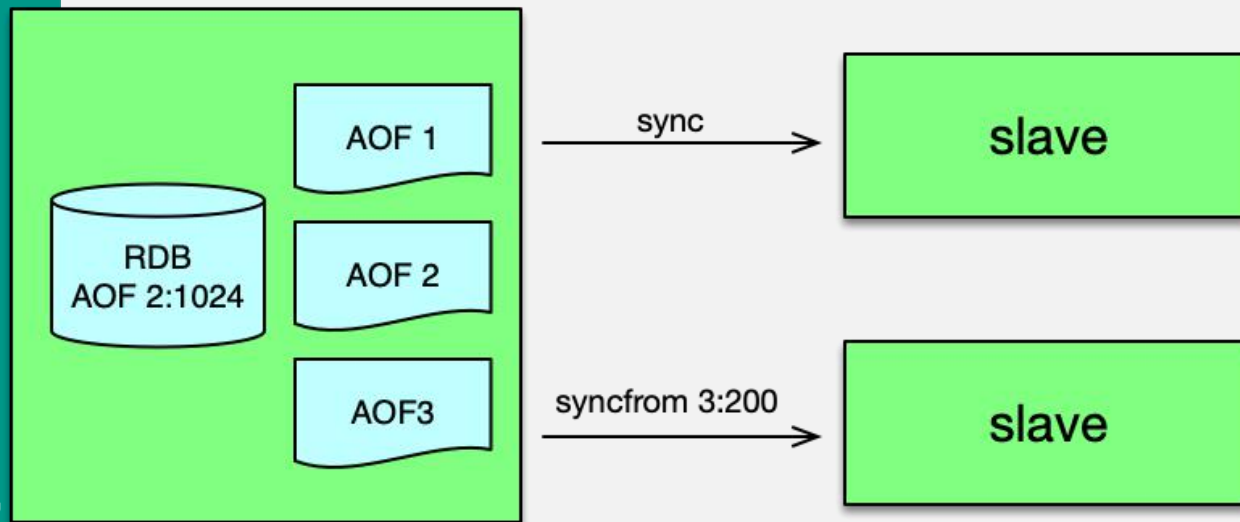
- 问题
 - 主从复制受限复制缓冲，容易导致全量复制
- 解决
 - 结合RDB+AOF策略，构建完全增量复制方案



功能扩展

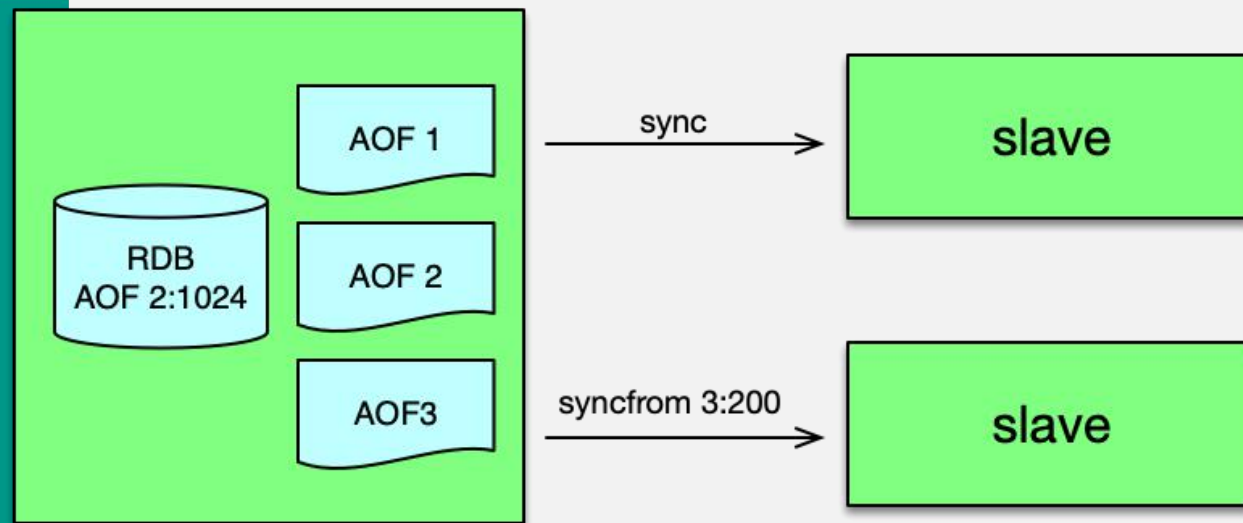
完全增量方案

- Aof 按后缀id递增，滚动存储最新写指令
- Rdb文件记录当前内存数据，及构建时刻AOF文件的id 及 位置
- Master 通过独立的复制线程向slave同步数据
- 第一次复制为全量复制，之后不受内存及时间限制，只要aof文件没被删除，全部为增量复制
- 第一次复制，复制线程发送rdb给slave后，再将rdb记录的Aof文件位置之后的所有数据，也发送给slave，完成全量同步

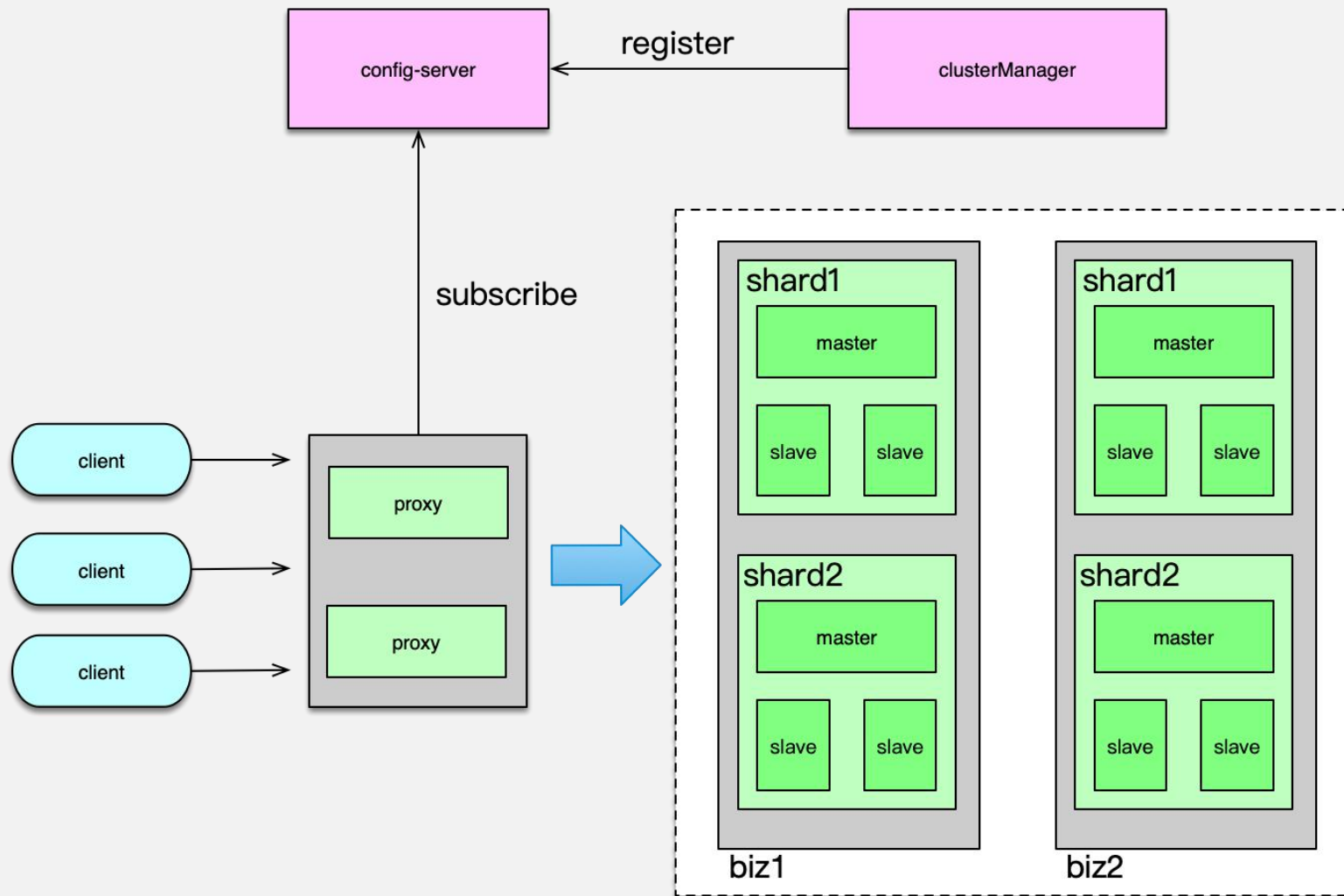


功能扩展

- 完全增量方案
 - 后续同步, slave 首先传递之前复制的Aof文件id及位置
 - Master 的复制线程根据slave的aof文件位置, 读取之后内容, 发送给 slave
 - Master同步数据时, 对每个slave启用一个独立线程, 复制过程不影响请求访问



Redis 功能扩展



功能扩展

集群管理

- Redis 剥离集群逻辑，不再维护 slot 相关信息
- 构建 clusterManager，负责数据维护、迁移、状态管理
- Redis 集群访问由 proxy 或 smart client 进行
- 配置属性从配置中心获取及协调
- clusterManager 向配置中心注册业务资源部署及状态信息
- Proxy 及 Client 从配置中心获取及订阅配置信息

Next: 课时30 《面对海量数据，为什么无法设计出完美的分布式缓存体系》