

GOME

国美高性能缓存——Gcache

国美在线架构部
王復興

The GOME logo is written in a stylized, orange, handwritten-style font.

- 性能
- 总体架构
- 特性
- 数据迁移
- 未来规划

The logo for GOME, featuring the word "GOME" in a stylized, orange, handwritten-style font.

- 性能
- 总体架构
- 特性
- 数据迁移
- 未来规划



性能——集群2

(统计时间截止2014-11-12)

QPS峰值157.5w/s

日均访问量100亿+
11.11达578亿

超过100ms延迟占比
1/100w

包含57项业务

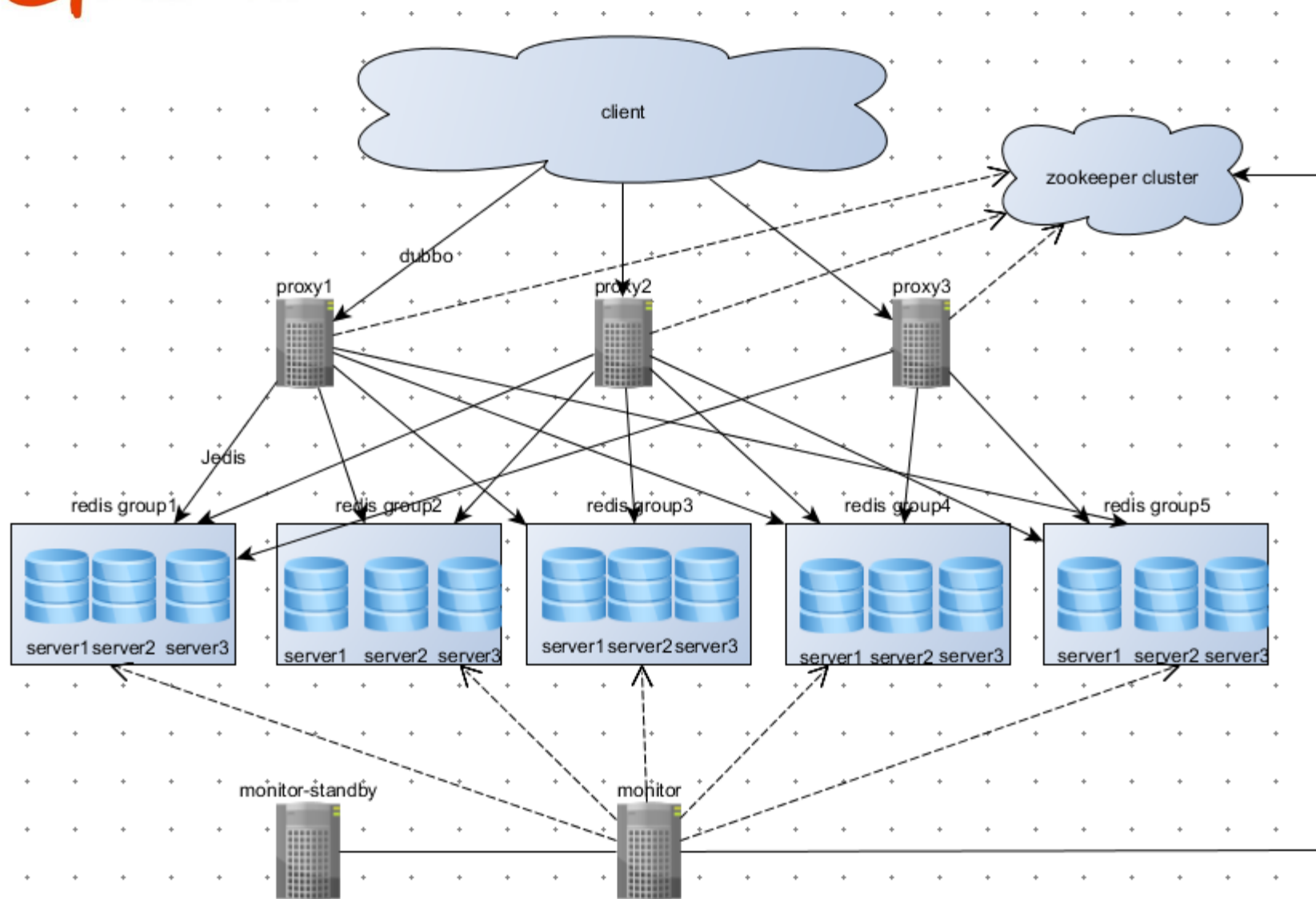
8台物理机, 10000Mbps
每台: 24core 128g 20redis

The GOME logo is written in a stylized, orange, handwritten-style font.

- 性能
- 总体架构
- 特性
- 数据迁移
- 未来规划

GOME

总体架构



角色:

Client: gcache客户端

Proxy: 访问redis的代理,负责分发请求

Redis Group: redis的副本集(M-M,M-S)

Monitor: redis的监控

Zookeeper: 存储路由信息

高可用:

Proxy至少一个节点存活,可随意加减

Zookeeper可全部宕, proxy缓存了路由

Redis Group中至少存活一个节点

Monitor 双机热备

伸缩性:

Proxy可动态加机器,client通过zk感知

Group动态增加,扩展集群/新业务存储

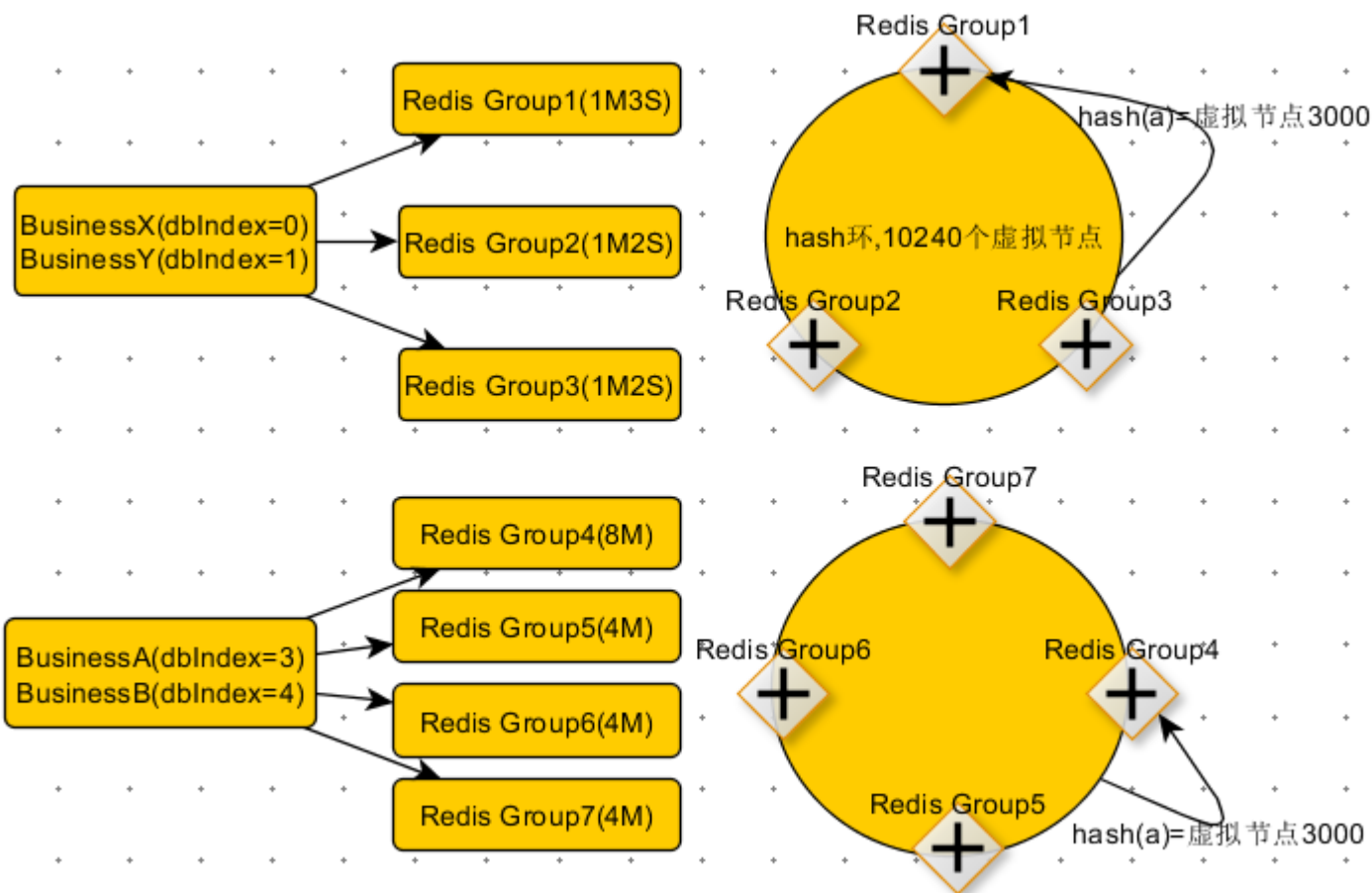
Group内redis动态增加,应对读多的操作

The GOME logo is written in a stylized, orange-red script font.

- 性能
- 总体架构
- 特性
- 数据迁移
- 未来规划

GOME

特性——业务隔离,存储分组



Business:

挂载Redis Group
不同业务,同一redis,用dbindex隔离
不同业务可分开存储在不同redis

Redis Group:

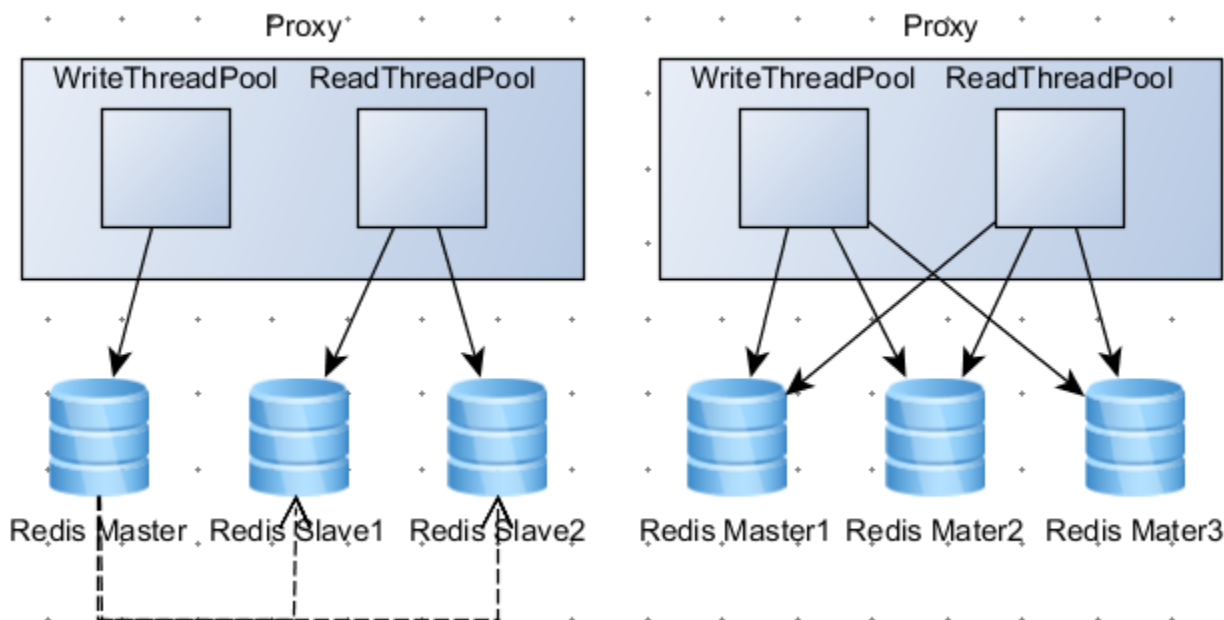
挂载redis副本集,各副本数据一样
支持M-M,M-S
部署结构与数量根据需求定

Hash环:

默认10240个虚拟节点
Redis Group均匀放在虚拟节点上

GOME

特性——读写分离，主从，多主



读写分离:

Proxy上的读写线程池分开
Redis master写,slave读

多主:

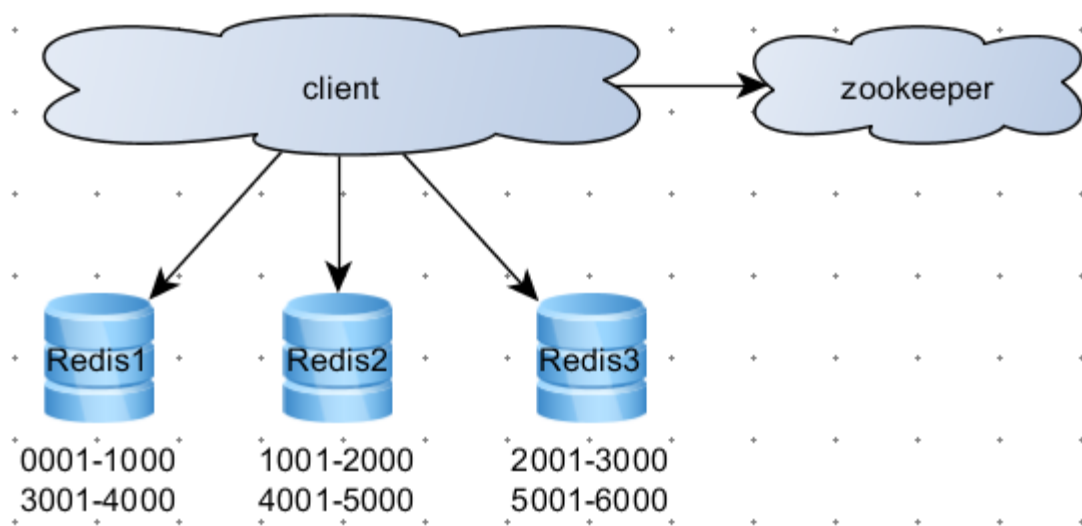
Proxy负责多写
多写并发进行,响应时间取决于最长路径
一致性差,可用性强
适用于无并发写入

主从:

Proxy负责写入到主,由redis同步到从
Master宕,对应分片短时间3s不可写
适用于用户并发操作



特性——高性能Id生成器



Zookeeper: 登记了可用的redis列表,
步长,取值范围,异步取id的阈值等

Redis:

每次生成的id按redis的个数错开
任意redis挂掉, client到其它redis上取

Client:

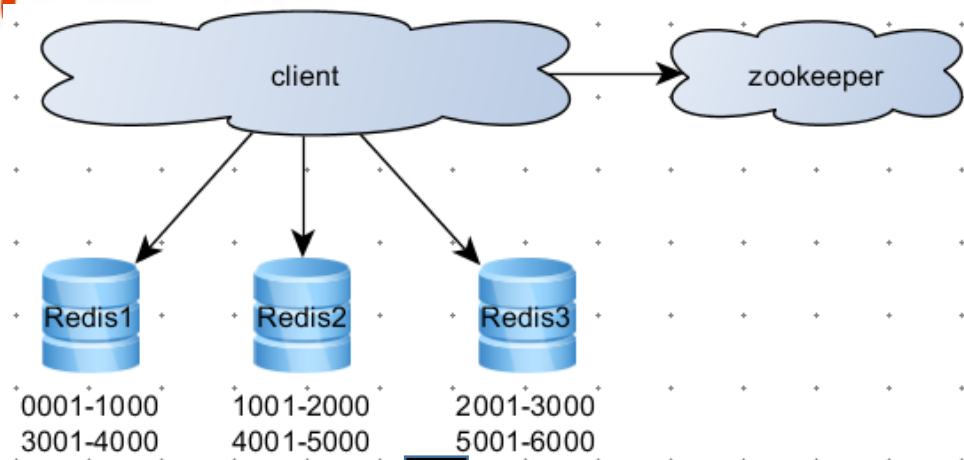
随机选择redis每次默认取1000个id
当1000个id快用完时,异步取下一批id

Client异步线程:

当并发量过大时,每次自动取 $n \times 1000$
当并发量变小时,恢复到每次取1000

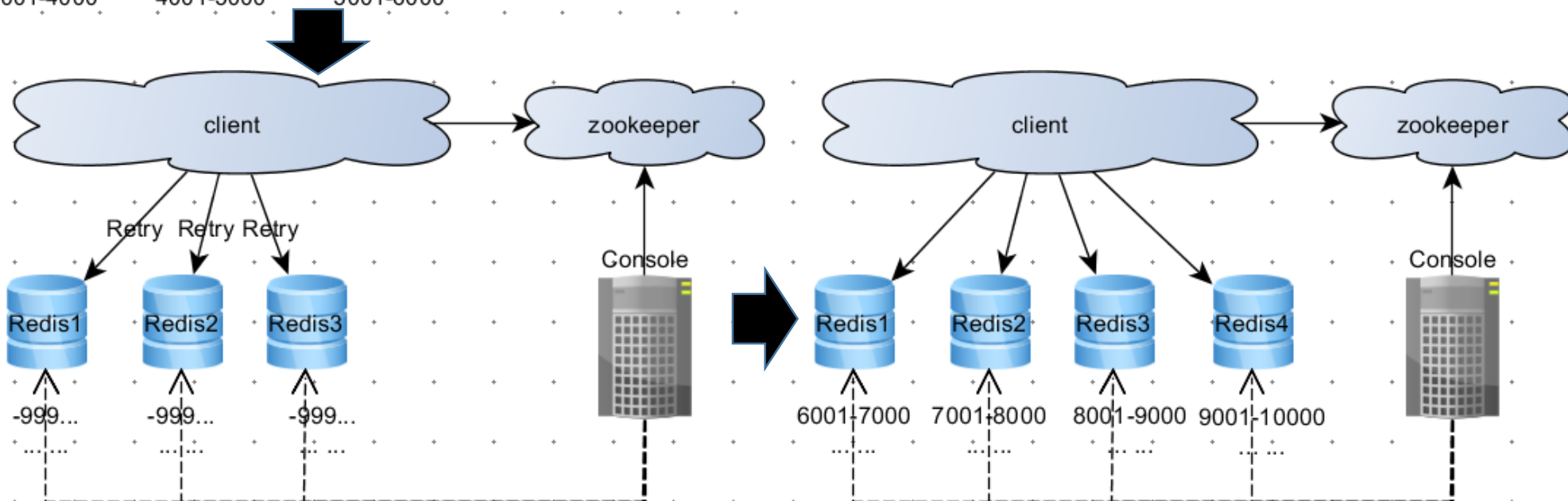
GOME

特性——Id生成器,如何扩容?



步骤:

1. 取出所有redis中最大的id,并设置成-999...
2. Client取到-999..., 每50ms重试
3. Console根据1的最大id,重新平衡redis Id
4. 客户端重试发现取得正值, 扩容完成





特性——业务升降级，限速，隔离

业务延迟过大
自动或手动调低并发

关键业务和非关键业务
存储隔离

各个业务根据特点指定最高的并发量

一键下掉业务对应的路由

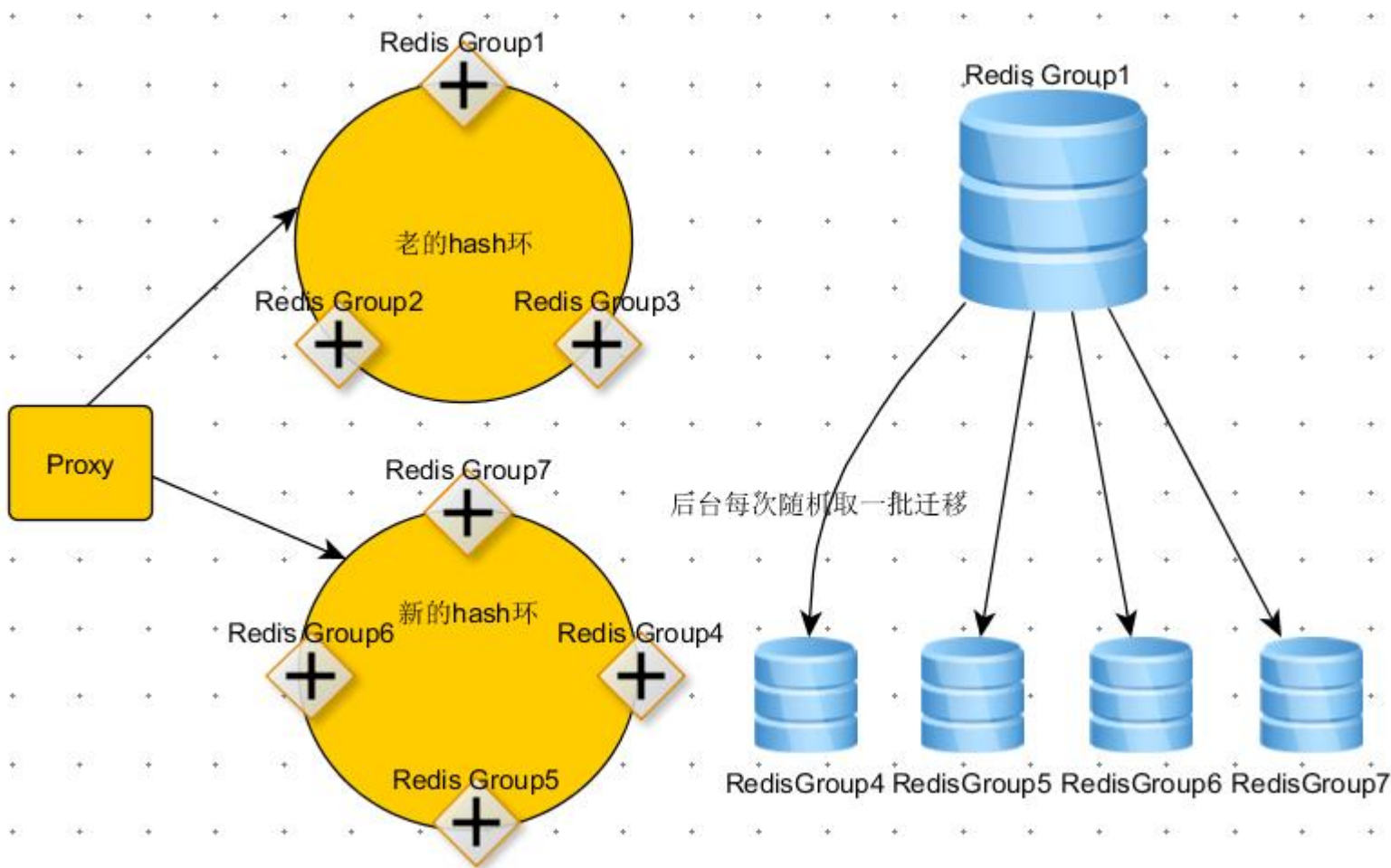
访问量/延迟太大的业务直接报警

The GOME logo is written in a stylized, orange, handwritten-style font.

- 性能
- 总体架构
- 特性
- 数据迁移
- 未来规划

GOME

数据迁移——实时、高并发



迁移时(proxy):

1. Proxy接收到读写key的操作
2. 按新的hash环请求key是否存在
3. 如果存在则直接操作
4. 如果不存在则触发迁移再操作

迁移时(后台):

- 每次迁移一个真实节点
- 每次随机取出一批keyValue
- 迁移完成后删除老数据

并发:

当多个proxy以及后台并发操作相同key时怎么办?

迁移采用Setnx(不存在则放入)
这样只会有一个操作成功,其余操作都按顺序操作



gcache VS redis cluster3.0

	gcache	redis cluster3.0
proxy	有proxy(升级/动态控制方便)	无proxy,redirect后客户端可记住slot
连接数	连接数少(客户端与proxy长连接)	连接数过多(客户端保持住和很多节点的连接,redis节点与节点间的通信连接)
namespace	有,方便做业务隔离	无(同一集群不同业务容易相互影响)
hashTag	无hashTag	有hashTag,但是容易导致数据局部过热
MM/MS支持	主从,多主可选择	主从
failover判断	monitor监控判断	gossip协议保证一致性

The GOME logo is written in a stylized, orange-red script font.

- 性能
- 总体架构
- 特性
- 数据迁移
- 未来规划



未来规划

性能提升

1. GoLang做proxy
2. 每个redis.dbIndex一个线程
3. 每个redis.dbIndex一个aof
4. redis单线程写/多线程读

新业务观察区

1. 防止新业务影响稳定的业务
2. 访问量/读写比例/延迟/增长趋势

智能

1. 根据存储量访问量等自动迁移
2. 性能差的业务提供优化建议

GOME

谢谢大家

王復兴(国美在线架构部)
Email: wangfuxing@yolo24.com