



Redis 大数据之路

@唐福林

<http://weibo.com/tangfl>

<http://blog.fulin.org>

关于我（们）

- 新浪微博
- 微博开放平台 open.weibo.com
- Feed 架构
- @timyang
- 招聘进行中

大纲

- 关于大数据
- Redis 简介
- 新浪微博中的Redis大数据之路
 - 通知
 - 好友关系
 - 计数器
- 经验教训



关于大数据

- 线下大数据 vs 线上大数据
- 数据挖掘 vs 在线服务
- 持久化的大数据 vs 内存中的大数据
- 结构化大数据 vs 半结构化大数据
- 个人定义：数据要求比单台机器能力高一个数量级



Intro to Redis

- REmote DIctionary Server
- NoSQL by @antirez by VMWare
- redis.io github.com/antirez/redis
- start at 2009, now latest stable 2.4.10
- Key - String, Hash, List, (Sorted)Set, Pub/Sub
- Great Performance

Intro to Redis

- Written in C , Single thread , event driven
- Fork : copy on write by OS
- Replication
- Persist
 - aof
 - rdb
- All Data In Memory

Redis大数据之路

- 新浪微博中的Redis大数据之路
 - 通知
 - 好友关系
 - 计数器



Redis大数据之路

- 新浪微博中的Redis大数据之路
 - 通知
 - 好友关系
 - 计数器



Redis大数据之通知

系统消息

我的通知

我的通知 (共14条)



系统管理员
3月20日 15:45

系统通知

您好，您的举报信息我们已经收到并处理，非常感谢您对我们工作的支持。



轻博客
3月16日 11:07

Qing用户使用习惯调查

亲爱的唐福林，欢迎您成为Qing社区中的一份子，为了让您在这里的社区生活更加便捷，我们进行了本次问卷调查，希望能与您共同创造一个划时代的Qing，问卷地址：[http://t.cn/SxSMii](#)，期待您的参与——新浪轻博客团队。



达人
3月3日 12:34

恭喜，你的微博头像已通过审核

恭喜，因你的达人积分超过200分，头像和昵称等资料真实并被审核通过，你的微博达人红五星特殊身份标识已经点亮，赶快向你的好友、粉丝展示你的微博达人特殊身份和达人积分吧，并邀请你的亲朋好友一起成为微博达人！微博达人积分和升级规则，请查看：<http://t.cn/SxSMii>

- 查看评论
- 查看粉丝
- 查看私信
- 查看@我
- 查看群内消息
- 查看相册消息
- 查看通知
- 查看邀请

Redis大数据之通知

- 业务场景
 - 用户通知（通知单个用户）
 - 公共通知（通知全站所有用户）
 - 新通知提醒
- 技术难点

Redis大数据之通知

- 存储 by redis
 - 索引 key - list
 - uid - notice id list
 - public notice id list
 - 内容 key - value
 - notice id - notice content

Redis大数据之通知

- 存储 by redis
 - 提醒 key - value
 - uid - since public notice id
 - uid - since notice id ?

Redis大数据之通知

- 技术实现
 - 发一条用户通知
 - 发一条公共通知
 - 提醒功能
 - 用户的通知列表页

Redis大数据之通知

- 问题
 - content 读取压力
 - 加 redis slave
 - 公共通知灰度发布

Redis大数据之通知

- 问题
 - content 存储压力
 - 业务允许删除老数据 - 滚动
 - 大部分通知内容相似 - 模板

Redis大数据之通知

- 问题
 - content 长尾
 - 全内存，TCO 太高
 - 太多 redis slave，维护成本太高
 - 迁移到 handler socket

Redis大数据之通知

- 经验教训
 - Redis 是非常好用的
 - 内存是很贵的
 - 如果 Redis 的性能都成了瓶颈，那么我们应该先重新审视一下业务设计了

Redis大数据之路

- 新浪微博中的Redis大数据之路
 - 通知
 - 好友关系
 - 计数器



Redis大数据之好友关系

The screenshot shows the 'Friends' page for user 'leeyanva' on Sina Weibo. The page is divided into several sections:

- Header:** Sina Weibo logo, navigation tabs (Home, Plaza, Weibo, Application, Game), search bar, and user profile (Tang Fulin).
- Left Sidebar:** Navigation menu for 'Followers/Fans' (关注/粉丝), including 'Follow' (关注), 'Mutual Follow' (互相关注), 'Silent Follow' (悄悄关注), 'Fans' (粉丝), 'Invite Friends' (邀请站外好友), 'Find People' (找人), and 'Guess You Like' (猜你喜欢).
- Main Content:**
 - Followed:** '我关注了275人' (I followed 275 people). Includes filters (All, Mutual, Unsorted, Family, Tech, Colleague, More), a search bar, and a '+ Create Group' button.
 - Group Management:** '该组共有1人' (This group has 1 person). Includes 'Edit Group', 'Delete Group', and 'Browse Group Weibo' options.
 - Add to Group:** A form to '将关注人添加到分组' (Add followed people to group) with an 'Add' button.
 - User Profile:** Profile for 'leeyanva' (Beijing Chaoyang District). Stats: 1073 Follows, 1198 Fans, 1099 Weibo posts. Bio: 'IT猎头顾问。更多职位信息查看: http://www.douban.com/people/leeyanva/, 执着的摩羯女'.
 - Recent Post:** A tweet about technical skills: '技术啊//@54chen: 这手法很牛B, 与信用卡开门是一个算法。//@乌云-漏洞报告平台: 黑客无处不在啊.....(今天 12:37)'. Includes a 'Chat' button and 'Follow via Sina Weibo' link.
 - Followers:** '我关注的人中: 张俊林say、log2n、JohnbinWang等11人 关注了她' (Among the people I follow: Zhang Junlin say, log2n, JohnbinWang, etc. 11 people followed her).
 - Common Interests:** '我们共同关注: hellodba、牛牛、indigo等34人' (We follow in common: hellodba, Niuniu, Indigo, etc. 34 people).
- Right Sidebar:**
 - People You May Be Interested In:** '可能感兴趣的人' (People you may be interested in). Includes 'Swap' button and list of users like 'Tang Senjin' (同事, 8 indirect follows) and 'yuyo' (8 common friends).
 - Popular User Recommendations:** '人气用户推荐' (Popular user recommendations). Includes 'Swap' button and user 'Yan Dan'.

Redis大数据之好友关系

- 业务场景
 - 用户关注列表
 - 互相关注
 - 关注备注
 - 关注分组
 - 用户粉丝列表
- 技术难点



Redis大数据之好友关系

- 业务场景
 - 我和TA的共同关注
 - 我关注的人也关注了TA
 - 特殊分组：“未分组”
 - 分组中的互相关注/互相关注中的分组
- 技术难点



Redis大数据之好友关系

- 技术实现
 - Mysql
 - fromuid, touid, addtime, remark
 - hash by fromuid
 - hash by touid
 - fromuid, groupid, touid

Redis大数据之好友关系

- 技术实现
 - Redis : hash VS sorted set
 - key: fromuid
 - field: touid
 - value: addtime or remark
 - group 分组功能?

Redis大数据之好友关系

- 技术实现
 - 加关注/取消关注
 - 添加备注
 - 修改分组
 - 判断是否已关注
 - 取列表
 - 集合运算

Redis大数据之好友关系

- 问题
 - 一致性
 - 多个 mysql 表
 - 多个 redis 实例

Redis大数据之好友关系

- 问题
 - hgetAll 慢
 - hash-max-zip-size
 - 增加 mc cache
 - 粉丝列表做限制



Redis大数据之好友关系

- 问题
 - TCO
 - 全内存: TB level
 - 快速增长: GBs/day
 - 冷热区分不明显
 - 性能 vs 功能
 - 暂时无解

Redis大数据之好友关系

- 经验教训
 - 性价比随数据量增长而下降
 - 大数据的一致性
 - Big data, Cost, Performance, BCP 只能三选二?

Redis大数据之好友关系

唐福林 

graph 数据量和访问量都无比巨大，现征集各路技术高手来挑战，抗住业务，或挖掘价值，都欢迎！

[@梁斌penny](#): 几个新浪微博数据，每天产生微博1亿条，流量过10亿，注册用户2.5亿，真实有效用户不到1亿，每用户关注200人，关注行为超过400亿。新浪微博用户关系graph巨大，每天产生大量价值急待整理和沉淀，蛮子文摘用人工，我们用人工智能。。

3月17日 19:46 来自小米手机

转发(40) | 评论(13)

+加标签

3月17日 20:14 来自iPhone客户端

转发(5) | 删除 | 收藏 | 评论(5)



Redis大数据之路

- 新浪微博中的Redis大数据之路
 - 通知
 - 好友关系
 - 计数器



Redis大数据之计数器

- 业务场景
 - 以用户为维度
 - 微博数，关注数，粉丝数，收藏数
 - 以微博为维度
 - 转发数，评论数
- 技术难点



Redis大数据之计数器

- 技术实现
 - mc + mysql (原始列表数据)
 - Redis : key - value
 - key : uid or mid
 - value : count

Redis大数据之计数器

- 问题
 - 一致性
 - count vs list

Redis大数据之计数器

- 问题
- TCO
 - redis cost 100+ bytes to store a count
 - hash : store multi counts in a hash
 - rediscounter : use array instead of hash table

Redis大数据之计数器

- 问题
 - 长尾（微博维度计数）
 - 10+ Billion counts
 - 1% hot : Only hot data in memory
 - mget \leq 10ms
 - 暂时无解

Redis大数据之计数器

- 经验教训
 - 线上大数据
 - 内存中的大数据
 - 长尾大数据



经验教训

- Redis 适用场景
 - 数据量不太大的存储
 - 数据量大的缓存

经验教训

唐福林 ★

数据规模100G以内，想做 cache 就做 cache，想做 storage 就做 storage，但规模再往上，就只能做 cache 了 // @jackbillow: 我们更多的是考虑做cache，合适的场景做storage。redis与memcached相比更多的优势在于丰富数据类型带来的快速开发和其作为cache的安全性，而不在于性能。

@nosqlfan: 这两年Redis火得可以，Redis也常常被当作Memcached的挑战者被提到桌面上来。关于Redis与Memcached的比较更是比比皆是。然而，#Redis#真的在功能、性能以及内存使用效率上都超越了#Memcached#吗？看看Redis作者怎么说 <http://t.cn/zOfdbDm>



3月12日 22:01 来自新浪微博

转发(147) | 评论(24)

+加标签

3月15日 10:15 来自新浪微博

转发(13) | 删除 | 收藏 | 评论(2)



火龙果•整理
uml.org.cn

经验教训

- 数据冷热区分
 - 小而热的数据 - 内存
 - 大而冷的数据 - 磁盘
 - 大而热的数据 - SSD?



经验教训

- 大数据的一致性
 - 业务层很难保证
 - 中间层？ 存储层？
 - 最终一致性？ session 一致性？
 - 架构设计层面需要考虑更多



Thanks

PS. We are hiring !
contact me via @唐福林



Q & A

@唐福林

<http://weibo.com/tangfl>

<http://blog.fulin.org>



火龙果•整理
uml.org.cn