

微博cache设计谈

新浪微博 @TimYang

Agenda

- Feed架构简介
- 微博Cache设计

微博技术的核心

- 数据的分发、聚合及展现
- 每条微博, 在技术上也称为status或feed
- 如


唐骏的微博 <http://t.sina.com.cn/tangjun>

第一, 今后我的名片上会加印一个博士在名字后。第二, 法律会让现在和今后那些捏造事实污陷他人者付出代价。第三, 我还是我, 什么都没改变, 一个我行我素明天开始你行我素的唐骏! ...最近有点烦有点烦, 好在明天就不烦了...!

7月6日 20:44 来自短信

转发(6010) | 收藏 | 评论(40894)

V 新浪认证

 唐骏
其他

15 750819 523
关注 粉丝 微博

Feed架构

- 微博两种feed设计模式
 - Push(推)
 - Pull(拉)
 - 复合型

Push

- 将feed比喻成邮件
 - Inbox: 收到的微博
 - Outbox: 已发表微博
- 发表：存到所有粉丝inbox(重)
- 查看：直接访问Inbox(轻)

Push

- 优点：实现简单， 首选
- 缺点：分发量



The screenshot shows a Weibo profile for the user 姚晨 (Yao Chen). The profile includes a cartoon avatar of a yellow character with a clock face, the text "悠哉中", and a green "+ 加关注" button. The user's location is listed as "北京, 朝阳区" (Beijing, Chaoyang District) and their bio as "一颗很逊的卤蛋。" (A very mediocre卤蛋). The profile also shows a "推荐给朋友" (Recommend to friends) button. On the right side, the user's statistics are displayed: 510 关注 (Followers), 2133375 粉丝 (Fans), and 1934 微博 (Weibos). The "粉丝" (Fans) count is circled in red. Below the statistics, there is a section for "个人资料" (Personal Information) which states "还没有填写, 请稍候。" (Not yet filled in, please wait).

姚晨 

<http://t.sina.com.cn/yaochen>

 北京, 朝阳区

博客: <http://blog.sina.com.cn/yaochen>

一颗很逊的卤蛋。

[+ 加关注](#)

[推荐给朋友](#)

 新浪认证

510 关注 | 2133375 粉丝 | 1934 微博



个人资料
还没有填写, 请稍候。

Pull

- 发表：存到自己outbox(轻)
- 查看：取所有关注对象Inbox(重)
- Online computation

Pull

- 优点：节约存储
- 缺点：计算量大，峰值问题

共同的难题


- 峰值挑战
- 我们使用异步处理方式

微博小秘书的微博 <http://t.sina.com.cn/sinat>

南非#世界杯#终于迎来了高潮中的高潮！德国与英格兰奉献了一个精彩的半场，流畅的进攻、历史性的误判、永不妥协的精神，无论如何这都将成为历史的经典！同时，每秒世界杯微博峰值达到2500条，再创围脖新高！围脖球迷朋友们，让我们为精彩欢呼吧！加油，德国！加油，英格兰！

6月27日 23:02 来自新浪微博

转发(248) | 收藏 | 评论(144)

 新浪认证

微博小秘书
北京 海淀区

1979	4247588	1058
关注	粉丝	微博

Cache

memory is the new disk,
and disk is the new tape.

for "real-time" web applications,
and systems that require massive scalability

- Jim Gray

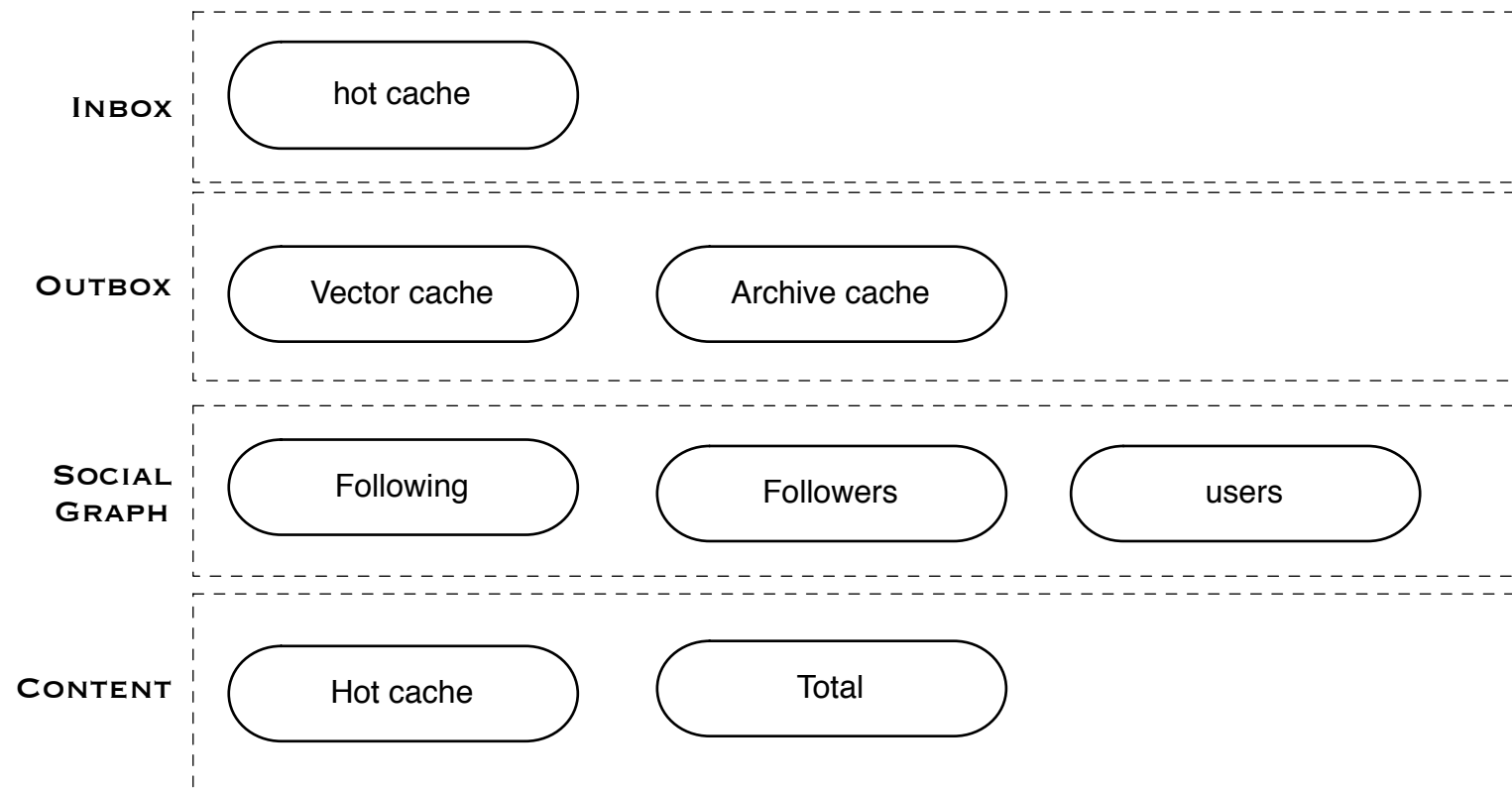


Cache设计

决定了一个微博系统的优劣

微博cache架构

WEIBO CACHE ARCH



Inbox

- 微博中“我的首页”
- cache 内容为 id list
- 只为在线用户设置(hot cache)
- 添加新元素需要先get再set
- 可用redis list/set

Outbox

- 我发出的微博
- 最新的id list(e.g. 100) 用于聚合
- 历史id LRU

Social Graph cache

- Following ids
- Followers加载开销比较大
 - 上百万粉丝
 - 越大的集合越容易变更
 - 改变则需delete all

Content cache

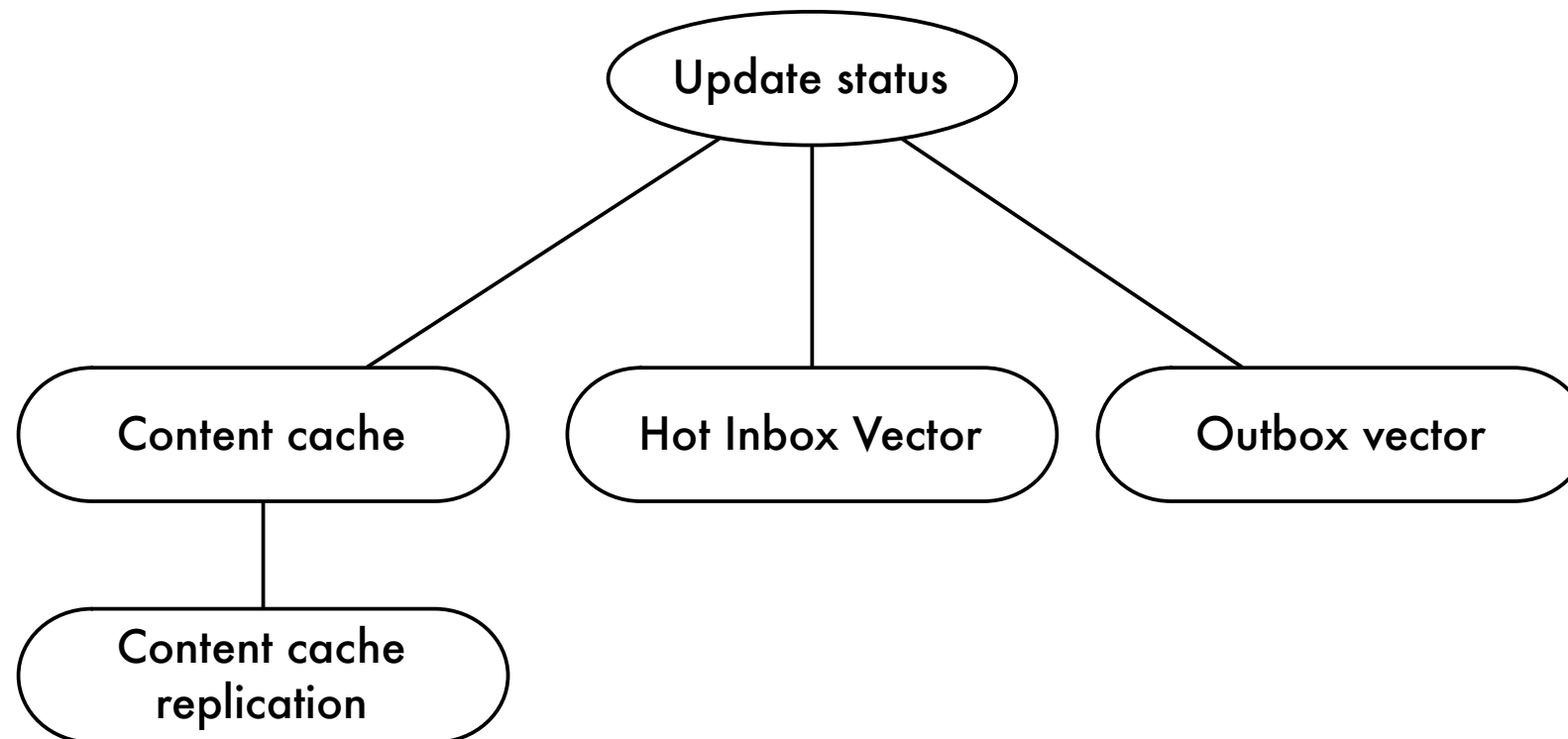
- 微博内容体cache
- 热内容
 - 多份, 防止单点访问瓶颈
- 最终格式预生成
 - Open API需要返回xml, json...格式

Cache流程

- 发表
- 首页展现

发表流程

UPDATE WORKFLOW

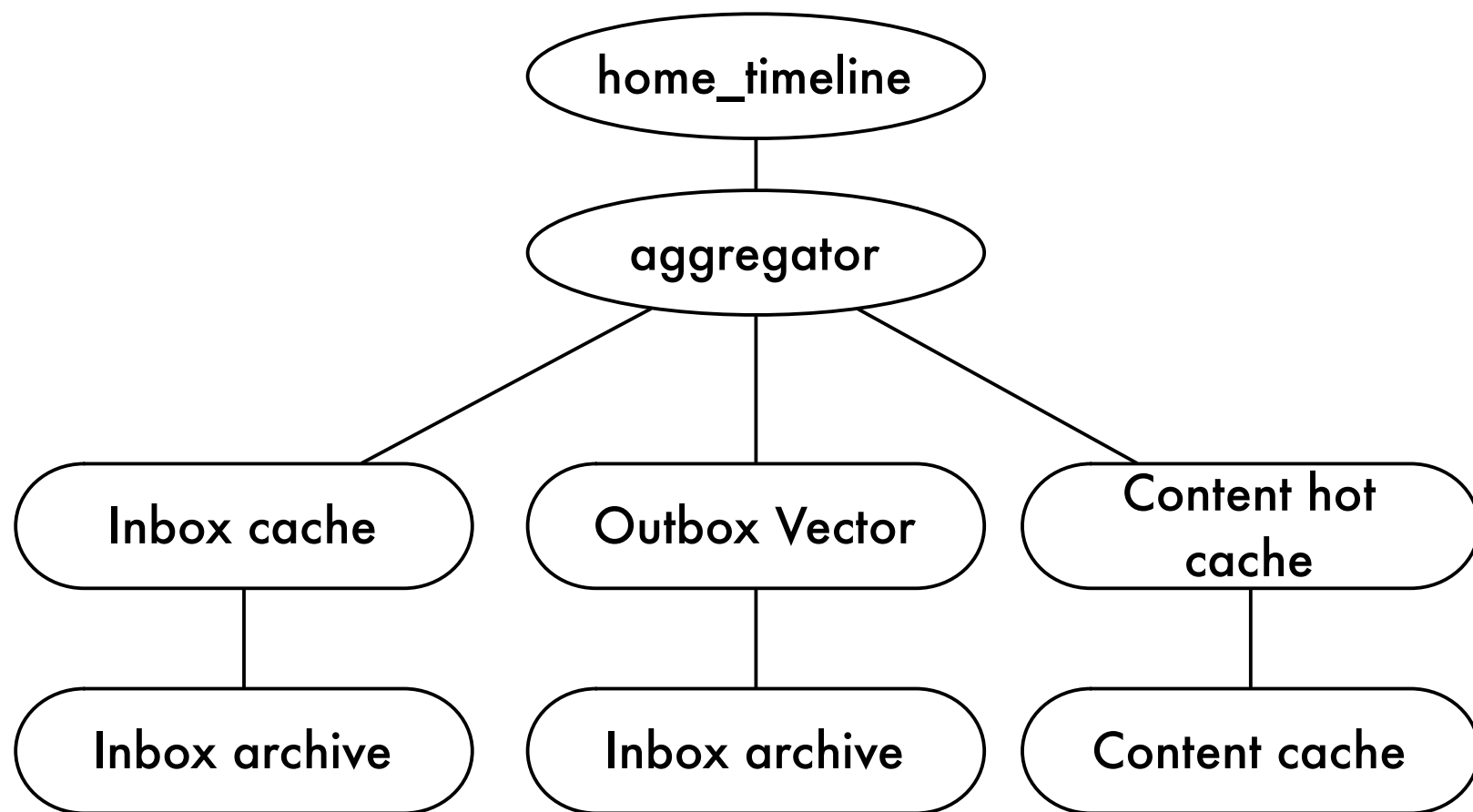


发表流程

- 修改outbox hot vector
- 加载followers list
- 修改inbox列表

首页feed

HOME TIMELINE WORKFLOW



获取首页feed流程

- 检查inbox cache是否可用
- 获取关注列表
- 聚合内容, 从 following 关系
- 根据id list返回最终feed聚合内容

cache经验谈

- 流量、带宽
- hot keys
- 规划
- mutex

流量

- 以打开首页时候获取Content cache为例
- multi get n 条feed($n = \text{items/页}$, e.g. 50)
- cache 大小 = $n * (\text{feed长度} + \text{扩展字段})$, e.g. 2k)
- 并发请求, 如 1,000次/秒

带宽

- 1,000并发, 需要800Mbps带宽
- 1万并发, 需要8Gbps 内网流量

带宽

- 在IG内网, 只能压力到 300~400Mbps
- 需要优化
 - 将热门数据加载到local cache
 - 压缩
 - 复制

hot keys

- content cache of 姚晨
- create local cache
 1. get user_yaochen_local
 2. get user_yaochen
 1. set user_yaochen_local:value
 3. 删除时需要delete all

cache 规划

- 将不同业务、不同长度的key存储到不同的memcache池
 - 不同的业务有不同的生命周期
 - LRU cache, 小量
 - Memory storage, 大部分

mutex

- 姚晨的content item未cache
 - 每线程需要从db加载数十条微博数据
 - 上千个并发线程同时穿透到db
- key_mutex 先添加成功, 再更新key
- mutex 失败则等待, 重试

未完待续

- 想更多关注微博平台技术
 - S2 技术沙龙, 每月一期
 - S2圆桌讨论, 一线架构师线下交流
 - 新浪微博平台开发者大会

S2技术沙龙

- 关注Web 2.0及Social技术
- What's S2
 - Social Platform &
 - Social Applications
- www.s2forum.org

新浪微博开发者大会

- 社会化平台需要的架构与存储
- Sina App Engine
- 数据挖掘
- 合作与商业模式
- 开发者与平台

Q&A

- email: iso1600@gmail.com
- 新浪微博：@TimYang