

高可用架构设计与实践

讲师：孙玄@58

【声明】

本视频和幻灯片为炼数成金网络课程的教学资料，所有资料只能在课程内使用，不得在课程以外范围散播，违者将可能被追究法律和经济责任。

课程详情访问炼数成金培训网站

<http://edu.dataguru.cn>

关于我

-  58集团技术委员主席
-  58同城高级系统架构师
-  即时通讯、转转、C2C技术负责人
-  前百度高级工程师
-  代表58同城对外嘉宾分享
 - QCon
 - SDCC
 - DTCC
 - Top100
 - 程序员
 - UPYUN
 - TINGYUN
 -

代表58对外交流

- Qcon(全球软件开发大会)
- SDCC(中国开发者大会)
- Top100(全球案例研究峰会)
- DTCC(中国数据库技术大会)
- 《程序员》撰稿2次
- 58技术发展这10年[计划中]



炼数成金课程

课程



《MongoDB 实战》

- 已开课
- 欢迎大家报名学习



《大规模高性能分布式存储系统设计与实现》

- 已开课
- 欢迎大家报名学习

上节课程回顾

- 数据存储的重要性
- 数据存储原理与设计
- 数据存储高可用的几个原理(定理)
- 数据存储层冗余我们如何做?
- 数据存储层数据备份如何落地?
- 数据存储层失效转移机制如何设计?
- 数据存储层数据一致性如何做到?
- 数据存储层库表设计实践 (见MySQL)
- 数据存储层如何做无缝的迁移
- 数据存储层高可用架构设计最佳实践是什么?
- 我们的实践案例;



Outline

- 👤 为什么需要缓存?
- 👤 缓存适用的场合
- 👤 高可用架构使用缓存类型有哪些? 各自作用是什么?
 - local、进程、分布式
- 👤 高可用架构使用分布式缓存类型? 如何选择?
- 👤 高可用架构缓存冗余如何设计?
- 👤 高可用架构缓存一致性如何保证?
- 👤 高可用架构缓存命中率如何保证?
- 👤 高可用架构缓存设计的最佳实践是什么?
- 👤 我们的实践案例;



为什么需要缓存?

缓存作用

- 加速请求的响应时间，响应延迟小
- 减少对固化存储的读压力

缓存适用场景

缓存适用场景

- 静态资源的缓存
- 较少更改资源的缓存
- 读多写少场景
 - 互联网
 - 读10写1
- 不适合场景
 - 频繁更新
 - 读少写多

高可用架构使用缓存类型有哪些？

缓存类型

- 本地缓存
- 进程内缓存
- 分布式缓存

高可用架构使用缓存类型各自作用是什么？

缓存作用

— 本地缓存

- 静态不变数据
- 减少网络I/O交互
- 58分类类名
 - 一级分类、二级分类、三级分类
 - » 数码、手机、苹果

高可用架构使用缓存类型各自作用是什么？

缓存作用

— 进程内缓存

- 和进程相关的动态数据
- 动态变化数据
- 不适合固化存储
- IM
 - IM状态
 - IM IP
- 相关业务逻辑
 - IM扫超时

高可用架构使用缓存类型各自作用是什么？

缓存作用

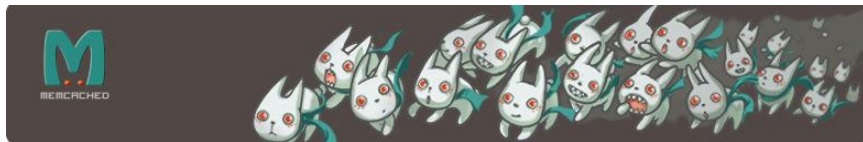
— 分布式缓存

- 缓存相对静态的数据
 - IM个人信息
 - IM个人消息
 - C2C商品信息
- 缓存数据量较大，单机无法存放
- 提高查询速度

高可用架构使用分布式缓存类型?

分布式缓存类型

— Memcached



- 分布式的缓存服务
- 支持仅支持Key-Value
- Key是字符串
- Value可以是字符串、也可以是Binary
- Server不是分布式
- 分布式支持靠客户端
- 性能较高 (W+吞吐量)
- 不可持久化, 仅仅缓存
- 使用简单, 使用较多

高可用架构使用分布式缓存类型？

分布式缓存类型



— Redis

- 分布式的缓存服务
- Key-Value
- Key是字符串
- Value可以是字符串、Set、Map、List等，支持类型丰富
- Server不是分布式
- 分布式支持靠客户端
- 性能较高（5W+吞吐量）
- Master-Slave方式，具有高可靠性、高可用性
- 具备持久化功能
 - RDB快照
 - AOF配置刷盘

高可用架构使用分布式缓存类型?

分布式缓存类型

— 自主研发

- 分布式的缓存服务
- 根据业务情况个性化定制
- 满足业务
- 性能更好
-
- 成本更高(研发)

高可用架构使用分布式缓存如何选择?

分布式缓存选择

— 业务场景

- 简单的Key-Value
 - Memcache
- 复杂的Key-Value
 - 计数、链表、集合、哈希等
 - Redis

— 性能要求

- Redis单机性能更好

高可用架构缓存冗余如何设计

高可用架构缓存的两个流派

- 缓存高可用保证
 - 缓存7*24可用
 - 防止穿透到数据库，对数据库冲击
- 缓存不高可用保证
 - 缓存就是缓存
 - 一旦缓存不可用，查询数据库
 - 那么问题来了，数据会不会挂掉？
 - 缓存集群，宕机某台几率大，宕机多台几率小
 - 宕机某台，对数据库影响不大，不会产生雪崩
 - 预估宕机一台，数据库可以承受的最大压力
 - » SSD
 - » 分布式部署
- 我个人倾向于后者

高可用架构缓存冗余如何设计

高可用架构缓存

- 缓存冗余
- 一份数据写入2份缓存
- 2份缓存都提供读写服务
- 写操作
 - 2份缓存都要写成功
- 读操作
 - 先到其中一台，读到直接返回，读不到读另一台，读到返回，若读不到，读数据库，并把读到数据回填2个缓存中；
 - 当其中一个缓存宕机后，读另一个缓存，当宕机恢复后，读请求同上；

高可用架构缓存一致性如何保证？

高可用架构缓存一致性

- 多份数据副本，数据一致性问题存在
 - 数据库、多份缓存
- 强一致性较难
 - 分布式锁、分布式事务
- 追求最终一致性
 - 互联网业务特点

高可用架构缓存一致性如何保证？

高可用架构缓存最终一致性

— 情景一

- 先更新数据库，再更新缓存
- 有问题吗？

— 情景二

- 先更新缓存，再更新数据库？
- 有问题吗？

高可用架构缓存一致性如何保证？

高可用架构缓存最终一致性

— 情景一

- 先更新数据库，再更新缓存
- 有问题吗？

— 情景二

- 先更新缓存，再更新数据库？
- 有问题吗？

高可用架构缓存一致性如何保证？

高可用架构缓存最终一致性

- 有更新操作时，更新数据库，同时delete缓存项
- 1秒钟后，再次失效一次缓存项
 - 双重失效，尽可能保证失效成功率
- 设置缓存项Expired Time
 - 根据业务特点设置
 - 10Min、1Hour
- 失败
 - 记录日志
 - 脚本定期修正
- 读请求
 - 读缓存，命中直接缓存直接返回
 - 不命中，读数据库，把结果回填缓存中

高可用架构缓存命中率如何保证？

高可用架构缓存命中率

- 业务不同，命中率不同
- 命中率80%+
- 相对静态数据缓存
- 缓存时间长些
- 定期查看缓存命中情况，适当调整缓存对象
-

高可用架构缓存最佳实践是什么？

高可用架构缓存最佳实践

- 业务特点，选用多级缓存
 - Local
 - Process
 - Distribute
- 缓存高可用性保证
- 缓存宕机，数据库最大压力评估
- 缓存一致性
 - 最终一致性
 - 强一致性场合少
 - 分布式锁、分布式队列

我们的实践案例

实践案例一

- 百度空间feed读压力大
- 保证缓存高可用性
 - 同前面讲述

我们的实践案例

实践案例二

- 58帮帮缓存

- MongoDB

- 《MongoDB在58同城的应用实践》

- » DTCC 2016

本课总结

- 👤 为什么需要缓存?
- 👤 缓存适用的场合
- 👤 高可用架构使用缓存类型有哪些? 各自作用是什么?
 - local、进程、分布式
- 👤 高可用架构使用分布式缓存类型? 如何选择?
- 👤 高可用架构缓存冗余如何设计?
- 👤 高可用架构缓存一致性如何保证?
- 👤 高可用架构缓存命中率如何保证?
- 👤 高可用架构缓存设计的最佳实践是什么?
- 👤 我们的实践案例;





THANK YOU