去哪儿? Qunar.Com

NoSQL 误用和常见陷阱分析

孙立@qunar.com
weibo.com@sunli1223

大纲

被误用的NoSQL

NoSQL陷阱

NoSQL与MySQL

NoSQL无处不在

NoSQL运维

被误用的NoSQL

非常容易出现的错误使用方法

循环网络调用

- Memcached的循环和批量GET对比

```
Map<String, String> result=new HashMap<String, String>();
     for (int i = 0,len=keys.length; i < len; i++) {
          //循环获取memcached数据
          result.put(keys[i], memcacheGet(keys[i]));
     }
```

10个key消耗10ms

```
//使用批量get协议
  Map<String, Object> objMap = client.get(Arrays.asList(keys));
```

10个key消耗2ms

5倍性能的影响

循环网络调用

- Redis的循环和批量GET对比

```
Map<String, String> result=new HashMap<String, String>();
     for (int i = 0,len=keys.length; i < len; i++) {
          //循环获取memcached数据
          result.put(keys[i], jredis.get(keys[i]));
     }
```

100个key消耗10ms

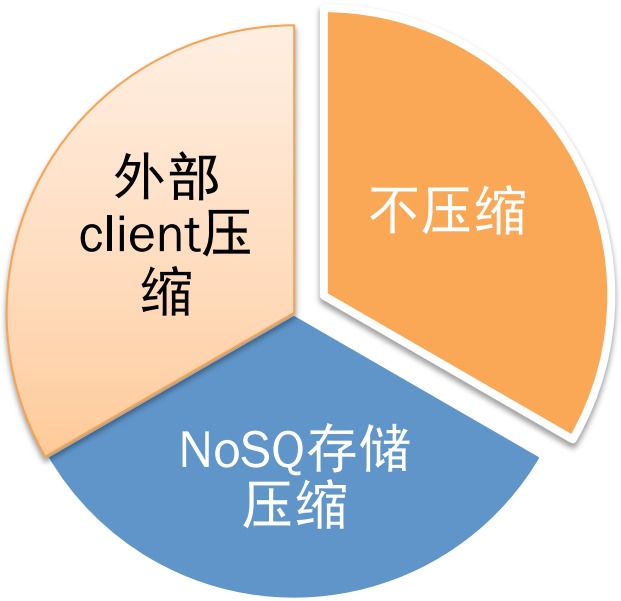
```
//循环从redis get数据
  jredis.mget(keys);
```

100个key消耗5ms

2倍性能的影响

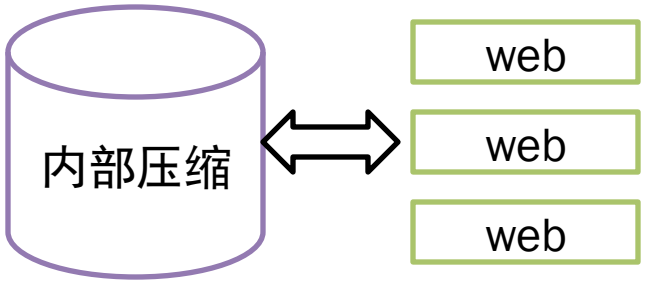
不压缩大数据

- 压缩的分类

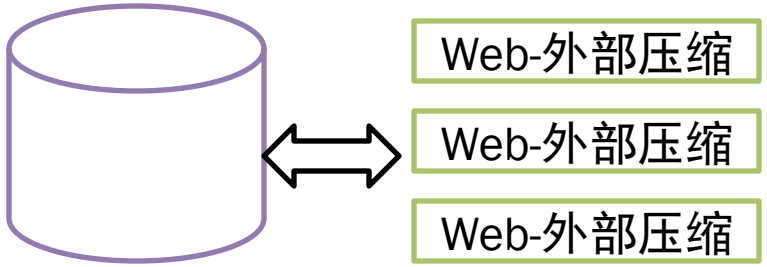


不压缩大数据

- 内部压缩和外部压缩



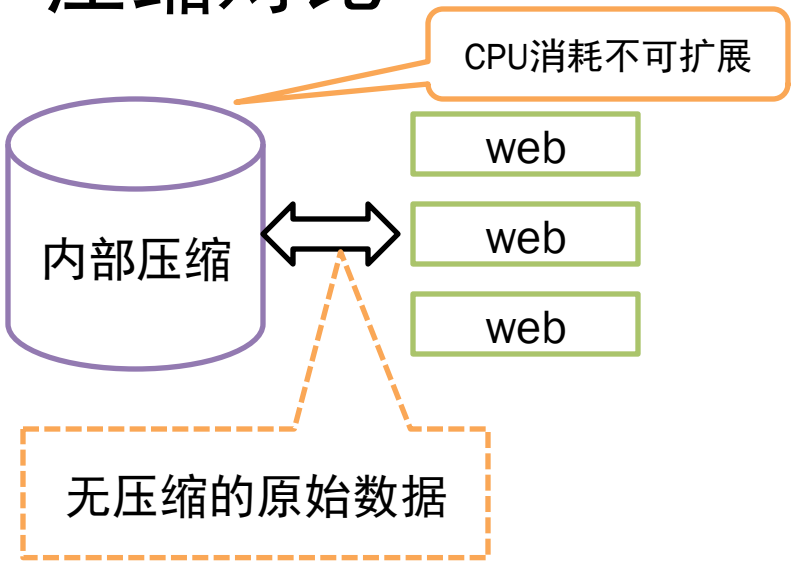
NoSQL内部压缩，可以减小存储，提升IO性能，不能提升网络IO性能

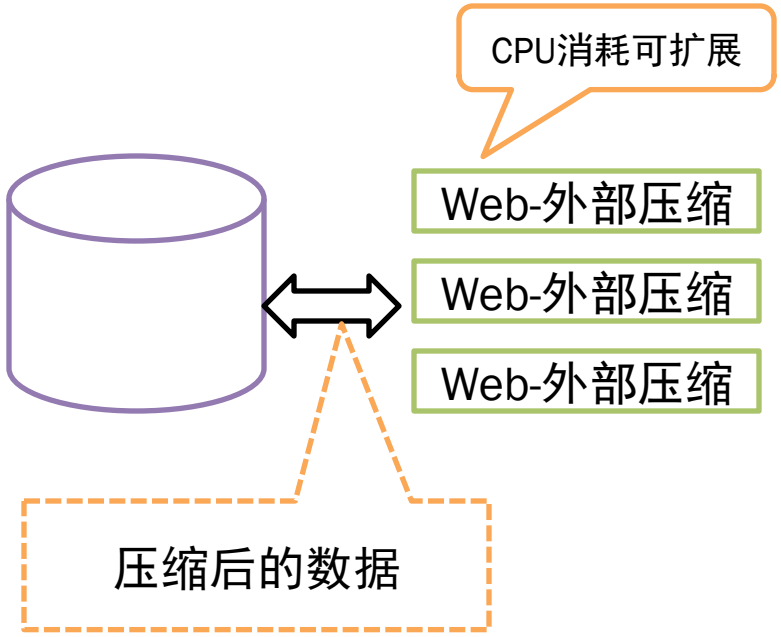


外部压缩，可以减小存储，提升IO性能，并且能够提升网络IO性能

不压缩大数据

- 压缩对比





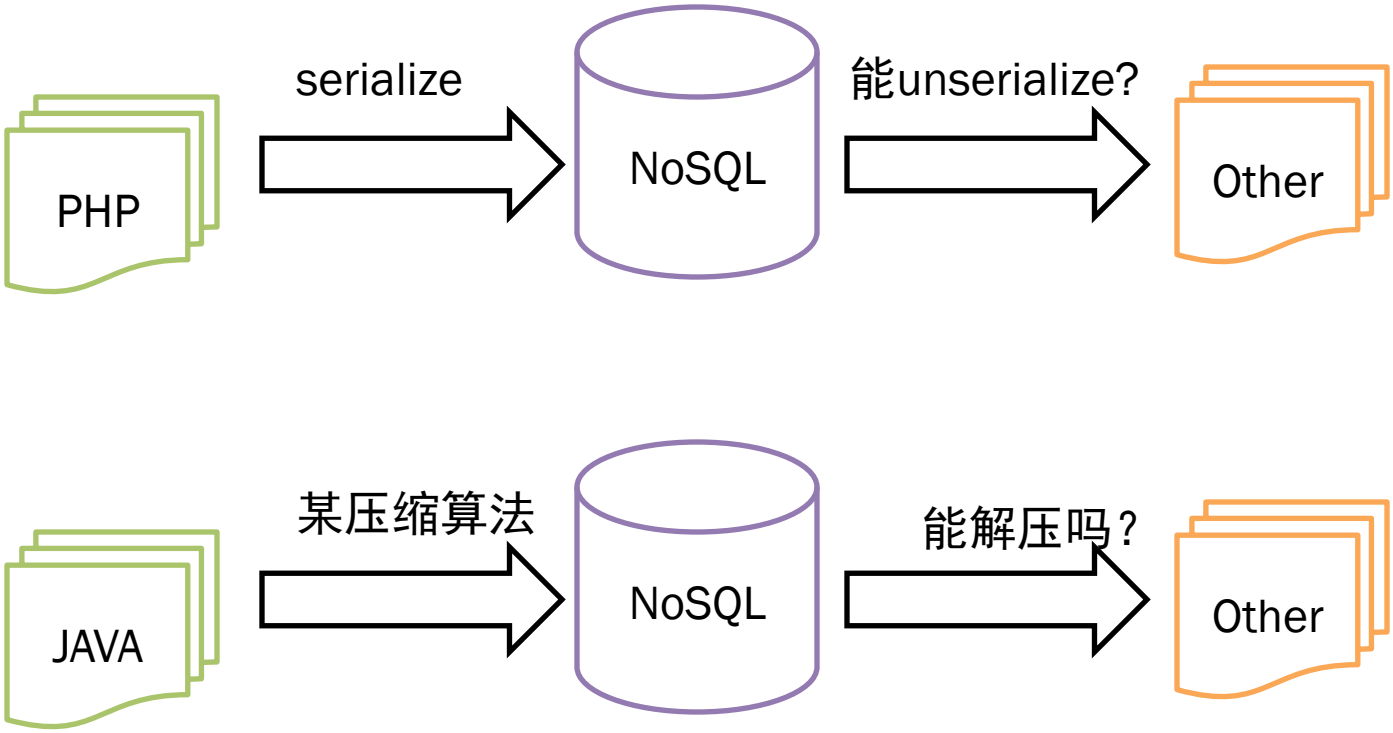
(1000*1024*1024/8)/(10*1024)=12800 qps

(1000*1024*1024/8)/(4*1024)=32000 qps

1000Mb网卡单条数据10KB的理论qps

去哪儿? Qunar.Com

跨语言交互



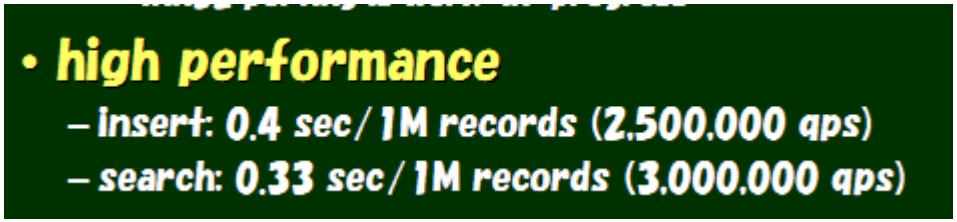
NoSQL陷阱

NoSQL是一个新兴的话题，大量的NoSQL产品也都是新出来的，难免出现各式各样的陷阱。

不能避免陷阱，但是得掉进去可以爬出来。

官方数据很美好

- 官方数据
- 很少有提及缺点的



Tokyo Cabinet 的官方数据。实际测试你会发现，数据增加后，速度会骤减

去哪儿? Qunar.Com

场景错误

- Redis做持久存储
  - 单点、复制问题
  - 数据超过内存性能急剧下降
  - 扩容问题
- Redis做Cache存储
  - 性能极高
  - 数据结构丰富
  - 可以持久化、避免雪崩现象

细节描述不清楚

- Ttserver=>兼容memcached协议

不支持memcached的flag参数

早期版本increment与memcached不一致

Stats命令不一致

去哪儿? Qunar.Com

缓存重建

- 系统重启后，大部分请求到磁盘
- 整个系统的性能可能出现抖动
- 出现连锁雪崩反应

NoSQL陷阱-32bit问题

- Ttserver -2GB（32bit）
- Mongodb-2.5GB (32 bit)

Ttserver在32bit下，到达2g数据大小
将出现无法启动的现象

去哪儿? Qunar.Com

版本升级问题

- 版本升级带来兼容问题（官方未声明的）
- 版本升级可能导致适用场景变化

NoSQL与MySQL

不要犹豫该用MySQL还是NoSQL，在你还没有掌握NoSQL前，最好先在小项目尝试下。

去哪儿? Qunar.Com

选择NoSQL需要考虑

- 数据的安全性-是否久经考验
- 事务性的保障
- 数据的重要性
- 是否有DBA运维
- 未来的业务需求变化

性能之争—差别在哪里?

- 普通磁盘的IOPS(几百个)
- 寻道时间、延迟时间
- 顺序读和顺序写吞吐上百MB/S

如果是SSD?

随机写变顺序写

内存索引-热数据cache

读写算法优化（场景化）

网络协议优化

NoSQL在普通磁盘的优化

NoSQL无处不在

不管你信不信，你很可能早已在接触NoSQL了。

比如：

- Memcached缓存
- SVN使用的BDB

去哪儿? Qunar.Com

为什么要构建自己的NoSQL

- 考察清楚场景和需求
- 现有产品满足需求成本高
- 针对特殊场景，也许比想象的简单
- 可掌控

构建自己的NoSQL

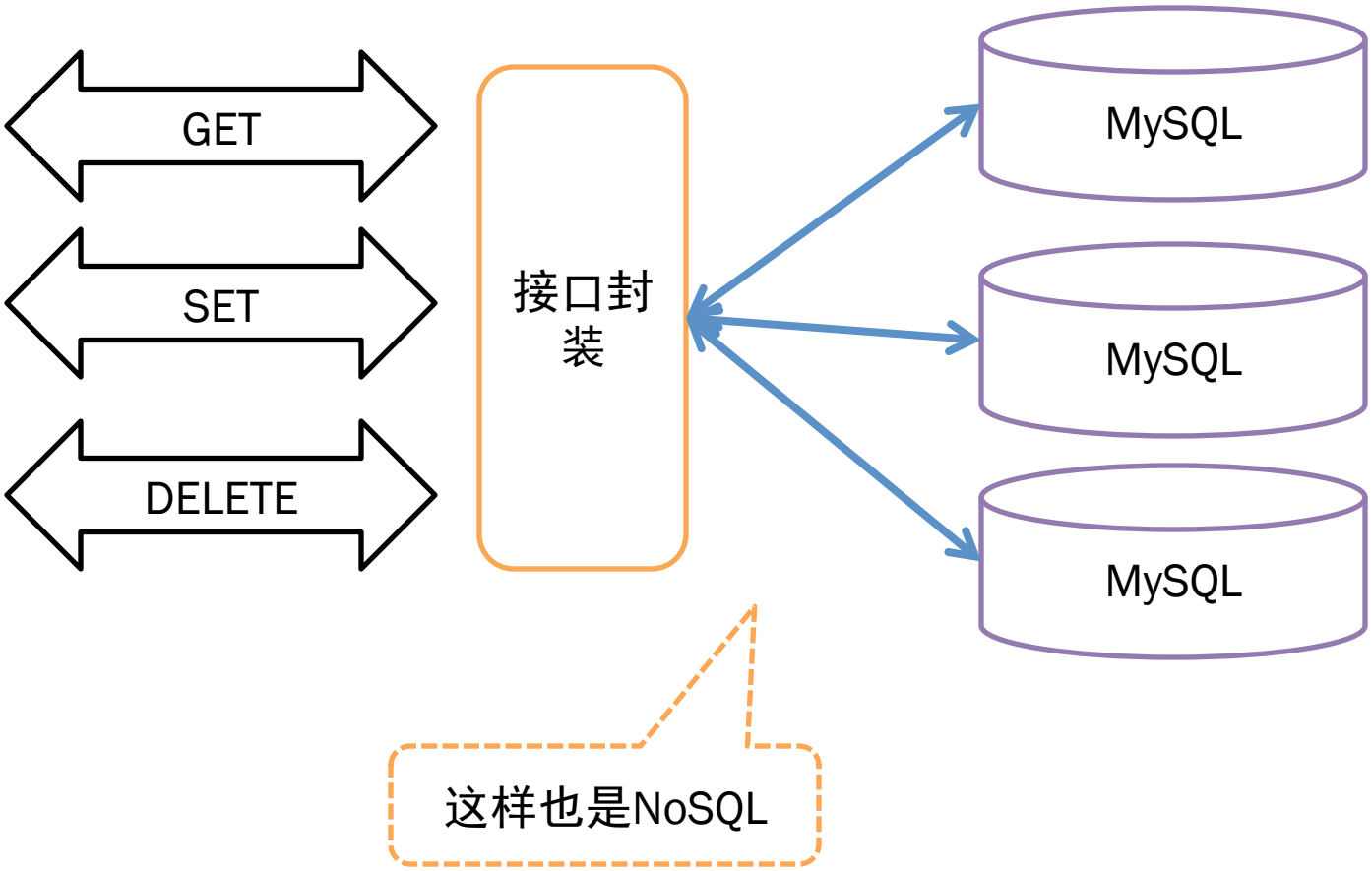
- IP查询

TreeMap<IPEntry, String> 可轻松完成

```
1  class IPEntry implements Comparable<IPEntry> {
2          final long start;
3          final long end;
4          IPEntry(long start, long end) {
8          public int compareTo(IPEntry t) {
9              long t1 = start - t.start;
10             if (t1 < 0)
11                 return -1;
12             long t2 = end - t.end;
13             if (t1 >= 0 && t2 <= 0)
14                 return 0;
15             return 1;
16         }
17         public String toString() {
20 }
```

构建自己的NoSQL

- 通过MySQL构建



NoSQL运维

并不像官方宣称的那样，NoSQL无需DBA运维

去哪儿? Qunar.Com

运维NoSQL并不容易

- 文档齐全吗？
- 网上交流多嘛？
- 周边工具齐全吗？
- 出现意外问题你能搞定吗？

出现意外问题，很难求助到人

监控-运维之本

除了操作系统最基本的监控，还应该监控

- IO
- CPU
- 延迟
- QPS
- 抖动
- 数据量

去哪儿? Qunar.Com

备份很重要

- 避免单点
- 定期数据库备份
- 开发前做好切换准备（能切换到其他产品）

谢谢！


QCon
北京站·2012年4月18~20日
www.qconbeijing.com（11月启动）
QCon杭州站官网和资料
www.qconhangzhou.com
全球企业开发大会
INTERNATIONAL
SOFTWARE DEVELOPMENT
CONFERENCE