


性能优化--锁、缓存

 菠菜谈天说地
百家号 | 05-02 16:16

关注

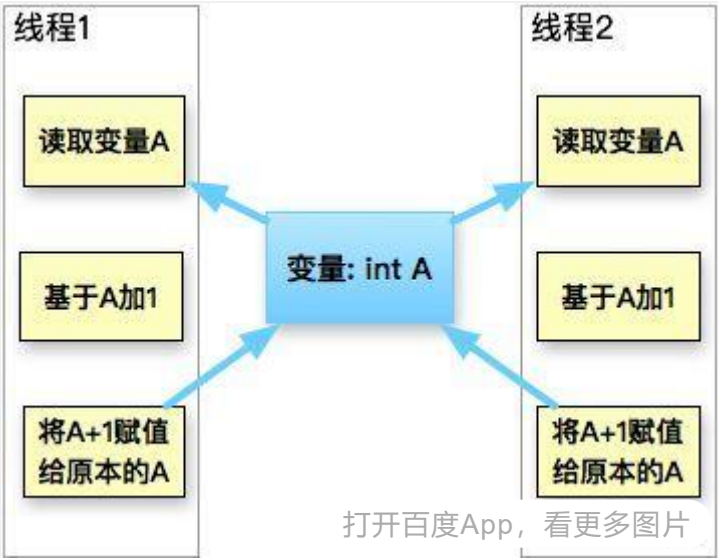
前言

掌握了《性能优化--基础》后，我们再来谈谈锁和缓存。菠菜会介绍一些在高并发场景下使用锁跟缓存的经验，希望能给大家带来不一样的视角来理解。

锁

为什么要使用锁

众所周知，使用锁是保障在并发场景下写数据时的正确性。比如，两个线程同时给一个int型变量做"+1"操作，大体流程如下图：



而在并发场景下，线程1、2的各个步骤先后顺序是无法控制的，有可能出现如下情况：假设A=1，线程1、2同时获取A的值都得到1，各自加1后都得到相同结果2，然后再将结果赋值给A，那相当于给A赋值了两次，每次都是2，但我们期望的结果是3。这就是需要锁的来源。

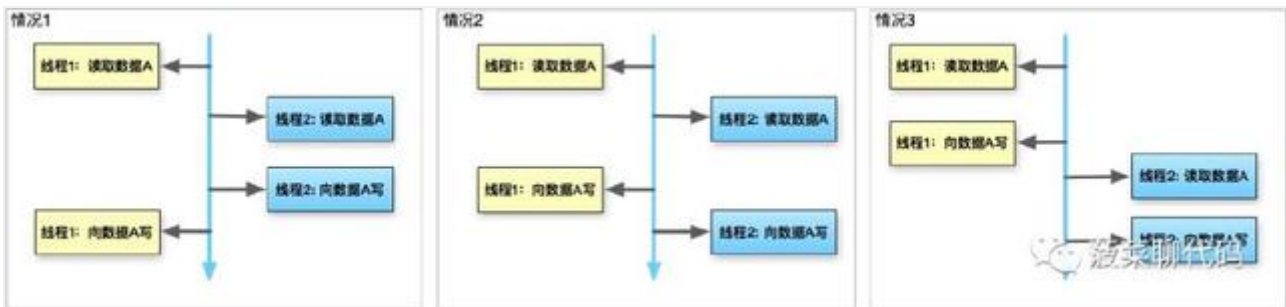
锁的本质是将一系列的操作原子化，即变成原子操作。线程1的三个步骤必须一次性全部执行完，才能执行线程2的步骤。这样最终就能得到期望的结果3。

复杂的并发场景下如何将冲突想清楚

在实际的工作中，并发往往比上图描述的场景复杂的多，那我们该如何想清楚呢？菠菜教你四步即可找到并想清楚冲突点：

1. **冲突点的定位**，多线程对同一个数据有写操作，一般会有冲突。
2. **简化线程数量**，n多线程的并发场景下，仅需想清楚两个线程间的交叉冲突即可，不需要考虑全部线程间的冲突。
3. **简化操作步骤**，无论有多少行代码，都可以抽象出两步，一步是取冲突的数据，一步是写冲突的数据。
4. **确认要不要锁**，如果在冲突的情况下数据被写成非预期的值，则需要加锁，否则不需要。

根据这三个步骤，都可以将任意有可能存在冲突的场景抽象成下图：



三种情况中，情况3是不会产生冲突的，而情况1、2会产生冲突，所以大家着重考虑这两种情况。最后一步确定，在这种场景下，数据是否会被写错，若写错了就需要加锁。

这里举一个不需要加锁的场景--缓存，多线程同时从缓存取数据，若发现缓存中没有数据便去DB中查询，然后放入缓存中，在这种情况下会出现缓存短时间内被多次刷新，但刷新的最终结果是期望的值，所以不需要锁。

锁的粒度

在高并发场景下，一定要控制好锁的粒度，否则会导致性能急剧下降。那需要从两个点来考虑锁的粒度：

1. **参与同一个冲突的线程数量尽可能的少**。例如之前的文章《12306--如何实现查票、购票》中，购票的锁只有当用户购买同一张票的不同站点的时候才会冲

去百度App听

突，如果每次购票都是锁住列车，那这个冲突的线程的数量就非常高了，性能会非常差

2. **锁的时间尽可能的短。**在lock以及unlock之间，尽可能的减少跟锁无关的操作，降低它们之间的时间间隔，性能会大大提升。

分布式锁

之前讲的都是单台机器上同一个进程里不同线程间的锁，而在实际开发中有可能遇到多进程间锁甚至是跨机器间的锁：

1. **进程间锁**，可以使用文件锁来实现，即对同一个文件进行锁操作来达到进程间锁的效果。
2. **机器间锁**，可以使用具有原子操作能力的中间存储来解决，比如memcached的increment操作、redis的setnx/incr都可以。对于increment类似的操作，我们可以认为当increment获取的值为1的时候即获得了锁（解锁del掉key即可）。另外死锁也是需要注意的，比如一台机器获取了锁，但是还没解锁就宕机了，这时候后续都永远不可能获得锁了，为了避免这种情况，可以给存储设置过期时间。当然还有更加复杂的paxos、raft等这里就不过多描述了，大家可以自行百度呦。

除了使用锁，有些场景大家可以直接使用**原子变量**，对了，还有一个无锁的队列叫做**无锁环形队列**，大家可以自行了解。

缓存


最后来谈谈缓存。为什么用缓存？常见的答案是DB扛不住啊，用了缓存性能会提高很多啊，但再深层次的原因是什么呢？加了缓存性能一定就高了吗？

菠菜认为，**缓存是利用更高性能的内存替代较弱性能的硬盘从而提高性能**。而常见的缓存方式分为本地缓存跟远程服务缓存。其实我们通过上一篇《性能优化--基础》文章可以知道，远程缓存会有网络IO操作，量级从纳秒变成了毫秒，高了两个量级（1,000,000倍）。

去百度App听

加了缓存性能就一定高吗？答案是NO，这要看缓存命中率。假设我们使用Redis做MySQL的缓存，一般的伪代码如下：

```
cache = getFromCache(key)
if cache == nil {
    cache = getFromDB(key)
    setCache(key, cache)
}
```

 菠菜聊代码

若缓存命中率低，换句话说大部分时候无法从cache里获取数据，而必须从db里获取的话，这时候如果去掉缓存，会减少2次额外的网络IO操作，性能会快不少。

另外，当初NoSQL开始火起来的时候，我们可以很清晰的感知到NoSQL与SQL(MySQL、PostgreSQL等等)的性能差距很大，其实很核心的原因是NoSQL大量的使用了内存替代硬盘从而性能大增（如MongoDB会在大量的缓存）。

最后

玩转了多线程/多进程/多机器间锁的问题，对于之后架构大流量高并发的架构来说是至关重要的。而学了性能优化基础篇之后，再来看待之前习以为常的技术方案，就会理解更加深入。下次见咯~

本文由百家号作者上传并发布，百家号仅提供信息发布平台。文章仅代表作者个人观点，不代表百度立场。未经作者许可，不得转载。

[举报](#)

相关阅读

Winform线程显示处理进度

去百度App听

```
public partial class FormThread : Form
{
    1 个引用
    public FormThread()
    {
        InitializeComponent();
        Task.Run(() =>
        {
            for (int i = 0; i < 100; i++)
            {
                Thread.Sleep(10);
                this.progressBar.Value += 1;
            }
        });
    }
}
```

打开百度App 撩说

集性能与颜值于一身 这些新品你不了解一下？



打开百度App 太平洋电脑网

性能好颜值高，选这样一台游戏本准没错



打开百度App 太平洋电脑网

打开百度App看更多资讯

精彩视频

斯柯达不再有大众味了，外观远超CC，性能优化超思域

02:01

打开百度App 机灵小不懂

性能测试，高通835能带动狂野飙车9吗？这优化真不错

去百度App听

02:13

打开百度App 游戏大搜索

谁还说逆水寒优化不好，连掌上电脑的性能配置都可以玩！

01:00 [打开百度App](#) 游戏大搜索

F-16战斗机根据越南战争空战经验，大幅优化视距内格斗性能

01:49 [打开百度App](#) 今天告诉你

热门推荐

盘点在演艺圈日渐残酷的竞争下耸立不倒的五位小生，鹿晗只排第二



[打开百度App](#) 杨子侃电影

因抽烟数次流产？小仙女也是个老烟枪？网友：清纯人设已崩塌！



[打开百度App](#) 娱乐圈扒妹妹

绿色不愧是王源的应援色！薄荷少年演绎绿衣，是青春的气息



[打开百度App](#) 星仔说电影

去百度App听

晒晒我的新家，沙发墙做软包实在好看，吊顶漂亮整个逼格都变高了



打开百度App 小石很高兴

搞笑趣图



哎呦喂，可爱死了，好想摸

打开百度App 搞笑奇趣汇

这笔记本是我的，谁都别和劳资抢！



打开百度App 搞笑奇趣汇

大姐，你的这辆“大奔”好靓啊



打开百度App 囧人囧事2333

媳妇为了防晒，上街顶了一个水桶

去百度App听



打开百度App 熊孩子基地

热门评论

打开百度App,说说我的想法

去百度App听