一、介绍项目

二、介绍完成效果（视频）

三、介绍发现问题及解决（猜想或实现）

四、介绍拓展功能

消息队列

消息队列是为了解决生产和消费的速度不一致导致的问题，有以下好处：

1、减少请求响应时间。比如注册功能需要调用第三方接口来发短信，如果等待第三方响应可能会需要很多时间

2、服务之间解耦。主服务只关心核心的流程，其他不重要的、耗费时间流程是否如何处理完成不需要知道，只通知即可

3、流量削锋。对于不需要实时处理的请求来说，当并发量特别大的时候，可以先在消息队列中作缓存，然后陆续发送给对应的服务去处理

如果想要实现一个消息队列，可以参考这里

最简单的消息队列就是一个消息转发器，基本功能只有三个：消息存储、消息发送、消息删除，可使用LinkedBlockingQueue、ConcurrentLinkedQueue实现

Apache JMeter测试 java请求。

经测试发现，第一次建立线程池中线程所花时间为2ms左右，可以忽略，故从服务器线程开始run开始计时。

线程池中线程数量：10，30，50, 70，100， cache

客户端线程数量：10, 100，1000, 10000，100000

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 10 | 30 | 50 | 70 | 100 | cache |
| 10 | 0.003 | 0.002 | 0.003 | 0.003 | 0.002 | 0.003 |
| 100 | 1.001 | 0.629 | 0.520 | 0.506 | 0.511 | 0.512 |
| 1000 | 10.886 | 3.623 | 2.186 | 1.548 | 1.185 | 1.075 |
| 10000 | 111.890 | 40.689 | 37.131 | 18.382 | 11.958 | 2.411 |
| 100000 | 2537.81 | 887.501 | 868.782 | 810.247 | 785.427 | 781.043 |

200000 2427.142 2441.427

注：1.当客户端线程10000次左右时出现不明卡顿（java 线程数量过大时有时候cpu占用会突然变高）。

2.当客户端线程数超过100000时接近线程最大数临界值。

3.一般当线程完成数量达到一半以上时开始频繁卡顿，尤其是运行到3/4时最明显。

4.切换不同进程，特别是需要切换I/O设备时，会更容易导致线程卡顿（猜测）。

5.经测试发现，当线程数量达到250000时，会出现内存泄露错误。（尝试JVM优化）

new Thread的弊端如下：

a. 每次new Thread新建对象性能差。

b. 线程缺乏统一管理，可能无限制新建线程，相互之间竞争，及可能占用过多系统资源导致死机。

c. 缺乏更多功能，如定时执行、定期执行、线程中断。

相比new Thread，Java提供的四种线程池的好处在于：

a. 重用存在的线程，减少对象创建、消亡的开销，性能佳。

b. 可有效控制最大并发线程数，提高系统资源的使用率，同时避免过多资源竞争，避免堵塞。

c. 提供定时执行、定期执行、单线程、并发数控制等功能。

Java线程池：在面向对象编程中，创建和销毁对象是很费时间的，因为创建一个对象要获取内存资源或者其它更多资源。在Java中更是如此，虚拟机将试图跟踪每一个对象，以便能够在对象销毁后进行垃圾回收。所以提高服务程序效率的一个手段就是尽可能减少创建和销毁对象的次数，特别是一些很耗资源的对象创建和销毁。如何利用已有对象来服务就是一个需要解决的关键问题，其实这就是一些"池化资源"技术产生的原因。

newFixedThreadPool和newCachedThreadPool比较分析：

newFixedPool作用：创建一个固定线程数的线程池，在任何时候最多只有nThreads个线程被创建。如果在所有线程都处于活动状态时，有其他任务提交，他们将等待队列中直到线程可用。如果任何线程由于执行过程中的故障而终止，将会有一个新线程将取代这个线程执行后续任务。

newCachedThreadPool作用：一个可根据需要创建新线程的线程池，如果现有线程没有可用的，则创建一个新线程并添加到池中，如果有被使用完但是还没销毁的线程，就复用该线程。终止并从缓存中移除那些已有 60 秒钟未被使用的线程。因此，长时间保持空闲的线程池不会使用任何资源。

只有当执行短寿命异步任务时，使用newCachedThreadPool，如果提交需要较长时间处理的任务，最终会创建太多的线程。如果您以更快的速度向newCachedThreadPool（http://rashcoder.com/be-careful-while-using-executors-newcachedthreadpool/）提交长期运行任务，则可能会达到 100% CPU。