

多线程技术在网络通信系统的设计与研究

王启志

和勤通信技术有限公司 浙江 杭州 310000

摘要: Java语言是一种适合分布计算的新一代程序语言,是一种新概念。多线程技术是Java高级程序语言解决并发处理问题的解决方案,被广泛应用于具有网络通信功能的软件开发工作。本文分析了线程之间通信的原理,介绍了Java多线程技术实现网络通信的原理,指出了Java多线程技术实现线程安全及同步控制的有效途径,提出了基于Java多线程技术的网络通信系统实现方法改进方案。

关键词: Java高级程序语言;多线程技术;网络通信系统

Java最明显的特征是在语言当中融合多线程机制。所谓的线程,是指在一个程序当中的执行流。Java是面向对象的高级编程语言,它利用网络程序包,实现各种模式网络通信。它可以不依赖任何机器结构,有可移植性、安全性、健壮性和高性能等特点,可以通过多线程来提供并发机制。另外,它有强大与独特的通用机制与能力,简化了程序的繁琐,而且可以不受环境限制。现在,Java已经成了最好的网络开发语言。

1 多线程简介

多线程指的是一种程序的运行机制,可以允许程序中同时执行不止一个指令流。其中每个都被称作一个线程,它们彼此之间是相互独立的。线程同时又可以叫做轻量级进程,它与进程一样,都有独立执行控制、操作系统进行调度。不同的是,线程没有自己独立的存储空间,它和在其所属进程中的线程,共同享有一个存储空间。这样一来,线程的通信就变得简单很多。

各个线程执行是并发的。从逻辑上来说,是同时的,不管在物理上面是否如此。假如系统仅有一个CPU,在这种情况下,真正的同时是不可能实现的。但是因为CPU速度很快,所以用户无法察觉其中差别。

多线程技术对线程的管理,就是让线程之间共享资源,通过程序干预来控制线程的执行,在必要时刻还要干扰线程获取资源来实现系统的共享资源异步访问机制。多线程实现了线程的资源异步访问,也实现了线程之间的通信和执行排序任务。把多线程的机制包含在语言当中,这是Java一个重要的特点。而有了这个特点,利用Java多线程的编程接口,方便开发人员写出可以支持多线程的程序,从而能够有效减少并行程序的设计难度,提高程序的执行效率。

作者简介: 出生年月:1978.10.27民族:汉性别:男籍贯:浙江省磐安县 职位:运维技术部经理 学历:本科

2 Java多线程技术实现网络通信的原理

网络通信系统一般分为服务器和客户端两个组件。早期的服务器端口只能和一个客户端进行通信,当前一个通信未完成时后续访问都处于等待状态,只有关闭端口重新连接才能进行新的线程通信。随着程序设计语言的不断发展和网络环境的不断优化,网络通信系统需要具备多服务器和多客户端相互访问的能力。因此,在解决网络通信问题时就需要多线程技术的支持。

Java的多线程技术支持线程共享存储空间,并且实现了不同线程之间的信息同步与交流,即网络通信。Java语言处理多线程问题的步骤如下:(1)对处理多线程的条件指令进行校验判断,只有在校验通过的情况下才可以让后续的线程继续执行;(2)当某一个线程指令完成后,要通知条件指令进行二次校验,满足条件后让其他线程继续执行;(3)执行上述两个步骤,让线程不断循环执行下去,实现了不间断的网络通信功能。

3 Java多线程技术实现线程安全及同步控制

多线程编程会遇到最大的挑战就是线程安全和同步控制问题。线程安全是指线程并行执行时出现的数据和执行问题,比如线程A在进行集合的遍历查询,线程B则在进行集合的清空操作,这两个线程A和B如果同时进行会出现问题,谁先执行也会对对方的正常执行造成影响。同步控制是指在线程A操作数据时通知其他线程无法获得当前数据,只有线程A完成后其它线程才可以进行访问。Java多线程技术实现线程安全及同步控制的新方法有synchronized同步锁、wait/notify等待/通知机制、信号量机制等。(1)synchronized同步锁。Java使用synchronized关键字来实现线程同步锁机制,线程可以使用synchronized来申请资源锁定,别的线程在访问带有synchronized关键字的资源时就会自动进入等待状态。当持有synchronized锁的线程完成操作后,可以释放同步锁

让其他线程继续访问系统资源。(2) wait/notify等待/通知机制。在处理线程数量较大,彼此之间频繁通信的多线程场景下,可以使用唤醒等待机制来实现网络通信功能。Java中的wait/notify2个方法就实现了线程的等待通知机制。wait/notify指令要在synchronized同步锁的基础上执行。对于获得synchronized同步锁的线程,如果在执行过程中需要等待,可以调用wait方法来进入堵塞队列,与此同时synchronized锁也会失效,其他线程开始获得同步锁进行处理,处理完成后调用notify方法来归还synchronized同步锁。当堵塞队列中有较多线程排队时,调用了notify方法的线程会随机通知一个等待线程进行激活,也可以调用notifyAll方法来通知所有等待的线程,得到通知的线程会进入就绪队列进行后续处理。

4 Java多线程实现通信编程

网络通信编程又称为Socket编程,Socket是Java多线程技术的主要类实现。Socket编程基于TCP网络,分别实现服务器和客户端。调用ServerSocket来构建服务器线程,为服务器设置固定的IP和端口号;多个客户端程序可以借助Socket实现与服务器的通信和交互,连接结束后服务器还会创建一个Socket对象来记录该次连接。Java多线程实现通信编程的过程如下:(1)网络通信系统服务器端实现。创建服务对象ServerSocket,让服务器线程等待客户端请求,并返回一个代表本次连接的Socket对象。ServerSocket对象的主要方法有:新建对象,让更多的连接进行定义;按照一定规则构建数据交换机制,记录各个线程的输入输出情况;关闭连接和其他系统资源。

(2)网络通信系统客户端实现。服务器端进入等待状态后,可以初始化Socket对象来向服务器端发送请求连接申请,获得申请许可后则按照一定规则进行数据交互,完成连接和操作后也要进行客户端的连接关闭实现。

5 Java多线程实现网络通信系统的改进思路

传统的网络通信系统的服务器程序只能完成一次读取操作,当Socket对象生成并完成请求应答后,再进行

后续请求就要断开重连,否则就会对系统服务器造成较大性能影响。为了解决这一问题,可以使用无限循环语句while(true)来实现服务器端的反复进行,但由于程序完成一次信息读取后就会进入等待状态,再使用无限循环语句也不会有什么实际的意义。为了解决无限循环锁死的问题,使用现成技术手段让服务器在循环内读取Socket信息。具体方法就是在服务器结构中加入内部类ReadMessageThread,实现多客户端访问服务器的安全和同步管控。

服务器程序与客户端自由通信的前提是调用serverSocket.accept()方法,但由于该方法在服务器端口只执行了一次,只能与一个客户端进行通信,另外端口被占用也无法进行自有通信。对这一问题改进的思路是,让服务器端Socket对象与各个客户端Socket之间建立虚拟线路。具体的改进位置如下:(1)创建用于监听线程的类ConnectSocket,该类在客户端访问服务器场景中生效,可以为每个客户端新建一个对应的服务器网络对象Socket,建立多对多的客户端连接后,生成并启动通信线程类ReadMessageThread(2)修改线程类ReadMessageThread,使得服务器端Socket对象成为通信线程的一个参数。

结语

线程是系统进行调度与运行最小的单位。多线程可以开发有并行性与高可靠性的程序,而且能节省系统的资源。从原理与应用的角度探讨Java多线程的技术,结合具体多用户的并发服务器通信程序来讲述开发通信程序的方法。这对Java多线程技术在网络通信系统中的实际运用思路研究有很强的指导意义。

参考文献

- [1]赵从军.Java语言程序设计使用教程[J].大连理工大学学报,2017(12).
- [2]刘瑞祥.基于Java语言多线程计算[J].计算机工程,2015(14).