



java多线程（八）为什么弃用stop和suspend

原创

2015年08月27日 09:12:23

标签：多线程 / java / 线程 / 编程 / 安全

5363



5

转载请注明出处：http://blog.csdn.net/xingjiarong/article/details/47984659



初始的java版本中定义了一个stop方法来终止一个线程还定义了一个suspend方法来阻塞一个线程，直到另一个线程调用resume方法。这两个方法在Java SE 1.2之后就被弃用了，因为这两种方法都不安全，下面我们分别来讨论一下为什么不安全和应该怎样做才是安全的。

一、stop方法为什么不安全

其实stop方法天生就不安全，因为它在终止一个线程时会强制中断线程的执行，不管run方法是否执行完了，并且还会释放这个线程所持有的所有的锁对象。这一现象会被其它因为请求锁而阻塞的线程看到，使他们继续向下执行。这就会造成数据的不一致，我们还是拿银行转账作为例子，我们还是从A账户向B账户转账500元，我们之前讨论过，这一过程分为三步，第一步是从A账户中减去500元，假如到这时线程就被stop了，那么这个线程就会释放它所取得锁，然后其他的线程继续执行，这样A账户就莫名其妙的少了500元而B账户也没有收到钱。这就是stop方法的不安全性。

二、suspend方法为什么被弃用

suspend被弃用的原因是因为它会造成死锁。suspend方法和stop方法不一样，它不会破换对象和强制释放锁，相反它会一直保持对锁的占有，一直到其他的线程调用resume方法，它才能继续向下执行。

假如有A，B两个线程，A线程在获得某个锁之后被suspend阻塞，这时A不能继续执行，线程B在或者相同的锁之后才能调用resume方法将A唤醒，但是此时的锁被A占有，B不能继续执行，也就不能及时的唤醒A，此时A，B两个线程都不能继续向下执行而形成了死锁。这就是suspend被弃用的原因。

三、如何正确的终止一个线程

```
[java]
1. class MyThread extends Thread {
2.     public volatile boolean flag = true;
3.
4.     public void run() {
5.         while (flag) {
6.             System.out.println("线程正在执行");
7.             try {
8.                 Thread.sleep(500);
9.             } catch (InterruptedException e) {
10.                // TODO Auto-generated catch block
11.                e.printStackTrace();
12.            }
13.        }
14.        System.out.println("线程已终止");
15.    }
16. }
17.
18. public class Main {
19.
20.     public static void main(String[] args) {
21.         MyThread myThread = new MyThread();
22.         myThread.start();
23.         try {
24.             Thread.sleep(3000);
25.         } catch (InterruptedException e) {
26.             // TODO Auto-generated catch block
27.             e.printStackTrace();
28.         }
29.         myThread.flag = false;
30.         System.out.println("主线程结束！");
31.     }
32. }
```

我们可以从代码里 看到作者写的代码 在代码中的Thread.sleep(500)这里会不断的检测flag的值 为true而结束线程的执

联系我们



请扫描二维码联系客服

✉ webmaster@csdn.net

☎ 400-660-0108

👤 QQ客服 🗣 客服论坛

关于 招聘 广告服务 百度

©1999-2018 CSDN版权所有

京ICP证09002463号

经营性网站备案信息

网络110报警服务

中国互联网举报中心

北京互联网违法和不良信息举报中心

博主最新文章

更多文章

NFV&SDN: Friends or Enemies?

Edge Computing Implementation Base d on Docker Containers

Making up VMs from Many Weak Nod es of Edge Computing

Edge Computing Implementation Base d on VM

Edge Computing Application：Real-Ti me Face Recognition Based on Cloudl et

文章分类

环境搭建	27篇
php	1篇
java	16篇
算法与数据结构	1篇
操作系统	5篇
综合	6篇

展开

博主专栏

 9 篇	java多线程 📖 37262
 18 篇	写最好的设计模式专栏 📖 58035
 10 篇	OpenStack配置 📖 33527

展开

文章存档

2018年1月	1篇
2017年9月	7篇
2016年12月	1篇
2016年4月	1篇
2016年3月	1篇
2016年2月	14篇

展开

博主热门文章

登录

注册



如果不设置话，可能会造成子线程多执行一次的错误，例如子线程将flag读到自己线程栈中，flag的值为true，此时子线程的交出执行权，操作系统将执行权交给了主线程，主线程执行flag=false；的操作，希望子线程不要再执行了，但是这一改变子线程是不能看到的，所以子线程还会再向下执行一次，然后重新读取flag的值的时候才会终止。

当然你也可以将对flag所有操作都封装在synchronized关键字修饰的方法中，实现互斥访问，也可以达到相同的效果。



四、如何正确的阻塞线程

如果想安全的阻塞线程，引入一个变量suspendRequested并在run方法安全的地方测试它，安全的地方是指该线程没有封锁其他线程需要的对象的地方。当线程发现如果变量已被设置，将会保持等待状态直到它再次被获得为止。

```
[java]
1. public volatile boolean suspendRequested = false;
2. private Lock suspendLock = new ReentrantLock();
3. private Condition suspendCondition = suspendLock.newCondition();
4.
5. public void run() {
6.
7.     while (true) {
8.         if (suspendRequested) {
9.             suspendLock.lock();
10.            try {
11.                while (suspendRequested)
12.                    suspendCondition.await();
13.            } catch (Exception e) {
14.
15.            } finally {
16.                suspendLock.unlock();
17.            }
18.        }
19.    }
20. }
21.
22. public void requestSuspend() {
23.     suspendRequested = true;
24. }
25.
26. public void requestResume() {
27.     suspendRequested = false;
28.     suspendLock.lock();
29.     try {
30.         suspendCondition.signalAll();
31.     } catch (Exception e) {
32.         e.printStackTrace();
33.     } finally {
34.         suspendLock.unlock();
35.     }
36. }
```

版权声明：本文为博主原创文章，转载请注明出处，查看原文，请访问：http://blog.csdn.net/xingjiarong
https://blog.csdn.net/xingjiarong/article/details/47984659

本文已收录于以下专栏：java多线程

世界上最好的语言，PHP凭什么这么“说”

课程包含PHP零基础入门、PHP基础编程、数据库、项目实战、linux、框架开发到高级+企业项目实战，深入浅出地剖析和分解了PHP企业级开发项目在实际工作中的应用。

查看更多>>

18502

查看更多>>



目前您尚未登录，请 登录 或 注册 后参与评论

0312222

联系我们



请扫描二维码联系客服
webmaster@csdn.net
400-660-0108
QQ客服 客服论坛

关于 招聘 广告服务 百度
©1999-2018 CSDN版权所有
京ICP证09002463号

经营性网站备案信息
网络110报警服务
中国互联网举报中心
北京互联网违法和不良信息举报中心

0312222

python绘制非常漂亮的图表

11790

C语言scanf（）和gets（）及printf（）和puts（）的区别



11693

Java中的线程Thread方法之---suspend()和resume()

前篇说到了Thread中的join方法，这一篇我们就来介绍一下suspend()和resume()方法，从字面意义上可以了解到这两个方法是一对的，suspend()方法就是将一个线程挂起(暂停)，re...

 jiangwei0910410003 2014-02-25 14:37:58  15496



Java线程stop和suspend的废弃

 blackwuxin 2012-02-15 14:34:36  9289

从JDK1.0定义了stop和suspend方法,stop用来直接终止线程,suspend会阻塞线程直到另一个线程调用resume. stop和suspend都有些共同的点：都试图专横的控制一个给定...




java多线程（八）为什么弃用stop和suspend

 xingjiarong 2015-08-27 09:12:23  5363

初始的java版本中定义了一个stop方法来终止一个线程还定义了一个suspend方法来阻塞一个线程，直到另一个线程调用resume方法。这两个方法在Java SE 1.2之后就被弃用了，因为这两种方...

我爱学Java之Thread中stop(),suspend(),resume()为什么不安全


当调用stop()方法时会发生两件事：1.即刻停止run()方法中剩余的全部工作，包括在catch或finally语句中，并抛出ThreadDeath异常(通常情况下此异常不需要显示的捕获)，因此可能...

 woigaolaoshi 2016-05-02 22:24:59  1976




禁止Java线程，弃用stop()方法，也要小心interrupt()方法

来自http://www.blogjava.net/jinfeng_wang/archive/2008/04/27/196477.html 程序是很简易的。然而，在编程人员面前，多线程呈现出了一组...


 hjiacheng 2017-06-15 11:09:28  573

Java中为什么反对使用Thread.stop, Thread.suspend, Thread.resume and Runti...

Java中为什么反对使用Thread.stop, Thread.suspend, Thread.resume and Runtime.runFinalizersOnExit?...

 loongshawn 2016-11-04 14:05:41  1696

Java Thread Stop方法以及替换实现

 longronglin 2008-11-29 11:32:00  25198



Stop方法不推荐使用，我给个具体的例子：public class DeprecatedStop extends Object implements Runnable { public void run() { while (true) { try { Thread.sleep(1000); } catch (InterruptedException e) { } } } }

Java中的线程Thread方法之---stop()

 jiangwei0910410003 2014-02-25 09:59:47  20839


搞过Java线程的人都知道，stop这个方法是臭名昭著了，早就被弃用了，但是现在任然有很多钟情与他的人，永远都放不下他，因为从他的字面意思上我们可以知道他貌似可以停止一个线程，这个需求是每个搞线程开发...

android 休眠唤醒机制分析(三) — suspend

 u012728256 2017-04-20 11:21:12  1279

前面我们分析了休眠的第一个阶段即浅度休眠，现在我们继续看休眠的第二个阶段 — 深度休眠。在深度休眠的过程中系统会首先冻结所有可以冻结的进程，然后依次挂起所有设备的电源，挂起顺序与设备注册的顺序相反，这...

Android (Linux) Suspend流程

 mirkerson 2014-03-17 18:21:09  4323

分类：Linux驱动 Android待机唤醒2013-04-23 08:48 1523人阅读 评论(0) 收藏 举报 目录(?)[+] 1. Linux Suspend简...

软件加密，无需硬件，离线可用，安全易用

主动防御,支持unity3D加密,防止代码反编译,防止资源被拷贝！



java thread的stop，suspend，resume等方法废弃的原因

如下是官方文档，先贴上，抽时间完善 Why Are Thread.stop, Thread.suspend, Thread.resume and Runtime.runFinaliz...

 gediseer 2016-12-23 09:23:30  8935

如何正确的挂起一个线程（为什么suspend被弃用）

suspend在JDK1.5时被弃用，因为它经常导致线程死锁。suspend不会破坏对象，但是如果用suspend挂起一个持有锁的线程，那么其他线程就无法进入临界区，从而导致死锁。

联系我们




请扫描二维码联系客服
✉ webmaster@csdn.net
☎ 400-660-0108
👤 QQ客服 🗣 客服论坛


关于 招聘 广告服务 百度
©1999-2018 CSDN版权所有
京ICP证09002463号

经营性网站备案信息
网络110报警服务
中国互联网举报中心
北京互联网违法和不良信息举报中心

为什么线程会放弃Suspend和Resume方法

avi9111

2016-04-25 02:34:13


2103

记录一下，多久没看youku了哦，有斗鱼，有乐视，还有翻墙，真的很久没打开优酷了 百度一下， 说什么 “suspend()方法就是将一个线程挂起(暂停)，resume()方法就是将一个挂起线程复...




5

stop和suspend 不推荐使用的原因

a314773862


2017-01-06 09:55:08

233


op () 方法作为一种粗暴的线程终止行为，在线程终止之前没有对其做任何的清除操作，因此具有固有的不安全性。 用Thread.stop()方法来终止线程将会释放该线程对象已经锁定的所有监视器。如果以前...



为什么不使用suspend和stop方法

u012036312

2015-08-27 11:22:00

1019

JDK1.0定义了stop和suspend方法： stop用来直接终止线程,suspend会阻塞线程直到另一个线程调用resume. stop和suspend 都有一些共同的点：都试图专横的控制一个...

java线程以及stop,sleep的注意


azhegps

2016-01-11 22:34:45


1261

1,线程 每一个线程都有自己运行的内容。这个内容可以称为线程要执行的任务。多线程好处：解决了多部分同时运行的问题。多线程的弊端：线程太多回到效率的降低。其实应用程序的执行都是cpu在做着快速的切换完成...

java 线程中suspend ， resume方法独占


mjlf to

2017-02-09 22:20:55


422

在使用suspend与resume方法是， 如果使用不当， 极易造成公共的同步对象的独占， 使得其他的线程无法访问公共的同步对象。 如：package com.mjlf.myBatis.thread...

java中，为什么弃用stop和suspend方法


wwyl1001

2016-02-25 14:50:41


637

一、stop和suspend方法作用 它们是Thread类的方法 stop：用来终止一个线程 suspend：用来阻塞一个线程直至另一个线程调用resume 二、弃用原因 stop天...

suspend造成死锁的例子和原因分析


magictong

2009-05-08 18:02:00


2903

今天loris分享了一个关于使用suspend造成死锁的例子，部分代码如下：UINT AFX_CDECL ThreadTest(LPVOID pvParam){ while (TRUE) { HM...

Java中如何正确的终止线程

zbw18297786698

2016-12-02 11:42:43

14813


1、Java中API自带的stop()方法，来终止线程 查阅JDK，不难发现Thread提供了一个stop()方法，但是stop()方法是一个被废弃的方法。为什么stop()方法被废弃...

联系我们




请扫描二维码联系客服

webmaster@csdn.net

400-660-0108

QQ客服 客服论坛

[关于](#) [招聘](#) [广告服务](#)  [百度](#)

©1999-2018 CSDN版权所有

京ICP证09002463号

经营性网站备案信息

网络110报警服务

中国互联网举报中心

北京互联网违法和不良信息举报中心