

cjf068

[博客](#) [微博](#) [相册](#) [收藏](#) [留言](#) [关于我](#)



cjf068

浏览: 5102 次

性别: 

来自: 北京

 我现在离线

最近访客 [更多访客>>](#)



[zhangc3086](#)



[ashal](#)



[s929498110](#)



[hongyaho](#)

文章分类

- [全部博客 \(19\)](#)
- [远程监控 \(1\)](#)
- [线程池 \(1\)](#)
- [算法 \(11\)](#)
- [面试题 \(3\)](#)
- [系统维护 \(1\)](#)
- [杂烩 \(1\)](#)
- [JMX \(1\)](#)

社区版块

- [我的资讯](#) (0)
- [我的论坛](#) (44)
- [我解决的问题](#) (0)

存档分类

- [2012-03](#) (10)
- [2012-02](#) (9)

[使用JMX 监控类的实例数量](#)

博客分类: [JMX](#)

近日遇到一个JVM内存溢出问题，因为问题不好查询，遂想到通过JMX来监控实例纵多的BEAN类。

- 1.定义基础Bean类，该类就做两件事：
  - (1)保存一个统计子类实例数的列表
  - (2)在默认构造器中，对相应的子类计数值加1
  - (3)在finalize方法中，对回收的实例对应的计数值减1
- 2.需要监控的Bean类，都要继承该父类，且必须提供默认构造器
- 3.定义JMX 的MBean，提供实例数量查询功能

我的简单实现如下：

基础Bean：StaBaseBean

Java代码



```
1. package org.jf.beans;
2.
3. import java.util.ArrayList;
4. import java.util.Collections;
5. import java.util.HashMap;
6. import java.util.List;
7. import java.util.Map;
8.
9. /**
10.  *
11.  * 具有统计功能的Bean
12.  * @author junfeng.chen
13.  *
14.  */
15. public class StaBaseBean
16. {
17.
18.     private static Map<String,Integer> sta_map = Collections.synchronizedMap(new HashMap<String,Integer>());
19.
20.
21.     /**
22.      *
23.      * @param obj
24.      * @return
25.      */
26.     public static int getInstanceCount(Object obj)
27.     {
28.
29.         Class clazz = obj.getClass();
30.         String class_name = clazz.getName();
31.         Integer ints = sta_map.get(class_name);
32.         if(ints!=null)
33.             return ints.intValue();
34.         return 0;
```

■ [更多存档...](#)

评论排行榜

■ [大数乘法](#)

■ [求最小的k个数问题](#)

■ [java C/S结构系统远程监控RM-beta1](#)

最新评论

[cjf068](#): LuckYes 写道楼主如果用最小堆的话, 最好用调整堆的方式来 ...

[求最小的k个数问题](#)

[LuckYes](#): 楼主如果用最小堆的话, 最好用调整堆的方式来构建堆, 这样效率更高 ...

[求最小的k个数问题](#)

[cjf068](#): 这个算法的基本思路, 是小学3年级的 算法, 就是简单的把乘法运算 ...

[大数乘法](#)

[liujunsong](#): 这个算法的基本思路, 是小学3年级的 算法, 就是简单的把乘法运算 ...


[大数乘法](#)

[shuidexiongdi](#): 去年我也写了一个<http://shuidexiongdi.it> ...

[大数乘法](#)

```
35.     }
36.
37.     public static int getInstanceCount(String class_name)
38.     {
39.
40.         Integer ints = sta_map.get(class_name);
41.         if(ints!=null)
42.             return ints.intValue();
43.         return 0;
44.     }
45.
46.     public StaBaseBean()
47.     {
48.         Class clazz = this.getClass();
49.         Integer ints = sta_map.get(clazz.getName());
50.         if(ints == null)
51.             ints = new Integer(0);
52.         ints = ints + 1;
53.         sta_map.put(clazz.getName(), ints);
54.         System.out.println(clazz.getName()+" -create-"+ints);
55.     }
56.
57.
58.     public static Map<String,Integer> getInstanceCountMap()
59.     {
60.         Map<String,Integer> map1 = new HashMap<String,Integer>();
61.         List<String> keys = new ArrayList<String>();
62.         keys.addAll(sta_map.keySet());
63.         for(String key:keys)
64.         {
65.             map1.put(key, sta_map.get(key));
66.         }
67.         return map1;
68.     }
69.
70.     @Override
71.     public void finalize()
72.     {
73.         Class clazz = this.getClass();
74.         Integer ints = sta_map.get(clazz.getName());
75.         if(ints!=null)
76.             ints = ints - 1;
77.         if(ints !=null && ints>=0)
78.             sta_map.put(clazz.getName(), ints);
79.         System.out.println(clazz.getName()+" -garbeg-"+ints);
80.         //      System.out.println(clazz.getName()+" count = "+ints);
81.     }
82. }
```


实例数量查询MBean接口InstanceCounterMBean

Java代码 

```
1.     package org.jf.jmx.beans;
2.
3.     public interface InstanceCounterMBean {
4.
5.         public int getInstanceCount(String class_name);
6.
7.     }
```

实例数量查询MBean实现类:

InstanceCounter

Java代码 

```
1. package org.jf.jmx.beans;
2.
3. import java.util.Map;
4.
5. import org.jf.beans.StaBaseBean;
6.
7. public class InstanceCounter implements InstanceCounterMBean
8. {
9.
10.     @Override
11.     public int getInstanceCount(String class_name)
12.     {
13.         return StaBaseBean.getInstanceCount(class_name);
14.     }
15.
16.     public Map<String,Integer> getInstanceCountMap()
17.     {
18.         return StaBaseBean.getInstanceCountMap();
19.     }
20.
21. }
```

应用程序启动以后，即可通过jmx代理，连接应用程序，通过jmx代码查看MBean并调用其方法，如本实现中，通过Jconsole连接应用程序，找到对应的mbean，调用getInstanceCount即可查询当前内存中指定类的实例数

[src.zip](#) (3.6 KB)

下载次数: 36

[查看图片附件](#)

分享到:  

[常用地址](#) ►

2012-03-20 20:31:50 | 浏览 35 | [评论\(0\)](#) | 分类: [编程语言](#) | [相关推荐](#) 

评论

发表评论



[您还没有登录,请您登录后再发表评论](#)