# **Java线程状态中BLOCKED和WAITING有什么区别？**



[小南家的青蛙](https://blog.csdn.net/babyblue_963" \t "https://blog.csdn.net/jackie_xiaonan/article/details/_blank) 2015-08-23 17:48:40 IMG_257 23198 IMG_258 收藏 8

分类专栏： [Java](https://blog.csdn.net/babyblue_963/category_1327818.html" \t "https://blog.csdn.net/jackie_xiaonan/article/details/_blank) 文章标签： [Java](https://www.csdn.net/tags/NtTaIg5sMzYyLWJsb2cO0O0O.html" \t "https://blog.csdn.net/jackie_xiaonan/article/details/_blank) [Thread State](https://so.csdn.net/so/search/s.do?q=Thread State&t=blog&o=vip&s=&l=&f=&viparticle=" \t "https://blog.csdn.net/jackie_xiaonan/article/details/_blank)

版权

刚才在看CSDN的问答时，发现这个问题。原问题的作者是在观察jstack的输出时提出的疑问，那么BLOCKED和WAITING有什么区别呢？  
答复在JDK源码中可以找到，如下是java.lang.Thread.State类的一部分注释。

/\*\*  
\* Thread state for a thread blocked waiting for a monitor lock.  
\* A thread in the blocked state is waiting for a monitor lock  
\* to enter a synchronized block/method or  
\* reenter a synchronized block/method after calling  
\* {@link Object#wait() Object.wait}.  
\*/  
BLOCKED,

/\*\*  
\* Thread state for a waiting thread.  
\* A thread is in the waiting state due to calling one of the  
\* following methods:  
\* {@link Object#wait() Object.wait} with no timeout  
\* {@link #join() Thread.join} with no timeout  
\* {@link LockSupport#park() LockSupport.park}  
\*  
\*  
\* A thread in the waiting state is waiting for another thread to  
\* perform a particular action.  
\*  
\* For example, a thread that has called Object.wait()  
\* on an object is waiting for another thread to call  
\* Object.notify() or Object.notifyAll() on  
\* that object. A thread that has called Thread.join()  
\* is waiting for a specified thread to terminate.  
\*/  
WAITING,

从中可以清晰的得到线程处于BLOCKED和WAITING状态的场景。

## **BLOCKED状态**

线程处于BLOCKED状态的场景。

* 当前线程在等待一个monitor lock，比如等待执行synchronized代码块或者使用synchronized标记的方法。
* 在synchronized块中循环调用Object类型的wait方法，如下是样例  
  synchronized(this)  
  {  
  while (flag)  
  {  
  obj.wait();  
  }  
  // some other code  
  }

## **WAITING状态**

线程处于WAITING状态的场景。

* 调用Object对象的wait方法，但没有指定超时值。
* 调用Thread对象的join方法，但没有指定超时值。
* 调用LockSupport对象的park方法。

提到WAITING状态，顺便提一下TIMED\_WAITING状态的场景。

## **TIMED\_WAITING状态**

线程处于TIMED\_WAITING状态的场景。

* 调用Thread.sleep方法。
* 调用Object对象的wait方法，指定超时值。
* 调用Thread对象的join方法，指定超时值。
* 调用LockSupport对象的parkNanos方法。
* 调用LockSupport对象的parkUntil方法。

欢迎访问Jackie的家，http://jackieathome.sinaapp.com/，如需转载文章，请注明出处。