

- - -

Problems Detected



Our **machine learning (ML)** algorithms have detected problems in your application which may cause **application unresponsiveness**. To see the identified problems and recommended solutions, please [subscribe to our plan](#) (ft-subscription.jsp).

Select Plan (ft-subscription.jsp)

Thread Count Summary

(To learn about different thread states through real-life example, check out this [video tutorial](#) (https://www.youtube.com/watch?v=fzYLtYaJ_D0))

342
BLOCKED

[View Details](#)

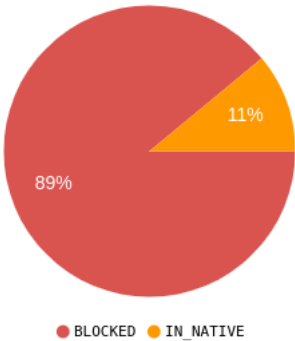
(same-state-threads.jsp?state=BLOCKED&dumpId=1)

42
IN_NATIVE

[View Details](#)

(same-state-threads.jsp?state=IN_NATIVE&dumpId=1)

Thread state %

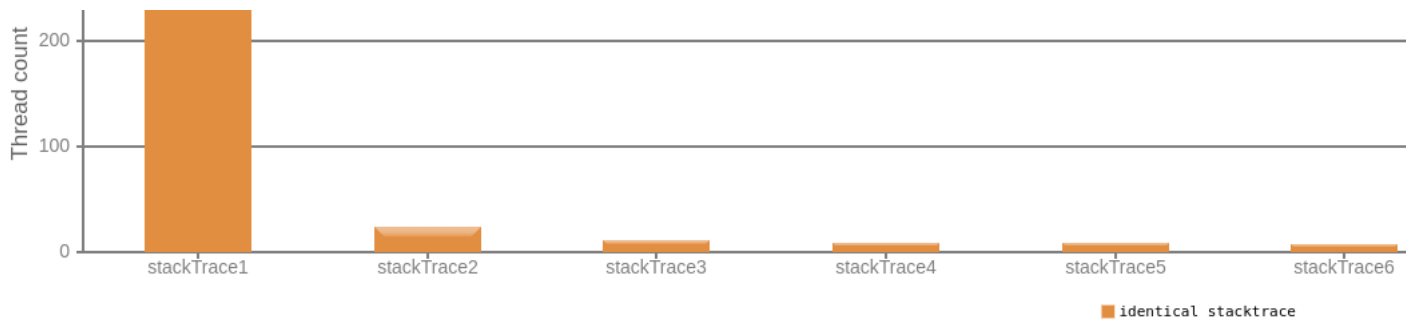


Total Threads count: **384**

Threads with identical stack trace

(Threads with identical stack traces are grouped here. If lot of threads start to exhibit identical stack trace it might be a concern, learn [RSI Pattern](#) (https://blog.fastthread.io/2016/02/22/thread-dump-analysis-pattern-repetitive-strain-injury-rsi/))





Thread Count

Identical Stack trace

266 BLOCKED threads	<ul style="list-style-type: none"> - sun.misc.Unsafe.park(boolean, long) @bci=0 (Compiled frame; information may be imprecise) - java.util.concurrent.locks.LockSupport.parkNanos(java.lang.Object, long) @bci=20, line=215 (Compiled frame) - java.util.concurrent.locks.AbstractQueuedSynchronizer\$ConditionObject.await(long, java.util.concurrent.TimeUnit) @bci=97,... - org.apache.tez.runtime.library.common.shuffle.impl.ShuffleManager\$RunShuffleCallable.callInternal() @bci=125, line=327 (...) - org.apache.tez.runtime.library.common.shuffle.impl.ShuffleManager\$RunShuffleCallable.callInternal() @bci=1, line=311 (Co... <p>...</p> <p>See complete stacktrace. (threads.jsp?tracel=22&pc&dumpld=1&type=rs)</p>
24 IN_NATIVE threads	<ul style="list-style-type: none"> - sun.nio.ch.EPollArrayWrapper.epollWait(long, int, long, int) @bci=0 (Compiled frame; information may be imprecise) - sun.nio.ch.EPollArrayWrapper.poll(long) @bci=18, line=269 (Compiled frame) - sun.nio.ch.EPollSelectorImpl.doSelect(long) @bci=28, line=93 (Compiled frame) - sun.nio.ch.SelectorImpl.lockAndDoSelect(long) @bci=37, line=86 (Compiled frame) - sun.nio.ch.SelectorImpl.select(long) @bci=30, line=97 (Compiled frame) <p>...</p> <p>See complete stacktrace. (threads.jsp?tracel=15&pc&dumpld=1&type=rs)</p>
11 BLOCKED threads	<ul style="list-style-type: none"> - java.lang.Object.wait(long) @bci=0 (Compiled frame; information may be imprecise) - org.apache.hadoop.ipc.Client\$Connection.waitForWork() @bci=59, line=1018 (Compiled frame) - org.apache.hadoop.ipc.Client\$Connection.run() @bci=57, line=1062 (Compiled frame) <p>...</p> <p>See complete stacktrace. (threads.jsp?tracel=10&pc&dumpld=1&type=rs)</p>
9 BLOCKED threads	<ul style="list-style-type: none"> - sun.misc.Unsafe.park(boolean, long) @bci=0 (Compiled frame; information may be imprecise) - java.util.concurrent.locks.LockSupport.parkNanos(java.lang.Object, long) @bci=20, line=215 (Compiled frame) - java.util.concurrent.SynchronousQueue\$TransferStack.awaitFulfill(java.util.concurrent.SynchronousQueue\$TransferStack\$S... - java.util.concurrent.SynchronousQueue\$TransferStack.transfer(java.lang.Object, boolean, long) @bci=102, line=362 (Compil... - java.util.concurrent.SynchronousQueue.poll(long, java.util.concurrent.TimeUnit) @bci=11, line=941 (Compiled frame) <p>...</p> <p>See complete stacktrace. (threads.jsp?tracel=5&pc&dumpld=1&type=rs)</p>
9 BLOCKED threads	<ul style="list-style-type: none"> - sun.misc.Unsafe.park(boolean, long) @bci=0 (Compiled frame; information may be imprecise) - java.util.concurrent.locks.LockSupport.park(java.lang.Object) @bci=14, line=175 (Compiled frame) - java.util.concurrent.locks.AbstractQueuedSynchronizer\$ConditionObject.await() @bci=42, line=2039 (Compiled frame) - java.util.concurrent.LinkedBlockingQueue.take() @bci=29, line=442 (Compiled frame) - java.util.concurrent.ThreadPoolExecutor.getTask() @bci=149, line=1074 (Compiled frame) <p>...</p> <p>See complete stacktrace. (threads.jsp?tracel=30&pc&dumpld=1&type=rs)</p>
7 BLOCKED threads	<ul style="list-style-type: none"> - sun.misc.Unsafe.park(boolean, long) @bci=0 (Compiled frame; information may be imprecise) - java.util.concurrent.locks.LockSupport.parkNanos(java.lang.Object, long) @bci=20, line=215 (Compiled frame) - java.util.concurrent.locks.AbstractQueuedSynchronizer\$ConditionObject.awaitNanos(long) @bci=78, line=2078 (Compiled fr... - java.util.concurrent.LinkedBlockingQueue.poll(long, java.util.concurrent.TimeUnit) @bci=62, line=467 (Compiled frame) - org.apache.hadoop.ipc.CallQueueManager.take() @bci=22, line=287 (Compiled frame) <p>...</p> <p>See complete stacktrace. (threads.jsp?tracel=37&pc&dumpld=1&type=rs)</p>
6 BLOCKED threads	<ul style="list-style-type: none"> - java.lang.Thread.sleep(long) @bci=0 (Compiled frame; information may be imprecise) - org.apache.hadoop.hdfs.client.impl.LeaseRenewer.run(int) @bci=478, line=411 (Compiled frame) - org.apache.hadoop.hdfs.client.impl.LeaseRenewer.access\$600(org.apache.hadoop.hdfs.client.impl.LeaseRenewer, int) @bci... - org.apache.hadoop.hdfs.client.impl.LeaseRenewer\$1.run() @bci=69, line=307 (Compiled frame) - java.lang.Thread.run() @bci=11, line=748 (Compiled frame) <p>...</p> <p>See complete stacktrace. (threads.jsp?tracel=7&pc&dumpld=1&type=rs)</p>
5 BLOCKED threads	<p>stacktrace</p> <p>See complete stacktrace. (threads.jsp?tracel=1&pc&dumpld=1&type=rs)</p>
5 BLOCKED threads	<ul style="list-style-type: none"> - sun.misc.Unsafe.park(boolean, long) @bci=0 (Compiled frame; information may be imprecise) - java.util.concurrent.locks.LockSupport.parkNanos(java.lang.Object, long) @bci=20, line=215 (Compiled frame) - java.util.concurrent.locks.AbstractQueuedSynchronizer\$ConditionObject.awaitNanos(long) @bci=78, line=2078 (Compiled fr... - java.util.concurrent.ScheduledThreadPoolExecutor\$DelayedWorkQueue.take() @bci=124, line=1093 (Compiled frame) - java.util.concurrent.ScheduledThreadPoolExecutor\$DelayedWorkQueue.take() @bci=1, line=809 (Compiled frame) <p>...</p> <p>See complete stacktrace. (threads.jsp?tracel=14&pc&dumpld=1&type=rs)</p>

4 BLOCKED threads	<div><div>- sun.misc.Unsafe.park(boolean, long) @bci=0 (Compiled frame; information may be imprecise)</div><div>- java.util.concurrent.locks.LockSupport.park(java.lang.Object) @bci=14, line=175 (Compiled frame)</div><div>- java.util.concurrent.SynchronousQueue\$TransferStack.awaitFulfill(java.util.concurrent.SynchronousQueue\$TransferStack\$S...</div><div>- java.util.concurrent.SynchronousQueue\$TransferStack.transfer(java.lang.Object, boolean, long) @bci=102, line=362 (Compil...</div><div>- java.util.concurrent.SynchronousQueue.take() @bci=7, line=924 (Compiled frame)</div><div>...</div><div>See complete stacktrace. (threads.jsp?traceld=21&pc&dumpld=1&type=rs)</div></div>
-------------------	---

Most used methods

(Frequently executed methods are reported. If lot of threads executes same method, it may be a concern. Learn [All roads lead to Rome pattern](#) (<https://blog.fastthread.io/2016/02/01/thread-dump-analysis-pattern-all-roads-lead-to-rome/>))

Thread Count	Method	Percentage
308 threads	<div>sun.misc.Unsafe.park(boolean, long).</div> <div>To see stack trace click here. (threads.jsp?traceld=3&dumpld=1&type=tm)</div>	80%
37 threads	<div>sun.nio.ch.EPollArrayWrapper.epollWait(long, int, long, int).</div> <div>To see stack trace click here. (threads.jsp?traceld=2&dumpld=1&type=tm)</div>	10%
21 threads	<div>java.lang.Object.wait(long).</div> <div>To see stack trace click here. (threads.jsp?traceld=5&dumpld=1&type=tm)</div>	5%
8 threads	<div>java.lang.Thread.sleep(long).</div> <div>To see stack trace click here. (threads.jsp?traceld=1&dumpld=1&type=tm)</div>	2%
2 threads	<div>java.net.SocketInputStream.socketRead0(java.io.FileDescriptor, byte[], int, int, int).</div> <div>To see stack trace click here. (threads.jsp?traceld=8&dumpld=1&type=tm)</div>	1%

[Show all methods >>](#)

CPU consuming threads

(If application is consuming high CPU, investigate below threads. Learn [Athlete pattern](#) (<https://blog.fastthread.io/2016/06/23/really-running/>))

☒ None

Blocking Threads - Transitive Graph

(Threads that block other threads are displayed here. Blocking threads makes application unresponsive, learn [Traffic Jam pattern](#) (<https://blog.fastthread.io/2016/03/07/thread-dump-analysis-pattern-ripple-effect/>))

☒ No transitive blocks found

GC Threads

(Garbage collection threads count reported. Learn [Scavengers pattern](#) (<https://blog.fastthread.io/2015/09/02/thread-dump-analysis-pattern-several-scavengers/>))

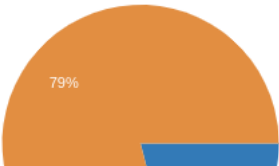
Not reported in the thread dump

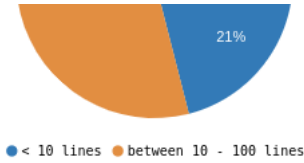
Threads Stack Length

(Lengthy stacks can cause StackOverflowError. [Learn more](#) (<https://blog.fastthread.io/2018/09/24/stackoverflowerror/>))

☒ No Problem in Stack trace length.

Stack Length	Thread count
< 10 lines (same-state-threads.jsp?state=length&pc&length=PCaXMCBsav)	81
between 10 - 100 lines (same-state-threads.jsp?state=length&pc&length=)	303





Complex DeadLocks

(Learn more about [Complex Deadlock](https://blog.fastthread.io/2016/05/18/circular-deadlock/) (https://blog.fastthread.io/2016/05/18/circular-deadlock/))

✓ No Complex Deadlocks found

Dead Lock

(Learn more about [Deadlock](https://blog.fastthread.io/2016/04/25/deadlock/) (https://blog.fastthread.io/2016/04/25/deadlock/))

✓ No Deadlock found

Finalizer Thread

(If finalizer thread is BLOCKED or WAITING for a prolonged period, it can result in OutOfMemoryError, to learn more visit [Leprechaun Trap pattern](https://blog.fastthread.io/2015/11/20/thread-dump-analysis-pattern-leprechaun-trap/) (https://blog.fastthread.io/2015/11/20/thread-dump-analysis-pattern-leprechaun-trap/))

✓ No problem with Finalizer Thread.

Exception

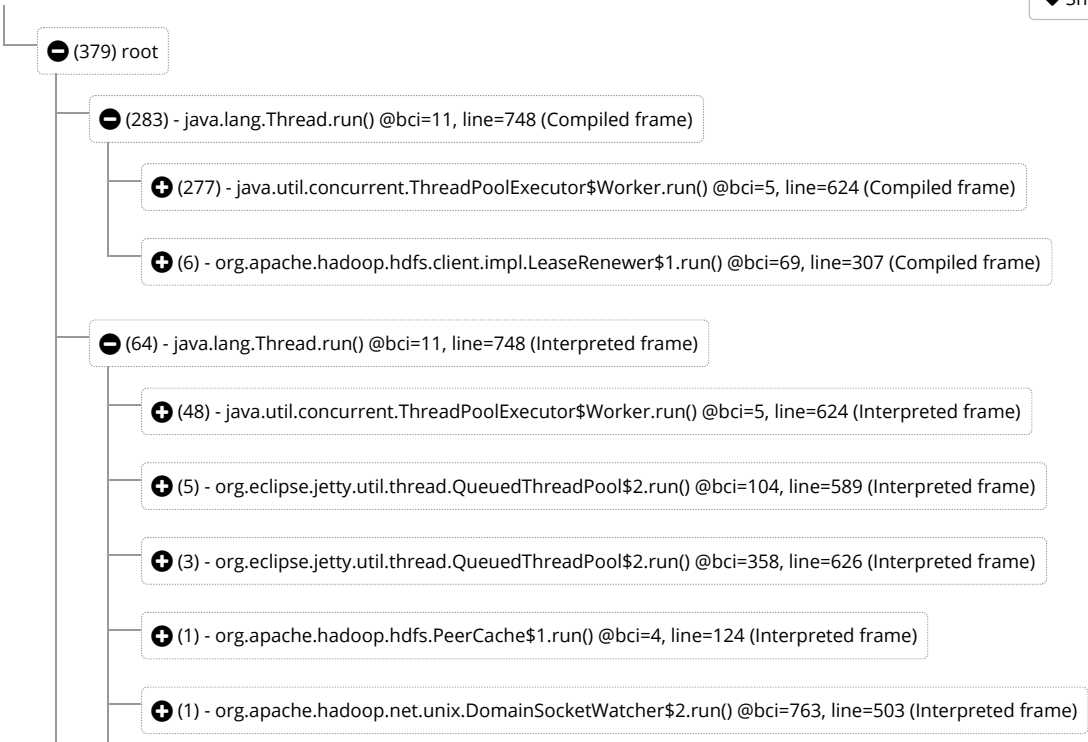
(If threads throwing commonly known Exceptions/Errors are reported here. [Learn more](https://blog.fastthread.io/2020/06/10/threads-throwing-exception/) (https://blog.fastthread.io/2020/06/10/threads-throwing-exception/))

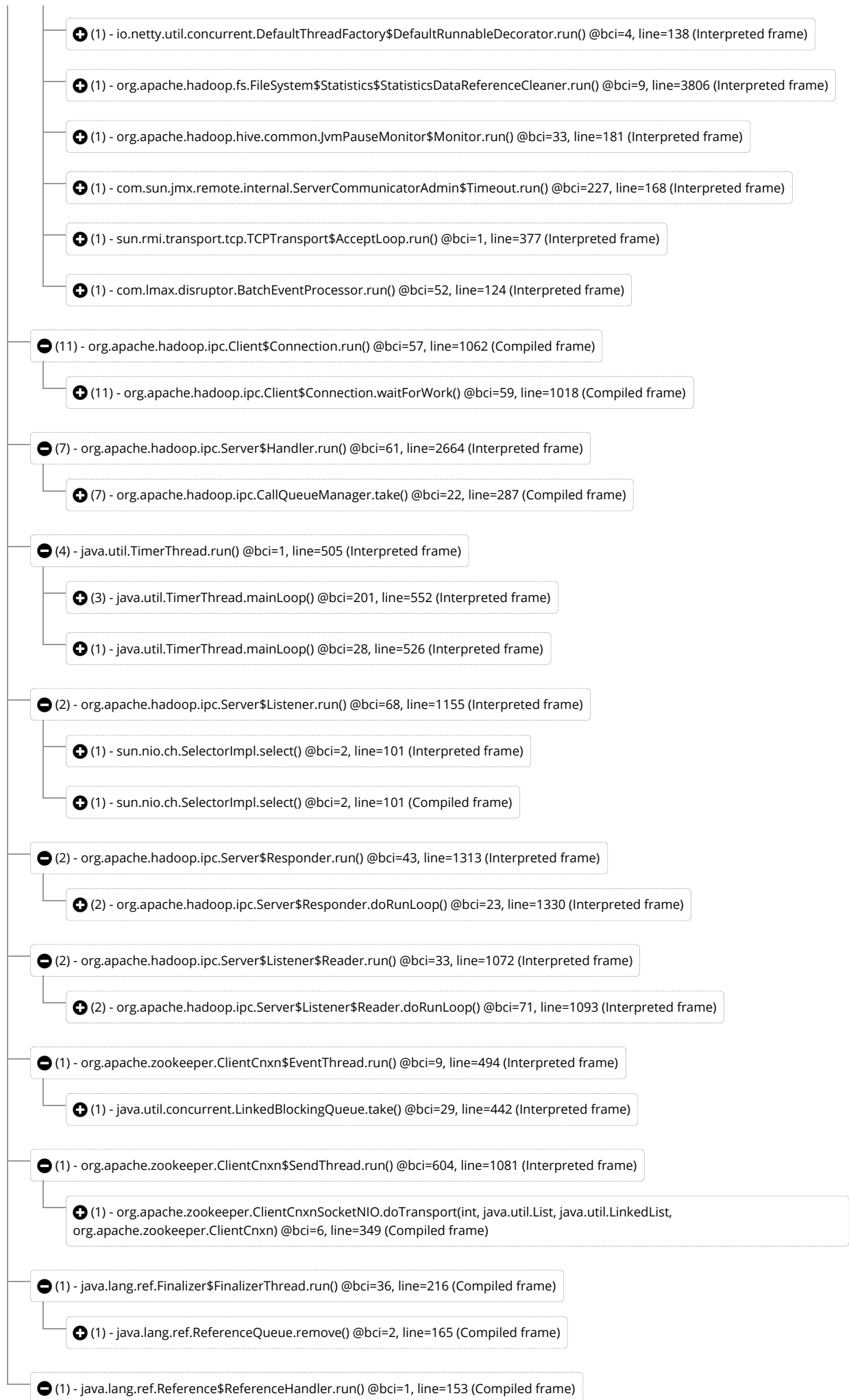
✓ No known exceptions are reported.

Bottom up Call Stack Tree

(All threads stacktrace are combined in to one single tree. Learn [it's benefits](https://blog.fastthread.io/2018/04/27/benefits-of-call-stack-tree/) (https://blog.fastthread.io/2018/04/27/benefits-of-call-stack-tree/).)

↓ Show Top Down call stack





⊕ (1) - java.lang.ref.Reference.tryHandlePending(boolean) @bci=54, line=191 (Compiled frame)