Managing the Java Thread Lifecycle: Steps Involved in Starting a Thread

Douglas C. Schmidt
d.schmidt@vanderbilt.edu
www.dre.vanderbilt.edu/~schmidt

Institute for Software Integrated Systems
Vanderbilt University
Nashville, Tennessee, USA
Learning Objectives in this Lesson

• Understand the layers involved in start a Java thread
• Recognize the steps involved in starting a Java thread
Steps Involved in Starting a Java Thread
Steps Involved in Starting a Java Thread

- The following steps are involved when starting a Java thread on the Android open-source platform

See [source.android.com](http://source.android.com)
Steps Involved in Starting a Java Thread

• The following steps are involved when starting a Java thread on the Android open-source platform

1. `myThread.start()`
Steps Involved in Starting a Java Thread

• The following steps are involved when starting a Java thread on the Android open-source platform

1. `myThread.start()`
2. `Thread.start()` // Java method

See `libcore/luni/src/main/java/java/lang/Thread.java`
Steps Involved in Starting a Java Thread

• The following steps are involved when starting a Java thread on the Android open-source platform

1. myThread.start()
2. Thread.start()
3. VMThread.create() // Native method

See libcore/luni/src/main/java/java/lang/VMThread.java
Steps Involved in Starting a Java Thread

- The following steps are involved when starting a Java thread on the Android open-source platform:

1. `myThread.start()`
2. `Thread.start()`
3. `VMThread.create()`
4. `Dalvik_java_lang_VMThread_create()`
   // JNI method

See `dalvik/vm/native/java_lang_VMThread.cpp`
Steps Involved in Starting a Java Thread

- The following steps are involved when starting a Java thread on the Android open-source platform:

1. `myThread.start()`
2. `Thread.start()`
3. `VMThread.create()`
4. `Dalvik_java_lang_VMThread_create()` // Dalvik method
5. `dvmCreateInterpThread()` // Dalvik method

See `dalvik/vm/Thread.cpp`
Steps Involved in Starting a Java Thread

- The following steps are involved when starting a Java thread on the Android open-source platform:

1. `myThread.start()`
2. `Thread.start()`
3. `VMThread.create()`
4. `Dalvik_java_lang_VMThread_create()`
5. `dvmCreateInterpThread()`
6. `pthread_create(...)`, `interpThreadStart`
   // Pthreads method

See [bionic/libc/bionic/pthread.c](bionic/libc/bionic/pthread.c)
Steps Involved in Starting a Java Thread

- The following steps are involved when starting a Java thread on the Android open-source platform

1. `myThread.start()`
2. `Thread.start()`
3. `VMThread.create()`
4. `Dalvik_java_lang_VMThread_create()`
5. `dvmCreateInterpThread()`
6. `pthread_create(..., interpThreadStart) // Pthreads method`

This is the entry point function used to transition between C & Java code
Steps Involved in Starting a Java Thread

• The following steps are involved when starting a Java thread on the Android open-source platform

1. myThread.start()
2. Thread.start()
3. VMThread.create()
4. Dalvik_java_lang_VMThread_create()
5. dvmCreateInterpThread()
6. pthread_create(..., interpThreadStart)
7. Android Linux kernel...

See [source.android.com/source/building-kernels.html](source.android.com/source/building-kernels.html)
Steps Involved in Starting a Java Thread

- The following steps are involved when starting a Java thread on the Android open-source platform:

1. `myThread.start()`
2. `Thread.start()`
3. `VMThread.create()`
4. `Dalvik_java_lang_VMThread_create()`
5. `dvmCreateInterpThread()`
6. `pthread_create(..., interpThreadStart)`
7. `Android Linux kernel...`
8. `interpThreadStart(void* arg) // Adapter`
Steps Involved in Starting a Java Thread

- The following steps are involved when starting a Java thread on the Android open-source platform

1. myThread.start()
2. Thread.start()
3. VMThread.create()
4. Dalvik_java_lang_VMThread_create()
5. dvmCreateInterpThread()
6. pthread_create(..., interpThreadStart)
7. Android Linux kernel...
8. interpThreadStart(void* arg)
9. dvmCallMethod(self, run, self->threadObj)
   // Dalvik method

See [dalvik/vm/interp/Stack.cpp](dalvik/vm/interp/Stack.cpp)
Steps Involved in Starting a Java Thread

1. The following steps are involved when starting a Java thread on the Android open-source platform

1. `myThread.start()`
2. `Thread.start()`
3. `VMThread.create()`
4. `Dalvik_java_lang_VMThread_create()`
5. `dvmCreateInterpThread()`
6. `pthread_create(..., interpThreadStart)`
7. `Android Linux kernel...`
8. `interpThreadStart(void* arg)`
9. `dvmCallMethod(self, run, self->threadObj)`
10. `MyThread.run()` // User-defined hook method
End of Managing the Java Thread Lifecycle: Steps Involved in Starting a Thread