NDK Renderscript 64 bit is not supported

+1 23 Hotlists (6) Mark as Duplicate

Comments (13) Dependencies Duplicates (0) Blocking (0) Resources (6)

Fixed Bug P3 + [AOSP] assigned [AOSP] Released Migrated-Tools-NDK

STATUS UPDATE No update yet. Edit

DESCRIPTION ra...@gmail.com created issue #1

Hi dear NDK team,

I found that Renderscript 64 bit is not supported in the newest version android-ndk-r10d-xxx.
I got compiling error:

```
[arm64-v8a] Compile RS     : hellocomputendk <= mono.rs
cd /home/jerry/Tools/android-ndk-r10d/samples/HelloComputeNDK/jni/ &&
/home/jerry/Tools/android-ndk-r10d/toolchains/renderscript/prebuilt/linux-x86_64/bin/llvm-rs-cc
-o /home/jerry/Tools/android-ndk-r10d/samples/HelloComputeNDK/obj/local/arm64-v8a/objs/hellocomputendk/
-d /home/jerry/Tools/android-ndk-r10d/samples/HelloComputeNDK/obj/local/arm64-v8a/objs/hellocomputendk -MD -reflect-c++
-I/home/jerry/Tools/android-ndk-r10d/toolchains/renderscript/prebuilt/linux-x86_64/lib/clang/3.5/include
-I/home/jerry/Tools/android-ndk-r10d/platforms/android-21/arch-arm64/usr/include/rs/scriptc
-I/home/jerry/Tools/android-ndk-r10d/samples/HelloComputeNDK/jni -Wall -Werror -m64    mono.rs
/home/jerry/Tools/android-ndk-r10d/toolchains/renderscript/prebuilt/linux-x86_64/bin/bcc_compat -O3
-o /home/jerry/Tools/android-ndk-r10d/samples/HelloComputeNDK/obj/local/arm64-v8a/objs/hellocomputendk/mono.bc.o -fPIC -shared
-rt-path /home/jerry/Tools/android-ndk-r10d/platforms/android-21/arch-arm64/usr/lib/rs/libclcore.bc
-mtriple arm-none-linux-gnueabi  /home/jerry/Tools/android-ndk-r10d/samples/HelloComputeNDK/obj/local/arm64-v8a/objs/hellocomputendk/mono.bc &&
/home/jerry/Tools/android-ndk-r10d/toolchains/aarch64-linux-android-4.9/prebuilt/linux-x86_64/bin/aarch64-linux-android-g++ -shared -Wl,-soname,librs.mono.so
-nostdlib /home/jerry/Tools/android-ndk-r10d/samples/HelloComputeNDK/obj/local/arm64-v8a/objs/hellocomputendk/mono.bc.o
/home/jerry/Tools/android-ndk-r10d/platforms/android-21/arch-arm64/usr/lib/rs/libcompiler_rt.a
-o /home/jerry/Tools/android-ndk-r10d/samples/HelloComputeNDK/libs/arm64-v8a/librs.mono.so
-L /home/jerry/Tools/android-ndk-r10d/platforms/android-21/arch-arm64/usr/lib
-L /home/jerry/Tools/android-ndk-r10d/platforms/android-21/arch-arm64/usr/lib/rs -no-canonical-prefixes -lRSSupport -lm -lc &&
/home/jerry/Tools/android-ndk-r10d/toolchains/aarch64-linux-android-4.9/prebuilt/linux-x86_64/bin/aarch64-linux-android-g++ -MMD -MP -MF
/home/jerry/Tools/android-ndk-r10d/samples/HelloComputeNDK/obj/local/arm64-v8a/objs/hellocomputendk/mono.o.d -fpic -ffunction-sections
-funwind-tables -fstack-protector -no-canonical-prefixes -fno-exceptions -fno-rtti -O2 -g -DNDEBUG -fomit-frame-pointer -fstrict-aliasing -funswitch-loops -finline-limit=300
-I/home/jerry/Tools/android-ndk-r10d/platforms/android-21/arch-arm64/usr/include/rs/cpp
-I/home/jerry/Tools/android-ndk-r10d/platforms/android-21/arch-arm64/usr/include/rs
-I/home/jerry/Tools/android-ndk-r10d/samples/HelloComputeNDK/obj/local/arm64-v8a/objs/hellocomputendk
-I/home/jerry/Tools/android-ndk-r10d/sources/cxx-stl/stlport/stlport
-I/home/jerry/Tools/android-ndk-r10d/sources/cxx-stl//gabi++/include
-I/home/jerry/Tools/android-ndk-r10d/samples/HelloComputeNDK/jni -DANDROID  -Wa,--noexecstack -Wformat -Werror=format-security  -frtti
-I/home/jerry/Tools/android-ndk-r10d/platforms/android-21/arch-arm64/usr/include -fno-rtti
-c  /home/jerry/Tools/android-ndk-r10d/samples/HelloComputeNDK/obj/local/arm64-v8a/objs/hellocomputendk/./ScriptC_mono.cpp
-o /home/jerry/Tools/android-ndk-r10d/samples/HelloComputeNDK/obj/local/arm64-v8a/objs/hellocomputendk/mono.o
bcc_compat: Not enough positional command line arguments specified!
```

I also found that the android-ndk-r10d/platforms/android-21/arch-arm64/usr/lib/rs/libRSSupport.so and libRScpp_static.a are actually 32-bit libraries!

Then when will Renderscript 64-bit supported, will it be soon? Thank you! :)

✓ Links (5)

"http://mono.rs"
"http://librs.mono.so"
"See https://code.google.com/p/android/issues/detail?id=182…"
"1. get the prebuilts of "librsjni.so" and "libRSSupport.so" from here https://android.googlesource.com/platform/prebuilts/sdk/+/602e78a5c5745f49910a1c4546946b1e912c466b/renderscript/lib/arm
"https://android.googlesource.com/platform/prebuilts/sdk/+/android-6.0.0_r13/re… "

COMMENTS All comments

sr...@google.com <sr...@google.com> #2

There is no current support for a 64-bit RS compatibility library. It is something that we hope to add in the future. We also did not enable 64-bit native RenderScript in this NDK release, althou
in a future release.

ra...@gmail.com <ra...@gmail.com> #3

Dear Stephen Hines, thank you very much for your reply and, I appreciate your hard working. :)

en...@google.com <en...@google.com>

Assigned to sr...@google.com.

sr...@google.com <sr...@google.com>

Reassigned to je...@google.com.

---

ch...@orr.me.uk <ch...@orr.me.uk> #4 Se

Is this planned to be supported soon?

I just got this crash report from a SM-G925F (Galaxy S6 Edge) running Android 5.1.1, with an app which is using the Renderscript support library:

Caused by: android.support.v8.renderscript.RSRuntimeException: Error loading RS jni library: java.lang.UnsatisfiedLinkError: dalvik.system.PathClassLoader[DexPathList[[zip file "/data/app/c
1/base.apk"],nativeLibraryDirectories=[/data/app/com.example.foo-1/lib/arm64, /vendor/lib64, /system/lib64]]] couldn't find "librsjni.so"

(Though shouldn't newer devices have Renderscript built in, and not need to fall back to the support library?)

---

mi...@google.com <mi...@google.com> #5 Se

Yeah, we are working on that.

(For your question, actually the support lib will first load librsjni.so and then decided whether to use the native RS on device or the support lib runtime)

[Deleted User] <[Deleted User]> #6 Oc

#9 If you are using Build Tools 23.0.1 try using 22.0.1, there appears to be a build issue with RenderScript. Our app has the same stack trace problem on the S6 Edge and it was due to the bu
fine when I build with 22.0.1)

See https://code.google.com/p/android/issues/detail?id=182356

---

mi...@google.com <mi...@google.com> #7 Oc

Current buildTools does not bundle 64bit RenderScript, current workaround would be keeping RenderScript related libs (librsjni.so, libRSSupport.so, and librs.*.so) in your lib/arm (or armv7) f
32bit RenderScript).

---

ch...@orr.me.uk <ch...@orr.me.uk> #8 Oc

Stupid question, perhaps, but could you explain exactly what you mean by using "lib/arm" (or "lib/armv7")?

I'm familiar with bundling native libraries in "src/main/jniLibs/armeabi(-v7a)?", but not with just "arm" or "armv7".

In any case, I'm not explicitly including the RenderScript binaries — I'm including the RenderScript support library simply via the `renderscriptTargetApi` and `renderscriptSupportModeEnablec
i.e. the "librsjni.so", "libRSSupport.so" and "libblasV8.so" files (for armeabi-v7a, mips & x86) are included in my APK automatically.

So I'm not clear on what the workaround is that you're trying to describe.

Anyway, I think I see what my particular problem is: my APK includes *other* native libraries, which *do* provide "arm64-v8a" (and "x86_64") binaries.

This therefore seems to make the app run in 64-bit mode, rather that in 32-bit compatibility mode, causing the app to search explicitly for 64-bit RenderScript libraries, and ignores the existing

Unfortunately, I don't have a Galaxy S6 to test this on...

---

mi...@google.com <mi...@google.com> #9 Oc

#13 Yeah, you are right. I meant jniLibs/armeabi(-v7a).

I see. One way to workaround it:
1. get the prebuilts of "librsjni.so" and "libRSSupport.so" from here https://android.googlesource.com/platform/prebuilts/sdk/+/602e78a5c5745f49910a1c4546946b1e912c466b/renderscrip
2. in Android Studio or other IDEs, add the two libs to your arm64-v8a folder.

No x86_64 prebuilts are available yet, but if you need it, it is ok to build from android-6.0.0_r1. Let me know if you need instructions for that.

In the meantime, I will ask our SDK team to bundle the 64bit RenderScript libs, hoping to fix this and other issues with buildTools soon.

---

ma...@zavislak.net <ma...@zavislak.net> #10 No

I'm getting

android.support.v8.renderscript.RSRuntimeException:
Error loading RS jni library:
java.lang.UnsatisfiedLinkError: dlopen failed:
"/data/app/**-redacted**-1/lib/arm64/librsjni.so" has bad ELF magic

when using the prebuilts from https://android.googlesource.com/platform/prebuilts/sdk/+/602e78a5c5745f49910a1c4546946b1e912c466b/renderscript/lib/arm64/

libRSSupport.so
libblasV8.so
librsjni.so

when putting them into /src/main/jniLibs/arm64-v8a

on a Nexus 5x / 6.0 / MDB08L

al...@gmail.com <al...@gmail.com> #11 No

Hello! I've managed to get it working on arm64 by using these libraries:

https://android.googlesource.com/platform/prebuilts/sdk/+/android-6.0.0_r13/renderscript/lib/arm64/

I'm testing on OnePlus Two / 5.1.1

Can someone confirm that these are working on 6.0 also?

no...@gmail.com <no...@gmail.com> #12 No

#16 It works fine on my Nexus 6p

mi...@google.com <mi...@google.com> #13 F

Marked as fixed.

NDK r14 beta2+ supports arm64 and x86_64 for RenderScript.