Camera2 implementation: dynamic camera switching at times cause OnImage available exception: buffer is inaccessible

+1 2   Hotlists (3)   Mark as Duplicate

Comments (9)   Dependencies   Duplicates (0)   Blocking (0)   Resources (4)

Obsolete   Bug   P3   + Add Hotlist   [AOSP] assigned

STATUS UPDATE No update yet. Edit

DESCRIPTION cm...@gmail.com created issue #1   Oct 17, 2021 05:49PM

Build: AI-203.7717.56.2031.7784292, 202110010236,

AI-203.7717.56.2031.7784292, JRE 11.0.10+0-b96-7249189x64 JetBrains s.r.o., OS Linux(amd64) v5.11.0-37-generic, screens 3840.0x2160.0, 2560.0x1600.0

AS: Arctic Fox | 2020.3.1 Patch 3; Kotlin plugin: 203-1.5.31-release-550-AS7717.8; Android Gradle Plugin: 4.2.2; Gradle: 6.7.1; Gradle JDK: version 11.0.10; NDK: from local.properties: (not specified), latest from SDK: 22.1.7171670; LLDB: pinned revision 3.1 not found, latest from SDK: (package not found); CMake: from local.properties: (not specified), latest from SDK: 3.18.1-g262b901, from PATH: 3.16.3

IMPORTANT: Please read https://developer.android.com/studio/report-bugs.html carefully and supply all required information.

aTalk (https://github.com/cmeng-git/atalk-android) implements media call;
Recently aTalk is trying to migrate to use camera2 API; see the attached two files for the main implementation. For other sources please refer to the above link.

When user tries to switch between front and back camera during call, at times it faces problem as shown in the attached logcat below.

a. switchCamera() is called with the user selected new facing camera.
b. it first performs a closeCamera() before start() the new selected camera session.
c. switching camera success rate is not 100%.

The logcat shows that the camera switching work on the first 3 occasions, but not the 4th.
It is found that the "OnImage available exception: buffer is inaccessible" problem happens when the ImageReader is closed. i.e.

```
        if (null != mImageReader) {
            Timber.d("Closing camera - Image reader!");
            mImageReader.close();
            mImageReader = null;
        }
```

when this happen, it leads to
" Unable to set property "debug.sf.dequeuebuffer" to "1": connection failed; errno=13 (Permission denied)"
and the whole process of the media call is halted with camera preview frozen and data streaming stop.

I have tried to rearrange the order of closing the CameraSession, CameraDevice and ImageReader, but does not help.
Please advice what is the proper way of stopping the imageReader without causing the observed problem
i.e. "OnImage available exception: buffer is inaccessible"

```
//----------------------------------------------------
// =========== camera switch #1 sucessful ===============
2021-10-17 14:09:32.913 30123-2071/org.atalk.android D/(PreviewStream.java:292)#switchCamera: Switching camera: androidcamera:0/1
2021-10-17 14:09:32.913 30123-2071/org.atalk.android D/(CameraStreamBase.java:396)#closeCamera: Closing camera - Capture session!
2021-10-17 14:09:32.914 30123-2071/org.atalk.android D/(CameraStreamBase.java:402)#closeCamera: Closing camera - Camera device!
2021-10-17 14:09:32.994 30123-32494/org.atalk.android I/CameraManagerGlobal: Camera 1 facing CAMERA_FACING_FRONT state now CAMERA_STATE_IDLE for client org.atalk.android API Level 2
2021-10-17 14:09:33.063 30123-32494/org.atalk.android I/CameraManagerGlobal: Camera 1 facing CAMERA_FACING_FRONT state now CAMERA_STATE_CLOSED for client org.atalk.android API Level 2
2021-10-17 14:09:33.065 30123-2071/org.atalk.android D/(CameraStreamBase.java:409)#closeCamera: Closing camera - Image reader!
2021-10-17 14:09:33.093 30123-2071/org.atalk.android D/(CameraStreamBase.java:228)#initCameraOnRotation: set camera display orientation swap: true : 90
2021-10-17 14:09:33.094 30123-2071/org.atalk.android D/(CameraStreamBase.java:186)#startImpl: Camera data stream format: [width=1280, height=720]=>YUV Video Format: Size = [width=720, height=1280] MaxDataLength = -1 DataType = class [B yuvType = -1 StrideY = -1 StrideUV = -1 OffsetY = -1 OffsetU = -1 OffsetV = -1
2021-10-17 14:09:33.109 30123-32041/org.atalk.android I/CameraManagerGlobal: Camera 0 facing CAMERA_FACING_BACK state now CAMERA_STATE_OPEN for client org.atalk.android API Level 2
2021-10-17 14:09:33.248 30123-1945/org.atalk.android D/(PreviewStream.java:137)#onConfigured: Preview capture started!
2021-10-17 14:09:33.618 30123-31893/org.atalk.android I/CameraManagerGlobal: Camera 0 facing CAMERA_FACING_BACK state now CAMERA_STATE_ACTIVE for client org.atalk.android API Level 2

// =========== camera switch #2 sucessful ===============
2021-10-17 14:09:40.504 30123-2184/org.atalk.android D/(PreviewStream.java:292)#switchCamera: Switching camera: androidcamera:1/0
2021-10-17 14:09:40.504 30123-2184/org.atalk.android D/(CameraStreamBase.java:396)#closeCamera: Closing camera - Capture session!
```

Reporter	cm...@gmail.com
Type	Bug
Priority	P3
Severity	S3
Status	Won't fix (Obsolete)
Access	Default access View
Assignee	ba...@google.com
Verifier	--
Collaborators	
CC	cm...@gmail.com gg...@google.com
AOSP ID	--
ReportedBy	Developer
Found In	--
Targeted To	--
Verified In	--
In Prod	

```
2021-10-17 14:09:40.506 30123-2184/org.atalk.android D/(CameraStreamBase.java:402)#closeCamera: Closing camera - Camera device!
2021-10-17 14:09:40.618 30123-2176/org.atalk.android I/CameraManagerGlobal: Camera 0 facing CAMERA_FACING_BACK state now CAMERA_STATE_IDLE for client org.atalk.android API Level 2
2021-10-17 14:09:40.701 30123-2176/org.atalk.android I/CameraManagerGlobal: Camera 0 facing CAMERA_FACING_BACK state now CAMERA_STATE_CLOSED for client org.atalk.android API Level 2
2021-10-17 14:09:40.702 30123-2184/org.atalk.android D/(CameraStreamBase.java:409)#closeCamera: Closing camera - Image reader!
2021-10-17 14:09:40.732 30123-2184/org.atalk.android D/(CameraStreamBase.java:228)#initCameraOnRotation: set camera display orientation swap: true : 90
2021-10-17 14:09:40.732 30123-2184/org.atalk.android D/(CameraStreamBase.java:186)#startImpl: Camera data stream format: [width=1280, height=720]=>YUV Video Format: Size = [width=720, height=1280] MaxDataLength = -1 DataType = class [B yuvType = -1 StrideY = -1 StrideUV = -1 OffsetY = -1 OffsetU = -1 OffsetV = -1
2021-10-17 14:09:40.743 30123-31894/org.atalk.android I/CameraManagerGlobal: Camera 1 facing CAMERA_FACING_FRONT state now CAMERA_STATE_OPEN for client org.atalk.android API Level 2
2021-10-17 14:09:40.855 30123-1945/org.atalk.android D/(PreviewStream.java:137)#onConfigured: Preview capture started!
2021-10-17 14:09:41.407 30123-31895/org.atalk.android I/CameraManagerGlobal: Camera 1 facing CAMERA_FACING_FRONT state now CAMERA_STATE_ACTIVE for client org.atalk.android API Level 2

// =========== camera switch #2 sucessful ===============
2021-10-17 14:09:45.367 30123-2289/org.atalk.android D/(PreviewStream.java:292)#switchCamera: Switching camera: androidcamera:0/1
2021-10-17 14:09:45.367 30123-2289/org.atalk.android D/(CameraStreamBase.java:396)#closeCamera: Closing camera - Capture session!
2021-10-17 14:09:45.368 30123-2289/org.atalk.android D/(CameraStreamBase.java:402)#closeCamera: Closing camera - Camera device!
2021-10-17 14:09:45.470 30123-31894/org.atalk.android I/CameraManagerGlobal: Camera 1 facing CAMERA_FACING_FRONT state now CAMERA_STATE_IDLE for client org.atalk.android API Level 2
2021-10-17 14:09:45.544 30123-31894/org.atalk.android I/CameraManagerGlobal: Camera 1 facing CAMERA_FACING_FRONT state now CAMERA_STATE_CLOSED for client org.atalk.android API Level 2
2021-10-17 14:09:45.549 30123-2289/org.atalk.android D/(CameraStreamBase.java:409)#closeCamera: Closing camera - Image reader!
2021-10-17 14:09:45.577 30123-2289/org.atalk.android D/(CameraStreamBase.java:228)#initCameraOnRotation: set camera display orientation swap: true : 90
2021-10-17 14:09:45.577 30123-2289/org.atalk.android D/(CameraStreamBase.java:186)#startImpl: Camera data stream format: [width=1280, height=720]=>YUV Video Format: Size = [width=720, height=1280] MaxDataLength = -1 DataType = class [B yuvType = -1 StrideY = -1 StrideUV = -1 OffsetY = -1 OffsetU = -1 OffsetV = -1
2021-10-17 14:09:45.591 30123-2300/org.atalk.android I/CameraManagerGlobal: Camera 0 facing CAMERA_FACING_BACK state now CAMERA_STATE_OPEN for client org.atalk.android API Level 2
2021-10-17 14:09:45.725 30123-1945/org.atalk.android D/(PreviewStream.java:137)#onConfigured: Preview capture started!
2021-10-17 14:09:46.084 30123-2300/org.atalk.android I/CameraManagerGlobal: Camera 0 facing CAMERA_FACING_BACK state now CAMERA_STATE_ACTIVE for client org.atalk.android API Level 2

// ========= camera switch #3 failed: OnImage available exception: buffer is inaccessible ============
2021-10-17 14:09:50.256 30123-2434/org.atalk.android D/(PreviewStream.java:292)#switchCamera: Switching camera: androidcamera:1/0
2021-10-17 14:09:50.256 30123-2434/org.atalk.android D/(CameraStreamBase.java:396)#closeCamera: Closing camera - Capture session!
2021-10-17 14:09:50.257 30123-2436/org.atalk.android V/aTalk: [108891] net.sf.fmj.media.rtp.RTCPReporter.run() RTCP reporting for running again after 3420.349806085476 ms.
2021-10-17 14:09:50.258 30123-2434/org.atalk.android D/(CameraStreamBase.java:402)#closeCamera: Closing camera - Camera device!
2021-10-17 14:09:50.374 30123-32494/org.atalk.android I/CameraManagerGlobal: Camera 0 facing CAMERA_FACING_BACK state now CAMERA_STATE_IDLE for client org.atalk.android API Level 2
2021-10-17 14:09:50.446 30123-32494/org.atalk.android I/CameraManagerGlobal: Camera 0 facing CAMERA_FACING_BACK state now CAMERA_STATE_CLOSED for client org.atalk.android API Level 2
2021-10-17 14:09:50.447 30123-2434/org.atalk.android D/(CameraStreamBase.java:409)#closeCamera: Closing camera - Image reader!

2021-10-17 14:09:50.449 30123-1945/org.atalk.android E/(PreviewStream.java:165)#lambda$new$0$PreviewStream: OnImage available exception: buffer is inaccessible
2021-10-17 14:09:50.473 30123-2434/org.atalk.android D/(CameraStreamBase.java:228)#initCameraOnRotation: set camera display orientation swap: true : 90
2021-10-17 14:09:50.473 30123-2434/org.atalk.android D/(CameraStreamBase.java:186)#startImpl: Camera data stream format: [width=1280, height=720]=>YUV Video Format: Size = [width=720, height=1280] MaxDataLength = -1 DataType = class [B yuvType = -1 StrideY = -1 StrideUV = -1 OffsetY = -1 OffsetU = -1 OffsetV = -1
2021-10-17 14:09:50.485 30123-32494/org.atalk.android I/CameraManagerGlobal: Camera 1 facing CAMERA_FACING_FRONT state now CAMERA_STATE_OPEN for client org.atalk.android API Level 2
2021-10-17 14:09:50.614 30123-1945/org.atalk.android D/(PreviewStream.java:137)#onConfigured: Preview capture started!
2021-10-17 14:09:51.172 30123-31894/org.atalk.android I/CameraManagerGlobal: Camera 1 facing CAMERA_FACING_FRONT state now CAMERA_STATE_ACTIVE for client org.atalk.android API Level 2
2021-10-17 14:09:52.681 30123-32494/org.atalk.android W/libc: Unable to set property "debug.sf.dequeuebuffer" to "1": connection failed; errno=13 (Permission denied)
2021-10-17 14:09:52.681 30123-32494/org.atalk.android E/BufferQueueProducer: [ImageReader-1280x720f23m1-30123-6] (id:75ab00000006,api:4,p:763,c:30123) waitForFreeSlotThenRelock: timeout
2021-10-17 14:09:53.646 30123-1931/org.atalk.android V/aTalk: [108844] net.sf.fmj.media.rtp.RTCPReporter.run() RTCP reporting for running again after 5912.912200066227 ms.
2021-10-17 14:09:53.682 30123-31894/org.atalk.android W/libc: Unable to set property "debug.sf.dequeuebuffer" to "1": connection failed; errno=13 (Permission denied)
2021-10-17 14:09:53.682 30123-31894/org.atalk.android E/BufferQueueProducer: [ImageReader-1280x720f23m1-30123-6] (id:75ab00000006,api:4,p:763,c:30123) waitForFreeSlotThenRelock: timeout
2021-10-17 14:09:53.684 30123-2436/org.atalk.android V/aTalk: [108891] net.sf.fmj.media.rtp.RTCPReporter.run() RTCP reporting for running again after 4644.629607686943 ms.
2021-10-17 14:09:54.683 30123-2176/org.atalk.android W/libc: Unable to set property "debug.sf.dequeuebuffer" to "1": connection failed; errno=13 (Permission denied)
2021-10-17 14:09:54.683 30123-2176/org.atalk.android E/BufferQueueProducer: [ImageReader-1280x720f23m1-30123-6]
```

```
(id:75ab00000006,api:4,p:763,c:30123) waitForFreeSlotThenRelock: timeout
2021-10-17 14:09:55.684 30123-31894/org.atalk.android W/libc: Unable to set property "debug.sf.dequeuebuffer" to "1": connection
failed; errno=13 (Permission denied)
2021-10-17 14:09:55.684 30123-31894/org.atalk.android E/BufferQueueProducer: [ImageReader-1280x720f23m1-30123-6]
(id:75ab00000006,api:4,p:763,c:30123) waitForFreeSlotThenRelock: timeout
2021-10-17 14:09:56.686 30123-2176/org.atalk.android W/libc: Unable to set property "debug.sf.dequeuebuffer" to "1": connection
failed; errno=13 (Permission denied)
2021-10-17 14:09:56.686 30123-2176/org.atalk.android E/BufferQueueProducer: [ImageReader-1280x720f23m1-30123-6]
(id:75ab00000006,api:4,p:763,c:30123) waitForFreeSlotThenRelock: timeout
2021-10-17 14:09:56.836 30123-1915/org.atalk.android W/(MediaStreamStatsImpl.java:342)#computeRTTInMs: invalid_rtt:
stream=101575514 ssrc=95775726, rtt(ms)=3200, now=be34d439, lsr=be2e7374, dlsr=32d91
2021-10-17 14:09:57.687 30123-31894/org.atalk.android W/libc: Unable to set property "debug.sf.dequeuebuffer" to "1": connection
failed; errno=13 (Permission denied)
2021-10-17 14:09:57.687 30123-31894/org.atalk.android E/BufferQueueProducer: [ImageReader-1280x720f23m1-30123-6]
(id:75ab00000006,api:4,p:763,c:30123) waitForFreeSlotThenRelock: timeout
2021-10-17 14:09:58.342 30123-2436/org.atalk.android V/aTalk: [108891] net.sf.fmj.media.rtp.RTCPReporter.run() RTCP reporting for
running again after 4324.150177778006 ms.
2021-10-17 14:09:58.689 30123-2176/org.atalk.android W/libc: Unable to set property "debug.sf.dequeuebuffer" to "1": connection
failed; errno=13 (Permission denied)
2021-10-17 14:09:58.689 30123-2176/org.atalk.android E/BufferQueueProducer: [ImageReader-1280x720f23m1-30123-6]
(id:75ab00000006,api:4,p:763,c:30123) waitForFreeSlotThenRelock: timeout
```

deleted
0 B ?
Restricted

deleted
0 B ?
Restricted

✓ Links (4) Hide all

Links (4)

" …PORTANT: Please read https://developer.android.com/studio/report-bugs.html carefully and supply all required informati… "	cm...@ #1
"aTalk (https://github.com/cmeng-git/atalk-android) implements media call;"	cm...@ #1
"For steps to capture a bug report, please refer: https://developer.android.com/studio/debug/bug-report#bugreportdevice"	gg...@ #3
"http://atalk.sytes.net"	cm...@ #5

COMMENTS

All comments ▾ ↓ Oldest first

cm...@gmail.com <cm...@gmail.com> #2 Oct 17, 2021 07:14PM ⋮

Further investigation shows that the actual problem at transferHandler.transferData(this) in:

```
    // Acquired image always in landscape mode
    private final ImageReader.OnImageAvailableListener mOnImageAvailableListener = reader -> {
        try (Image image = reader.acquireNextImage()) {
            if (ImageFormat.YUV_420_888 == image.getFormat()) {
                calcStats(); // Calculate statistics for average frame rate
                synchronized (bufferQueue) {
                    bufferQueue.addFirst(image);
                }
                transferHandler.transferData(this);
            }
        } catch (Exception e) {
            Timber.e(e, "OnImage available exception: %s", e.getMessage());
        }
    };
```

where the buffered image is being processed in YUV420PlanarRotate(image, copy, w, h, mCameraRotation); when the yBuffer is being accessed.
How can aTalk determines if the acquired image is valid for process, or even better is to stop the OnImageAvailableListener all together on closing Camera ?

```
2021-10-17 16:04:48.873 21268-23804/org.atalk.android D/(CameraStreamBase.java:420)#closeCamera: Closing camera -
Image reader!
2021-10-17 16:04:48.893 21268-23696/org.atalk.android E/(PreviewStream.java:176)#lambda$new$0$PreviewStream: OnImage
available exception: buffer is inaccessible
    java.lang.IllegalStateException: buffer is inaccessible
        at java.nio.DirectByteBuffer.get(DirectByteBuffer.java:222)
        at
org.atalk.impl.neomedia.jmfext.media.protocol.androidcamera.PreviewStream.YUV420PlanarRotate(PreviewStream.java:267)
        at org.atalk.impl.neomedia.jmfext.media.protocol.androidcamera.PreviewStream.read(PreviewStream.java:198)
        at
org.atalk.impl.neomedia.protocol.RewritablePushBufferDataSource$MutePushBufferStream.read(RewritablePushBufferDataSour
ce.java:299)
        at net.sf.fmj.media.protocol.CloneableSourceStreamAdapter.copyAndRead(CloneableSourceStreamAdapter.java:375)
        at
```

```
net.sf.fmj.media.protocol.CloneableSourceStreamAdapter$PushBufferStreamAdapter.read(CloneableSourceStreamAdapter.java:
60)
        at net.sf.fmj.media.parser.RawPushBufferParser$FrameTrack.transferData(RawPushBufferParser.java:612)
        at
net.sf.fmj.media.protocol.CloneableSourceStreamAdapter$PushBufferStreamAdapter.transferData(CloneableSourceStreamAdap
ter.java:72)
        at
org.atalk.impl.neomedia.protocol.StreamSubstituteBufferTransferHandler.transferData(StreamSubstituteBufferTransferHandler.j
ava:75)
        at
org.atalk.impl.neomedia.jmfext.media.protocol.androidcamera.PreviewStream.lambda$new$0$PreviewStream(PreviewStream.ja
va:173)
        at org.atalk.impl.neomedia.jmfext.media.protocol.androidcamera.-$$Lambda$PreviewStream$bb7u-VOtG7MWXmhHvtPU-
dVQYVU.onImageAvailable(Unknown Source:2)
        at android.media.ImageReader$ListenerHandler.handleMessage(ImageReader.java:812)
        at android.os.Handler.dispatchMessage(Handler.java:106)
        at android.os.Looper.loop(Looper.java:246)
        at android.os.HandlerThread.run(HandlerThread.java:67)
```

Message last modified on Oct 17, 2021 07:25PM

gg...@google.com <gg...@google.com> #3 Oct 19, 2021 02:37AM ⋮

Assigned to gg...@google.com.

Thank you for reporting this issue. For us to further investigate this issue, please provide the following additional information:

Device used -- Device Make, Model, Android OS Version
Which device did you use to reproduce this issue?

Please provide a sample project(complete project) or apk to reproduce the issue. Also mention the steps to be followed for reproducing the issue with the given sample project or apk.

Steps to reproduce
What steps are needed to reproduce this issue?

What is the expected output?

What is the current output?

Share complete Android bug report (to be captured after reproducing the issue)
For steps to capture a bug report, please refer: https://developer.android.com/studio/debug/bug-report#bugreportdevice

Screen record of the issue
Please capture screen record or video of the issue using following steps:
adb shell screenrecord /sdcard/video.mp4
Subsequently use following command to pull the recorded file:
adb pull /sdcard/video.mp4
Attach the file to this issue.

Note: Please avoid uploading directly to the issue using attachments. Please upload to google drive and share the folder to android-bugreport@google.com, then share the link here.

cm...@gmail.com <cm...@gmail.com> #4 Oct 22, 2021 08:27PM ⋮

The test is carried on my Note-10+ Android-11 API-30

Attached below is the apk built from the attached source

What steps are needed to reproduce this issue?
a. Install the aTalk apk on two android devices
b. Register an xmpp account on each android device with an commercial xmpp server.
c. Made a video call between the two devices; you may want to disable the call zrtp encryption for the test.
d. During the call, long press on the camera icon to switch between front and back camera.
e. You may need to switch between the front/back cameras a few times, before seeing the problem i.e. the failure is not 100%

See previous comments on the problem observed, which is not expected when switching camera.

Please refer to aTalk online help for use of the app, access via the aTalk main menu.

For all other requests, please build from the source and observed yourself.
Note: you may need to execute getJniLibs/initJniLibs in build.gradle to build the Jni libraries before the actual apk build.

deleted Restricted
0 B ?

deleted Restricted
0 B ?

cm...@gmail.com <cm...@gmail.com> #5 Oct 22, 2021 08:29PM ⋮

You may use the following two xmpp accounts on atalk.sytes.net for your testing.

test1@atalk.sytes.net / pwd 1234
test2@atalk.sytes.net / pwd 1234

gg...@google.com <gg...@google.com> #6 Oct 22, 2021 10:56PM

Thank you for reporting this issue. We've shared this with our product and engineering teams and will continue to provide updates as more information becomes available.

gg...@google.com <gg...@google.com> #7 Jan 17, 2022 06:22PM

The fix to ImageReader will be included in next android OS release, but there is currently no way to update ImageReader out of band with the rest of the OS.

An app level workaround would be to protect image reads from ImageReader (like calls to `ImageReader.acquireNextImage()` and such), and changes to ImageReader (like calls to `ImageReader.close()` or `ImageReader.setOnImageAvailableListener()`) with a mutex, and track ImageReader's state in a field.

The code could look something like:

For closing or registering ImageReader

```
synchronized (imageReaderLock) {
  if (null != mImageReader) {
    Timber.d("Closing camera - Image reader!");
    mImageReader.close();
    mImageReader = null;
    mImageReaderClosed = true; // New field to track ImageReader's state
  }
}
```

and in OnImageAvailableListener:

```
private final ImageReader.OnImageAvailableListener mOnImageAvailableListener = reader -> {
  synchronized (imageReaderLock) {
    if (mImageReaderClosed) { // Query ImageReader's state before getting the next frame
      return; // Do not process if imageReader is closed
    }

    try (Image image = reader.acquireNextImage()) {
      // . . .
    } catch (Exception e) {
      Timber.e(e, "OnImage available exception: %s", e.getMessage());
    }
  }
};
```

This should prevent the onImageAvailableListener from trying to fetch an image after the reader is closed.

The bookkeeping might get more complicated if there are more than one ImageReaders and listeners involved, but this is the basic gist of the solution. For example, if you are switching between ImageReaders frequently, you might have to create pairings of ImageReader instances and their current closed status (either through class composition, or with some other data structure).

cm...@gmail.com <cm...@gmail.com> #8 Jan 18, 2022 12:14PM

Thanks for the recommendation.

aTalk v2.8.0 has already implemented similar approach at App level to resolve the image reader problem.

gg...@google.com <gg...@google.com> Jan 31, 2022 03:41PM

Reassigned to ba...@google.com.

ba...@google.com <ba...@google.com> #9 Jan 31, 2022 05:39PM

Status: Won't Fix (Obsolete)

As per comment#8 we are closing the issue.