

Turnip Without Root

Lucas Fryzek

XDC 2026, Toronto, 2026-09-29

Intro

- Who's this talk for?
- Maybe you have a Qualcomm powered phone?
- Maybe you're interested in graphics driver development?
- Maybe you've heard about all this Turnip stuff for making games run better on android?

Intro

- If you answered yes, this is probably for you
- Introduction to doing mesa development on unmodified android phones without root

Quick Overview

- What makes unmodified android phones "harder" to do mesa dev on?
- They only run downstream KGSL kernel driver
- Android doesn't let you easily swap out driver to test a new version with a user debug OS image

Quick Overview

- How do we solve this?
- Turnip has KGSL kernel backend
- We can't replace driver at system level but we can at app level
 - If we build apps ourselves

Developer Crash Course

Developer Crash Course - Building

- **We have docs!**
- Can build mesa/turnip using normal Android NDK
- Important build flags are
 - `-Dplatforms=android`, `-Dplatform-sdk-version=34`, `-Dandroid-stub=true`
 - `-Dfreedreno-kmds=kgs1`

Developer Crash Course - Testing

- What's the most important software for a driver developer?

Developer Crash Course - Testing

- What's the most important software for a driver developer?
- Games

Developer Crash Course - Testing

- What's the most important software for a driver developer?
- ~~Games~~
- Conformance Test Suite (CTS)
 - We'll get to the games later

CTS

- How can we run on unmodified android phone
- Use VK-CTS in **android executable mode**
 - Lets you run CTS like a normal execute on `adb shell`
 - we don't need to see graphics output, just pass/fail
- Need to copy library to somewhere with exec permission
 - like `/tmp`, or `/data/local/tmp`

But how can we replace the driver?

Replace the driver

- Use a dirty LD_PRELOAD hack

```
void *android_load_sphal_library(const char *name, int flags)
{
    if (!strcmp(name, "vulkan.adreno.so"))
        return dlopen("/tmp/libvulkan_freedreno.so", flags);

    ...

    return android_load_sphal_library_ptr(name, flags);
}
```

Replace the driver

- LD Preload hack only works well from CTS running on `adb shell`
- If we want to replace driver in real android app we need something else

Replace the driver

- That something else is **libadrenotools**
- Handles annoying namespace stuff to load the driver in apk
- Makes it so calls to GetProcAddress will get the ones from custom driver build
- You need to build this into your own app
 - Can't just replace the driver in already built APKs

What about Games?

What about Games?

- Lots of different projects on Android to run games
 - Some let you load your own driver
- Example: GameNative
 - Driver package is:
 - `libvulkan_freedreno.so` We built this in step 1
 - `meta.json` just some information about the name to use in the UI and the version

Now what?

- Well you can build your own driver
- You can run CTS
- You can run Games

Now what?

- Go fix bugs and be a driver developer!

Limitations

- Without root you can't get crash logs from KGSL
- Not all debug tools work easily
 - Need to patch replay tools to load the custom driver
- If you're doing profiling not all trace events available with root

Limitations

- Not every device is supported upstream yet
 - You can check [freedreno_devices.py](#) for what is
 - but maybe you could be the one to do it!
- Following this in a presentation isn't easy
 - Blog post to follow soon (**TM**)

Discussion

Join us!

<https://www.igalia.com/jobs>



