



Vendor hooks in Android kernels?

Why?

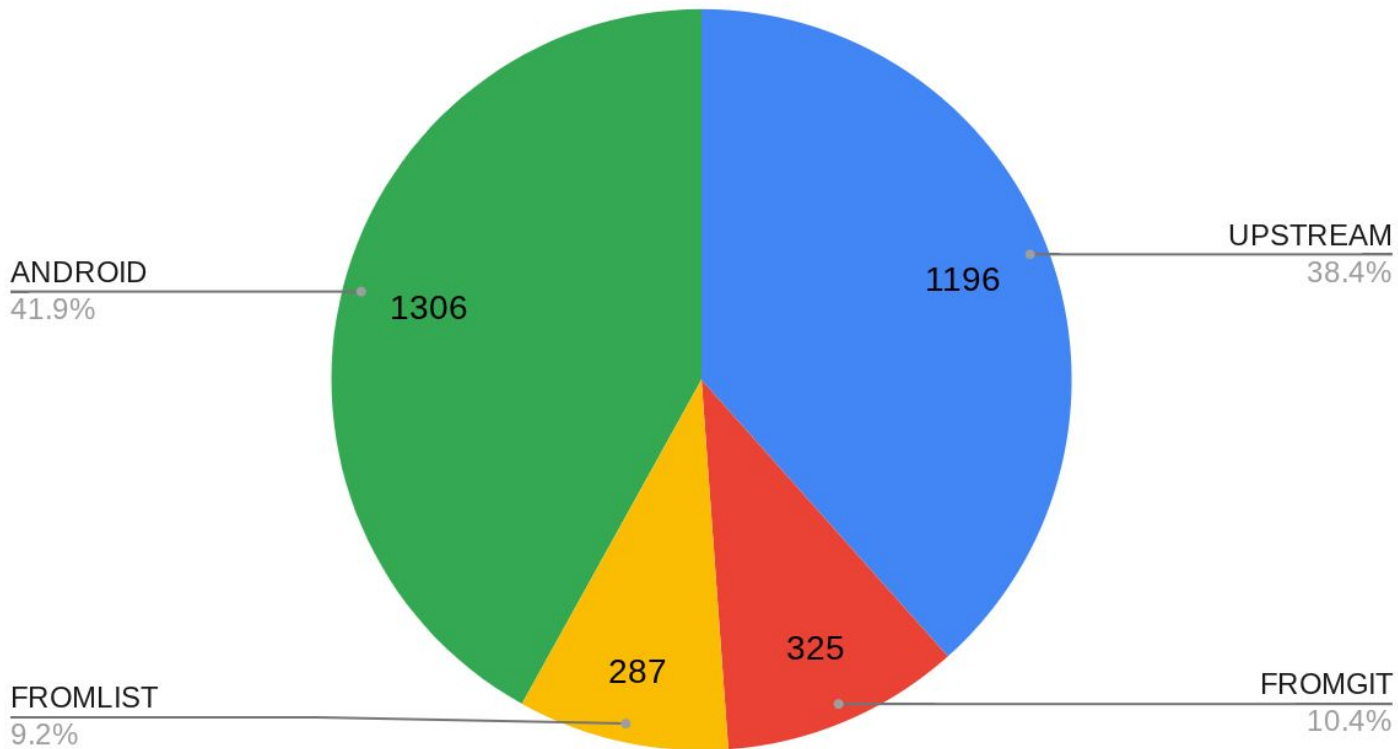
What do vendors do with them?

LINUX
PLUMBERS
CONFERENCE

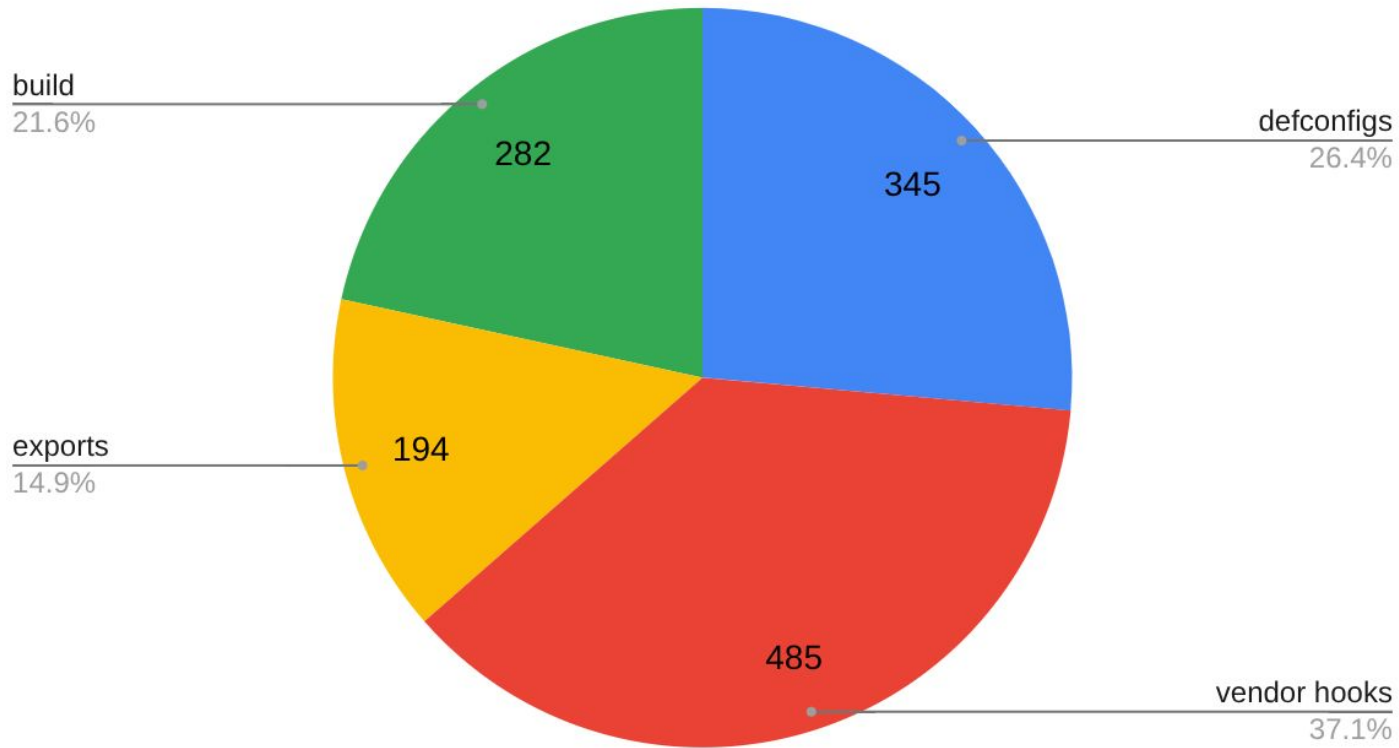
PRAGUE, CZECHIA / OCT. 5-7, 2026

Todd Kjos <tkjos@google.com>

6.6 Out-of-tree patches



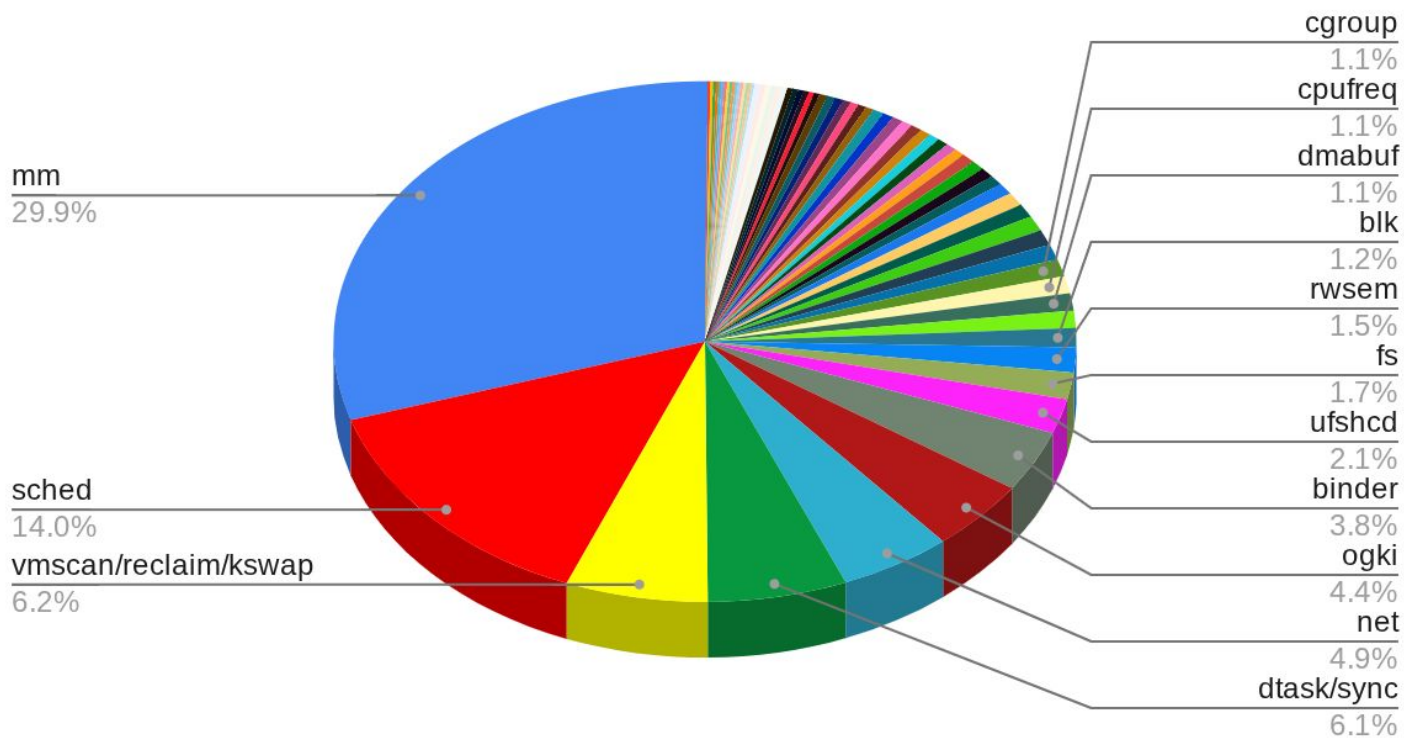
ANDROID patch types (excluding ABI-related patches)



LINUX
PLUMBERS
CONFERENCE

PRAGUE, CZECHIA
OCT. 5-7, 2026

6.6 Vendor Hooks by Subsystem (Total: 840)



Why do partners use vendor hooks?

- **Urgency:** no time to work with upstream
- **Upstream too Slow:** will work with upstream, but need it now
- **Lack of upstream users:** SoC is not supported upstream, so cannot provide a driver for the feature
- **GKI Policy:** Must use vendor hooks if not upstreamable
- **Narrowness:** feature not interesting outside small set of devices
- **Bandwidth:** No organizational bandwidth to work with upstream
- **Secrecy:** don't want to reveal feature implementation earlier than necessary
- **Inertia:** it works so don't want to change it



LINUX

PLUMBERS CONFERENCE

PRAGUE, CZECHIA / OCT. 5-7, 2026



What do they look like?

Vendor Hooks in the GKI kernel:

```
int foo(struct task_struct *task)
{
    int ret = 1;
    trace_android_vh_my_feature(&ret, task)
    if (!ret) return;
}
```

In vendor module:

```
void my_feature(int *ret, struct task_struct *task)
{
    *ret = do_stuff(task);
}

void connect_to_vh()
{
    rc = register_trace_android_vh_my_feature(my_feature,
    NULL);
}
```



LINUX

PLUMBERS CONFERENCE

PRAGUE, CZECHIA / OCT. 5-7, 2026



Some example use-cases

- **Scheduler Load Tracking:**
 - WALT (Window-Assisted Load Tracking)
- **Priority inversion prevention:**
 - Locking primitives have hooks that give sched priority to tasks blocking "important" tasks
- **Event timing:**
 - `trace_android_vh_mm_compaction_begin()`
 - `trace_android_vh_mm_compaction_end()`
- **Custom event handling:**
 - `trace_android_vh_scheduler_tick()`



LINUX

PLUMBERS CONFERENCE

PRAGUE, CZECHIA / OCT. 5-7, 2026



Discussion:

How can we reduce reliance on vendor hooks?

- What can the Android ecosystem do better?
- What can Google do better?
- What can we do upstream to help solve problems more efficiently?

