

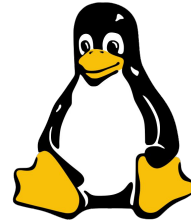
postmarketOS - Digital Independence in Your Pocket

Luca Weiss, FrOSCon 2025

About me

- **Luca Weiss**
- Software Longevity Team at Fairphone
- postmarketOS Core Contributor
- OpenRazer Maintainer

fairphone



 fosstodon.org/@z3ntu

 lucaweiss.eu

 postmarketOS

The smartphone landscape

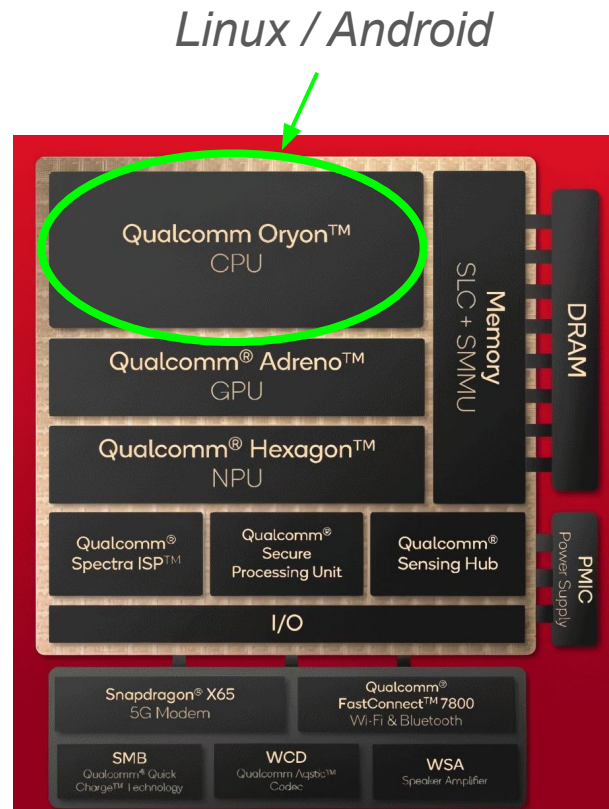
- Duopoly of operating systems:
 - iOS (iPhone, Apple)
 - Android (various manufacturers)
- Basically all Android phones are shipping Google Play
 - 'Required' / expected by vast majority of customers
- Android phone manufacturers are bound by Google's requirements and policies
 - Good: minimum security requirements, compatibility with Android apps
 - Bad: Reduced independence, strict requirements, required to work with US company

Why is this duopoly a problem?

- Google and Apple control the whole market
- **Privacy:** required to bundle privacy-invasive apps
- **Independence:** Google controls all of Android, development is done top-down
- **Proprietary code:** no smartphone can work without, default Board Support Package (BSP) contains lots of it
 - While manufacturer has some proprietary source code, many parts are also prebuilt
- **Software updates:** bound by Google to approve every software update

The proprietary low level software

- Linux Kernel: GPL-2.0 licensed
 - All(*) kernel-related parts need to be open-sourced
 - Many kernel-to-user space interfaces are not standardized
- Hardware Abstraction Layer (HAL)
 - Bridge between kernel and higher level frameworks
 - Tightly coupled to underlying kernel interfaces
 - Unable to easily replace kernel driver with an 'upstream' one
- Modem, boot firmware, etc.
 - Running mostly on separate CPU cores
 - All proprietary
 - Essentially impossible to replace with open source



What about 3rd party Android ROMs?

- Plenty independent or community driven Android ROMs exist
 - LineageOS, /e/OS, CalyxOS, etc.
- More open-source than the stock OS (basically always)
 - Still far from 100% open-source
 - Many proprietary blobs get taken from stock OS
 - Reliant on stock OS to update all proprietary components
 - No real effort to replace proprietary components with open source re-implementations



Introducing postmarketOS



The Linux distribution for mobile devices and beyond...

- Fully open-source user space (and kernel space)
- Focus on long-term sustainability
 - No (big) device-specific adaptations, everything is quite generic across devices
- Independent project, nearly fully volunteer driven and donation-funded
- Based on Alpine Linux
- You get a package manager!
- Tooling + packages + build infra + documentation

Why we don't do re-use Android parts

Halium is the collaborative project to unify the Hardware Abstraction Layer for projects which run GNU/Linux on mobile devices with pre-installed Android.

- Re-using original Android kernel
- Re-using Android runtime and HALs
- Provides plumbing to make e.g. modem or camera work from (glibc/musl) user space

I do appreciate projects using Halium, because:

- It's a pragmatic approach to make things work
- Provides an environment for making a usable operating system

But...

- Relying on proprietary software
- Lack of updates of Android parts
- Integration into regular Linux user space (e.g. Phosh) is tricky / not well maintained
- Doesn't encourage the development of 'native' solutions

What is using an upstream kernel enabling?

- **Staying up-to-date with kernel developments**
 - User space generally likes new features in the kernel
 - systemd, Docker, etc. (including Android)
 - Minimum kernel requirements going up
- **Staying secure**
 - Kernels without updates are insecure kernels
 - Kernels are not maintained forever (LTS: max. 6 years)
 - Merging stable kernel into Android forks is tricky
- **Being future-proof**
 - Upstream support means updating to new kernel versions is easy!
 - (ignoring potential regressions, but if those are fixed, everybody gets the fix)
- **Simplified support for distros**
 - On Android generally every device has its own kernel fork
 - Currently in postmarketOS: Kernels maintained per SoC
 - Dozens of devices using the same exact kernel binary
 - Kernel interfaces are standardized - doesn't require special user space

What postmarketOS is like right now

- Used by people as daily driver
 - Some(?) carry a dumb phone for phone calls
- Limited device support, limited support per device
 - Devices: 129 'community', 292 'testing' (+ 257 'downstream')
 - Differs between devices a lot - some are quite functional, some have barely anything working
- Not yet ready for the *average* person
 - But ready for the average **hacker**
- Functional vs. stable
 - **Functional:** a feature works when tested in isolation
 - **Stable:** a feature reliably works every time
 - Stability is certainly lacking but being worked on, e.g. audio

What the plans for the future are

No fixed roadmap since we're volunteer-driven, but there are certainly many things being worked on!

- Getting systemd ready (incl. musl patches)
- Immutable system, reliable updates
- Automated testing setup
 - Test changes in postmarketOS in various devices
 - Boot-test kernels automatically
 - And much more!
- Tackling long-term issues, like audio stability and reliability
- A new name, reflecting what our goal is
 - Not just an OS for after (**post**) the **market** is done with a device
 - Certainly started out as reviving e-waste, but we want to do so much more!
- Better installation instructions and documentation
- Improving our infrastructure, governance, etc.

What you can do with postmarketOS *today*

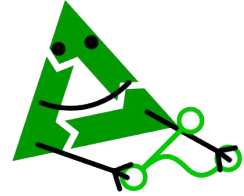
- Install it, boot it, hack it!
- ‘Testing’/’Community’/’Main’ ports are the most useful
 - These categories run a mainline(-ish) kernel
 - ‘Downstream’ ports are very limited - they run downstream kernel
- Headless or GUI?
- Run any server-side software on it
 - Docker containers
 - Kubernetes cluster from multiple phones
 - Host a webserver
 - Play around with software and learn things!
- Or some other creative way
 - Media playback control tablet, solar-powered off-grid server, etc.

How you can help (for the techies)

postmarketOS will get better with contributions!

Improve the ecosystem!

- Kernel development (hardware enablement, stability)
- Middleware (feedbackd, libcamera)
- UI contributions (e.g. Phosh to improve UX, implement features)
- App contributions (improve existing apps, develop new ones)
- Maintain packages in Alpine aports / postmarketOS pmaports



How you can help (as non-dev)

There's endless ways to contribute without coding!

- Try postmarketOS on *your* device
- Write down your experience
- Update/improve documentation
- Create/update translations for UI and apps
- Spread the word at conferences - all of them
- Help write our monthly blog posts
- Help with the podcast (organizing, editing, etc.)
- Help organize funding grants

See postmarketos.org/contribute!

If you have ideas: email team@postmarketos.org



FroSCon

Thank you!

Questions?

