From IoT to Mobile

Oniro's Technical Journey Toward Digital Independence

Francesco Pham
Oniro PMC Lead - Huawei

The Digital Sovereignty Problem

Two platforms dominate mobile computing

- Concentrated power & structural dependency
- Limited strategic options
- Privacy and security challenges

Europe needs alternatives that deliver:

- Strategic independence
- Multi-device interoperability
- Privacy compliance & local accountability

What is Oniro?

Open-source OS platform managed by Eclipse Foundation

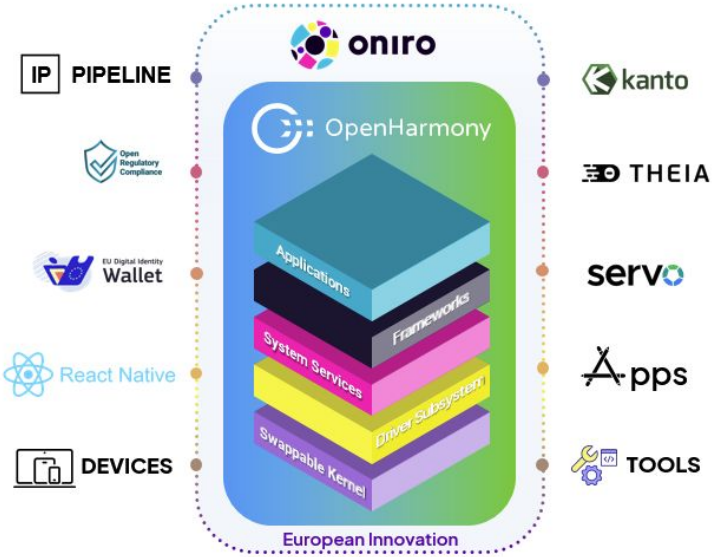
Built on **OpenHarmony** (a mature open-source project):

- Hundreds of products in production
- Thousands of contributors
- Millions of lines of code

Unique dual-foundation model:

- OpenAtom Foundation (China) + Eclipse Foundation (Europe)
- Shared technology, European governance
- Global scale with regional trust

Mission: Make cross-device platforms accessible for global adoption



From Foundations to Global Access

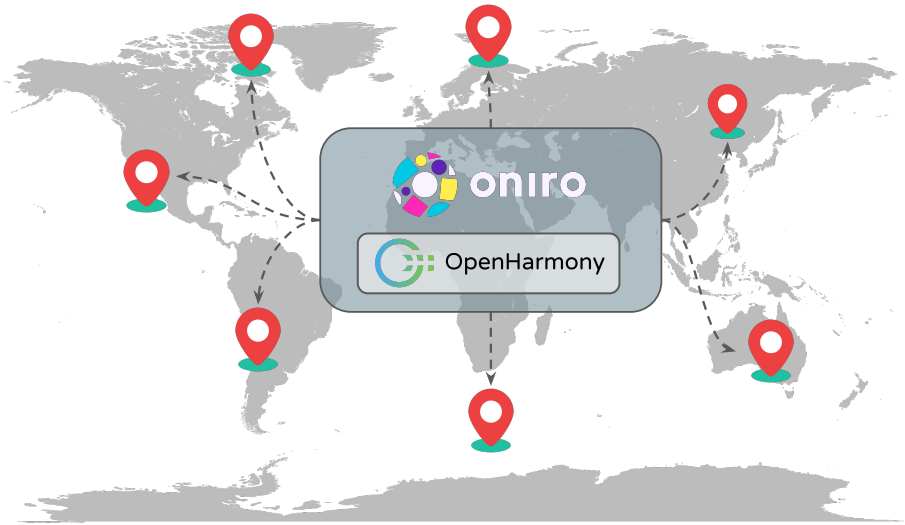
Why Oniro bridges OpenHarmony to the world:

Barriers outside China:

- Platform unfamiliarity (GitCode vs GitHub)
- Language barriers
- Limited access to familiar devices/tools

Oniro removes friction:

- GitHub mirrors with daily synchronization
- English-first documentation
- Globally available hardware support
- Developer-friendly tooling

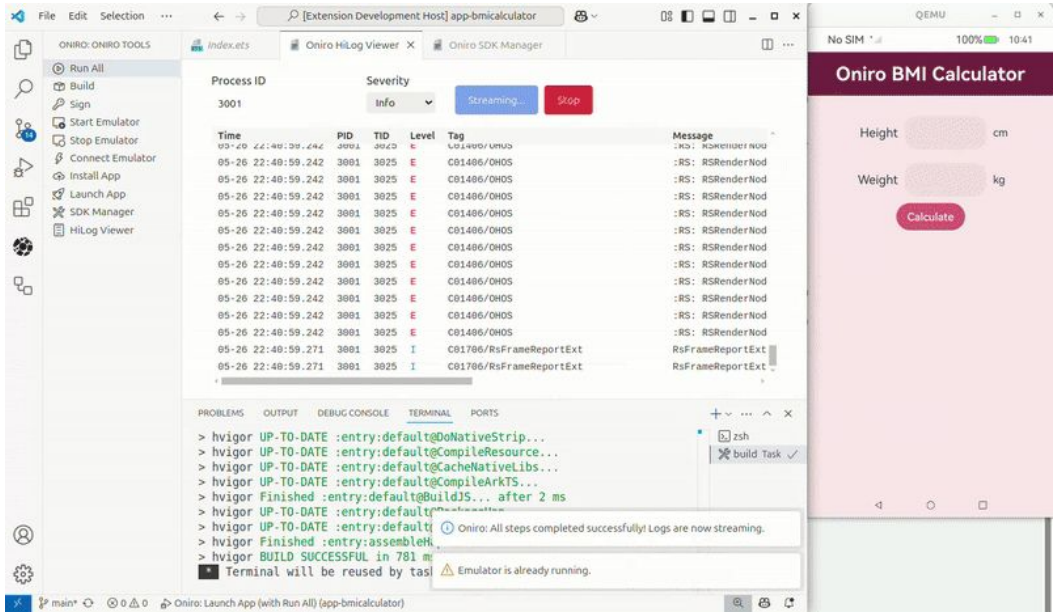


Developer Experience – Oniro IDE

VS Code extension for Oniro/OpenHarmony development

Key features:

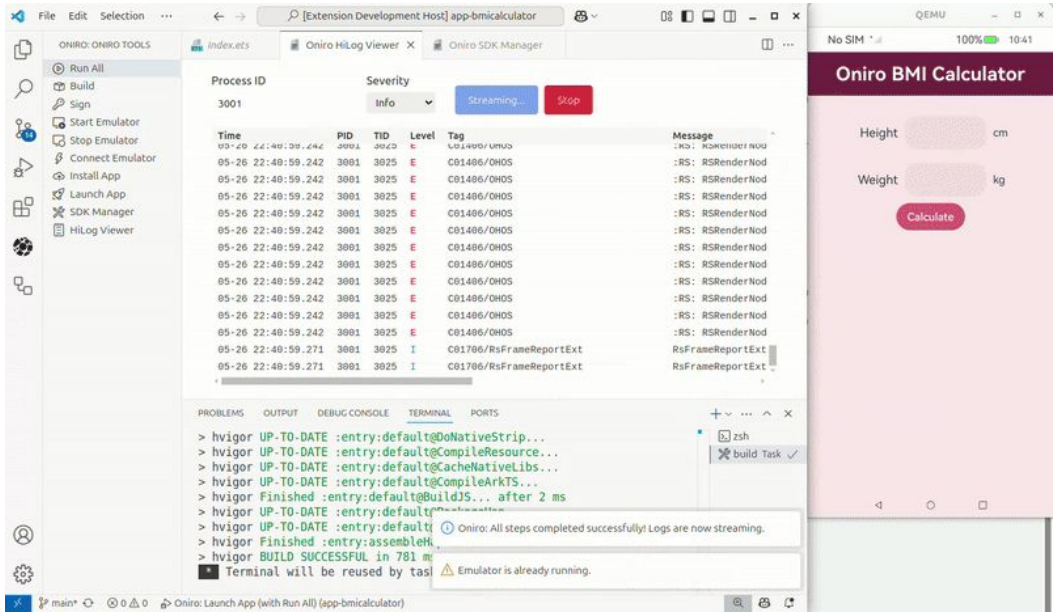
- SDK management & command-line tools
- Emulator integration
- Signature generation for debugging
- Build HAP packages
- Install, start apps, and filter logs (HiLog)



Developer Experience – Oniro IDE

Impact:

- Fast development workflow in familiar editor
- Linux support (no Windows requirement)
- AI-assisted development ready
- Lowers entry barrier for new contributors



Testing Without Hardware – QEMU Emulator

QEMU-based Emulator

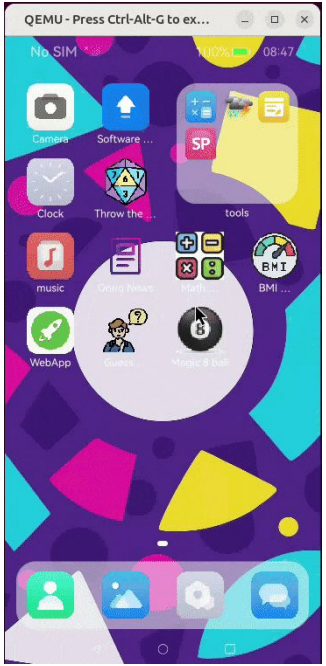
Technical approach:

- Started with Yocto meta-layer + LXC container
- Native OpenHarmony integration
- x86_64 architecture → hypervisor acceleration
- Full graphical support (Mesa drivers)

Enables:

- Instant testing + IDE integration
- CI/CD pipelines
- Rapid prototyping

OpenHarmony accessible to everyone—no physical device needed



Real Hardware – IoT to Mobile

Platform maturity across device types

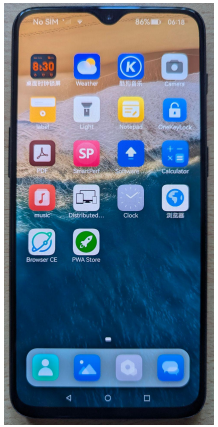
IoT & Embedded:

- OpenHarmony reference boards (Hoperun)
- Raspberry Pi support (globally accessible)

Mobile devices:

- OpenHarmony Developer Phone (Unisoc chipset)
- OnePlus 6T (community port)
- European mobile reference device (proof-of-concept)

Goal: Expand to globally accessible hardware



Challenges of Mobile Hardware Porting

The proprietary driver problem strengthens the duopoly:

- Manufacturers don't open-source mobile drivers
- Drivers built for Android (Bionic libc), incompatible with alternatives
- Binary blobs create vendor lock-in and reinforce platform monopolies

Recent recognition: FSF's Librephone Initiative (October 2025)

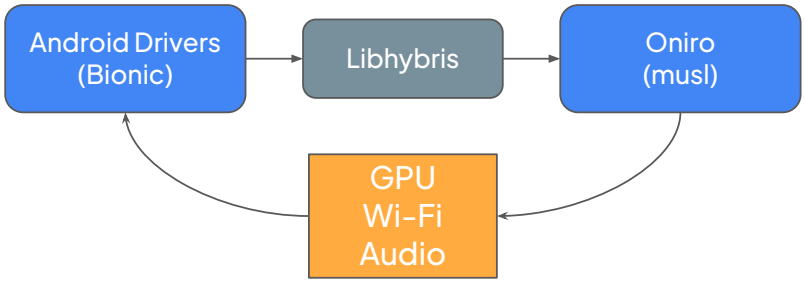
- Aims to reverse-engineer obstacles to mobile phone freedom
- Close gaps between Android distributions and software freedom
- Led by Rob Savoye, building on LineageOS work
- "Making fully free software for modern phones will not be quick, easy, or cheap"

Challenges of Mobile Hardware Porting

Oniro's pragmatic approach: LibHybris integration

- Adaptation layer translating Android calls to OpenHarmony's musl libc
- Enables GPU, Wi-Fi, peripheral support
- **Workaround, not ideal**: Binary blobs remain a black box
- First integration complete, more work ongoing

Bridging proprietary Android drivers for open systems → Libhybris



This could unlock Oniro for existing global mobile phones

Unified Cross-Device Architecture

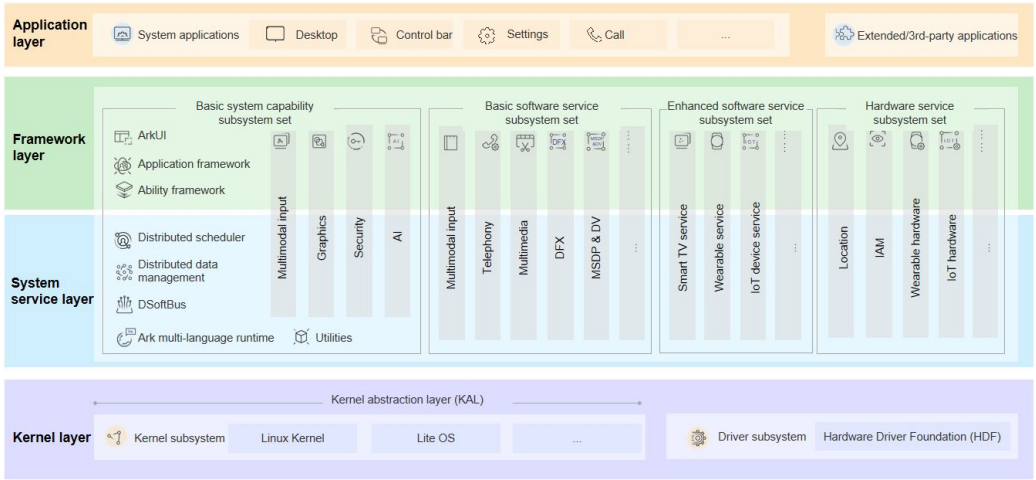
One platform, multiple device classes:

System	Devices	Capabilities
Mini	Sensors, wearables	MCU, lightweight
Small	Cameras, routers	Security, media
Standard	Phones, tablets	Full UI, 3D GPU

Unified Cross-Device Architecture

Key technologies enabling interoperability:

- Multi-kernel support: **LiteOS** for mini system, **Linux** for standard system
- HDF: Unified driver abstraction for portability
- DSoftBus: Seamless device discovery & communication
- Distributed capabilities: Data sync, remote app invocation, device virtualization ("Super Device")



Growing the App Ecosystem

Multiple strategies for app availability:

Infrastructure:

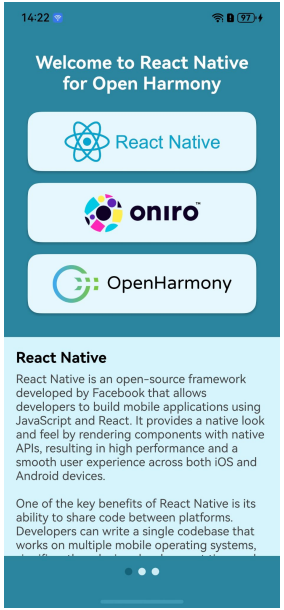
- Oniro App Store (F-Droid-like, open-source distribution)
- CI/CD for apps and system images
- Community apps: SoundCloud, Telegram…

Growing the App Ecosystem

Cross-platform frameworks:

- React Native (hundreds of apps already available)
- Kotlin Multiplatform & Flutter (under research)
- Rust frameworks (Tauri, Robius, Dioxus)

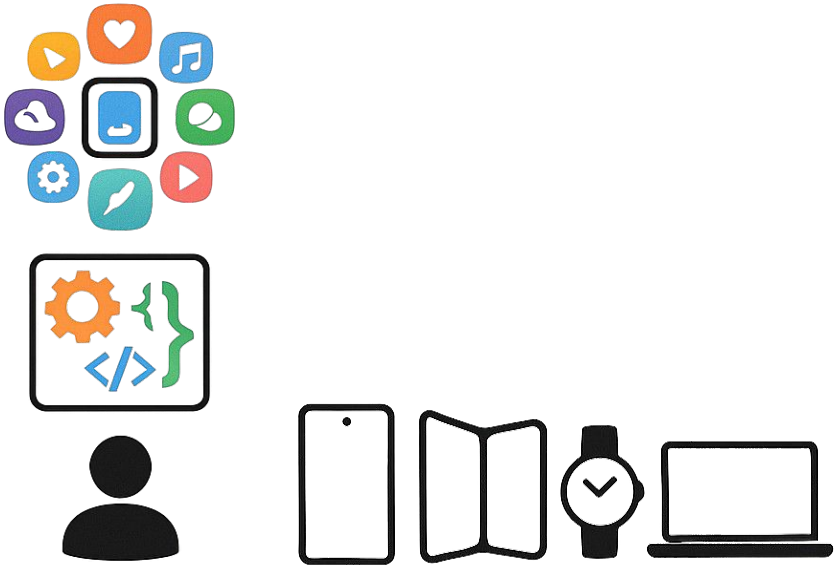
Targeting multiple platforms with a single codebase.



Roadmap

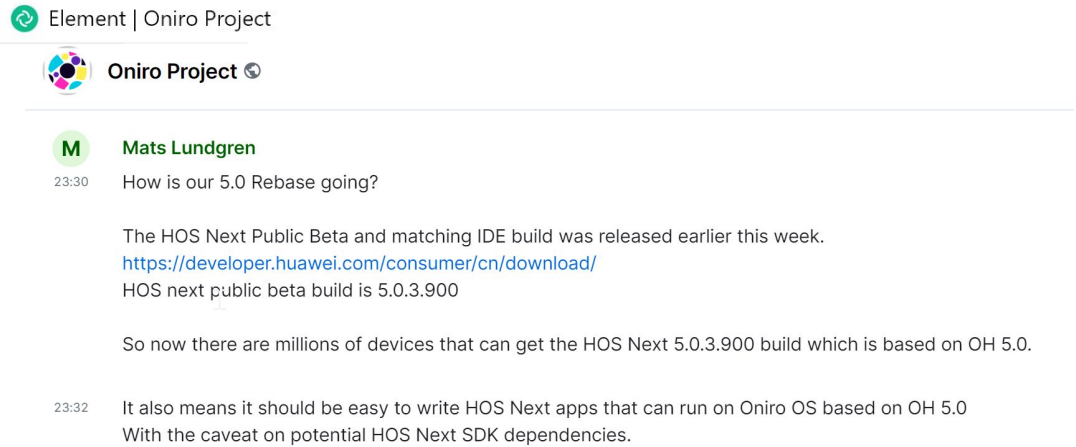
Future works:

- Richer app ecosystem through
- Wider third party frameworks integration
- Stronger global developer ecosystem
- More phones & form factors



Oniro Community

- Connect with other developers and the Oniro team.
- Ask questions, seek help, and engage in discussions.
- Join the [Oniro Matrix channel]



matrix

Human zone!
AI not invited ☺️

Join Us in Shaping the Future

Contribute to Oniro development. Let's create an open, secure mobile ecosystem together.

Think Global and Code Local

oniroproject.org

Find me on LinkedIn