Accelerating Global OpenHarmony Adoption through Oniro

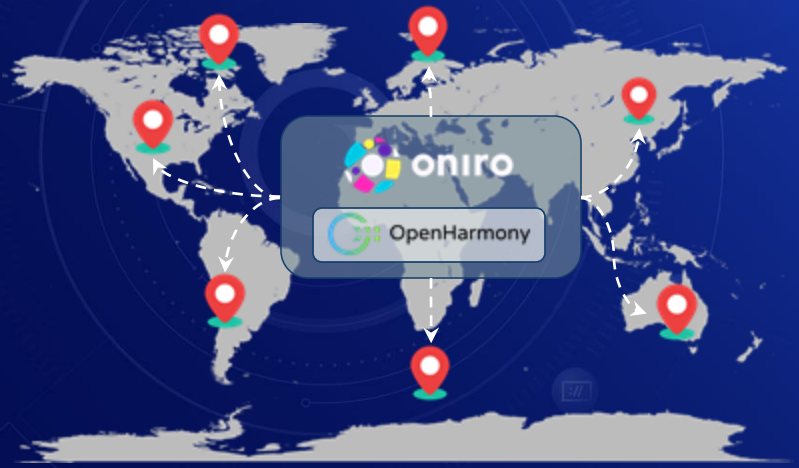
Tooling, Devices & Infrastructure

Francesco Pham

2025 OpenHarmony Technical Conference (Global)

Why Oniro?

- OpenHarmony: powerful & mature
- Global adoption needs **accessibility & reach**
- Oniro bridges core architecture → worldwide developer experience
- Built for **international contributors** and markets

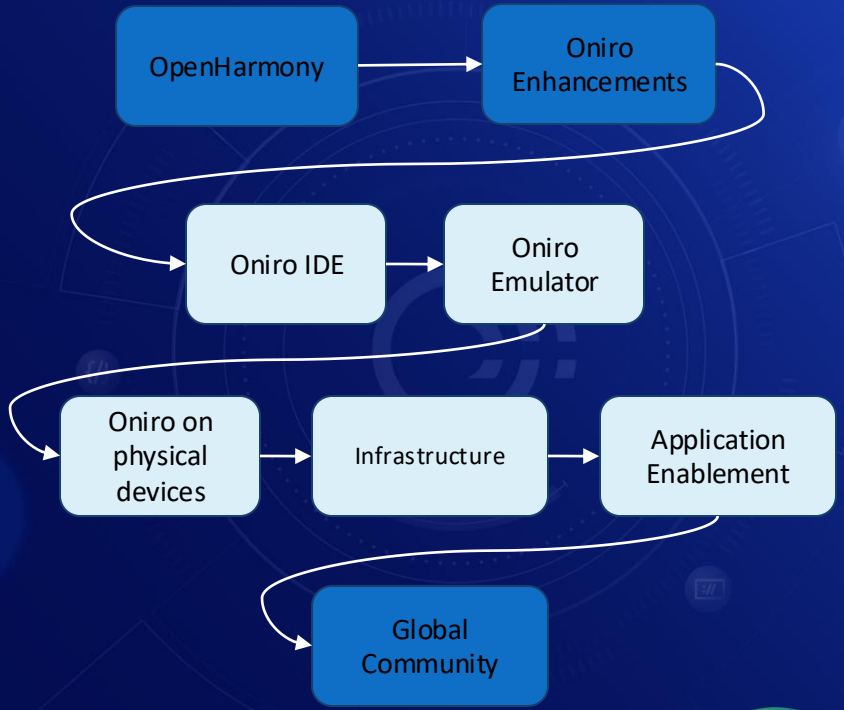


Oniro in the OpenHarmony Ecosystem

- Eclipse Foundation project building on OpenHarmony
- Enhancements for **global developer experience**
- Strong focus on **IP compliance** & licensing clarity
- Contributions flow **upstream**

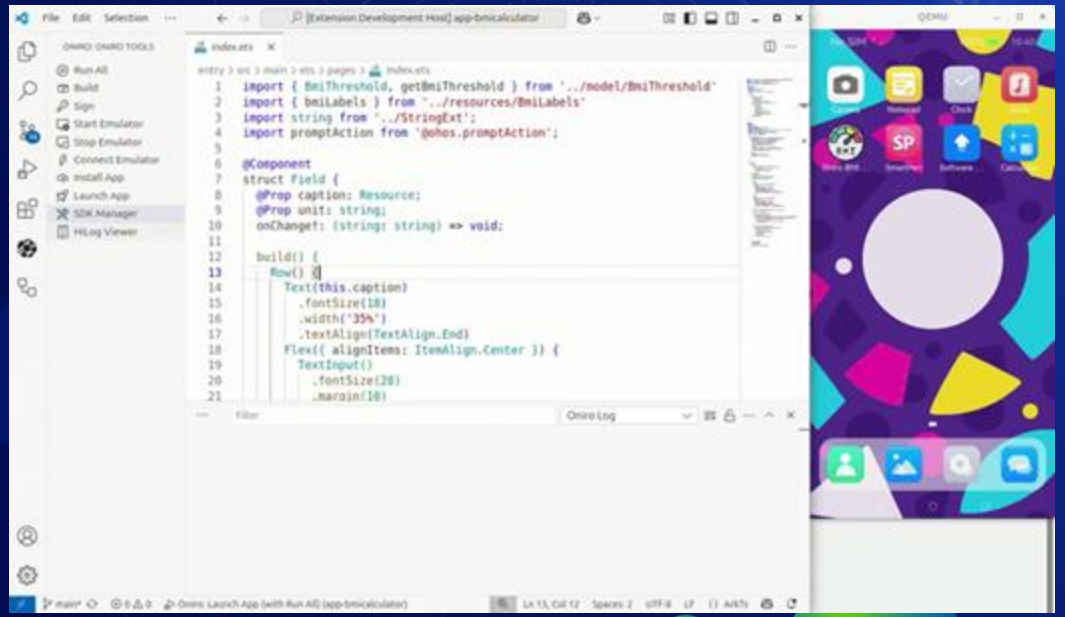
From Foundations to Global Adoption

- Oniro builds on **OpenHarmony's strong base**
- Adds **tooling, devices, infrastructure**
- Goal: **global, contributor-friendly ecosystem**



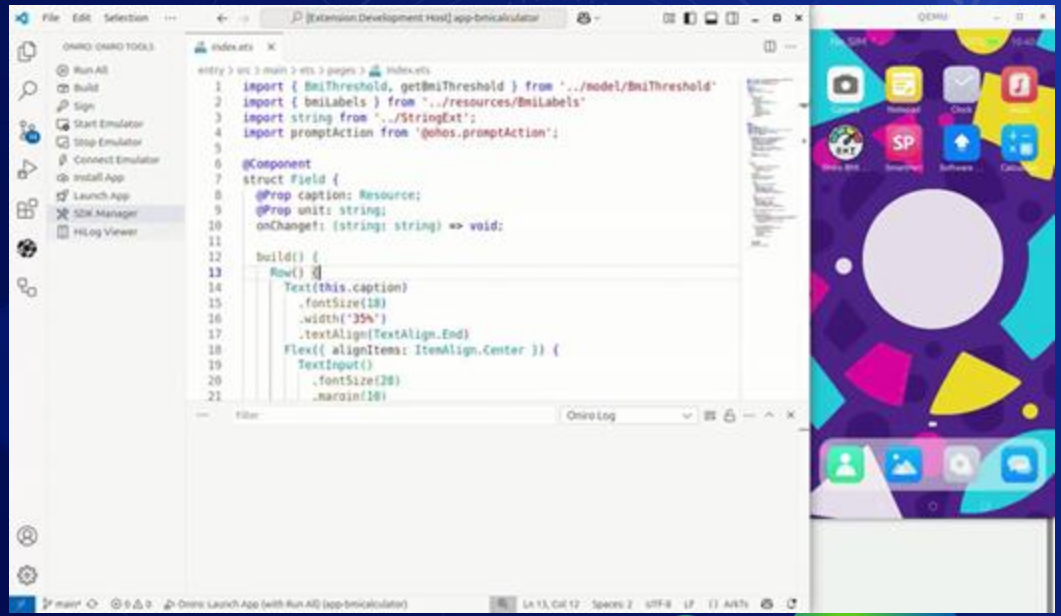
Oniro IDE: Developer-Friendly Workflow

- Lightweight **VS Code Extension**
- Build → Sign → Deploy → Run → Debug in one place
- Integrated SDK & emulator management
- Works for **app** & **system** developers



Oniro IDE Impact

- Reduces setup friction
- Accelerates iteration cycles
- Lowers entry barrier for new contributors
- Supports ArkTS app development



Emulation with QEMU

- Develop & test without physical hardware
- Oniro/OpenHarmony runs in virtualized environment
- Built with meta-openharmony Yocto layer
- Support for x86_68 architecture and Hypervisor virtualization
- Ideal for rapid prototyping & CI/CD testing
- Open source and easily accessible

Oniro on physical devices

- Verified Oniro support on:
  - **HiHope HH-SCDAYU200** (reference board)
  - **Raspberry Pi 4** (popular dev board)
  - **OnePlus 6T** (community port)
- Complements the **many products** already using OpenHarmony
- Offers developers **real hardware** for testing & innovation

Developer Phone Bring-Up

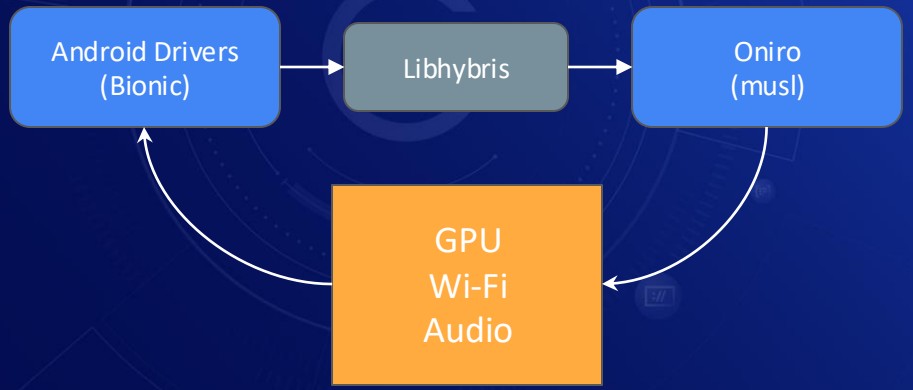
- Realistic mobile reference device
- Partner: Volla — open hardware & FOSS friendly
- LXC bring-up + Oniro patches on Android Common Kernel 5.10
- Display output (DRM) with software rendering
- Next: GPU accel, audio, sensors, telephony

Challenges of Mobile Hardware Porting

- Proprietary driver dependencies
- Android-centric hardware design
- Compatibility with non-Android OS

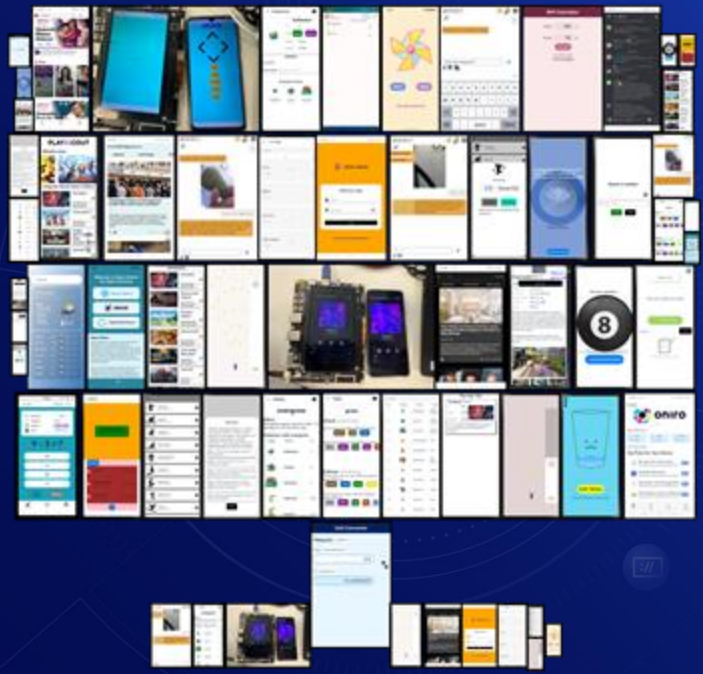
Bridging proprietary Android drivers for open systems

Libhybris



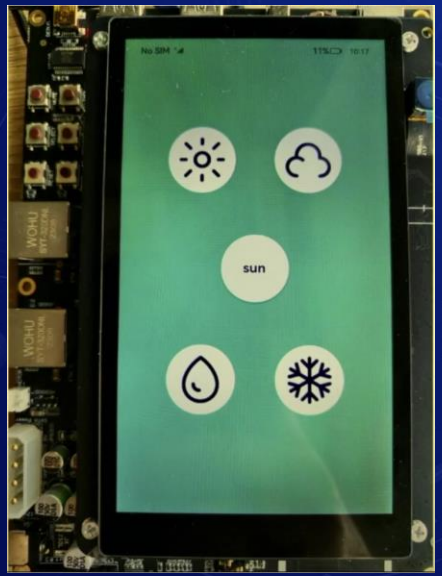
Growing the App Ecosystem

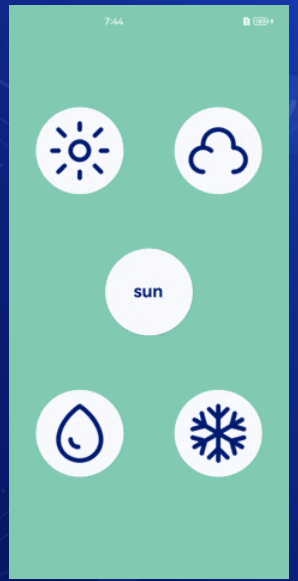
- Enhancing app development experience
- Oniro community app enablement: SoundCloud, Telegram, Discord…
- Open source app distribution: Oniro App Store



Growing the App Ecosystem – React Native

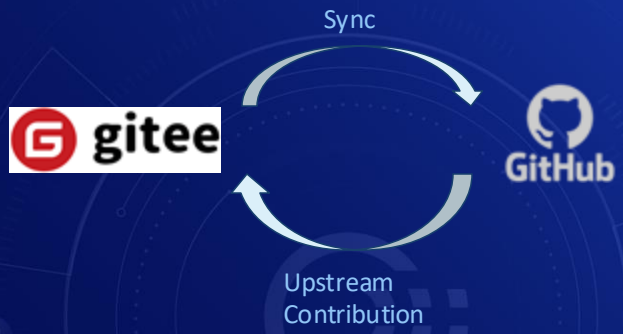
- **React Native** Support: Enabling cross-platform app development on Oniro.
- Build apps with **JavaScript** and **React**.
- Target **multiple platforms** with a single codebase.





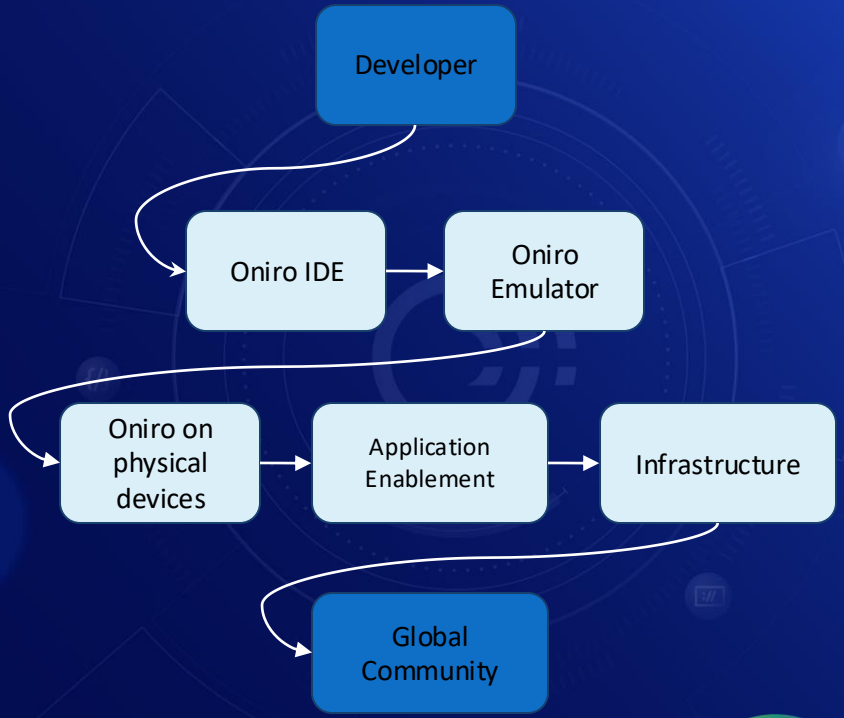
Global Infrastructure

- GitHub mirrors of OpenHarmony sources (daily sync)
- Improves access & transparency for global developers
- Low-latency code access outside China
- CI/CD workflows for testing App & System builds
- **English-first** contributor docs at docs.oniroproject.org



From Tooling to Global Adoption

- Every step reduces barriers
- Contributions flow upstream
- Target: accessible, privacy-respecting, open OS



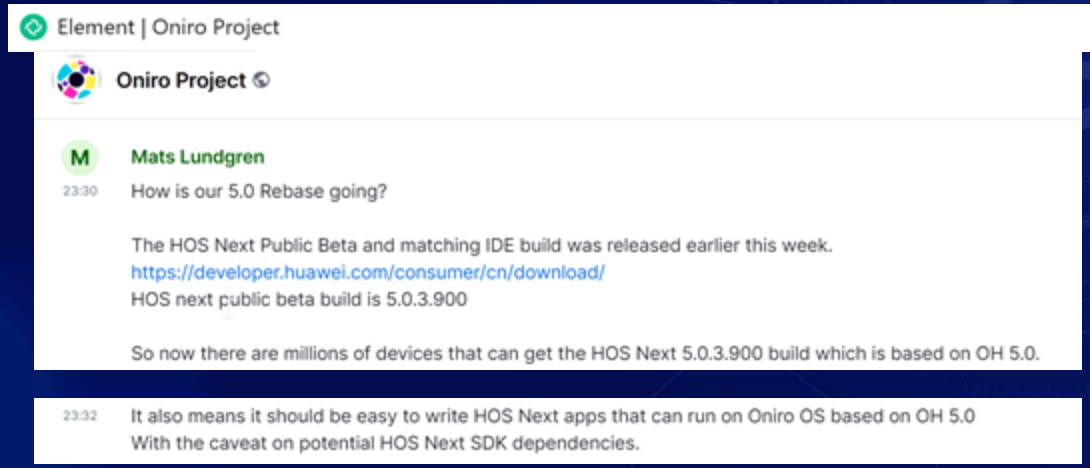
Roadmap

Future works:

- Richer app ecosystem
- 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