

From development boards to mobile phones Porting Oniro to Volla devices

Volla Community Days '25



Francesco Pham Oniro PMC Lead – Huawei

Introduction

- Supported Boards:
 - OpenHarmony supports boards like Hoperun Dayu200 (Rockchip).
 - Oniro adds support for Raspberry Pi.
- From development boards to mobile
- Goal: Oniro on Volla devices





Why Oniro on Volla?

- Alternative to traditional Android-based FOSS
- Cross-device interoperability
- Modular architecture for IoT to mobile
- Based on strong foundations of OpenHarmony

Cross-device Intelligent Interoperablity





- Mini system: low-memory devices (≥128 KiB), ARM Cortex-M or RISC-V processors
- Small system: ≥1 MiB of memory, ARM Cortex-A processors
- Standard system: ≥128 MiB of memory, typically ARM Cortex-A processors





Multi-kernel support

- LiteOS for mini system
- Linux for standard system
- And ...

Application layer	System applications	System applications Desktop Control I				ol bar	्रि Settings 🔍 Call										🔛 Extended/3rd-party application					
Framework ayer	Basic system subsyste M Application framework Ability framework	m set	ity -	<u>(f)</u>	0 0 0 0				vare se stem se			Ð	nced so subsys	tem se		Q		vare se system				
System service layer	 Distributed scheduler Distributed data management DosoftBus Ark multi-language runtir 	Multimodal input	Graphics	Security	Ы	Multimodal input	Telephony	Multimedia	DFX	MSDP & DV		Smart TV service	Wearable service	loT device service		Location	IAM	Wearable hardware	IoT hardware			
⊷ ernel layer	Kernel subsystem	Linux Ke		iernel a	bstraction Li	layer (K/ ite OS	AL)						💱 Dri	ver sub	system	Hardw	are Driv	er Fou	ndation	(HDF)		



HDF (Hardware Driver Foundation)

- a unified framework for hardware drivers abstraction
- Supports cross-platform compatibility and modular design.
- Includes standardized APIs (HDI), driver lifecycle management, and PnP support.

	Basic system subsyste					/are sei tem sei		Enha		oftware stem se	service _		Hardware service subsystem set						
Framework ayer		out 🛛	P	6	0-0 -0 -0		ිම	Ţ,	DFX		÷	0	e e		Q	ି	G	0-0	
System service layer	 Distributed scheduler Distributed data management DootBus 	Multimodal input	Graphics	Security	А	Multimodal input	Telephony	Multimedia	DFX	MSDP & DV	Smart TV service	Wearable service	loT device service		Location	IAM	Wearable hardware	IoT hardware	
	Ark multi-language runtin	ne (t	ð Utilit	ies															



Distributed Capabilities

- **DSoftBus**: Unified device discovery & communication.
- Distributed Data & Scheduler: Cross-device data sync & task management.
- **Device Virtualization**: Multiple devices act as one "Super Device".
- **Distributed File System** (HMDFS): Seamless remote file access.
- **One-Time Development**: Write once, run across device types.

ayer System applications Desktop 🔂 Control bar						💮 Settings 🔍 Call								S Extended/3rd-party applications					
ramework ayer			a ()					vare ser tem set			6	subsy:	oftware set	Q		vare ser ystem s	set		
ystem ervice layer	 Distributed scheduler Distributed data management DSoftBus Ark multi-language run 	Mattimodal	Graphics Security	AI	Multimodal input	Telephony	Multimedia	DFX	MSDP & DV		Smart TV service	Wearable service	loT device service	Location	IAM	Wearable hardware	IoT hardware	8	
+			Kernel a	abstraction	layer (KA	AL)				•									



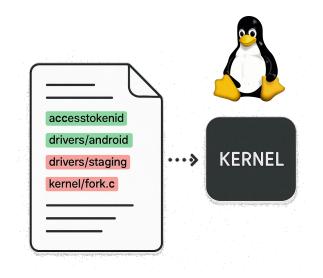
- ArkUI: A declarative UI framework (built-in UI components, Responsive layouts, animations and event handling mechanisms)
- ArkTS: A statically typed, compiled programming language extending TypeScript

Application ayer	System applications 💭 Desktop 🔁 Control					rol bar	ol bar 💮 Settings 🔍 Call									Extended/3rd-party applicatio					
ramework ayer	Basic system subsyste Crig ArkUI Application framework Same Ability framework	m set	ଜୁ	3	0 1 2 1 2 1 0				vare se			Ð	Ø	stem se		Q		vare se system			
System ervice layer	 Distributed scheduler Distributed data management DooftBus Ark multi-language runting 	me Multimodal input	Graphics	Security	AI	Multimodal input	Telephony	Multimedia	DFX	MSDP & DV		Smart TV service	Wearable service	loT device service	11	Location	IAM	Wearable hardware	IoT hardware		
+			K	(ernel a	bstractior	layer (K/	AL)														



OpenHarmony Kernel Patches

- Linux kernel 5.10 or 6.6
- Core Driver Infrastructure
 - Access tokens and Binder IPC
- Drivers for Diagnostics
- HDF (Hardware Driver Foundation)





Mainline Support and Demos







OpenHarmony on OnePlus 6T



Oniro on gemu-based emulator



Porting Oniro to Volla X23

Current status

- Setup of Oniro within LXC
- Applied Oniro patches to Android Common Kernel 5.10
- Graphical support with software rendering and DRM display output.
- Setting up system and library dependencies.





Porting Oniro to Volla X23

Next steps

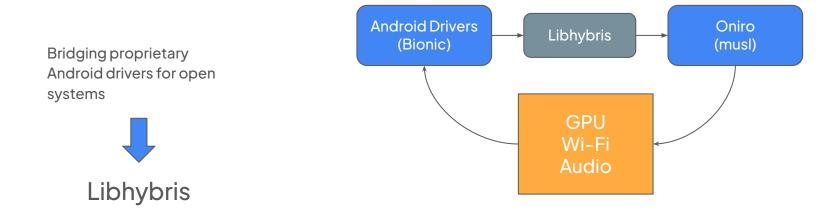
- Adapting drivers to Oniro
- Enable graphical acceleration
- Enable audio, sensors, telephony.
- Create a flashable image





Challenges of Mobile Hardware Porting

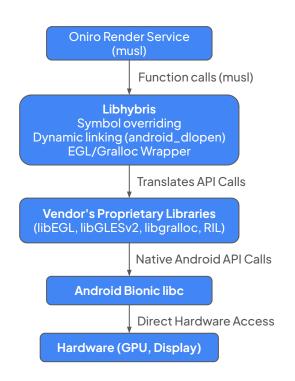
- Proprietary driver dependencies
- Android-centric hardware design
- Compatibility with non-Android OS





Current Status of Libhybris Integration

- Adapting it to musl (not glibc).
- Manage build system differences
- Integrating with OpenHarmony GN build system
- Managing file system structure differences





Roadmap and Broader Impact

Near term goals:

- Refine Oniro integration on Volla X23
- Transition to latest OpenHarmony version

Long term goals:

- Oniro on more phones and form factors: Volla Phone Quintus/Tablet
- Inspire an open, secure, intelligent mobile platform built in Europe.



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

Think Global and Code Local



oniroproject.org





Find me on LinkedIn



