



# oniro

## Introduction to Oniro OpenHarmony Graphics Stack and Device Bring-Up

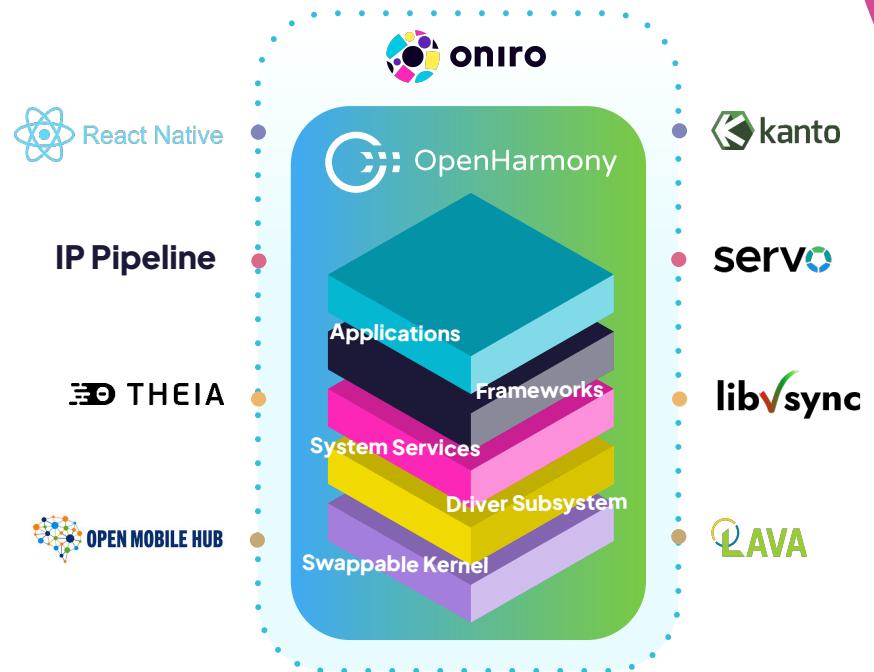
ASPLOS'25/EuroSys'25 Tutorial



Francesco Pham  
Oniro PMC Lead - Huawei

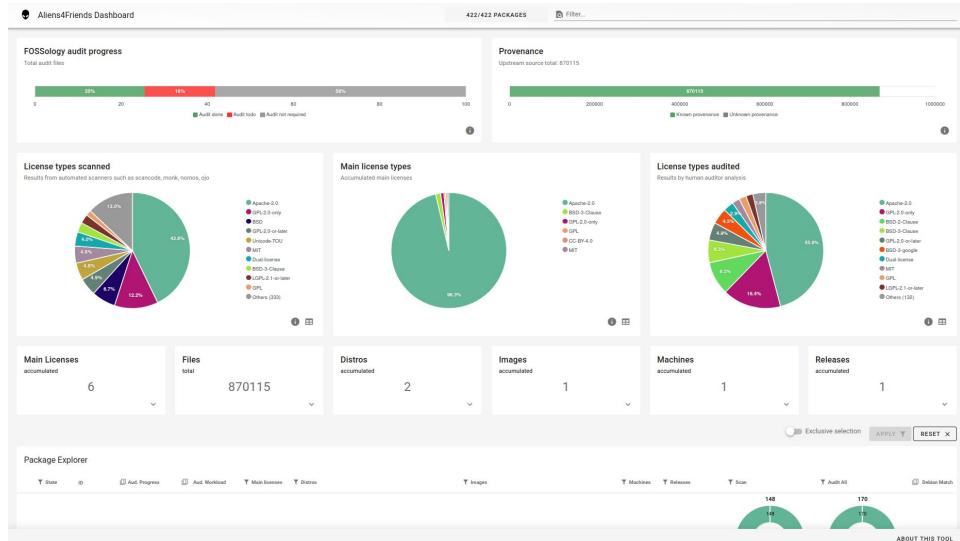
# What is Oniro?

- Open-source, vendor-neutral OS managed by the Eclipse Foundation
- Extends OpenHarmony with features for global adoption
- Collaboration between Eclipse & OpenAtom Foundations for transparency & community-driven development



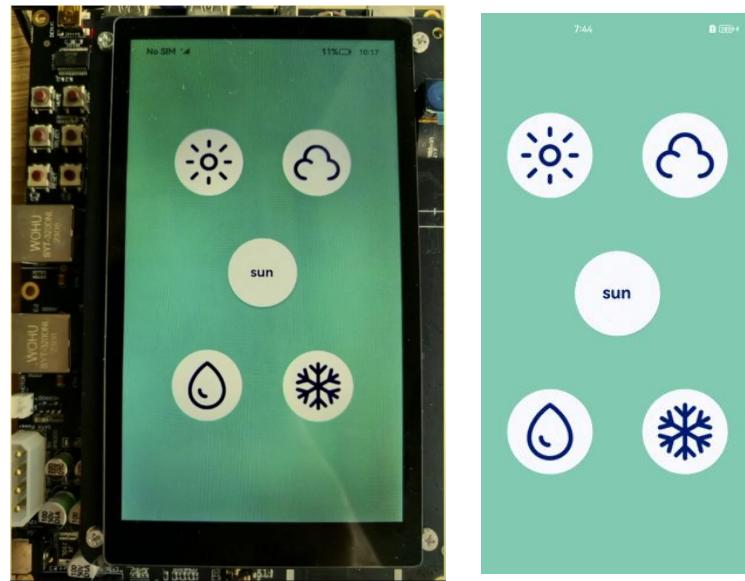
# How Oniro Enhances OpenHarmony – IP Compliance

- Ensures compliance with intellectual property & licensing standards.
- Uses an advanced IP compliance toolchain.
- Facilitates worry-free development and distribution.



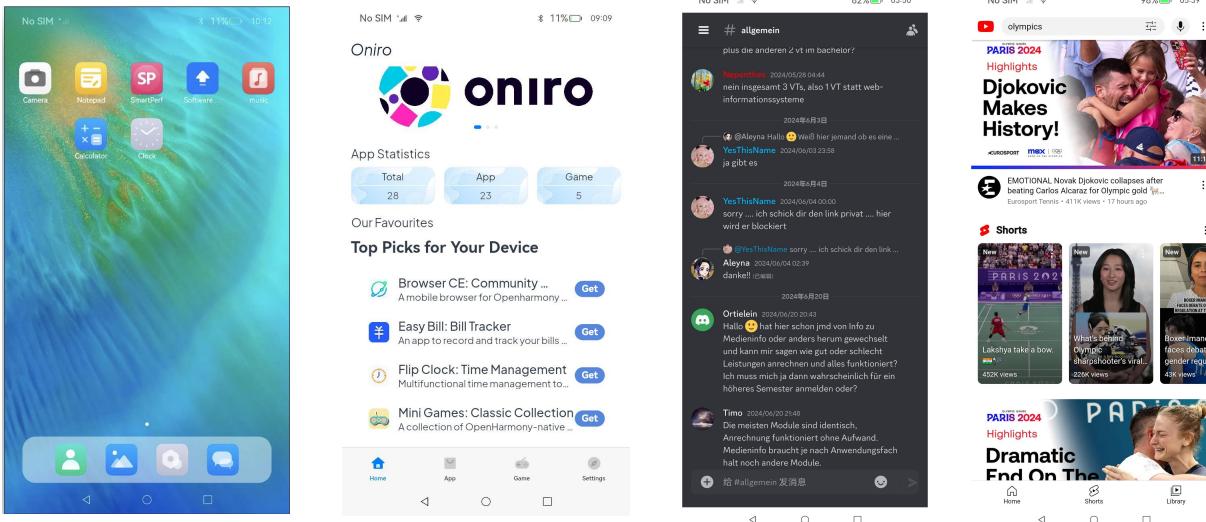
# How Oniro Enhances OpenHarmony – React Native

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



# How Oniro Enhances OpenHarmony - App Store

- Oniro App Store: Creating an open-source app distribution platform.
- Open and transparent distribution channel.
- Reach a wider audience for your Oniro applications.



# Oniro Mirrors

OpenHarmony

OpenHarmony是由开放原子开源基金会（OpenAtom Foundation）孵化及运营的开源项目，目标是面向全场景、全连接、全智能时代，搭建一个和平台，促进万物互联产业的繁荣发展。

<https://openharmony.io> [contact@openharmony.io](mailto:contact@openharmony.io)

Overview Repositories 717 Issues 16723 Pull Requests 7482 Feeds People 212

社区已于2024-09-29发布OpenHarmony-v5.0.0-Release版本。详情请参考<https://gitee.com/openharmony/docs/blob/master/zh-cn/release-notes/OpenHarmony-v5.0.0-release.md>



Eclipse Oniro Mirrors

oniro

Overview Repositories 714 Projects Packages People

README .md

## Welcome to the Eclipse Oniro Mirrors

This space mirrors all repositories from [the OpenHarmony organization at gitee](#). These mirrored repositories are **read-only** and consumed by Eclipse Oniro build system.



OpenHarmony

OpenHarmony is a large-scale open-source project. It follows a layered architecture, starting from the hardware layer at the bottom, followed by the framework layer, and application layer from the bottom up. System functions are expanded by levels, from system to subsystem, and further to component. In a multi-device deployment scenario, unnecessary components can be excluded from the system as required. The following figure shows the technical architecture of OpenHarmony.

• OpenHarmony codebase at Gitee

Rust 15% TypeScrip 4% Python 4%

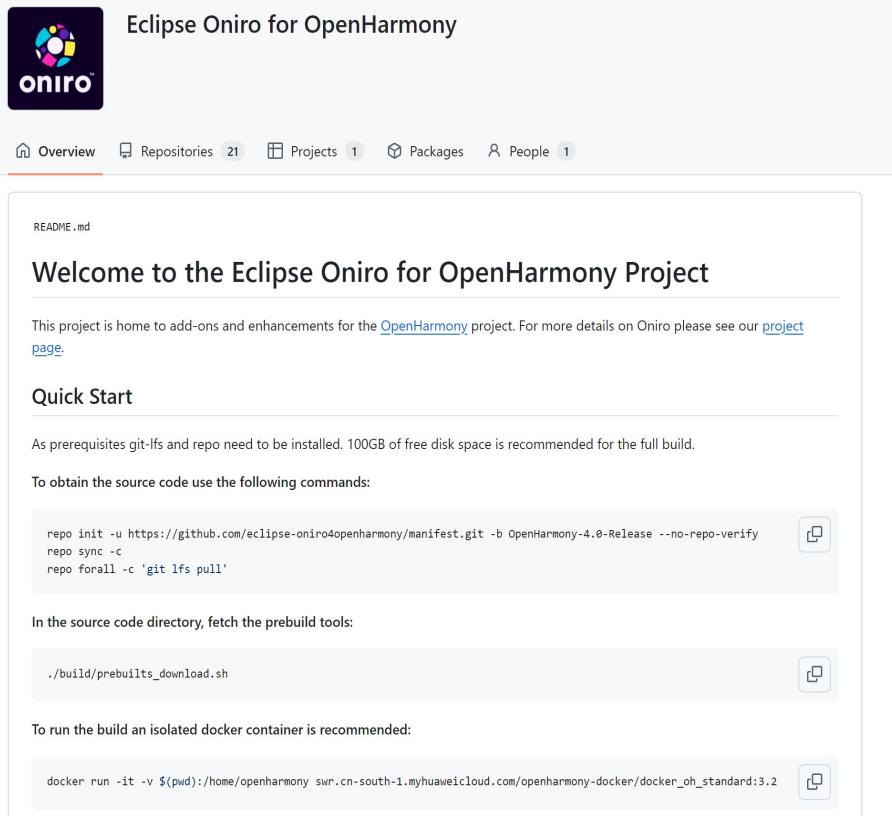
- OpenHarmony mirror on GitHub
- Daily synchronization
- Quick, reliable and convenient access for a vast GitHub developer community
- Read-only

Dockerfile 1 C++ 1

startup\_init Public third\_party\_abseil-cpp Public



# Oniro Repositories



The screenshot shows the GitHub repository page for "Eclipse Oniro for OpenHarmony". The repository has 21 repositories, 1 project, 1 package, and 1 person. The README.md file contains the following content:

**Welcome to the Eclipse Oniro for OpenHarmony Project**

This project is home to add-ons and enhancements for the [OpenHarmony](#) project. For more details on Oniro please see our [project page](#).

### Quick Start

As prerequisites git-lfs and repo need to be installed. 100GB of free disk space is recommended for the full build.

To obtain the source code use the following commands:

```
repo init -u https://github.com/eclipse-oniro4openharmony/manifest.git -b OpenHarmony-4.0-Release --no-repo-verify
repo sync -c
repo forall -c 'git lfs pull'
```

In the source code directory, fetch the prebuild tools:

```
./build/prebuilt_download.sh
```

To run the build an isolated docker container is recommended:

```
docker run -it -v $(pwd):/home/openharmony swr.cn-south-1.myhuaweicloud.com/openharmony-docker/docker_oh_standard:3.2
```

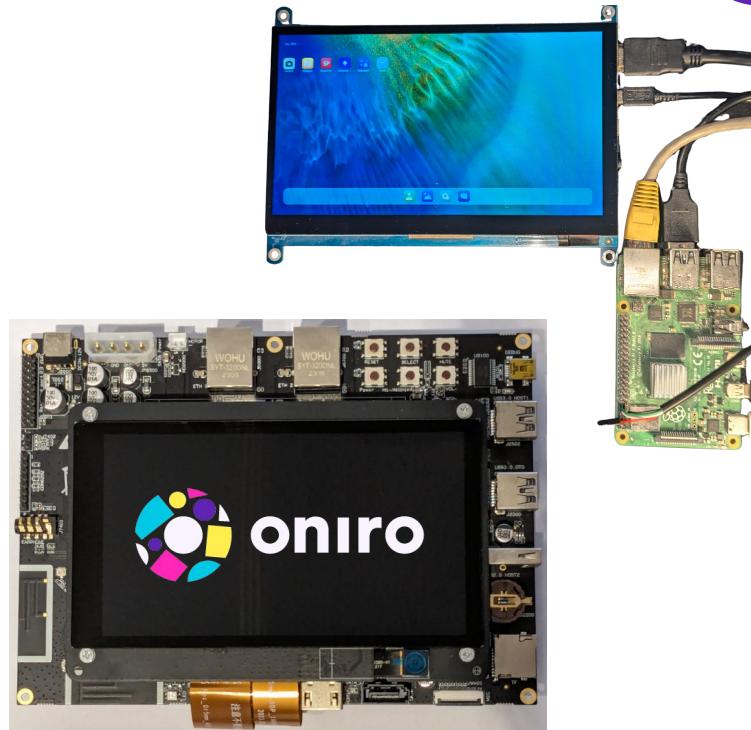
- The Eclipse Oniro codebase
- The code developed by the Oniro community

Think Global  
Code Local



# Hardware enablement – Supported Boards

- Importance of Development Boards: Allow developers to quickly build and test applications on their preferred hardware.
- Supported Boards:
  - OpenHarmony supports boards like Hoperun Dayu200 (Rockchip).
  - Oniro adds support for Raspberry Pi.



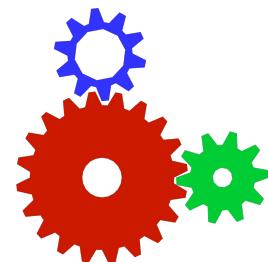
# Hands-on Demo – Oniro on Raspberry Pi

- Goal: Flash and boot Oniro/OpenHarmony on Raspberry Pi
- Steps:
  - a. Clone Oniro repositories
  - b. Build the OS image
  - c. Flash to Raspberry Pi



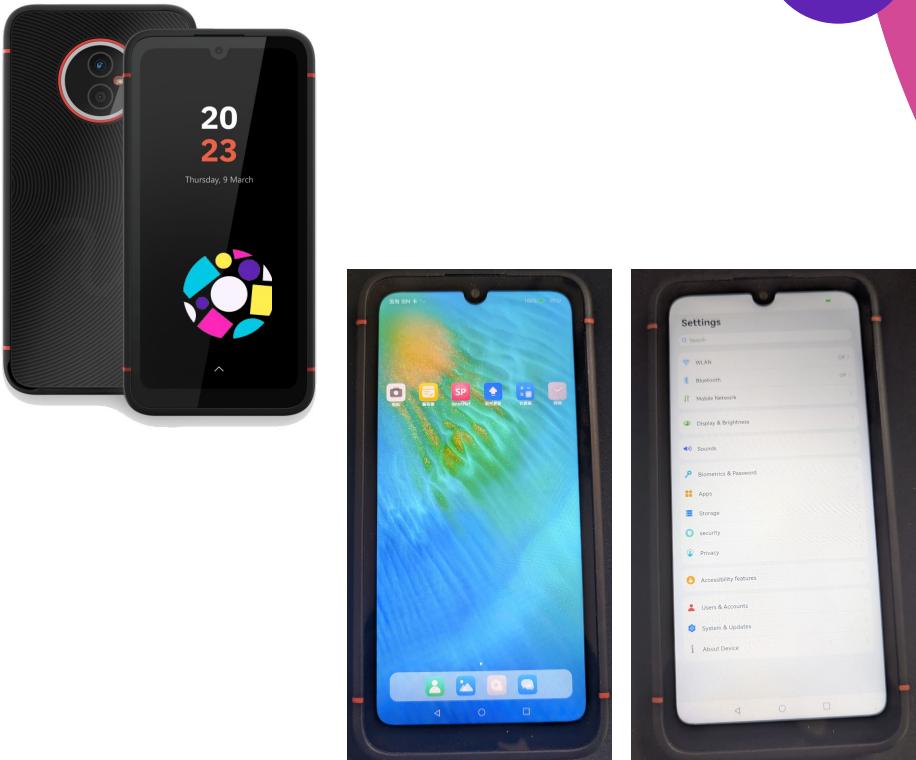
# Hardware enablement – Rapid Prototyping

- **Software Rasterization with Mesa 3D:**
  - Provides a GPU-independent rendering solution for early development
  - Ideal for platforms lacking native GPU driver support
- **LXC Containers:**
  - Enables running OpenHarmony within existing environments for faster iterations
  - Simplifies testing and debugging on diverse hardware



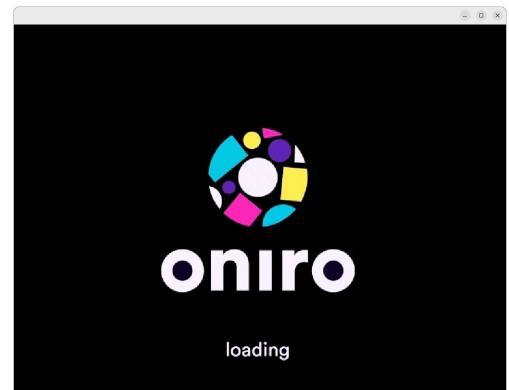
# Bringing Oniro to Mobile Phones

- **Target Devices:** Ongoing work on Volla X23 and other mobile developer phones
- **Challenges:**
  - Proprietary GPU drivers and lack of open-source alternatives
  - Bootloader and hardware abstraction limitations
- **Libhybris Integration:**
  - Bridges Android GPU drivers to OpenHarmony for enhanced compatibility



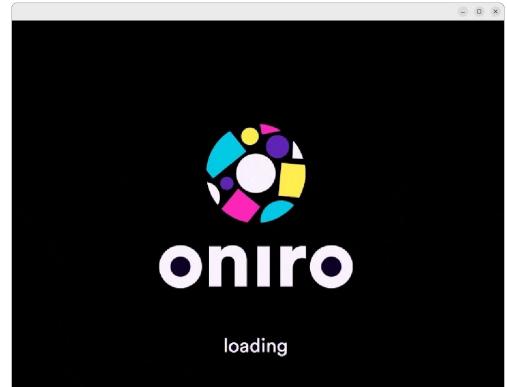
# Hardware Enablement - QEMU

- **Why QEMU?** Ideal for testing OpenHarmony on virtual hardware
- **meta-openharmony Yocto Layer:**
  - Simplifies hardware enablement for QEMU and other platforms
  - Facilitates rapid prototyping and testing
- **Benefits:**
  - Enables testing applications without the need for physical devices
  - Facilitates a faster development workflow



# Hands-on Demo – Oniro on QEMU

- Why QEMU? Ideal for testing OpenHarmony on virtual hardware
- Benefits:
  - Enables testing applications without the need for physical devices
  - Facilitates a faster development workflow
- Demo:
  - Boot Oniro
  - Install and run apps

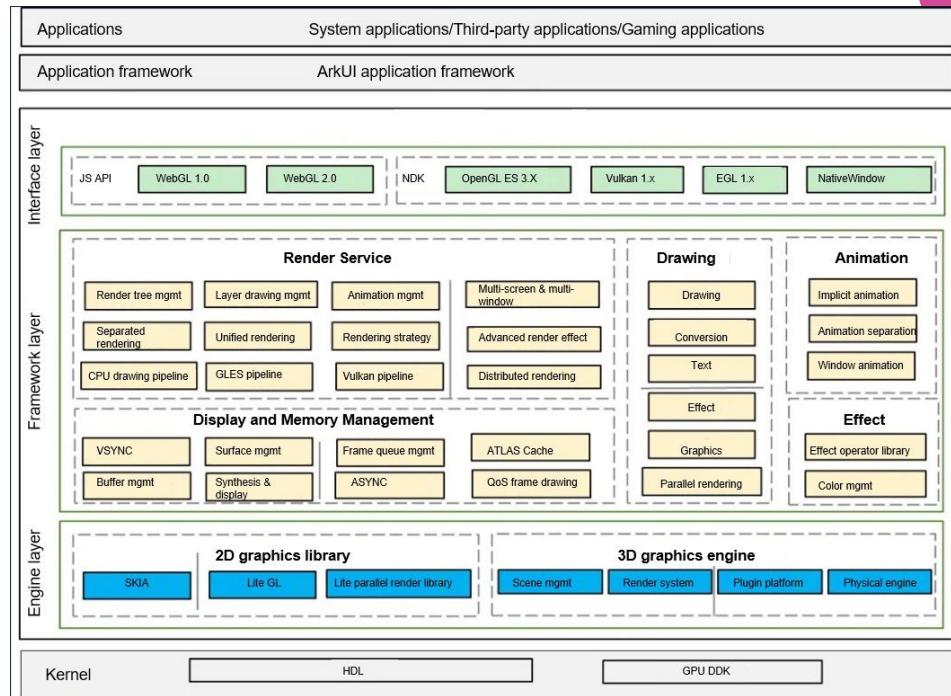


# Linux Kernel Patches for OpenHarmony

- **Core Driver Infrastructure:**
  - `drivers/accesstokenid`: Manages access tokens for tasks, enhancing security and access control.
  - `drivers/android`: Modifications to Binder IPC for debugging and optimizations.
- **Staging Drivers for Diagnostics:**
  - `blackbox`: Fault logging framework for debugging crashes.
  - `hievent` and `hilog`: Event and general-purpose logging mechanisms.
  - `hungtask` and `zerohung`: Advanced hung task detection and system responsiveness tools.
- **Kernel Modifications:**
  - `kernel/fork.c`: Adds support for `token` and `f-token` fields in `task_struct`.
- **Focus:** Enhances diagnostics, debugging, and OpenHarmony integration.

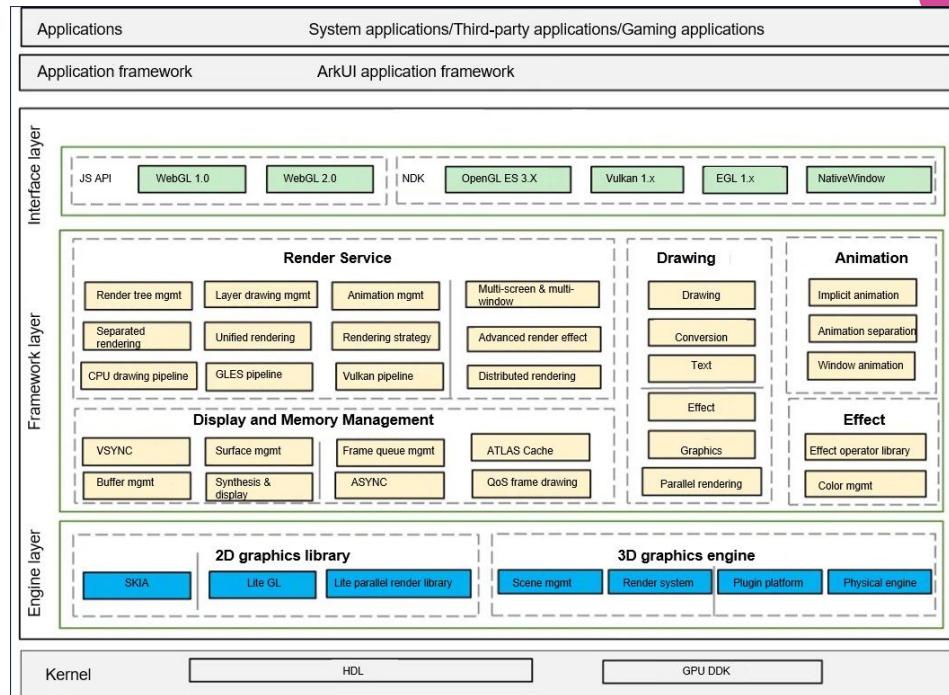
# Why Graphics Matter in OpenHarmony?

- Essential for usability, developer adoption, and user experience
- Mobile & IoT Devices: Flexible graphics pipeline to support diverse hardware
- Challenges: Hardware abstraction, performance optimization, and multi-device enablement



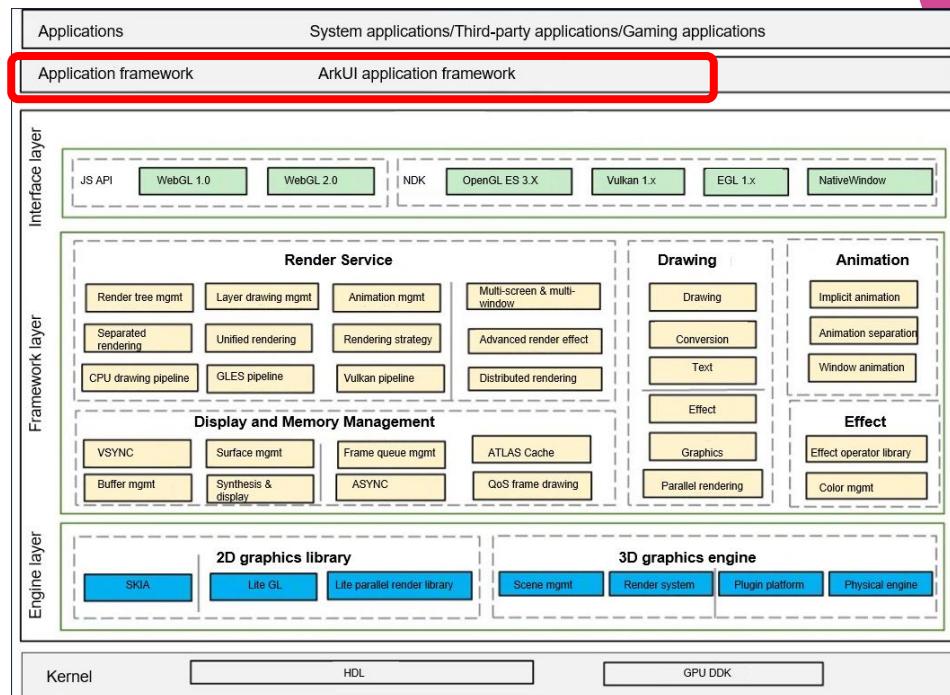
# OpenHarmony Graphics Stack Overview

- **ArkUI Framework:** UI toolkit with XComponent for OpenGL and advanced 3D rendering
- **Render Service:** Central compositing engine for optimized rendering
- **Window Manager:** Manages windows, transitions, and rendering policies
- **Display & Memory Management:** HDI abstraction for hardware compatibility



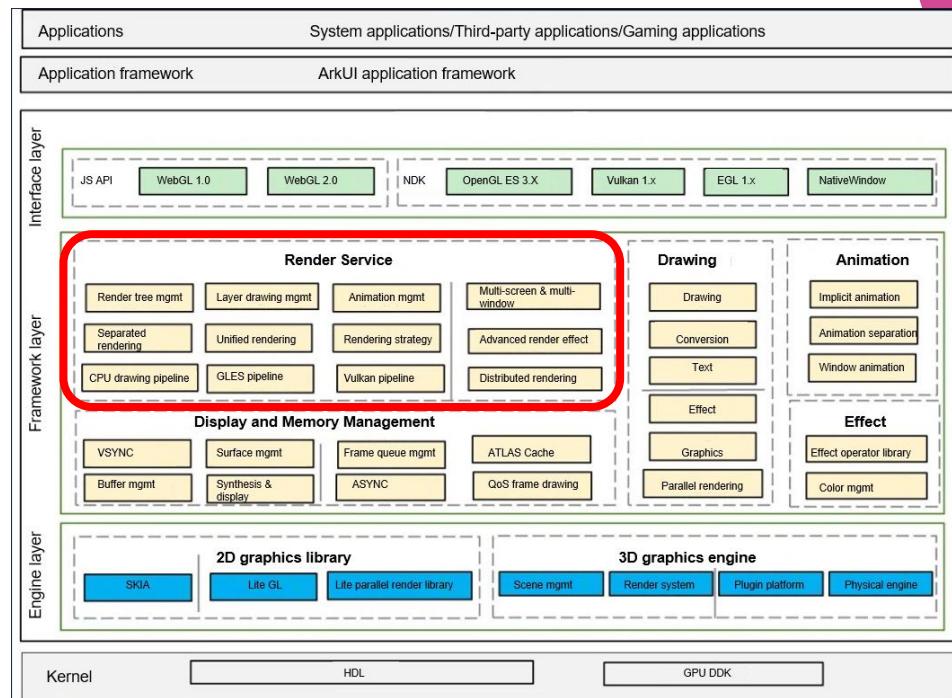
# ArkUI Framework & Interface Layer

- ArkUI: Declarative UI framework
- Supports JS API, WebGL, NDK
- Enables 3D rendering via XComponent + NativeWindow
- Integrates with GLES, Vulkan, OpenGL ES



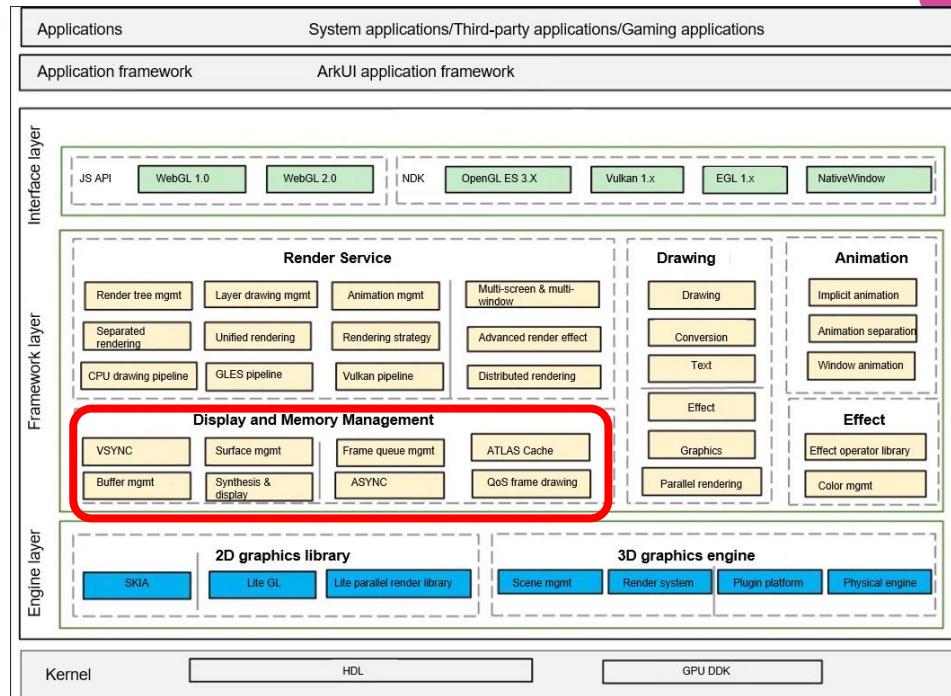
# Render Service – The Heart of Compositing

- Central rendering engine (replaces Wayland)
- Converts ArkUI → Drawing Tree → GPU pipeline
- Supports GLES, Vulkan, CPU pipelines
- Optimized for multi-window, animation, effects



# Display Framework & Hardware Enablement

- Display & Memory Mgmt: VSYNC, buffer, surface
- HDI: GPU abstraction via Composer & Buffer APIs
- GPU driver adaptation via HDF
- Display HAL + LCD Panel Driver for low-level control



# Future Roadmap

- Short-term:
  - Enable GPU acceleration (move from software rendering)
  - Improve libhybris compatibility
- Long-term:
  - Support new devices (beyond Raspberry Pi & Volla X23)
  - Optimize graphical performance

# Why Oniro Matters for Open Source & Research

- Enables FOSS-based mobile/IoT development
- Bridges OpenHarmony with global open-source communities
- Potential for research in OS performance, security, and UI optimization

# Oniro Documentation

- Comprehensive documentation is available to help you get started quickly.
- Find tutorials, API references, and more.
- Link:  
<https://docs.oniroproject.org/>



The screenshot shows a web browser displaying the URL [docs.oniroproject.org/contributing.html](https://docs.oniroproject.org/contributing.html). The page has a header with the Oniro logo and a search bar. A sidebar on the left contains a navigation menu with sections like Eclipse Oniro Project, Developer Boards, Contributing (which is expanded to show sub-sections: Quick start guide for new developers, Eclipse Contributor Agreement, DCO-signoff, GitHub Contributions, Bug Handling Process, REUSE Compliance, and Contributing to Projects not Maintained by Team), and Oniro IDE and Oniro Legacy Documentation. The main content area is titled "Contribution" and discusses the contributing process and guidelines. Below it is a "TABLE OF CONTENTS" section listing the same sub-sections as the sidebar.

docs.oniroproject.org/contributing.html

Search Oniro

Template Repository

## Contribution

The contributing process and guidelines part of this document must be applied to any repository in the scope of the project. Each repository must include this information in its `CONTRIBUTING.md` file and optionally, complement the process and guidelines with repository-specific requirements.

### TABLE OF CONTENTS

- Quick start guide for new developers
- Eclipse Contributor Agreement
- DCO-signoff
- GitHub Contributions
- Bug Handling Process
- REUSE Compliance
- Contributing to Projects not Maintained by Team

# Oniro Community

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



Element | Oniro Project



Oniro Project



Mats Lundgren

23:30

How is our 5.0 Rebase going?

The HOS Next Public Beta and matching IDE build was released earlier this week.

<https://developer.huawei.com/consumer/cn/download/>

HOS next public beta build is 5.0.3.900

So now there are millions of devices that can get the HOS Next 5.0.3.900 build which is based on OH 5.0.

23:32

It also means it should be easy to write HOS Next apps that can run on Oniro OS based on OH 5.0.

With the caveat on potential HOS Next SDK dependencies.

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

