Next-Generation Digital Cockpit Design with QNX

Randy Martin
Senior Technical Product Manager

Digital Cockpit Development Challenges

Technology Challenges

SoC's and capabilities rapidly evolving

- Portability between SoC vendors?

Base software technologies changing rapidly

- Quarterly Android Automotive releases
- Frequent Linux releases
- Evolving frameworks and APIs

Development Team Challenges

Large, distributed development teams

Varying demand for test environments and targets

Need for dynamic scaling to accommodate team changes

Need for efficiency in DevOps

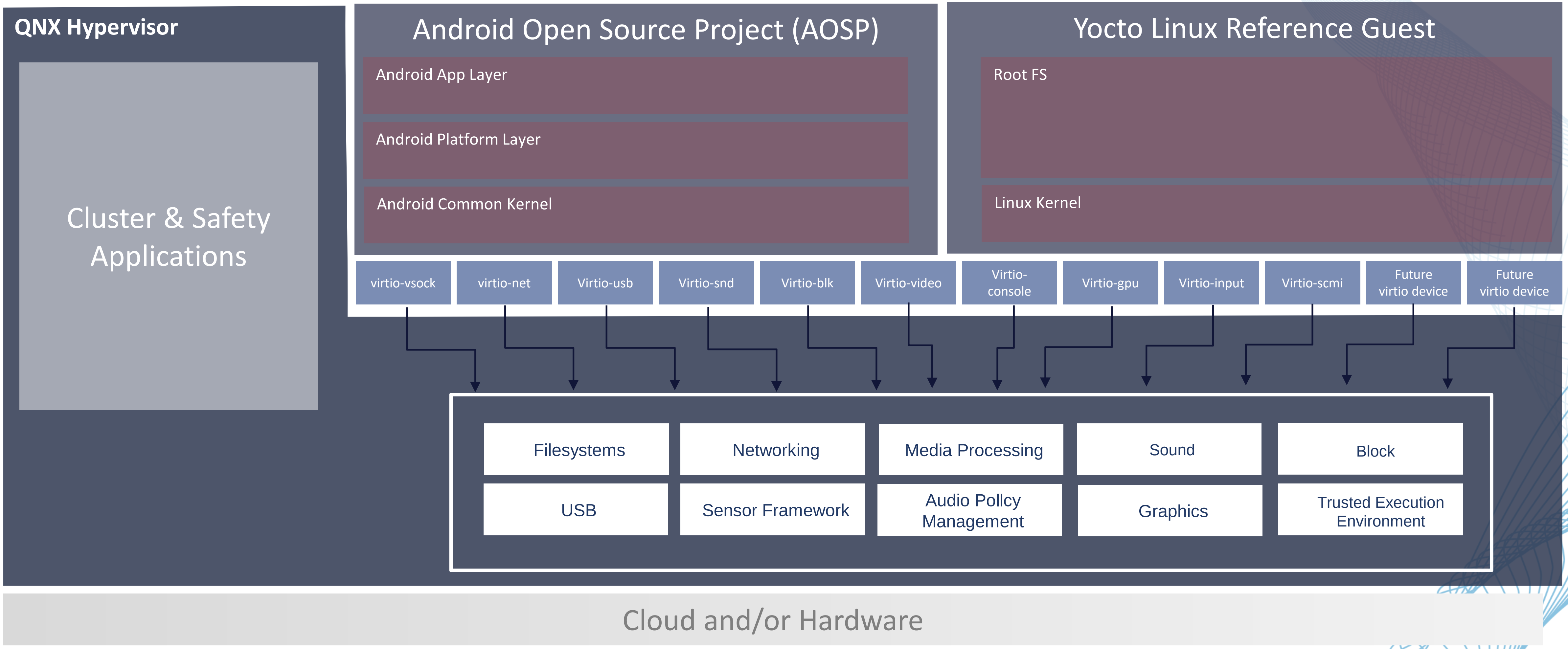
Need to manage development costs

Market Challenges

Building compelling experience for consumers

Managing feature set and costs for differing makes & models

QNX Advanced Architecture and Virtualization Framework



QNX Hypervisor

Cluster & Safety Applications

Android Open Source Project (AOSP)

Android App Layer

Android Platform Layer

Android Common Kernel

Yocto Linux Reference Guest

Root FS

Linux Kernel

virtio-vsock

virtio-net

Virtio-usb

Virtio-snd

Virtio-blk

Virtio-video

Virtio-console

Virtio-gpu

Virtio-input

Virtio-scmi

Future virtio device

Future virtio device

Filesystems

Networking

Media Processing

Sound

Block

USB

Sensor Framework

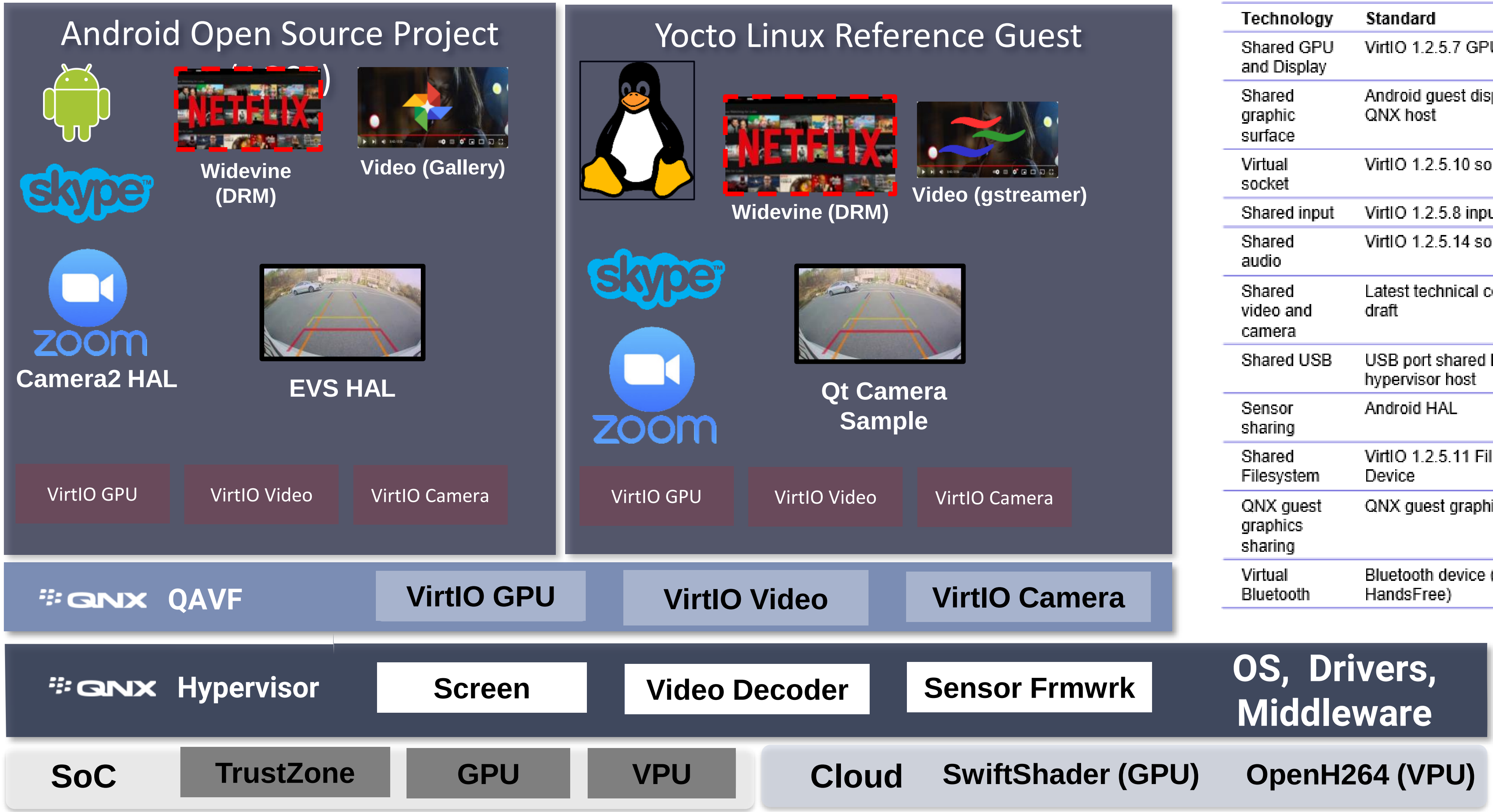
Audio Pollcy Management

Graphics

Trusted Execution Environment

Cloud and/or Hardware

QNX Advanced Virtualization Framework – Evolution
(Part 1 of 2)



Android Open Source Project

Widevine (DRM)

Video (Gallery)

Camera2 HAL

EVS HAL

VirtIO GPU

VirtIO Video

VirtIO Camera

Yocto Linux Reference Guest

Widevine (DRM)

Video (gstreamer)

Qt Camera Sample

VirtIO GPU

VirtIO Video

VirtIO Camera

QNX QAVF

VirtIO GPU

VirtIO Video

VirtIO Camera

QNX Hypervisor

Screen

Video Decoder

Sensor Frmwrk

OS, Drivers, Middleware

SoC

TrustZone

GPU

VPU

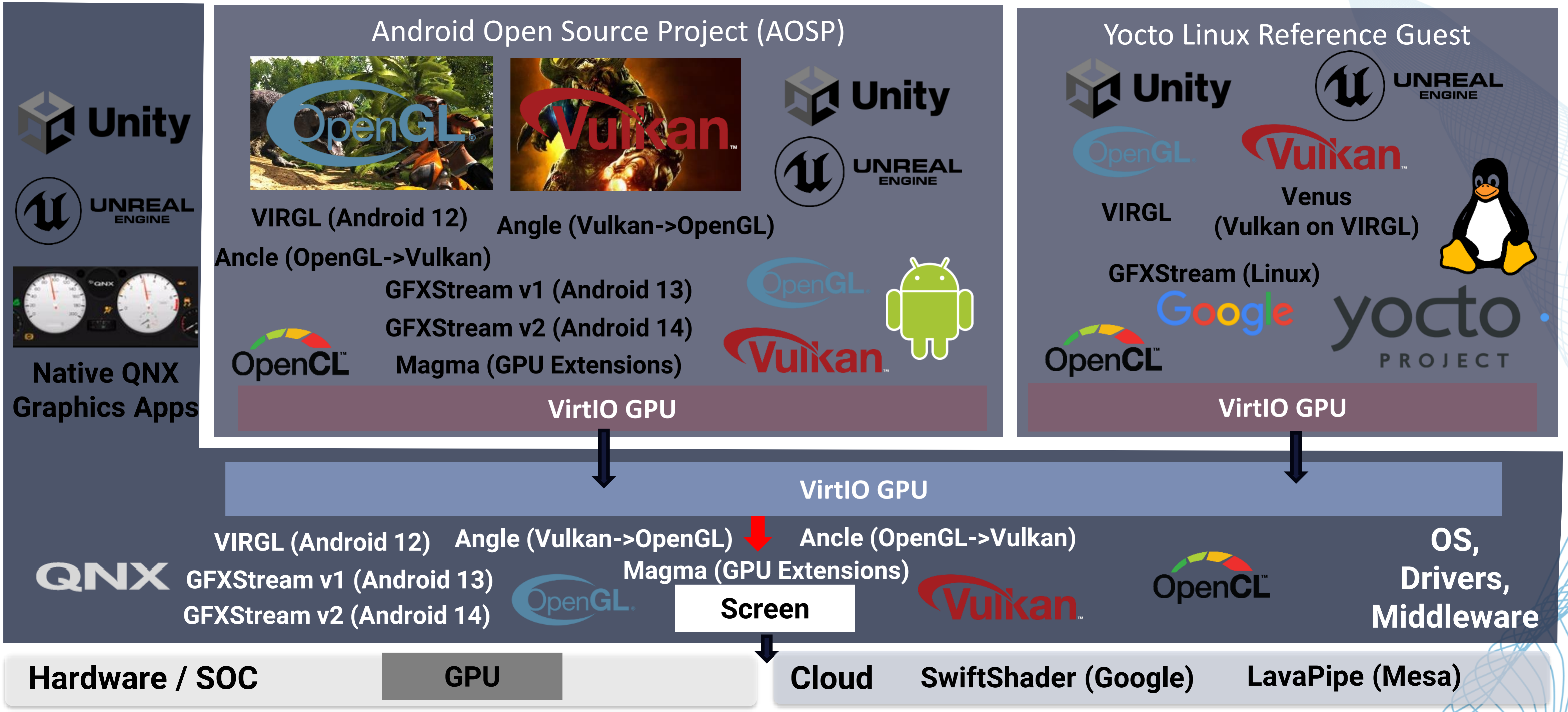
Cloud

SwiftShader (GPU)

OpenH264 (VPU)

Technology	Standard	Guest Support
Shared GPU and Display	VirtIO 1.2.5.7 GPU Device	Android and Linux
Shared graphic surface	Android guest display in QNX host	Android and Linux
Virtual socket	VirtIO 1.2.5.10 socket device	Android and Linux
Shared input	VirtIO 1.2.5.8 input device	Android and Linux
Shared audio	VirtIO 1.2.5.14 sound device	Android and Linux
Shared video and camera	Latest technical committee draft	Android HAL specific (Linux requires integration)
Shared USB	USB port shared by guest or hypervisor host	Android and Linux
Sensor sharing	Android HAL	Android
Shared Filesystem	VirtIO 1.2.5.11 File System Device	Android and Linux
QNX guest graphics sharing	QNX guest graphics	QNX
Virtual Bluetooth	Bluetooth device (media, HandsFree)	Android (Linux requires integration)

QNX Advanced Virtualization Framework – Evolution
(Part 2 of 2)



Native QNX Graphics Apps

Android Open Source Project (AOSP)

VIRGL (Android 12)

Angle (Vulkan->OpenGL)

Ancle (OpenGL->Vulkan)

GFXStream v1 (Android 13)

GFXStream v2 (Android 14)

Magma (GPU Extensions)

VirtIO GPU

Yocto Linux Reference Guest

VIRGL

Venus
(Vulkan on VIRGL)

GFXStream (Linux)

VirtIO GPU

VirtIO GPU

VIRGL (Android 12)

Angle (Vulkan->OpenGL)

Ancle (OpenGL->Vulkan)

GFXStream v1 (Android 13)

GFXStream v2 (Android 14)

Magma (GPU Extensions)

Screen

OS, Drivers, Middleware

Hardware / SOC

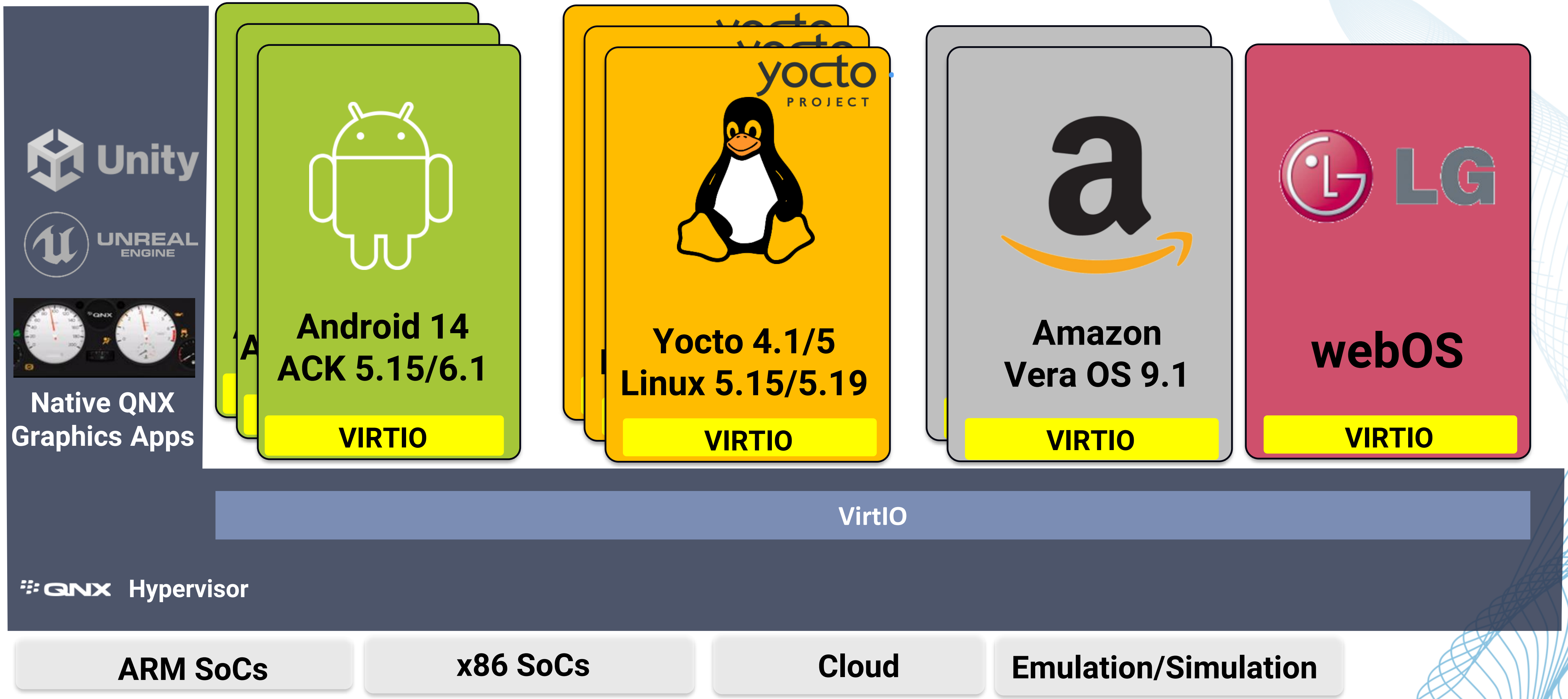
GPU

Cloud

SwiftShader (Google)

LavaPipe (Mesa)

QNX Managers Complexity



Native QNX Graphics Apps

Android 14
ACK 5.15/6.1

VIRTIO

Yocto 4.1/5
Linux 5.15/5.19

VIRTIO

Amazon
Vera OS 9.1

VIRTIO

webOS

VIRTIO

VirtIO

QNX Hypervisor

ARM SoCs

x86 SoCs

Cloud

Emulation/Simulation

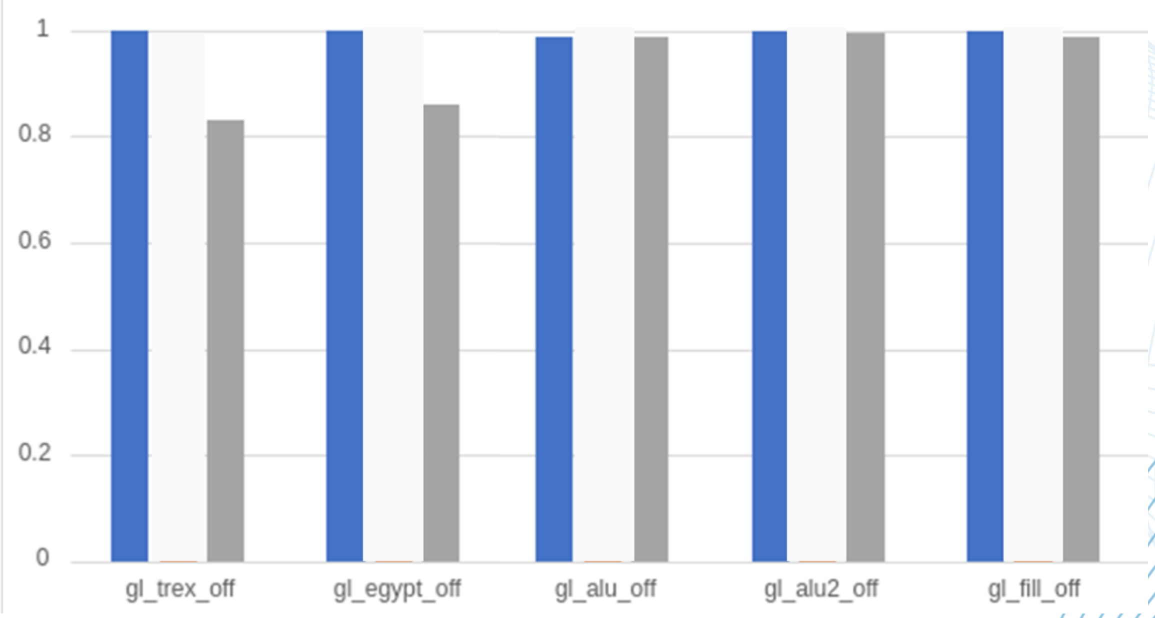
VirtIO Performance: Graphics

GFXBench: ON Screen

VirtIO (Android Guest) = QNX Native (host)

60 FPS

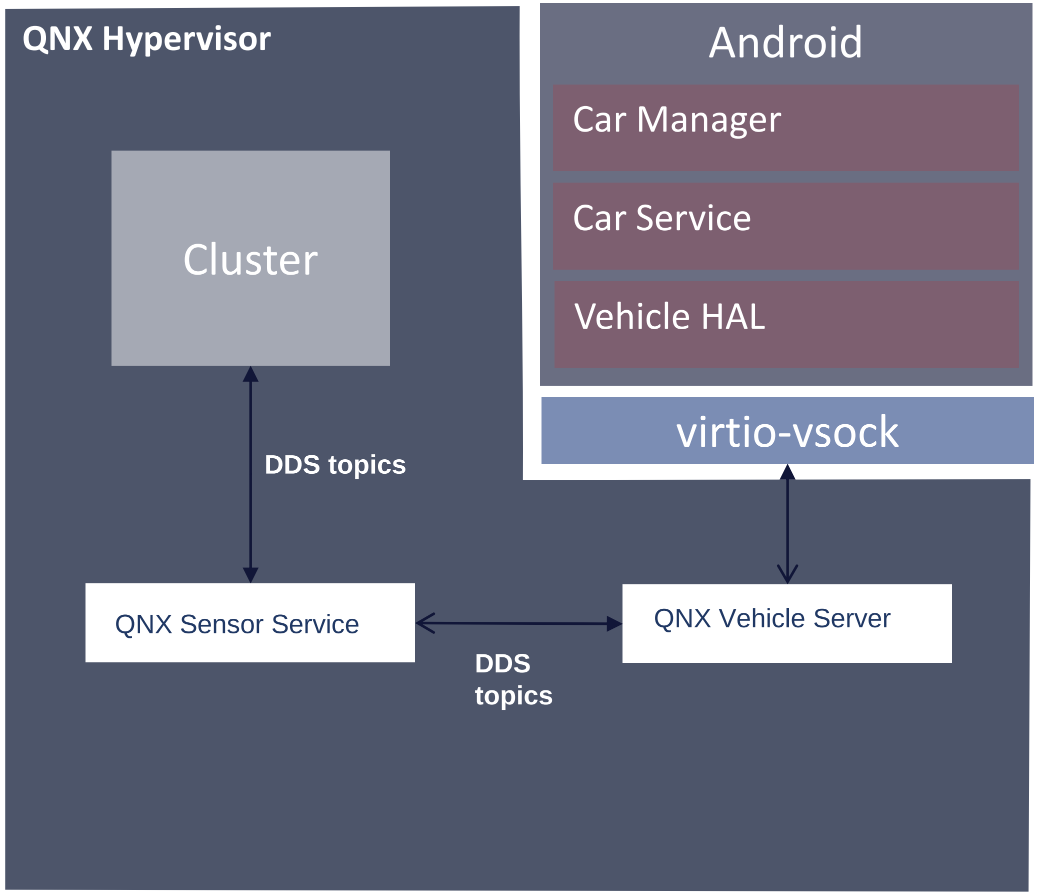
GFXBench: OFF Screen



Native QNX Graphics (reference)

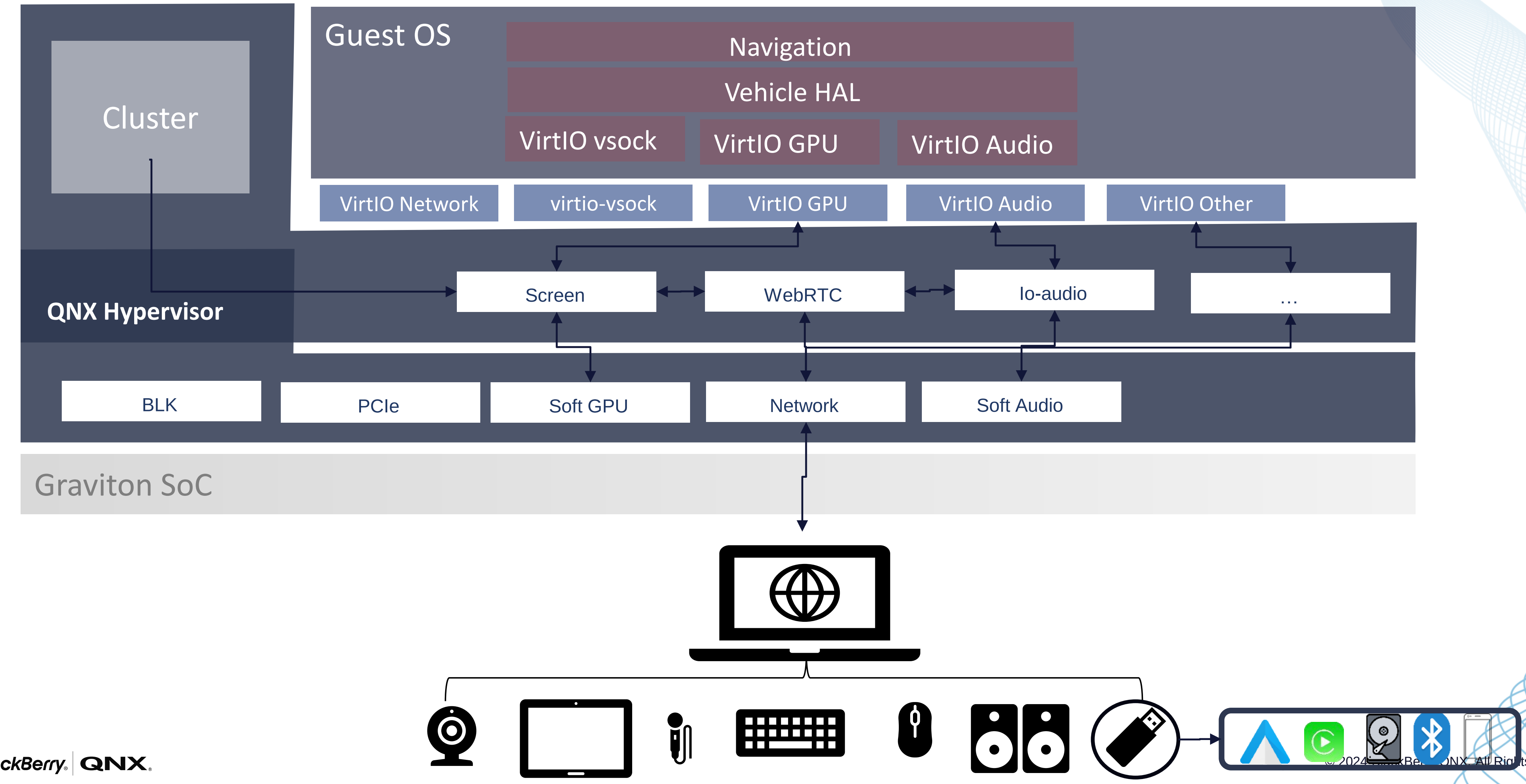
VirtIO-GPU (Android Guest)

QNX Advanced Architecture and Framework - Example

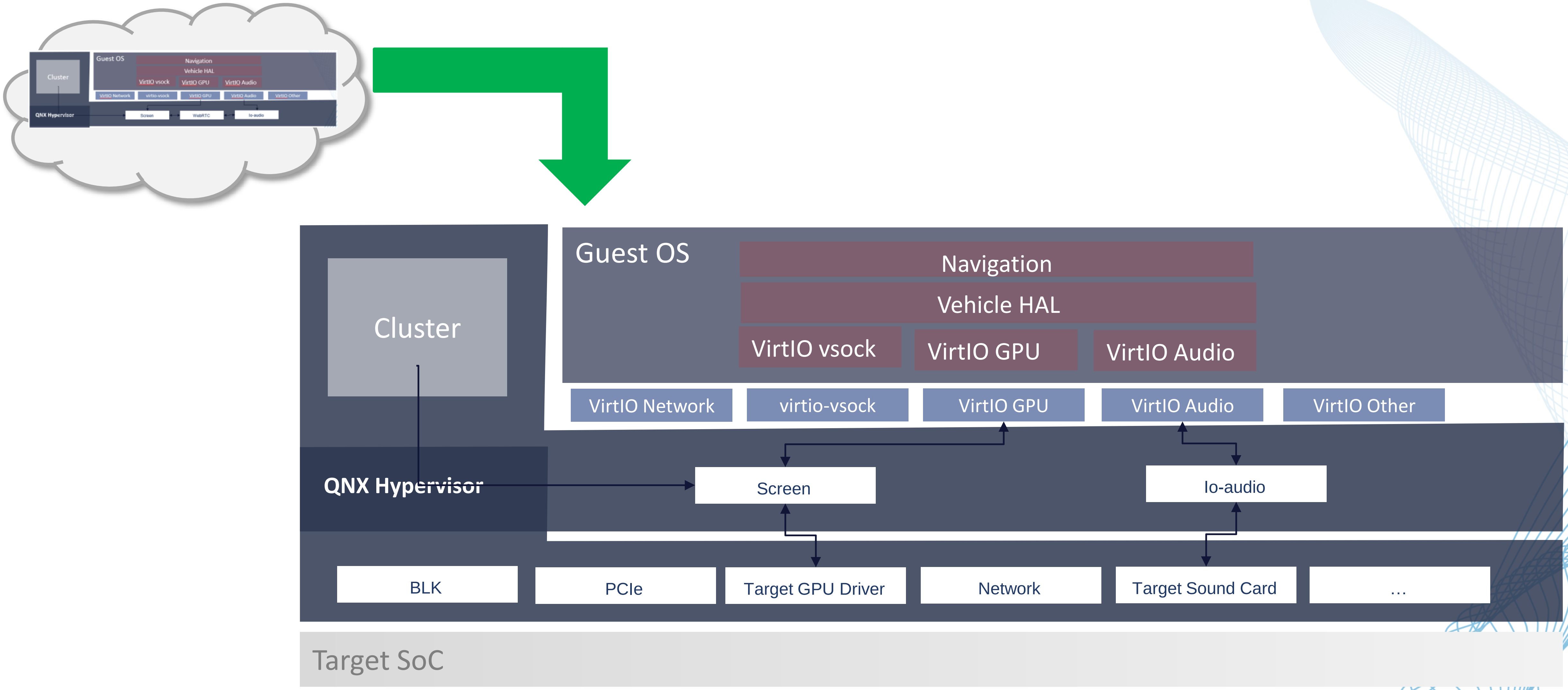


- The sensor-service handles requests to change the value of vehicle properties and writes those changes to the DDS topics.
- The vehicle-server will take the vehicle properties from the DDS topics and convert them to the Android format and send them to the Android Vehicle HAL (and vice-versa)
- The Cluster listen for DDS topic changes and will request property changes (like gear, turn signals etc) after the buttons are selected in the cluster control page.

Digital Cockpit Development in the Cloud



Digital Cockpit from Clout to Target



Guest OS

Navigation

Vehicle HAL

VirtIO vsock

VirtIO GPU

VirtIO Audio

Cluster

VirtIO Network

virtio-vsock

VirtIO GPU

VirtIO Audio

VirtIO Other

QNX Hypervisor

Screen

Io-audio

BLK

PCIe

Target GPU Driver

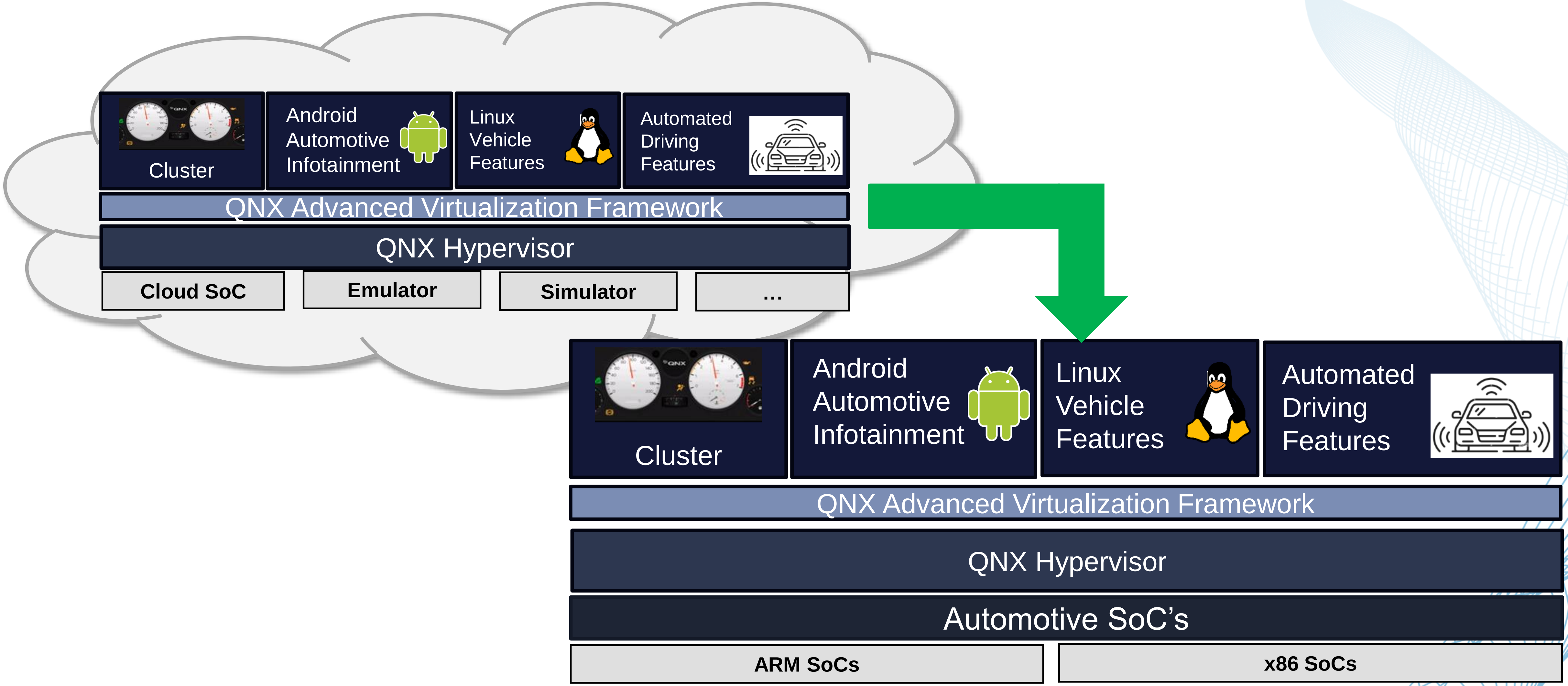
Network

Target Sound Card

…

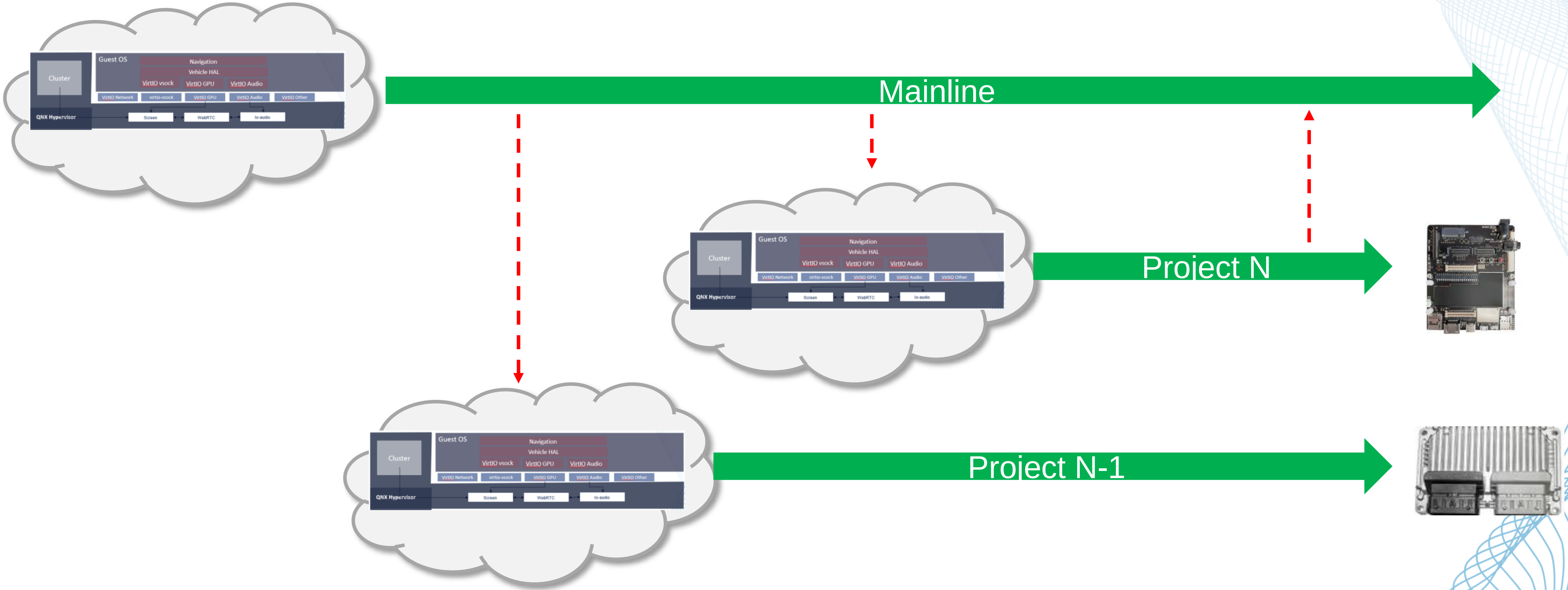
Target SoC

QNX Software as the Solution



Cluster
Android Automotive Infotainment
Linux Vehicle Features
Automated Driving Features
QNX Advanced Virtualization Framework
QNX Hypervisor
Cloud SoC
Emulator
Simulator
…
Cluster
Android Automotive Infotainment
Linux Vehicle Features
Automated Driving Features
QNX Advanced Virtualization Framework
QNX Hypervisor
Automotive SoC’s
ARM SoCs
x86 SoCs

Digital Cockpit Development Efficiency

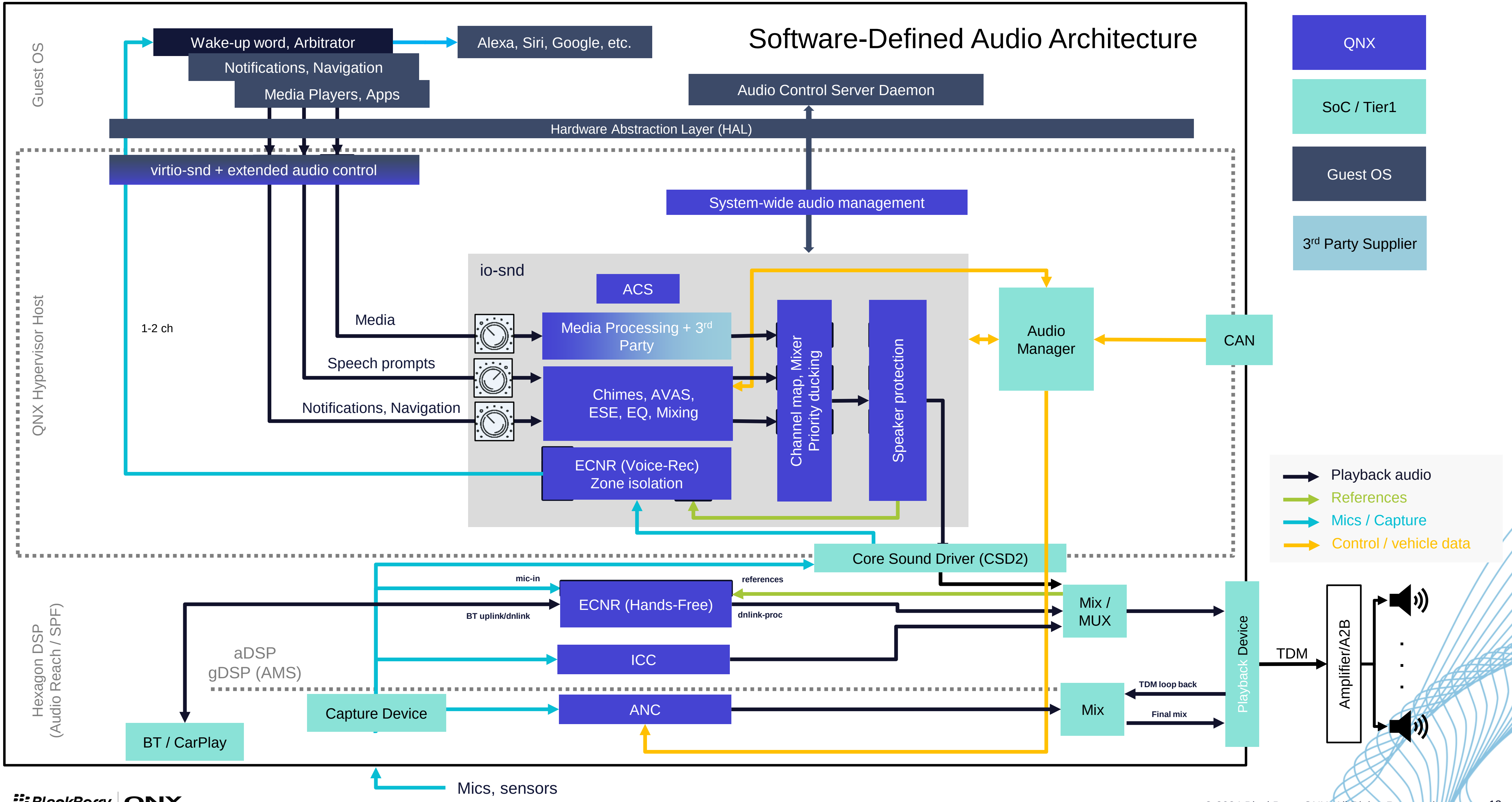


Mainline

Project N

Project N-1

Software-Defined Audio Architecture



Guest OS

Wake-up word, Arbitrator

Notifications, Navigation

Media Players, Apps

Alexa, Siri, Google, etc.

Audio Control Server Daemon

Hardware Abstraction Layer (HAL)

QNX Hypervisor Host

virtio-snd + extended audio control

System-wide audio management

1-2 ch

io-snd

ACS

Media

Speech prompts

Notifications, Navigation

Media Processing + 3rd Party

Chimes, AVAS, ESE, EQ, Mixing

ECNR (Voice-Rec) Zone isolation

Channel map, Mixer Priority ducking

Speaker protection

Audio Manager

CAN

Core Sound Driver (CSD2)

Hexagon DSP (Audio Reach / SPF)

aDSP gDSP (AMS)

mic-in

references

BT uplink/dnlink

dnlink-proc

ECNR (Hands-Free)

ICC

ANC

Capture Device

BT / CarPlay

Mix / MUX

Mix

TDM loop back

Final mix

Playback Device

TDM

Amplifier/A2B

Mics, sensors

QNX

SoC / Tier1

Guest OS

3rd Party Supplier

→ Playback audio

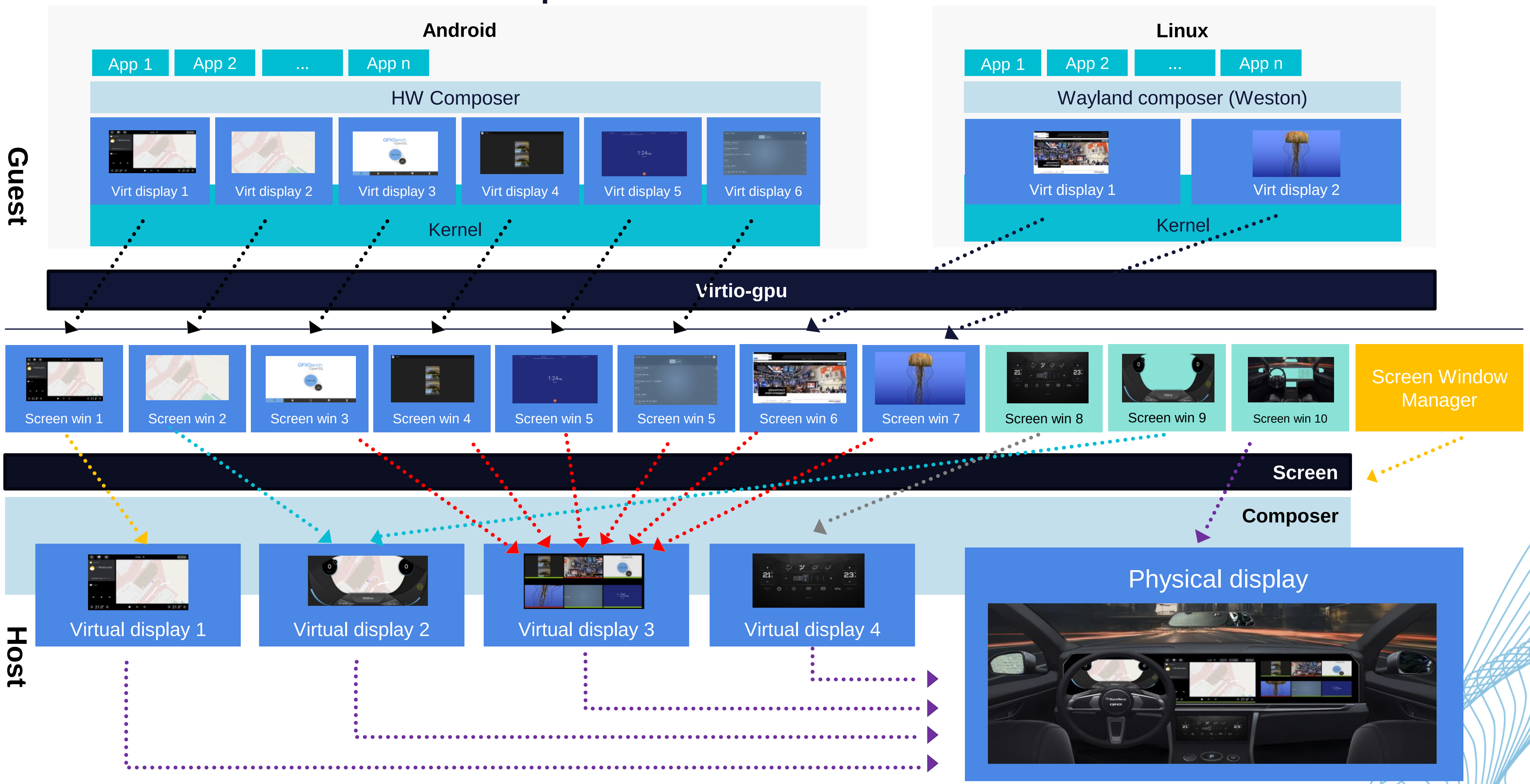
→ References

→ Mics / Capture

→ Control / vehicle data

BlackBerry | QNX

Software Defined Composition



Addressing Digital Cockpit Technology Challenges

Constant software churn

Vulkan OpenCL Unity

Managing portability among automotive SoCs

SoC

Qualcomm MEDIATEK NVIDIA SAMSUNG intel AMD

Manage software certification costs

OS Integration Packages

- Semi-annual integration package updates to ensure Android Automotive OS and Yocto Linux compatibility with QNX Software

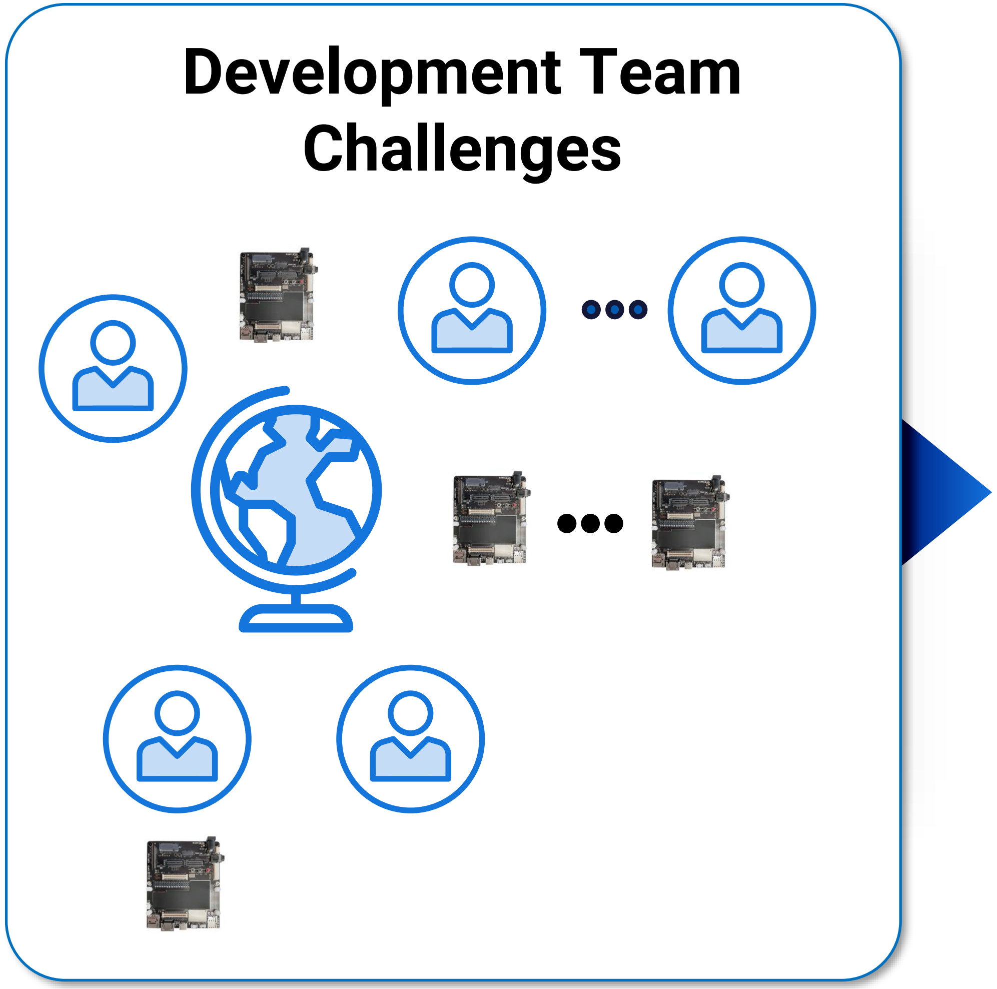
Wide support of automotive and cloud hardware with managed roadmap

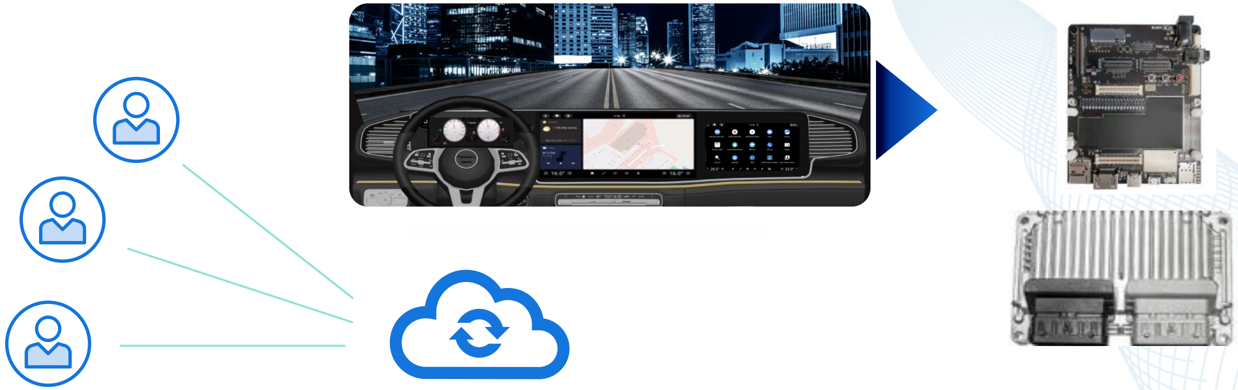
intel Qualcomm NVIDIA AMD SAMSUNG MEDIATEK

Create a mixed criticality system without sacrificing performance

- QNX provides safety certified hypervisor, operating system and other software components

Addressing Development Team Challenges





Cloud First: Rapid Prototyping & Early Testing

✓ Fast cycle time; develop, integrate, test quickly. Derisk development

Quick, easy scaling & developer collaboration

✓ Leverage QNX virtual environments to quickly deploy test resources to team members as needed

Development Efficiency

✓ Dynamically provision QNX test environments to automate testing as part of Continuous Integration / Deployment pipelines

Addressing Market Challenges

"… taking what used to take months to be achieved down to 24 hours in some cases"

"… enabling the delivery of infotainment tech to customers 100 times faster than previous processes"

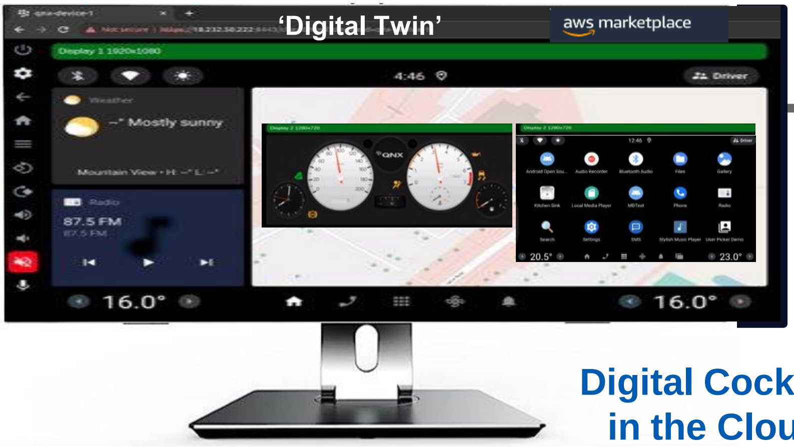
"… With our virtual cockpit, we're revolutionizing not just our approach, but also that of our suppliers and partners in the industry"

Users of the Product

Building on our multi-year partnership, Stellantis is using Amazon Web Services (AWS) computing power and BlackBerry technology for its virtual cockpit that can create and test new versions of car controls and systems in as little as 24 hours instead of months. Get ready for a more connected ride ahead!

– Adam Selipsky, Former AWS CEO (LinkedIn 2024)

Digital Cockpit ECU



Digital Cockpit in the Cloud

Thank you