Enabling a Software-Defined Automotive Edge with VirtIO Based Device Virtualization

Dec 2023

AGL Software-defined Vehicle Expert Group Leader

Jerry Zhao, Panasonic Automotive Systems Co., Ltd.

Agenda

- Why: Industry Trends with Software-Defined Vehicles
- What: Architectural Changes in the Automotive World
- How: Decoupling Software from Hardware with Device Virtualization
- To where: Moving Forwards - Constructing a bright and open future of SDV with AGL

Industry Trends with Software-Defined Vehicles (SDV)

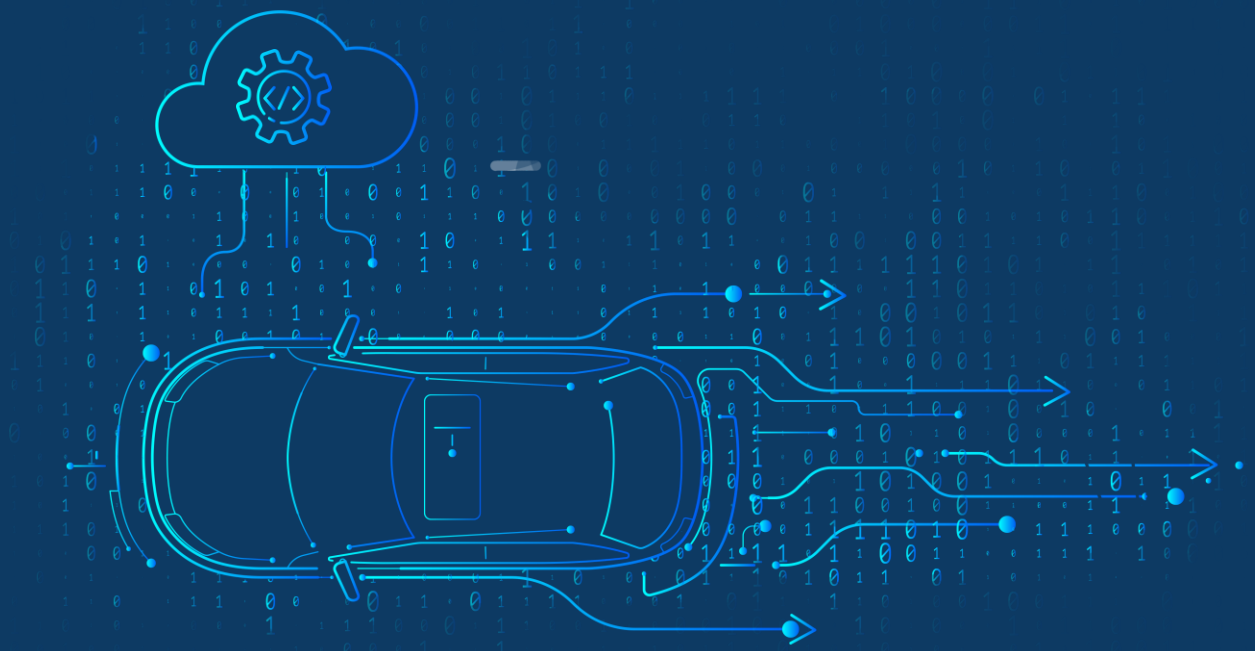
Industry Trend with SDV

Software-Defined Vehicles

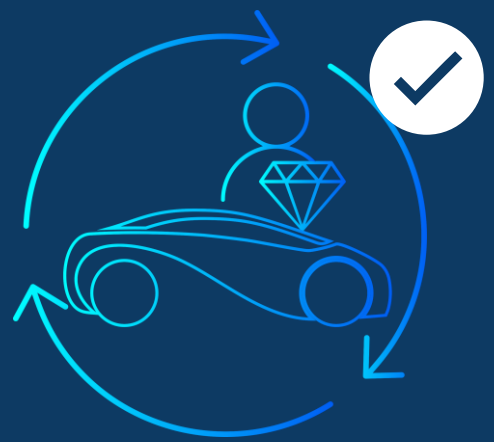
Drastic Changes
in EE Architecture

Open Source,
De Facto Standard

Connected
& AI-powered

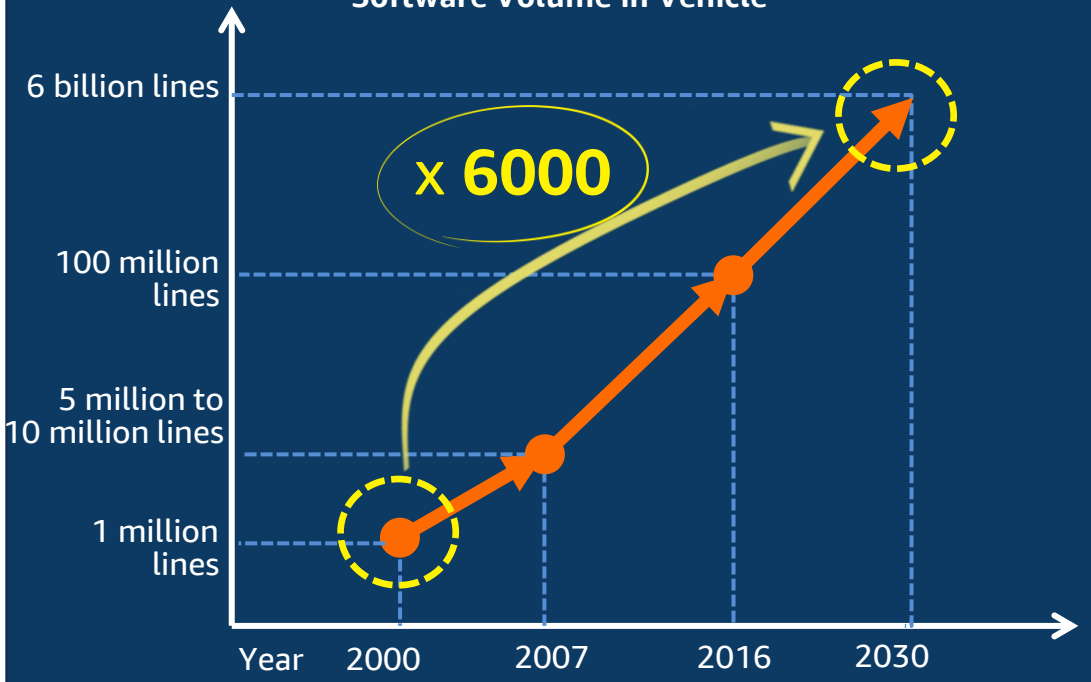


Speedily Delivered Values



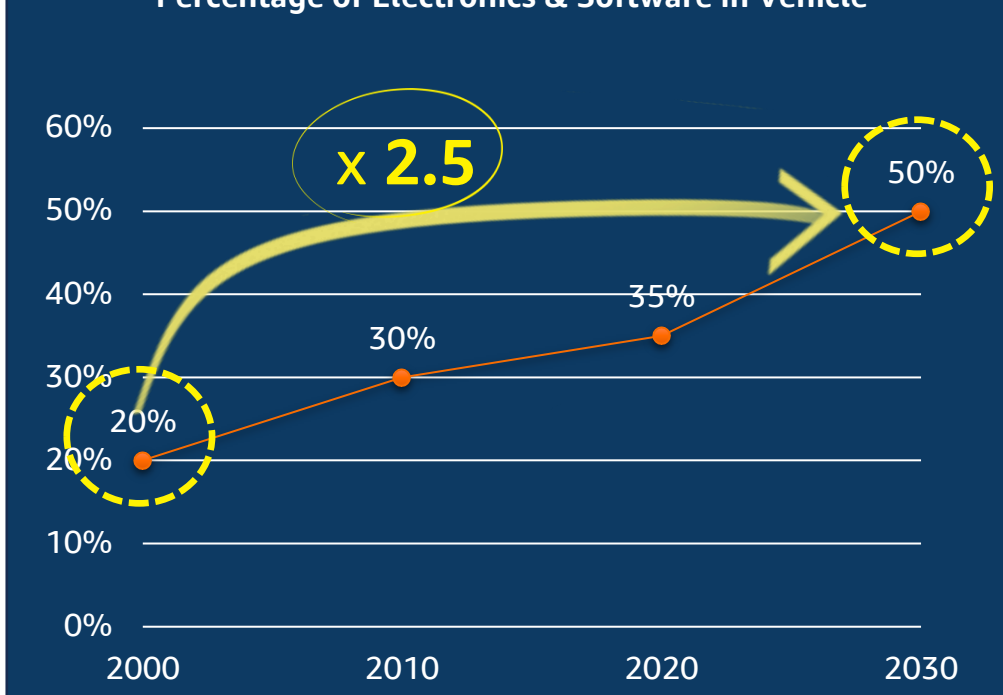
Shift to SDV Industry Trend

Increased Complexity of Vehicle Software (*1)



*1: Source: Ministry of Economy, NXP Semiconductors, Quora, Ignition in Action, NYC AVITAION, Trade and Industry "Toward acceleration of productivity improvement by IT" Mitsubishi UFJ Morgan Stanley Securities' materials, etc.

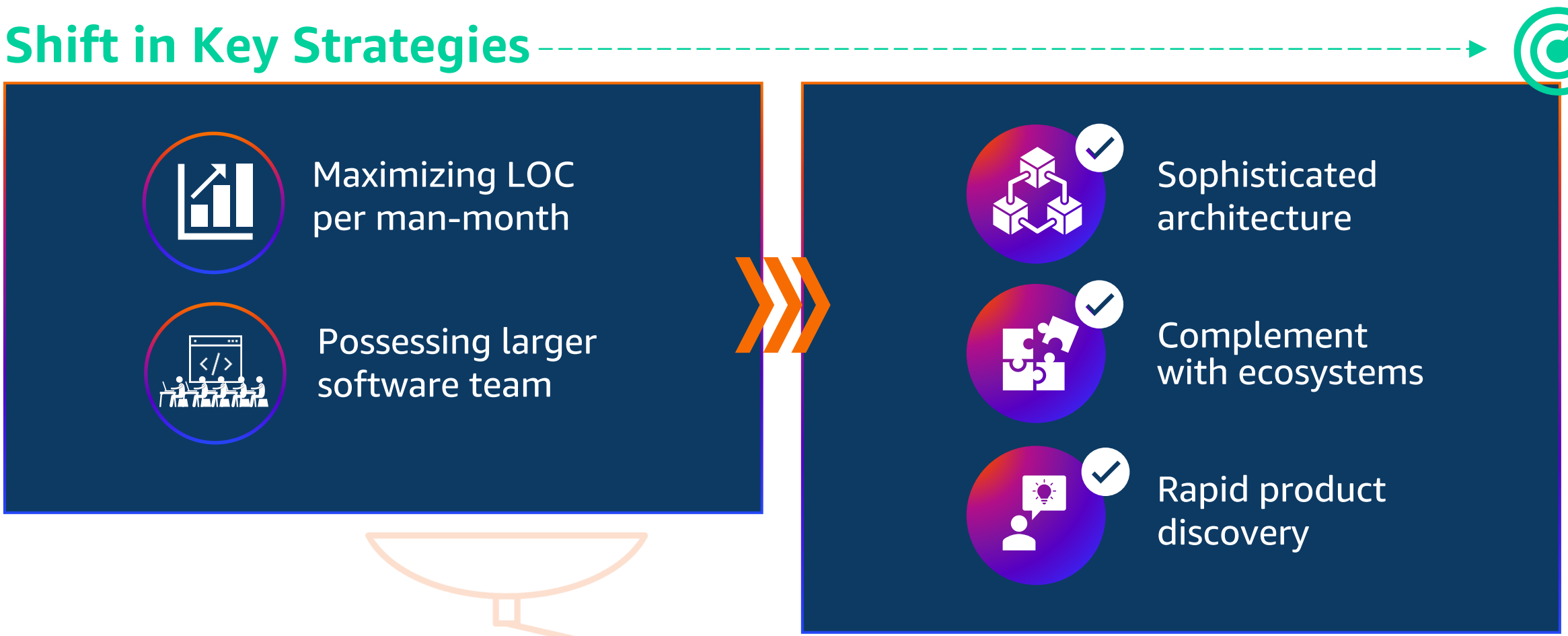
Increased Cost Contribution of Vehicle Software (*2)



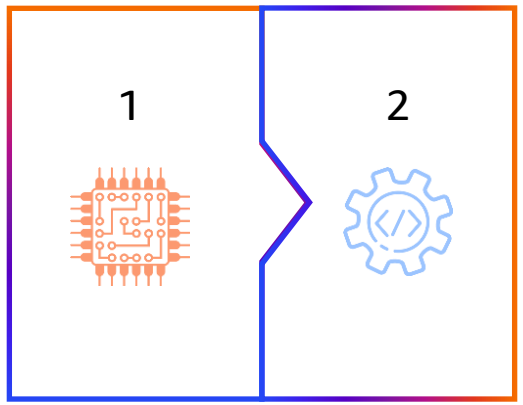
*2: Source: Lux Research

Automotive Industry Game Changer

Shift in Key Strategies



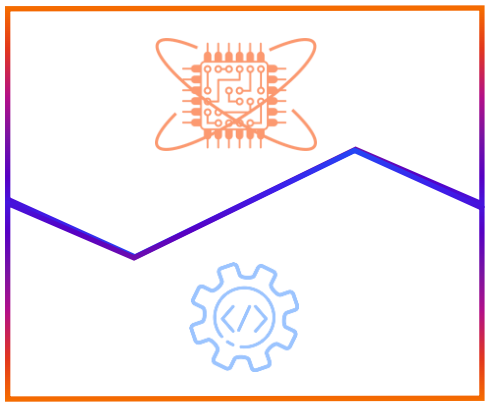
From Hardware First To Software First



Traditional

Manufacture HW prototype and develop SW

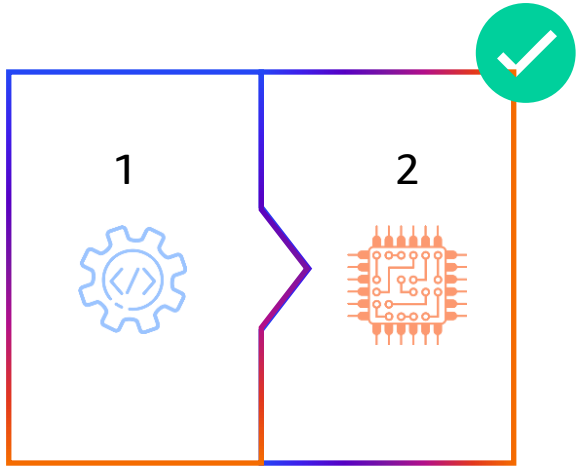
- Long wait time for limited HW
- High sample cost



HW Emulation

Emulate HW and develop SW simultaneously

- Limited to low-level SW & HW
- Costly & time-consuming



Cloud-Native

Develop SW on Cloud and select optimal HW

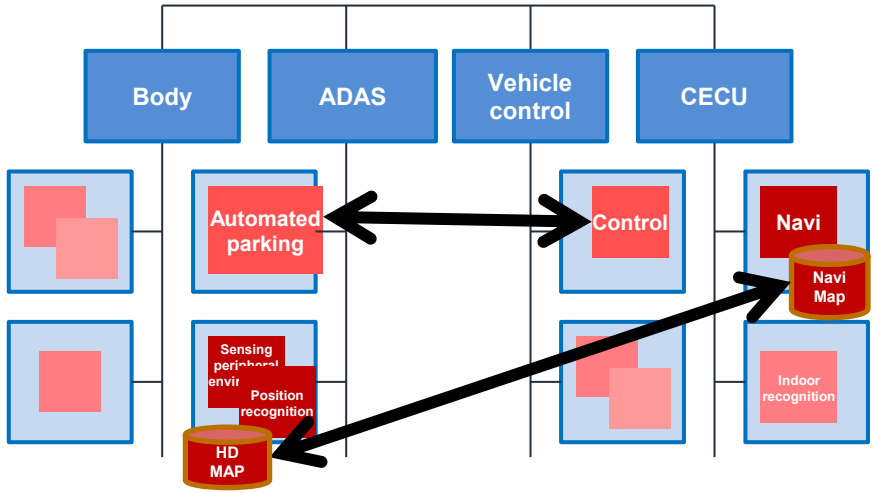
- Rapid function update
- Scalable for large-scale development

Architectural Changes in the Automotive World

Desirable Direction of Automotive System Architecture

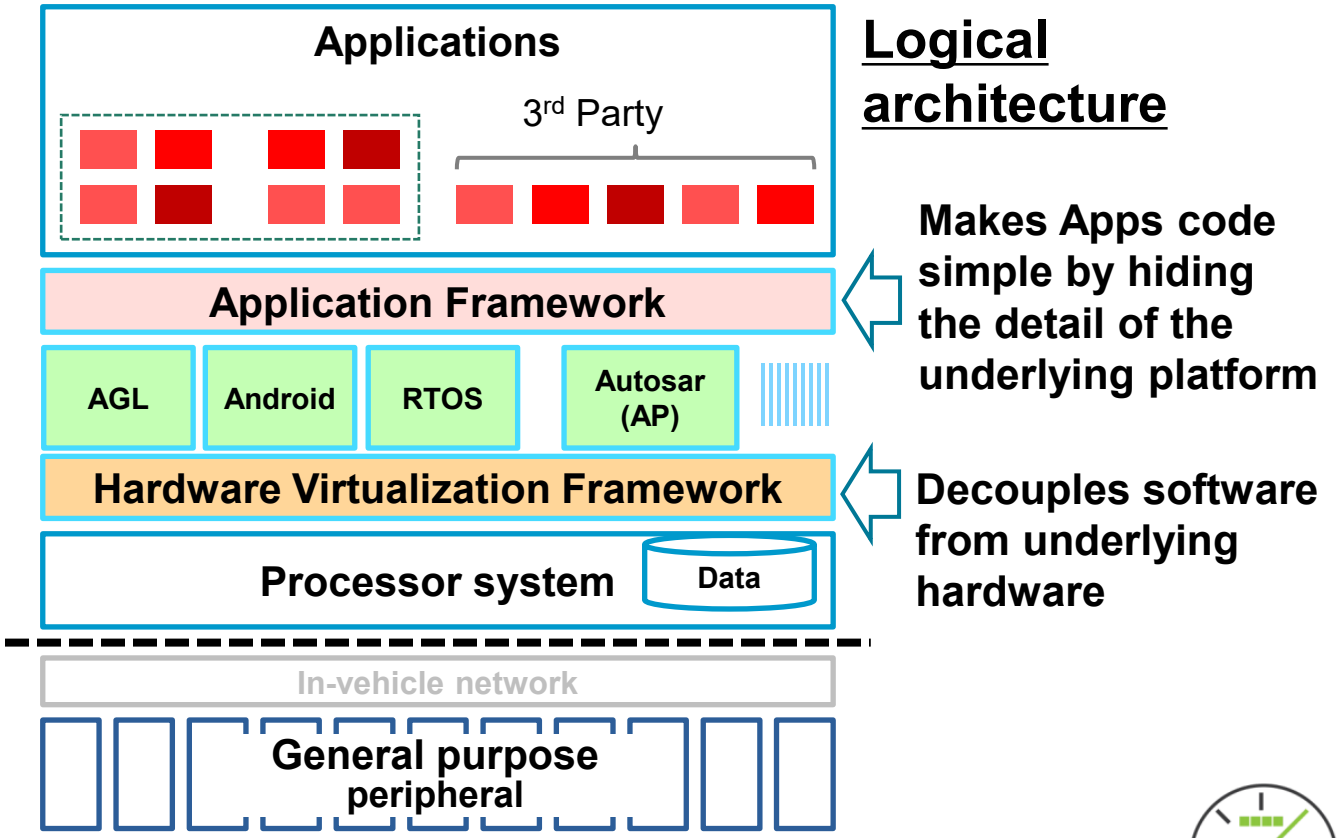
ECU consolidation is not a purpose but means --- The true purpose is to establish the optimal architecture for evolution of software.

"Those who can advance their software more rapidly will gain crucial competitive advantage."



Advancement of technology and updates are difficult.
Overlap of computing resources is an issue also.

Domain Controllers
ECUs
Applications and Data



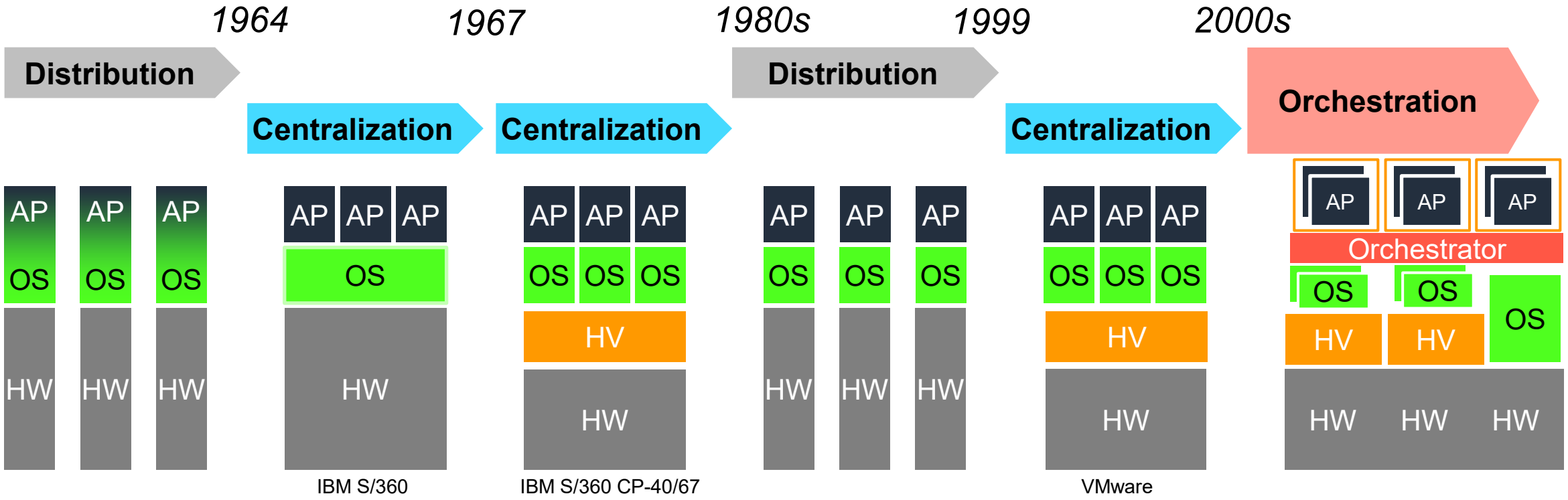
Logical architecture

Makes Apps code simple by hiding the detail of the underlying platform

Decouples software from underlying hardware

Historical Trend of General Computing Architecture (Distribution and Centralization)

The history of general computing architecture is **repeating the cycle between centralization and distribution**, and the automotive industry is following a similar path.

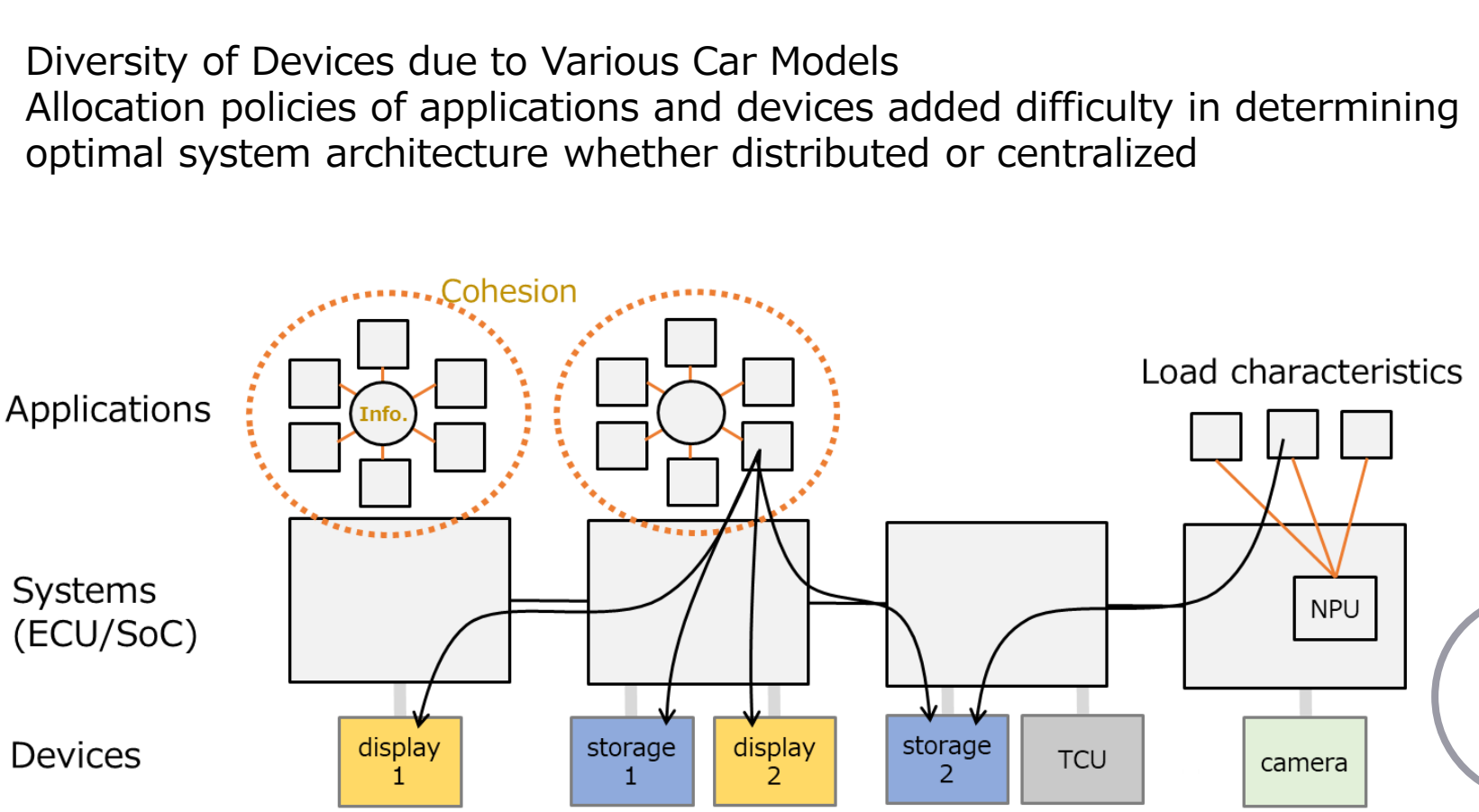


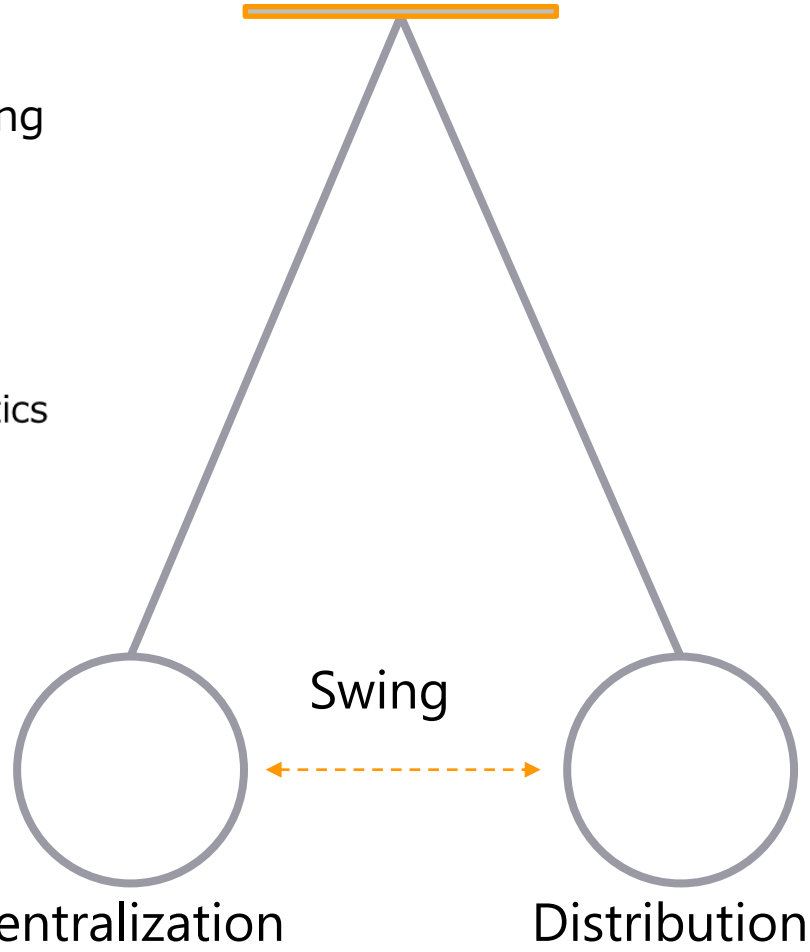
Created by Panasonic Automotive Systems referring to ITmedia IT solution cram school [Graphic explanation] History of virtualization on a single sheet
https://blogs.itmedia.co.jp/itsolutionjuku/2015/06/post_90.html

Greater Complexity in Automotive to Determine Optimal Architecture

Complicated natures of both devices and applications make a greater complexity for automotive

- Diversity of Devices due to Various Car Models
- Allocation policies of applications and devices added difficulty in determining optimal system architecture whether distributed or centralized

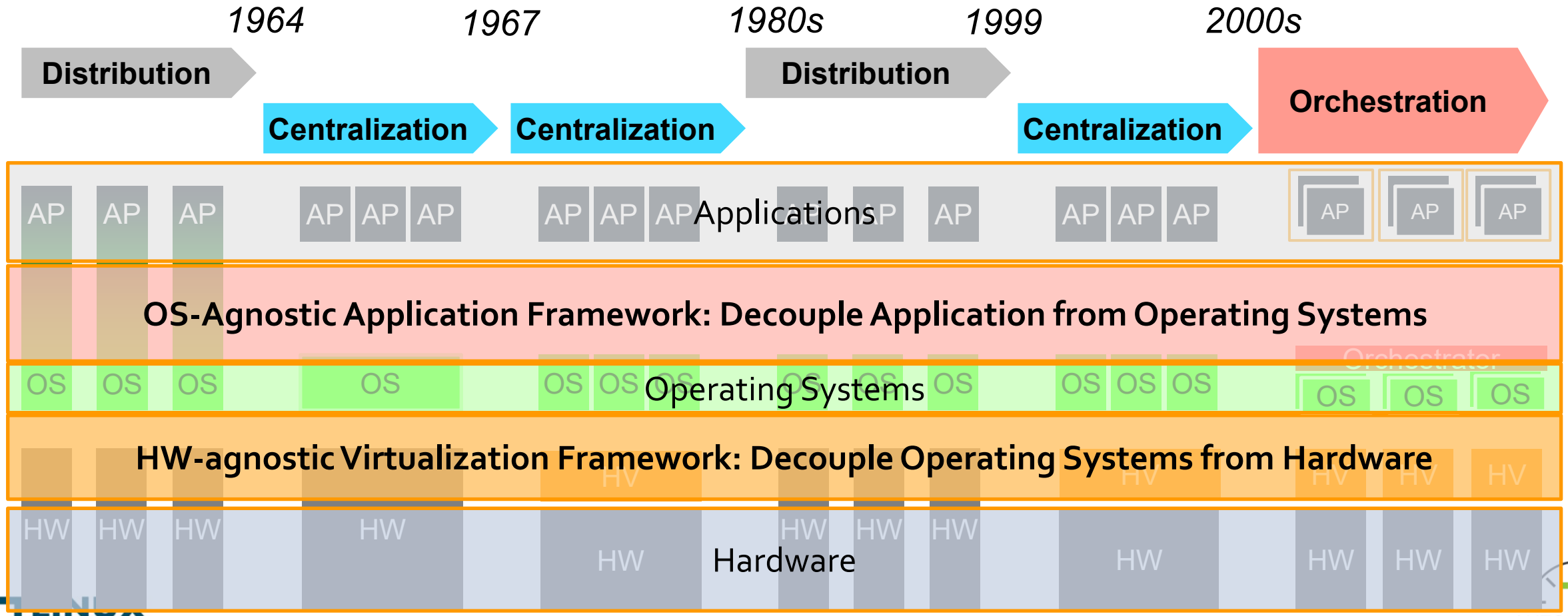




Historical Trend of General Computing Architecture (Distribution and Centralization)

No matter how the underlaying computing architecture has changed, a consistent objective is to decouple apps (directly contributed to user values) from underlying computing architecture

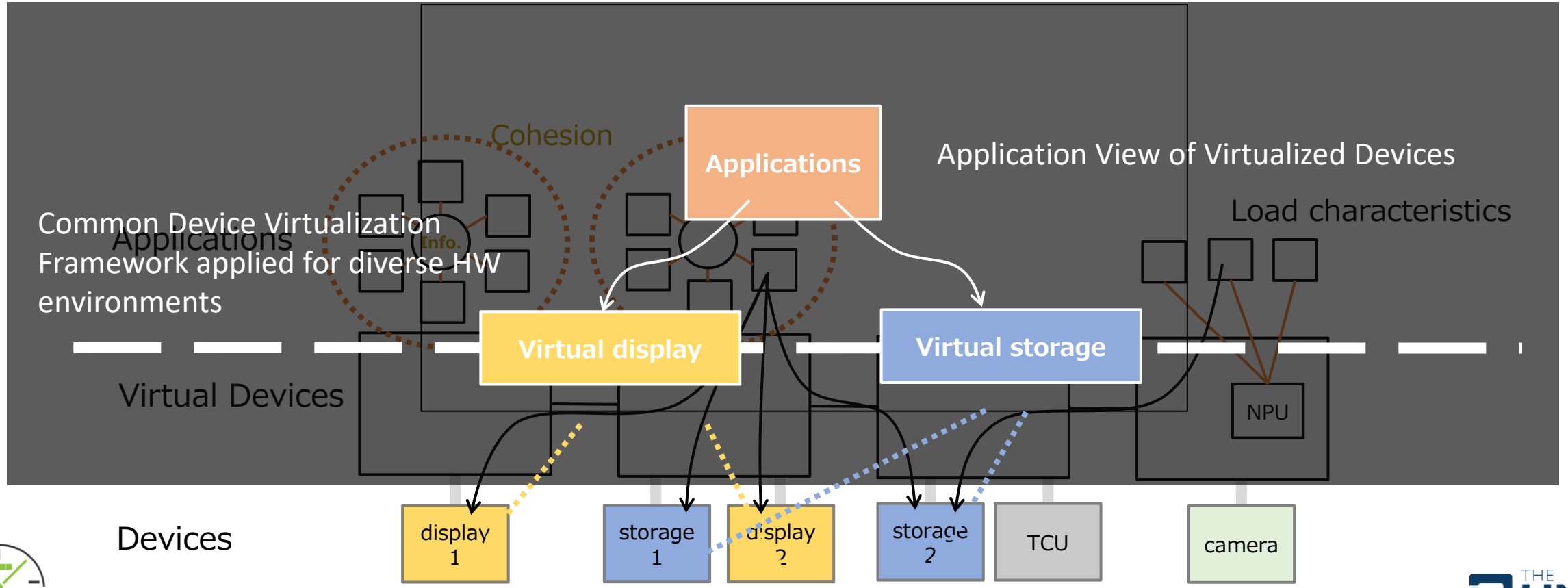
→ An Operating-System-Agnostic Application Framework and a Hardware-Agnostic Abstraction Framework are continuously to be the key to drive industry shift from hardware-centric to software-defined



Decoupling Software from Hardware with Device Virtualization

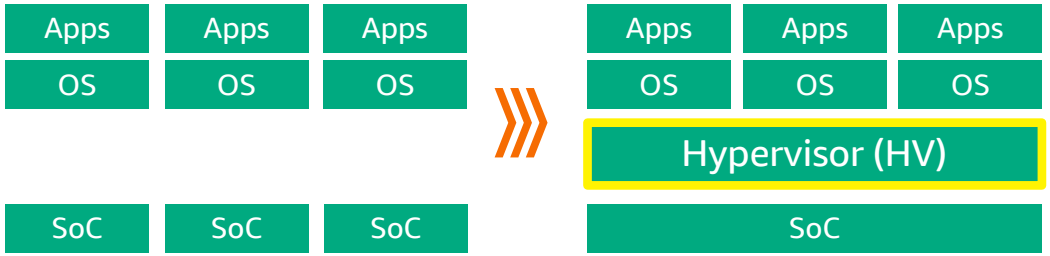
Device Virtualization: Key to Software Defined Vehicles

Software Defined Vehicle needs a common device virtualization framework to decouple software implementation from diverse hardware targets across vehicle variants/generations, architectures (single/multiple-ECU) and development environments (real/virtual ECU)



Decouple Hardware And Software

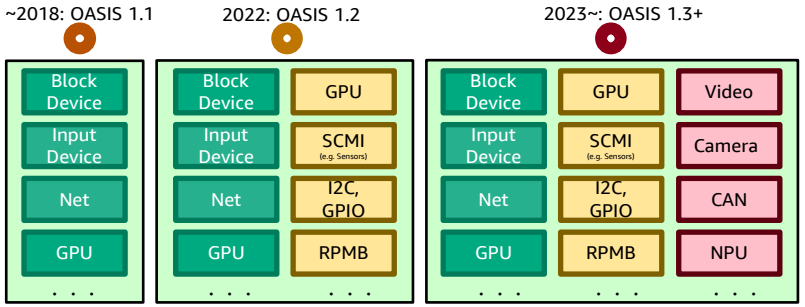
Consolidation requires Hypervisor



Hypervisor needs Standard Interfaces

Standard Specification

OASIS OPEN



Standard Implementation

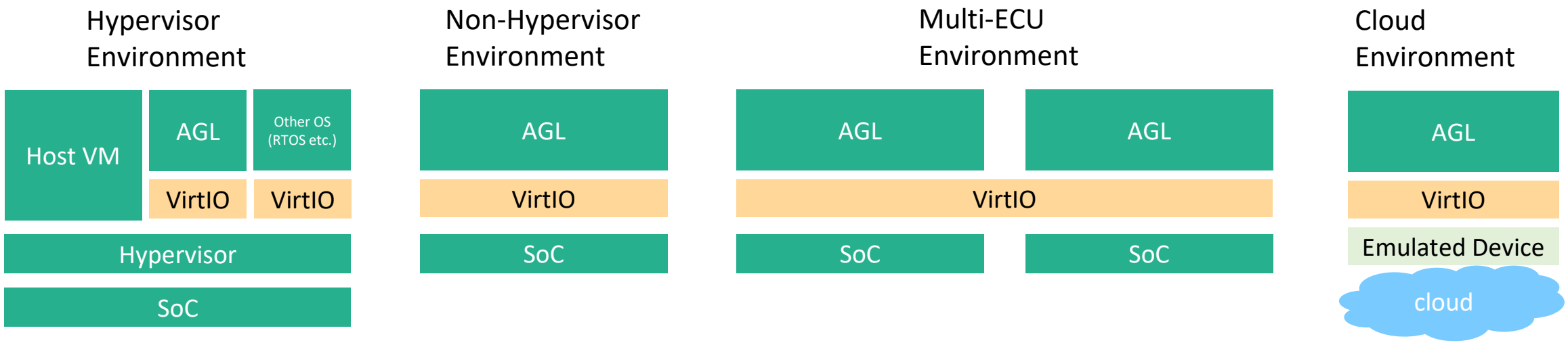
AUTOMOTIVE GRADE LINUX

AUTOMOTIVE GRADE LINUX

THE LINUX FOUNDATION

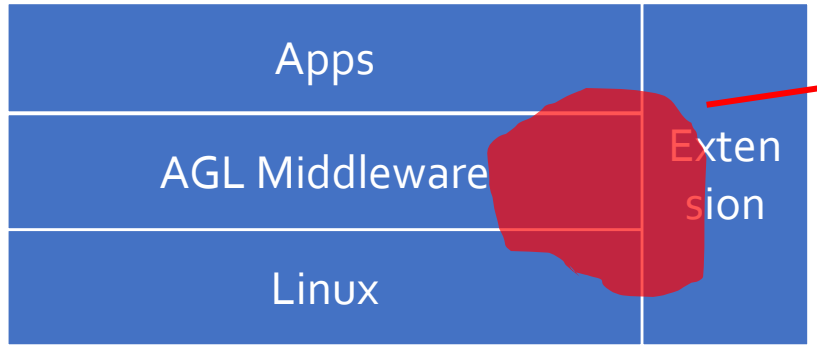
Overview of Device Virtualization in AGL - Concept

Device Virtualization with VirtIO benefits in establishing
a complete and healthy ecosystem for AGL to enhance interchangeability and interoperability in various scenarios.



Pains around Virtualized AGL in the Past

Fragmentation
Serious Barrier
For "Virtualization
Ready BSP"



Dependency
on specific virtualization solution

Must Adapt to Every Single Incompatible Interface!

Proprietary Para Virtual Devices XA

Proprietary Para Virtual Devices YA,YB…

Proprietary Para Virtual Devices ZC,ZD

Hypervisor X

Hypervisor Y

Hypervisor Z

Limited Freedom to Choose Hypervisor and SoC Combination!

Proprietary BE Device Drivers XA, YA

SoC A

Proprietary BE Device Drivers YB

SoC B

Proprietary BE Device Drivers YC, ZC

SoC C

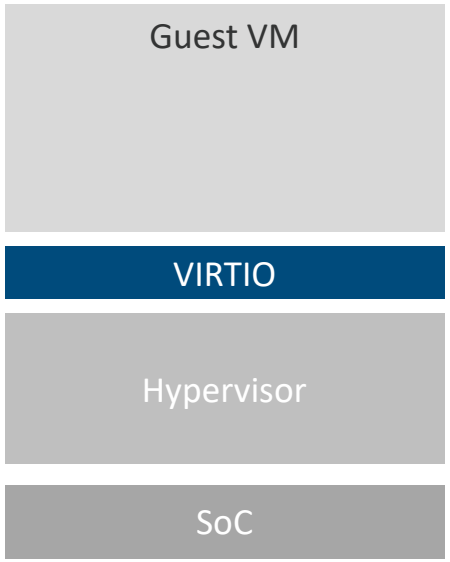
Proprietary BE Device Drivers ZD

SoC D

Excerpt from Panasonic's Keynote Presentation at the AGL AMM July 2020

Enter Standard Virtualization Framework - VirtIO

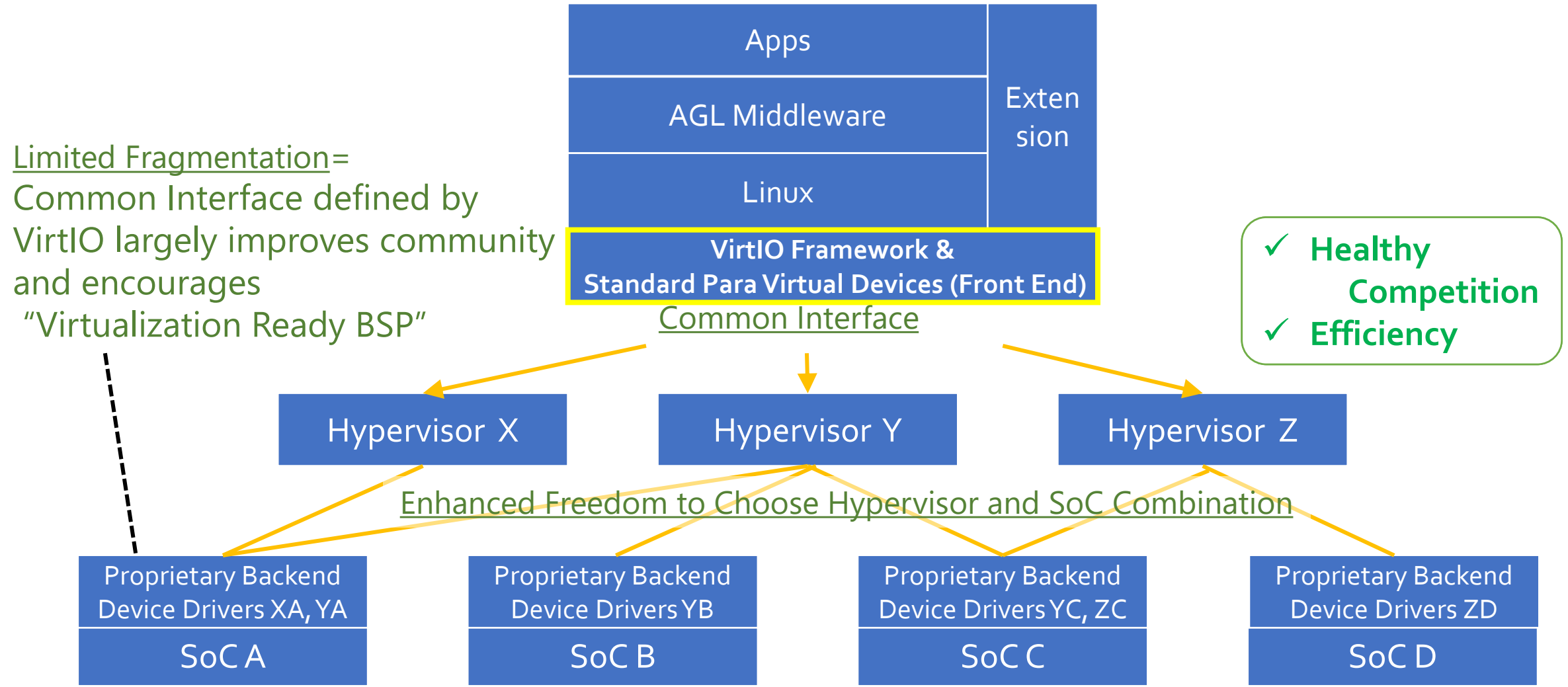
- Developed in 2008 as a hypervisor neutral way of accessing devices
- Provide virtual machines access to Input/Output
- A standardized interface for I/O between virtual machines and hypervisors
- Abstract device functionality instead of hardware
- Drivers are widely available in all major operating systems (Linux, Android, BSD, Windows, etc)
- Supported by all clouds and enterprise hypervisors



- Reliable and proven technology
- Versatile abstraction model
- Scalable and high performance
- Multiple interoperable implementations
- Broad ecosystem across multiple industries

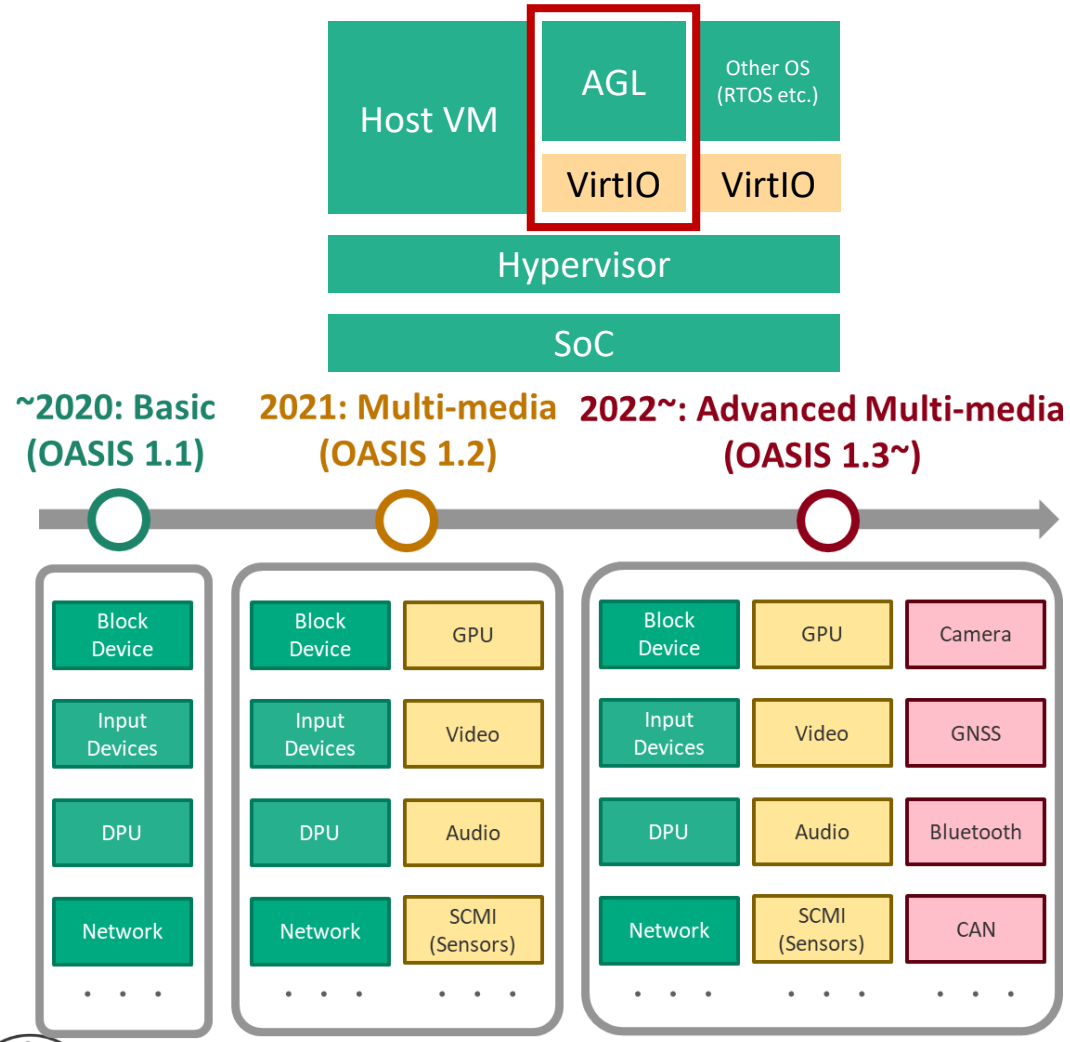
Excerpt from Virt-EG Member Company OpenSynergy's Presentation at AGL F2F Oct 2022

VirtIO as a Common Framework for Virtualized AGL



Excerpt from Panasonic's Keynote Presentation at the AGL AMM July 2020

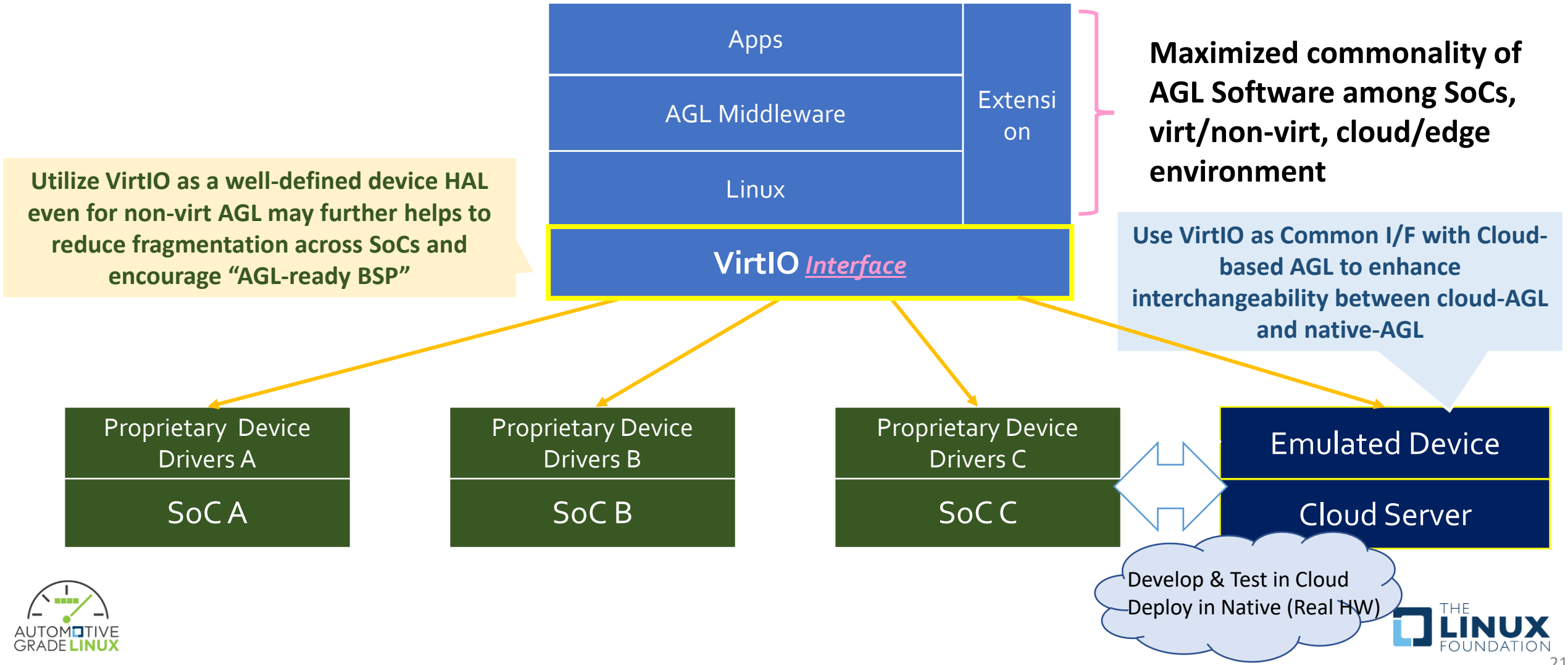
VirtIO Work for Hypervisor Environment



Virt-EG VirtIO Device Development Focus

- Kooky Koi (2021.2): VirtIO has been officially supported by AGL as common device framework for Virtualized with basic device support.
  - Block device: virtio-blk
  - Console: virtio-console
  - Display and GPU: virtio-gpu
  - Touch Panel: virtio-input
  - Network: virtio-net
  - Random Number Generator: virtio-rng
  - Virtual Socket: virtio-socket
- Lucky Lamprey (2021.7):
  - New feature in existing virtio devices
    - Multi-touch support in virtio-input
  - New virtio devices
    - Sound Card: virtio-sound
- Magic Marlin (2022.2):
  - New virtio devices
    - Video Decoder/Encoder: virtio-video
    - Camera: Camera streaming features based on virtio-video
- Nifty Needlefish (2022.7):
  - New virtio devices
    - SCMI: virtio-scmi (acceleromoter & gyroscope sensors)
    - Bluetooth: virtio-bt
- Optimistic Octopus (2022.2):
  - New virtio devices
    - CAN: virito-can

VirtIO Beyond Virtualized AGL



VirtIO Work for Non-Hypervisor Environment

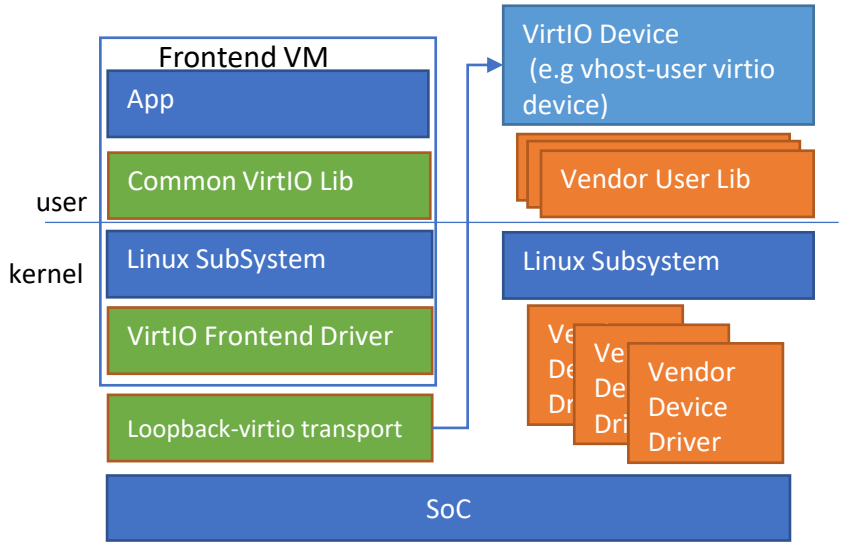
Non-Hypervisor Environment

- Finished Design & Implementation of a common virtio-based HAL layer “virtio-loopback” portable to execute on both native and virtual environments with basic devices (blk, rng, input) support
- Continue next-step work to support more devices this year to enable a complete AGL UCB running on the top of virtio-loopback devices
- Plan to extend the use case from single-ECU to multi-ECU and cloud-native

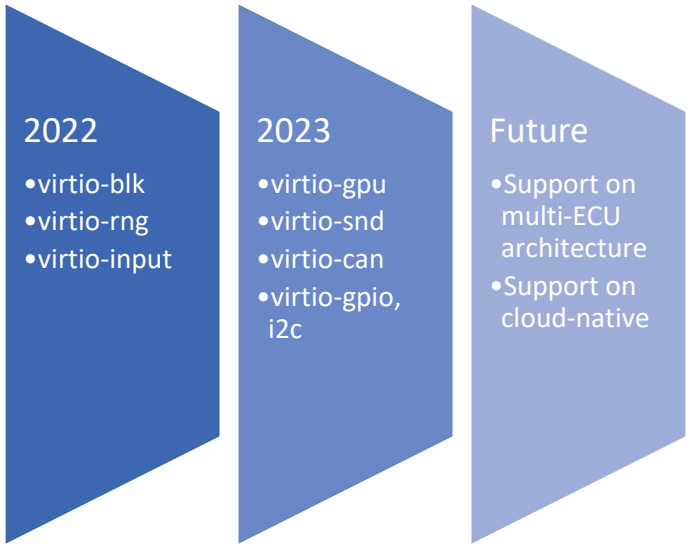
Priority of Device Virtualization Voted by each AGL EG (2021)

Device	Total Score	Priority
Input Device	29	1
Display	27	2
GPU	26	3
CAN bus	20	4
Block Device	19	5
Audio	18	6
Ethernet	11	7
Bluetooth	9	8
SPI	8	9
Serial console	8	9
SCMI	8	9

High Level Architecture Design



Status and Future Plan

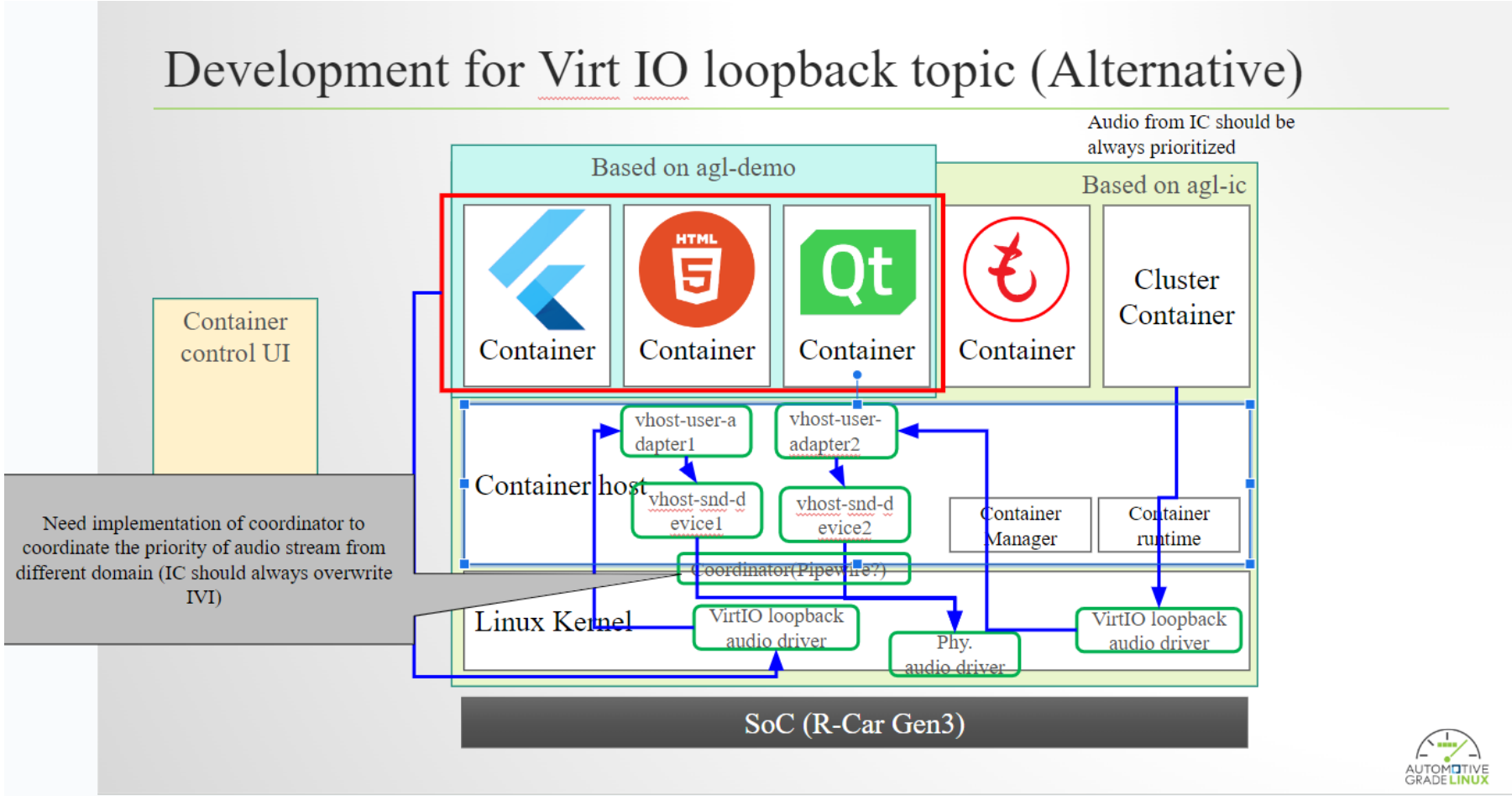


AUTOMOTIVE GRADE LINUX

THE LINUX FOUNDATION

Cross-EG Collaboration with IC-EG: virtio-snd in container environment

- https://docs.google.com/presentation/d/1e6hkOrLEKaQkMq3OXjflCBEDd5mXRyNt/edit#slide=id.g295258e7679_5_1



VirtIO Work for Non-Hypervisor Environment

For technical insights, check the following session today titled with
“**Virtio-loopback: The AGL Hardware Abstraction Layer for Non-Virtualized Environments ”** by Michele
from Virtual Open System on Dec 6 16:55 Reception Room 1

Wednesday, December 6 • 16:05 - 16:45 Back To Schedule

Virtio-loopback: The AGL Hardware Abstraction Layer for Non-Virtualized Environments - Michele Paolino, Virtual Open Systems

Click here to add to My Schedule.

https://sched.co/1TvYf Tweet Share

virtio-loopback is a virtio Hardware Abstraction Layer (HAL) for non-virtualized environments designed and implemented by the Software Defined Vehicle Expert Group (SDV). Thanks to this technology, it is possible to abstract hardware dependencies and run the same software seamlessly on both virtualized and non virtualized systems. During this session, the latest development results related to GPU, sound, CAN, console and GPIO devices will be shared with the community,.

Speakers

Michele Paolino
architect, Virtual Open Systems
virtualization architect active in AGL since 2016

Wednesday December 6, 2023 16:05 - 16:45 JST
Reception Room 1
Automotive Linux Summit (ALS), Software Defined Vehicles
Audience Level Beginner

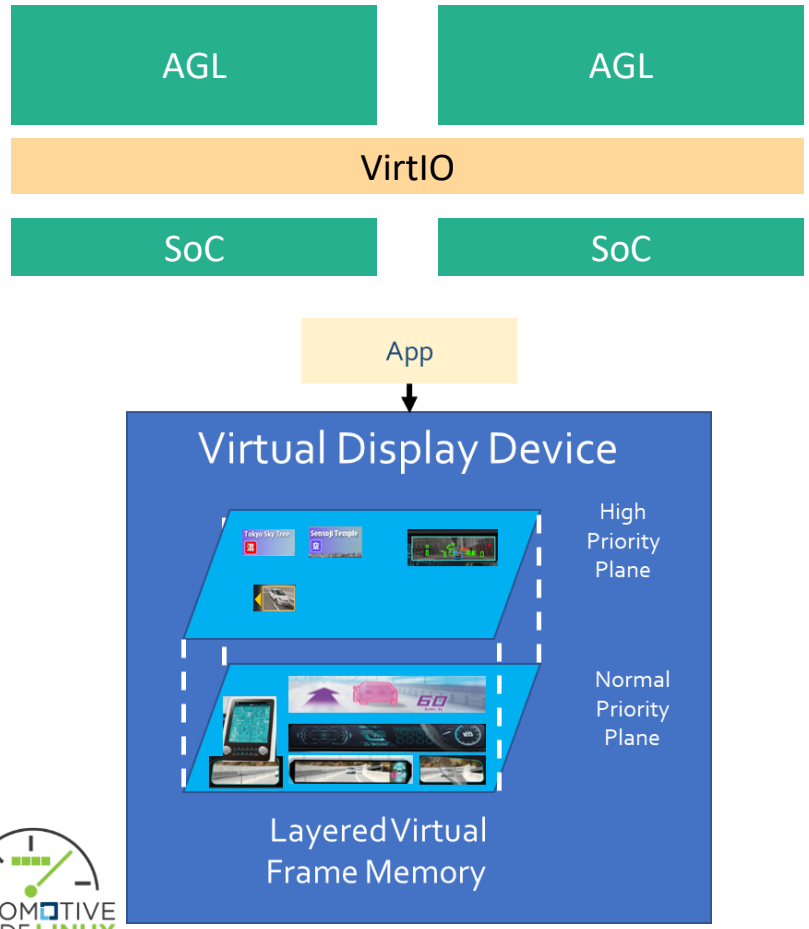
AUTOMOTIVE GRADE LINUX

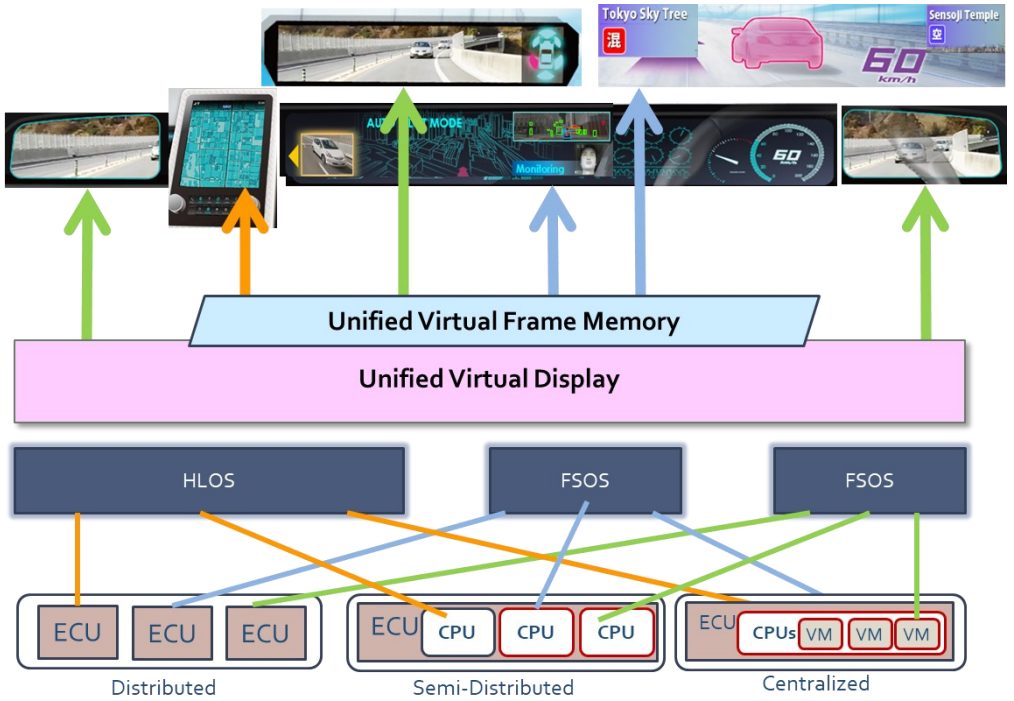
THE LINUX FOUNDATION

VirtIO Work for Multi-ECU

A Unified Virtual Display based on VirtIO-GPU (“Unified HMI” technology) can be established to have Integrated control of multiple display on distributed SoC systems

Multi-ECU Environment



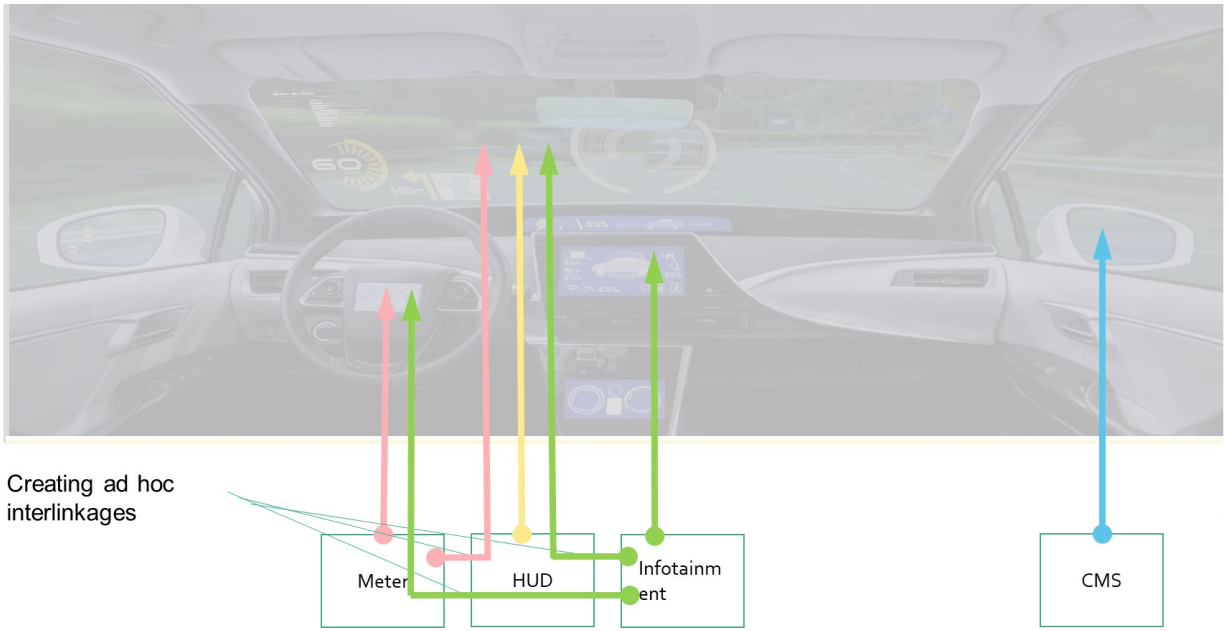


- **Mappig multiple physical displays** of cockpit & cabin into a **single large virtual display**
- **Rendering each application** to its **arbitrary region**

VirtIO Work for Multi-ECU

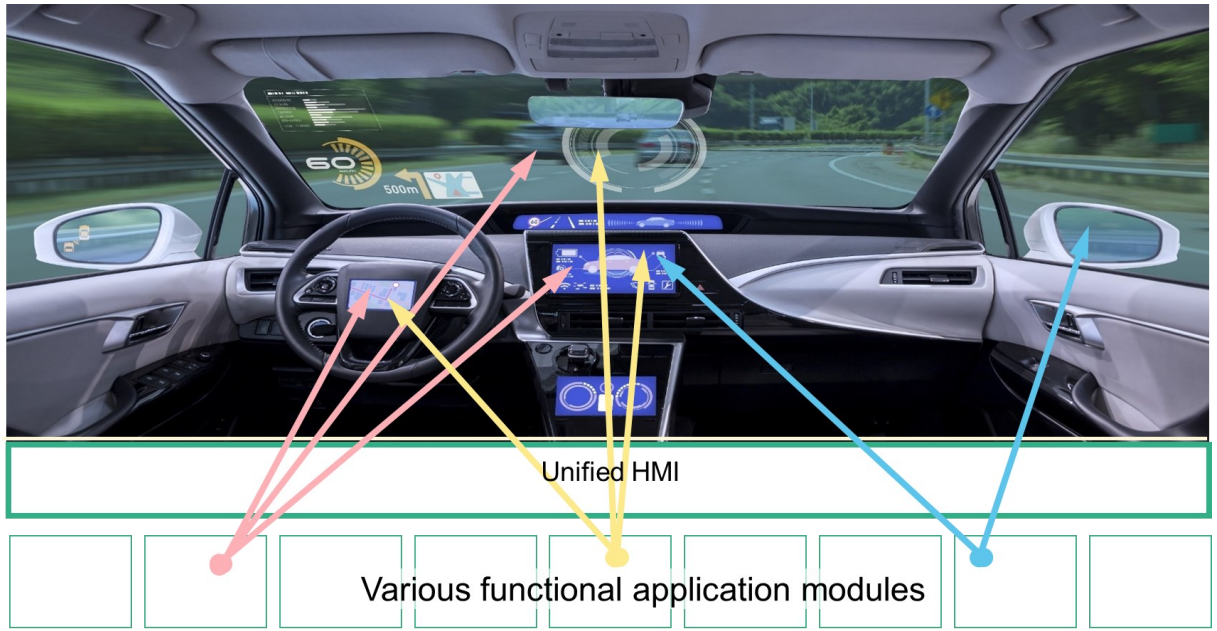
Legacy HMI System

Strict Restriction on ECU & Function-Display Relationship causing harmful Impediment for Cockpit UX



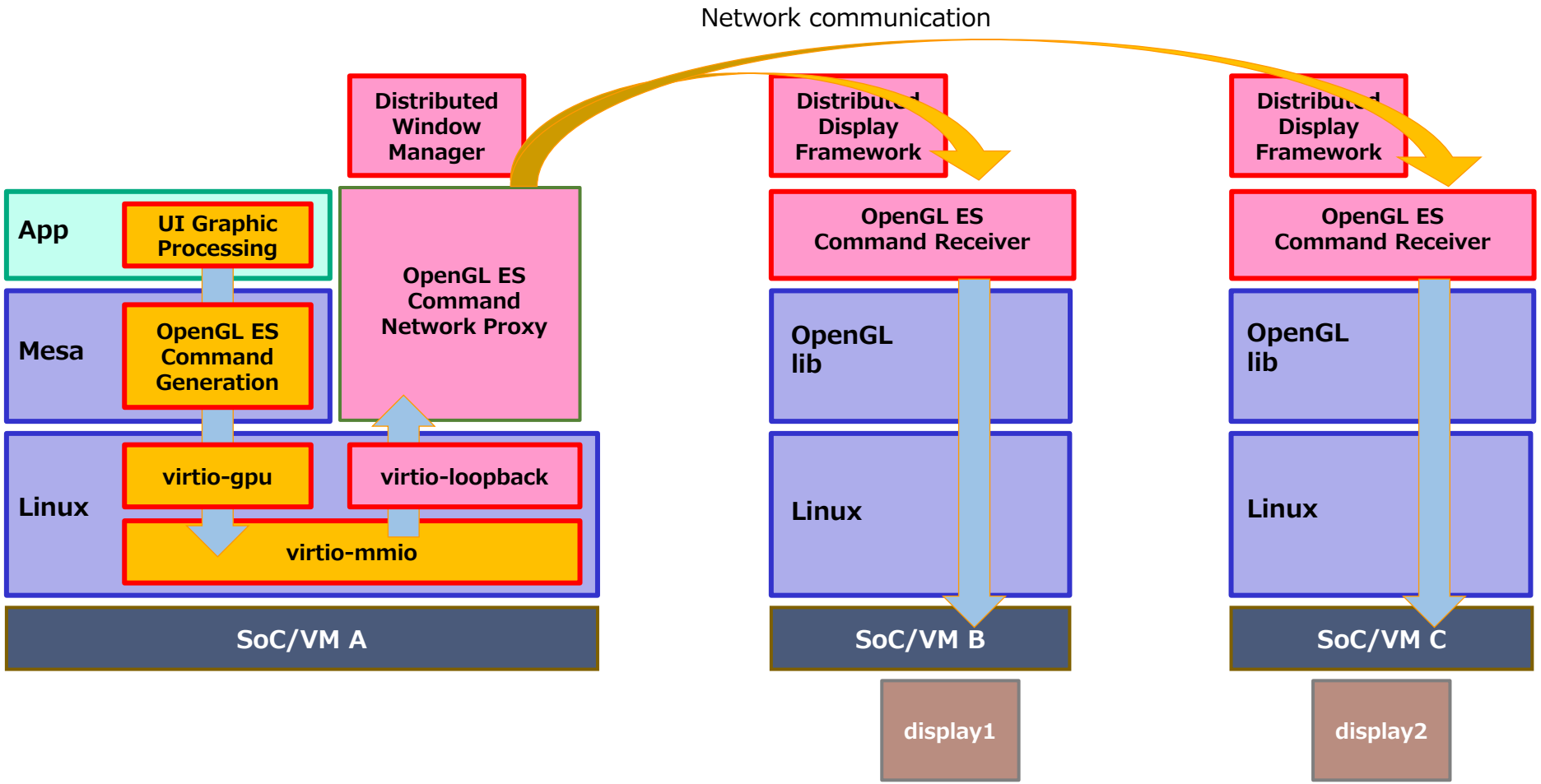
Unified HMI System

Full Flexibility on ECU & Function-Display Relationship for Cockpit UX Innovation



Unified HMI Schematic Architecture

① Apps are rendered with virtual GPU (VirtIO-GPU)
② Graphics are drawn by remote system through proxy requests
③ Layouts are managed by the distributed window manager



Unified HMI Insight

For technical insights, check the following session today titled with **“Achieving a Software-Defined Multi-Display System with Unified HMI”** by Deguchi-san from Panasonic Automotive Systems Co., Ltd. on Dec 6 14:15 Reception Room 1

Wednesday, December 6 • 14:15 - 14:55

Back To Schedule

Achieving a Software-Defined Multi-Display System with Unified HMI - Yukinari Deguchi,

Click here to add to My Schedule.

https://sched.co/1TvbH Tweet Share

In recent years, the increasing number of in-vehicle displays has created a demand for flexible application display across multiple screens, introducing new UI/UX possibilities. However, developing this flexibility using existing graphic frameworks is costly, as display interconnection must be customized for each hardware platform. There are needs in automotive industry for a “Software-Defined” display framework that separates software from hardware. Our Unified HMI system is a display virtualization technology that allows for flexible development of the entire cockpit UI/UX across multiple displays, regardless of hardware configuration. Unified HMI supports software-defined UI/UX development on different SoCs and OS (e.g., AGL and Android), reduces development time and costs by developing seamlessly in cloud virtual ECU, and enables continuous OTA upgrades to meet customer preferences throughout the vehicle's long lifespan.

Wednesday December 6, 2023 14:15 - 14:55 JST

Reception Room 1

Automotive Linux Summit (ALS), AGL Long-Term Support & Maintenance

Audience Level Intermediate

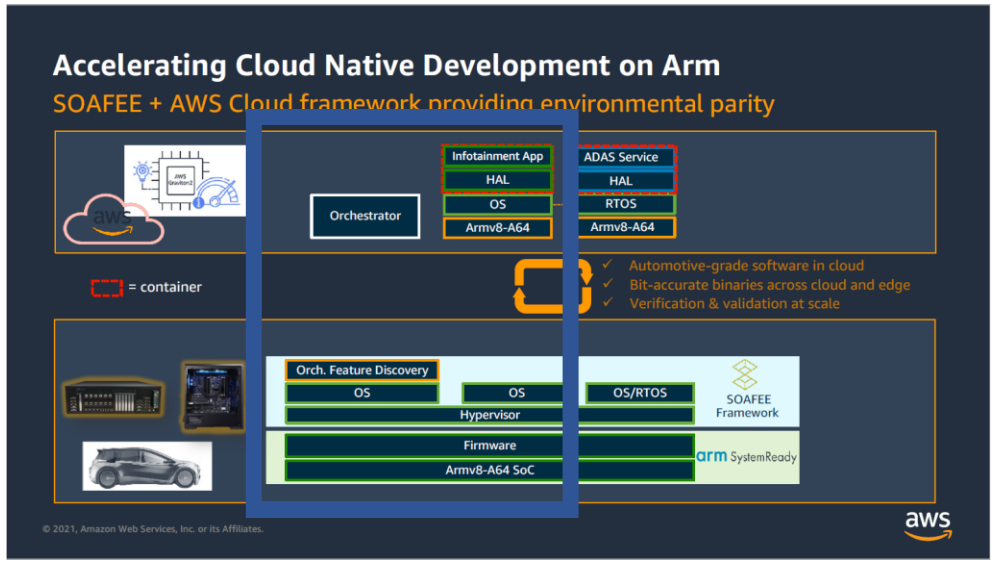
AUTOMOTIVE GRADE LINUX

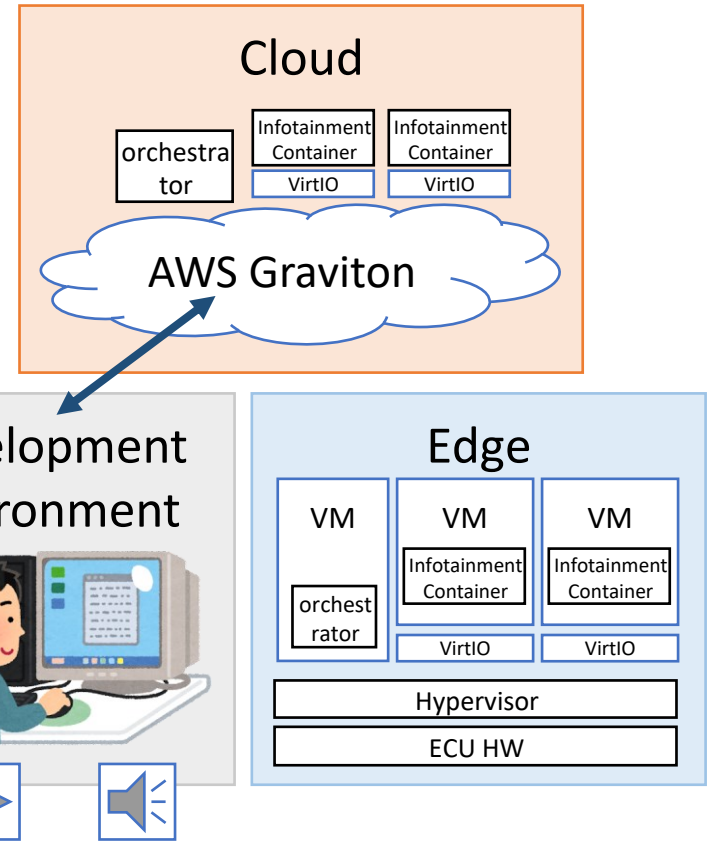
THE LINUX FOUNDATION

VirtIO Work for Cloud-Native Environment - AGL on AWS Graviton

- **Establish a reference environment of cloud-native AGL**
- **Make VirtIO & Orchestration work on both cloud and edge AGL instances, and enable developer to develop HMI services on cloud environment which graphic & audio can be verified on local clients**

Instrument Cluster & IVI which are most related to UI/UX and need rapid development & OTA are one of the automotive devices that enjoy benefits from cloud-native most.





Graphic & Sound are necessary for development with Cloud

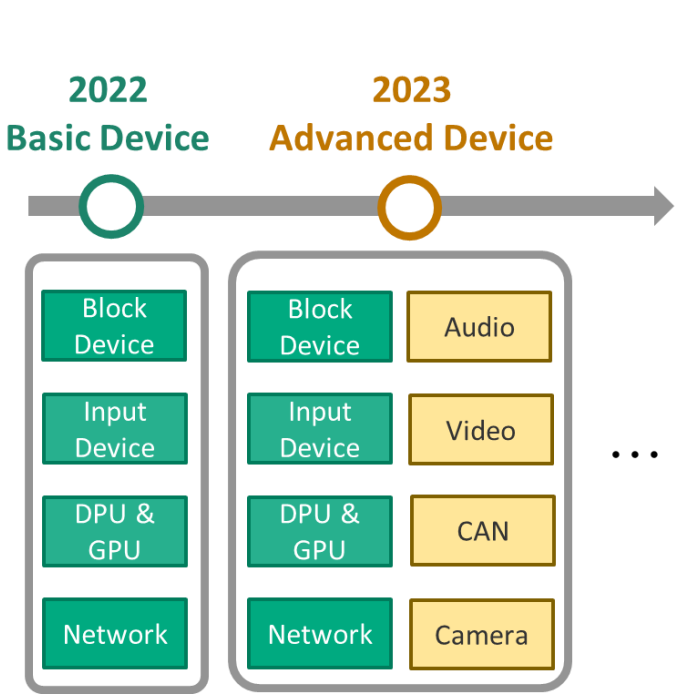
VirtIO Work for Cloud-Native Environment - AGL on Mac

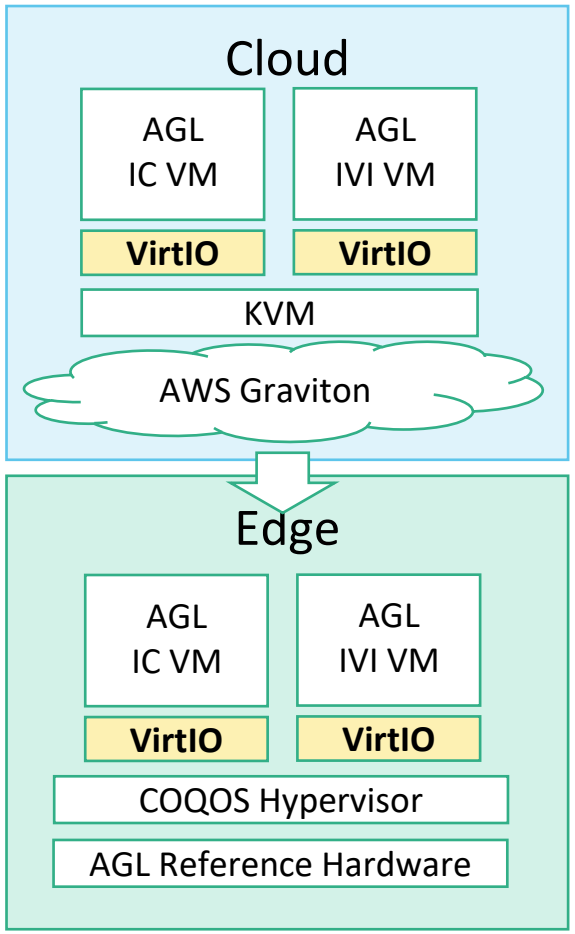
- **EG Member Francois from the company Shokubai has created a demo to run the AGL with VirtIO on the top of MacBook with Apple MacOS 13 virtualization framework.**
- **This can be done without any changes to AGL thanks to VirtIO.**

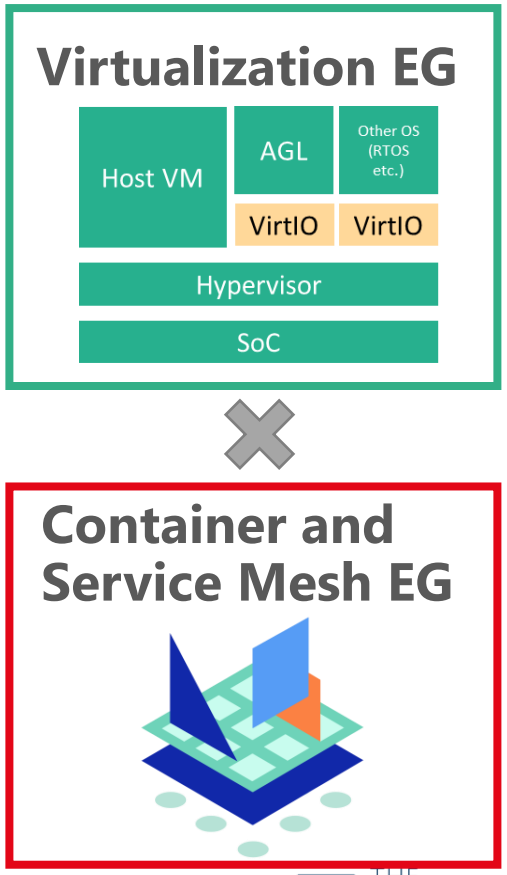
https://www.youtube.com/watch?v=5DT-l2sWeVY

VirtIO Work for Cloud-Native Environment – Demo@CES2023 AGL Booth

- Collaborative efforts between Virt-EG and Container-EG to enable cloud-native AGL
- Identical AGL IVI binary is now able to run on both cloud and edge with graphics visible
- A PoC had been shown in the CES2023 AGL Booth and will show an updated version in Embedded World







VirtIO Work for Cloud-Native Environment

Moving Forwards: Constructing a bright and open future for SDV with AGL

Advance Notice: Cloud-Native Unified HMI demo in Panasonic Kiosk@AGL Booth

Panasonic AUTOMOTIVE

By Integrating Unified HMI with VirtIO-based Cloud Native Development Environment, a "Software-Defined" Multi-Display Cockpit Systems can be achieved to enable a "UX/SW first" and continuously evolved approach of cockpit UI/UX, without dependency on underlaying HW architecture

①Design layout across multi-display on cloud

Layout Design Tool (Stuido-UI)

②Verify HMI in a Virtual Cockpit

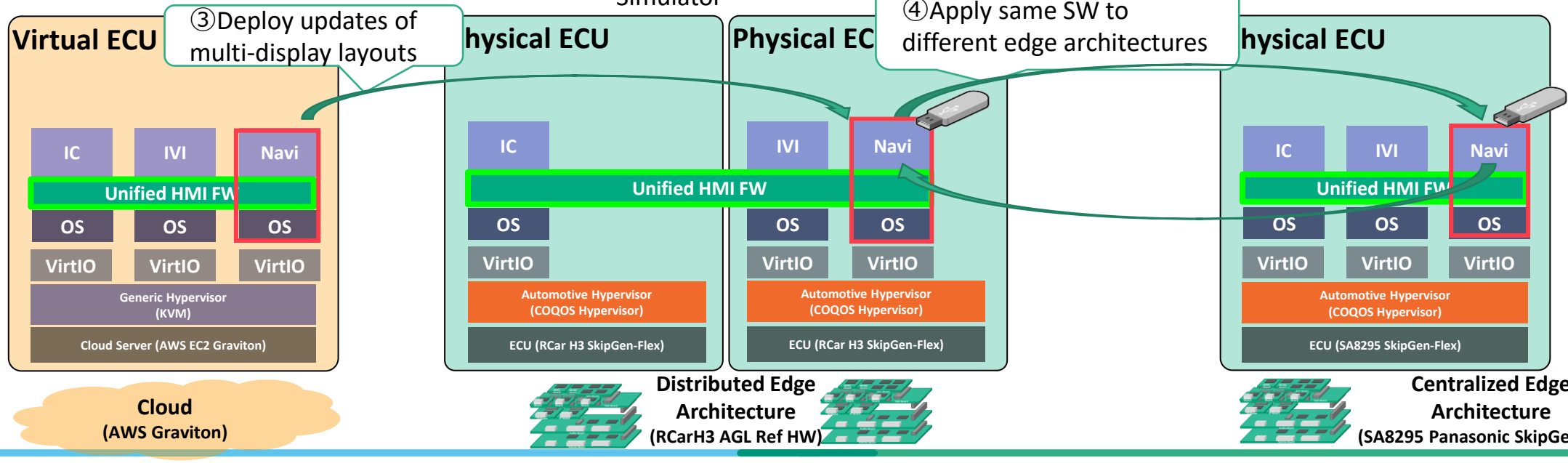
Virtual Cockpit Simulator (Studio-CG)

Demo Points

- Develop cockpit HMI across multi-display in cloud without physical HWs
- Unmodified SW assets to be applied to various HW architectures of different car model/generations

Upgrade from CES2023 Demo

- Unified HMI works on AWS cloud and same software developed virtual on cloud to be able to applied to no matter of edge architecture of distributed/centralized ECU and SoC type.
- Layouts of multi-display can be easily designed on cloud and verified UI/UX in Virtual Cockpit Simulator

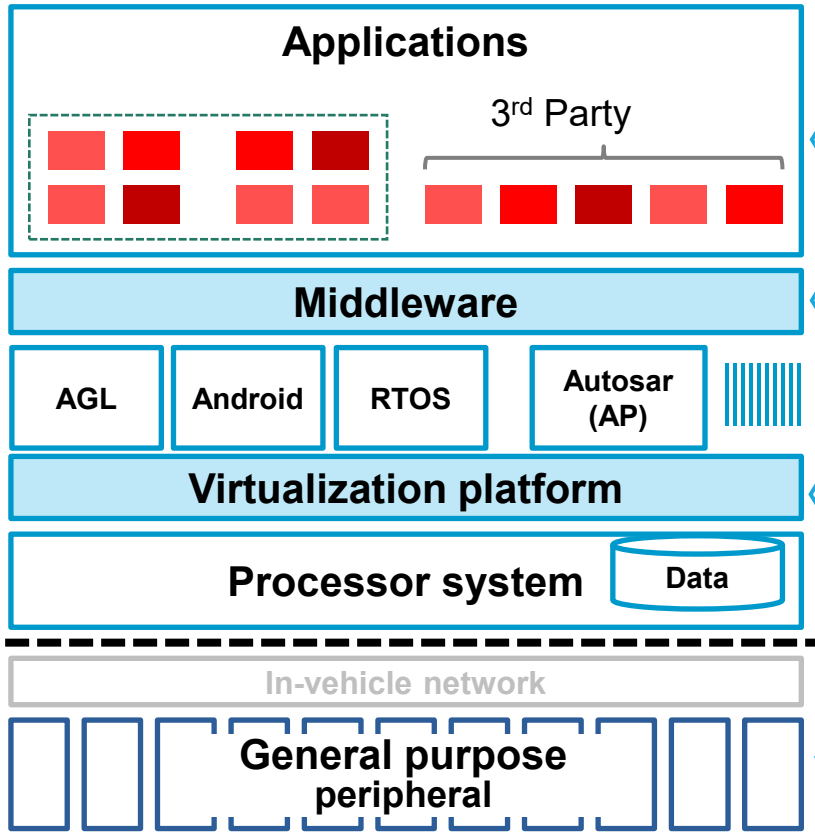


1+1>2 in SDV: Micro-**S**ervices **D**ances With **V**irtualization

Panasonic AUTOMOTIVE

A new Software-Defined Vehicle Expert Group combining expertise of two original groups in VirtIO and Microservice Orchestration has been set up to drive forward AGL development on SDV with even greater speed and efficiency.

Logical architecture



Microservice based application architecture to enable grater flexibility, stability and agility in building automotive applications

Installation of in-vehicle middleware will be accelerated, enhancing ecosystem.

Abstraction of hardware (shift to software-based hardware) → Easier control using apps

Peripherals where technological innovation advances in a unique way based on a standard interface

Container & Mesh Expert Group

Environment Parity of Application-level: Decouple application from operating system

Virtualization Expert Group

Environment Parity of OS-level: Decouple operating system from hypervisor & hardware

Software-defined Vehicle Expert Group

Environment Parity of ALL: Decouple overall software from hardware

Last but Not Least

Join us to co-develop the critical technologies enabling Software-defined Vehicle to define a bright future for vehicles.

Meeting Notes

- Confluence
  - https://confluence.automotivelinux.org/display/VE/Meeting+Notes+2023

Meeting Time

- Zoom
  - Tuesday 11:00am to 12:00pm UTC in Odd Week

Meeting Members

- AVL
- AWS
- Denso
- Harman
- Konsulko
- OpenSynergy
- Open Virtual System
- Panasonic
- Pumped Fuel
- Qualcomm
- Renesas
- Red Hat
- Renesas
- etc.

AUTOMOTIVE GRADE LINUX

THE LINUX FOUNDATION

Thank you