



ACRN

A Big Little Hypervisor for IoT Development

May 2018

Contents

PROJECT OVERVIEW

TECHNICAL OVERVIEW

GOVERNANCE

JOIN US!

ACRN

PROJECT OVERVIEW



Current State of Affairs – IoT Development

IoT device devices requires support for a myriad of HW resources on one platform:

Ex: Camera, Audio, Graphics, Networking, etc.



Needs to be able to support and run multiple operating systems simultaneously

Ex: Linux, Android, Windows, RTOS etc.



Needs to be able to support and run multiple SW tools and applications depending on the IoT device's usage model



Need a way to consolidate ECUs and SW technology investment to lower total BOM costs for IoT devices productization



Virtualization is Key



Current Data Center Hypervisors

- Too large for embedded IoT development
- No safety-critical workload considerations
- Requires too much overhead for embedded development

Current Embedded Hypervisors

- Highly dependent on closed source proprietary solutions
- Expensive
- Makes product longevity difficult
- Hard partition, no ability to share resources

No Open Source Hypervisor solution currently exists that is
optimized for embedded IoT development



ACRN™ is a flexible, lightweight reference hypervisor, built with real-time and safety-criticality in mind, optimized to streamline embedded development through an open source platform

Project ACRN™ Pillars



ACRN™ is a flexible, lightweight reference hypervisor, built with real-time and safety-criticality in mind, optimized to streamline embedded development through an open source platform

Small footprint

- Optimized for resource constrained devices
- Few lines of code: Approx. only 25K vs. <156K for datacenter-centric hypervisors

Built with Real Time in Mind

- Low latency
- Enables faster boot time
- Improves overall responsiveness with hardware communication

Built for Embedded IoT

- Virtualization beyond the “basics”
- Virtualization of Embedded IoT dev functions included
- Rich set of I/O mediators to share devices across multiple VMs

Safety Criticality

- Safety critical workloads have priority
- Isolates safety critical workloads
- Project is built with safety critical workload considerations in mind

Adaptability

- Multi-OS support for guest operating systems like Linux and Android
- Applicable across many use cases

Truly Open Source

- Scalable support
- Significant R&D and development cost savings
- Code transparency
- SW development with industry leaders
- Permissive BSD licensing

ACRN™ & OSV/ISV Vendors



Project's Goal

Provide an embedded hypervisor reference solution to enable OSV/ISVs

A **transparent enabler** that provides:

- A **common architecture** to be used as-is
- A **high quality reference stack** optimized for embedded development

Productize on top of ACRN directly by **adding value** with:

- Proprietary Service OS or RTOS
- Commercial Licensing
- Commercial Support

Moving the industry towards faster overall product development

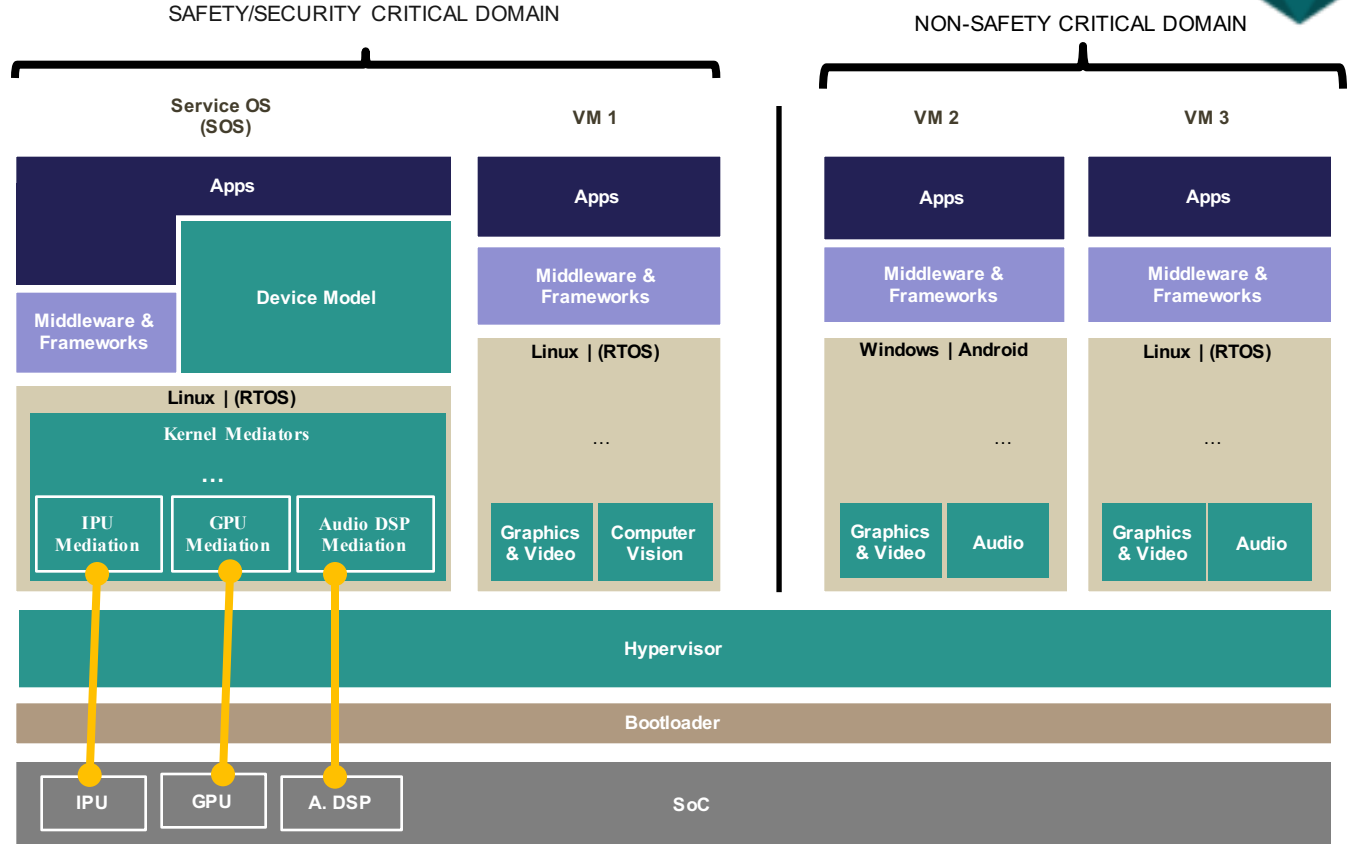
ACORN

TECHNICAL OVERVIEW

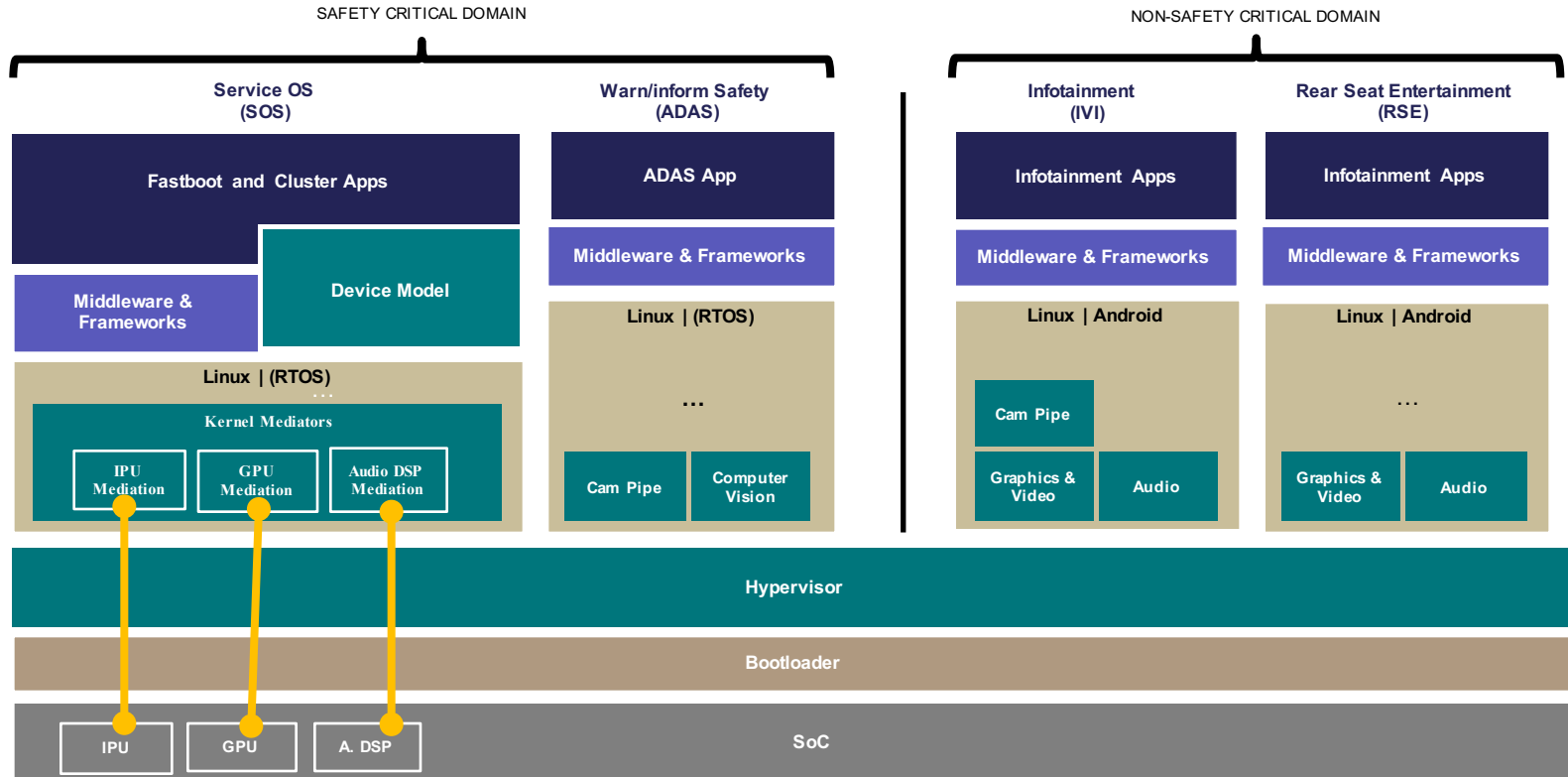
ACRN™ Reference Stack



- Hypervisor sits right on top of bootloader for **fast booting** capabilities.
- **Partitioning of resources** to ensure safety-critical and non-safety critical domains are able to coexist on one SoC.
- **Rich IO mediators** that allow for the Service OS to communicate directly to SoC ensuring low latency for safety critical elements
- Allows for **multiple operating systems** to be supported by one SoC through efficient virtualization



ACRN™ Usage: Automotive Example



Current Project Status



Code

- Support for hypervisor and device model included
- Support for Storage, Network and Console Mediators
- Integration with a Linux-derivative Service OS in progress
- Currently supporting the Intel Apollo Lake NUC platforms
- Project code available for download on [GitHub](#) today.

Documentation

- All project documentation including release notes, API documentation and all other documentation related to the code will be available on project [GitHub](#) today.
- Other project documentation planned includes porting of Ubuntu for ACRN, and other how-to documentation to get a developer up and running on ACRN.

ACRN™

CI

- Currently utilizing Intel CI processes for the open source the project.
- Planned collaboration with the open source community on input for CI infrastructure for the project to a neutral host site

On the Horizon

- Mediator Support for Audio, IPU, GFx, CSE, TSN etc. planned for expanded embedded IoT use cases of the project
- Creating a process to address safety-critical requirements together with the project community
- Adding additional hardware platform support to create a hardware agonistic project

ACORN

GOVERNANCE

Governance Ethos



Community Involved

- Encourage and welcome participation in the project

Transparency

- Committed to working transparently in the open to encourage the open source community's involvement

Code > Currency

- The project is committed to a true open governance where code contributions is the main objective

Governance: Technical Steering Committee



ACRN

JOIN US!

Project ACRN™

For Embedded Developers



- Build complex embedded systems requiring various levels of safety-criticality
- Resource sharing allows for maximizing potential of resource-constrained devices while ensuring safety critical workloads are given priority to platform hardware
- Open source code allows for developmental cost savings as transparent code ensures code compatibility, flexibility in vendor selection as well as helping to ensure longevity of product lifecycle while reducing support and maintenance costs of the product's life.
- Small footprint, lightweight code base to ensure inclusion of only necessary code optimized for embedded development. No superfluous code.



Call to Action



Join us!

If you support the ACRN project and feel that this is the right thing for the embedded ecosystem, join us in moving this project forward together as a community member.

We need code contributors, users, and project direction influencers!



Contribute code!

Make a difference to the project by committing code, help us become a better project.

Project code merged in the past 6 months allows you to become a voting member of the Technical Steering Committee.

Project PR Plan



Q1

ELC North America March 12-14

- Keynote on March 13th announce project with at least 1 automotive partner and endorsements from early adopters
- Project will be hosted by Linux Foundation. Press release by Linux Foundation to announce project
- Demo and Session at conference with project
- Open source project via GitHub
- Neutral, open source project name
- Website

Q2-Q3

Continue to build
Project momentum

Q4

ELC Europe October 22-24, 2018

- Beta features of project are available
- Press release to announce the partners that signed on to the project and to announce the β features release of the project
- Fully formalize TSC with voting members via meritocracy



ACRN

SAFETY/SECURITY CRITICAL DOMAIN

NON-SAFETY CRITICAL DOMAIN

