Consolidate Real-Time and HMI with ACRN Hypervisor

Jason Chen/Fengwei Yin/Jack Ren, Intel ACRN Team

Table of Contents

What is ACRN™

- The Big Little Hypervisor for IOT

A flexible, open-source, lightweight hypervisor for IOT workload consolidation

A Linux Foundation Project Launched in March 2018

https://projectacrn.org

Value Proposition

Small Footprint

- Optimized for IOT class solutions
- Significantly smaller footprint than datacenter targeted hypervisors

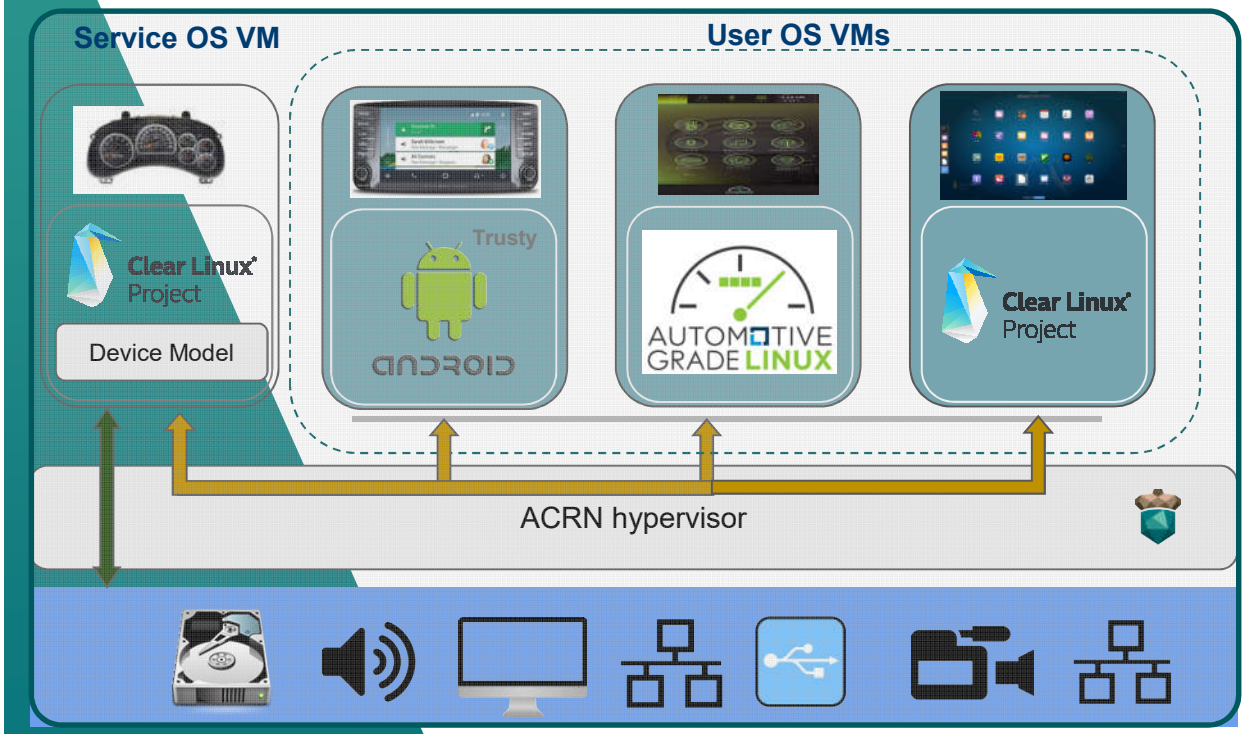
Heterogeneous Workloads Consolidation

- Real time & Non-Real time
- Functionally Safe & non-safe

Open-source with Flexible Licensing

- BSD license enables proprietary Guest OS
- True Open source with a vibrant Community

ACRN 1.0



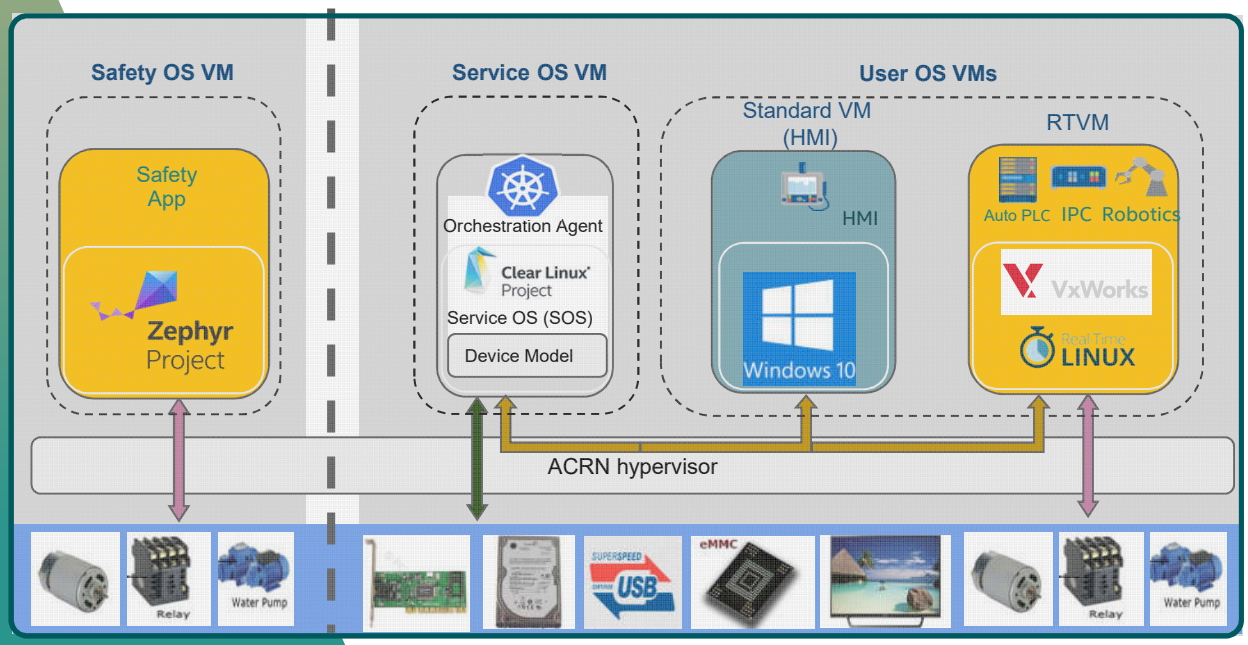
Ready for Production

- **Released in May 2019**

Key Features

- **Safety and Security Isolation (Cluster + IVI)**
- **Extensive Sharing Capabilities**
- Graphics, media, USB, audio, camera etc.
- Advanced DMA/graphics buffer sharing
- **Multiple OS Support**
- Clear Linux, Yocto, Ubuntu
- Android, AGL, AliOS
- **MISRA-C Compliance**

Industrial: Safety + RT + HMI



Key Challenges:

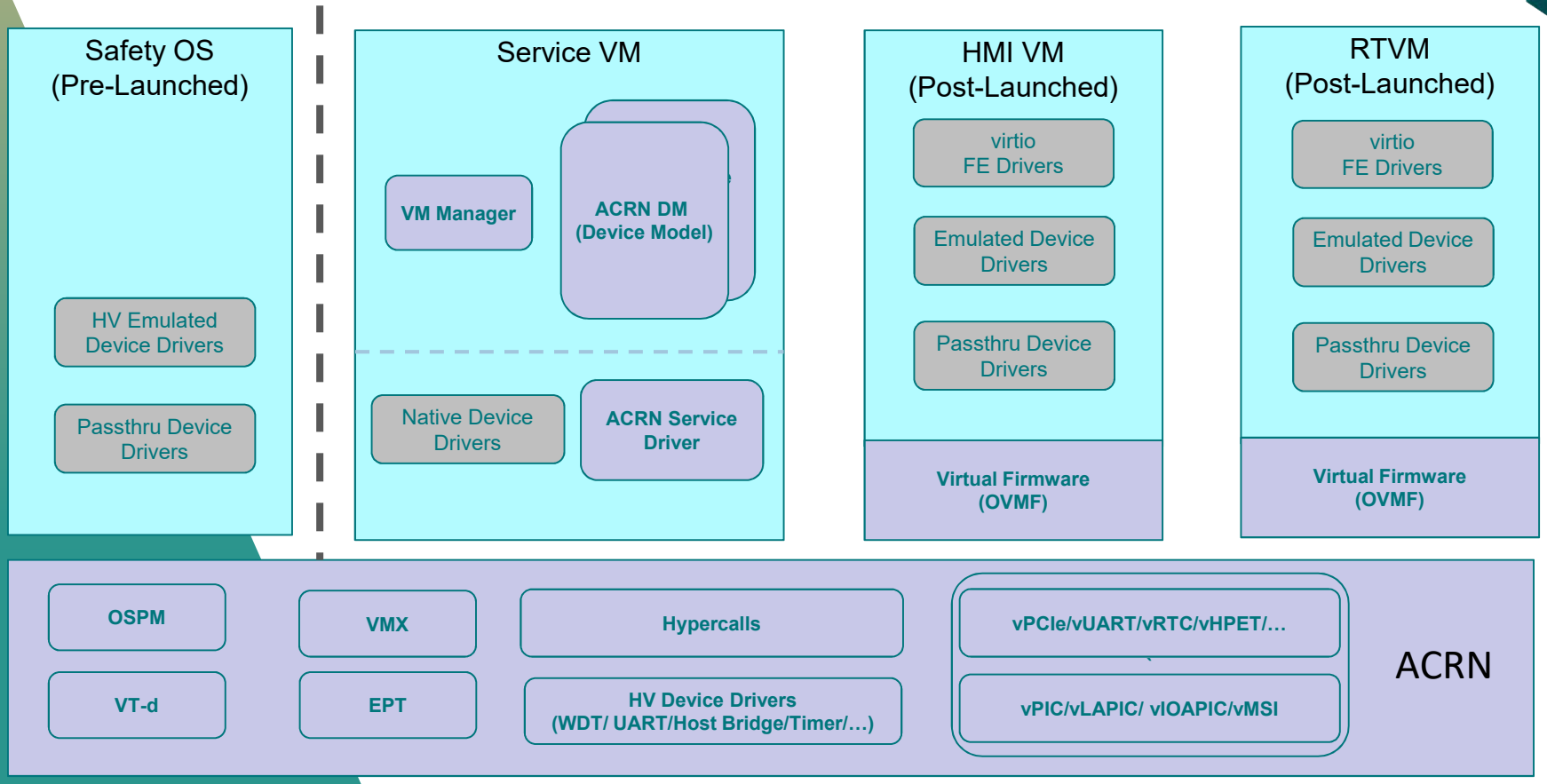
❑ **Mixed Criticality:**
- Real-Time vs non Real-Time
- Safety vs non-Safety
- Isolation vs Sharing

❑ **Real-Time (Hard / Soft)**
- GBE packet IO control loop < 12us
- MSI interrupt latency < 4us
- Cyclictest jitter < 10us

❑ **HMI**
- Window10

❑ **Functional Safety**
- IEC 61508-3 (Industrial)
- ISO 26262 (Automotive)

ACRN 2.0 Architecture

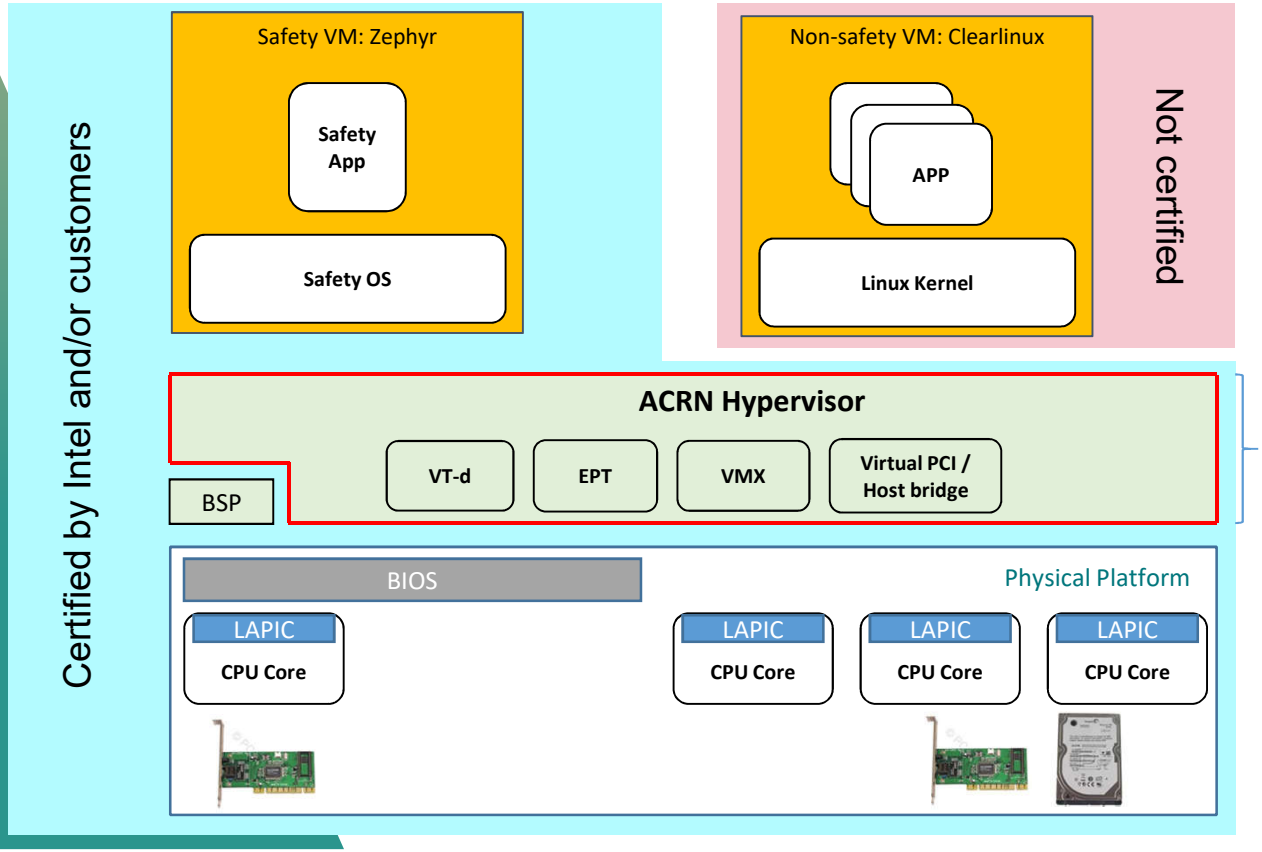


Industrial Usage: Device Mapping Table

Devices	Soft RTVM (RT Linux)	Hard RTVM (VxWorks)	HMI (Windows/Linux)
RTC	Virtual	Virtual	Virtual
PCI	Virtual	Virtual	Virtual
UART	Passthru	Passthru	Virtual/Passthru
GBE Network	Virtual (PMD) /Passthru	Virtual (PMD) /Passthru	Virtual
TSN (i210)	Passthru	Passthru	N/A
Storage	Virtual (PMD) /Passthru	Virtual(PMD) /Passthru	Virtual
FPGA	Passthru	Passthru	N/A
GPU	N/A	N/A	Mediated Passthru
Audio	N/A	N/A	Passthru
USB	N/A	N/A	Virtual
Watchdog Timer (WDT)	Virtual	Passthru / Virtual	Virtual

ACRN + Zephyr as Safety Domain

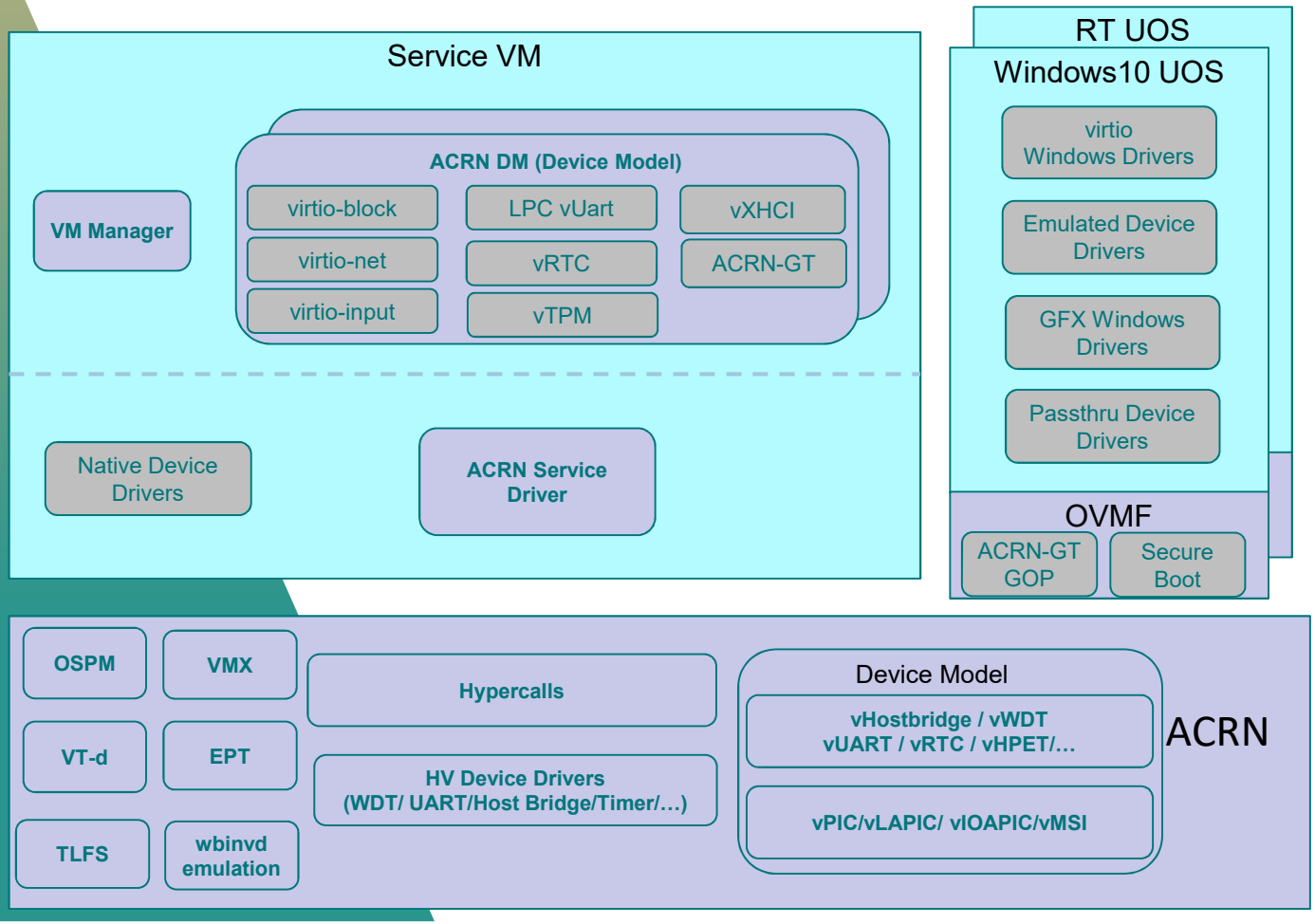
Use Scenario



- 2 Partitions with mixed-criticality
- Static core & memory partitioning

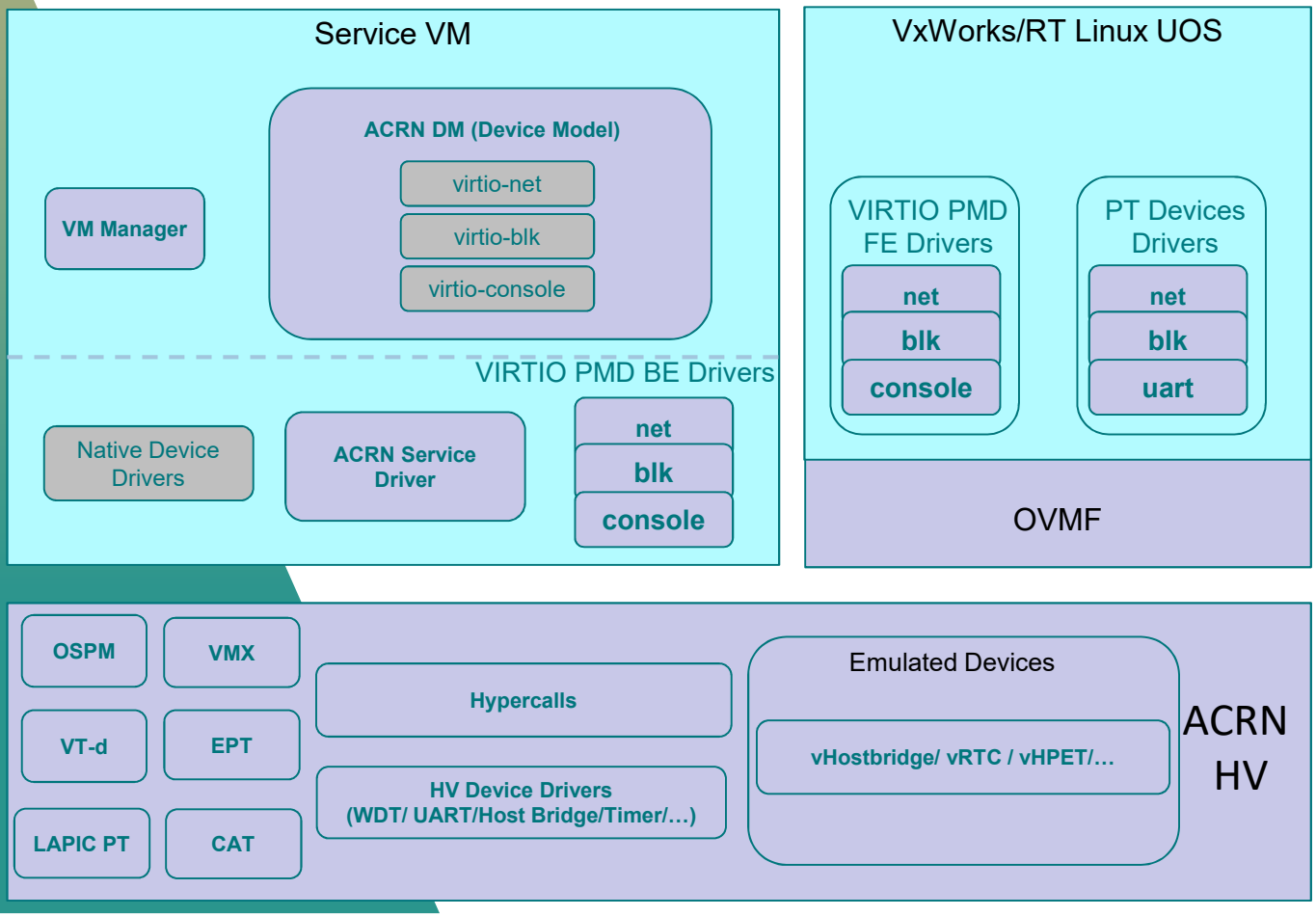
IEC 61508 & ISO 26262 Certification Scope for ACRN

Windows as HMI domain



- ACRN-GT GOP is added into OVMF to support windows early display and windows installation display.
- Support OVMF secure boot with vTPM for Windows secure boot chain.
- Support the Microsoft defined TLFS(Hyper-V Hypervisor Top-Level Functional Specification) minimum requirements and optional performance optimization requirements.
- Utilize Microsoft DISM tool to pre-install virtio-win drivers and gfx driver to the Windows install .iso file.
- Use GT-CLOS to prevent Windows from Cache interference

VxWorks/RT Linux as Control domain



- ❑ Pass-through LAPIC (except ICR/XAPICD/LDR) to:
  - avoid VM-exit
- ❑ Enable CAT to:
  - isolate cache for RT VM
- ❑ Enable virtio BE/FE as PMD to:
  - avoid VM-exit
- ❑ Configure native BIOS to disable:
  - Hyper-threading
  - Speed Step
  - Speed Shift
  - C-state
  - GT RC6
  - GFX Lower Power Mode
  - Native ASPM
  - …

Configuration for Real Time Latency Evaluation

Configuration:

- HW: Intel(R) Core(TM) i7-8650U CPU @ 1.90GHz, 8G Memory, 1M L2 cache, 8M L3 cache
- Benchmark: cyclictest (measure the scheduler jitter), running in Real-Time VM

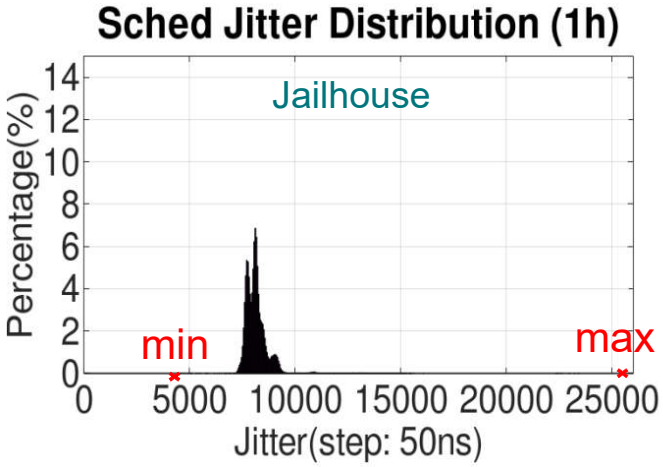
ACRN:

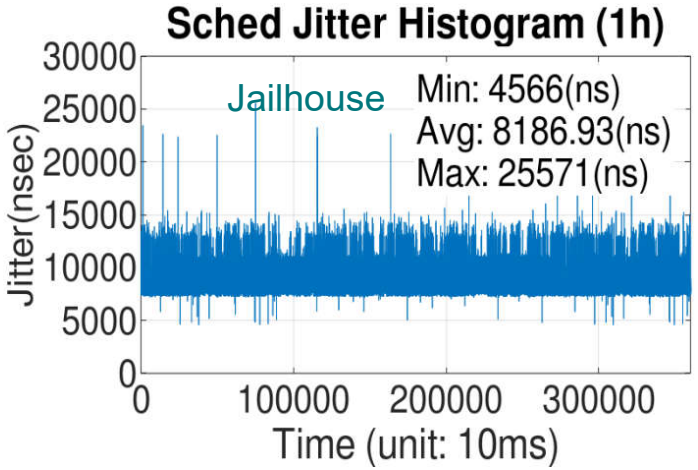
- Service OS VM: Linux kernel v4.14.68-rt42
- Real-Time VM: Preempt-RT Linux: 4.14.68-rt42, with 2GB memory

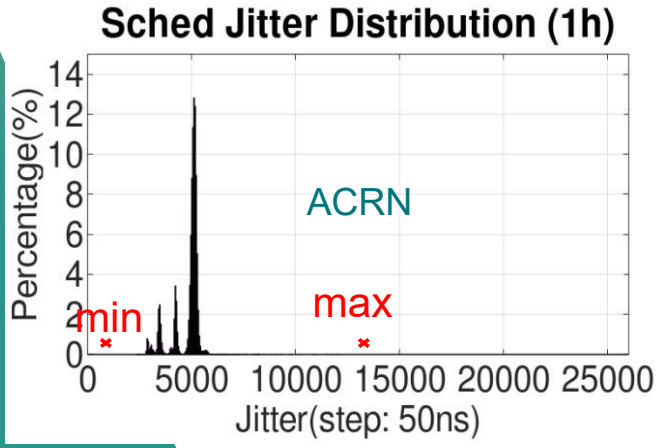
Jailhouse:

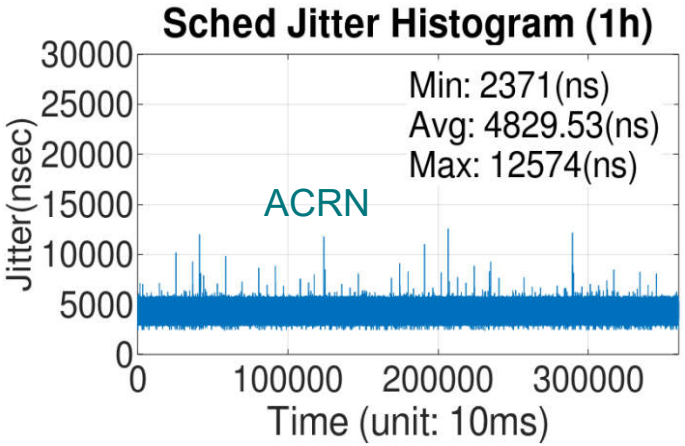
- Root cell: Linux kernel v4.14.68-rt42
- Non-root cell: Preempt-RT Linux: Linux:4.14.71-rt44+, with 2GB memory

cyclictest: ACRN vs Jailhouse









Data reused from paper: <<ACRN: A Big Little Hypervisor for IOT Development>>, VEE'2019

ACRN Open Source Roadmap in 2019

	*Feature and dates for reference only and subject to change without notices				
Area	**V1.0@Q1'19**	**Q2'19**	**Q3'19**	**Q4'19**	**2020**
HW	• APL NUC (UEFI) • KBL NUC (UEFI) • APL UP2 (SBL)	• APL NUC (UEFI) • KBL NUC (UEFI) • APL UP2 (SBL)	• APL NUC (UEFI) • KBL NUC (UEFI) • APL UP2 (SBL) • Denverton SoC	• APL NUC (UEFI) • KBL NUC (UEFI) • APL UP2 (SBL) • Denverton SoC	
Hypervisor	• Power Management (S3/S5) • ACRN partition mode • Local APIC passthrough • Real-Time VM support	• VxWorks as Guest • Zephyr as Guest • ACRN Real-Time baseline • ACRN Hybrid mode • OVMF for Clear Linux Guest support • IOMMU interrupt remapping • VM Configuration Unify	• Real-Time for Preempt-RT Linux • Real-Time for Pseudo Locking • Real-Time profiling tool • Real-Time Performance optimization • Kata Container support • OVMF GOP driver for GVT-g • Device Posted Interrupt(PI) • Multiple IOAPIC support	• Windows as guest • VxWorks as guest • Zephyr as Safety OS • CPU sharing • Docker support based on Kata Containers	
I/O virtualization	• GPIO virtualization • QoS – Support RunC • TPM2.0 Sharing (Security)	• SR-IOV for share mode • HPET Virtualization • Open vSwitch • I2C virtualization	• USB hub virtualization	• Kubernetes support based on Kata Containers • GVT-g Gen11 support • GVT-g for Windows as Guest	

Call to Action

https://projectacrn.org

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.

All Contributions Matter

In open source projects a contribution can be anything which helps the project to accomplish its mission. Examples of Contributions beyond just code include:

Financial Assistance, Requirements Gathering, Documentation, Testing, Bug Reporting