

Instrument Cluster Container Integration Demo

Background of this demo

AGL Instrument Cluster Expert Group aim to :

- Create a base platform for the instrument cluster by using Linux.
- Solve some of the product development issues in AGL community.

Automotive challenges:

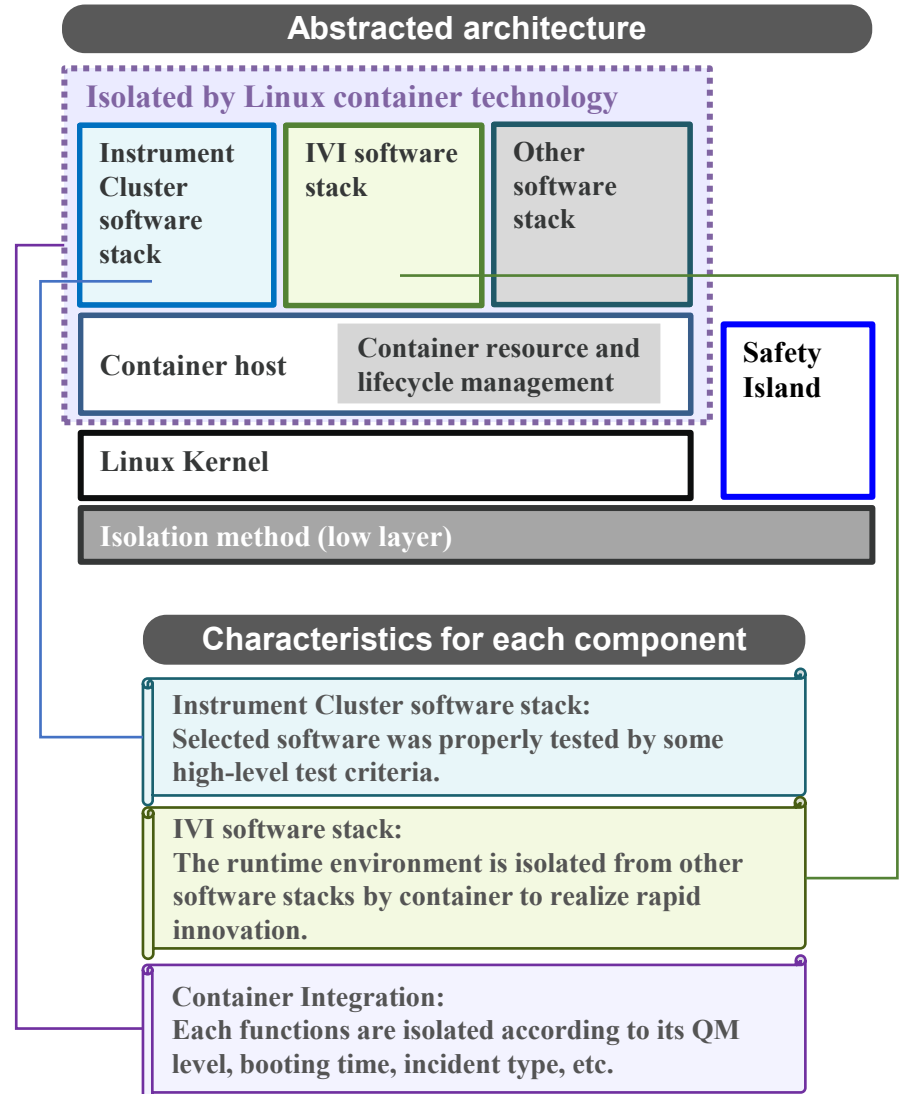
- Automobiles need to save energy and reduce chips to solve environmental problems and a global chip shortage.
- It is necessary to reduce chips and harnesses by concentrating them in SoC to reduce weight and energy consumption.

Integrated into a single SoC challenges:

- Rapid Innovation and higher quality Management (QM) must be compatible.
- QM space needs to be protected, but “rapid innovation” space needs to be updatable.

Our concept and advantages:

- In-Vehicle Infotainment (IVI) and Instrument Cluster on one SoC using Linux container technology, less complexity than hypervisor + RTOS combination.
- Efficient hardware resource allocation.
- Not affect Instrument Cluster from IVI container update.



Overview for demo system

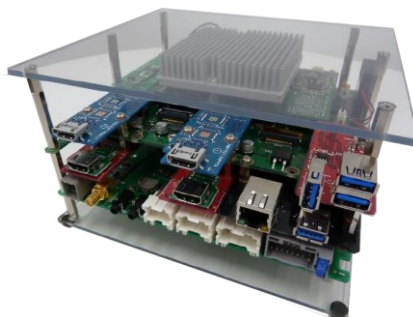
Surround Speakers:
Both IVI sound and alarm sound play on one set of surround speakers. When an alarm sound is playing, IVI sound is forced muted by audio management mechanism between containers.

Surround Speakers



AGL Ref. HW:
All software run on one R-Car H3 SoC (octa core). That has 8GByte memory, 2 HDMI Display out and more connectivity.

AGL Ref. HW



IVI :
IVI feature is displayed on a touch panel display. This demo support two type of IVI both portrait and landscape. Those are possible to update (exchange) to any one.

IVI Display



Web UI by tablet:
Web UI shows the demo system status. That support shows information for each container, which IVI container running on, and CPU resource usage. The container exchange can operate using Web UI.

Web UI



Instrument Cluster Display



Demo Ctrl. Keyboard.



Instrument Cluster:
Instrument Cluster displays speed, vehicle alert, and more information to the driver using a display. The alarm sound plays by this feature if needed.

Demo Control Keyboard:
The demo control keyboard provides the console of this demo system. That supports selecting a container to exchange, container force rebooting to IVI, and vehicle fault injection to the instrument cluster.

Instrument Cluster Container Integration Demo

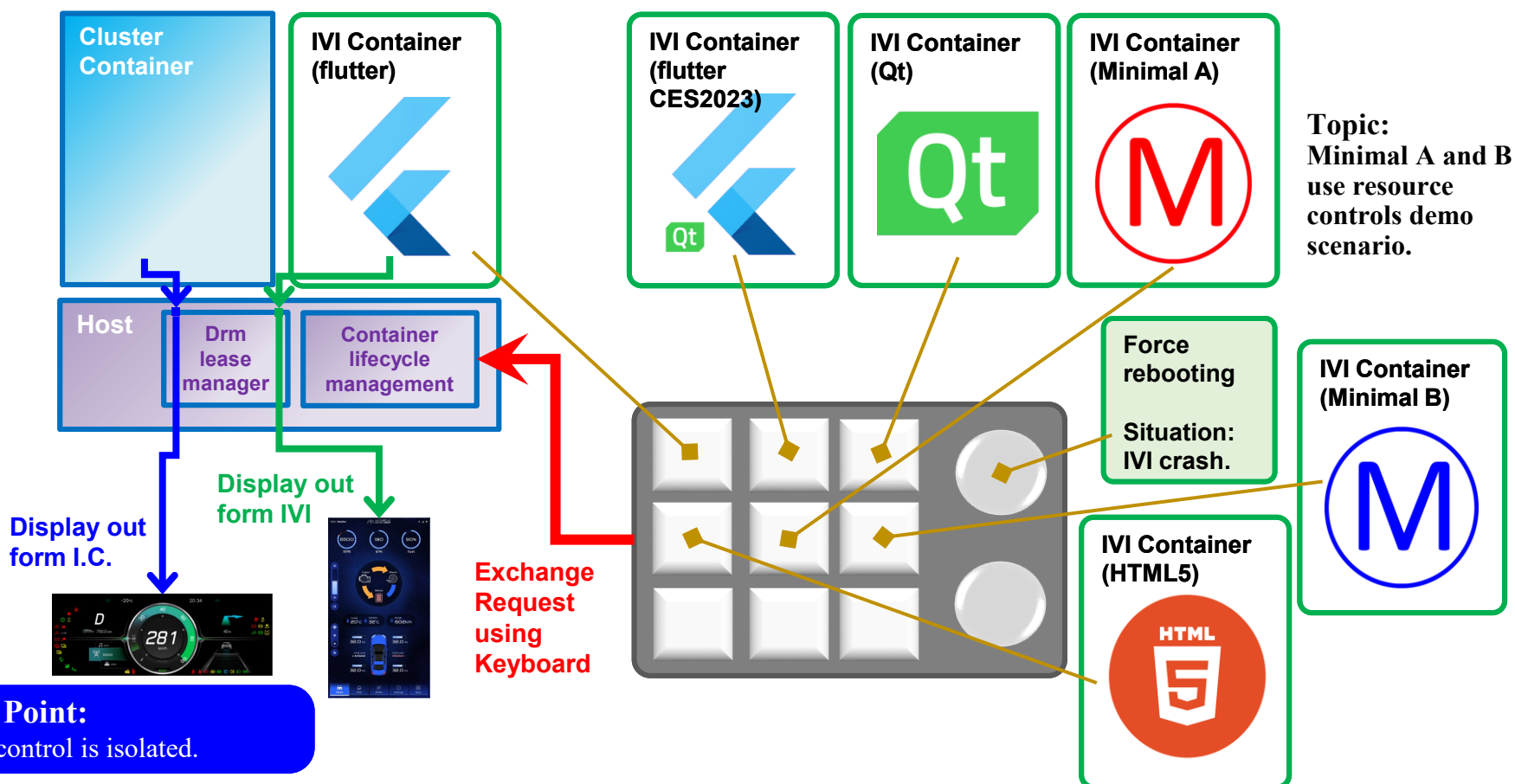
Container Update (Exchange)

Important Points:

This demo shows the safety software upgrade use-case in runtime for IVI without affecting the instrument cluster. The A/B update method is the safety update technique for Linux systems.

This demo has six pre-installed IVI containers such as flutter based latest IVI, flutter based ces2023 version, HTML5, Qt, and two minimal IVIs. This situation is a similar scene to exchange for A/B updating.

When you select a new IVI using a demo control keyboard or a web UI, this system will shut down, exchange, and boot the new IVI. In this timing, not affect a instrument cluster such as display information and alarm sound playing.



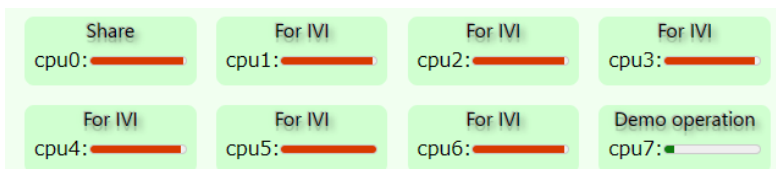
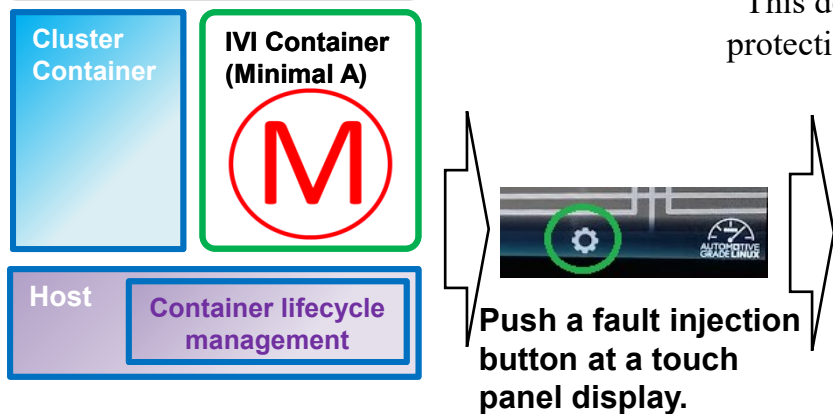
Resource Control

Important Points:

The instrument cluster shows critical information about the vehicle to the driver. This function must not be affected by IVI system incidents.

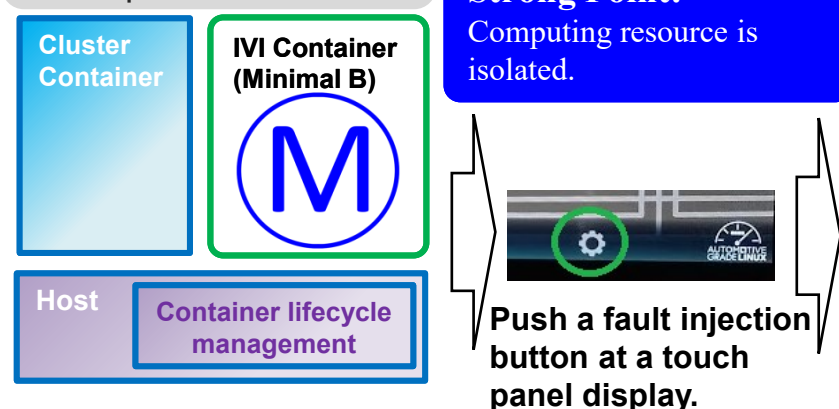
This demo shows the CPU resource protection from the IVI incident. When this protection is enabled, the instrument cluster is not affected by CPU overload in IVI.

Without protection: **Red M**

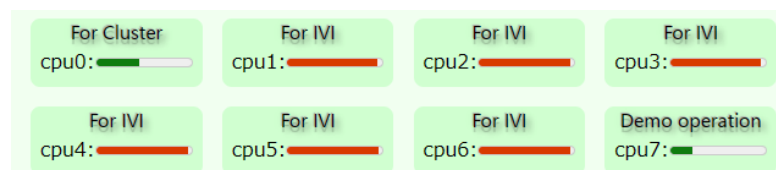


When this system integrates without resource control, IVI overload affects all of the systems. That uses full resources of cores (without demo operation core). As a result, the instrument cluster slows down to drawing and causes choppy alarm sound.

With protection: **Blue M**



Strong Point: Computing resource is isolated.



When this system integrates with resource control, IVI overloads do not affect all of the systems. That uses its pre-assignment resources only (without instrument cluster and demo operation core). As a result, the instrument cluster keeps drawing and alarm sound. This protection is already supported in AGL upstream.

Instrument Cluster Container Integration Demo

Transparent Audio Management

Important Points:

In-vehicle system has many sound sources such as radio, media player, navigation, and alarm sound. These sources have priority based on their role.

AGL has an audio management feature for IVI already. This demo shows the inter-container audio management feature.

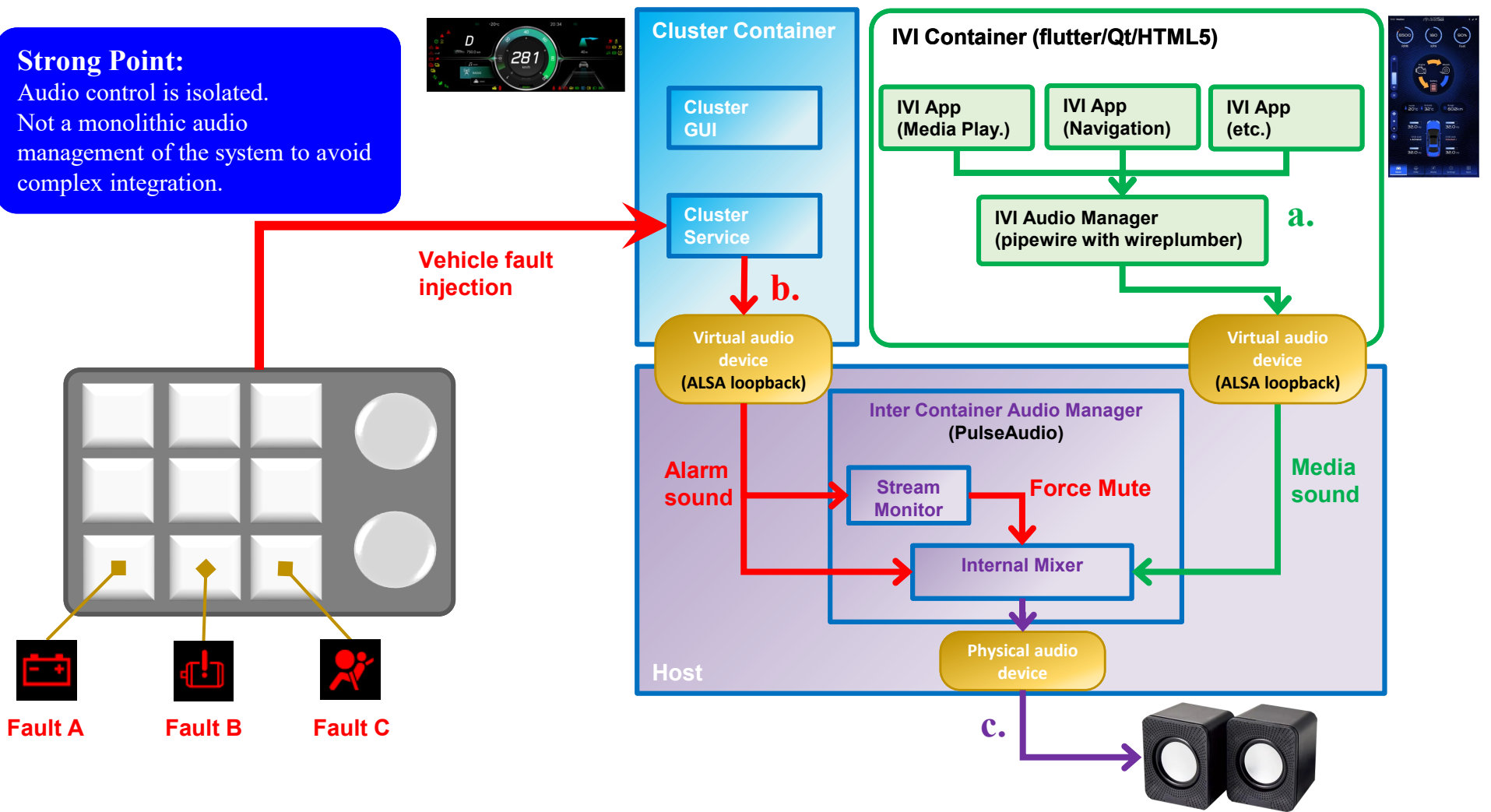
Our architecture realizes transparent audio management. That uses layering architecture, not a monolithic audio management all of the system.

Use-case scenario:

- Generally, only an IVI sound (e.g. Media Player) is playing. When the navigation plays a guidance sound, it's managed by the IVI internal audio manager.
- If the vehicle ECU sends a fault signal to the instrument cluster, that shows a telltale and plays an alarm sound. In this timing, IVI sound shall be forced mute by inter container audio manager.
- All managed sounds out to the physical audio device mixed by the internal mixer.

Strong Point:

Audio control is isolated.
Not a monolithic audio management of the system to avoid complex integration.



Additional Information

You can get more information for AGL.

For ALL

[AGL Official Web.](#)

[AGL CES Info.](#)

[AGL Publication](#)

[AGL X\(twitter\)](#)

For Japanese

[AGLJP Portal](#)

[AGLJP X\(twitter\)](#)

