

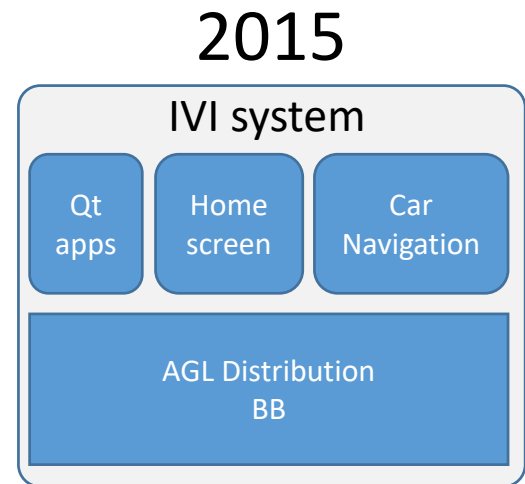
UI and Graphic EG

AGL would evolve as a Cockpit system more?

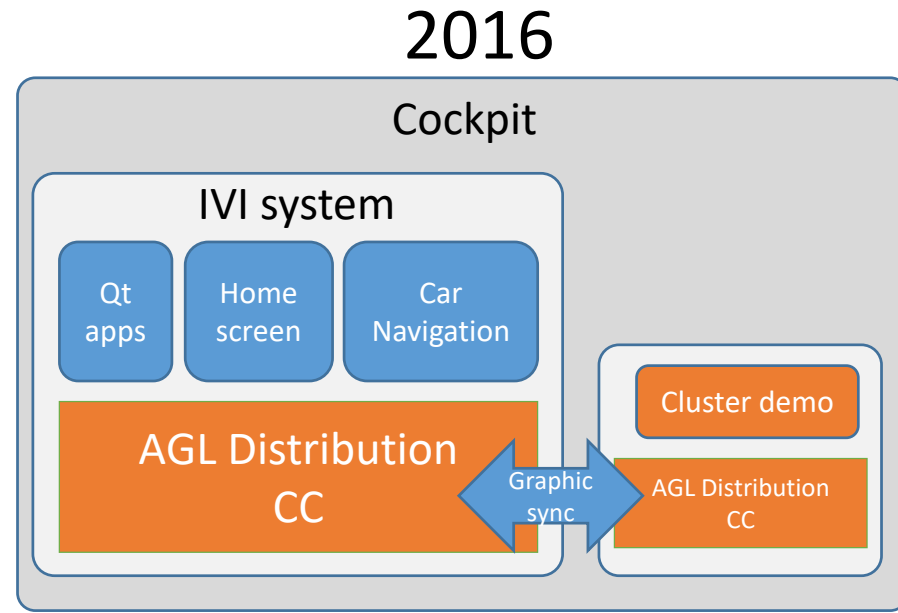
- To be collaborated with other Operation systems.
- To support more application.



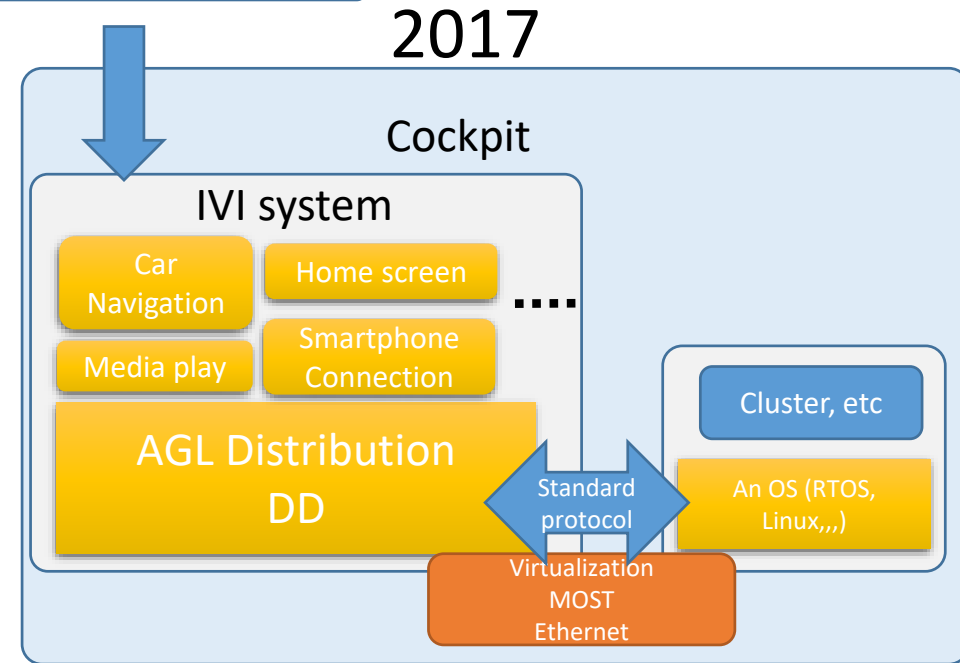
Get Synergy from Application vendors who CAN NOT contribute app as open source. Multimedia, Car Navigation, HomeScreen



AGL Distribution releases for Single IVI system



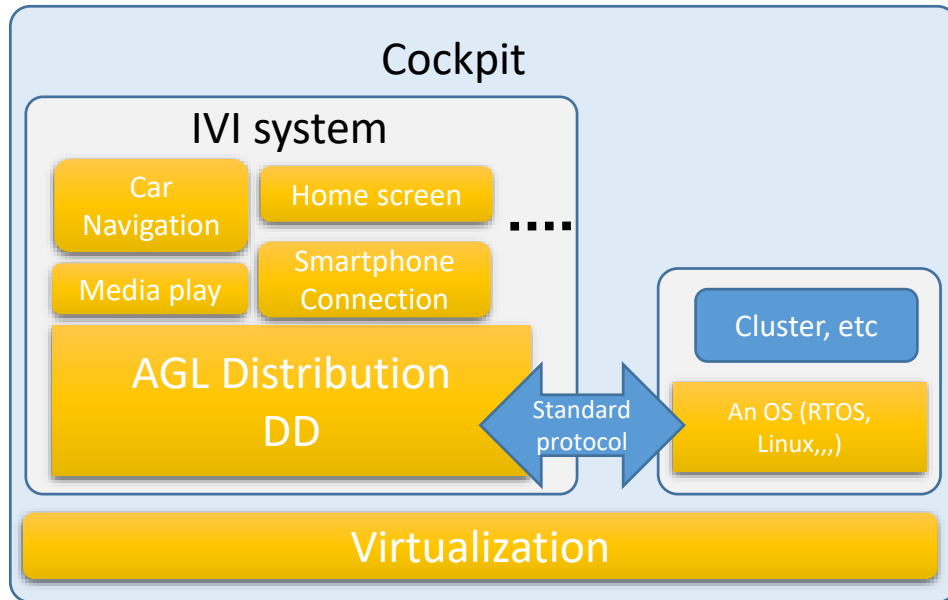
AGL Distribution collaborates another ECU; E.g. Cluster Demo.



- The reduction of lead time to integrate commercial application.
- Standard protocol Collaboration with OSs
 - Virtualization support

Evolution continuously

Standard protocol Collaboration with Various Operation systems.



- An Operation system will be replaced by each OEM/Tiers for product
 - OS will be various
 - RTOS, Linux,,,,
 - The way of physical will be various as well
 - MOST, Ethernet, LVDS, Virtualization shmem,,,,,,

Goal: AGL distribution support standard protocol to be connected to Other OSs easily.

Solution:

- Graphics:
 - Make the logical protocol as standard in Wayland/Weston (*) Wayland/Weston is now default of Fedora. Good place to contribute.
 - Contribute Plugin to hide physical.

[<- DENSO as a UI and Graphics EG lead, would like to lead this activity.](#)

(*) engineers who has an relationship with Wayland/Wesson needs

UI and Graphics Task

- Internode display protocol
 - Standard protocol Collaboration with Various Operation systems.
- Wayland update to 1.11
- Refactor Home Screen including splitting out window Manager
- Theming and skinning
- Improved PulseAudio and Audio Management configuration and policies.
- Replacement of QtMultimedia for media management?
- Speech services API and integration into reference apps.

This session

2/10 9:00-

2/8 session
By N.Nishiguchi

Graphics sharing protocol for multiple ECUs environment

Wataru Mizuno

Advanced Driver Information Technology Corporation

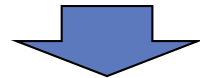
ADIT is joint venture company of DENSO Corporation and Robert Bosch GmbH/Robert Bosch Car Multimedia

- **I worked for ICT Dept. DENSO Corp 2014-2016**
 - ◆ I was in charge of development of future cockpit system
 - HMI to reduce driver's distraction

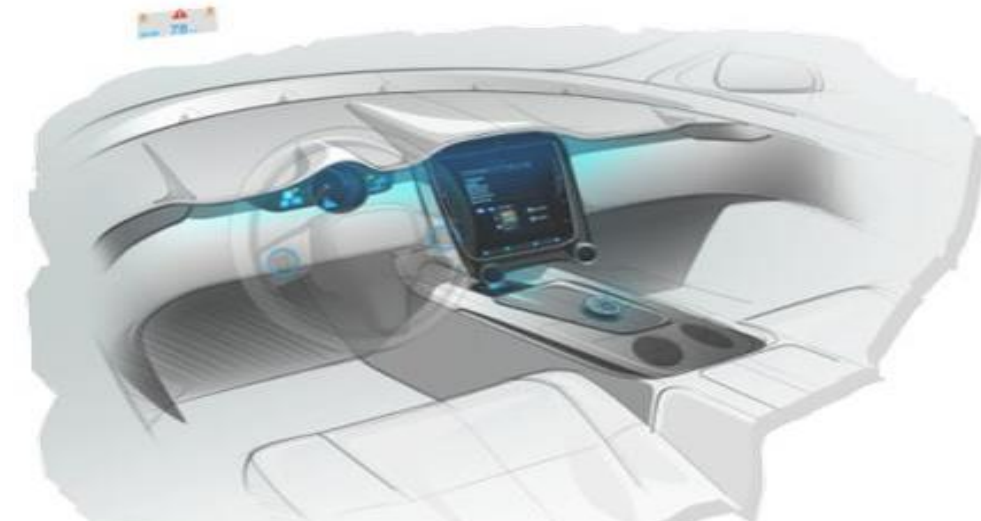
- **I started to work for ADIT since 2016**
 - ◆ I am in charge of development of graphics middleware
 - Wayland / Weston
 - Wayland IVI extension

- Information for drivers is increasing
 - ◆ Increase of functions in each ECUs
 - Safety functions, telematics, V2X ...
 - ◆ Increase of displays
 - Increase of TFT Cluster
 - Increase of HUD(Head up display)

-> Too much information becomes distraction



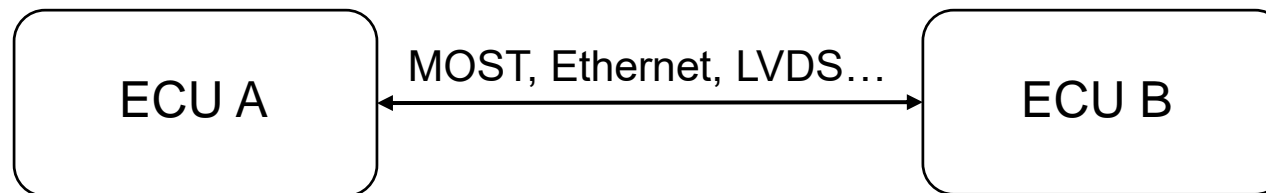
- More comprehensive HMI is needed
 - Display information on appropriate place and time for driver
 - Comprehensive UI design



Future concept cockpit of DENSO

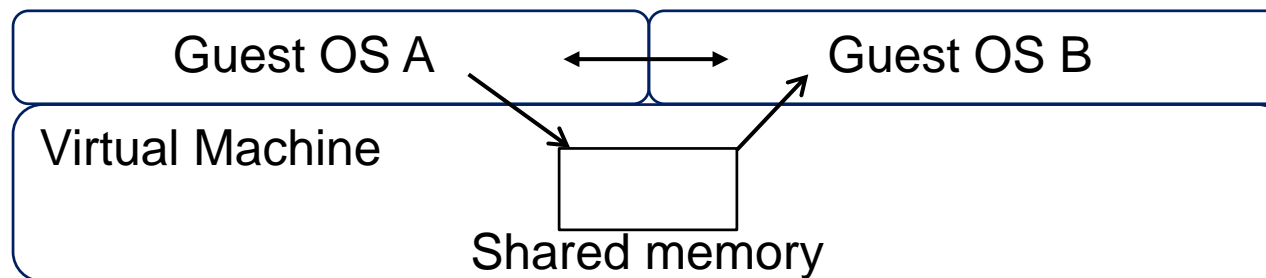
Graphics sharing between multi ECUs is needed

- Sharing between multi ECUs
 - ◆ Data are shared over network



Requirement : Same application is available on both single ECU and multi ECUs

- Sharing between guest OSs on virtualization environment
 - ◆ Data are shared on shared memory or over virtual network



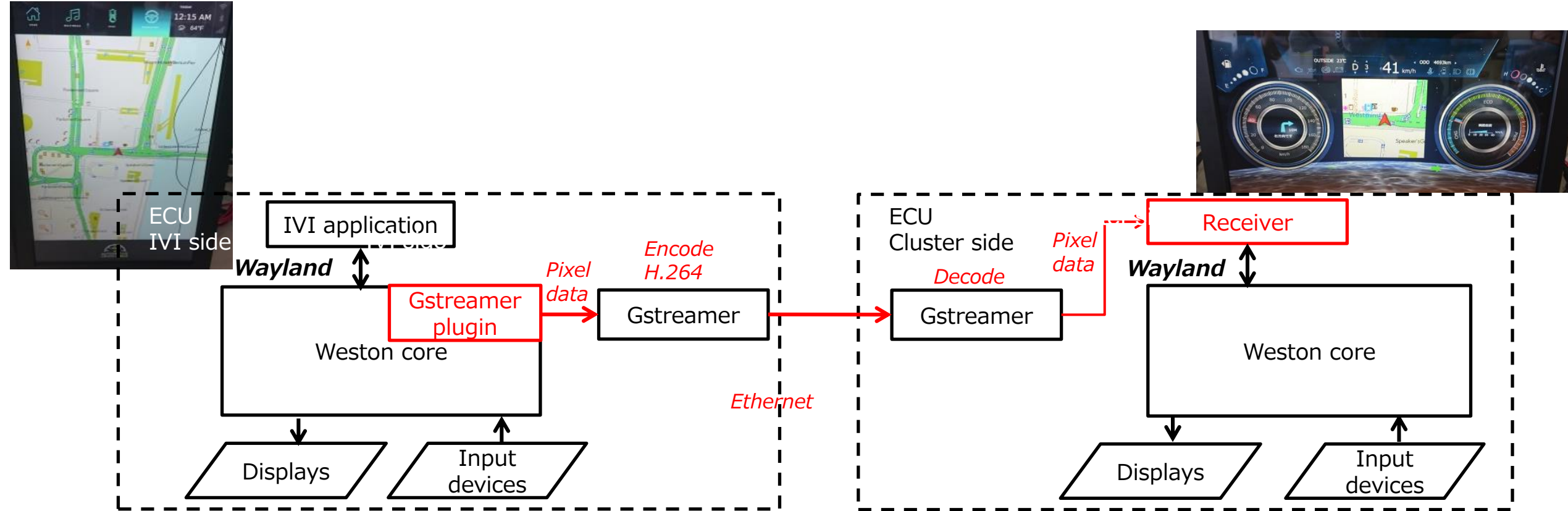
Requirement : Same application is available on various kind of systems
Graphics sharing by using shared memory

Both graphic sharing mechanism and protocol are needed

- Share navi map between IVI and Cluster
 - ◆ IVI side navi map is shared to the center of the cluster side



■ CES Cluster demo architecture



We realized graphics sharing mechanism


- **Implement the protocol for sharing graphics between multi ECUs**
 - ◆ IVI side send graphics information to Cluster side
 - Cluster side can change layout the kind of graphic
 - ◆ Cluster side send message complete rendering, request graphics..., to IVI side
 - Complete message is used for synchronization between multi ECUs
 - Request message is used when Cluster changes layout
- **Share the graphics by surface**
 - ◆ Good for sharing rear view camera, maps...
 - ◆ For standardization mechanism
 - Wayland treats graphics as surface



Not composited rear view camera surface



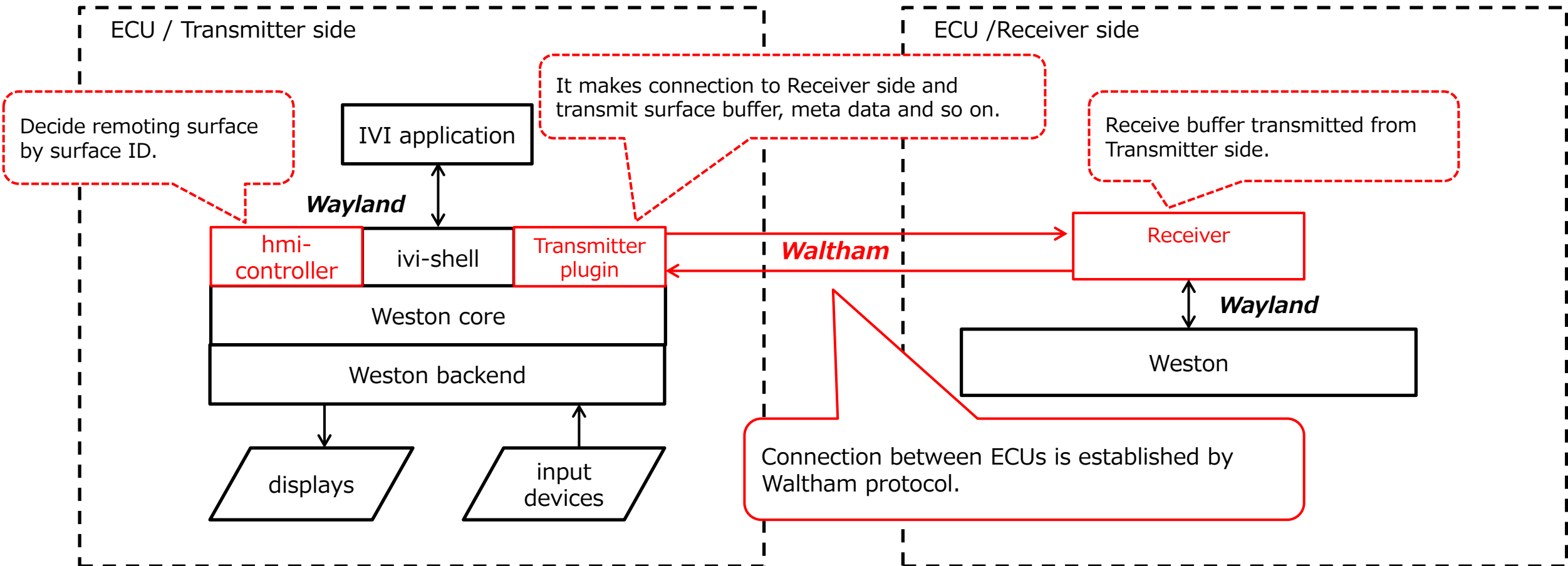
Composited rear view camera surface

- **Waltham enable surface sharing between multi ECU**
 - ◆ Waltham is IPC library of Wayland
 - ◆ <https://github.com/waltham/waltham>
- **Developed by Wayland community** 
 - ◆ Many developers contribute this community and upstream patches
 - ◆ <https://wayland.freedesktop.org/>
 - ◆ Automotive : DENSO, BOSCH and other companies are using
 - ◆ DENSO and BOSCH are driving Waltham project with collabora

- **No special implementation to application needed for remoting**
 - ◆ Weston module “hmi-controller” decide share surface or not by surface ID
 - hmi-controller is weston standard module
- **Designed to adopt various structure of systems**
 - ◆ The physical layer can be suit for structure of system
 - Examples
 - ◆ Between remote ECUs : MOST, Ethernet AVB
 - ◆ Between guests OSs on VM : Shared memory
- **Waltham would be a standard protocol for graphics sharing**

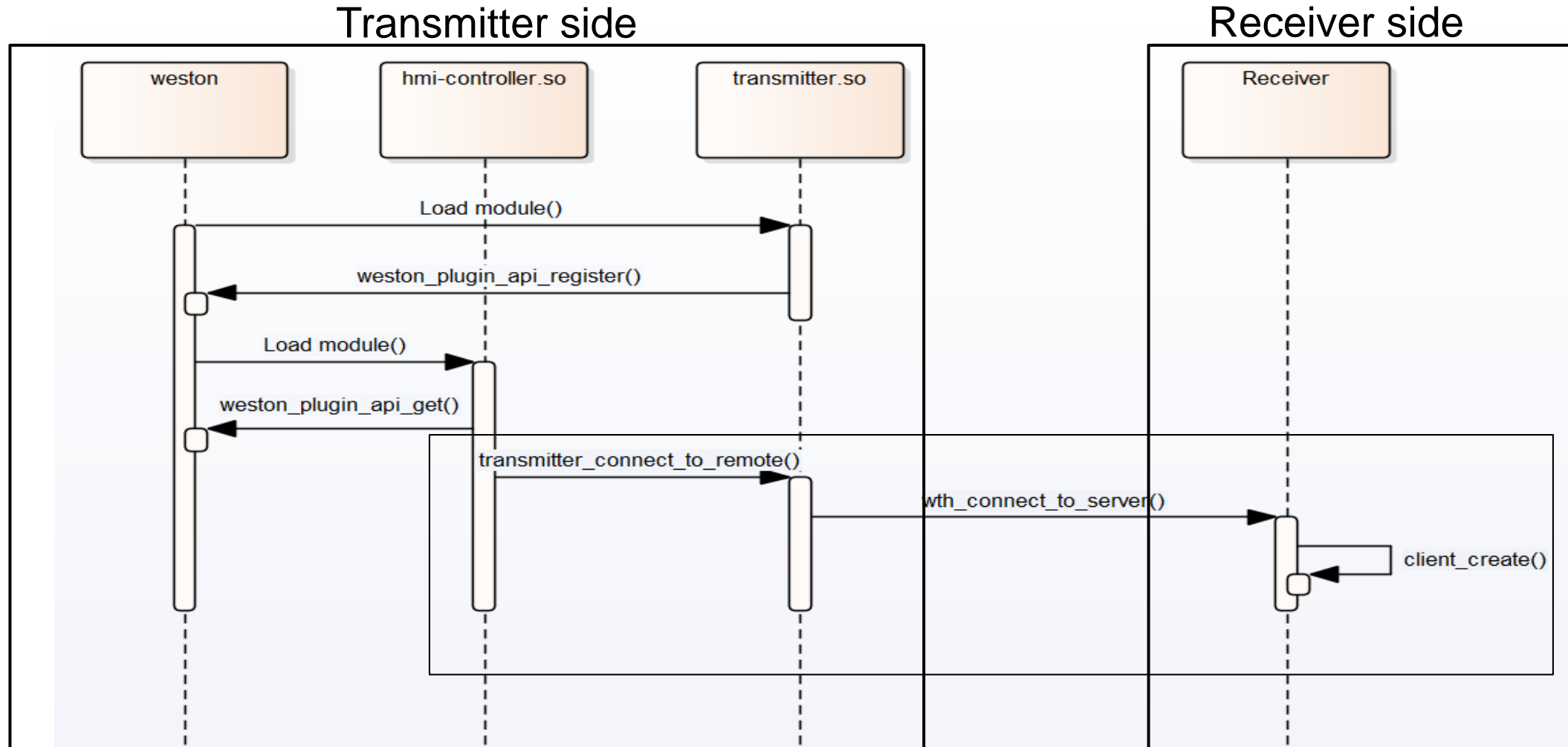
Surface sharing architecture of Waltham

- hmi-controller : Handle HMI layout and remoting surface
- Transmitter plugin : Handle connection and transmit surface

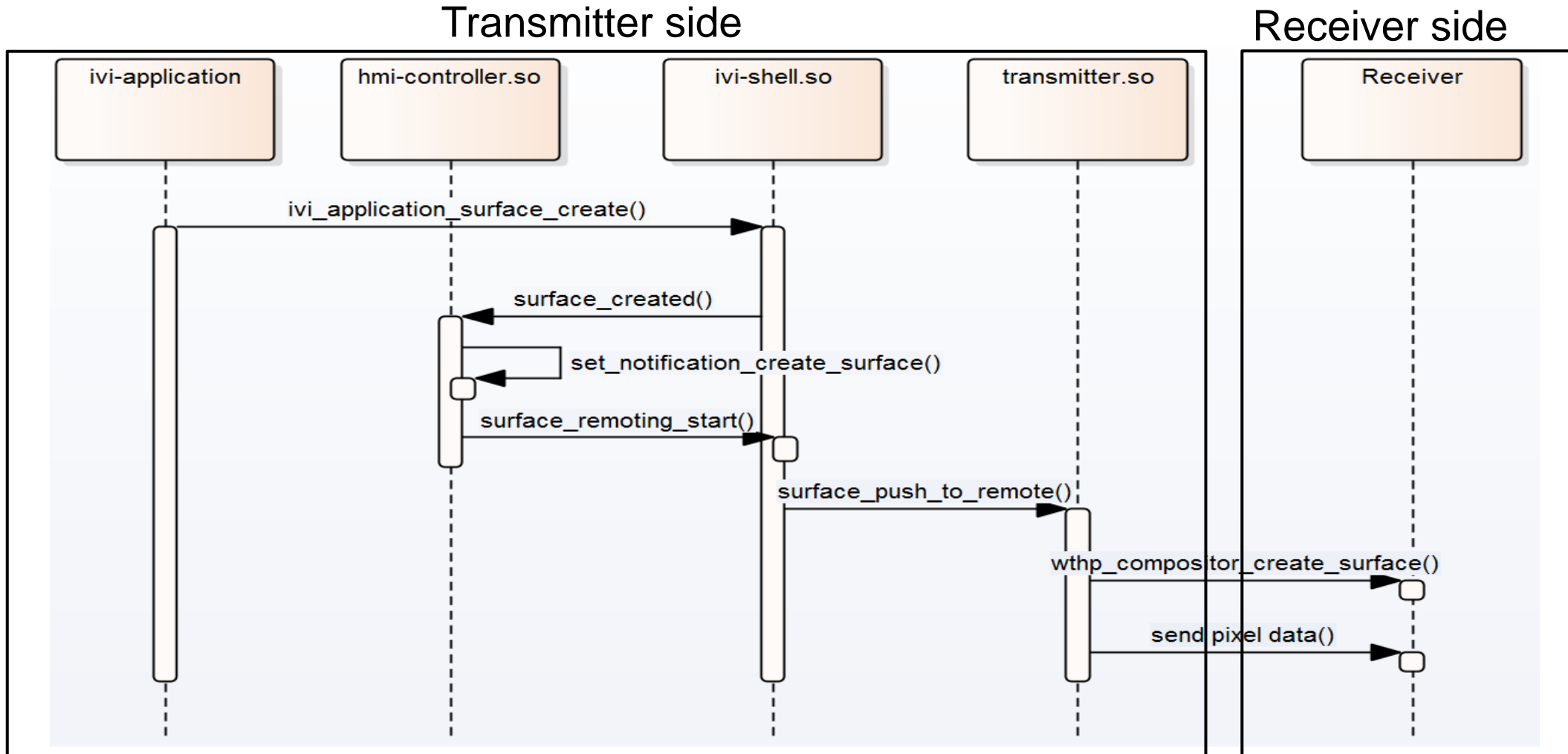


Initialization Weston with Waltham

- The hmi-controller needs to start connection processing
- Receiver waits from transmitter connection



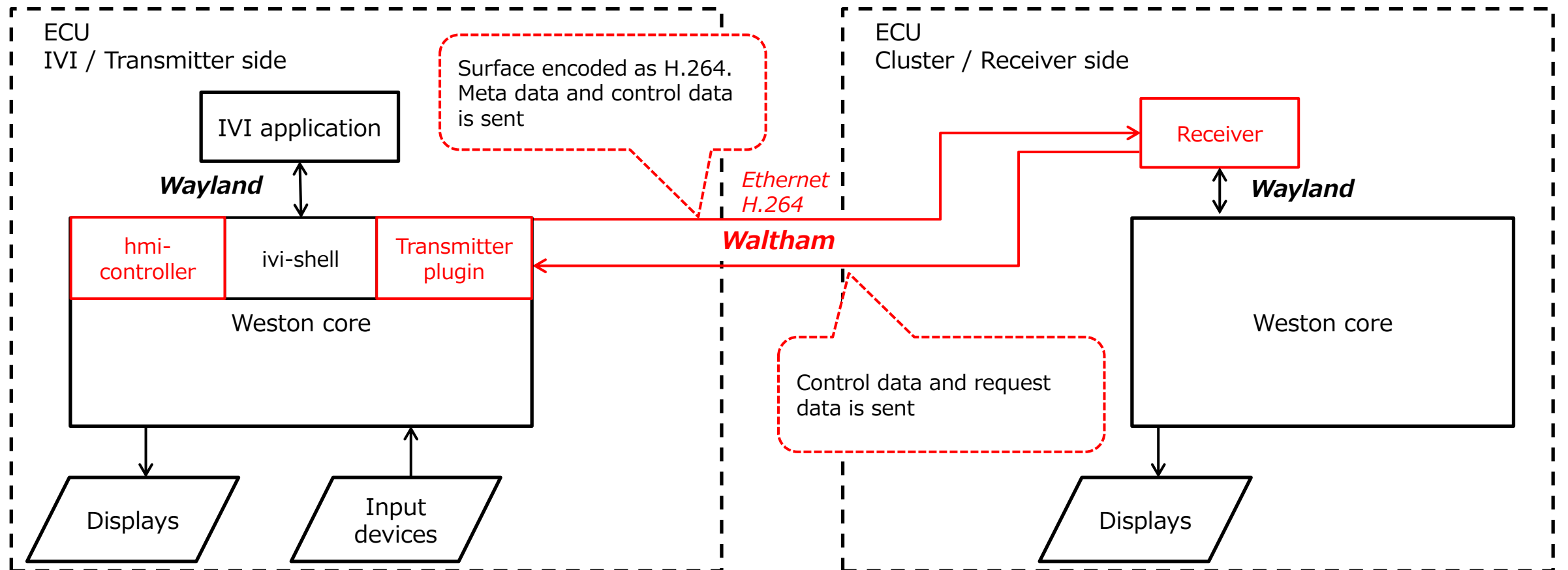
- The ivi-application does not need to do for remoting
 - ◆ The hmi-controller decide remoting surface by surface ID



Use case : IVI shares its surface with cluster on Multi ECU system

- IVI shares its surface of navi, cover art and so on, cluster gets IVI's surface and composite it to cluster view

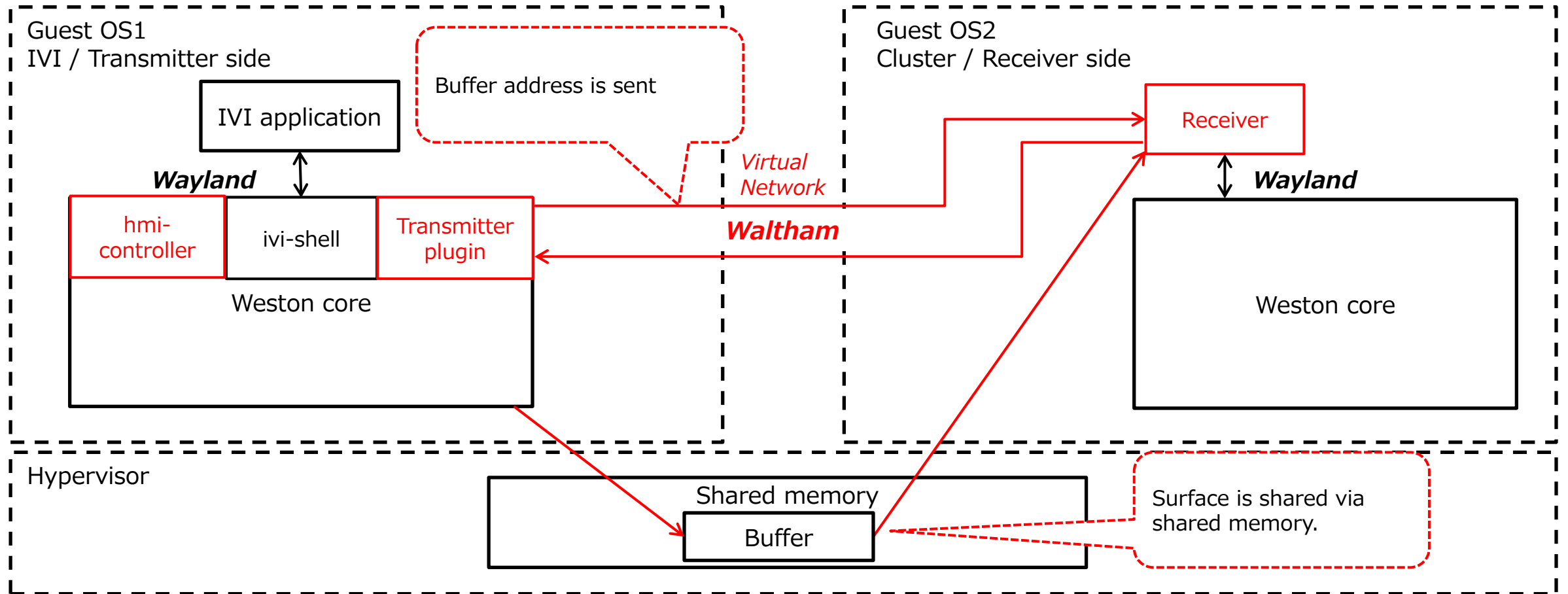
- Transmitter and Receiver can synchronize
- Receiver can request the kind of surface



Use case : IVI shares its surface with cluster on Virtualization system

- IVI shares its surface of navi, cover art and so on, cluster gets IVI's surface and composite it to cluster view

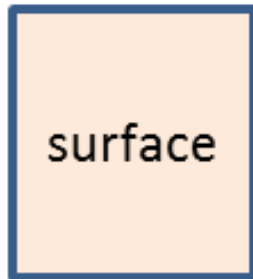
-Buffer address is shared to receiver side OS



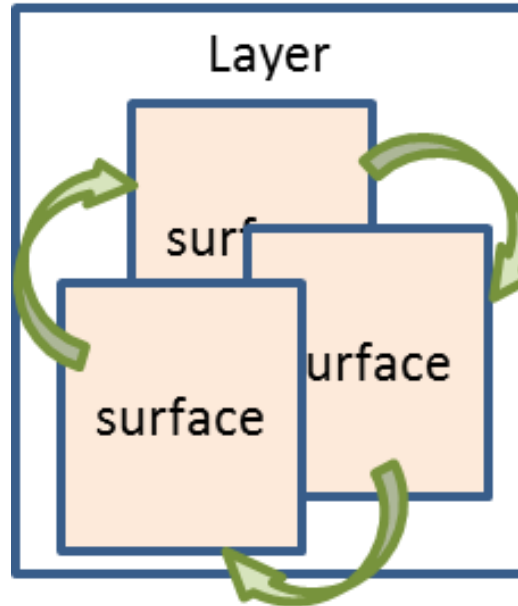
- **Graphics sharing is needed for future cockpit system**
- **AGL cluster demo : Next step is implementation of protocol**
- **Waltham would be the best solution for graphics sharing**
 - ◆ Standard protocol for sharing
 - ◆ Wayland application for single ECU is available as it is
 - ◆ Suit for various kind of systems

You can join Waltham project !

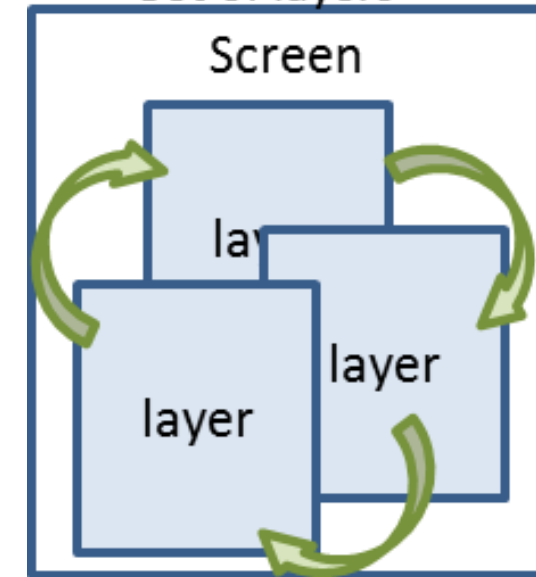
Surface
Smallest parts of screen



Layer
Set of surfaces



Screen
Set of layers



Interface	Waltham original	Detail
wth_display		In wayland protocol it is wl_display
wthp_registry		The Group of global objects. In wayland protocol it is wl_registry
wthp_callback		The callback interface. In wayland protocol it is wl_callback
wthp_compositor		The compositor. In wayland protocol it is wl_compositor
wthp_blob_factory	○	It creates wthp_buffer. Wthp_buffer is used for row pixel transfer
wthp_buffer		The buffer. In wayland protocol it is wl_buffer
wthp_surface		The surface. In wayland protocol it is wl_surface
wthp_seat		The group of input devices. In wayland protocol it is wl_seat
wthp_pointer		The pointer device. In wayland protocol it is wl_pointer
wthp_keyboard		The keyboard device. In wayland protocol it is wl_keyboard
wthp_touch		The touch screen device. In wayland protocol it is wl_touch
wthp_output		The Compositor output region. In wayland protocol it is wl_output
wthp_region		The region interface. In wayland protocol it is wl_region

- **Reciever application should be implemented with the interface**
- **ivi-application does not use them**
 - ◆ Transmitter absorbs the interface difference
- **Easy to add interface by edit command.xml**