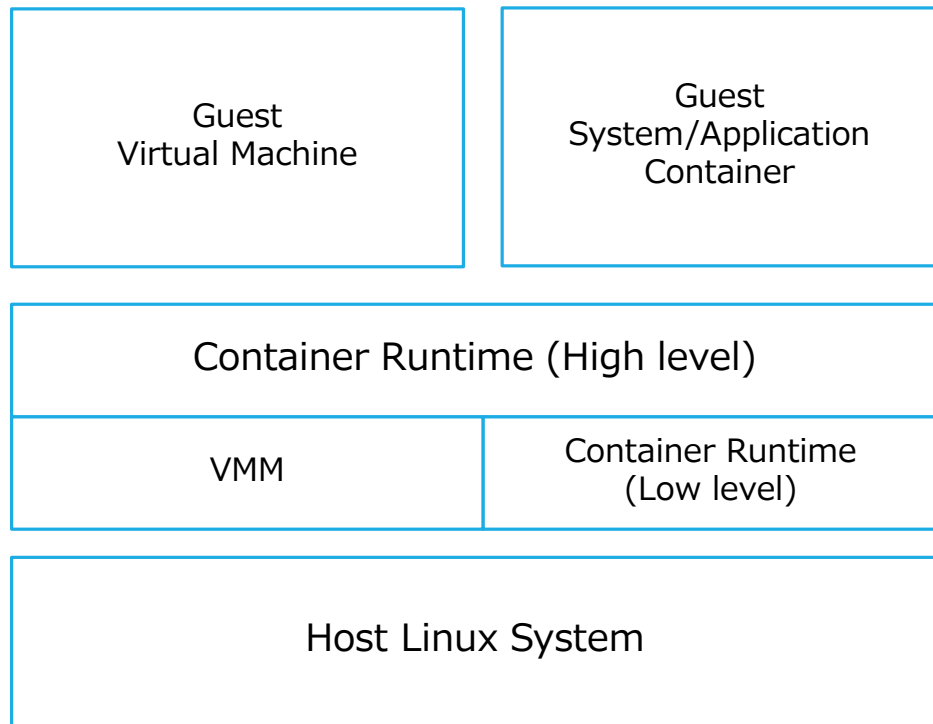# AGL UCB Containers

NOT ONLY FOR INSTRUMENT CLUSTER

Virtual Face to Face Technical Workshop @ Dec 8-9, 2020
Instrument Cluster EG
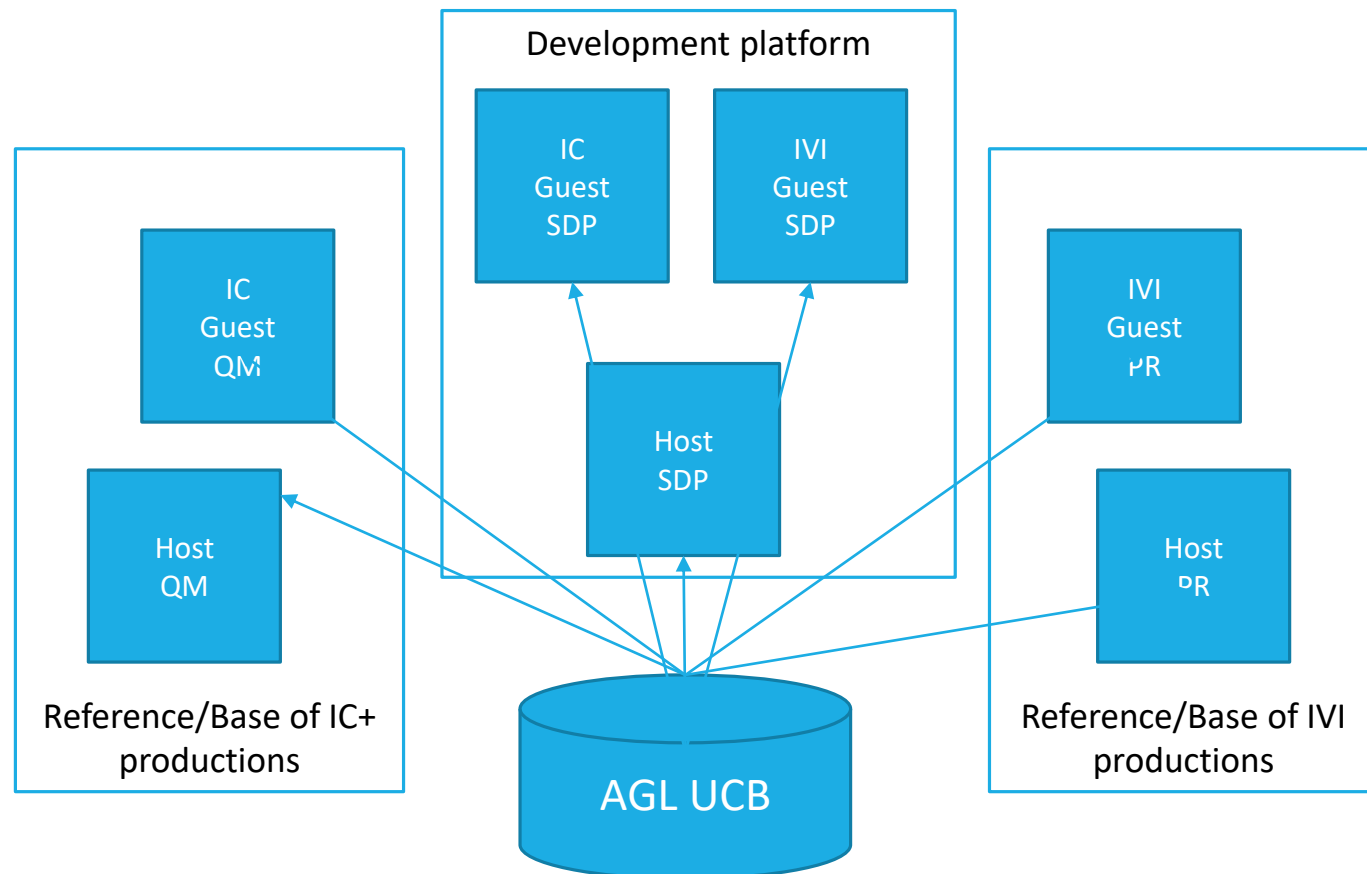
# Basic Concept

## CR can support both containers and VMs

| | |
|---|---|
| Guest Virtual Machine | Guest System/Application Container |

| | |
|---|---|
| Container Runtime (High level) | |
| VMM | Container Runtime (Low level) |

Host Linux System

Choice of container or VM depends on the requirements of partitioning

## Multiple guest/host images from single UCB

Development platform

| IC Guest SDP | IVI Guest SDP |
|---|---|

IC Guest QM

Host SDP

IVI Guest PR

Host QM

Host PR

Reference/Base of IC+ productions
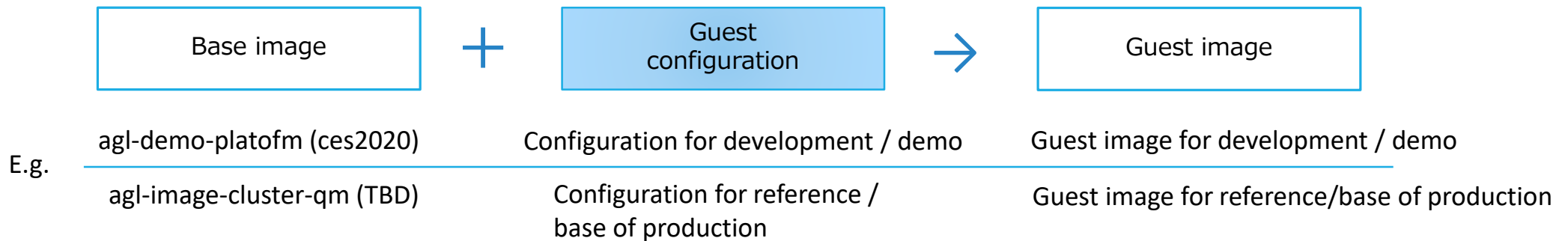
AGL UCB

Reference/Base of IVI productions

Single host/guest can't cover all uses because the requirements are too different

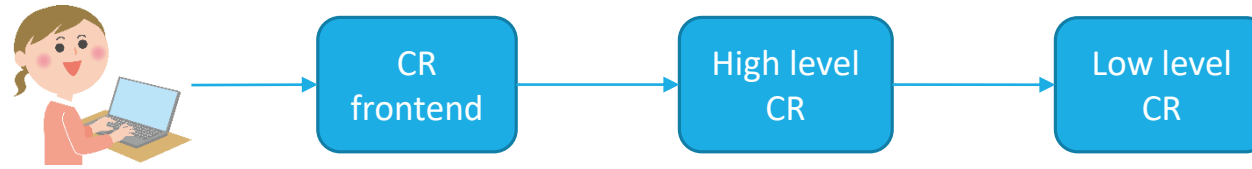# The basic concept of building multiple type of images

## Host

| Base image | + | Container Runtime | → | Host image |
|---|---|---|---|---|

E.g.

| agl-image-weston | lxc+lxc launcher (ces2020) | Host image for development / demo |
|---|---|---|
| agl-image-boot-qm (TBD) | lxc + cmgr-qm (TBD) | Host image for reference/base of production |

## Guest

| Base image | + | Guest configuration | → | Guest image |
|---|---|---|---|---|

E.g.

| agl-demo-platofm (ces2020) | Configuration for development / demo | Guest image for development / demo |
|---|---|---|
| agl-image-cluster-qm (TBD) | Configuration for reference / base of production | Guest image for reference/base of production |

# Container Runtime

## Generic

| | | |
|---|---|---|
| CR frontend | High level CR | Low level CR |

| LXD/LXC | | |
|---|---|---|
| LXD | | LXC |

| Docker | | |
|---|---|---|
| dockerd | containerd | runC |

## Embedded / Automotive
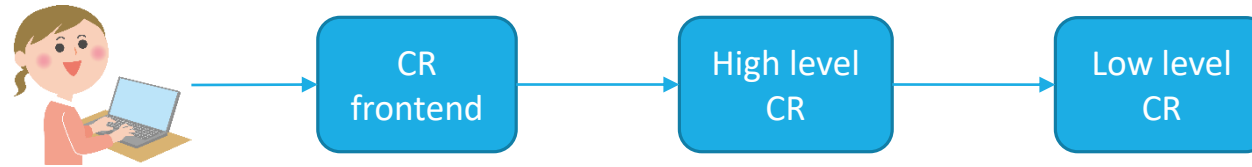
**For development** — CR frontend → High level CR → Low level CR — **Interactive**

**For production embedded** — ~~CR frontend~~ → High level CR → Low level CR — **Stand-alone**
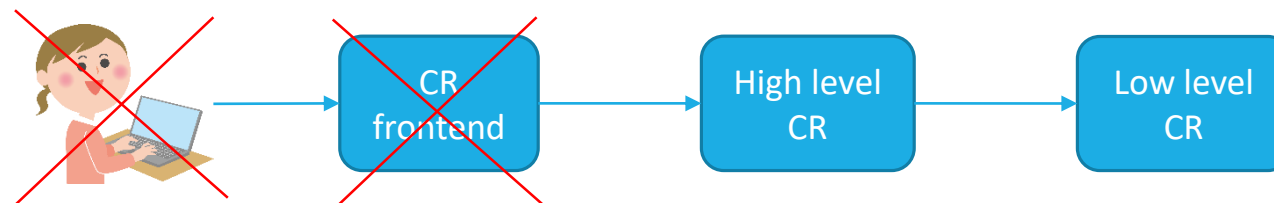
# Strategy of Integration

1st CR of AGL is LXC                                    because small and simple
1st target of integration is SDP/PoC/Demo              because much of the AGL UCB is reusable

| | Host Image | Guest Image | Guest Configuration | High level CR | Low Level CR |
|---|---|---|---|---|---|
| SDP/PoC/Demo<br><br>Prototypes already exist as CES2020 demo | Reuse<br><br>agl-image-weston<br>agl-image-drm (TBD) | Reuse<br><br>agl-demo-platform<br>agl-cluster-demo-platform | New<br><br>Yocto integration with AGL features for containers | Reuse & New<br><br>For Demo, lxc-launcher (Weston/ivi-shell on host required)<br><br>For Development, DRM Lease Mng + LXD? (TBD) | New<br><br>LXC |
| Reference of Prodcution | New<br><br>Build up from minimal, QM required | New<br><br>For IC, build up from minimal, QM required<br><br>For IVI, TBD | AGL features for container (NEW)<br>*.bbappend(NEW) | New<br><br>TBD | New<br><br>LXC |

# Detail of prototypes, CES2020 demo (container ver.)

## Graphics

Nested wayland compositor



**IC**
- receiver
- weston
- wayland-0

**IVI**
- tbtnavi
- homescreen
- weston
- wayland-1
- wayland-0

**Host**
- Weston (ivi-shell)
- HDMI-A-2
- remote-1
- HDMI-A-1
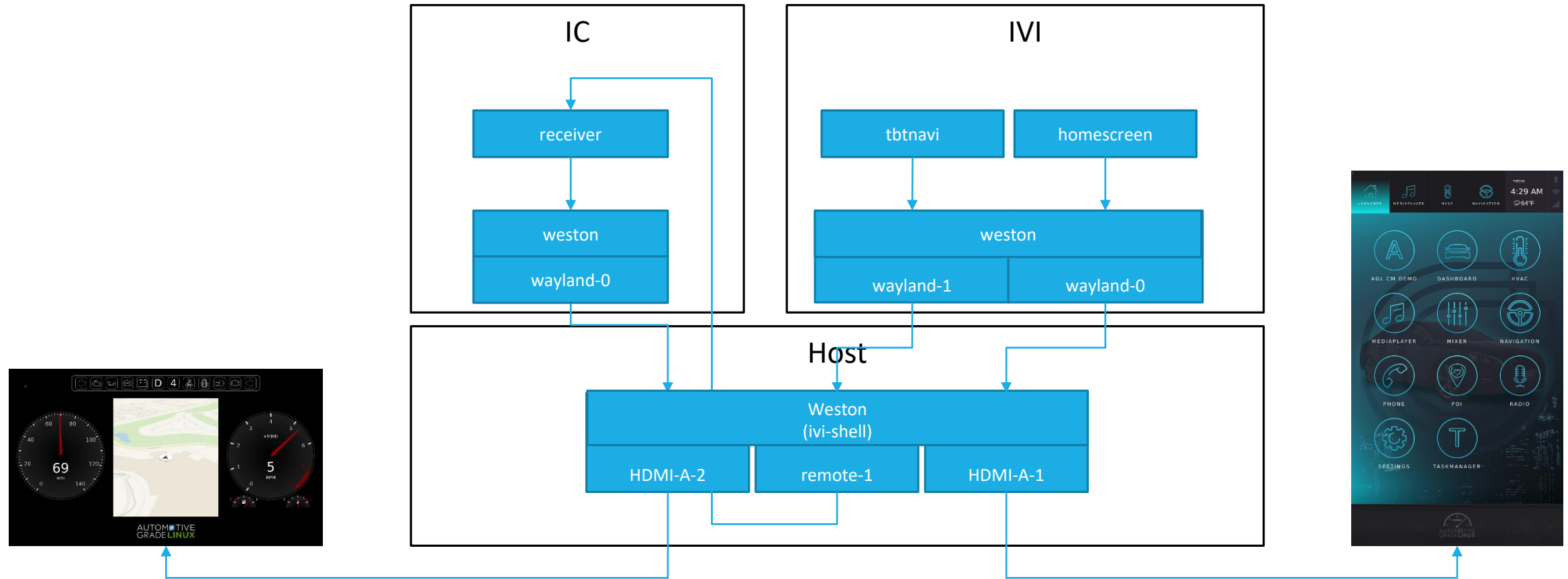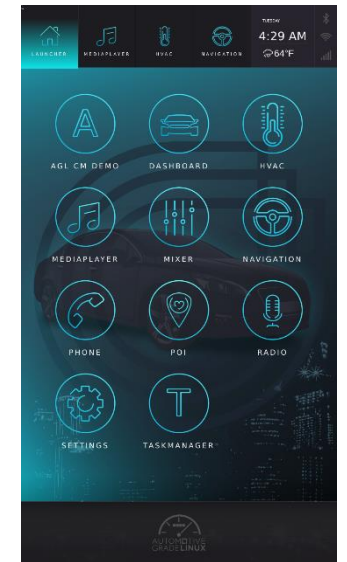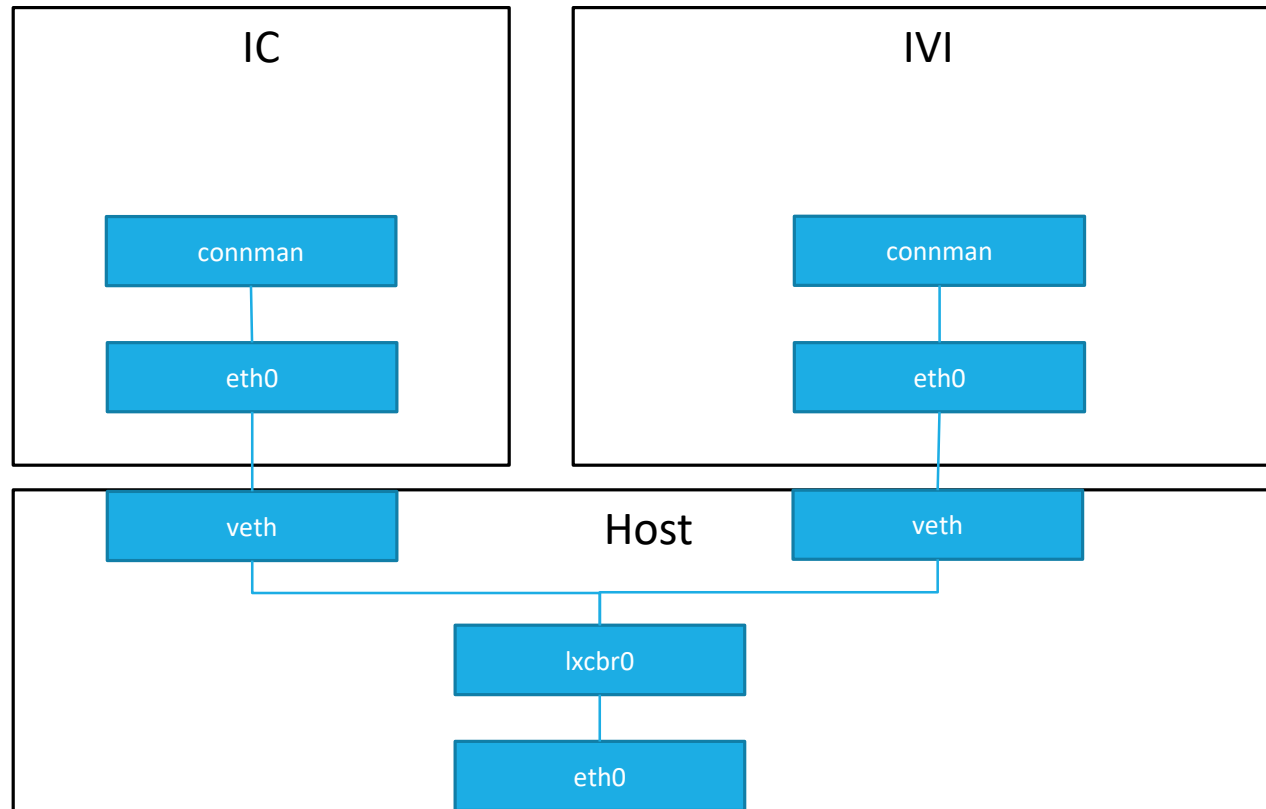
# Detail of prototypes, CES2020 demo (container ver.)

## Network

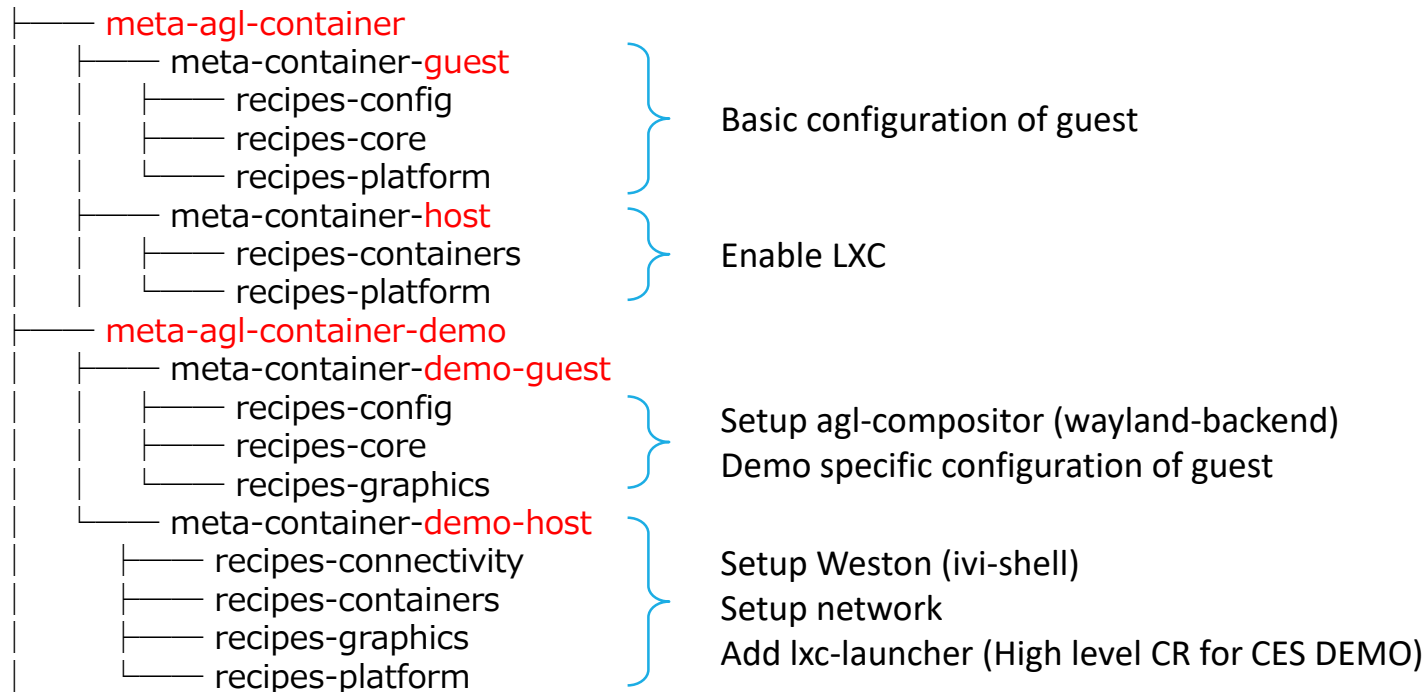Lxc-net: lxcbr0 (host) + veth (guest)

# meta layers

## Porting to master (WIP: Plan to merge KK, meta-agl-devel)

meta-agl-devel

```
├──── meta-agl-container
│     ├──── meta-container-guest
│     │     ├──── recipes-config
│     │     ├──── recipes-core          ⎫  Basic configuration of guest
│     │     └──── recipes-platform      ⎭
│     ├──── meta-container-host
│     │     ├──── recipes-containers    ⎫  Enable LXC
│     │     └──── recipes-platform      ⎭
├──── meta-agl-container-demo
│     ├──── meta-container-demo-guest
│     │     ├──── recipes-config        ⎫  Setup agl-compositor (wayland-backend)
│     │     ├──── recipes-core
│     │     └──── recipes-graphics      ⎭  Demo specific configuration of guest
│     └──── meta-container-demo-host
│           ├──── recipes-connectivity  ⎫  Setup Weston (ivi-shell)
│           ├──── recipes-containers
│           ├──── recipes-graphics         Setup network
│           └──── recipes-platform      ⎭  Add lxc-launcher (High level CR for CES DEMO)
```

# AGL features

## Porting to master (WIP: plan to merge KK, meta-agl-devel)

[meta-agl-devel]

agl-container
agl-container-guest:( agl-container )
agl-container-host :( agl-container  agl-profile-graphical )

Basic configuration

agl-lxc

Enable LXC

agl-container-demo
agl-container-demo-guest :( agl-container-guest  agl-container-demo )
agl-container-demo-host :( agl-container-host agl-container-demo  agl-lxc agl-weston-remoting)

Setup agl-compositor (wayland-backend)
Setup Weston (ivi-shell)
Setup network
Demo specific configuration of guest
Add lxc-launcher

agl-container-demo-ic :( agl-cluster-demo agl-lxc agl-container-demo-guest)
agl-container-demo-ivi :( agl-demo agl-cluster-demo-support  agl-weston-remoting agl-lxc agl-container-demo-guest)

ic/ivi specific
demo features

# Issues/Improvement required

- Weston on host is not always start automatically
  - Frequently fail to open tty7 at staring Weston when boot system

- Manual operation is required (not out-of-box)
  - ~~Need bitbake 3times if try to build CES2020 demo set (host, ivi and ic image are necessary to run)~~
  - ~~Burn 3 images into single rootfs image~~
    - ~~Opt1) WIC/Sdcard with 3 partitions~~
    - ~~Opt2) Single partitions (host), extract ivi/ic tarball on it manually~~
  - [Update after Virtual F2F]
    - Yocto's container image and multiconfig are suggested (Thank you, Jan-Simon and Scott!)
      - Building Containers with OpenEmbedded by Scott
    - Begun to consider adoption of them
      - Multiconfig: need to align aglsetup.sh (which generates local.conf)

- QEMU + 2 monitor outputs (limitation)
  - It will work when using GTK frontend without GPU acceleration (virgl)
    - SDL frontend not show 2$^{nd}$ display
    - GTK frontend not work with virgl and 2 monitors (max_outputs=2 and gl=on)

# What's next?

## For IC production embedded)

- DRM lease manager
  - Nested wayland compositor no longer be required
  - Waltham transmitter  (it requires drm-backend on guest, before DRM lease manager, it cannot)
- Container runtime configuration
  - Compatible with OCI?
- Guest management both container and virtualization
  - E.g. LXD can manage VM (QEMU/KVM) as well as containers

## For development / PoC / Demo platform or other purpose

- Another CR
  - For more general purpose, not only for IC
    - E.g. To support Apps development of cross-platform framework like Flutter / React Native / Unity
  - Docker for embedded / automotive?
  - K8s for embedded / automotive?
  - Other lightwegith containers?