

New AGL Layer Structure

Existing Layer Structure

			h3ulcb+agl-demo	
bsp	meta-agl-refhw		-	support for the Renesas R-Car H3 based AGL reference
	...		-	
	meta-raspberrypi		-	Yocto BSP layer for the Raspberry Pi boards
	meta-rcar		-	support for Renesas Electronics's ADAS platforms
	meta-renesas	meta-rcar-gen3	x	support for Renesas Electronics's platforms
	...		-	
external	alexa-auto-sdk		-	
	meta-clang		-	
	meta-codechecker		-	
	meta-gplv2		-	
	meta-iot-cloud		x	
	meta-openembedded	meta-oe	x	
		meta-python	x	
		meta-networking	x	
		meta-filesystems	x	
		meta-perl	x	
		...	-	
	meta-python2		x	
	meta-qt5		x	
	meta-security		x	
	meta-spdxscanner		-	
	meta-updater		-	
	meta-updater-qemux86-64		-	
	meta-updater-raspberrypi		-	
	meta-virtualization		-	
	poky	meta	x	
		meta-poky	x	
		...	-	
meta-agl	meta-agl-bsp		x	agl specific board support
	meta-agl-core		x	agl core software stack layer
	meta-agl-core-test		x	
	meta-agl-ic		-	common software stack layer for ic eg
	meta-agl-ivi		-	common software stack layer for ivi-pr eg
	meta-app-framework		x	SMACK based agl app framework layer
	meta-netboot		-	
	meta-pipewire		x	layer for pipewire
meta-agl-demo			x	agl demo software layer
meta-agl-devel	meta-agl-drm-lease		-	
	meta-agl-jailhouse		-	
	meta-egvirt		-	
	meta-oem-production-readiness		-	

	meta-speech-framework	-	
--	-----------------------	---	--

Current IC Layer Structure

		h3ulcb host	h3ulcb Cluster- guest	h3ulcb Cluster Standalone (TODO:)	Require QM	
container- bsp	meta-rcar- gen3-host	x	-	-	yes	host extension for meta-rcar-gen3
	meta-rcar- gen3-guest	-	x	-	yes	guest extension for meta-rcar-gen3
meta- renesas	meta-rcar-gen3	x	x	x	no	support for Renesas Electronics's platforms
meta-agl	meta-agl-bsp	x	x	x	no	agl specific board support *. This layer is shared by both agl ivi and agl cluster. When this layer modify to update for ivi use case, it propagate to agl cluster. This point is serious issue, but I don't have good idea without "agl cluster doesn't use meta-agl and meta-agl-bsp".
	meta-agl-core	x	x	x	yes	agl core software stack layer Reduced from existing meta-agl-core
	meta-agl-core- test	-	-	-	no	
	meta-agl-host	x	-	-	yes	Software stack layer for AGL containerization host
	meta-agl-ic	-	x	x	yes	common software stack layer for ic eg Software stack layer for Instrument Cluster (without IVI based existing Instrument Cluster)
	meta-agl-ivi	-	-	-	no	common software stack layer for ivi pr eg Software stack layer for IVI (with IVI based existing Instrument Cluster)
	meta-app- framework	-	-	-	no	
	meta-netboot	-	-	-	no	
	meta-pipewire	x	-	-	no	More detail, depends on audio management work.
meta-agl- devel	meta-agl-drm- lease	x	x	-	yes	
	meta-agl- jailhouse	-	-	-	no	
	meta-egvirt	-	-	-	no	
	meta-oem- production- readiness	-	-	-	no	
	meta-speech- framework	-	-	-	no	
meta-agl-qm		x	x	x	yes	Including assessment data for a part of poky and meta-oe.
meta-agl-demo-cluster		-	x	x	no	Demo layer for Instrument Cluster (without IVI based existing Instrument Cluster)
meta-agl-demo		-	-	-	no	agl demo software layer
poky	meta	x	x	x	no	
	meta-poky	x	x	x	no	
	meta-poky-bsp	x	x	x	no	Will be remove
	...	-	-	-	no	
meta- openemb edded	meta-oe	x	x	x	no	
	meta-python	x	x	x	no	
	...	-	-	-	no	

*Note. The meta-agl-qm provide OSS assessment result for poky, meta-openembedded and other external layers. It does not cover to meta-agl-core, meta-agl-host and meta-agl-ic.

AGL IVI specific OSS (it's including ivi-pr) and configuration change shall move into the meta-agl-ivi.

Current IC layer structure is not including meta-codechecker, meta-clang and meta-spdxcanner. These layer will be adding.

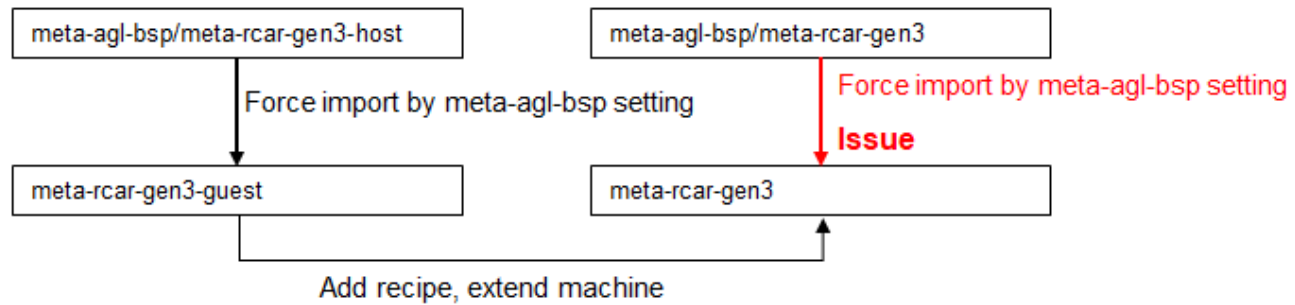
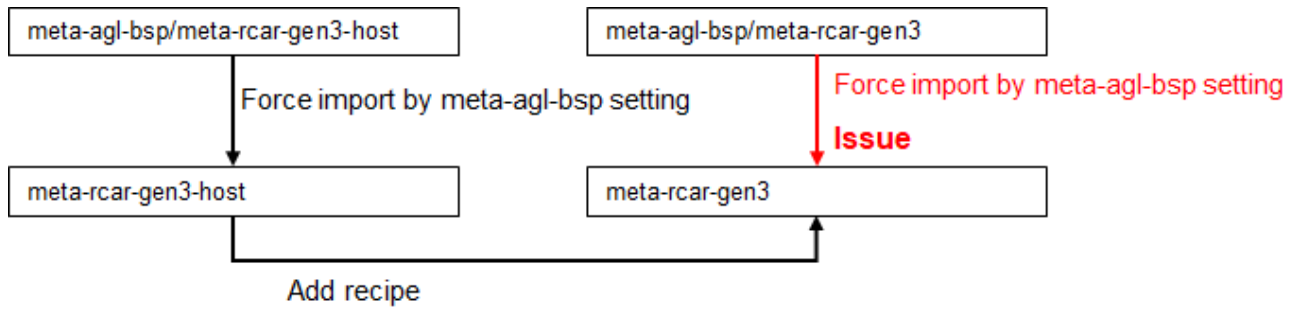
Current IC layer structure don't have 'bsp' and 'external' directory. It will change.

Currently, should separate IC demo and IVI based demo. Because existing meta-agl-demo fully depend to meta-app-framework.

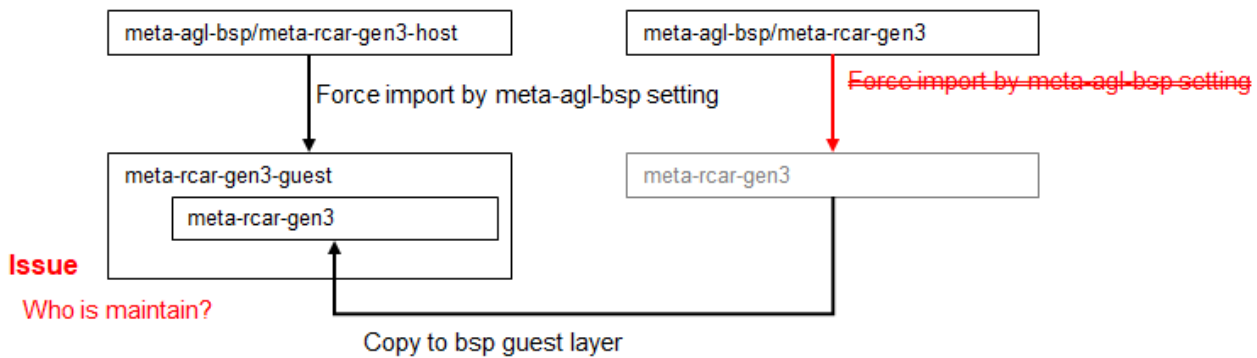
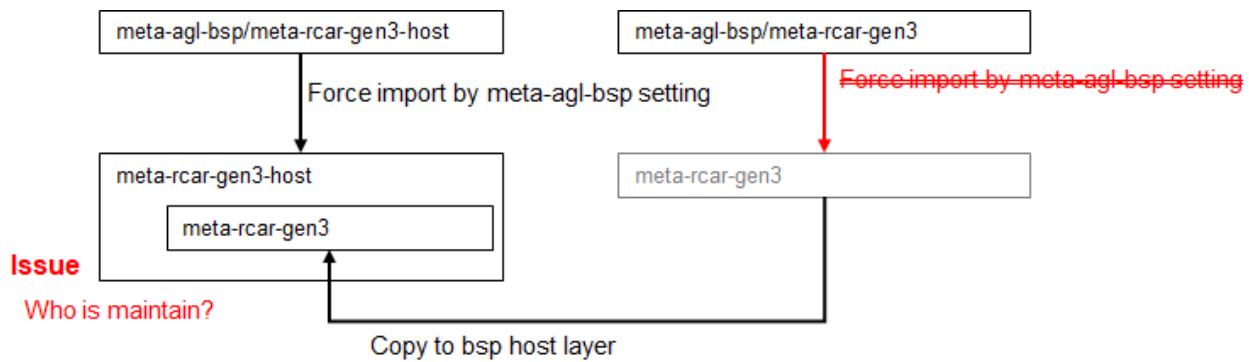
We will develop security mechanism for embedded containerized platform, but it is not current work. 1st step is a un-privilege gest. We can't import SMACK.

BSP Issue

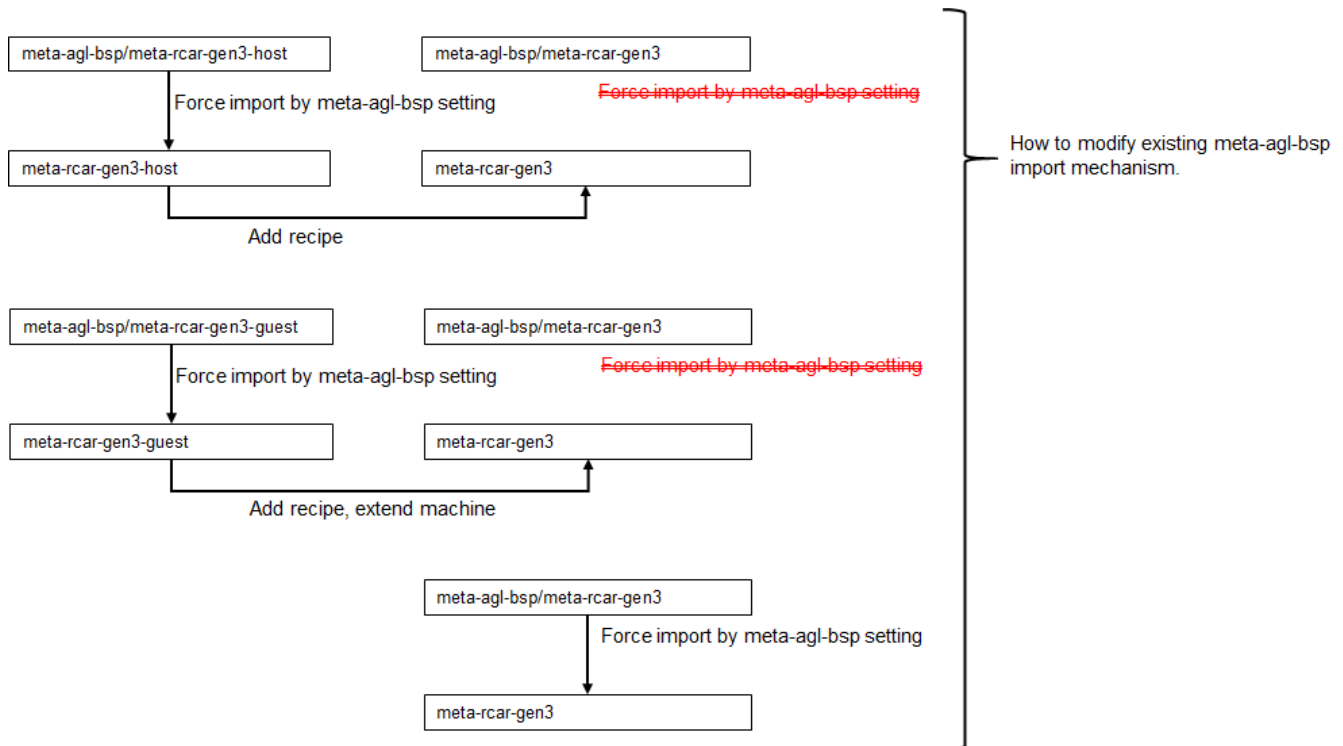
Issue for current AGL design.



Idea 1: Heavy maintenance



Idea 2: Best solution, but how to do?



Which recipe should move into meta-agl-ivi?

meta-agl\$ grep SUMMARY -R meta-agl-core/*

recipe	SUMMARY	Should be	Why
meta-agl-core/recipes-connectivity/connman-ncurses/connman-ncurses_git.bb	SUMMARY = "A simple ncurses interface for connman"	meta-agl-ivi	This OSS will assess?? May not.
meta-agl-core/recipes-graphics/wayland/weston-ini-conf.bb	SUMMARY = "Configuration file for the Weston and AGL Wayland compositors"	meta-agl-core	Not specific IVI
meta-agl-core/recipes-graphics/wayland/agl-shell-activator_git.bb	SUMMARY = "Application that switches/activates other application's windows "	TODO	No details
meta-agl-core/recipes-graphics/wayland/agl-compositor_git.bb	SUMMARY = "Reference Wayland compositor for AGL"	meta-agl-core	Not specific IVI
meta-agl-core/recipes-graphics/rba/librba_1.0.bb:	SUMMARY = "RBA"	meta-agl-core	Not specific IVI
meta-agl-core/recipes-graphics/rba/rba-config.bb	SUMMARY = "Rule Based Arbitrator Model required for RBA policy compositor."	meta-agl-core	
meta-agl-core/recipes-platform/images/agl-image-boot.bb	SUMMARY = "An AGL small image just capable of allowing a device to boot."	meta-agl-core	
meta-agl-core/recipes-platform/images/agl-image-core-autobuilder.bb	SUMMARY = "A very basic Wayland image with a terminal"	meta-agl-ivi	Depend to IVI specific package groups
meta-agl-core/recipes-platform/images/agl-image-weston.bb	SUMMARY = "A very basic Wayland image with a terminal"	meta-agl-ivi	Depend to IVI specific package groups
meta-agl-core/recipes-platform/images/agl-image-minimal-crosssdk.bb	SUMMARY = "Cross SDK of Full AGL Distribution for core profile"	meta-agl-core	
meta-agl-core/recipes-platform/images/agl-image-minimal.bb	SUMMARY = "An AGL small image just capable of allowing a device to boot."	meta-agl-core	
meta-agl-core/recipes-platform/packagegroups/packagegroup-agl-core-devel.bb	SUMMARY = "Provides a set of tools for development for AGL DISTRO"	meta-agl-core	
meta-agl-core/recipes-platform/packagegroups/packagegroup-agl-graphical-multimedia.bb	SUMMARY = "The software for application framework of AGL IVI profile"	meta-agl-ivi	Defined for IVI specific package groups. Not common.
meta-agl-core/recipes-platform/packagegroups/packagegroup-agl-profile-graphical.bb	SUMMARY = "The middleware for AGL IVI profile"	meta-agl-ivi	Defined for IVI specific package groups. Not common.
meta-agl-core/recipes-platform/packagegroups/packagegroup-agl-core-os-commonlibs.bb	SUMMARY = "The packages of middlewares for AGL IVI profile"	meta-agl-ivi	Defined for IVI specific package groups. Not common.
meta-agl-core/recipes-platform/packagegroups/packagegroup-agl-core-boot.bb	SUMMARY = "Minimal boot requirements"	meta-agl-core	
meta-agl-core/recipes-platform/packagegroups/packagegroup-agl-core-security.bb	SUMMARY = "The packages of middlewares for AGL IVI profile"	meta-agl-ivi	Defined for IVI specific package groups. Not common.

```
find meta-agl-core -name *.bb*
meta-agl-core/recipes-kernel/linux/kernel-devsrc.bbappend
meta-agl-core/recipes-platform/images/agl-image-boot.bb
meta-agl-core/recipes-platform/images/agl-image-core-autobuilder.bb
meta-agl-core/recipes-platform/images/agl-image-weston.bb
meta-agl-core/recipes-platform/images/agl-image-minimal-crosssdk.bb
meta-agl-core/recipes-platform/images/agl-image-minimal.bb
meta-agl-core/recipes-platform/packagegroups/packagegroup-agl-image-boot.bb
meta-agl-core/recipes-platform/packagegroups/packagegroup-agl-image-minimal.bb
meta-agl-core/recipes-platform/packagegroups/packagegroup-agl-core-devel.bb
meta-agl-core/recipes-platform/packagegroups/packagegroup-agl-graphical-multimedia.bb
meta-agl-core/recipes-platform/packagegroups/packagegroup-agl-profile-graphical.bb
meta-agl-core/recipes-platform/packagegroups/packagegroup-agl-graphical-weston.bb
meta-agl-core/recipes-platform/packagegroups/packagegroup-agl-core-connectivity.bb
meta-agl-core/recipes-platform/packagegroups/packagegroup-agl-core-os-commonlibs.bb
meta-agl-core/recipes-platform/packagegroups/packagegroup-agl-core-boot.bb
meta-agl-core/recipes-platform/packagegroups/packagegroup-agl-core-security.bb
meta-agl-core/classes/image_types_wic_workaround.bbclass
meta-agl-core/classes/sanity-meta-agl-core.bbclass
meta-agl-core/classes/cloc.bbclass
meta-agl-core/recipes-core/systemd/systemd-conf_%.bbappend
meta-agl-core/recipes-core/systemd/systemd_%.bbappend
meta-agl-core/recipes-devtools/gdb/gdb_%.bbappend
meta-agl-core/recipes-devtools/cmake/cmake_%.bbappend
meta-agl-core/recipes-graphics/wayland/walther_%.bbappend
meta-agl-core/recipes-graphics/wayland/walther_git.bb
meta-agl-core/recipes-graphics/wayland/weston-init-conf.bb
meta-agl-core/recipes-graphics/wayland/weston-init.bbappend
meta-agl-core/recipes-graphics/wayland/weston-init-conf.bbappend
meta-agl-core/recipes-graphics/wayland/agl-shell-activator_git.bb
meta-agl-core/recipes-graphics/wayland/walther-transmitter-plugin.bb
meta-agl-core/recipes-graphics/wayland/weston_8.0_%.bbappend
meta-agl-core/recipes-graphics/rba/librba_1.0.bb
meta-agl-core/recipes-connectivity/connman/connman_%.bbappend
meta-agl-core/recipes-connectivity/busybox/busybox_%.bbappend
meta-agl-core/recipes-support/ptest-runner/ptest-runner_2_%.bbappend
meta-agl-core/recipes-extended/gperf/gperf_%.bbappend
```

How to understand distribution structure?