Proposal to Layer Design of AGL Distribution

June 22, 2015
AGL Distribution Project
Written by Tadao Tanikawa, Panasonic

Contents

- The Layers of AGL Distribution
- Design of meta-agl/meta-agl-demo
- Relations of recipes and subsystems
- Specification of agl-image-minimal
- Specification of agl-image-ivi
- Specification of agl-image-demo
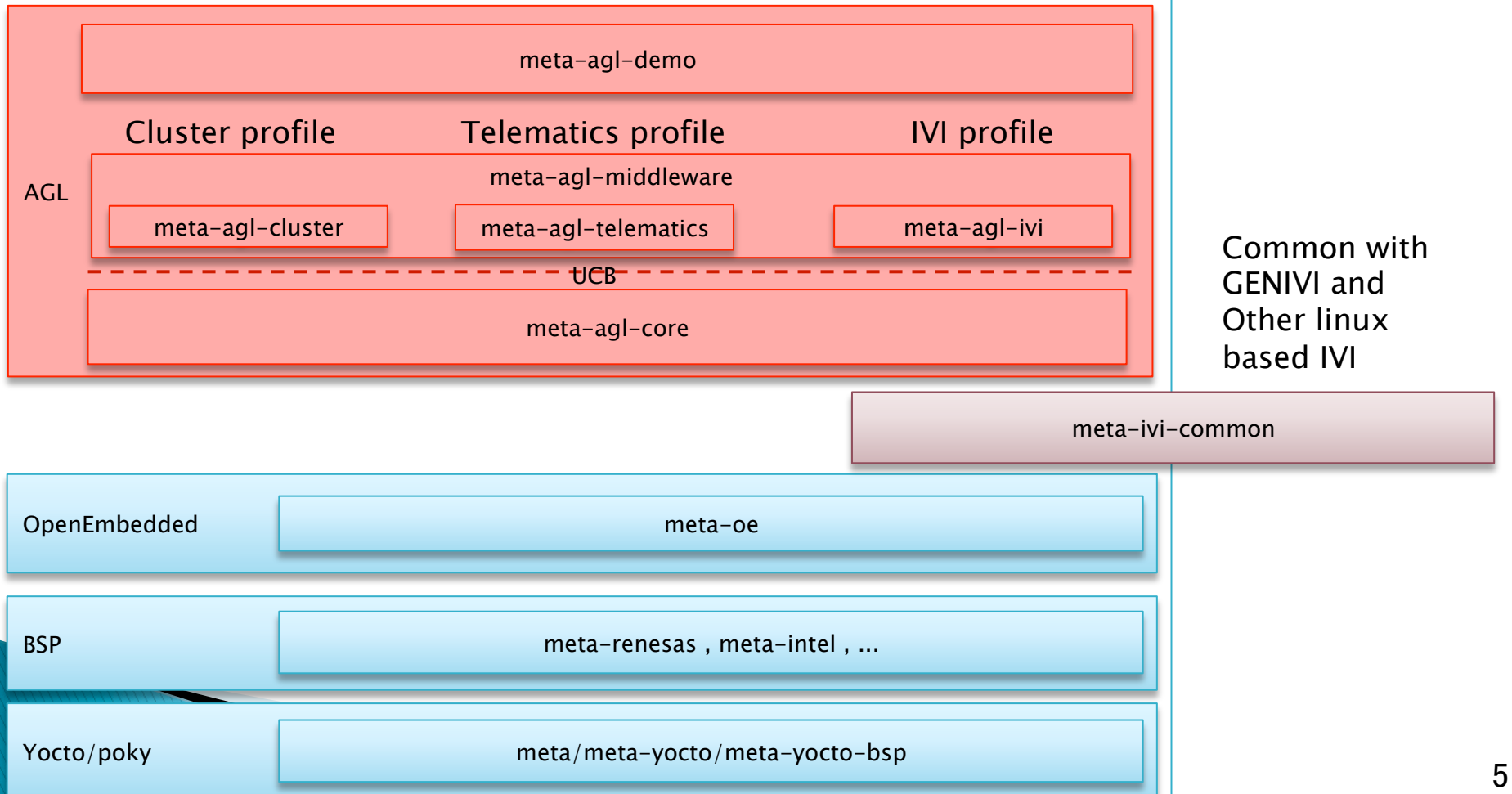- License

Reminder : 3 Phases of AGL Distro Project

▸ Phase 1 goals
  ◦ Create an AGL Distro
  ◦ Replacement for Tizen IVI, provide the same infrastructure that Tizen IVI provided
  ◦ Unify as much as possible AGL, Tizen IVI and GENIVI
  ◦ Design the layers such that the base distro can be used for IVI, Cluster, Telematics
  ◦ Create the recipes and layers
  ◦ Create test framework
  ◦ App framework and demo is out of scope for Phase 1
  ◦ Support ARM and x86 (Minnowboard or VTC 1010)
  ◦ Release phase 1 by end of August

▸ Phase 2 goals
  ◦ Identify release cadence and support going forward (bug fixing, security updates, frequency of releases)
  ◦ App framework(s)
  ◦ Demo applications (Home Screen, Media browser/player, Vehicle data, Settings)
  ◦ Option for Native vs HTML5 apps
  ◦ Support for QEMU or virtualized emulator?
  ◦ Release Phase 2 by end of 2015

▸ Phase 3 goals
  ◦ SDK
  ◦ Profiles for Cluster, HUD, Telematics

The Layers of AGL Distribution(1)

- [agl-discussions] Shared Yocto IVI Layer Strategy (Thread in AGL open mailing list)
  - "Andersson, Gunnar" <gunnar.x.andersson@volvocars.com> proposed 'Shared Yocto IVI Layer Strategy'
    - In the proposal, he describes 'meta-ivi-common' which contains all of IVI related software components.
  - "Streif, Rudolf" <rstreif@jaguarlandrover.com > also post in that thread,
    - an open source layer stack could look like
      - meta-agl or meta-genivi-demo (UI and distro layer)
      - meta-genivi-baseline
      - meta-ivi-common
      - meta-bsp (whatever BSP is needed)
      - meta-oe
    - A custom OEM or T1-specific stack could look like:
      - meta-oem-distro (OEM distro and UI)
      - meta-oem-middleware (OEM middleware components)
      - meta-genivi-baseline
      - meta-ivi-common
      - meta-bsp (whatever BSP is needed)
      - meta-oe
- To build AGL Distro v1.0 (AGL Distro Project Phase 1/Phase 2) up from Yocto/OE, We need common layer like that 'meta-ivi-common'.
  - Because we decided to import software from GENIVI and Tizen.

- Unfortunatly 'meta-ivi-common' is not available in our yocto layers yet although we need it now for Phase 1 release (2015/8E).

- In order to meet our schedule, we have to build our own layer which has the similar role as 'meta-ivi-common'.
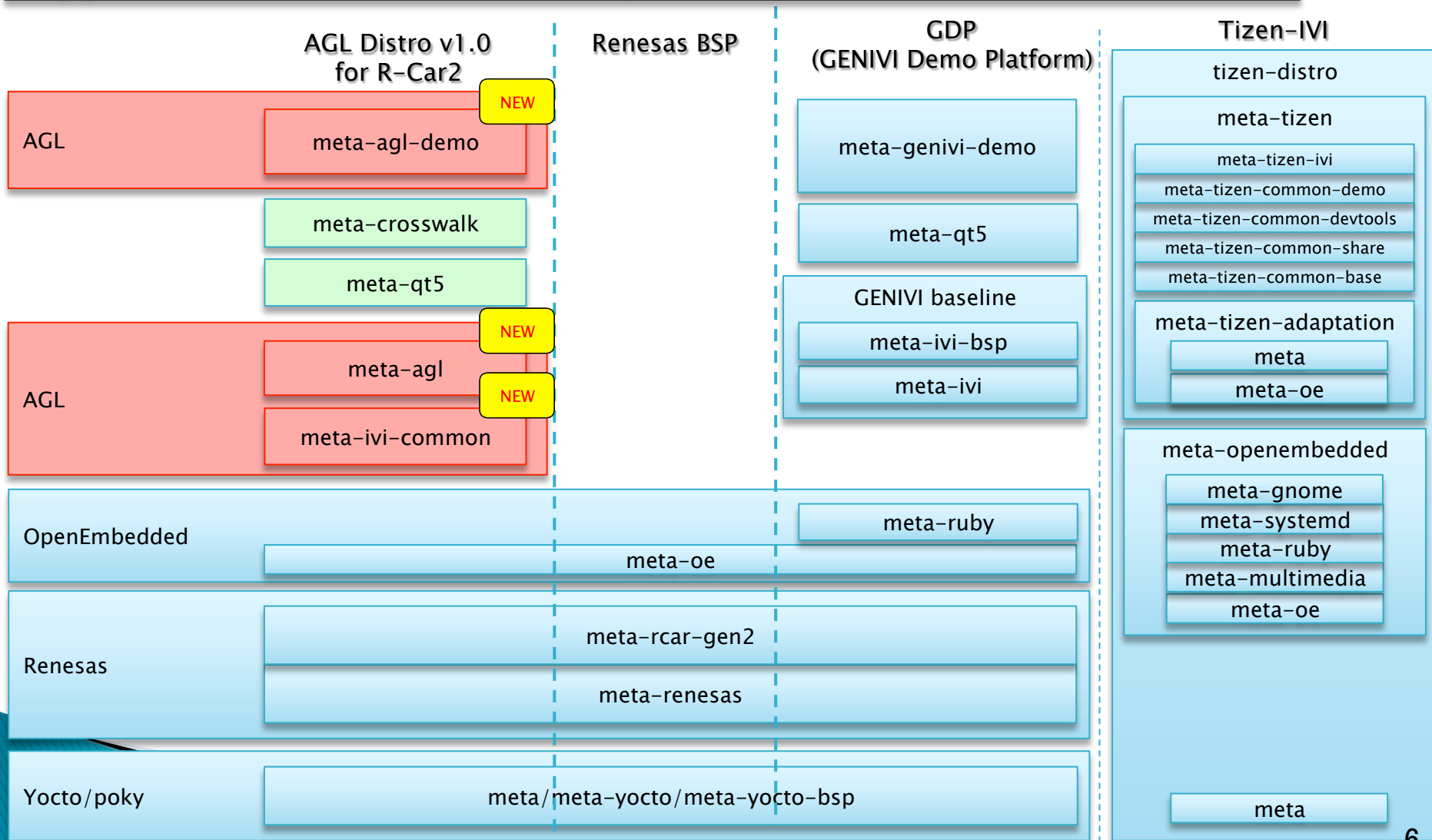 - In the future, 'meta-ivi-common' will come, we should use it.

The Layers of AGL Distribution(2)

Conceptual drawing of Phase 3 layers

AGL

meta-agl-demo

Cluster profile   Telematics profile   IVI profile

meta-agl-middleware

meta-agl-cluster   meta-agl-telematics   meta-agl-ivi

UCB

meta-agl-core

Common with GENIVI and Other linux based IVI

meta-ivi-common

OpenEmbedded	meta-oe
BSP	meta-renesas , meta-intel , ...
Yocto/poky	meta/meta-yocto/meta-yocto-bsp

The Layers of AGL Distribution (3)

Suggestion of Phase 1 layers and comparison to current GENIVI and Tizen IVI

	AGL Distro v1.0 for R-Car2	Renesas BSP	GDP (GENIVI Demo Platform)	Tizen-IVI

AGL Distro v1.0 for R-Car2:
- AGL: meta-agl-demo (NEW)
- meta-crosswalk
- meta-qt5
- AGL: meta-agl (NEW), meta-ivi-common (NEW)
- OpenEmbedded: meta-oe
- Renesas: meta-rcar-gen2, meta-renesas
- Yocto/poky: meta/meta-yocto/meta-yocto-bsp

GDP (GENIVI Demo Platform):
- meta-genivi-demo
- meta-qt5
- GENIVI baseline: meta-ivi-bsp, meta-ivi
- meta-ruby

Tizen-IVI:
- tizen-distro
- meta-tizen: meta-tizen-ivi, meta-tizen-common-demo, meta-tizen-common-devtools, meta-tizen-common-share, meta-tizen-common-base
- meta-tizen-adaptation: meta, meta-oe
- meta-openembedded: meta-gnome, meta-systemd, meta-ruby, meta-multimedia, meta-oe
- meta

Design of meta-agl/meta-agl-demo

- **meta-agl**
  - This layer is for minimal/baseline image of AGL Distribution
    - Phase 1: the minimal and baseline systems are defined
      - The minimal image to boot system (**images/agl-image-minimal**)
      - The baseline image includes platform and automotive services (**images/agl-image-ivi**)
        - This image is depended on **meta-ivi-common**
    - Phase 2 : QEMU/Virtualized emulator?
    - Phase 3 :
      - SDK (core-image-sdk)
      - profiles for Cluster, HUD, Telematics
- **meta-ivi-common**
  - This layer contains the common packages used in AGL, Tizen and GENIVI
    - Phase 1: this layer is required for baseline image, agl-image-ivi.
      - The recipes should be defined by each subsystems ?
- **meta-agl-demo**
  - This layer is for AGL Demo Platform.
    - Phase 2
      - Native / Web Application frameworks
      - Demo applications
        - Home Screens, Media browser/player, Vehicle data, Settings
      - Demo platform image (images/agl-image-demo)
- [TBD] **meta-qt5** is a layer for native app framework (upstream)
  - Phase 2
- [TBD] **meta-crosswalk** is a layer for web app framework (upstream)
  - Phase 2

Relations of Recipes and Subsystems

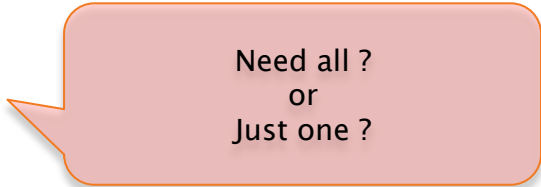| Layers	Directories for Recipes	Maintainer	Developer
meta-agl-demo	recipes-demo-platform	*Demo Applications*	*Demo Applications*
	recipes-qt (TBD)	*Native App Framework*	*Native App Framework*
	recipes-crosswalk (TBD)	*Web App Framework*	*Web App Framework*
meta-agl	recipes-core	*OS/Common Libs*	*OS/Common Libs*
	recipes-graphics	*Graphics*	*Graphics*
	recipes-kernel	*Kernel*	*Kernel*
	recipes-multimedia	*Multimedia*	*Multimedia*
	recipes-automotive	*Automotive*	*Automotive*
	recipes-connectivity	*Connectivity*	*Connectivity*
	recipes-navi-lbs	*Navi/LBS*	*Navi/LBS*
	recipes-devtools	*Meta-AGL (overall)*	*Anyone who needs devtools pkgs.*
	recipes-bsp	*Kernel*	*Kernel*
	recipes-extended	*Meta-AGL (overall)*	*Anyone who needs non-core pkgs.*
	recipes-support	*Meta-AGL (overall)*	*Anyone who needs support pkgs.*
meta-ivi-common	It would be the same structure as the **meta-agl**.		

T.B.D
Security and Speech Services

Any good way/idea to migrate meta-tizen/meta-ivi into meta-agl (meta-ivi-common) ?

Specification of agl-image-minimal

- Starting point of development AGL distribution
    - Almost same configuration as R-Car2 BSP
        - No package from meta-ivi and meta-tizen

- Images
    - **agl-image-minimal**
        - No GUI
    - **agl-image-minimal-weston**
        - Minimal GUI environment (wayland/weston)
    - **agl-image-minimal-x11**
        - Minimal GUI environment (X11)

Need all ?
or
Just one ?

- Package groups
    - These will be defined referring the R-Car2 BSP.

- <u>DISCUSSION: Which Yocto version shall we use?</u>
    - Yocto 1.6 (Tizen compatible)
    - Yocto 1.7 (GENIVI Compatible)
    - Yocto 1.8 (Latest Yocto)
    - Yocto 1.9 (Oct 2015 release)

Specification of agl-image-ivi

- The goal of phase 1
  - The baseline image of AGL Distribution

- Image
  - **agl-image-ivi**
    - It is **agl-image-minimal[-weston|-x11]** with some platform services and automotive services.
    - It contains the common packages used in AGL, meta-ivi and meta-tizen.

- Package groups
  - <u>Each sub-system team needs to select the packages that needs to be brought over from meta-ivi and meta-tizen.</u>

T.B.D

Specification of agl-image-demo

▸ The goal of Phase 2
  ◦ AGL Demo Platform ver 1.0
    • Web/Native Application frameworks
    • Home Screens, Media browser/player, Vehicle data, Settings

▸ Image
  ◦ **agl-image-demo**
    • It should be **agl-image-ivi** with Web/Native application frameworks and DEMO Applications.

▸ Package groups
  ◦ ...

T.B.D