



Automotive Grade Linux Lessons Learned

ELISA Workshop

Goddard Space Flight Center

December 10, 2024

Walt Miner

Senior Director AGL Community

The Linux Foundation

Who is This Guy?

- AGL Development and Community Manager since 2014
- Previously at MontaVista and Mentor Graphics
- Continental AG
- Motorola Mobile Devices
- Motorola Telematics Communications Group
- Defense Electronics at ITT Avionics and Northrop Grumman





Intro to AGL

What is AGL?



AUTOMOTIVE
GRADE LINUX

the only organization
addressing
all software in the car

 Infotainment	 Instrument Cluster	 Heads-up Display (HUD)
 Telematics/Connectivity	 Functional Safety	 Advanced Driver Assistance Systems (ADAS)

- Non-profit organization
- Open source Linux-based collaborative project
- Hosted at Linux Foundation
- Collaborating to build the car of the future through *rapid innovation* by uniting the automotive and software industries

Total of 9 OEMs supporting AGL!



> Millions of vehicles on the road with AGL
AGL first shipped in the 2018 Toyota Camry

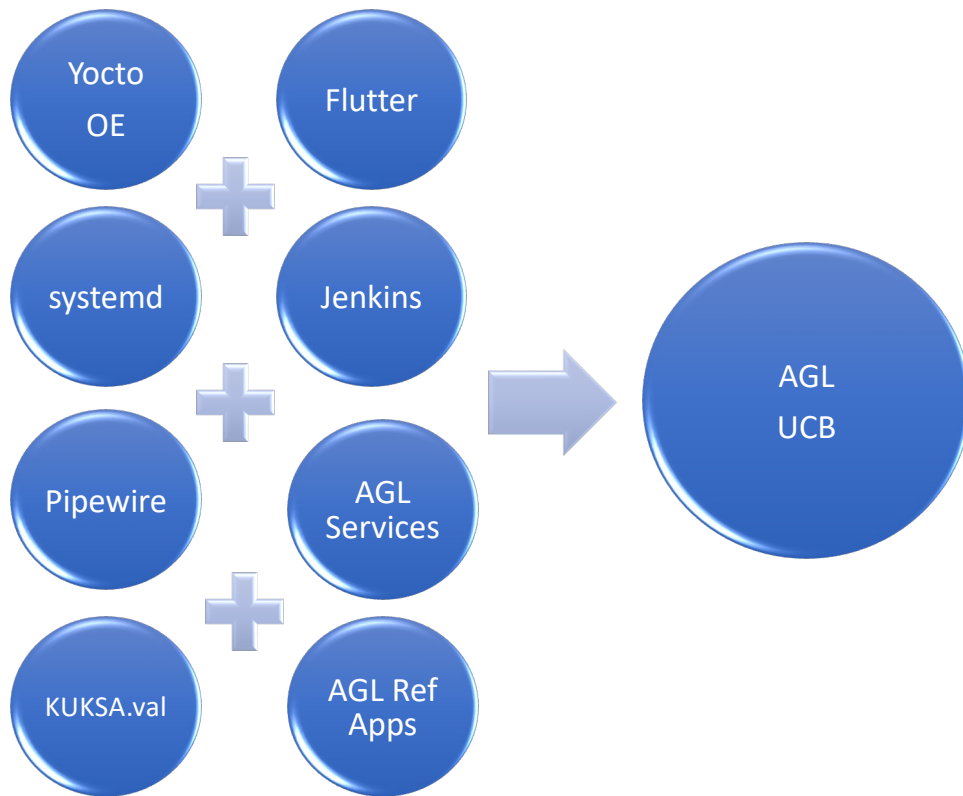


Vehicles with AGL:

- Toyota Camry, RAV4
- Toyota Prius Prime
- Lexus RX350, RX450
- Subaru Legacy, Outback
- Mercedes-Benz Vans
- More to come....

Camry image for depiction purposes only, actual vehicle may vary.

AGL Distro “Unified Code Base”

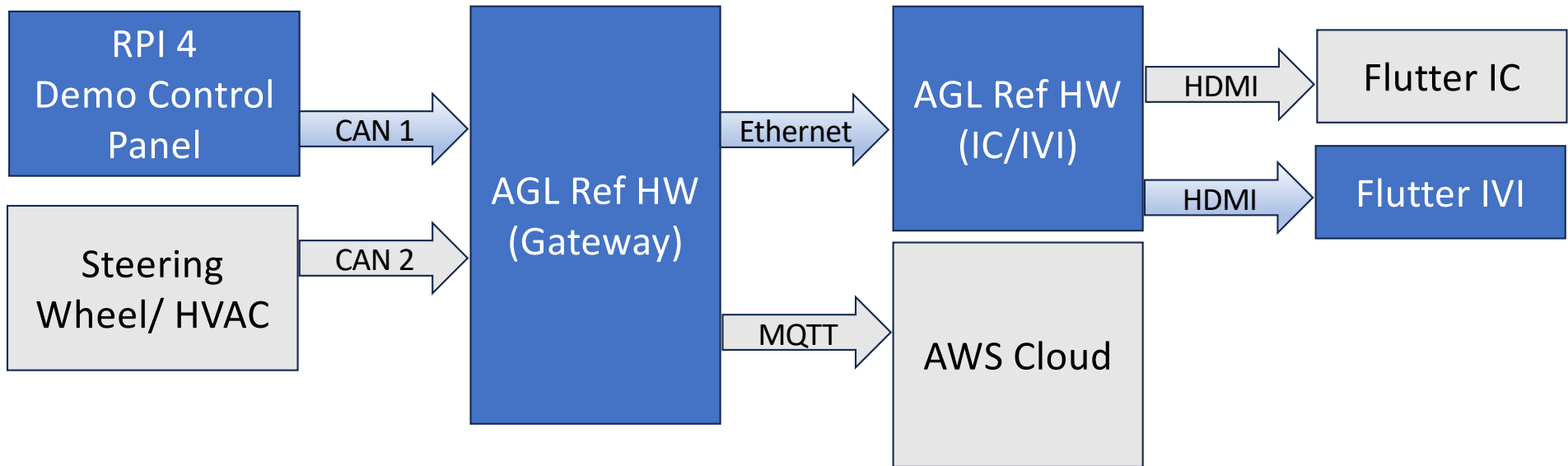


- *Unifying the best of Open Source into a single shared code base for the entire industry!*
- *Reduce fragmentation, focus on innovation and new features!*
- *Completely open source and transparent!*
- *Fully customizable; supported by an ecosystem of Tier Ones and service providers*

Device Profiles

- IC Expert Group has multiple IC configurations including bare metal Linux and container versions
- IVI Expert Group bringing embedded Flutter into vehicle
- Gateway device – data collection and aggregation for telematics, V2C and telemetry using VSS

Embedded World – Gateway demo



- Demo Control Panel sends speed and other vehicle data. Steering Wheel/HVAC CAN sends data usually directly connected to the IVI/IC hardware.
- Gateway aggregates data and forwards to AWS Cloud and IC/IVI
- Makes use of COVESA VSS and KUKSA.val implementation as vehicle data model
- All code is available in Ricefish 18.0.0













History of AGL

History of AGL

- AGL was initiated by the Linux Foundation in 2012.
- AGL was founded with the support of automotive OEMs including Toyota, Nissan, Hyundai, and Jaguar-Land Rover
- First two years efforts were bifurcated – JLR doing one thing, Japanese members working separately
- 2014 - Dan Cauchy brought on board as executive director
- 2015 – Moved from Tizen and OBS to Yocto as primary upstream. First All-Member Meeting held. IVI System Requirements Specification released. 60% member growth
- 2016 – Introduced Unified Code Base (UCB) with first release: Agile Albacore

AGL Aquarium

Two releases per year, every year since 2016!

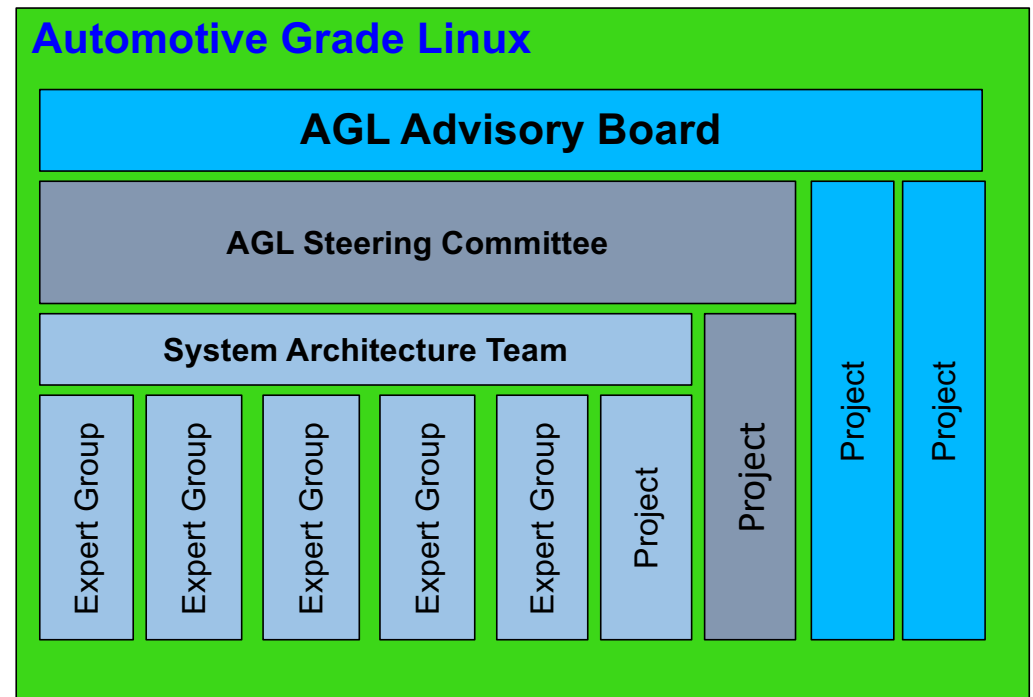
AGL Release	Code Name	Release Date	Fish
AA (1.0)	Agile Albacore	January 2016	
BB (2.0)	Brilliant Blowfish	July 2016	
CC (3.0)	Charming Chinook	January 2017	
DD (4.0)	Daring Dab	July 2017	
EE (5.0)	Electric Eel	January 2018	
FF (6.0)	Funky Flounder	Sept 2018	
GG (7.0)	Grumpy Guppy	March 2019	
HH (8.0)	Happy Halibut	August 2019	
II (9.0)	Itchy Icefish	March 2020	
JJ (10.0)	Jumping Jellyfish	Sept 2020	
KK (11.0)	Kooky Koi	Feb 2021	
PP (16.0)	Prickly Pike	Aug 2023	
QQ (17.0)	Quirky Quillback	Feb 2024	

History of AGL

- 2016 – Introduced Unified Code Base (UCB) with first release: Agile Albacore
- 2017 - Toyota Adopts AGL in production vehicles starting with the 2018 Camry.
- 2021 – Virtualization Expert Group formed – focus in expanding VirtIO for automotive use cases
- 2022 – Flutter Embedder developed and maintained by Toyota adopted for user experience
- 2023 – Software Defined Vehicle Expert Group formed
- 2024 – OSPO EG formed
- 2025 – UCB Terrific Tuna will be version 20 – ten years after starting work on the UCB

AGL Governance

- Bottom up approach, most technical decision making is in the hands of the Expert Groups
- Advisory Board responsible for overall direction, budget, strategy
- Steering Committee oversees AGL technical activities, software feature priorities
- System Architecture Team oversees end-to-end system and consistency
- Expert Groups can run one or more projects
- Projects may be directly managed by SAT, SC, AB



AGL Membership Levels

- Platinum – Automatic AB and SC seats with 2x votes
- Gold - Automatic AB and SC seats with 1x vote
- Silver – may be elected to AB and SC with 1x vote
- Bronze – no AB or SC seat
- Majority of membership dues is spent on development
 - 2 Linux Foundation FTE
 - Contractors funded for targeted development and QA

AGL Advisory Board 2024

AISIN



DENSO



MAZDA

Panasonic

RENESAS



TOYOTA

Active Expert Groups

- Find latest activities for all Expert Groups on our [Confluence Page](#)



AUTOMOTIVE
GRADE LINUX

Information about AGL can also be found on our [documentation site](#) and [wiki page](#).

Software

Want to find the latest releases and download software? Get the [latest release notes](#) and information on downloading source code, the Software Bill of Materials, pre-built binaries and more.

Expert Group Pages

Expert Group Pages

App FW and Security EG Home	Connectivity EG Home	Continuous Integration and Automated Testing	Instrument Cluster Home	IVI Production Readiness
OSPO EG Home	Reference Hardware System Architecture Home	SDV EG	System Architecture Team Home	Vehicle to Cloud EG Home

Open Source Program Office (OSPO) EG

- The Linux Foundation has numerous resource available for open source projects and OSPOs
- AGL forming an OSPO EG led by Toyota to address the special issues in the automotive community
 - Encourage participation in OSS communities
 - Contribution process for automotive companies
- Biweekly meetings starting November 12

Code Development

AGL UCB Architecture

Application Environment – Allows app developers to create apps using the toolkit of their choice

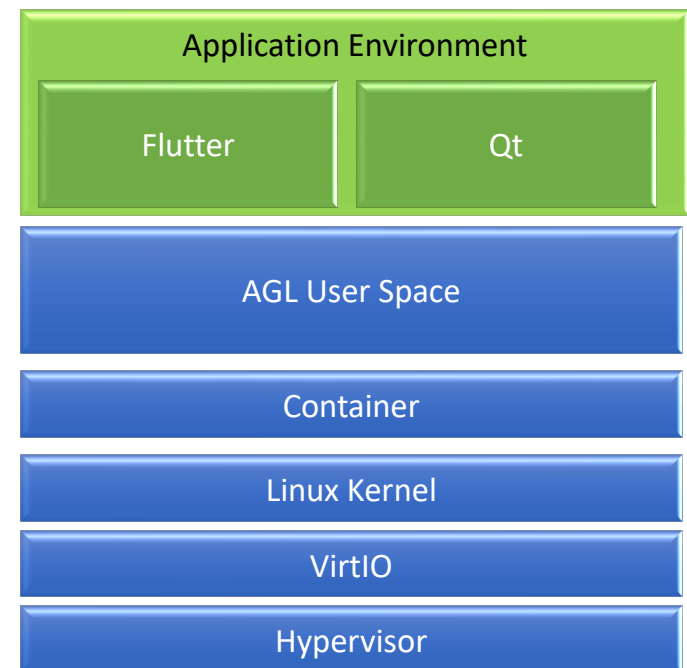
AGL User Space – Provides basic OS services as well as services unique to automotive use cases

Containers – less isolation than VM. Can run multiple versions of AGL for different device profiles

VirtIO – Provides a “straightforward, efficient, standard and extensible mechanism for virtual devices”*

Hypervisor – Allows AGL to be run in a virtual machine in mixed criticality use cases (AGL + RTOS) and/or with multiple instances of AGL for different device profiles

Device Profiles – include IC, IVI, telematics. AGL has multiple versions of each with different application environments

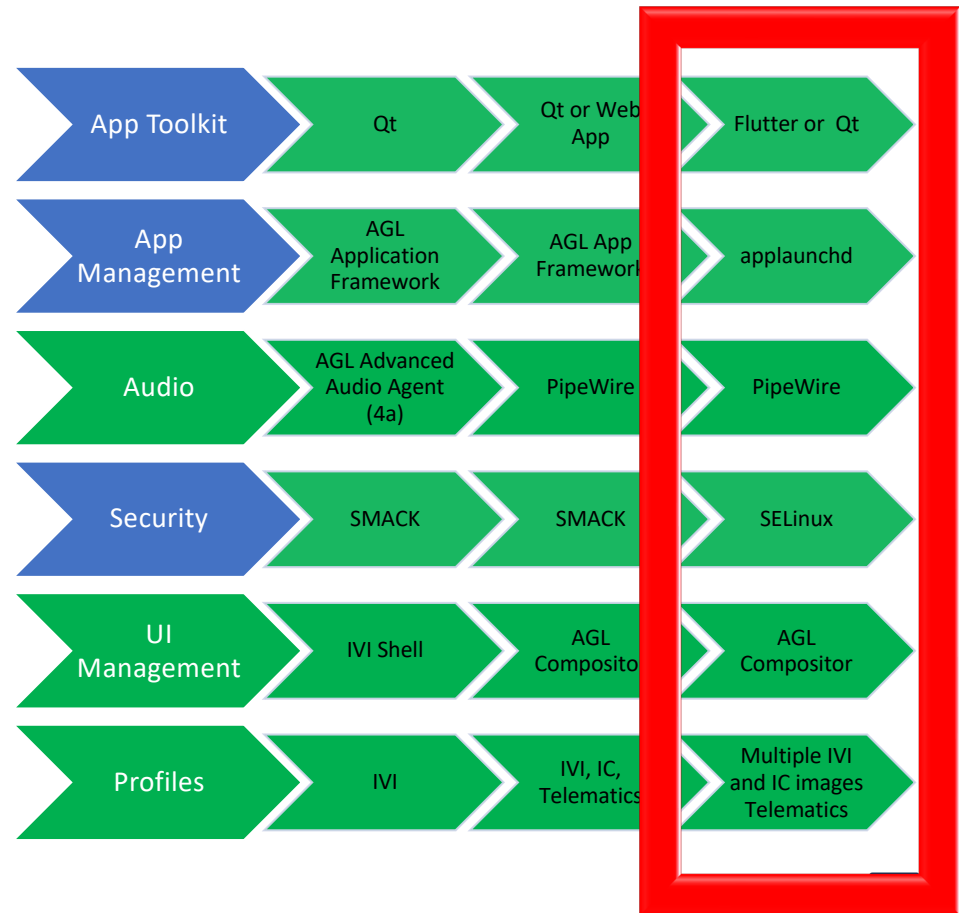


Code First/ Upstream First

- AGL has invested in automotive software components not available anywhere else
- Continually evaluating open source technologies to find best in class for automotive use cases
- AGL has invested (provided developers and upstreamed code) in open source projects such as Pipewire, Yocto, Lava, and others.
- Willing to collaborate with anyone who brings code

AGL Transformation

- 2022 saw us transition AGL Reference Apps and Service Binders to newer technologies
- Supplemented Qt reference Apps with Flutter
- Replaced Vehicle Signal Manager with kuksa.val and VSS
- Applaunchd created – now using gRPC!



AGL and Yocto

- AGL is a Gold Member of the Yocto Project and the YP and OpenEmbedded serve as our primary upstream
- YP enables multiple boards, device profiles, and even multiple images per device profile to be created from a single source tree
- Generally tracking YP LTS versions

AGL and Yocto LTS

- First Yocto LTS dunfell was tracked by AGL Lucky Lamprey for 12.1.x until it went EOL in April 2024
- YP 4.0.x (kirkstone) continues through April 2026
- New Yocto LTS 5.0 (Scarthgap) released in April 2024
- Quirky Quillback (17.0.x) will get YP Kirkstone updates until Kirkstone EOL

Yocto Project 5.0 (Scarthgap)

- Yocto Project's spring 2024 Long-term support (LTS) release released in April 2024
- Support currently planned until Spring 2028
 - This is notably different than the Dunfell and Kirkstone LTS releases, which were announced as 2 years, then extended to 4 years
- Yocto release cycle is roughly every 6 weeks
 - YP point releases pulled in and released by AGL within a couple of weeks
- Excellent presentation given by Scott Murray during ELC Europe about how we made the transition to Scarthgap
 - https://youtu.be/J_5wV0HbfWo?si=tZEhBYsHfhXi3DP&t=506

"next" Development

- New "next" branch already started
- Somewhat forced due to a breaking change in upstream YP master
 - New unpack directory and UNPACKDIR variable
- The AGL CI automatic rebase has been disabled to avoid meta-agl changes potentially breaking the YP autobuilder testing
 - It seems likely we will have to keep things this way until the next LTS in 2026

Collaboration Opportunities

- [AGL Confluence](#) with links to all Expert Groups
- All Expert Groups Meet every other week
- Weekly developer call on Thursdays at 9 am EST
- Mail list - <https://lists.automotivelinux.org/groups>
- AGL Discord Server - <https://discord.gg/ZztCaVeQVG>

AGL Resources

- [Confluence](#) – follow links to expert groups
- [Wiki](#) – including [latest schedule](#) and [release notes](#)
- [Documentation Site](#)
- [Developer Mail list](#)
- [Getting started guide](#)
- [Git repos](#) and [gerrit](#)
- [All-Member Meeting archive](#) with videos and slides
- [Jira](#)

Lessons Learned

Getting Code from Companies is Hard

- Very few automotive OEMs or their Tier One suppliers had OS contribution processes when we started the project
 - Same is probably true for aerospace companies and their suppliers
- It is still very hard for most automotive companies to contribute code – OSPO EG is an attempt to fix that
- We have had 2+ year waits for promised contributions to materialize

Make Use of Open Source Experts

- AGL changed from an FTE donation to a cash contribution model in 2015 – easier for companies to contribute cash than developers
- Analyzed potential packages for gaps in automotive use cases (audio, Bluetooth, connectivity, and others)
- Engaged package maintainers and open source experts to close the gap upstream rather than maintain code out of tree

Engaging outsiders is hard, but worthwhile

- A common open source myth is that there are hundreds of developers waiting to volunteer on your project – you need to find the few that are really there if there is no clear way to monetize the project
- Make multiple channels available for engagement (email, IRC, Discord, Slack, developer calls, meetups)
- Real automotive hardware is very expensive - find lower cost hardware (RPI, Beagle Bone) or emulators to get people started

Relying on board vendors for kernels

- AGL is code first – rely on board vendor for BSPs
- Yocto kernel update cycles very slow
- Need to solve the lag in BSP updates from LTS kernels to automotive hardware

Q&A

I'll be here all week or at least until the EOD Wednesday

THANK YOU
