Linux You Can Drive My Car

Hit the Open Road with Automotive Grade Linux
Open Source Summit North America - 2017

Walt Miner (@VStarWalt)
Community Manager, AGL, The Linux Foundation
Automotive Grade Linux

Collaborating to build the car of the future through rapid innovation

http://AutomotiveLinux.org
What is AGL?

- Non-profit
- Open source Linux-based collaborative project
- Hosted at Linux Foundation
- Focused on rapid innovation of vehicle software
Goals of AGL

• Build a single software platform for the entire industry
• Develop 70-80% of the starting point for a production project
• Reduce fragmentation by combining the best of open source
• Develop an ecosystem of developers, suppliers, expertise all using a single platform
AGL is CODE FIRST

AGL is a “Code First” organization!
Specifications lead to fragmentation!
**AUTOMOTIVE LINUX**

the only organization addressing all software in the car

<table>
<thead>
<tr>
<th>Infotainment</th>
<th>Instrument Cluster</th>
<th>Heads-up Display (HUD)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Telematics/Connectivity</td>
<td>Functional Safety</td>
<td>Advanced Driver Assistance Systems (ADAS)</td>
</tr>
</tbody>
</table>

**Autonomous Driving**
AGL Growth

AGL now has 105 members!
10 OEMs are members of AGL
Over 60% growth in past year
850+ developers on AGL mailing list
Total of 10 OEMs supporting AGL!
<table>
<thead>
<tr>
<th>Platinum</th>
<th>Bronze</th>
</tr>
</thead>
<tbody>
<tr>
<td>DENSO</td>
<td>Advanced Telematics</td>
</tr>
<tr>
<td>mazda</td>
<td>ALPS</td>
</tr>
<tr>
<td>Panasonic</td>
<td>ARM</td>
</tr>
<tr>
<td>Renesas</td>
<td>audiokinetic</td>
</tr>
<tr>
<td>Suzuki</td>
<td>Baylibre</td>
</tr>
<tr>
<td>Toyota</td>
<td></td>
</tr>
<tr>
<td>HONDA</td>
<td>Bright Box</td>
</tr>
<tr>
<td>NTT DATA</td>
<td>China Mobile</td>
</tr>
<tr>
<td>NTT DATA MSE Corporation</td>
<td>cinemo</td>
</tr>
<tr>
<td>Gold</td>
<td>Enea</td>
</tr>
<tr>
<td>HONDA</td>
<td>ETRI</td>
</tr>
<tr>
<td>MOBIS</td>
<td>FERRARI</td>
</tr>
<tr>
<td>Gold</td>
<td>Fujitsu</td>
</tr>
<tr>
<td>HONDA</td>
<td>Global Logic</td>
</tr>
<tr>
<td>MOBIS</td>
<td>Harmann</td>
</tr>
<tr>
<td>Gold</td>
<td>HITACHI</td>
</tr>
<tr>
<td>HONDA</td>
<td>Hi Corp.</td>
</tr>
<tr>
<td>MOBIS</td>
<td>Ihi</td>
</tr>
<tr>
<td>Gold</td>
<td>HiKari</td>
</tr>
<tr>
<td>HONDA</td>
<td>Infineon</td>
</tr>
<tr>
<td>MOBIS</td>
<td>Karamba Security</td>
</tr>
<tr>
<td>Gold</td>
<td>Konsulko</td>
</tr>
<tr>
<td>HONDA</td>
<td>LG</td>
</tr>
<tr>
<td>MOBIS</td>
<td>Linao</td>
</tr>
<tr>
<td>Gold</td>
<td>Linaro</td>
</tr>
<tr>
<td>HONDA</td>
<td>Link Motion</td>
</tr>
<tr>
<td>MOBIS</td>
<td>Marek</td>
</tr>
<tr>
<td>Gold</td>
<td>Mentor Automotive</td>
</tr>
<tr>
<td>HONDA</td>
<td>Microchip</td>
</tr>
<tr>
<td>MOBIS</td>
<td>Microware</td>
</tr>
<tr>
<td>Gold</td>
<td>NXP</td>
</tr>
<tr>
<td>HONDA</td>
<td>NVIDIA</td>
</tr>
<tr>
<td>MOBIS</td>
<td>Nexius</td>
</tr>
<tr>
<td>Gold</td>
<td>NXP</td>
</tr>
<tr>
<td>HONDA</td>
<td>Nosilab</td>
</tr>
<tr>
<td>MOBIS</td>
<td>O3iGo</td>
</tr>
<tr>
<td>Gold</td>
<td>OpenGL</td>
</tr>
<tr>
<td>HONDA</td>
<td>Opensynergy</td>
</tr>
<tr>
<td>MOBIS</td>
<td>Oracle</td>
</tr>
<tr>
<td>Gold</td>
<td>OpenSynergy</td>
</tr>
<tr>
<td>HONDA</td>
<td>OpenSynergy</td>
</tr>
<tr>
<td>MOBIS</td>
<td>Oracle</td>
</tr>
<tr>
<td>Gold</td>
<td>Oracle</td>
</tr>
<tr>
<td>HONDA</td>
<td>OpenSynergy</td>
</tr>
<tr>
<td>MOBIS</td>
<td>Oracle</td>
</tr>
<tr>
<td>Gold</td>
<td>OpenSynergy</td>
</tr>
<tr>
<td>HONDA</td>
<td>OpenSynergy</td>
</tr>
<tr>
<td>MOBIS</td>
<td>Oracle</td>
</tr>
<tr>
<td>Gold</td>
<td>OpenSynergy</td>
</tr>
<tr>
<td>HONDA</td>
<td>OpenSynergy</td>
</tr>
<tr>
<td>MOBIS</td>
<td>Oracle</td>
</tr>
<tr>
<td>Gold</td>
<td>OpenSynergy</td>
</tr>
</tbody>
</table>

Total of 105 member companies in 2017!
## Top 25 Git Committers in 2017

<table>
<thead>
<tr>
<th>Commits</th>
<th>Name</th>
<th>Company</th>
</tr>
</thead>
<tbody>
<tr>
<td>577</td>
<td>Romain Forlot</td>
<td>IoT.bzh</td>
</tr>
<tr>
<td>250</td>
<td>Jose Bollo</td>
<td>IoT.bzh</td>
</tr>
<tr>
<td>127</td>
<td>Ronan Le Martret</td>
<td>IoT.bzh</td>
</tr>
<tr>
<td>79</td>
<td>Jan-Simon Moeller</td>
<td>Linux Foundation</td>
</tr>
<tr>
<td>43</td>
<td>Matt Ranostay</td>
<td>Konsulko</td>
</tr>
<tr>
<td>32</td>
<td>Stephane Desneux</td>
<td>IoT.bzh</td>
</tr>
<tr>
<td>30</td>
<td>Jens Bocklage</td>
<td>Mentor Graphics</td>
</tr>
<tr>
<td>30</td>
<td>Petteri Aimonen</td>
<td>Individual</td>
</tr>
<tr>
<td>30</td>
<td>Loic Collignon</td>
<td>IoT.bzh</td>
</tr>
<tr>
<td>25</td>
<td>Karthik Ramanan</td>
<td>TI</td>
</tr>
<tr>
<td>21</td>
<td>Anton Gerasimov</td>
<td>Advanced Telematics Systems</td>
</tr>
<tr>
<td>20</td>
<td>Martin Kelly</td>
<td>Xevo</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Commits</th>
<th>Name</th>
<th>Company</th>
</tr>
</thead>
<tbody>
<tr>
<td>18</td>
<td>Sebastien Douheret</td>
<td>IoT.bzh</td>
</tr>
<tr>
<td>16</td>
<td>Matt Porter</td>
<td>Konsulko</td>
</tr>
<tr>
<td>16</td>
<td>Scott Murray</td>
<td>Konsulko</td>
</tr>
<tr>
<td>7</td>
<td>Kotaro Hashimoto</td>
<td>Mitsubishi Electric</td>
</tr>
<tr>
<td>7</td>
<td>Naoto Yamaguchi</td>
<td>AisinAW</td>
</tr>
<tr>
<td>6</td>
<td>Marcus Fritsch</td>
<td>Mentor Graphics</td>
</tr>
<tr>
<td>6</td>
<td>Milan Srdinko</td>
<td>ALPS</td>
</tr>
<tr>
<td>5</td>
<td>tte.zheng_wenlong</td>
<td>Toyota</td>
</tr>
<tr>
<td>4</td>
<td>Harunobu Kurokawa</td>
<td>Renesas</td>
</tr>
<tr>
<td>4</td>
<td>Philippe Coval</td>
<td>Samsung</td>
</tr>
<tr>
<td>4</td>
<td>Yuichi Kusakabe</td>
<td>Fujitsu-Ten</td>
</tr>
<tr>
<td>3</td>
<td>Multiple</td>
<td></td>
</tr>
</tbody>
</table>

1394 Total Commits  
45 Committers  
21 Companies  

- 01 Jan 2017 – 26 May 2017  
- Commits to master
## Commits by Company in 2017

<table>
<thead>
<tr>
<th>Company</th>
<th>Commits</th>
</tr>
</thead>
<tbody>
<tr>
<td>IoT.bzh</td>
<td>1039</td>
</tr>
<tr>
<td>Linux Foundation</td>
<td>79</td>
</tr>
<tr>
<td>Konsulko</td>
<td>77</td>
</tr>
<tr>
<td>Individual</td>
<td>38</td>
</tr>
<tr>
<td>Mentor Graphics</td>
<td>36</td>
</tr>
<tr>
<td>TI</td>
<td>25</td>
</tr>
<tr>
<td>Advanced Telematics Systems</td>
<td>21</td>
</tr>
<tr>
<td>Xevo</td>
<td>20</td>
</tr>
<tr>
<td>AisinAW</td>
<td>7</td>
</tr>
<tr>
<td>ALPS</td>
<td>7</td>
</tr>
<tr>
<td>Mitsubishi Electric</td>
<td>7</td>
</tr>
<tr>
<td>Renesas</td>
<td>7</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>1394</strong></td>
</tr>
<tr>
<td><strong>Committers</strong></td>
<td><strong>45</strong></td>
</tr>
<tr>
<td><strong>Companies</strong></td>
<td><strong>21</strong></td>
</tr>
</tbody>
</table>

- 01 Jan 2017 – 26 May 2017
- Commits to master
Other Numbers

- On pace to double last year (1791 commits)
- 19 new committers in 2017
- 64 unique individuals and 27 unique companies have contributed to AGL since the start of the project
- 7 individuals not affiliated with a member company have contributed
- 356 Jira issues closed since start of 2017
Thanks for all the fish…

- **AGL Releases:**
  - AA – Agile Albacore – Jan 2016
  - BB – Brilliant Blowfish – July 2016
  - CC – Charming Chinook – Dec 2016
  - DD – Daring Dab – July 2017
  - EE – Electric Eel – Dec 2017
  - FF – Funky Flounder – July 2018
Vision for 2017

• Daring Dab July 2017
• Electric Eel Planned for CES 2018
• AGL App Framework Complete
• AGL Reference Apps available for both Qt5 and HTML5
• All APIs available as App FW Service Binders
• IC, Telematics and other profiles
• Complete set of documentation
• Daily, Weekly, and Event based board testing
• Published Test Results
2017 AGL Schedule

Charming Chinook
- 3.0.0
- 3.0.1
- 3.0.2
- 3.0.3
- 3.0.4
- 3.0.5

Daring Dab
- Feature Development
- RC1
- RC2
- RC3
- 4.0.0
- 4.0.1
- 4.0.2
- 4.0.3
- 4.0.4

Electric Eel
- Feature Development
- RC1
- RC2
- RC3.0.0

See https://wiki.automotivelinux.org/schedule#latest_schedule for latest schedule information

As of 08 Sep 2017
CHARMIND CHINOOK AND CES 2017
Charming Chinook

- Released January 6
- Patch Releases
  - 3.0.1 – 30 Jan 2017
  - 3.0.2 – 08 Mar 2017
  - 3.0.3 – 29 Apr 2017
  - 3.0.4 – 05 Jul 2017
  - 3.0.5 – 22 Aug 2017
Charming Chinook

- Release notes, binary downloads, source code and more at
  
  https://wiki.automotivelinux.org/agl-distro/release-notes
DARING DAB
Daring Dab

• Updated to Yocto 2.2 (morty)
  • Pulls in updated kernel and user space packages
  • Improved tooling

• App Framework Improvements
  • CC Lessons Learned
  • App Templates
  • Security updates
  • App Installation and First-time Startup

• SOTA Updates
Daring Dab

- App FW Service Binders
- API V2
  - Stub generation from OpenAPI JSON
  - Permission integration through OpenAPI
  - Monitoring
- System-D
  - Private user-ID
  - Smack label
  - Cgroups
Daring Dab

• SDK Improvements
  • Docker image to eliminate host dependency issues
  • Available for reference boards with published images that include graphics drivers
  • Enables rapid AGL application development
  • Support for Qt (HTML5 planned!)
  • No Yocto knowledge is needed or assumed for SDK users
Daring Dab

• Secure signaling and notifications to and from vehicle buses
  • Optimized usage of CAN sockets
  • Migrated to BCM sockets
  • Map logical CAN bus names from CAN databases to physical can as present on hardware.
Daring Dab

- Available App FW Service Binders
  - Audio Manager
  - Bluetooth
  - Tuner
  - CAN signaling
  - Media Scanner
Daring Dab

• Apps
  • App business logic separated from API
  • Templates created to ease new App development
  • Homescreen improvements
  • CAN Test App – New
  • Phone App works with Bluetooth HFP
• Media Player
  • Bluetooth A2DP and AVRCP
  • USB and SD card detection
Daring Dab

- BSPs - Default tunings per architecture (e.g., Intel, ARM 32 and ARM 64)
Daring Dab Board Support

- Reference BSPs – Fully supported by manufacturer, CI, etc.
  - Renesas R-Car 3 – M3 board (ARM 64)
  - Renesas R-Car 2 - Porter board (ARM 32)
  - Intel - Minnowboard Turbot and Joule (Intel 64 bit)
  - TI - Jacinto 6 - Vayu board (ARM 32)
  - QEMU (Intel 64 bit)
- Community BSP – Best effort by AGL community
  - Raspberry Pi 2/3 (ARM 32 bit)
  - NXP – i.MX6 – SABRE (ARM 32 bit)
  - QCOM – Dragonboard 410-c (ARM 64 bit)
  - TI BeagleBone Black (ARM 32 bit medium profile)
ELECTRIC EEL
Electric Eel

<table>
<thead>
<tr>
<th>Jul</th>
<th>Aug</th>
<th>Sep</th>
<th>Oct</th>
<th>Nov</th>
<th>Dec</th>
<th>Jan</th>
</tr>
</thead>
</table>

Electric Eel

New Features Developed

Stabilize

Patch Updates

App Development

Start of Development 7.10

RC1 9.30
RC2 10.30
RC3 11.30
Final Release 12.15
CES 2018
5.0.1
Electric Eel

- Yocto 2.3 – Pyro branch
- Rework Yocto Layers for headless devices
- Window Manager and Graphics Architecture
- Home Screen as an app
- Audio Manager architecture
- Additional Service Binders
  - Telephony
- CI and Automated Test Improvements
  - Fuego and Lava integration
  - “Lab in a box”
- CES 2018 Demo apps
GETTING INVOLVED - CODE STRUCTURE
Software Configuration Requirements

- AGL Demonstrator Code
- AGL Community Development
- AGL Extra Features
- AGL Core Distribution

Readily determine the required contents of the AGL distribution for product developers.
AGL Core Distribution

- Stable Yocto release
- Reference BSPs fully supported by the board manufacturer or chip vendor
- Documentation and tooling for building and deploying reference BSPs
- Tooling to allow selection of optional features in the core build
- Test results provided using AGL Test Framework
- Fully supported with updates for at least 6 months
- Defined by Yocto layer – meta-agl
Software Configuration Requirements

Provide a mechanism for enabling optional and/or experimental features

- AGL Demonstrator Code
- AGL Community Development
- AGL Extra Features
- AGL Core Distribution
AGL Extra Features

- Builds on AGL Core Distribution
- Features are fully tested and supported as part of AGL release
- AGL environment set up provides extra features that may be enabled by device creators
- Device profiles (e.g., Telematics, ADAS) will be provided in AGL Extra Features
- Yocto layer – meta-agl-extra
AGL Community Development

- Place for developing code that may eventually make it into AGL Core or Extra Features
- Snap shot builds for experimental features to facilitate collaboration
- Community BSPs without official support
- No formal QA – basically whatever the community can provide
- Defined by Yocto layer – meta-agl-devel
Software Configuration Requirements

AGL Demonstrator Code

AGL Community Development

AGL Extra Features

AGL Core Distribution

Environment for demonstrator and new feature development
AGL Demonstrator Code

- Code developed to demonstrate specific features and/or releases of AGL
- CES and other demos
- Intended for “one shot” development
- Provided “as-is”
- Yocto layer – meta-agl-demo
Release Management

- Twice per year release of AGL Distribution includes
  - AGL Core Distribution and Extra Features
  - All code and tooling with test results
  - Full test results for reference BSPs
  - As-Is demo code, Community Developed features, and BSPs
- Support biannual releases with code fixes for six months
- Long term support (2+ years) for selected releases
- Daily snapshot builds for specific configurations
- Pre-release candidates to allow developer collaboration and coordinated testing
AGL Yocto Layers

AGL Core Distribution

AGL Community Development

AGL Extra Features

AGL Demonstrator Code

meta-agl-devel
- meta-agl-sota
- meta-<BSP>

meta-agl-extra
- meta-iot-appfw
- meta-qt5

meta-agl
- meta-agl
- meta-agl-bsp
- meta-ivi-common
- meta-agl-security

- meta-poky
- meta-oe
- meta-<BSP>
GETTING INVOLVED – EXPERT GROUPS
Expert Groups

- Six Current EGs
  - App Framework and Security
  - Connectivity
  - UI and Graphics
  - CI and Automated Test (CIAT)
  - Navigation
  - Virtualization

- EGs focus on requirements and architecture
  - Kernel, Common Libs and OS are subsystem teams
App Framework and Security EG

- Application lifecycle (install, run, remove, applications)
- SDK and application developer experience both in security and APIs
- Security framework, policies, and strategy for the distribution
- Network and vehicle firewalls in conjunction with the Connectivity EG
- Software Update and secure update
- Diagnostic log and trace
- Secure boot

https://wiki.automotivelinux.org/eg-app-fw
App Framework Tasks

- Upcoming Improvements to App Framework
- Conversion to systemd completed in CC
- Complete cgroups implementation
- Namespace (Will be included in Daring Dab)
- Resource management using cgroups (memory, CPU usage, CPU affinity, network bandwidth)
App Framework Tasks

• Identity and user management via Bluetooth/NFC
• Key management for app installation and the manifest that gives the rights within cynara policy checker
• Consistent templates and documentation for creation of apps and widgets
• Documentation of how to convert legacy apps to AGL Apps using Qt5 or HTML5 backend.
• Consent management for managing resources and applications. Examples include payments, enabling LBS.
UI and Graphics EG

• AGL Compositor, Layer Manager, Window Manager and GPU interface

• Multimedia video manager (including multi-display and display sharing) and audio manager, and media manager/player.

• Browser Engine

• Speech Recognition

https://wiki.automotivelinux.org/eg-ui-graphics
Graphics and UI Tasks

• Wayland update to 1.11 from 1.9
• Refactor Home Screen including splitting out Window Manager
  • Updated Window Manager for better secondary display support
  • Pop-up support
  • Focus Management of out of focus applications
• Latest Proposal
Graphics and UI Tasks

- Internode display protocol
  - Control IC display output from navi running on IVI system
- Theming and skinning
- Improved PulseAudio and Audio management configuration and policies
- Speech services API and integration into reference apps
- Support for Chromium browser engine
Connectivity EG

- **Vehicle Connectivity (CAN, MOST, LIN, AMB)**
- **Network and vehicle firewalls**
- **Bluetooth, Wifi, NFC**
- **Smart Device Link (SDL)**
- **Cloud Connectivity (Iotivity)**
- **Connected Car**
- **Remote Vehicle Interactions (RVI)**

[https://wiki.automotivelinux.org/eg-connectivity](https://wiki.automotivelinux.org/eg-connectivity)
CI and Automated Test EG

• Build and smoke test of Gerrit submissions on all hardware
• Daily snapshot build and testing
• Device tests on real hardware
• Test environments such as JTA and Lava
• Test suites such as LTP
• UI testing (OpenQA)

https://wiki.automotivelinux.org/eg-ciat
CIAT Task List

- Upstream our fork of Fuego from JTA
- Create system to publish and evaluate test results
- Add new reference boards to Lava and Jenkins
- Ensure CI system builds and checks for optional features when testing patches
- Static code analysis tools
Navigation EG

- Navigation API
- Location Based Services API
- Reference Navigation and POI apps
- Speech recognition integration

https://wiki.automotivelinux.org/eg-navi
Virtualization EG

- Hypervisor/ LXC
- Looking at Xen, Jailhouse, and others

https://wiki.automotivelinux.org/eg-virt
DEVELOPER RESOURCES
AGL Documentation

• Getting Started
  https://wiki.automotivelinux.org/start/getting-started

• AGL documentation site
  http://docs.automotivelinux.org/

• AGL Wiki
  https://wiki.automotivelinux.org/

• AGL Jira
  https://jira.automotivelinux.org/
Get The Code

• Pre-built binaries and source tar balls available
  • [https://www.automotivelinux.org/software/download](https://www.automotivelinux.org/software/download)
• Latest Source Code and Build Instructions
  • [https://wiki.automotivelinux.org/agl-distro/source-code](https://wiki.automotivelinux.org/agl-distro/source-code)
• Release Notes
  [https://wiki.automotivelinux.org/agl-distro/release-notes](https://wiki.automotivelinux.org/agl-distro/release-notes)
AGL Code

- AGL gerrit
  
  http://gerrit.automotivelinux.org/

- AGL git
  
  https://git.automotivelinux.org/
Build Options

- Once you have the repos set up use
  
  $ source meta-agl/scripts/aglsetup.sh –h

- To determine available boards and build options

- Example – Build QEMU AGL Demo
  
  $ source meta-agl/scripts/aglsetup.sh -m qemux86-64 agl-demo agl-netboot agl-appfw-smack

  $ bitbake agl-demo-platform
SDK for App Developers

- Docker image to eliminate host dependency issues
- Available for reference boards with published images that include graphics drivers
- Enables rapid AGL application development
- Support for Qt (HTML5 planned!)
- Documentation
- No Yocto knowledge is needed or assumed for SDK users
2017 F2F Workshops

- Feb 8 -10 – AGL AMM - Tokyo
- Apr 4 – 5 – Microchip - Karlsruhe, Germany
- May 31 – Automotive Linux Summit - Tokyo
- July 11 – 12 – San Jose
- Aug 30 – 31 – Yokohama, Japan
- Sep 5 – 7 – Vannes, France
- Sep 13 – 14 - Montreal
- October 18 – 20 - Fall AMM – Dresden, Germany
- Nov 14-16 – CES Integration Session – Yokohama
- Dec 13-14 – CES Integration Session - Yokohama
Q & A
Q & A

• This is the segment where
  • You ask intelligent, well thought out questions
  • I ramble pointlessly and unintelligently

• And/Or
  • You ask “dumb” questions
  • I respond with concise, insightful, and well-reasoned answers
THANK YOU