GENIVI Android Automotive Special Interest Group

Gunnar Anderson – Development Lead at GENIVI
Philippe Robin – PMO Lead at GENIVI

GENIVI Virtual Technical Summit | May 12, 2020
Why is GENIVI Alliance working on Android Automotive?

GENIVI has promoted Open shared software and standards and created business opportunities in the automotive industry for over 10 years

- **2016-2017** – Cross-Domain Interaction, adapting to Multi-OS reality, safety demands
- **2018-...** – Multi-OS, Connected Cockpit, Virtualization,
  = Integration technologies for diverse, distributed and cloud-connected EE architectures.
- **2020-...** – Big-picture, end-to-end integration, adapt to latest industry trends

Example: Android Automotive Special Interest Group
PROJECT CHARTER - Android™ Automotive Special Interest Group (SIG)

Opportunity Statement
Automotive OEMs are increasingly adopting Android Automotive (embedded) as a solution for their IVI stack. This adoption has introduced a series of challenges around integrating the Android Automotive embedded solution into existing legacy software and into other systems present in the vehicle (security, vehicle data, etc.).

Through a GENIVI-hosted Android Automotive SIG project, OEMs, their suppliers and the broader cockpit software ecosystem can discuss requirements, identify gaps and provide an aligned, community voice for discussion with the Google Android Automotive team.

Current Project Focus:
- **Vehicle HAL / Vehicle Data APIs project**
  - System Level Architecture for accessing vehicle data
    - Including integration with the “rest” of the vehicle, using e.g. Some/IP
  - Software Architecture / API definition, including authentication of app
  - Proof-of-concept implementation
- **Audio HAL project**
  - System Level Audio
  - Software Architecture / API definition
  - Proof-of-concept implementation

Proposed Areas of Project Focus
- List of extensions required for Android in an automotive environment
  - location-based services - capability to plug in “other” navigation engines in Android Automotive considered as important
    - looking for a topic owner
  - multi-display support with Android and non-Android systems - analysis in-progress by the graphics sharing team
  - lifecycle, diagnosis and health monitoring
- Long-term maintenance
  - Defining boundaries where Tier 1s / OEMs must take primary responsibilities over Google Android Automotive team support
  - Keeping an automotive system updated to support new versions of Android
Vehicle Data APIs / VHAL Report
Android Automotive SIG - Vehicle Data APIs - VHAL – Report

System-level architecture for accessing vehicle data – starting point

![System-level architecture diagram]
Different software architectures for accessing vehicle data

Main thread of work: Establish a unified way on accessing vehicle data

• 4 architectural concepts for accessing vehicle data identified
  • External Data Server
  • Data Server inside the Framework (so-called Internal Data Server)
  • SomeIP stack inside the AOSP Framework
  • Google VHAL + OEM Extensions

• Implementation in-progress for External Data Server concept (see next slide)
  • Other concepts will be presented in the workshop on Day 2 – 13 May – 15:00-18:00 CET
Design choices made by the group

- VSS adoption (Vehicle Signal Specification) – will be introduced in the CCS project report
- Coexistence of Google VHAL and Data Server
- Application authentication
- Interface with the «rest» of the vehicle, e.g. via Some/IP
External Data Server Proof-of-Concept

Work breakdown structure is available, implementation in-progress

- Demo will be shown in the workshop
- Authentication Service implementation
  - Sources: [https://github.com/stefanwysocki/aasig_dev_platform/tree/develop/vendor/genivi/modules/VssAuthenticationService](https://github.com/stefanwysocki/aasig_dev_platform/tree/develop/vendor/genivi/modules/VssAuthenticationService)
- External Data Server implementation
  - **Data server**: Publishing of GraphQL comprehensive example done
    - Sources: [https://github.com/GENIVI/vss-graphql](https://github.com/GENIVI/vss-graphql)
  - **VSS feeder**: Basic functionality is working
    - Sources: [https://github.com/GENIVI/vss-feeder](https://github.com/GENIVI/vss-feeder)

Several low-hanging fruit activities identified call for volunteers!! Please contact the team
Android Automotive SIG - Vehicle Data APIs - VHAL – Report

• Team participants: BMW, Daimler, Jaguar Land Rover, TietoEVRY, Bosch, EPAM

• Agenda for this week’s workshop:
  • Project overview & proof-of-concept demo
  • Topics discussion
    • Google Vehicle Properties Implementation based on GraphQL Service.
    • Permission groups specification.
    • Translation of permission groups.
    • JWT Token what will be included and how it will be done? And generation process?
  • Feature content definition for next milestones
  • Technical readiness level assessment and discussion on how and when to reaching out to Google
Audio HAL Report
• Both strategies have advantages and disadvantages
  • Relying on Android only does not fulfill some safety requirements
  • Considering Android as sources and sinks only does not take enough advantage of it
• Therefore each topic has to be analysed and assigned: inside or outside Android
• Reference design from Google side by side to the GENIVI Proof Of Concept proposal
• Android design relies on an External System (Mixer, Amplifier, Safety signals,…) 
• GENIVI Proof Of Concept tries to provide an concrete instantiation of the Audio Control Split
Android Automotive SIG – Audio HAL – Management Report

- Team participants: BMW, Mobis, TietoEVRY, EPAM, Bosch, Analog Devices

- Main thread of work - How to Control a Car Audio System with Android
  - 1st proof-of-concept implementation started - Getting raw streams out of Android
    - Demo will be shown in the workshop
    - Source code repository in preparation

- Agenda for this week’s workshop:
  - Project overview & proof-of-concept demo
  - Proof-of-concept details discussion
    - Feature content definition for next milestones: Audio Manager integration, avoiding Audio Manager modifications and long term API support strategies
    - Proof-of-concept prerequisites (external raw streams, injecting input streams)
  - Going through list of prioritized topics for and refining the priorities asking participants about their opinion and feedback
  - Technical readiness level assessment and discussion on how and when to reaching out to Google
  - Backup topics: Early Audio, Audio Calibration / Equalization, Controlling Audio Effects (Google might improve it in next version)
Contributing
Timeline

- Milestone 1 – **GENIVI Virtual Technical Meeting** (12-14 May)
- Milestone 2 - Internal milestone (early Q3 - July)
- Milestone 3 - Fall All Member Meeting, Leipzig, Germany (last week of October)
- Milestone 4 - CES 2021, Las Vegas, USA (early January 2021)

- **GENIVI Virtual Technical Meeting**
  - AASIG VHAL Session : Wednesday 13 May – 15:00-18:00 CET
  - AASIG Audio HAL Session: Thursday 14 May – 9:00-12:00 CET.
Contributing

- **Weekly telcos**
  
  - Tuesdays – 17:00 CET (US friendly time) – Vehicle Data APIs / VHAL
  
  - Thursday - 11:30 CET (India & Asia friendly time) – Audio HAL


- **Wiki**
  
  - Android Automotive Project Wiki: [https://at.projects.genivi.org/wiki/x/XgA4Ag](https://at.projects.genivi.org/wiki/x/XgA4Ag)
  
  - Vehicle Data Access / VHAL - External Data Server Proof-of-Concept Work Breakdown Wiki: [https://at.projects.genivi.org/wiki/x/RgXYAg](https://at.projects.genivi.org/wiki/x/RgXYAg)

- Audio HAL - System Level Audio Wiki: [https://at.projects.genivi.org/wiki/x/BAlyAw](https://at.projects.genivi.org/wiki/x/BAlyAw)
Thank you!

Visit GENIVI:
http://www.genivi.org
http://projects.genivi.org

Contact us:
help@genivi.org
Vehicle Data Access - via Internal Data Server implementation

- Defined by:
  - Google
  - Genivi
  - OEM / Tier1

Diagram:
- Application layer
- Framework layer
- HAL layer
- Virtual ECU (same / different ECU)
- Head Unit (on-board)
- VSS Data Server
- VSS Feeder
- VSS Database
- TODO: Franca Service connection (SomeIP)
- TODO: Check Signal to Service translation in Adaptive Autosar
- TODO: Create a layer concept for the Franca to VSS leaf mapping.
- TODO: Implement Feeder as a PoC
- TODO: Agree on PoC Use Cases Services
- TODO: Create PoC SomeIP simulation component to playback some agreed use cases
- TODO: Select and implemented VSS data storage inside the Feeder
- TODO: APP access for in-vehicle data (e.g. App manifest permissions layer concept)
- TODO: APP authorization to access the Data Server (e.g. JWT)
- TODO: Finalize the permission layer concept.
- TODO: App request/response serialization
- TODO: Resolve requested date for the APP from the VSS data structure.
- TODO: Change/write data values for requested data leafs
- TODO: Handle the subscription for APPs.
- TODO: Write a generator which will handle permissions in the data server.
- TODO: Implementing the APP permissions based on permission defined/proposed in VSS layers.
- TODO: Implement access token request
- TODO: Generate access token for the APP including the APP permissions
- TODO: App request/response serialization for the Client (JSON to Java)
- TODO: Request APP permissions from Package Manager
- TODO: Implement Token management for each Android APP

Rational:
1.
Vehicle Data Access via Customized HAL - Google VHAL + OEM Extensions inside

Architectural proposal I (via custom HAL)
Vehicle Data Access – via Some/IP Service - SomeIP stack inside the Framework

Look at vsomeip port to AOSP: [https://github.com/GENIVI/vsomeip/pull/107](https://github.com/GENIVI/vsomeip/pull/107)

Architectural proposal III (via Global SomeIP Service)