

Welcome to the GENIVI Open Source Projects Wiki



GENIVI has expanded its scope from its strength in Linux-based IVI and automotive open source software to focus on helping automakers integrate the multiple operating systems present in the centralized and connected cockpit. We call this new strategy **Multi-OS Integration**.

[Vehicle E/E and software architecture trends](#) have led GENIVI to advance into concepts of central domain ECUs, cross domain ECUs (domain fusion) and eventually to a vehicle computer approach. Longer-term, GENIVI is also exploring [vehicle cloud computing and connected services](#).

New projects launched in this Multi-OS integration scope include:

- [Android™ Automotive Special Interest Group \(SIG\)](#)
- [Multi-OS Integration Project](#).

Work done in late 2017 and throughout 2018 in what GENIVI called [domain interaction projects](#) laid a strong foundation for this new multi-OS integration scope expansion. The following domain interaction projects are active and have relevance in the new integration scope:

- [Graphics Sharing & Distributed HMI](#)
- [Advanced Hypervisor APIs](#)
- [Generic Communication Protocols](#).

Some *quick to read* and interesting results are published as [Technology Briefs and Whitepapers](#).

Login account

We are currently reworking the account creation process.

For the moment, to apply for a login account for JIRA and Confluence, please email your request to [Gunnar Andersson](#) and [Nicholas Contino](#) and make sure you get a response from one of us that a ticket has been created. ⚠️ *Do not create an account in JIRA using the "Sign Up" link.*

For bug & project tracking, use [JIRA](#)

Multi-OS Integration Work

- [Android™ Automotive SIG Project](#)
- [Multi-OS Integration Project](#)
- [Domain Integration Projects Description](#)
 - How to join - Quick links: [\[GSHA\]](#), [\[GPRO\]](#), [\[HV\]](#), [\[CCS\]](#), [\[SHDA\]](#)
- [Technology Briefs and Whitepapers](#)
 - Ramses case study : Distributed Graphics Control Through API Remoting
 - [tech brief](#)
 - [code](#)

Architecture

- [Overview of Reference Architecture](#) - (You may want to start at the *beginning* of the [GENIVI Resource Kit](#))

Software Components and Standard Interfaces/APIs

- [IPC CommonAPI C++](#)
- [vSomeIP](#)
- [Diagnostic Log-and-Trace](#)
- [Vehicle Signal Specification](#)
- [...more](#)

Platforms/Baselines

- [Yocto Baseline \(meta-ivi\)](#)

The [GENIVI Security Team](#) is actively exploring ways to make vehicle software systems more secure against threats and hacks of all types.

GENIVI continues to provide a [baseline](#) for [GENIVI Compliant™ IVI platforms](#) and the [GENIVI Development Platform](#) (GDP) will remain accessible for demonstrators and other code development activities.

GENIVI continues to host a number of IVI software components in our [github repository](#).

For more information on engaging in GENIVI, please contact help@genivi.org or visit www.genivi.org.

Demonstrators, development, and tooling

- [GENIVI Development Platform Tools Team](#)

How-Tos, tutorials and other instruction

- [For Maintainers](#)
- [Propose a project](#)
- [Submit code](#)
- [Create a recipe for inclusion in the baseline](#)
- [Google Summer of Code 2018 instructions and ideas](#)