



Simple Steps to Contributing Code to GENIVI

Thursday, April 28th, 11:00 – 11:30

Claus-Peter Wiedemann
Senior Manager and LRT Lead
BearingPoint

The GENIVI License Review Team (LRT) ...

- ... maintains the GENIVI license compliance infrastructure
 - Public Licensing Policy and internal license compliance wiki pages
 - Process elements related to license compliance (e.g. code contribution)
 - Templates for component specifications and other documentation
 - Tooling (Black Duck Protex, Fossology, SPDX)
- ... fosters license compliance of GENIVI projects
 - Weekly scans of the GENIVI projects and compliance release scans
 - Interpretation of the scan results (context based issue identification)
 - Remediation support to maintainers/contributors
 - Verification of required elements in the source code (e.g. license headers) and on the project web site
- ... educates members
 - Central resource for any license related questions (e.g. license to choose, obligations,...)
 - Regular sessions at the GENIVI AMMs
 - On-site education for members (on demand)

Agenda

- What makes Open Source Communities successful?
- Why should I contribute to GENIVI projects?
- What are the steps to contribute to GENIVI?
- What should I check before contributing?
- Q&A

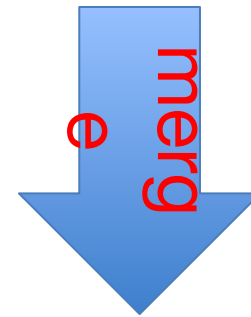
Why are Open Source communities successful?

- Transparency
- Collaboration
- Self-organization
- Meritocracy
- Due diligence



What many automotive companies do today

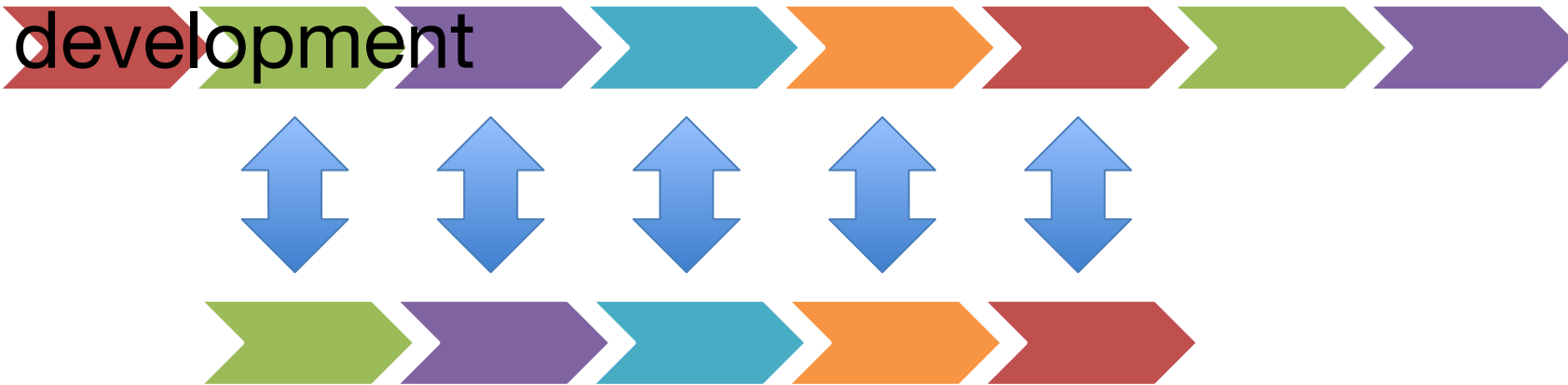
Community project



Specialized version (local)

What automotive companies should do instead

Community project



Local changes / additions pushed upstream

- Contributions are vital for the GENIVI OSS community
- Community can give feedback and innovate also on local changes/additions
- Local maintenance effort can be vastly reduced

Why contribute? Why give precious code away?

Precious?

- Code that has (and will) cost a lot to develop and maintain?
- Code that implements functionality customers will pay *extra* for in the *future*?

Maintaining and improving non-differentiating code

- is costly
- occupies resources who could work on innovation

You cannot compete with the innovation velocity of OSS communities anyway

Using OSS strategically and contributing to OSS communities allows you to focus on innovation that pays off

Contribution Benefits

Contribute to OSS projects you depend on, in order to

- share bug fixes and enhancements and significantly reduce maintenance cost
- gain influence on roadmap and feature development (meritocracy)

Create OSS projects from home-grown, non-differentiating components, in order to

- set the standard and save on integration
- build and motivate a community to drive innovation
- reduce development and maintenance effort



Simple steps to contribute to GENIVI

- 1) Select the code to contribute and obtain (internal) approval(s)
- 2) Prepare the code
- 3) Prepare and send the submission email

You're done!

NEVER submit any code to GENIVI without (internal) approval(s)

Selecting Code – What to consider

Business aspects

- Only non-differentiating functionality
- Relevant for your products (past/future)
- Do not try to value the code based on (past, sunk) development cost

Technical aspects

- No dependencies on proprietary code/functionality
- Some uniqueness (features, performance, footprint,...)
- Reasonable documentation
- Reasonable quality

Legal aspects

- Ownership/copyright or appropriate license available
- No obvious 3rd party patent or trademark issues

Preparing the code – Don't forget to ...

... add appropriate license headers to files with meaningful content, especially

- code (including xml, fidl,...)
- documentation

No license headers are usually needed for

- (build-)scripts
- (auto-)generated Code
- property files
- Readme, Release Notes

If unsure, add a license header

```
/*
 * @licence app begin@
 * SPDX license identifier: MPL-2.0
 *
 * Copyright (C) 2011-2015, BMW AG
 *
 * This file is part of GENIVI Project DLT - Diagnostic Log and Trace.
 *
 * This Source Code Form is subject to the terms of the
 * Mozilla Public License (MPL), v. 2.0.
 * If a copy of the MPL was not distributed with this file,
 * You can obtain one at http://mozilla.org/MPL/2.0/.
 *
 * For further information see http://www.genivi.org/.
 * @licence end@
 */

/*!
 * \author
 * Alexander Wenzel <alexander.aw.wenzel@bmw.de>
 * Markus Klein <Markus.Klein@esk.fraunhofer.de>
 * Mikko Rapeli <mikko.rapeli@bmw.de>
 *
 * \copyright Copyright © 2011-2015 BMW AG. \n
 * License MPL-2.0: Mozilla Public License version 2.0 http://mozilla.org/MPL/2.0/.
 *
 * \file dlt-daemon.c
 */

#include <netdb.h>
```

Preparing the code – Don't forget to ...

... add to the project root directory a LICENSE and COPYING file (new projects only)

[git.projects.genivi.org / dlt-daemon.git / tree](http://git.projects.genivi.org/dlt-daemon.git/tree)

[summary](#) | [shortlog](#) | [log](#) | [commit](#) | [commitdiff](#) | [tree](#)
[snapshot](#)

Added programme to test repeated calls of dlt_init and dlt_free

```

-rw-r--r-- 127 .gitignore blob | history | raw
-rw-r--r-- 555 AUTHORS blob | history | raw
-rw-r--r-- 8240 CMakeLists.txt blob | history | raw
-rw-r--r-- 460 COPYING blob | history | raw
-rw-r--r-- 5069 INSTALL blob | history | raw
-rw-r--r-- 16726 LICENSE blob | history | raw
-rw-r--r-- 2321 README blob | history | raw
-rw-r--r-- 24000 ReleaseNotes.txt blob | history | raw
-rw-r--r-- 209 automotive-dlt-c++.pc.in blob | history | raw
-rw-r--r-- 789 automotive-dlt.pc.in blob | history | raw
-rw-r--r-- 3694 automotive-dlt.spec.in blob | history | raw
drwxr-xr-x - cmake tree | history
-rw-r--r-- 426 distfiles blob | history | raw
drwxr-xr-x - doc tree | history
drwxr-xr-x - examples tree | history
drwxr-xr-x - gtest-1.7.0 tree | history
drwxr-xr-x - include tree | history
drwxr-xr-x - src tree | history
drwxr-xr-x - systemd tree | history

```

```

# @licence make begin@
# SPDX license identifier: MPL-2.0
#
# Copyright (C) 2011-2015, BMW AG
#
# This file is part of GENIVI Project DLT - Diagnostic Log and Trace.
#
# This Source Code Form is subject to the terms of the
# Mozilla Public License (MPL), v. 2.0.
# If a copy of the MPL was not distributed with this file,
# You can obtain one at http://mozilla.org/MPL/2.0/.
#
# For further information see http://www.genivi.org/.
# @licence end@

```

Mozilla Public License Version 2.0

1. Definitions

1.1. "Contributor"

means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. "Contributor Version"

means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"

means Covered Software of a particular Contributor.

1.4. "Covered Software"

means Source Code Form to which the initial Contributor has attached



GENIVI Public License Policy

- GENIVI approved licenses (green, yellow, red)
- GENIVI Default Licenses
- License headers files for all GENIVI approved licenses for code and documentation files

Public Policy for GENIVI Licensing and Copyright Version 2.0

Content	
1	Introduction..... 2
1.1	Scope..... 2
1.2	Definitions..... 2
1.3	Disclaimer..... 2
2	License Classifications..... 3
2.1	Green-light Licenses..... 3
2.1.1	Green-light Licenses for Code Artifacts..... 3
2.1.2	Green-light Licenses for Documentation Artifacts..... 5
2.2	Red-light Licenses..... 5
2.2.1	Red-light Licenses for Code Artifacts..... 5
2.3	Orange-light Licenses..... 6
2.3.1	Orange-light Licenses for Code Artifacts..... 6
2.3.2	Orange-light Licenses for Documentation Artifacts..... 8
3	Licenses and Copyright for GENIVI Hosted Code..... 9
3.1	Default License..... 9
3.1.1	Default License for Code Artifacts..... 9
3.1.2	Default License for Documentation Artifacts..... 9
3.2	Exceptional Licenses..... 9
3.2.1	Exceptional Licenses for Code Artifacts..... 9
3.2.2	Exceptional Licenses for Documentation Artifacts..... 10
3.3	Copyright..... 10
3.4	License Header Templates..... 12
3.4.1	License Header Templates for Code Artifacts..... 12
3.4.2	License Header Templates for Documentation Artifacts..... 14

V2.0 is now available at

https://collab.genivi.org/wiki/download/attachments/7710652/Public_Policy_for_GENIVI_Licensing_and_Copyright_version_2+0.pdf

Preparing the submission email

[audio-manager] [PATCH] Porting Pulse Routing Interface from AM v1.x to AM v3.0

Holger Behrens [holger.behrens at windriver.com](mailto:holger.behrens@windriver.com)

Mon Oct 20 12:53:54 UTC 2014

- Previous message: [\[audio-manager\] \[PATCH\] Porting Pulse Control Interface from AM v1.x to AM v3.0](#)
- Next message: [\[audio-manager\] \[PATCH\] AudioManager Proof of Concept](#)
- Messages sorted by: [\[date \]](#) [\[thread \]](#) [\[subject \]](#) [\[author \]](#)

From: Adrian Scarlet <[adrian.scarlat at windriver.com](mailto:adrian.scarlat@windriver.com)>

This Routing Interface is needed for any application that will be developed on top of AM and will use PulseAudio Sound Server to control the sources and sinks present on the system. It must be loaded by AM;

The interface can be built by supplying cmake with the `-DWITH_PULSE_ROUTING_PLUGIN=ON`;
After building one configuration file will be available:

1. `libPluginRoutingInterfacePULSE.conf` - configuration file for Pulse Routing Plugin; it will be loaded at runtime by the Pulse Routing Interface;

Changed files:

`CMakeLists.txt`

Added new folders:

`PluginRoutingInterfacePulse/`

Added new files:

`PluginRoutingInterfacePulse/CMakeLists.txt`
`PluginRoutingInterfacePulse/README`
`PluginRoutingInterfacePulse/data/libPluginRoutingInterfacePULSE.conf`
`PluginRoutingInterfacePulse/include/RoutingSenderMainloopPULSE.h`
`PluginRoutingInterfacePulse/include/RoutingSenderPULSE.h`
`PluginRoutingInterfacePulse/src/RoutingSenderMainloopPULSE.cpp`
`PluginRoutingInterfacePulse/src/RoutingSenderPULSE.cpp`

Signed-off-by: Adrian Scarlet <[adrian.scarlat at windriver.com](mailto:adrian.scarlat@windriver.com)>

<code>CMakeLists.txt</code>		24 +-
<code>PluginRoutingInterfacePulse/CMakeLists.txt</code>		103 +++

Preparing the submission email

Mandatory for all submissions to GENIVI:

"Signed-off-by: Firstname Lastname <firstname.lastname@example.com>"

By adding this signed-off-by line, you

- indicate your acceptance of the **GENIVI Certificate of Origin (COO)**

and, if you submit code on behalf of an organization, you

- declare that

"I am authorized to submit this contribution on behalf of the corporation holding the copyright"



GENIVI Certificate of Origin (COO)

By making a contribution to this project, I certify that:

- a) The contribution was created in whole or in part by me and I have the right to submit it under the open source license indicated in the file; or
- b) The contribution is based upon previous work that, to the best of my knowledge, is covered under an appropriate open source license and I have the right under that license to submit that work with modifications, whether created in whole or in part by me, under the same open source license (unless I am permitted to submit under a different license), as indicated in the file; or
- c) The contribution was provided directly to me by some other person who certified (a), (b) or (c) and I have not modified it.
- d) I understand and agree that this project and the contribution are public and that a record of the contribution (including all personal information I submit with it, including my sign-off) is maintained indefinitely and may be redistributed consistent with this project or the open source license(s) involved.



What happens if my organization has signed the GENIVI Contributor License Agreement?

Nothing*).

But:

- the submitter (i.e. the individual who sends the submission email) must be explicitly authorized by being listed on the CLA.
- GENIVI will reject unauthorized submissions

If unsure, contact the GENIVI Community Manager before submitting code to GENIVI

*) Companies with CLAs have additional options for contributing code to GENIVI which are not covered here.

Existing CLAs

- BMW
- Codethink
- Ford Motor Company
- LG Electronics
- Magneti Marelli
- PSA Peugeot Citroen
- Valeo

For authorized submitters see

<https://collab.genivi.org/wiki/display/genivi/Contributor+License+Agreements>

Dos and Don'ts in a Nutshell

- Use all available information, e.g. mailing lists/discussion boards to evaluate whether the contribution is of community interest (architecture, functionality) before you contribute
- Adhere to the project's coding, documentation, and licensing standards
- Work closely with the project maintainer and resolve issues quickly
- Provide information about 3rd party code included in the contribution
- Do not contribute patched upstream code, contribute the patch instead
- Do not contribute code with unknown/unclear origin or license, or without proper (internal) authorization

Guidelines for submitters

<http://www.genivi.org/contribute>

Q&A

THANK YOU!

BearingPoint

Claus-Peter Wiedemann
Senior Manager

BearingPoint
Erika-Mann-Str. 9
80636 München
Germany

T +49 89 54033 6367
F +49 89 54033 7940
M +49 172 2757415

www.bearingpoint.com

claus-peter.wiedemann@bearingpoint.com