



Collaboration in Automotive - The Eclipse Automotive Industry Working Group

Andreas Graf, Ralph Mueller, Ignacio Garro, Stefan Voget

► **To cite this version:**

Andreas Graf, Ralph Mueller, Ignacio Garro, Stefan Voget. Collaboration in Automotive - The Eclipse Automotive Industry Working Group. Embedded Real Time Software and Systems (ERTS2012), Feb 2012, Toulouse, France. hal-02263411

HAL Id: hal-02263411

<https://hal.archives-ouvertes.fr/hal-02263411>

Submitted on 4 Aug 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Collaboration in Automotive - The Eclipse Automotive Industry Working Group

Andreas Graf, Ralph Mueller, Ignacio Garro, Stefan Voget

Abstract The Automotive Industry is constantly introducing new and improved features based on advanced electronics and software. The use of these consumer electronics and software has required the automotive industry to define processes and tools that manage the interactions within an organization and within their extended supply chain. To address the growing complexity and time-to-market pressures, the automotive industry needs a common development tool-chain to support the development and testing of these new types of features. Today, many automotive companies use Eclipse to assist in the development of new features. However, a lack of integration between technology stacks, consistent use of tools through-out the supply chain and missing functionality has limited the effectiveness of create a complete development tool-chain. OEMs, 1st-tiers and consulting companies are founding the Eclipse Automotive Industry Working Group to coordinate the activities within companies.

1 Motivation

The increasing complexity of embedded systems presents ever-growing challenges for projects. The implementation of functions demand more than merely designing and implementing individual components. The interplay of several components is

Andreas Graf
Spokesperson of Eclipse Automotive Working Group, itemis GmbH, e-mail: andreas.graf@itemis.de

Ralph Mueller
Eclipse Foundation e-mail: ralph.mueller@eclipse.org

Ignacio Garro
Continental e-mail: ignacio.garro@continental-corporation.com

Stefan Voget
Continental e-mail: Stefan.Voget@continental-corporation.com

almost always necessary to provide the required functionality. The resulting communication networks and their relations can become complex and unclear. Considering that such systems are mostly implemented as a collaborative effort of several organizational units, this could become technically challenging. According to Conway, this also influences technical characteristics and processes. Additionally, a Product Line Approach (PLA) of systems development requires development, validation and maintenance of different software variants. Processes and development are hindered by gaps in the toolchain, missing functionalities and high cost of customizing the tool chain.

To address the growing complexity and time-to-market pressures, the automotive industry recognizes the potential of the Eclipse platform to provide a common development tool-chain to support the development and testing of these new types of features. Several companies are preparing the next step from company-specific tool platforms to an industry collaboration scenario. The expectation is to reduce cost for common functionality, increase quality and to provide better tool integration. This collaboration is acted out as the “Eclipse Automotive Industry Working Group”.

2 Eclipse Industry Working Groups

Eclipse IWGs are established to facilitate the collaboration between Eclipse Foundation Members. The collaboration should be intended to focus, promote and augment Eclipse technologies to meet the needs of specific industries. The collaboration can take the form of the development of Materials for use by a specific community and/or joint Marketing Programs to promote Eclipse in a certain industry. Unlike Eclipse open source projects that are open to participation by anyone, participation in an Eclipse IWG is open only to Eclipse Members.

All participants in IWGs must respect the agreements, policies and procedures of Eclipse. In particular, and without limiting the foregoing, at all times, participants in all Eclipse IWGs must conform to the Eclipse IP Policies and Antitrust Policies.

All IWGs must operate in an open, transparent and vendor neutral manner.

The automotive industry working group currently does not require any fees from members. It is however expected to actively contribute to the working packages of the group.

3 Scope

The Eclipse Automotive Industry Working Group (EAIWG) will initially work to define, implement and package an Eclipse-based development tool-chain that will become an industry platform for tools for designing advanced automotive electronic systems. The industry platform will include key open source components

from Eclipse but also define extension points that can be used to integrate value add tools from a commercial ecosystem or in-house development.

This also addresses the organizational requirements on such a tool chains, such as safety relevant issues. The Working Group has defined four work packages

3.1 Technology Recommendations

OEMs use standards such as AUTOSAR, Flexray or Genivi ¹ in the development of automotive software. It is a necessary condition for OEMs, that the tool chain supports these standards. We believe that tools based on Eclipse are an efficient way to reduce gaps in the tool chain. It can be very helpful if Eclipse-based tools can not only be used as a stand-alone tool but also as integrated components in a larger environment. The interoperability between the different Eclipse plug-ins from tool vendors can accelerate the development process.

Eclipse is a de-facto standard in the field of developing software development tools (for example ARTOP ²) and as integrated development environment for programming automotive software. Therefore, a unified Eclipse platform is to be agreed on by all participants in the EAIWG. For this reason, the future participation of tool manufacturers in the EAIWG is absolutely necessary. A single platform can reduce internal costs and accelerate cross-company development processes. OEMs will use the Eclipse platform developed by the Working Group as a strategic development environment.

Creating a standard platform for automotive tools will help companies to faster innovate without compromising quality. Collaborating with an open source community will allow us to share the costs and best practices for building new tools with other companies in the automotive industry. Members see this open innovation as the future for their software tools strategy.

The currently suggested components are (for details see www.eclipse.org):

- Eclipse SDK
- EMF Query SDK
- EMF SDO XSD SDK
- EMF Transaction SDK
- EMF Validation SDK
- GEF SDK
- GMF SDK
- MDT OCL SDK
- MDT UML2 SDK
- Orbit ([org.apache.commons.lang ,org.jdom](http://org.apache.commons.lang.org/jdom))
- Sphinx

¹ non-profit industry alliance committed to driving the broad adoption of an In-Vehicle Infotainment open-source development platform)

² AUTOSAR Tool Platform, an industry driven user group

- RMF
- Acceleo
- Antlr
- M2M ATL / MTL
- Papyrus
- EAST-ADL 2 experimentation
- Xtext
- CDT
- WST
- JDT
- BIRT
- DTP
- Nebula Widgets
- Mylyn
- M2M QVTO
- CLI, Log4J
- TCF
- P2
- Device Debugging
- Update Manager

3.2 Support of large Models

Models that exist in the automotive engineering space are quite huge. Complex AUTOSAR system descriptions can easily achieve the limit of current 32bit Eclipse systems for a number of reasons: The Eclipse model framework EMF optimizes space by using load-on-demand, but does not easily free unneeded model elements. In addition, AUTOSAR models make use of a lot of strings. And these are taking up quite some memory in Java.

However, within the Eclipse and open source domain, technologies to solve those problems are already available. The CDO (Connected data object) projects e.g. provides an alternative storage and management infrastructure for EMF that handles large and complex models and distributed workflows quite well. A port of AUTOSAR and Sphinx to make use of CDO or similar technologies would easily improve the support of Eclipse for large models in the automotive domain.

The work package aims at coordinating the activities, that are already happening at the partners' companies.

3.3 C/C++ Development Tooling (CDT) extensions

The work package deals with the possible improvements to be introduced in the CDT component of Eclipse in order to make its application to the embedded automotive environment more adequate.

The work package has detected a main action area in the definition of a standardized interface to provided extended information about the memory structure and content of executables: listing of sections and segments, variables with address and type, function address and size, etc. Right now, access to the binary artefacts of the compiler depends on a heterogenous tool chain with a lot of different tools. This complicated set of interfaces makes an integration into the user interface of the CDT more difficult. As such, a commonly defined API will improve the workflow. The approach could be compared to the standard API for accessing debuggers.

Another focus is the improvement of the toolchain management for embedded projects. This improvements shall ease the interchange of CDT projects between industry partners providing standardized support methods like standardized definition of the toolchain with its dependencies, installation of the toolchain or compliance to the defined toolchain. These currently rely heavily on project specific conventions, making the interchange very difficult.

Further collected interest topics include the definition of meta-information for traceability and the extension of the CDT editor to support structogram views.

3.4 Functional Safety Relevant Development Process and Environment

In the automotive domain a new safety related standard - the ISO26262 - is currently being finalized. The standard is an adaptation of the functional safety base norm IEC 61508 for specific needs of related E/E systems in road vehicles. This adaptation applies to all activities during the safety lifecycle of safety-related systems related with electrical, electronic, and software elements that provide safety-related functions.

It is currently not clear how the development and models necessary for safety documentation and analysis can and should be integrated in order to minimize modeling effort, to keep consistency between artifacts and to enable effective reusability and change management. Methods which allow automotive products developed according to the ISO26262 have to be applicable to such an integrated model.

Current research and standardization activities are on the way to develop methods, meta-models and DSLs to enable integrated modeling through the whole development life cycle. Important activities are driven by the EAST-ADL association and by AUTOSAR. The EAST-ADL provides a DSL based on abstraction levels from functional modeling through detailed HW- and SW-design. AUTOSAR supplements this DSL by a meta-model for software configuration of electronic control units.

Also several activities to develop associated tool platforms already exist.

- ARTOP is an industry driven user group to develop a tool platform for the AUTOSAR meta-model.
- Sphinx is an Eclipse project that develops necessary extensions of EMF based platform support to enable ARTOP.

A similar project for providing a technology reference platform for EAST-ADL is missing so far. To enable and push the realization of this technology platform is the first goal of this work-package in the automotive industry working group.

SAFE (Safe Automotive software architecture) is an ITEA2 public funded project that provides methods, e.g. for efficient capturing of safety goals and requirements as well as for safety evaluation or conformance testing. These will be enhanced, in order to benefit from the integrated model. To allow evaluation of the methods within significant industrial case studies, a technology reference platform will be defined and extended with a set of appropriate plug-Ins.

So far, no Eclipse project exists to state as a home for the upcoming plug-Ins. To define appropriate projects within Eclipse and to coordinate with the existing research and standardization projects is the second goal of this work-package in the automotive industry working group.

Focus of this WP shall be to analyze which functions should have an Eclipse based open source tool-chain in a safety development environment. Furthermore, this WP shall act as an interface between Eclipse and other automotive research projects (e.g. forwarding requirements). The WP will work in providing requirements on a tool platform and defining a list of tools to be implemented.

4 Summary

Eclipse already plays a key role in the automotive development department. This is reflected by the fact that companies like BMW, Robert Bosch GmbH and Continental AG have become members of the Eclipse Foundation. As in other industries, the companies shift their focus to stronger collaboration (as shown by AUTOSAR, common tool platforms like Artop). The Eclipse Automotive Industry Working Group will allow for the participating companies to leverage the benefits of collaboration, improving the tool chain while reducing cost.

5 References

- ARTOP, www.artop.org
- AUTOSAR, www.autosar.org
- EAST-ADL, www.east-adl.info
- Eclipse, www.eclipse.org

- Eclipse Auto IWG, wiki.eclipse.org/Auto_IWG
- ITEA2, www.itea2.org
- Genivi, www.genivi.org
- SAFE, www.safe-project.eu
- Sphinx, www.eclipse.org/sphinx