Polarsys
towards Long-Term Availability of Engineering Tools for Embedded Systems

Gaël BLONDELLE
OBEO
7, avenue Didier Daurat – BP 30004 – 31700 Blagnac - France
Phone +33 672 120 226 gael.blondelle@obeo.fr

Co-authors: Paul ARBERET (CNES), Alain ROSSIGNOL (Astrium), Björn LUNDELL (University of Skövde), Christian LABEZIN (Xipp), Romain BERRENDONNER (Adacore), Pierre GAUFILLET (Airbus), Raphael FAUDOU (Atos Origin), Benoît LANGLOIS (Thales), Luc MAISONOBE (CS), Pierre MORO (EADS Space Transportation), Jorge RODRIGUEZ (Indra), José Manuel PUERTA PEÑA (TCPSI), Eric BONNAFOUS (CS), Ralph MUELLER (Eclipse Foundation)

1. ABREVIATIONS

- CCB Change Controle Board
- OSS Open Source Software: Solutions
- OPEES Open Platform for the Engineering of Embedded Software
- DOD Department Of Defense
- ITEA2 A strategic pan-European programme for advanced pre-competitive R&D in Software-intensive Systems and Services. ITEA2 is one of the main Eureka cluster programmes.
- IWG ECLIPSE Industrial Working Group
- IT Information Technology
- LTS Long Term Support
- SDL Specification and Description Language

2. INTRODUCTION

The Polarsys Industrial Working Group (http://www.polarsys.org) addresses specific issues of industrial users who develop and maintain Embedded Systems:

- A satellite needs to be supported for more than 25 years.
- An aircraft flight control system must be "certified" according to safety standards, including the tooling used to produce the software.

During the whole life cycle of the product, system architects and developers of embedded systems need maturity assessment and long lasting support for their tooling.
These concerns are also relevant in other safety-regulated environments like railway or energy, and for other domains with long life products like telecommunication or healthcare.

As stated in [1] “Software in the embedded systems domain needs to be maintained for a very long time”. As a consequence very strained situations may build up if the commercial vendor of adopted proprietary software leaves the market. A general perception is that such problem can be avoided with Open Source software. However, long-term sustainability of Open Source software in this domain requires contributions from volunteers to be enhanced with resources from large commercial players.

In 2009, the OPEES ITEA2 project, which stands for Open Platform for the Engineering of Embedded Systems, was created to tackle these specific issues and propose both technological and organizational solutions.
Convinced that Open Source is a good way to ensure long-term support availability, the OPEES project members decided in 2011 to join forces with the Eclipse Foundation and to create the Polarsys Industrial Working Group. This group, leaded by industrial users, fosters the development of the ecosystem, and provides services like: a catalog of components with maturity assessment; an infrastructure to host long term support development, build and test; and Change Control Boards where industrial users share needs and drive the development of Open Source components.

The OPEES ITEA2 project is a good foundation to bootstrap such a community as it already involves almost 30 partners, comprising industrial users, academics and service providers, covering various domains like aerospace and telecom, and with members from all around Europe in France, Spain, Belgium, Norway and Sweden.

This paper presents the objectives and rational of the OPEES project, why we decided to create Polarsys as an Industrial Working Group in the Eclipse Foundation, and what services are being implemented in this context.

This work is funded by the OPEES ITEA 2 project.

3. OBJECTIVES

The mission statement of OPEES is "to settle a community and build the necessary means and enablers to ensure long-term availability of **innovative engineering technologies in the domain of dependable / critical software-intensive embedded systems**". Therefore its main goals are:

1) To build an ecosystem in the open source frame with the relevant business models in order to ensure long term availability of OPEES components and engineering tools and to favour a sustainable development of an embedded software industry aligned with the industrial strategic intents.
2) To assess, define and experiment methods, processes and guidelines ensuring the required **level of quality** for OPEES components and engineering tools in order to be able to use them in "certified" development processes; this includes interoperability specifications to allow the construction of integrated solutions and definition of inspection, V&V and qualification best practices.
3) To leverage early results and actors at the national and international levels to give a **Worldwide dimension** to the initiative.

4. USER'S NEEDS

OPEES fills a gap in technology adoption of System/Software Engineering tools and components for embedded systems: It enables a **larger and quicker adoption of the technology** that has been developed under public and private funding as it allows to meet the operational needs of users who do want to adopt these new technologies.

Software is recognized as the key driver for innovation and business development in many domains and embedded software and systems are recognized as a fast growing part of it (+14% yearly as stated in [3]).

One of the characteristics of dependable Embedded Systems is that they have a **very long maintenance phase**. In some industries, embedded systems must be maintained 20+ years (Energy), 30+ years (Automotive) and 50+ years (Aerospace).
Therefore, the **long-term availability of tools and components** is a mandatory requirement for most of the industrial users. When considering new technology adoption, users evaluate the continuity of service on the long run. This includes all the features that are requested to use the technology on a daily basis by industrial organizations: the product itself, the requested documentation, maintenance services, training, support services and customisation.

Moreover, when we consider **critical-software engineering tools**, the market is even broken into smaller "niche" areas whereas higher stability of tools is required. The resulting market evolution shows the following trends:

- From the very beginning, the tools needed to develop the software at the requested dependability/productivity levels were not available on the market place, so industrial companies in Europe developed their own solutions (make) in different fields: automatic code generation, process management, test automation, etc.
- Then there was a progressive reorientation of these companies on their core competences that led to the move from "make" to "buy". Some internal tools were transferred to software editors that industrialized them (e.g.: SCADE, RTRT, Reqtify, etc.).
- Because of the massive investment required for those engineering tools for dependable systems (e.g.: DO178B qualified code generator) and due to the very small distribution scale, some products are acquired by new companies several times, whereas other almost disappear (e.g. ObjectGeode SDL framework).
- Some other tools remain at the stage of "academic" tools for similar reasons like the fact that they are too specialized and/or too costly for a profitable business in a very small market.

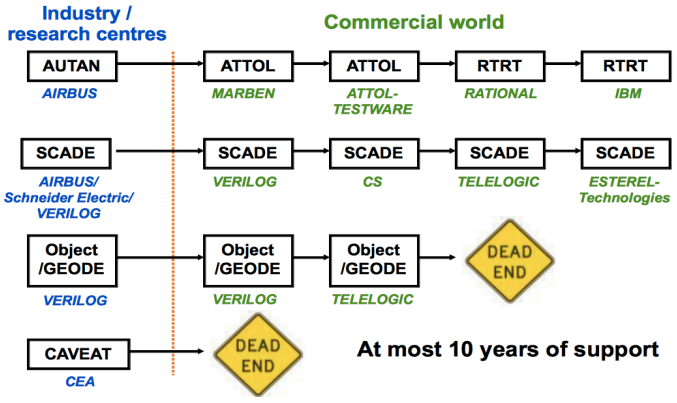


Fig. 1 Proprietary software : Dead End road ?

"For long-term availability of proprietary software the issue of support is critical". Results of the study by Lundell et al. suggest that, in this domain at least, the experience of support from commercial vendors of proprietary software can be very poor; and that - in practice - organizations are left without a viable support option anyway.
On the other hand, interviewees indicate that it is their experience that there are often consultants available that can provide support for Open Source software"[1].

This work is funded by the OPEES ITEA 2 project.

Another issue is the cost of the engineering tools. Most of the software is developed with partners and sub-contractors in an extended enterprise mode. Availability of development tools is not problem-free in this context (deployment costs, number of licenses, etc). Most of the time, the price of software tools (license, maintenance, etc) has nothing to do with their real cost which needs to be analysed from a Total Cost of Ownership / Operation perspective (TCO). In complex environments, the costs associated to scalability, upgrades, replacement, migration and decommission are higher than the licensing related costs.

So far, no business model based on proprietary software meet the needs of Embedded Systems developers.

5. LONG TERM SUPPORT

As mentioned in the ITEA report on Open Source Software (OSS)[4], OSS "is an important new development and an interesting option for software-intensive systems".
On the other side of Atlantic, the DOD has also stated that OSS will be supported with the next generation of weapon systems with a "transversal strategy across the different services":

- "Our objective is not to develop expensive software for the weapon systems or paying twice for it, rather than the opposite"
- "We will pay to get in but not to get out (the software)"
- "The only way for the survival"[5]

Additionally, as "many embedded systems require maintenance for many years, OSS can be an appropriate way to address risks related to lock-in and long-term maintenance of commodity software". [6]

Indeed, open source may bring answers to many issues and more specifically:

- Open Source projects contribute to standardization, a need for software industry as for others, although Louis Gerstner, IBM's CEO in the 1990s, noticed: "This is the only industry where competitors don't regularly agree on standards to enable greater value for customers."[7]
- Open source does not ease the access of competitors to the technology no more than proprietary software … (first cost item for a SW vendor is marketing effort)
- Open Source brings open formats, which allows easier migration between tools or between versions.
- Open source may improve the common use of the tools amongst the stakeholders of a same domain and therefore may allow better deployment in extended enterprise context
- Open Source clarifies the Intellectual Property implication of using the software in a transparent and open way, and gives the capability to lower risks related to IP in the context of long-term support.
- Open source avoids single source dependency: the knowledge is available

Open source allows building a **common place technology for 'commodities'**, which don't bring any competitive differentiation to the final users because they are used by all (e.g.: UML editors), although it is important to notice that it is a dynamic situation: what is specific today will becomes commodities tomorrow…

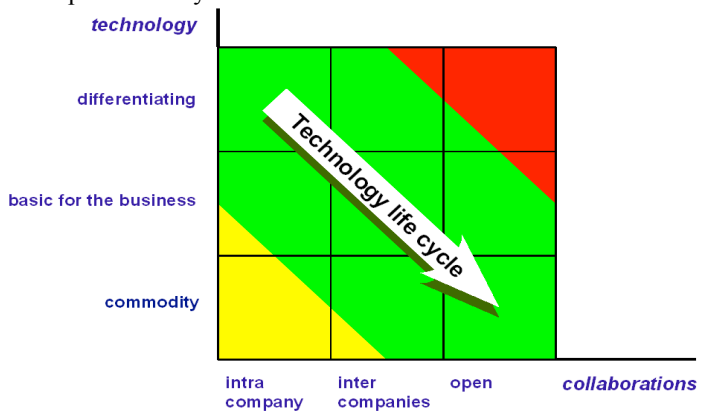


Fig. 2 Commodification of Industrial Software (Based on [6])

Based on van der Linden et al. [6], an organization faces the challenge of identifying how to combine the vertical movement with the move, for any software, from the left to the right in the commodification diagram, from in-house to (open) collaborations. Consequently, effective software development is characterized by the middle (green colored) area, from top left to bottom right.

A new software edition landscape has to appear, and it will not exist without support from industrial end users and public incentives. It must be:

- Realistic = not "make" again, and loose illusion that "buy" may rise spontaneously in every domain
- Funded = to bootstrap the virtuous circle, needed also to preserve /develop high added value jobs in Europe
- Organized = need for an international eco-system

It is noteworthy than in less than a decade, Open Source solutions have grown in the entire IT stack from OS to applications. Similarly, we observe that Open Source is conquering new parts of the Embedded Systems market, and we can expect that during the next decade, Open Source will be used in the whole Embedded Systems Stack.

This work is funded by the OPEES ITEA 2 project.

In this domain, the aim is to **federate contributors** scattered all over the world, together with software editors who develop their tools on top of the generic components shared as OSS, with a light structure (marketing/communication) to develop the community.

6. OPEN SOURCE LONG TERM SUPPORT AS AN ENABLER FOR A USER CENTRIC ECOSYSTEM

On of the main weakness of the usual proprietary software ecosystem is that the software users often feel that they have to adapt their methodology to the tools instead of adapting the tools to the methodology.
In OPEES, we consider the adoption of OSS as an enabler for ensuring both Long Term Availability and better adaptability of software to the user needs.
This situation is illustrated in Figure 3: At most 20% of Features Requests are usually taken into account by proprietary software vendors as they implement tradeoffs between the needs of their customer base. This number is commonly accepted by the proprietary software vendors.

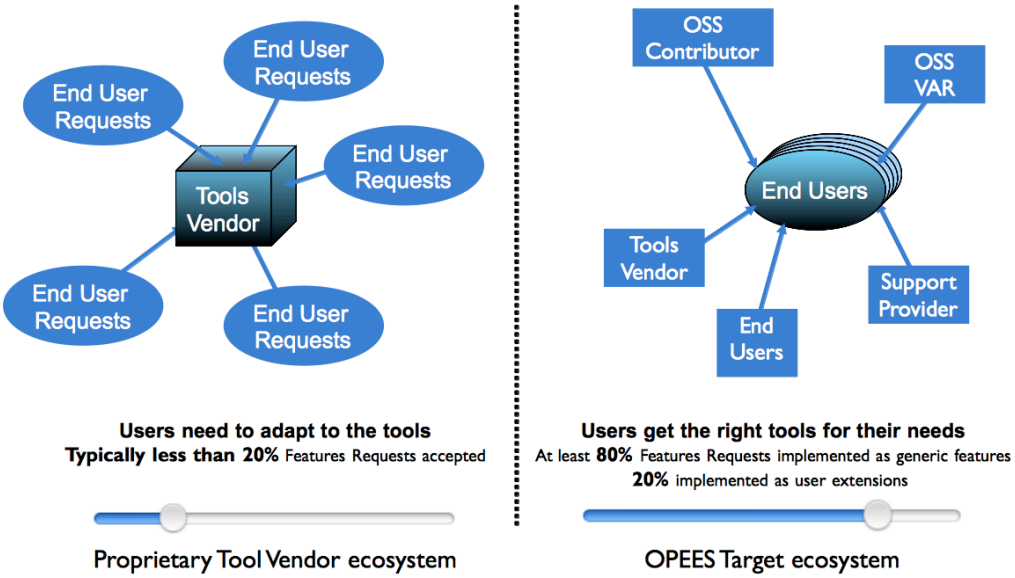


Fig. 3 Focus on End-User needs

In the ecosystem targeted by OPEES, we put the software users at the center of the ecosystem, and tools vendors, system integrators, OSS contributors or support providers collaborate with these industrial users in order to build the tools that best fulfill the user needs.
The Objective is that 80% of the features should be available as generic features in OSS, and that the ecosystem will be able to create the 20% features specific to a domain or as user extensions to the OSS projects.
Such a user centric ecosystem enables stronger collaboration between companies in the same domain, and also between companies in different domains with similar needs. As an example, system safety analysis relies on different standards in aerospace or automotive industry, respectively ARP4761 or ISO26262. But the tooling used to conduct such safety analysis could be largely common to the two domains.
In order to control such behaviors and align it with user needs, POLARSYS is implementing Change Control Boards (described later), which are the governance bodies where end users share needs and can drive the development of the Open Source projects.

7. OPTIONS FOR THE IMPLEMENTATION OF THE OPEES LONG TERM ENTITY

During the creation process of the OPEES legal entity, three models have been explored:

- Macro-organization: In this model, the OPEES entity is in charge of a large portion of the tasks needed to maintain and productize software on long periods of time. Users express their needs to the OPEES entity which defines roadmaps, passes competitive bids open to approved providers to implement the roadmap, and productizes the results of the developments to make them available to the users.
- Federative: In this model, the OPEES structure acts as a software editor and a single point of contact for the OPEES Products towards the Users. However, in order to get the appropriate level of resources and manpower, the OPEES structure enters into one-to-one arrangement with Providers, each Provider becoming the reference partner for one or several specific Products. To some extent, this works in a federative way.
- Micro-entity: In this operational mode, see Figure 4, the OPEES entity remains as small as possible and focuses on the development and the regulation of the OPEES ecosystem.

This work is funded by the OPEES ITEA 2 project.

OPEES project members evaluated that the Macro-entity model would be hard to accept by industrial users who would have to move a part of their purchase activity inside this entity. Additionally, this entity would necessitate high operational costs.
The Federative model has not been considered as a good option because OPEES subcontractor would have to change their business model in order to do most of their business in the domain with this OPEES entity. Additionally, it may be considered as a coalition.
Finally, project members figured out that in order to effectively start a legal entity, this entity had to be small and efficient and they selected the Micro-entity model.

In this Micro-entity operational mode (see Figure 4), the OPEES entity remains as small as possible. It implements a subsidiarity approach where it does not carry out technical activities in place of Providers but rather fosters the emergence of sustainability of Providers conforming to OPEES;
Users continue to purchase the Products to Providers; the OPEES entity mainly acts as a regulatory instance of the relationships between:

- Users and Providers;
- Providers and the community;
- Users and the community.

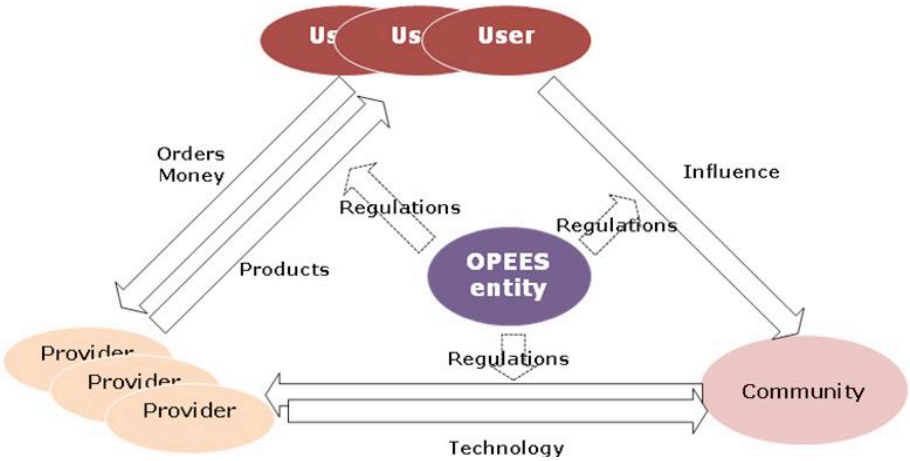


Fig. 4 OPEES as a micro-entity

8. POLARSYS: OSS LTS IMPLMENTATION IN COLLABORATION WITH THE ECLIPSE FOUNDATION

Once the OPEES members decided to implement the OPEES legal entity as a micro-entity, two scenarios were evaluated:

- Create an independent OPEES legal entity: A new Open Source Software (OSS) consortium focused on providing Long Term Support for tools used to develop Embedded and Critical Systems.
- Create a new group of interest in an existing OSS foundation, and especially create an Industrial Working Group (IWG) inside Eclipse.

After a careful evaluation, OPEES members unanimously[1] chose to create the OPEES legal entity as an Eclipse IWG, and selected Polarsys[2] as the name for this Industrial Working Group.

Among the main arguments, we can select the three following ones:

- It is easier to claim we build a sustainable entity to ensure long-term availability of software by joining a Foundation that has proven to be sustainable for ten years.
- The Eclipse Foundation started two years ago to define industry driven working groups, and other working groups are being installed inside Eclipse to deal with Long Term Support, for example in the domain of Information Systems.
- Finally the Eclipse Foundation is a well-recognized OSS organization that implements one of the best IP due diligences process in the OSS world, and provides a very good collaboration infrastructure. By joining the Eclipse Foundation, we estimate to shorten the time before we will be fully operational from at least 24 months to almost 12 months.

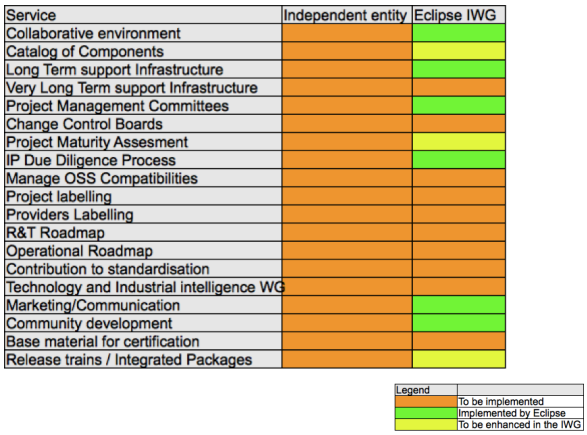
[1] At the unanimity of the expressed votes.
[2] http://www.polarsys.org

This work is funded by the OPEES ITEA 2 project.

9. LIST OF SERVICES TO SUPPORT

The members of the OPEES projects defined a list of services to be provided by the Polarsys IWG. Figure 5 describes this list of services and indicates:

- Services that are already provided by the Eclipse Foundation
- Services that must be implemented in the context of the Industrial Working Group
- Existing services that have to be enhanced in order to fulfill OPEES member needs



Service	Independent entity	Eclipse IWG
Collaborative environment		
Catalog of Components		
Long Term support Infrastructure		
Very Long Term support Infrastructure		
Project Management Committees		
Change Control Boards		
Project Maturity Assesment		
IP Due Diligence Process		
Manage OSS Compatibilities		
Project labelling		
Providers Labelling		
R&T Roadmap		
Operational Roadmap		
Contribution to standardisation		
Technology and Industrial intelligence WG		
Marketing/Communication		
Community development		
Base material for certification		
Release trains / Integrated Packages		

Legend	
	To be implemented
	Implemented by Eclipse
	To be enhanced in the IWG

Fig. 5 List of OPEES Services

We detail the most important services below.

- **Catalogue of Components**

  The catalogue defines a part of the Polarsys identity, given that the consortium is represented first by its members, second by the services it provides, and last by the components it references.

  Such a catalogue will gain value as it references more components and criteria:

  - Maturity and more generally life cycle of components and tools
  - Links between components in term of inclusion, compatibility
  - Availability of a quality and qualification kit

  The projects and components catalogue will be built in priority, by extending the structure of the existing Eclipse catalog of components in order to include maturity and quality assessment.

- **Long Term Support Infrastructure**

  The Eclipse Foundation will set up a common, vendor neutral, build and test infrastructure, for the Long Term Support (LTS) of Eclipse components.

  The objective of this infrastructure will be to host the support activities after the 9 months of community support provided by the Eclipse Community. Indeed, the Eclipse community produces a release train every year in June, and two maintenance releases, the Service Release 1 (SR1) in September, and the SR2 in February. After SR2, the whole community works on producing the next release train.

  The Long Term support infrastructure will be the place where Users and Service Providers will work in order to provide Long Term Support for several years (up to ten years). This infrastructure will be shared between the Polarsys community, focused on Embedded systems, and the Informations System community.

  But the Polarsys community will also implement additional services for the Very Long Term Support (VLTS) (up to 20, 30 or even 40 years). It involves the necessity to freeze a software version, to archive it, and to be able to restore it years after in order to fix a bug and rebuild the software.

  This service is complementary to other approaches like migration from one product or component to a newer one, but the migration aspect can be seen as a specific type of project, and supported as part of the Toolset centric or Consulting Professional services referenced by Polarsys.

- **Change Control Boards**

  A change control board should be created for each Polarsys project or product to be a place of exchange and decision about the evolution of the component.

  Polasys should provide a “Change Control Board” process framework with two levels:

  - Product / Group of components “Change Control Board”
  - Component “Change Control Board”

  Such a two-level process mimics what is available in Eclipse with the Top-Level projects and Components governance.

  A product with an established Change Control Board (CCB) is a full-featured Polarsys component.

  Typically, the product change control board should be composed of the end-users the most representative or the most concerned by the evolution of the components, and also include the service providers the most involved in this evolution.

  The CCB will complement the usual OSS project governance with Project Management Committees in which project committers decide about the future of the project.

  In the CCB, Polarsys members, and mainly the component end-users, will be able to influence a part of the component development and evolution.

  Label Support

This work is funded by the OPEES ITEA 2 project.

One of the unique characteristics of Polarsys will be the definition of a label for service providers.
This label will ensure that the services providers are committed to a long-term investment in the technologies they support, and they are skilled both in the component they want to support, and also in the LTS and VLTS processes.

- **Component maturity assessment**
  The maturity assessment of a component or project is another corner stone of Polarsys. The objective is to check that the component is ready for LTS and VLTS processes. For example, one of the constraints will be to use the common LTS build and test infrastructure for building, testing and releasing services releases for the component.
  Base material for certification
  Even if certification is always a domain specific process, it is mainly a documentation-based process.
  Therefore, a way to get better and lighter certification processes for Polarsys tools is to define for each product type and each domain, a certification process framework. This certification framework will leverage the experience of Polarsys members from previous certifications (either of the same product, or in the same domain).
  This certification framework, accessible to Polarsys labeled providers who lead a Polarsys project, and jointly defined with the Polarsys members, would enable the creation of quicker and lighter certification process for a tool in a new domain.

10. SUPPORTED COMPONENTS

A first set of Polarsys tools has been defined and will be assessed according to the OPEES objectives in the context of the ITEA project.
This list includes Topcased, GeneAuto, MOSKItt4ME, EGF, Polychrony, Frama-C, MDT/Papyrus, Acceleo, GPM and ATL.

This work is funded by the OPEES ITEA 2 project.

11. POLARSYS FOR LTS

The OPEES ITEA2 project, driven by a group of industrial users that want to use open source as a mean to ensure long-term support, has entered the implementation phase of a sustainable entity: the Polarsys Industrial Working Group, inside the Eclipse Foundation.

As a small and open organization, Polarsys intends to provide infrastructure and processes to foster the resulting ecosystem around engineering tools for dependable/critical software-intensive embedded systems.

In a context of software technology volatility, Polarsys will manage the changes and accommodate the technology trends by ensuring either long-term availability of tools or long-term availability of the services provided by the tools by tackling migration issues.

Polarsys is one of the first efforts to focus on open source and embedded software for critical systems and is open to welcome new members:

- industry actors willing to share engineering tools and techniques and to ensure long-term availability of these tools;
- software or service suppliers wanting to join this ecosystem
- academics who can propose application and maturation of their research tools & services

In 2012, both the OPEES project and the Polarsys IWG will operate at the same time. It will enable us to populate the Polarsys infrastructure with existing Open Source Software managed by OPEES members. But more importantly, the OPEES project members will experiment the services.
Finally, by the end of 2012, we expect to deliver worldwide the first releases of Polarsys software, with maturity assessment, software qualification kits, and ready for LTS.

Join the community! Join the discussion at http://www.polarsys.org!

12. BIBLIOGRAPHY

- [1] Lundell, B., Lings, B. and Syberfeldt, A. (2011) Practitioner perceptions of Open Source software in the embedded systems area, Journal of Systems and Software, Vol. 84, N°9, pp. 1540-1549
- [2] http://www.opees.org
- [3] http://www.artemis-ju.eu/
- [4] ITEA report on Open Source Software, January 2004
- [5] US DoD Conference on Open Technology Development "Open Technology – Realizing the Vision" March 2007
- [6] Van der Linden, F., Lundell, B. and Marttiin, P. (2009) Commodification of Industrial Software: A Case for Open Source, IEEE Software, vol. 26, no. 4, pp. 77-83, 2009
- [7] An Open Secret, The economist (0ct.22, 2005), quoted in ACM communications October 2005

This work is funded by the OPEES ITEA 2 project.