



HAL
open science

Software business models for AUTOSAR automotive world standard

Thomas Soulier

► **To cite this version:**

Thomas Soulier. Software business models for AUTOSAR automotive world standard. Conference ERTS'06, Jan 2006, Toulouse, France. hal-02270491

HAL Id: hal-02270491

<https://hal.science/hal-02270491>

Submitted on 25 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.

Software business models for AUTOSAR automotive world standard (V1.00)

A. Thomas Soulier¹

1: Siemens VDO AG, 1 av P Ourliac 31036 Toulouse Cedex - France B.P. 1149

Abstract:

AUTOSAR (AUTomotive Open Standard ARchitecture) as a promising initiative, will establish an open standard for automotive E/E architecture. AUTOSAR compliant Basic software will ensure independency of the application software from the underlying hardware while allowing modularity as well as reusability. From existing black box electronic control module business, three distinct layers, hardware, basic software and application software supported by robust toolkit may develop different kinds of business activities.

From the drawing of established business models landscape, we will determine the strategic options that could be offered for a Tier 1 supplier. And this will finally end up to innovative business model proposals and a migration plan in accordance with customer needs and supplying value through systems engineering activities, software products and professional services.

Keywords: Business Models, AUTOSAR, Open Source.

1 Introduction

Siemens VDO is currently selling embedded electronic systems and using the traditional HW business model. But tomorrow hardware will be clearly separated from software (SW) with the arrival of the AUTOSAR standard. So what can be learned from mature industries which have faced this kind of situation for decades and have continuously adopted a new business model in a rapid evolving environment. Moreover they have recently been confronted with the swell of revolution coming from Open Source software companies. We will confront existing business models and associated strategic options to the upcoming situation in the automotive E/E architecture business to evaluate innovative business model proposals.

2 Business Models landscape :

Best practices and experiences outside of automotive

2.1 Business model definition

Through the landscape of business model definitions, we can distinguish some common points and general characteristics. A business model is a method or approach towards a defined market aiming at company survival. Defining a business model can be summarized to the answer to two basic questions :

- How to develop a sustainable and profitable turnover?
- Where to position the company in the value chain?

Company's objectives consist in getting benefits from sales of product and/or service in ever evolving environment. In order to fulfil this goal companies try to gain market share and extend their potential market which becomes more and more difficult to achieve.

Clearly automotive actors will not accept radical changes in the traditional business model. That is why looking at other comparable business activities and especially software industry could facilitate business model definition. Thus this should drive a better market acceptance when proposing software products and services as a Tier one supplier.

2.2 Commercial software definition^[2]

With commercial software customers do not have access to the source code which is closed and most likely protected by copyrights and/or patents. Moreover commercial software companies do not grant the user any other rights beyond the use of the software package.

We can consider three main business models in the commercial software industry:

- Software **product** company
- Software **service** company
- **Hybrid** solutions company.

2.2.1 Software **product** companies

In this category, products are highly standardized and companies excel in acquiring and leveraging **generic** product knowledge. Therefore architecture level technology integration is important as well as product complementarity.

Moreover digital replication ensures minimal costs to mass-produce. As a consequence companies can benefit from large economies of scale and other efficiencies such as automation in design.

To support and target a large customer base, companies have large marketing, support and maintenance departments.

Microsoft business model

The objective of Microsoft is to become and remain as the market leader through high volume sales and de facto technical standards settings that "lock in" customers. Their software applications and databases only work on a particular operating system or hardware platform which ensures long-term relationships and a monopoly situation difficult to overturn.

The main problem for product company remains in high fluctuation effect. Indeed if the economy turns bad, the product does not fit any more the customers needs or become commodities by value dilution.

To face this problem, companies are progressively starting to sell more and more maintenance services (special product enhancements, regular upgrades sold under long-term contracts).

Key: Need to understand **general** customers need

Pros & Cons

- (+)Potential for higher profitability (gross margins up to 99%)
- (+)Low production costs
- (-)Require higher investments
- (-)Highly influenced by business fluctuations
- (-)Hard to write successful applications (killer apps)
- (-)SW can become commodities
- (-)Subject to discretionary IT spending

2.2.2 Software **service** companies

In this company category, all their revenues come from project-driven relationships with custom-built systems even if they rely heavily on reusing partial products for different types of applications.

High service revenues are often linked to the degree of complexity of the products. IT companies have adopted service-based offerings with their products because they are often too complex to package as "off-the-shelf" offerings.

Key: Need to understand **specific** customers need

Pros & Cons

- (+)Sustainable recurring revenues even in "bad" economic times
- (+)More room for growth
- (-)Higher marginal costs
- (-)Slower growth potential
- (-)Lower profit margin potential
- (-)More labor intensive than products
- (-)Hard to scale without adding people

2.2.3 Hybrid solution companies : the ideal solution?

The main advantages of those companies remains in both generating scale economies from sales of standardized products and more predictable recurring revenues like service companies.

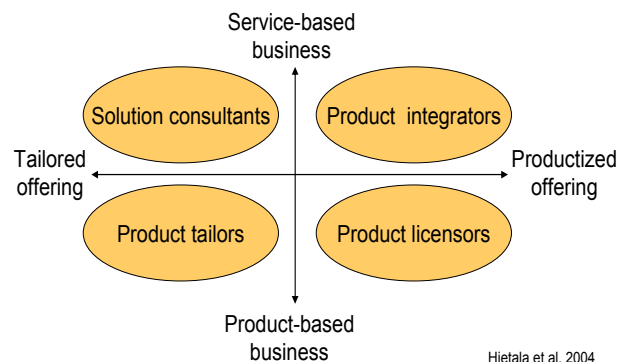


Figure : Business models of hybrid solution companies

Strategic decisions will have to be taken considering:

- to be either more product or more service oriented because it is too difficult to be equally good at both
- the degree of generic and customized software parts. Platform approaches are obviously to be preferred.

These decisions will impact the organizational capabilities and financial investments.

2.3 Open Source software^[3]

Even if the objective of this paper is not to define deeply the Open Source from its origine, it is worthwhile to provide the context.

Originally the movement came from the Free Software Foundation (FSF), established in 1985. The FSF is dedicated to promoting computer users' rights to use, study, copy, modify, and redistribute computer programs.

In 1997 the term "Open Source" was decided upon in a strategy session held in reaction to Netscape's announcement of a source code release called Mozilla. The term was used to clarify a potential confusion caused by the ambiguity of the word "free" in the English language. Later that year the Open Source Initiative formed and began using the term Open Source to describe software which refers to:

- Use, modify, redistribute
- Community
- License

Supporters of the Open Source claim that Open Source is NOT^[4]:

- new
- a set of development done by students
- of low quality
- inefficient for large-scale development
- difficult to scale

2.3.3 Licenses

Open source software is distributed under a variety of **different types of Open Source licenses**, all with the intent of guaranteeing that the code remains open.

By definition, these licenses must **not be specific** to certain software distributions and **cannot restrict any product** that is distributed alongside the Open Source software.

Through the broad landscape of Open Source license, we can distinguish very popular ones which have gained credibility and have largely been adopted by project groups :

- GNU General Public License (GPL) (67%)
- Lesser GPL (LGPL) (10%)
- A combination of Berkeley Software Distribution (BSD), Massachusetts Institute of Technology (MIT), Apache and Public Domain at 12%

Please refer to Appendix and references to get more detail on those licenses.

2.3.4 Community

The producers of Open Source products are typically a diverse group of developers who are salaries of companies involved in Open Source activities or talented developers working to gain visibility from their peers. The role of the community consists in controlling what happens with the product by making one crucial choice: the license.

Companies can take great benefit from projects sponsorship and partnership^[5]:

- Greater market exposure
- Expanded business opportunities
- Joint strategic sales and marketing initiatives

- Access to a broad and growing customer base

2.3.5 An innovative business models

Open Source initiatives have given birth to innovative business models often based on commercial software business models as shown by IT industry adoption in their portfolio and expertise. Even though there are still some innovative examples:

- Subscription business model

Another's business model is a Software as a Subscription one whereby customers receive a certified build of Open Source software that comes bundled with technical support and maintenance as an annual contract. Customers will purchase subscriptions with different options (degree and way of support, delay to deliver bug fixing). One good example is Covalent.^[6]

- Dual licensing model

Some Open Source companies own copyrights for their code and the software they develop is available under both an Open Source license such as GPL and a commercial license. Open source developers can choose the Open Source license. But any companies that wish to incorporate the code in a proprietary product must pay for it under the commercial license. This business model is used by Trolltech AS, MySQL AB^[7] and Sleepycat software.^[8]

This overview of innovative business model being done, it is necessary to highlight that a number of companies pioneering those business models failed driving a number of them out of business in the years 2000.

It shows that it is vital in such business models to be able to monetarize the services provided in order to survive.

2.4 Comparison between Open Source & commercial SW Business Models

Open Source business models have mainly been inspired by commercial ones. Some initiatives have been taken concerning license agreements and service offers. But we also observe that Open Source companies are mainly IT companies that decided to give away software development and open their code to focus on customized development and services. Some commercial companies are considering Open Source like Borland which will participate to Eclipse project for generic software

modules and develop internally customized and differentiating SW modules^[9].

If Open Source is not necessarily chosen, we can summarize by saying that software companies decide sooner or later:

- not to compete on generic SW modules but on customized ones
- not to compete on SW modules but on their integration

When a company delivers a software product, customers will require bug fixing and rapid problem solving which is not provided by Open Source communities.

Today, the growth rate of the main companies selling commercial and proprietary software is indeed the best proof that Open Source software will not replace commercial software soon^[10].

	Commercial software	Open source software
File format	Object	source
Targeted customer	oriented by general need (product business)	oriented by specific need (service business)
Development cost for generic SW code	unshared - fully paid and done by the company	shared - done by the members of the community
Bugs, updates	wait for SW vendor updates	Possibility to modify code and fix problems on its own

2.5 Strategic issues

We will describe common issues faced by all software companies.

2.5.1 Ownership of rights^[11]

Ownership of rights is fundamental and influences the company in pricing its software, in changing its licensing policy and in distributing software with different licenses.

The risk when using Open Source licenses is that the license may involve conflict by diluting the perceived added-value and even by creating lock-in situation with the impossibility to re-license.

When software protection is covered by copyright two ways exist to clear rights:

- Rewrite an original version for the SW based on the non protected basic ideas.

- Obtain rights with a license contract specifying such rights. The latter is certainly the most pragmatic way to access to the software functionality

2.5.2 Certification

Certification and quality assurance are fundamental issues, difficult to solve but important to gain recognition from customers and even to raise barriers to entry.

Many Open Source companies have built a branding image and developed certification programs to validate a company involvement in the community, the level of competencies of its developers, its code mastering and integration capabilities. (ex. Apache Software Foundation^[12], MySQL AB)

Commercial companies are also delivering services to increase the value of their technology and control its image. SAP delivers training sessions and different sort of certificate grades, such as an interface certification program, consulting, and access to test systems^[13].

What are the common points with the upcoming AUTOSAR standard considering the different business model described earlier ?

3 The automotive industry : from traditional black box business to innovative business models

3.1 AUTOSAR : managing the complexity^[14]

The AUTOSAR partnership is an alliance of OEM and Tier 1 automotive suppliers working together to develop and establish an open industry standard for automotive E/E architecture. It will serve as a basic infrastructure for the management of functions within both future applications and standard software modules.

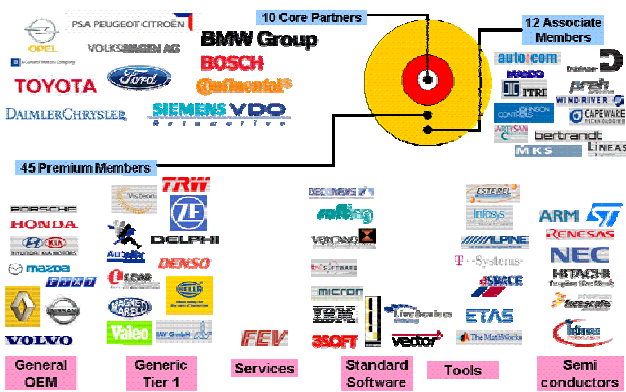


Figure : AUTOSAR Membership (June 2005)

The objective of the AUTOSAR consortium is to cooperate on standards and compete on implementation.

This architecture presents a lot of points in common with the Personal Computer from IBM. And we can also see some common points with Open Source.

⇒ The main difference here is that all actors exist but no one can say how deeply their role will be affected.

3.2 Customer needs

Today the automotive actors are pushing forward the standardization through AUTOSAR in order to:

- Improve quality and reliability of the E/E systems
- Manage the increasing E/E complexity associated with growth in the functional scope
- Improve their competitiveness, by easing the implementation of innovative functions, developed by themselves or by third parties
- Decrease the development costs, via software components reuse.
- Optimize and master the scalable E/E solutions

Tools for vehicle validation, for system modelization, for simulation and supporting the function partitioning with data exchange will become more complex.

3.3 Predictable impacts

3.3.1 On relationships

The current automotive embedded software landscape is composed of three major players: the OEMs, the Tiers 1 automotive suppliers and in some extent the software houses (software development companies).

⇒ Today's relationships between these three actors are changing.

Within an AUTOSAR framework, this representation could be impacted. The former bilateral relationship between OEMs and Tiers 1 is changing to a triangle including the Software Suppliers.

The OEMs and the Tier 1 are assessing the role that the software companies could play. It is thinkable to give them a major role to define the software architecture and the required tools for the implementation, configuration and validation. This is thinkable, but is it the unique solution?

The issue is to master a combination of expertise and it is crucial for the first years: automotive systems and electric/electronic architecture, automotive project management and real time embedded software engineering have to be mastered at once to insure the success of the first implementations.

In addition to the technical issues, the liability and the required initial investments seem to be a handicap for the software houses.

Taking this situation into consideration, what could be the best scenario to supply value to customers.

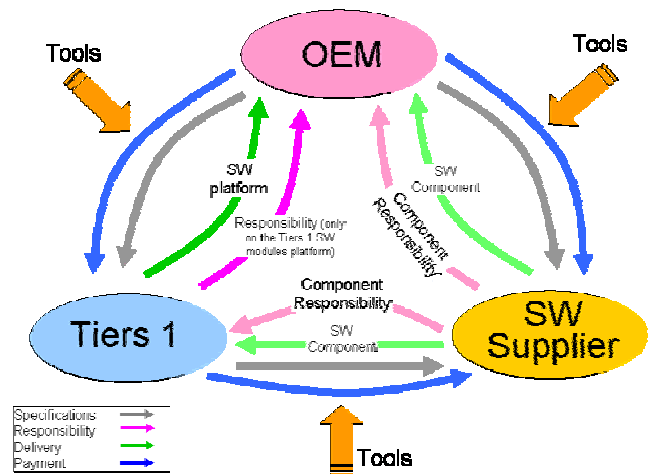


Figure : Relationships of actors within AUTOSAR environment

The above figure points out the need to define the level of responsibility for all the involved parties. It is today obvious that the level of responsibility is not the same among all these involved parties, whereas this aspect remains crucial within the evolving landscape.

On one hand, the software suppliers will handle the responsibility only for their component delivery (and not to an extended scope).

On the other hand, if the Tiers 1 act as system integrators (implementing the TPS modules), they will handle the responsibility for the whole system and for the complete software platform delivery.

In addition, the role of each party for investigation in case of technical issue still to be precisely defined.

3.3.2 On Business Models

Consequently, a new business model has to be set up in parallel with the platform business. As this standard will enable developing and offering

software Components Off The Shelf (COTS), AUTOSAR will split the current automotive business.

Today's ECU business that consists of selling "black boxes" may evolve into a market of different stand alone products like:

- ECU and Hardware components
- Software modules (basic layers and Application / Functions)
- System integration
- Associated Services (via dedicated Tools Chains and specialized methodology)
- Project management

The additional services (like System Integration, Software Calibration or functional partitioning at vehicle level) will increase in importance, as they are essential for a modular environment.

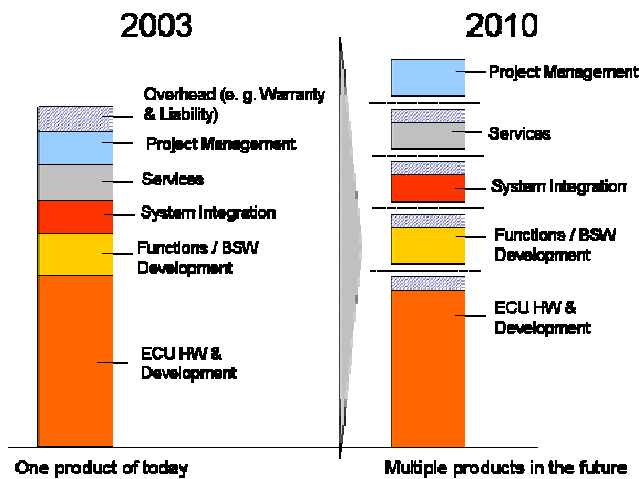


Figure : Impacts on Business Model

3.3.3 On the Value Chain

Some of the OEMs are willing to delegate some parts of the system development due to the increasing complexity, the time to market reduction and the growing need of innovation

Of course, the investment and risk sharing with the suppliers is still also a good motivation for the car makers.

Consequently, only the suppliers offering strong cooperation during the entire development phases (including pre-development and co-design) with OEMs will be successful in the future.

That's why the Siemens VDO AG platform strategy integrates a partnership development plan aspect with our customers in order to best fit with the impacts on the value chain.

Siemens VDO AG has initiated such project. It targets on paving the way to offer our customers solutions to make a real step forward into new

business models, legal aspect considerations as well as adapted technical processes.

3.3.4 Certification

The AUTOSAR consortium is working on the issue of an independent Conformance Test Agency (CTA). But with software each time minor modifications are realized conformance tests have to be done to validate the entire system.

To face those constraints different options are still open such as self certification or third parties certification.

3.4 Comparison between AUTOSAR & Open Source

AUTOSAR environment may seem, in a first insight, some common points with Open Source ecosystem explored earlier:

- Governance and decision process
- Work packages organization
- Search of innovative business model

However a further analysis shows that significant differences can be raised.

Firstly the Automotive industry being very concentrated can not be compared with Open Source adopter in Information Technology industry. As a consequence the community effect will not be leveraged and we know this is a key point in the success of Open source mechanism.

Secondly the AUTOSAR Consortium rules today does not allow Open source mechanism due to the restriction of some IP in the area of the members.

3.5 My "Open Source car", an utopia or a possible future ?

In any case Siemens VDO AG has to consider the different potential scenario to adapt the best to its customer needs.

Also in a longer basis the basic software may follow Open Source trends but this is only one of the possible scenarios.

To date Siemens VDO AG strategy is clearly to:

- Focus on customer needs by providing added value services
- Compete on innovative modules
- Compete on services
- Cooperate on AUTOSAR Specifications
- Develop a community of experts

- Build an image of quality by certification services
- Having a clear vision of the entire architecture to create intimate partnership with its customers

Considering all these new ideas for the automotive industry and possible scenarios, Siemens VDO is currently building innovative commercial offers with deep assessments of customer needs.

4 The Siemens VDO AG market proposals

4.1 Siemens VDO Platform solution

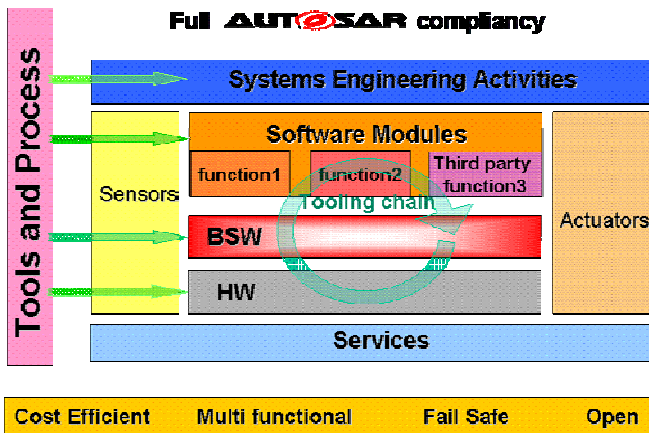


Figure : The Siemens VDO Platform Approach

To become a platform leader Siemens VDO is setting a strategy to:

- Create incentives for existing/new companies to join the platform to complement it efficiently with products, services and tools that are not within the Siemens VDO core business.
- Drive innovation with internal teams
- Address the entire software platform sold with all the different modules

4.2 Systems – Software – Services

The software platforms will include the three following elements:

- System Engineering activities (e.g. Electric/Electronic Vehicle Architecture Design, Expertise in LF/RF System integration with Access Control and Tire Guard systems, full engine management software platforms), to deliver complete integrated solutions for our customers.
- Functional Software modules (Tire Guard ASW or Battery Monitoring ASW), to be sold as products, within a AUTOSAR compliant platform environment.
- Professional Services along with its system offer (e.g. Active Driver Assistance System – ADAS – called Pro.pilot) vehicle function, HMI Cockpit Design, global Vehicle Energy Management) associated to software platforms and Systems activities supported by appropriated tools.
- Tools: at vehicle levels, dedicated tools for successful implementations will be needed.

At vehicle level for EE system and architecture management, at sub system level for calibration

and validation and at SW module level for configuration and generation.

4.3 A Siemens VDO AG Commercial Offer for a Basic Software Platform

When delivering a full BSW platform solution, Siemens VDO AG offer :

- The Software modules (usually in object or .lib format)
- The Technical documentations (User manual, tests plan, integration plan)
- A set of Tools to enable full development

In the meantime, Siemens VDO AG is setting up an adapted commercial offer detailed as follow :

- Software License Agreements
- Some additional services (Integration support, Software Customization, Training & Hotline and Maintenance)

This process of change of business models is defined with close relationships with customers to determine their needs and propose them suitable answers.

4.4 Liability

Concerning software products ownership of rights and liability are important issue:

- Responsibility and liability shall be defined by appropriate process, when a system supplier integrates third-party software or when software supplier provides third-party software to a system supplier

4.5 Migration plan

Given all these inputs, Siemens VDO is currently defining its migration plan internally and revising and adapting it to each customers according to the method described below.

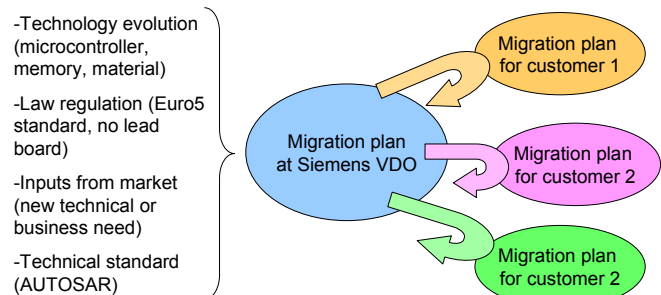


Figure : Migration plan method

5 Conclusion

The AUTOSAR motto is:

"cooperate on standard, compete on implementation".

As we have seen in this paper, some analogies could be done with the foundations, mechanisms of Open Source initiatives. In a first insight the model could look like applicable in the automotive industries and even attractive.

This paper developed the different restrictions could be raised.

The first requirement is: the basis of sound business models need to be clearly established in a win-win approach with cultural changes in Tiers 1 but also with the OEMs.

Siemens VDO is exploring those different scenarios in order to propose to its customer the most innovative business models and to anticipate any changes in the industry.

Siemens VDO AG intend to play a lead role in the definition of those business models to be always close to its customers needs in order to satisfy them.

6 Acknowledgement

Many thanks to my colleagues for their support:*

Expertise Support: Sébastien Calvez
Bernard Sanchez
Stéphane Fregosi
Pierre Baroghel

Reader : Florent Bonetto

7 References

- [1] Siemens VDO Communication:
 - "An open platform strategy in the context of AUTOSAR", Electronic systems for vehicles, Baden Baden, Germany, 2005.
 - Jean-Luc Maté, Vice President Siemens VDO AG Convergence 2004 Detroit - May 2004
Date 2005 Munich - March 2005
Usine nouvelle 2005 -June 2005
- [2] Business Models That Last: Balancing Products and Services in Software and Other Industries Paper 197 December 20 2003 Michael Cusumano
- [3] www.fsf.org
- [4] Perspectives on free and Open Source software, chap. 3, by J. Feller, B. Fitzgerald, S.A.Hissam and K. Lakhani Ed. MIT Press June 2005
- [5] <http://www.sap.com/partners/index.epx>
- [6] <http://www.covalent.com>
- [7] <http://www.mysql.com>
- [8] <http://www.trolltech.com/newsroom/announcements/00000161.html>
- [9] http://www.borland.com/us/company/news/press_releases/2005/02_28_05_borland_amplifies_eclipse_support.html
- [10] Open Source Versus Commercial Software: Why Proprietary Software is Here to Stay by Shawn Shell Oct 14 2005 <http://www.informit.com/articles/article.asp?p=420290>
- [11] Mikko Välimäki and Ville Oksanen: "Evaluation of Open Source licensing models for a company developing mass market software", Lawtech, 2002, http://www.hiit.fi/de/valimaki_oxanen_lawtech_2002.pdf
- [12] <http://www.apache.org>
- [13] <http://www.sap.com>
- [14] <http://www.autosar.org>
- [15] <http://www.opencascade.com>
- [16] <http://www.opencascade.org>
- [17] Innovation happens elsewhere : Open Source as Business Strategy by Ron Goldman and Richard P. Gabriel Ed. Morgan Kaufman 2005
- [18] Strategies and guidelines for creating business value through software – Dr Jyrki Kontio – Helsinki

8 Glossary

Commercial-Off-the-Self (COTS)

Commercially available generic SWasP. Usually for a given function or infrastructure layer that can be purchased and integrated with low customization or no customisation, thus facilitating customer infrastructure expansion and reducing costs.

GNU General Public License (GNU GPL)

The first version was published Richard Stallman, a former staff member at MIT computer lab, in 1989 as part of the GNU project and it has since become the flagship of the so called free software movement. It allows free distribution and modifying but all bundled and derivative works must be under GNU GPL.

IP (rights)

Legal framework for protecting new technical solutions or software using patents, trademarks and/or copyrights.

Liability

Responsibility for defects caused by the software used in third party hardware.

License

A license is a contract that authorizes the right of use, or manufacture, or trade (exclusive or not) of an intellectual property title, without becoming the owner of the licensed product.

LF/RF

Long Frequency / Radio Frequency.

SWasP

SoftWare as a Product (SWasP) is a software product that is designed, developed, tested and marketed as a stand-alone product in the product portfolio of a company.

TPS:

Third party software , Software to be integrated in a System.

Warranty

Guarantee obligations specified into the Software License Agreement, by which one of the two parties guarantees the other one the enjoyment of the good or the right.