Governance in Collaborative Open Source Software Development Organizations: A Comparative Analysis of two Case Studies

Master's thesis

Faculty of Business, Economics and Social Sciences

University of Bern

submitted to

Dr. Matthias Stürmer

Research Center for Digital Sustainability

Institute of Information Systems

by

Winkelmann, Rahel

from Siselen

11th semester

Matriculation nr.: 09-127-788

Study address:

Huberstrasse 22

3008 Bern

(Tel. 078 758 58 96)

(e-Mail: rahel.winkelmann@students.unibe.ch)

Bern, 20.01.2015

Abstract

While loose cooperation among several actors is common in the open source sector, companies merging into a professionally governed collaborative open source software development organization across industries is an emerging phenomenon. The purpose of this thesis is to shed light on this new approach of software development by creating a framework for building a collaborative open source software development organization. A comparative analysis examines the governance models of two different collaborative open source software development organizations from the organizational, financial and legal perspective and reveals the autonomous and the affiliated organization type and their key characteristics. Based on these findings and by means of four expert interviews a framework consisting of eight criteria that need to be considered in order to build a collaborative open source software development organization is created.

Zusammenfassung

In der Open Source Branche ist es gängig, dass sich verschiedene Akteure zur Softwareentwicklung zu losen Konsortien zusammenschliessen. Unternehmen, welche sich im professionellen Rahmen zu einer Organisation zusammenschliessen um gemeinsam Open Source Software zu entwickeln, sind jedoch ein neues Phänomen. Der Zweck dieser Arbeit ist es Aufschluss über diesen neuen Ansatz von Softwareentwicklung zu geben. Dies geschieht anhand der Entwicklung eines Rahmenkonzeptes über die Bildung einer Organisation, deren Mitglieder gemeinsam Open Source Software entwickeln. Eine vergleichende Analyse untersucht die Governance Modelle von zwei unterschiedlichen solchen Organisationen von der organisationalen, finanziellen und rechtlichen Perspektive und erläutert den autonomen und den angegliederten Organisationstyp mit den jeweiligen Kerncharakteristiken genauer. Basierend auf diesen Forschungsergebnissen und mit Hilfe von vier Experteninterviews wird ein Rahmenkonzept mit acht Kriterien, welche es zur Bildung einer solchen Organisation zu berücksichtigen gibt, erstellt.

Acknowledgement

- Dr. Matthias Stürmer for his support, motivation and for sharing his open source knowledge and experience.
- Steve Crumb (Executive Director, GENIVI Alliance),
Ralph Mueller (Managing Director Eclipse Foundation Europe GmbH),
Jeremiah Foster (Community Manager, GENIVI Alliance) and
Claus-Peter Wiedemann (Lead License Review Team, GENIVI Alliance) for four interesting conversations and sharing their knowledge and experience.
- Gabriela Gnägi and Nicole Perreten for revising the thesis.

Table of Contents

1 Introduction

1.1 Background

In recent years, software companies discovered a need to expand their research and development outside their organizational boundaries. Due to rapidly changing technology and the continually growing customer needs software companies were no longer able to provide the required functionalities and to remain competitive without external support. As a consequence of this trend, software companies started to cooperate and to build up networks for software development and software-based services, so called software ecosystems (Bosch 2009; Tiwana, Konsynsik and Busch 2010; Van den Berk, Jansen and Luinenburg 2010). Along with this movement the advancement of Linux and open source software experienced a boom and gained more and more significance (Linux Foundation 2014). Since it is nearly inevitable to encourage software development without making the software at least partially accessible to others, the software companies had to face the emergence of commercially used open source software. Open source software had no longer the reputation of a hacker-software but rather of a serious alternative to proprietary software. As a consequence of these two developments more and more proprietary software companies engaged in software ecosystems that were based on open source software (Benkler 2002; Fitzgerald 2006).

1.2 Problem Description

With the commercialization of open source in the last decade (00s), research has paid growing attention to this arising phenomenon. The causes, incentives, advantages and objectives of open source software have been dealt with in many publications (Kogut and Metiu 2001; Raymond 2001; Lerner and Tirole 2002; Bonaccorsi and Rossi 2003; von Hippel and von Krogh 2003; Fitzgerald 2006; West and O'Mahony 2008). Despite the fact that all these papers, theses and journal articles agree on the collaborative approach of the open source movement, they are divided over the exact denotation of this collaboration in the open source domain. Researchers either create their own term in order to describe the collaborative phenomenon or use the already existing notations arbitrarily. So far, these collaborative software development constructs are usually initiated by software companies, thus from the perspective of the developers. Hence the companies whose core business is software development are building the corresponding platforms intending to save costs in the software development process (Riehle 2010). However, a new phenomenon concerning companies that are not positioned in the software development industry has emerged recently. An increasing number of companies, whose core business is not software development but are reliant upon the latest findings regarding their applied software pursue a collaborative software development approach through merging their resources across industries on an open source base. The joining together across industries means that software users in terms of companies as well as software developers are part of these collaborations in order to collectively develop software (Linux Foundation 2014). Considering the fact, that these collaborative software development constructs initiated by software users and not by software developers are only on the rise since a few years, there was little evidence that this phenomenon is relevant and part of the future software development.

In 2014, however, the Linux Foundation published a trend report, which is based on a survey conducted with 686 software developers and business managers. Aside a few studies (Machbarkeitsstudie 2013; Liu, Hansen and Tu 2014) this report is one of the first that provides evidence of the practical relevance of collaborative software development on an open source base among commercially organized enterprises. The report (2014) reveals three key findings: First, it states *“companies get involved in collaborative software development to advance business objectives and to be part of industry innovation”*. Further the results show that *“investments in collaborative software development are on the rise”* (p. 2). And as a third finding the Linux Foundation report reveals that both, businesses and individual software developers gain from the trend toward collaboration. Businesses benefit in terms of a shorter product development cycle and faster time to market and individual developers regarding their personal skills. Overall, two out of three business managers consider collaborative software development as very important to their company (Linux Foundation 2014). Nevertheless, to the best of the author’s knowledge, up until now there has been little or no research on how to build a collaborative software development organization and which criteria need to be considered in doing so.

1.3 Aim of the Thesis

There are two objectives of this thesis. The first is to create a common understanding of the existing open source constructs and the different denotations used to describe them. In order to remove the inconsistency and create transparency regarding the vast number of terms in open source literature, the first research question to be answered is:

Which terms are used to describe collaborative open source software development in literature and how can they be classified?

The second objective is to evaluate two different governance models of collaborative open source software development organizations by means of a comparative analysis. The evaluation is supposed to reveal what different types of open source software development collaborations currently exist and what advantages or disadvantages they offer. The aim of this comparison is to develop a framework for building a collaborative open source software development organization. In view of that, the second research questions is:

What different types of collaborative open source software development organization do currently exist and which criteria need to be considered in order to build a collaborative open source software development organization?

1.4 Structure of the Thesis

In Chapter 2, the research approach as well as the research methods in terms of secondary and primary data collection are described. The selection process of the objects of investigation on the basis of literature and online research (secondary data), and the data collection through expert interviews (primary data) are explained in more detail. In the third chapter the conceptual basics and the theoretical background of open source are introduced and a classification of the most significant terms is made. Subsequently, in Chapter 4 the objects of investigation selected in Chapter 2 are analyzed in more detail. In Chapter 5, the comparative analysis is done and the key characteristics of the different objects of investigation are highlighted. In a last step, the findings of the preceding chapters and the expert interviews are compiled and put together in the form of a framework.

2 Research Approach

The research approach of this thesis is a comparative analysis. The analysis is a comparison between the governance models of two differing collaborative open source software development organizations. Due to the limited extent of this thesis, only the organizational, financial and legal perspective will be dealt with. The technical aspect will be deliberately neglected. The data gathered is composed of secondary data consisting of literature and online research and primary data consisting of qualitative interviews with four experts from the open source sector.

2.1 Secondary Data Collection

The selection process and by reference to which criteria the two objects of investigation have been chosen is based on the secondary data collection.

2.1.1 Literature and Online Research

In order to gain an overview of the topic of this thesis, a widespread literature review has been conducted. From the basics of open source via open innovation trough to software ecosystems, various papers, books and articles have been analyzed. The literature review provides the basis for answering the first research question (see Chapter 1.3) about the distinct use of terms concerning open source (see Chapter 3.5).

In order to answer the second research question (see Chapter 1.3) about collaborative open source software development organizations and to find appropriate objects of investigation, a comparison between a few of the existing organizations has been drawn. Since the collaborative open source software development approach is a recent phenomenon, there was only a limited number of open source organizations to choose from. Possible objects of investigation were the five organizations: the GENIVI Alliance, the Kuali Foundation, OneGov GEVER, OpenJustitia and the Polarsys Working Group hosted by the Eclipse Foundation. Those are introduced in the subsequent chapter.

GENIVI Alliance

The GENIVI Alliance is a non-profit organization among automotive manufacturer, suppliers and soft-, middle- and hardware providers, founded in 2009. The long-term goals of the association are to shorten the development-cycle, to fasten time to market and to reduce the costs for all involved parties developing In-vehicle Infotainment (GENIVI, About GENIVI 2014).

Kuali Foundation

Kuali is a foundation among several universities of the United States of America and South Africa. The foundation was established in 2004 and intended to create a financial system that is built by higher education for higher education (About the Kuali Foundation 2014).

OneGov GEVER

OneGov is a Swiss association founded by several public administrations in 2012. The purpose of OneGov is the shared development of eGovernment solutions based on different modules (onegovbox 2014). The superior system of these modules is called OneGov Box (onegovbox, OneGov Box 2014). OneGov GEVER is a web application and part of the OneGov Box. OneGov GEVER is used in the areas of records management, process control and business control (onegovbox, OneGov GEVER 2014).

OpenJustitia

OpenJustitia is an open source organization initiated by the federal court of Switzerland in 2011. The organization provides software modules that enable courts to manage their court decisions customized to their needs. The long-term goal are cost savings through reuse of the application by various Swiss courts in order to disburden the public sector as well as the taxpayer (Bundesgericht, OpenJustitia 2014).

Polarsys

Polarsys is a subgroup, a so-called working group that grew out of the Eclipse Foundation in 2011. Several industrial members and tool providers have founded Polarsys in order to create and support open source tools to develop embedded systems. Core areas of the Polarsys are the aerospace industry, defense and security, the energy sector, health care, telecommunications and transportation (Polarsys, About Us 2014).

2.1.2 Selection of Cases

The organizations examined in the course of this thesis were chosen with reference to five criteria the author perceived to be relevant: The year of funding, the amount of members, the geographical extension, the domain, the maturity of the organization and the consequent assumed relevance to the open source movement in a global context (see Table 1).

The first criterion (year of funding) is related to the length of time an organization has already been dealing with the subject of collaborative open source software development. It was assumed that the longer an organization is active in a specific business area, the more experience and knowledge has been acquired. In turn, this higher state of knowledge leads to a broader base of information.

The second and the third criterion (amount of members, geographical extension) examine the size and interconnectedness of the existing collaborative OSS development organizations. Factors, such as the mixture of members or the degree of internationality and the consequential varying experience, had significant influence on the selection of the investigated organizations.

The domain serves as a fourth criterion and the last criterion concerning the maturity level is derived from the preceding criteria. In other words, if an organization has been active in the open source sector for several years, has a high number of participants, is geographically widespread and operates in a dynamic business domain, it can be assumed that the maturity level of this particular organization is rather high. In the following, the five suggested organizations are compared with reference to the defined criteria.

Set of Criteria	Kuali Foundation	OneGov GEVER	Open Justitia	GENIVI Alliance	Polarsys Working Group
1. Year of Funding	2004	2012	2011	2009	2011
2. Amount of Members	68	15	16	> 160	11
3 Geographical Extension	Binational (USA & South Africa)	National (CH)	National (CH)	International (worldwide)	International (worldwide)
4. Domain	Educational Sector	Public Sector	Legal Sector	Automotive Sector	Aerospace Sector
5. Maturity Level (low, middle, high)	High	Low	Low	High	Middle
Selected Members	American and South-African Universities (e.g. Massachusetts Institute of Technology, University of California – Berkely, University of Washington, Stellenbosch University, University of Johannesburg)	Swiss municipality, cities and cantons (e.g. Gemeinde Baar, Burgergemeinde Bern, Stadt Thun, Stadt Biel, Kanton Zug)	Bedag Informatik AG, Tribuna Allianz, Eurospider AG	BMW Group, Renault, Volvo, Blackduck, Pelagicore	Airbus, Ericcson, Thales, Artal, Obeo

Table 1: Selection of Cases.

In terms of the age the Swiss associations together with the Polarsys Working Group are the youngest collaborative OSS development organizations founded in 2011 and 2012, respectively. Followed by the GENIVI Alliance and the Kuali Foundation established in 2009 and 2004. Considering the second criterion, due to its global activity and the high amount of members (see Table 1) the GENIVI Alliance seems to be the most interconnected and biggest collaborative OSS development organization among the five contemplated organizations. Polarsys in contrast has fewer members but is worldwide active as well. The two Swiss associations OneGov GEVER and OpenJustitia are solely active on their national markets and the Kuali Foundation operates on a binational level. In consequence of this, the potential participants are automatically limited. Comparing the size of the USA and South Africa to Switzerland, it seems obvious that the Kuali Foundation has a bigger potential to grow. However, in comparison to the GENIVI Alliance and the Polarsys Working Group where companies from around the world are represented, the Kuali Foundation is rather small.

The domain might also influence the geographical extension since the educational (Kuali Foundation), the public (OneGov GEVER) and the legal (OpenJustitia) sector are geographically restricted to national regulations. Compared with this, organizations in the private business sector have higher potential for growth and business development.

In sum, taking into consideration the described criteria above, the GENIVI Alliance and the Polarsys Working Group are the organizations that correspond best with the defined requirements. Despite the fact that the Kuali Foundation has been founded a few years before, the two collaborative OSS development organizations from the private business sector seem to be more mature and therefore more relevant to the open source movement than the educational (Kuali Foundation), public (OneGov GEVER) or legal (OpenJustitia) organizations. Furthermore, through the high amount of members (GENIVI Alliance) and the international activity both the GENIVI Alliance and the Polarsys Working Group have a broader base of knowledge and experience from different perspectives. All things considered, the analysis of the five criteria led to the decision to choose the GENIVI Alliance and the Polarsys Working Group as objects of investigation of this thesis.

2.2 Primary Data Collection: Qualitative Expert Interviews

In order to develop the data gathered through literature and online research, four qualitative interviews with experts operating in the open source sector for years have been conducted. The interviewees had been selected on the basis of the chosen objects of investigation and the significance of their position. In other words, people from the GENIVI Alliance or the Polarsys Working Group holding a leading position were selected. Those were contacted by e-mail via either the respective website or the social networking service LinkedIn. The four experts have been interviewed within one month (October/November 2014). One interview took just under an hour and the other three approximately half an hour (see Table 2). The interviewees were asked questions about the governance, the legal documents and the funding of their respective organization and how they assess the future of open source (see Appendix). In order to ensure the validity of the statements concerning the organizations involved in this thesis, the experts have been asked to read through the respective parts and to confirm the provided information.

Expert Interviews				
	Steve Crumb	**Jeremiah Foster**	**Ralph Mueller**	**Claus-Peter Wiedemann**
Date	23.10.2014	11.11.2014	20.11.2014	21.11.2014
Duration (Min.)	36:08	26:40	56:42	24:42
Organization	GENIVI Alliance	GENIVI Alliance	Eclipse Foundation	GENIVI Alliance
Function	Executive Director	Community Manager	Managing Director, Europe	Lead License Review Team
Type of Interview	Skype call	Skype call	Phone call	Phone call
Main Tasks	• Strategic issues • Daily business management • Advisory function	• Management and technical support for open source projects • External representation function (conferences and events)	• Business development in terms of member acquisition • Community management • Event management	• Review of license compliance • Assistance of GENIVI members in legal matters

Table 2: Overview of Expert Interviews.

3 Introduction to Collaboration in Open Source Software Development Organizations

In order to examine different collaborative open source software development organizations it is necessary to provide a better understanding of what *open source* precisely means. Therefore Chapter 3 in a first step explains the term open source, its guiding principles, how the intellectual property rights are managed in the form of licensing and the connection to the open innovation approach in more detail (see Chapter 3.1 – 3.4). In a second step a selection of varying open source expressions used in literature are elucidated and classified into categories (see Chapter 3.5).

3.1 Open Source Software (OSS)

The core of the term *Open Source* is the accessibility of the source code. Whether individuals, groups or companies, every user who is interested in the software is entitled to reuse and modify the source code. In turn, the users are obliged to provide their modifications back to the community (Lerner and Tirole 2001). In his book *Understanding Open Source and Free Software Licensing*, Laurent (2004) defines three main advantages of open source software compared to proprietary software. These are innovation, reliability and longevity. Since the developers who contribute in open source software development have no monetary incentives but are rather driven by intrinsic motivation, the intention to leverage the software's functionality increases the innovation potential. Secondly, due to the personal interest of many developers, bugs are fixed faster and more reliably than by the creator itself. And the third advantage of developing software according to the open source principle is the fact that open source software can be repeatedly renewed and is not dependent on the continuation of the original creator.

Riehle (2007, 2010) takes it one step further and separates open source software into community open source and single-vendor or commercial open source. Single-vendor open source is open source software that is owned by a single for-profit company, which maintains the copyright and the decisive power. Community open source in contrast is owned by a not-for-profit-community where the voluntary developers decide on the strategic direction. In reference to the topic of this thesis, only the community open source will be relevant subsequently.

3.2 The Open Source Initiative (OSI)

The *Open Source Initiative* is a non-profit corporation whose purpose is to strengthen the position of open source against proprietary software. The goal of the open source initiative is to create an understanding of the meaning of open source and to distribute this new approach of software development among proprietary software user. The open source definition is the basis of the open source initiative and serves as a guideline concerning the standards open source software must fulfill in order to be distributed under the term *Open Source Initiative Approved License*. The open source definition consists of ten requirements. The key features of open source software are the free availability, the copyright, the right to modify and the right of changed or unchanged redistribution of the source code. The open source definition aims at establishing transparency and preventing confusion in terms of which software are truly corresponding to the open source standards and which are not (Open Source Initiative 2014). Example of OSI approved licenses are the GNU General Public License (GPL) (see Chapter 3.3) and the Eclipse Public License (EPL) (see Chapter 4.2.3).

3.3 Legal Aspect of Open Source: Intellectual Property Rights and Licensing

The Swiss copyright law ensures the protection of the originator of intellectual property in relation to art, music or literature as well as software programs (Bundesgesetz über das Urheberrecht und verwandte Schutzrechte, Urheberrechtsgesetz (URG) 2011, Art. 2 Para. 3). In concrete terms the originator has the exclusive right to decide if its work shall be made available to third parties or not. This process of giving third parties the right to copy, modify, change or distribute its work, in our case the software, is commonly known as licensing. By doing this, the licensor regulates the rights and duties related to the use of the particular software towards the licensee (Lerner and Tirole 2002). In the traditional software industry it is common use that software providers reduce the availability of their software to the lowest level possible. This means that they have private property rights claims (Kogut and Metiu 2001). In contrast to that, the OSS collective pursues a completely different approach by using licensing but entitling third parties to use the software according to open source principles. In other words those intellectual property rights are deposited in the public domain (Kogut and Metiu 2001; Machbarkeitsstudie 2013). The first open source license was the General Public License (GPL) created by Richard Stallman in 1989 (Kogut and Metiu 2001). The GPL originated as a consequence of the GNU-Project whose purpose was to create a free counterpart to the operating system Unix. In technical terms GPL is based on Unix, however, in ideological terms they are completely different. That is why the name of the project is an acronym of *GNU's Not Unix*. In order to release the GNU operating system the GPL has been developed (GNU Operating System, About GNU 2014; GNU Operating System, Licenses 2014). Today, the GNU GPL tops the list of the top 20 Open Source Licenses and is therefore the most common used license in open source software projects (Blackduck Software 2014).

In dealing with open source licenses it is important to understand the difference between *free* in terms of freedom and *free* in terms of without charge. Free software in connection with OSS means solely that everyone is free to use the software in a way described in Chapter 3.1. It does not mean that it is prohibited to charge a price when distributing a modified version (GNU Operating System, What is copyleft? 2014). A further essential part of GPL is the copyleft that was shaped by Stallman. The term copyleft is a play on the word copyright and shall represent exactly the opposite. Whereas copyright takes away the users freedom, copyleft guarantees the users freedom. In concrete terms this means that the software itself is free (in terms of freedom) and anyone redistributing the free software, whether modified, extended or left unchanged, is required to pass along all the rights to a subsequent recipient (GNU Operating System, What is copyleft? 2014). With the copyleft clause the GPL is the most strict open source license. More precisely, users transferring any additional components to a GPL code are committed to distribute those add-ons under the license of GPL. Other OSS licenses pursue a more permissive approach and only require its users to declare the transferred license code whereas add-ons can be distributed under commercial terms and conditions or under any open source or even proprietary license (Überhorst 2009; Milinkovich 2010). An example for a more permissive OSS license is the Eclipse Public License (EPL) (see Chapter 4.2.3).

3.4 Open Innovation

Chesbrough (2006) defines the open innovation paradigm as *“the antithesis of the traditional vertical integration model where internal R&D activities lead to internally developed products that are then distributed by the firm”* (p. 1). Open innovation aims at the interorganizational collaboration between companies of the same industry. The basic idea of this new approach of innovation process is the question of why companies, which basically need the same resources, should invest in a complex and expensive innovation process, if they could do it in a collaborative, less expensive way (Enkel, Gassmann and Chesbrough 2009). Enkel et al. (2009) differentiate between three core processes in open innovation. These are the outside-in process, which integrates supplier, customer and other external sources in order to gain external knowledge. Further the inside-out process where ideas are transferred to the market by licensing intellectual property or multiplying technology and the third process, called the coupled process. The coupled process is a combination of the first two processes and aims at co-creation with companies out of the same- and/or across industry(ies). Through alliances and other forms of cooperation companies foster shared innovation (Enkel et al. 2009). It seems that the open innovation paradigm bears reasonable resemblance to the open source approach. Therefore the idea of open source can be considered as a pioneering area, characterized by

Raymond in 1997, that discovered the advantages of making use of external sources and bringing together knowledge in order to create qualitatively high products. Baldwin and von Hippel (2011) also ascertained in a study that there is a shift from producer innovation to user and open collaborative innovation. They distinguish between three types of innovator - single-user innovator, producer innovator and open collaborative innovation project. A single-user innovator is a single firm or an individual that develops an innovation for its own purposes. A producer innovator is a for-profit business that aims at selling or licensing its innovation to customers. And according to Baldwin and Hippel (2011) an open collaborative innovation project is defined as *"a project that involves contributors who share the work of generating a design and also reveal the outputs from their individual and collective design efforts openly for anyone to use"* (p. 1403). Considering the topic of this thesis, only the collaborative innovation will further be addressed. Baldwin and Hippel (2011) see the open innovation approach universally applicable to various economic sectors, which means that open source is rather part of the open innovation approach than an entirely different and new phenomenon. This leads us to the shift of open source considered as a marginal group of software developers in its early existence to the commercially recognized alternative of proprietary software today (see Chapter 1.1). The Linux Collaborative Development Trends Report (see Chapter 1.2) gives evidence that the collaborative open source software development establishes across several commercial industries. This trend is comparable to the coupled processes mentioned by Enkel et al. (2009). All things considered, it might be said that the emergence of open source in the early 90s (see Foster 2014) served as a reference point for open innovation. Over the years both trends evolved and eventually conflated into collaborative open source software development (see Figure 1).

.

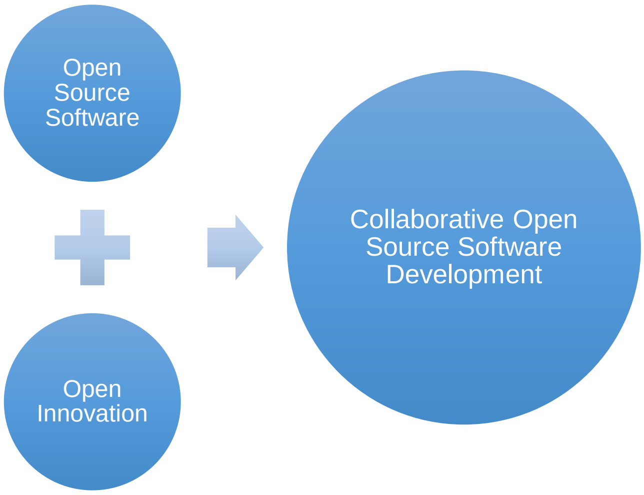


Figure 1: Synergy of Open Source Software and Open Innovation.

3.5 Collaborative Open Source Software Development

Chapter 3.5 shall answer the first research question defined in Chapter 1.3: *Which terms are used to describe collaborative open source software development in literature and how can they be classified?*

The term that describes the construct where open source software is collaboratively developed is not consistently used in literature as well as in practice. Terms such as open source software platform, open source software community, open source software projects, open source software foundation or open source software ecosystem are the most frequent used expressions and are often used arbitrarily or synonymously. To create transparency, a differentiation among the several terms shall be made as accurately as possible. In order to do this, the meaning of each expression is described first.

3.5.1 Open Source Software Platform (OSS Platform)

A software platform generally refers to the underlying hardware and / or software architecture of a system. It serves as a standard upon which other applications can be developed and is the crucial base for running any other software applications or programs (Beal 2014; Techopedia 2014; Techterms 2014). Economides and Katsamakas (2006) call these platforms, *technology* *platforms*. Examples of such technology platforms in terms of software are the operating systems Microsoft Windows (proprietary software) or Linux (open source software) and in terms of hardware the Intel processors (Economides and Katsamakas 2006). This general term gets more precise with the definition of Dedrick and West (2003) who define open source platforms as *“platforms based on open source operating systems”* (p. 241). Overall, when the term *Open Source Software Platform* is used in literature, it refers to the technical base that is crucial in order to initiate any sort of open source construct.

3.5.2 Open Source Software Community (OSS Community)

This term can be interpreted in different ways. Lerner and Tirole (2001, 2002) or Bonaccorsi and Rossi (2003) for instance perceive the term *Open Source Software Community* in the sense of how the Cambridge Dictionaries Online (2014) define a community: *„People who are considered as a unit because of their common interests, social group, or nationality”.* They use this term in order to describe a specific group of people who share pretty much the same opinions and attitudes against a particular issue. In this case, each software developer who develops and supports open source software would belong to this (one) community.

In contrast Nakakoji, Yamamoto, Kishida and Ye (2002) or Sharma, Sugumaran and Rajagopalan (2002) understand the term *Open Source Software Community* as a description for a system or an organization for a specific purpose. They consider an OSS community as a merger of various OSS developers to work on one or more specific projects based on an online OSS platform. Accordingly, this way of interpreting the OSS community leads to the conclusion that there might be countless differing open source software communities. Participants are software developers or software developer groups dispersed around the world communicating through asynchronous means of communication such as emailing or chats. There is neither a minimum nor a limiting number of participants. Therefore the size of OSS communities does vary widely. Furthermore, there is no monetary incentive to participate. The involvement is entirely voluntary (Nakakoji et al.

2002; Sharma et al. 2002; West and O'Mahony 2008). However, OSS communities usually grow out of a particular need of an individual or a company. Therefore the goal of open source software communities is to improve, review and adapt a specific source code in collaboration with experienced developers in order to develop a product of high quality (Nakakoji et al. 2002; West and O'Mahony, 2008). Additionally, according to Sharma et al. (2002) OSS communities are lead by a core group, which is responsible for a strategic direction as well as for a slight coordination of the different participants. Nevertheless, they state that there is a lack of formal organizational structure and governance that make OSS communities incomparable to traditional organizations (Sharma et al. 2002).

Onion Model

According to the often-referred model of Nakakoji et al. (2002) an OSS community can be compared to an onion (see Figure 2). An onion consists of various layers that reflect the different roles in an open source software community and are assigned to the participants, correspondent to their level of engagement. Nakakoji et al. (2002) define eight different roles where the degree of influence is decreasing the farther away the layers are from the center. Beginning in the middle of the onion the eight groups are the project leader, the core members, the active developers, the peripheral developers, the bug fixers, the bug reporters, the readers and on the outermost layer the passive users. Each group is briefly described in the following paragraph.

Project Leader

The project leader is figuratively speaking the heart of the onion and usually the person who initiated the project and accepts the overall responsibility. The project leader has a long-term vision and sets the corresponding overarching direction (Nakakoji et al. 2002; Kilamo, Hammouda, Mikkonen, Aaltonen 2012).

Core Members

Core members have most commonly been participating in a specific open source project for a long time and significantly contributed to the development. Due to their experience, they are responsible for the coordination of an open source software project (Nakakoji et al. 2002; Kilamo et al. 2012).

Active Developers

Together with the first two groups the active developers make up the driving forces in an OSS Community. They engage in a community on a regular basis and are concerned with the contribution of new features and bug fixing.

Peripheral Developers

In contrast to active developers, peripheral developers participate in existing software rather than contribute new functionalities. Further, their engagement is typically short-lived and irregular.

Bug Fixers

Opposed to the groups nearer to the middle, bug fixers have little knowledge of the source code the system is based on. Their knowledge is restricted to the part of the source code where the bug occurs. Either they discover the bug themselves or have been made aware of it by the bug reporters.

Bug Reporters

Bug reporters merely report bugs. They take on the role of testers and do not have the ability to fix bugs or read the source code.

Readers

Readers are active users and assume the role of peer reviewers. They use the system and try to gain knowledge about the functional principle of it by reading the source code.

Passive Users

Passive users are not involved in the development process. They only use the open source software, making them the largest group in an OSS community (Nakakoji et al. 2002).

This model can be regarded as a reference point for the structure of an OSS community, however it cannot be generalized. OSS communities may differ from this model in terms of the number of groups, their tasks and their responsibilities.

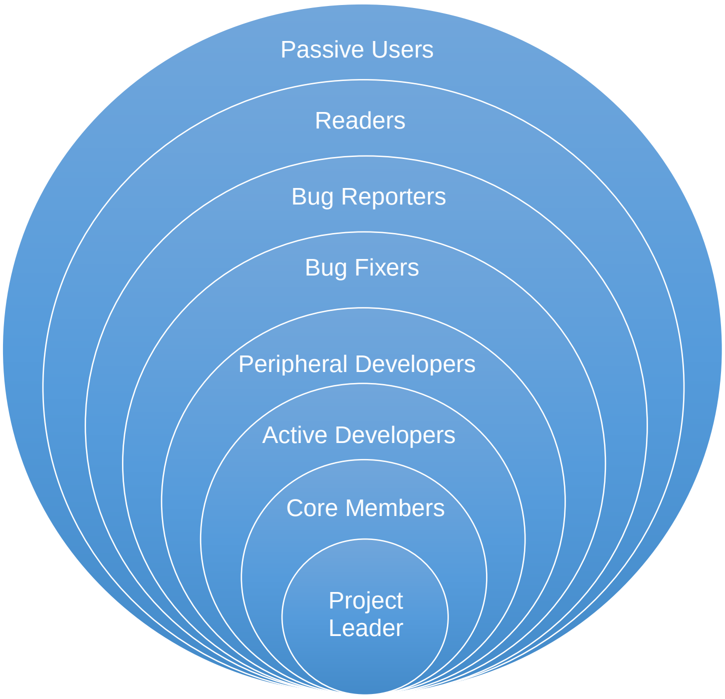


Figure 2: Onion Model (see Nakakoji et al. 2002).

3.5.3 Open Source Software Project (OSS Project)

The term *Open Source Software Project* is used in two different ways. On the one hand, researchers like Lerner and Tirole (2001, 2002) or Bonaccorsi and Rossi (2003) who perceive the OSS community as an ideology (see Chapter 3.5.2), use the term open source software project as the other researchers in the preceding paragraph use the term OSS community. Von Hippel and von Krogh (2003) define the open source software project as *“Internet-based* *communities of software developers who voluntarily collaborate to develop software that they or their organizations need"* (p. 209). Thus an OSS project is *the same* as the OSS community described before. From another perspective OSS projects are understood as projects *within* an open source software community. In other words, one community includes one or more OSS projects (Nakakoji et al. 2002).

3.5.4 Open Source Software Foundation (OSS Foundation)

An *Open Source Software Foundation* can be seen as the extension or as a further step in the development process of an OSS community. The existing communities, based on voluntary work and a slight extent of organizational structure and formalization are changed into a well-defined, transparently governed and organizationally structured non-profit organization (Riehle 2012). According to Riehle (2010*) "the main purpose of a foundation is to act as the steward of the software being developed and to ensure its long-term survival*" (pp. 86). He further defines the responsibilities and tasks of a foundation: a foundation is responsible for organizational, strategic and legal issues. These are for instance the formulation of a long-term strategy, the handling of necessary back-office processes, the management of intellectual property rights or the transparent information and communication with the community and its members.

3.5.5 Open Source Software Ecosystem (OSS Ecosystem)

Kilamo et al. (2012) describe a software ecosystem as *"a set of businesses that function as a single unit, instead of each participating enterprise acting individually"* and consider a software ecosystem as a *"business and governance model"* (p. 1468). A software ecosystem acts as an overall umbrella with the goal of providing technological, legal as well as administrative support to all its members and to give a strategic direction. Processes shall be simplified and consolidated in order to save time and costs (Bosch 2009; Kilamo et al. 2012). An *Open Source Software Ecosystem* consequently is a software ecosystem based on an open source software and not proprietary software platform.

3.5.6 Adapted Onion Model

In virtue of the misleading use of all these terms, it seemed crucial to demonstrate the distinction between them. Due to this, a model adapted to the onion model (see Chapter 3.5.2) but using a wider scope, which seems logical while analyzing the different terms described in this chapter, will be developed in order to answer the first research question of this thesis (see Figure 3). With reference to the OSS community (see Chapter 3.5.2) it seems legitimate to allocate two varying meanings to the term. First, there is the overall OSS community from Lerner and Tirole (2002) that describes the denotation in ideological terms. This means that members of this ideological direction represent this idea but are not necessarily active in this business. This kind of OSS community will further be named as the *Open Source Software Collective* and represents the outermost layer of the onion. The several communities that operate under the principles of this overarching idea would describe the denotation in social terms. This means that all the community members are actively interacting with each other as described in the onion model (see Chapter 3.5.2). Further the OSS foundation and the OSS ecosystem can be used synonymous. In order to avoid confusion about these two terms, the term *Open Source Software Association* is introduced and will be used in the course of this thesis representative for both of them. As soon as governance comes into play an OSS community is no longer a community but rather an association. An OSS association is considered as the governmental framework around a community and determines its strategic direction. The governmental structure transfers the rather informal construct of a community into an enterprise-like form. In consequence, typical functions of a traditional enterprise such as a marketing department or a legal department need to be implemented. In sum, the crucial difference between an OSS community and an OSS association is the legal form. A community is a rather vague construct of collaboration where no one formally has binding rights and duties. In contrast, an association is usually a legal entity in the form of a non-profit organization with a governmental structure where members are obliged to pay membership fees.

The conclusive placement of the term OSS project within this model turned out to be rather difficult. It therefore seemed reasonable to implement two subgroups that differentiate between the two versions described in the paragraph above (see Chapter 3.5.3). The first subgroup is the *community* open source software projects and is referring to the open source software projects *within* an OSS community. In contrast to those, a stand-alone project falls into the category of the *ordinary* open source software projects. In concrete terms this means that a *community* open source software project is one of multiple projects within a OSS community that all pursue the same purpose, whereas an *ordinary* open source software project is an autonomous project. Each of those ordinary projects might be seen as a little community, however, an ordinary project might also be driven by one person, which would not yet correspond to a community since a community is composed of more than one person. As a result, the differentiating criteria between these two subgroups is the amount of projects and the pursued direction. Either way, the OSS projects consist of the technical base and the members participating in that particular project. The technical base is understood as the OSS platform that is usually represented by the operating system.

The conducted distinction of the different open source terms is derived from the literature review. However, it is based on the subjective perception of the author and therefore not conclusive. Besides the terms described in this chapter, there are further expressions to be found in literature. Due to their perceived lower relevance for this thesis, they are not further described at this point. Additionally, it needs to be mentioned that there are often rather blurred than distinct boundaries between the various expressions and that it is not always possible to clearly assign a construct to a term. There are many open source entities that consist of parts from more than one of these terms and represent a combination of several features. This conflation relativizes the validity of this distinction. In due consideration of this overlap the more general term *Collaborative Open Source Software Development Organization* will be used in the further course of this thesis.

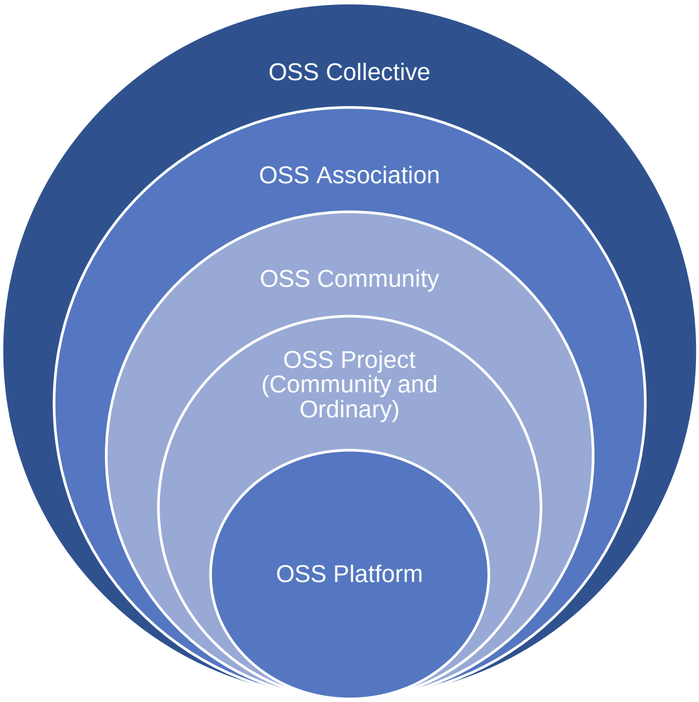


Figure 3: Adapted Onion Model.

4 Analysis of the GENIVI Alliance and the Polarsys Working Group

In Chapter 4 the two selected collaborative OSS development organizations are analyzed and introduced in more detail. The governance models of the GENIVI Alliance and the Polarsys Working Group are examined from the organizational, financial and legal perspectives. By doing this, the crucial points of how a collaborative OSS development organization is governed and built up are worked out.

4.1 *The GENIVI Alliance (GA)*

The following analysis is based on the expert interviews with the Executive Director of the GENIVI Alliance Steven Crumb, the Community Manager Jeremiah Foster, the Leader of GENIVI’s license review team Claus-Peter Wiedemann and the information provided by the organization on their website.

4.1.1 Introduction to the GENIVI Alliance

In order to create a basic understanding regarding the GENIVI Alliance, the key terms GENIVI and In-Vehicle Infotainment (IVI) are first described briefly in the following paragraph.

The term GENIVI is pronounced gee-nee-vee and consists of two parts. It is a combination of the words GENeva and In-Vehicle Infotainment. Symbolic of the emphasis of a peaceful cooperation among automaker, suppliers and technology providers, the Swiss city Geneva has been chosen because it is the international city of peace. IVI is the acronym of In-Vehicle Infotainment described in the paragraph below (GENIVI, FAQ 2014).

In-Vehicle Infotainment is the sum of vehicle entertainment and information applications. It describes the entirety of infotainment systems in a car including audio systems, news and multimedia access, navigation and location services, telephone or Internet services. In order to control the different services of an IVI system, the driver obtains support from the

Bluetooth technology or smartphones that can be controlled with the aid of voice commands, touch screen functionalities, knobs or other control devices built into the vehicle (Webopedia 2014).

The GENIVI Alliance is a non-profit corporation consisting of automotive manufacturers, suppliers and soft-, middle- and hardware providers initiated in 2009 by the companies BMW GROUP, Wind River, Intel, GM, PSA, Delphi, Magneti-Marelli and Visteon. The GENIVI Alliance's purpose is to advance the software development in the automotive industry through the use of a dynamic reusable IVI open source development platform. In order to achieve this goal the GENIVI Alliance aims at creating a common underlying framework composed of automotive original equipment manufacturer requirements, delivering specifications, reference implementations and certification programs on which further development of open source can be conducted (GENIVI, About GENIVI 2014). This common underlying framework is also called non-differentiating approach (see Crumb 2014). In other words, the GENIVI Alliance wants to develop *Commodity Software* (see Foster 2014). Commodity software means that there is a basic need of this particular software for all the market players operating in this industry and that there is neither rivalry nor competition related to this specific product. Due to the general need, it is vital to all of them to develop commodity software and bring it to market as fast and as inexpensive as possible. Therefore the long-term goals of the GENIVI Alliance are to shorten the development-cycle, to fasten time to market and to reduce costs for all involved parties developing In-vehicle Infotainment. According to Crumb (2014) the founding of the GENIVI Alliance is caused by two primary reasons - the continuously growing expectations and requirements of the end-consumer and the rapidly changing technology in the smartphone industry.

First, the amount of software equipment and applications in terms of information systems, navigation systems or entertainment systems implemented in a car increases steadily. As a result of this, the potential of driver distraction rises and this in turn leads to a need of higher safety measures. Along with the increasing functionality also the manufacturing and delivery costs of cars become more and more expensive. However, in contrast to the costs, the prices of automobiles do not significantly escalate. Indeed the car prices are increasing but not at the same level of speed as the costs.

Secondly, the automotive industry faces the problem of the fast moving industry of smartphone development. Since the smartphone industry started its triumph with the launch of the iPhone 3 from Apple in 2008 (Betschon 2014) people are used to small time interval updates, modifications and product launches that are brought to the market. Furthermore, they are accustomed to the interconnectedness, software provider offer for several devices. As a result, end-consumers expect the same technical equipment in their cars. However, end-consumers do not consider the fact, that the time to market of smartphones is significantly shorter than the time to market of cars. The latter takes up between three and five years whereas smartphones are developed and launched between twelve and eighteen months.

In order to meet the demand and fulfill customer needs the eight founding members merged into an alliance.

However, at the beginning the GENIVI Alliance was more of a commercial alliance than an open source organization. In other words, the driving force were not the actual basic open source principles but rather the cost-cutting (see Crumb 2014; Foster 2014). However, according to Crumb (2014) it is an ongoing process and the alliance members are continuously working on transferring the alliance into a fully open source organization.

Joining the GENIVI Alliance is open to all organizations whose business is related to IVI and who are interested in the successful further development of the platform (GENIVI, About GENIVI 2014). However, there are several automobile manufacturer and suppliers that are not members of the GENIVI Alliance. According to Crumb (2014) there is no selection of who is allowed to participate in the GENIVI platform. It is rather a question of whether particular organizations are interested in joining the Alliance or not. Crumb (2014) mentions two principal reasons why OEMs might deny a joining. He states *“there may not be a clear alignment of their organizational goals with the goals that GENIVI is attempting to meet - our mission of delivering an* *open source platform”* (see p. 90). Furthermore, Foster (2014) says *“they see open source rather conflicting with their business interests”* (see p. 95). This might be attributed to the traditional mindset of not sharing knowledge but rather develop proprietary software. Crumb (2014) as well states *“we are still working very hard at helping automotive organizations become comfortable with open source development. So there has been a lot of education, a lot of transition a lot of paradigm shift that has been necessary there and some automotive organizations are with it and understand it and some are still struggling with it because they are not used to giving software away. It’s how they make money, it is to sell software”* (see p. 93). *“The other thing,”* according to Crumb (2014) *“is that many organizations did not select the LINUX operating system as the basis for their IVI-systems and so they have less interest in participating”* (see p. 91). Since the GENIVI Alliance is based on the Linux kernel, which has been developed by Linus Torvalds in 1991 (Linux Foundation, About Us 2014), and the GNU operating system, the only precondition of becoming a GENIVI member is the compatibility with the Linux GNU system. Both the Linux kernel as well as the GNU operating system have been released under the General Public License (see Crumb 2014; see Chapter 3.3).

4.1.2 Organizational Perspective

This chapter deals with the questions of how the GENIVI Alliance is built up in managerial terms (see Chapter 4.1.2.1) and how the different members are classified (see Chapter 4.1.2.2).

4.1.2.1 Organizational Structure

The organizational structure of the GA consists of several teams and groups, which are subordinate to the board of directors (see Figure 4). The groups in turn are subordinate to the teams (GENIVI Bylaws 2011, Section 8, 8.1). The operations subcommittee and the program management office are affiliated with the board of directors and are comparable to supervisory authorities. In due consideration of some exceptions, the majority of the work in GENIVI is done by the member volunteers. Exceptions include leadership roles such as the executive director, a program management lead or the community manager (see Crumb 2014).

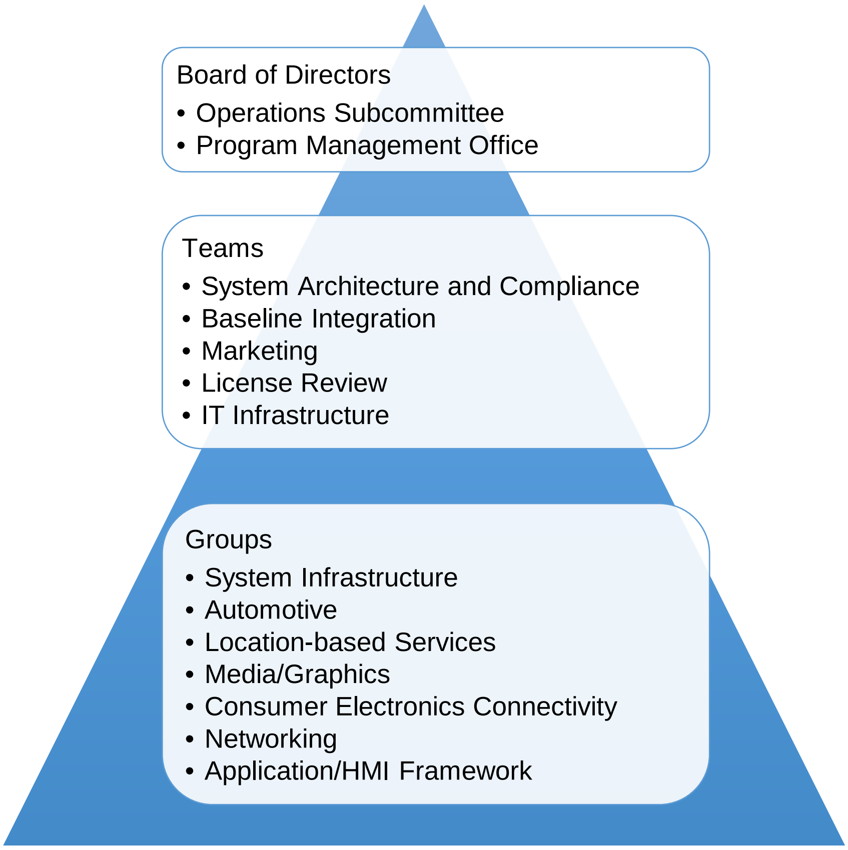

Board of Directors
• Operations Subcommittee
• Program Management Office
Teams
• System Architecture and Compliance
• Baseline Integration
• Marketing
• License Review
• IT Infrastructure
Groups
• System Infrastructure
• Automotive
• Location-based Services
• Media/Graphics
• Consumer Electronics Connectivity
• Networking
• Application/HMI Framework

Figure 4: Organizational Structure of the GENIVI Alliance.

Board of Directors

The board of directors is composed of representatives of the founding charter members, charter members and core members. Each founding charter member is entitled to a permanent seat, which means that they have a perpetual right to appoint a representative to the board of directors for a time period of one year (GENIVI Bylaws 2011, 4.6 (a)). Based on a particular election procedure the remaining board seats will be filled by charter members or core members (GENIVI Bylaws 2011, 4.4). The term of a director in an elected charter member seat or elected core member seat will be two and one year respectively. Currently, the board of directors is composed of nine representatives. The minimum number is three, the maximum number 15. Required officers are a president, a vice president, a secretary and a treasurer (GENIVI Bylaws 2011, 7.1).

The board of directors is responsible for legal and fiduciary issues and gives an overall strategic direction that is operatively implemented by the groups and teams. Duties such as the review of the annual budget, the evaluation of the Alliance's fulfillment of its purposes or the adoption and modification of the bylaws need to be fulfilled by the board of directors (GENIVI Bylaws 2011, 4.2 (l), (m), (p).

Operations Subcommittee

The operations subcommittee is a supervisory authority of sorts and occupies a support function towards the board of directors. It assists the board of directors in operational issues and monitors the annual strategic direction set by the board of directors with due diligence. The operations subcommittee works in close-collaboration with the program management office and the different teams. Its task is to increase the Alliance's efficiency and effectiveness by decreasing the number of used resources as much as possible and by meeting stakeholder expectations. With the assistance of the executive director, the chairman of the board is in charge of the operations subcommittee.

Program Management Office

The program management office takes on the role of a supervisory authority with regards to current and future releases' progress. They are monitoring, tracking and reporting programs' status.

Teams

System Architecture and Compliance Team

The system architecture team is responsible for the setup and the processes of the entire GENIVI platform and defines the way the different system components are connected and interact with each other. They further execute platform releases, are concerned with security and quality measures and define the compliance statement[1] (GENIVI Intellectual Property Rights Policy 2012, Section 1 d)) that describes requirements in relation to the GENIVI code (GENIVI Intellectual Property Rights Policy 2012, Section 1, (d), (e); GENIVI, Functional Organization Chart (2014).

Baseline Integration Team

The baseline integration team is part of the system architecture team. Its task is the definition of processes, policies and tools for the development of GENIVI software (GENIVI, Functional Organization Chart 2014).

Marketing Team

The marketing team has two main tasks. On the one hand it is responsible for the external perception of the Alliance and how it is presented in public, on the other hand it provides a platform for Alliance members to promote their products and services outside their organizational boundaries. In order to meet both requirements of the individual GENIVI members as well as of the entire Alliance, the marketing team sets its focus on three different aspects. There are the deliverables that represent the interests of the entire Alliance, such as the Alliance business development and ecosystem awareness. Secondly, there is the outreach approach focusing on potential synergies and collaborations with other organizations and on the recruitment of new Alliance members. The third aspect focuses on communication which means for example to manage membership marketing support, coordinate events and worldwide presence of the GA or maintain the public facing website (GENIVI, Functional Organization Chart 2014).

License Review Team

The main focus of the license review team lies on the license compliance that ensures that the different software components which are contributed to the GENIVI Alliance correspond to the licensing and the copyright policies. The team therefore defines procedures and creates legal documents that

[1] A specification describing the mandatory and optional requirements for compliance with a Final Specification and the software components contained in the Genivi Code that implement those requirements.

guarantee that the Alliances technical work is based on an open source appropriate legal base. They further occupy the position of a consulting center where GENIVI members get the possibility to seek advice in view of licensing questions. The license review team consists of five to ten team members, most of which have a technical background and a few of them a legal educational background (see Wiedemann 2014, Public Policy for GENIVI Licensing and Copyright Version 1.5 2014).

IT Infrastructure Team

The IT infrastructure is a collection of application and programming software that is hosted and maintained by the IT infrastructure team. Its purpose is the support of collaborative work between members across the entire GENIVI Alliance (GENIVI, Functional Organization Chart 2014).

Expert Groups

There are seven (System Infrastructure, Automotive, Location-based Services, Media/Graphics, Consumer Electronics Connectivity, Networking, Application/HMI Framework) expert groups whose responsibility is the definition of requirements related to their respective domain and the management of the integration process from software components into the GENIVI baseline and compliance statement (GENIVI, Functional Organization Chart 2014).

4.1.2.2 GENIVI Members

In total the GENIVI Alliance registers over 160 members. These member companies can be distinguished from each other by two criteria. Either the members are viewed by their industry or viewed by their member level (GENIVI, GENIVI Members 2014). In order to join the GENIVI Alliance a company needs to complete the application form and the participation agreement (GENIVI, Instructions for Obtaining Membership 2012).

Members sorted by their Industry

There are four different types of members if sorted by their industry (GENIVI, GENIVI Members 2014).

1. The Original Equipment Manufacturer (OEMs) are represented by approximately 10 companies, such as the BMW Group, Volvo or Renault
2. The member group of First Tiers[2] (Sarokin 2015) consists of more than 20 companies, such as Continental, Bosch or Magneti Marelli
3. The OSV (Operating System Vendors), Middleware, Hardware and Service Suppliers are the largest member group. They are represented by over 100 member companies, such as IBM, TomTom or BlackDuck.
4. The Silicon Vendors are represented by approximately 20 companies, such as Intel, ISSI or the Vivante Corporation.

Members sorted by their Member Level

In this category the members are, regardless of their industry-affiliation, classified into three member levels (GENIVI, GENIVI Members 2014). The differentiation depends on the degree of participation the members are interested in. As a consequence, there are varying annual membership fees as well as membership benefits depending on the category the member companies belong to. The fees members from different levels are obligated to pay lie between USD 5,000 and 150,000. The membership benefits are multifaceted and range from the right to be listed on the Alliance website via getting access to test tools and certification programs trough to obtaining eligibility for participating in the various teams, groups or even in the board. The three different degrees of participation are the following (GENIVI, Join the GENIVI Alliance 2014):

The Founding Charter and Charter Members

The first category is also referred to as the top-tier member category and is the most influential level of participation. Members that pay an annual membership fee of USD 150,000 get more influence in terms of the

[2] The term is especially common in the automobile industry and refers to major suppliers of parts to OEMs.

formulation of the strategy and the direction of the financial modes (see Crumb 2014). These members take advantage of all the provided benefits of the GA. Advantages that are reserved for founding charter and charter members only, are to receive a permanent board seat, to be entitled to propose new expert groups or to be eligible for officer positions. Further benefits are the eligibility to participate and / or lead teams and groups, to be part of several marketing measures and activities and to have various access rights (GENIVI, Join the GENIVI Alliance 2014).

These membership benefits remain reserved for merely four members, which are the two OEMs the BMW Group and the PSA Peugeot Citroën, the first tier Magnetti Marelli and the software provider XSe (GENIVI Members 2014).

Core Members

The second category focuses more on leadership issues. Core members pay an annual membership fee of USD 25,000 and are more involved in the organization in the sense of technical leadership and leadership in the delivery of GENIVIs output. Core members basically benefit from the same advantages as the members of the first category. The only difference is that core members, for taking advantage of some benefits, need to meet some requirements such as being an elected board member, an expert group lead or an architect (GENIVI, Join the GENIVI Alliance 2014). Companies of this category, with in total 34 members, are for example the semiconductor provider Intel, the software provider Pelagicore or the navigation system manufacturer TomTom (GENIVI Members 2014).

Associate Members

Members of the third category do not always participate in the development of software. They pay an annual membership fee of USD 5,000 and often aim for visibility and for being part of the business network. Visibility in this case means, that the company itself is being recognized by the other organizations in this industry. This might be supported by the membership benefits that entitle them to participate in Alliance-sponsored developer forums as well as to be listed on the Alliance website. However, it cannot be acknowledged as a fact that all the associate members are free riders[3]. There are also associate members that are highly involved in the organization and develop products of high quality (see Crumb 2014). Associate members profit from less than half of the benefits the two other membership categories profit. They are eligible to participate in the marketing or license review team or the expert groups, can take part in all the marketing activities and have some access rights (GENIVI, Join the GENIVI Alliance 2014). There are 122 members operating at the lowest level of participation. These are for instance the provider of open source software solutions BlackDuck, the hardware, software and IT-services provider IBM or the electronic company Pioneer (GENIVI Members 2014).

4.1.3 Legal Perspective

The GENIVI Alliance is organized according to section 501(c)(6) of the United States Code. This means that the GENIVI Alliance is a non-profit corporation and is tax-exempt (Articles of Incorporation of GENIVI Alliance, Article III, (1), (3)).

The most important legal documents of the GENIVI Alliance are the bylaws, the intellectual property rights policy and the participation agreement (see Figure 5).

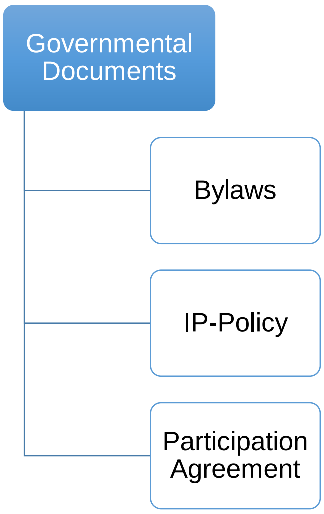


Figure 5: Governmental Documents of the GENIVI Alliance.

[3] Free riders are understood as companies that only participate in the organization to benefit in terms of marketing purposes or using newly developed technologies, but do not contribute constructive inputs themselves.

Bylaws and Intellectual Property Rights Policy

The GENIVI bylaws and the intellectual property rights policy are the legal foundation of the Alliance. The GENIVI bylaws are the general rules of cooperation by which the organization is managed (see Crumb 2014). This includes, among other things, definitions of crucial terms, the purpose of the GA, the rights and duties of the different organizational entities, electoral procedures, financial management issues, record and report guidelines, participant classifications or confidentiality issues (GENIVI Bylaws 2011).

The IPR-policy is an addendum to the bylaws and is composed of eight sections[4] (GENIVI Intellectual Property Rights 2012, Section 1 – 8). In general the IPR-policy regulates what happens to the intellectual property rights that originates within the frame of the GA or are contributed by participants (see Crumb 2014, Wiedemann 2014, GENIVI Bylaws 2011). Thereby the most relevant part of the policy seems to be *Section 3 Licensing of Intellectual Property Rights*, *Paragraph (a) Patent License and (e) Copyright License to Participants for a Final Specification*[5] (GENIVI Intellectual Property Rights 2012, Section 1 h)).

Section 3, Para. (a) says: *[…] each Participant hereby grants to the Alliance and all other Participants, under royalty free and other reasonable terms and conditions that are demonstrably free of any unfair discrimination, a non-exclusive, non-transferable, irrevocable (except upon breach by licensee), sublicenseable (through multiple tiers of sublicensees), worldwide license (without compensation) […].*

In other words, the patent license grants that the participant still owns the intellectual property rights but also grants to the GA and all its members the same rights. Those rights are making, using, importing, offering for sale,

[4] Definitions, Approval of Draft Specifications and Reference Implementations, Licensing of Intellectual Property Rights, Trademarks, Survival of Agreement to Grant License, Exception for Matters Out-of-Scope, Invention Agreement, Choice of Law

[5] Any document, including any updates or revisions, approved as a Final Specification by the Board of Directors in accordance with Section 2 hereof that is a (i) Component Statement, (ii) test specification, (iii) test plan, or (iv) Compliance Statement or certification program embodying technical requirements, interoperability requirements, conditions, protocols, testing policies and testing procedures.

leasing, licensing, selling and otherwise distributing Compliant Portions[6] (GENIVI Intellectual Property Rights 2012, Section 1 a)) related to this intellectual property.

The section about the copyright license conveys the same basic statement as paragraph (a) - that the intellectual property is still retained by the licensor. However while paragraph (a) deals with the rights of what can be done with the licensed product, the copyright license refers to what is allowed to change: *Each Participant hereby grants to the Alliance and each other Participant a worldwide, irrevocable, non-exclusive, non-transferable (except as otherwise provided in the Bylaws), sub-licensable (through multiple tiers of sublicensees), royalty-free copyright license to reproduce, create derivative works of, distribute, display, and perform the Contributions of the Participants […].*

Further paragraph (d) regulates the consequences of a conflation of several contributions: *[…] Subject to the Participant's copyright ownership in their Contributions, the Alliance shall own all right, title, and interest in the compilation of Contributions forming the Final Specifications and related works […].* This means that the participant solely owns the right of its individual contributed part but has no influence on what happens after combining it with other contributions.

Participation Agreement

The participation agreement is the governing document between the GA and a member organization (see Crumb 2014). It mainly deals with membership-fees and consequential payment conditions. By the signature of the participation agreement the company agrees *"to be bound by the terms hereof including full payment of the applicable membership fee, as well as the terms and conditions stated in the Articles of Incorporation and, Bylaws ("Organizational Documents") of the GENIVI Alliance as may apply to the Founding Charter Charter/Core/Associate participation classification stated in the Bylaws"* (GENIVI Alliance Participation Agreement 2014).

[6] Only those specific portions of products (hardware, software or combinations thereof) that: (i) implement and are compliant with all relevant portions of a Final Specification, and (ii) are within the bounds of the Scope.

4.1.4 Financial Perspective

There are three types of main costs the GENIVI Alliance has to cover (see Crumb 2014). First, there are labor costs that arise out of the little amount of contracted employees mentioned in Chapter 4.1.2.1. A second matter of expense are operating costs. Those include financial management issues as well as the membership processing. According to Crumb (2014) the GA *"provides a fairly comprehensive collaborative infrastructure"* (see p. 93) which includes Wikis, code or issue trackers for example. Thirdly, there are several events that need to be funded by the Alliance. On the one hand those are internal events like the all member meeting that takes place twice a year and on the other hand those are external events such as industry events the GA participates (see Crumb 2014).

These costs are mainly covered by the membership-fees the member companies are bound to pay annually. Another source of revenue might be charges that members need to pay in order to participate in member showcases or marketing events. However, according to Crumb (2014) "*the membership fee makes up for about 95% of the operating revenue"* (see p. 93).

4.1.5 Adapted Onion Model

According to the findings in this chapter the governance model of the GENIVI Alliance will be adapted to the adapted onion model developed in Chapter 3.5.6 (see Figure 6). The entire open source collective is no subject to change. Due to the fact that this solely describes an ideological direction, each and every one that supports the open source movement in a positive way automatically belongs to the open source collective. The GENIVI Alliance including all its components and the corresponding governance model based on legal documents, regulations and guidelines can be understood as the open source association that acts as a steward of the underlying entities. The open source community describes all the GENIVI members that would not be bound to each other without the legal umbrella of the association. As its name implies, the GENIVI open source projects are synonymous to the open source software projects whereby they are referring to the community OSS projects since they are within the community and not autonomous. The technical base of the GENIVI Alliance is the GNU operating system with the Linux kernel.

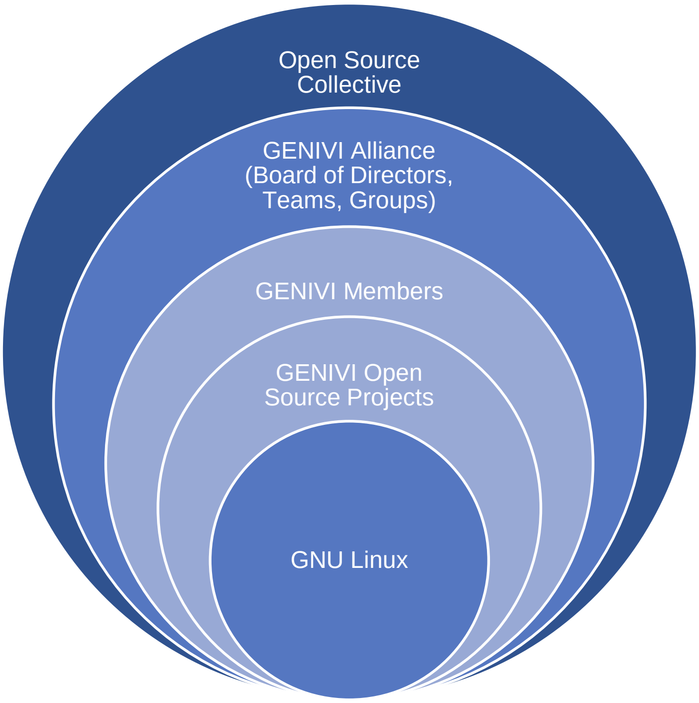

Open Source Collective
GENIVI Alliance (Board of Directors, Teams, Groups)
GENIVI Members
GENIVI Open Source Projects
GNU Linux

Figure 6: Onion Model Adapted to the GENIVI Alliance.

4.2 Eclipse Foundation (EF): The Case of the Polarsys Working Group

The following analysis is based on the expert interview with the Managing Director of the Eclipse Foundation Europe GmbH Ralph Mueller and the information provided by the organization on their website.

4.2.1 Introduction to the Eclipse Foundation and the Polarsys Working Group

The Eclipse Project (EP) grew out of a closed-source development project inside IBM and was founded by IBM in 2001. Three years later the Eclipse Foundation (EF) was the result of the idea to build a neutral, independent organization to govern the Eclipse Project. In order to create a neutral governance, several other large companies besides IBM got involved. The Eclipse Foundation was therefore initiated as an open source non-profit corporation whose purpose was the creation of a sustainable collaborative open source association as well as the promotion of Eclipse and open source technology in commercial and open source solutions. The members of the Eclipse community are individuals as well as organizations from the software industry, interested in creating a dynamic open source ecosystem based on the Java platform (see Mueller 2014; Eclipse, About Us).

Inspired by the Aerospace industry, the Working Groups (WG) have been initiated by the Eclipse Foundation in 2011. The Working Groups are affiliated with the Eclipse Foundation. They leverage the Eclipse governance, process and infrastructure to develop and promote domain specific open source projects or solutions. The purpose of the WGs is the advancement of open source based industry collaborations through providing best practices of open source development and basic services required for open innovation. The aim of those provided services is the reusability in order to facilitate the process of building a new open source organization. Companies should not be forced to create a new organization from scratch, neither from a technical perspective nor a governmental perspective (see Mueller 2014; Eclipse, Eclipse Working Groups 2014). In concrete terms those services are:

1. Vendor-Neutral Governance

The vendor-neutral governance is based on several legal documents such as bylaws, agreements and policies, which support emerging collaborations in terms of decision-making and conflict management.

2. Intellectual Property Management

The IP policies of the Eclipse Foundation serve as fundamental document to the Working Groups. This means that new established WGs can simply adopt the given IP policies.

3. IT Infrastructure

There is no need for new Working Groups to build up a completely new IT infrastructure. The Eclipse Foundation provides and manages all the resources that are required in order to maintain the Working Group IT environment.

4. Development Process

Based on the common open source development principles of free accessibility and reciprocity the Eclipse community has built a development process especially for collaborations with a high amount of different organizations.

5. Ecosystem Development

The ecosystem development service can be considered as marketing and promotion service. The purpose of this service is to spread the idea of the Working Groups through several marketing events, conferences or member meetings in order to enlarge the community and create a thriving open source ecosystem (Eclipse, Eclipse Working Groups 2014).

The managing director of the Eclipse Foundation Europe GmbH, Ralph Mueller (2014) entitles the provision of these services as the "*Out of the Box-Principle"* (see p. 101).

An example of a Working Group is Polarsys. Polarsys grew out of the need of the aerospace industry to create a development platform for aerospace-software. Since the products of the aerospace-industry are subject to a multiple decades long product life cycle, there was a need for a platform that is able to provide an appropriate IT infrastructure to sustainably host software projects. The companies had two options in order to solve their problem.

Either they would build an independent organization completely from scratch or they would negotiate with an already existing software development organization. The companies opted for the latter, given the Eclipse Foundations' seven years of experience in the business at that point in time. Thus the aerospace-consortium proposed to the Eclipse Foundation to join the foundation as sort of a subgroup. By accepting this suggestion the EF initiated the first Eclipse Working Group (see Mueller 2014).

4.2.2 Organizational Perspective

This chapter deals with the questions of how the Polarsys Working Group is built up in managerial terms (see Chapter 4.2.2.1) and how the different members are classified (see Chapter 4.2.2.2).

4.2.2.1 Organizational Structure

Six different committees that are entrusted with different tasks and responsibilities govern the Polarsys Working Group (see Figure 7) (Eclipse, Polarsys Working Group Charter 2012).

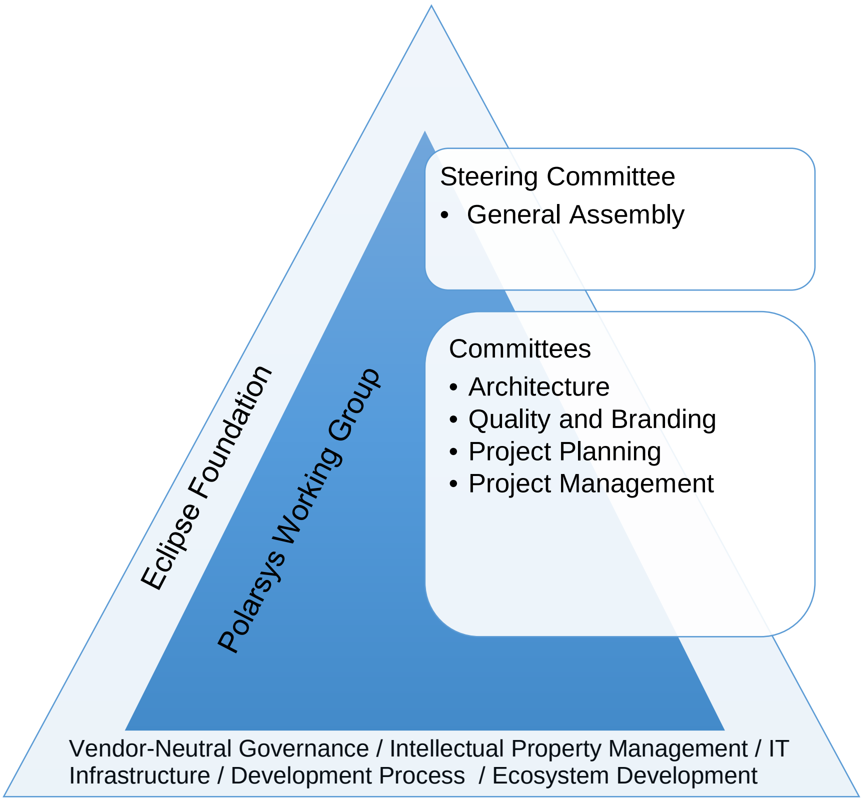


Figure 7: Organizational Structure of the Polarsys Working Group.

General Assembly

The only authority of the general assembly is the approval of changing the name of the Working Group. There is one seat for each steering committee and participant member (see Chapter 4.2.2.2) on the General Assembly (Eclipse, Polarsys Working Group Charter, General Assembly 2012).

Architecture Committee

The architecture committee is responsible for various aspects concerning the successful management of projects. They need to develop technical guidelines, review the projects in due consideration of these guidelines and ensure that those projects meet the expected requirements. The validation of new project proposals and the recommendation of technologies to the project planning committee fall in the responsibilities of the architecture committee as well. Further the architecture committee must ensure the achievement of Very Long Term Support (VLTS) objectives.

The architecture committee is composed of one seat for each steering committee member and one seat for each project planning committee member. Among the members of the architecture committee a chairman is elected whose task is to report to the steering committee (Eclipse, Polarsys Working Group Charter, Architecture Committee 2012).

Quality and Branding Committee

The quality and branding committee is responsible for the development, the continuous application and the validation of the maturity assessment program and the qualification kit. The purpose of the maturity assessment program and the qualification kit is to ensure that the components developed by the Polarsys WG correspond to the demanded quality requirements. In concrete terms the task of the maturity assessment program is to assess the maturity of components that are still in the development phase based on a particular scale. The qualification kit consists of several documents, such as development or test plans. These are necessary when Polarsys' components are implemented into certified embedded software and need to correspond to the particular specification (Eclipse, Polarsys Working Group Charter, Services 2012).

Project Planning Committee

The project planning committee's main task is to set up a strong relationship between the various stakeholders, whether developer, industrial user or researchers, of its open source projects. The committee is responsible for the management of several projects and takes care of rather medium to long, such as long-term development plans or user's needs, than short-term issues (Polarsys/TLPProposal 2012, Section 5).

Project Management Committee

The project management committee is a supervisory authority of sorts and is responsible for the proper and effective procedure of projects. In order to fulfill this task the project management committee predetermines an overall direction and prevents a project from failing by conducting a successful conflict management. Further, the project management committee is responsible for the compliance with open source principles and that possible subprojects also underlie the Eclipse IP policy and procedures. Usually the project management committee is dealing with only one project at the same time and is involved in short term issues (Polarsys/TLPProposal 2012, Section 4).

4.2.2.2 Polarsys Members

In total the Polarsys Working Group registers over 10 members. These member companies can be distinguished from each other by three criteria. Either the members are viewed by their industry, by their class of membership within the Eclipse Foundation or by their class of membership within the Polarsys Working Group (see Table 3).

Members sorted by their Industry	**Members sorted by their Class of Membership within the Eclipse Foundation**	**Members sorted by their Class of Membership within the Polarsys Working Group**
Industrial User	Strategic Members	Steering Committee Members
Tool Providers	Enterprise Members	Participant Members
Academics	Solutions Members	Guests
	Committer Members	Committers
	Associate Members	

Table 3: Three Criteria to Sort Polarsys Members.

Members sorted by their Industry

Considering the industries wherein the members conduct business, there are three different types of members (Polarsys, Members 2014).

1. The industrial user are represented by Airbus, Airbus Defence and Space, Airbus Helicopters, Ericsson and Thales.

2. The tool providers consist of the companies Artal, Atos, CEA List, Combitech, ESI group, Obeo, Soyatec and Zeligsoft.

3. The academics are the École Polytechnique Montréal and the University of Skövde.

Academics such as universities or research institutes occupy a special role within the Working Groups. Their participation in a Working Group is free of charge and mostly happens at explicit desire of the particular Working Group. The purpose of the collaboration between the academics and the WGs is to merge their knowledge and experience and to implement new methods and procedures that grew out of research (see Mueller 2014).

Members sorted by their Class of Membership within the Eclipse Foundation

The Eclipse Foundation differentiates between five classes of membership - the associate members, the solutions members, the enterprise members, the strategic members and the committer members (see Figure 8) (Eclipse, Types of Membership 2014). Having regard to the fact that only members of the solutions members class and upwards are allowed to participate in a Working Group (see Mueller 2014), this thesis will further not elaborate on the associate and committer members.

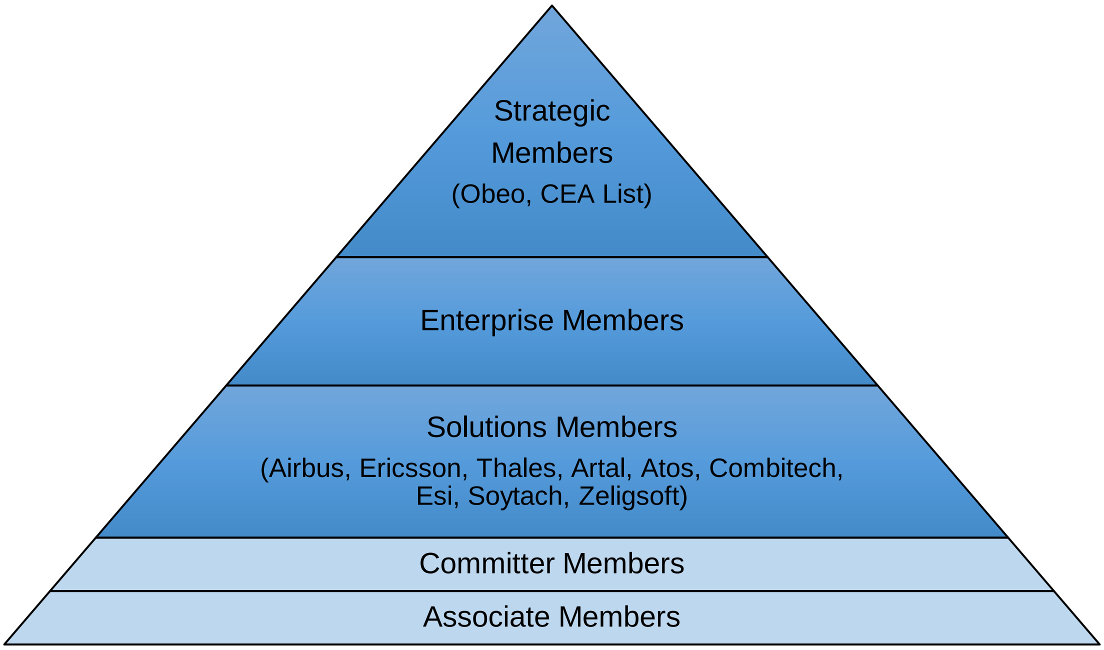


Figure 8: Polarsys Members Sorted by their Class of Membership within the Eclipse Foundation.

Solutions Members

Organizations belonging to this membership group incorporate Eclipse into their strategy and commit themselves to contributing an Eclipse-based product or service within twelve months after joining the foundation. They are further committed to publicly reveal their membership to the Eclipse Foundation for instance through a press release. In return, solutions members receive a number of benefits, such as diverse voting rights (bylaw or membership agreement amendments, project review, etc.) or a privileged treatment regarding sponsorships. Solutions members are entitled to participate in Working Groups. Based on the revenue of the company the annual membership fee varies from USD 1,500 to USD 20,000. Except the software provider CEA List and Obeo, which are strategic members, all the companies (Airbus, Ericsson, Thales, Artal, Atos, Combitech. Esi, Soytach, Zeligsoft) involved in the Polarsys Working Group are solutions members (see Figure 8) (Eclipse, Membership Rights 2014; Eclipse, Membership Types 2014).

Enterprise Members

Due to the fact that enterprise members mostly develop and use projects, products or services based on Eclipse, they are technologically as well as strategically reliant upon the Eclipse Foundation. In addition to the solutions members' benefits, they have wider access to a variety of information and are entitled to initiate or help initiate both open source projects and working groups. Unlike solutions members, enterprise members have the right to become a steering committee member of a Working Group. Their dues add up to USD 125,000 annually (Eclipse, Membership Rights 2014; Eclipse, Membership Types 2014).

Strategic Members

There are two types of strategic members – strategic developers and strategic consumers. Strategic developers are major contributors of technology to Eclipse and strategic consumers are major users of Eclipse technology. Through their right to a seat on the board of directors of the

Eclipse Foundation and on the foundation councils, they have power at the highest strategic and technological level. Related to Working Groups, strategic members have the same entitlements as the enterprise members. Membership fees are charged by reference to their annual revenue. For strategic developers the annual fee is between USD 25,000 and 250,000 and for strategic consumers between USD 50,000 and 500,000 (Eclipse, Membership Rights 2014; Eclipse, Membership Types 2014).

Members sorted by their Class of Membership within the Polarsys Working Group

Polarsys provides four different classes of membership – the steering committee members, the participant members, the guests and the committers (Eclipse, Polarsys Working Group Charter 2012).

Polarsys Steering Committee Members

Organizations belonging to the first category perceive Polarsys from a more strategic point of view and are interested in sustaining and supporting the WGs activities. Typically, this category consists of industry users of Polarsys products. Steering committee members pay an annual membership fee between USD 20,000 and 30,000. Steering committee members have all the provided benefits by the Polarsys WG except the write access to the open source code repository. Members of this category are the Airbus Group, CEA List, Ericsson and Thales (Eclipse, Polarsys Working Group Charter 2012).

Polarsys Participant Members

Organizations that are registered as a participant member use their membership in a more operational way. They are interested in developing the Polarsys ecosystem and provide products and services based on Polarsys. Typically, this category consists of service providers for Polarsys-specific technologies. Participant members pay an annual membership fee between USD 1,500 and 10,000. For the great part, participant members profit from the same services as the steering committee members. The slight difference between these two is the necessity of the participant members to be elected in order to be a member of the committees while committee steering members are automatically part of all of them. Further, they are not able to host custom build on the Working Group Infrastructure. This is a service that enables members to use the Polarsys test and build infrastructure for the creation of customized bundles that are for the creating members use only. Participant members are Artal, Atos, Combitech, Esi, Obeo, Soyatec and Zeligsoft (Eclipse, Polarsys Working Group Charter 2012).

Guests

The third category is composed of organizations that participate in the Working Group for one year at the steering committee’s invitation. Such guests might be academic, R & D partners or potential future members that are not yet sure about their definite accession. The only requirement towards guests is the signing of a participation agreement. They do not have to pay a membership fee (Eclipse, Polarsys Working Group Charter 2012).

Committers

Committers are individuals who have write access to the source repositories and other content on the Eclipse Foundation’s website. Since they have access rights on the highest level, committers are a special type of membership with an extraordinarily high responsibility. Due to this, in order to become a committer, an elaborate nomination and approval process needs to be undergone. Like the guests, they are excluded from paying a membership fee (Eclipse, Polarsys Working Group Charter 2012).

Application Process

As mentioned in the paragraph above, the precondition in order to join a Working Group is to hold at least the solutions members status. This means that a company interested in joining the Polarsys Working Group either has to be a member of the foundation already or, if not, needs to apply for membership. In order to be admitted into the EF the application form and the membership agreement need to be completed and submitted to the Eclipse

Foundation. After the successful admittance the members obtain the opportunity to join a Working Group through signing the participation agreement (see Chapter 4.2.3).

4.2.3 Legal Perspective

The Eclipse Foundation is based on several legal documents that determine the collaboration between the different members involved in the activities of the organization. In view of the large number of legal documents, including several agreements, licenses, guidelines and policies, it would go beyond the constraints of this thesis to elucidate these various documents. Therefore only those documents relevant with reference to Working Groups will be dealt with in the following paragraph (see Figure 9).

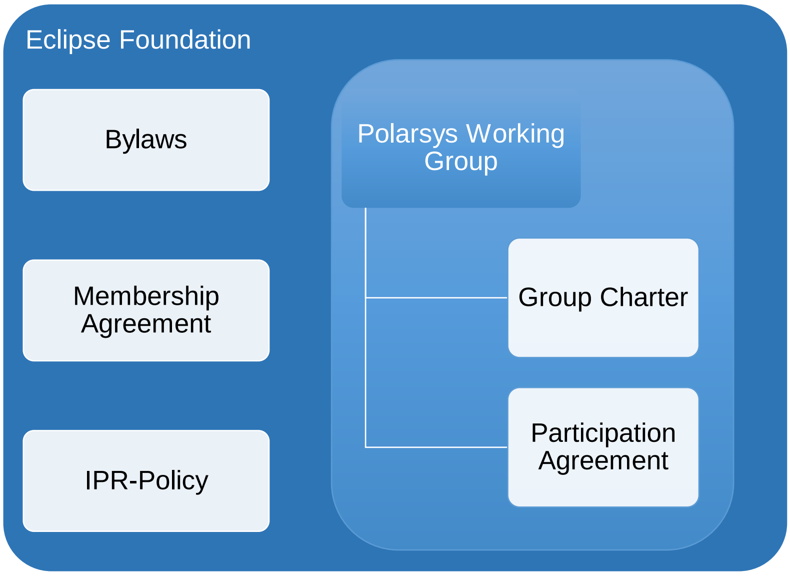


Figure 9: Governmental Documents of the Eclipse Foundation and the Polarsys Working Group.

Bylaws and Intellectual Property Rights Policy

The bylaws and the intellectual property rights policy build the legal basis of the Eclipse Foundation and with that of the Eclipse Working Groups. All necessary documents required in order to set up a foundation juridical compliant need to correspond to the bylaws and the IPR-policy (see Mueller 2014).

Summarized, the bylaws are concerned with the purpose of the Eclipse Foundation, the rights and duties of the board of directors, the committees of the boards and the officers, the membership classifications and the management of the Eclipse Foundation (Eclipse Foundation, Inc., Bylaws 2014).

The intellectual property rights policy provides guidelines relating to the acceptance, the redistribution and the hosting of content and other intellectual property issues (Eclipse Foundation, Inc., Intellectual Property Policy 2011). It states "*the EPL shall serve as the primary license under which the Eclipse Foundation shall accept Content (…)".* The EPL regulates the copyright and the patent licensing whereby "*(…) each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free",*

either

"patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution" (…) (Eclipse Public License 2014, V 1.0, 2. Grant of Rights, b)),

or

"copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution" (…) (Eclipse Public License 2014, V 1.0, 2. Grant of Rights, a)).

Not included to the Contributions[7] are *"(…) additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program" (*Eclipse Public License 2014, V 1.0, 1. Definitions)

In other words, as long as the add-on is an individual module or not a derivative work, it is excluded from paragraph 2. a) and b) of the EPL and can be distributed under another license, be it an OSS or a proprietary license (Eclipse Public License, FAQ 2014).

[7] a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
i) changes to the Program, and
ii) additions to the Program; where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf.

Membership Agreement

The membership agreement is a legally binding contract that governs the relationship between the Eclipse Foundation and the Eclipse members in general. This document describes the purpose and the goals of the foundation, determines the six possible membership classes (strategic developer, strategic consumer, enterprise, solutions, committer and associate members) as well as the rights and obligations of those (Eclipse Foundation, Inc., Membership Agreement 2010). By signing the membership agreement companies agree automatically to the IPR-Policy (Eclipse Foundation, Inc. Intellectual Property Policy 2014).

Participation Agreement

The participation agreement governs the relationship between a Working Group and its members. This agreement is based on the general template *Working Group Participation Agreement* that can be adapted to the specific needs of each WG (see Mueller 2014; Polarsys Industry Working Group Participation Agreement 2012).

The participation agreement of a WG determines the fee a Working Group member is obligated to pay annually. This annual Working Group fee is composed of the membership due that covers the basic services the Eclipse Foundation provide and the participation fee whose purpose is to cover possible additional services. Such additional services might be the use of the Long Term Support Infrastructure (LTS) or the recruitment of additional employees (see Mueller 2014).

Further the participation agreement defines the participation level. Each Working Group is free to choose which and how many different membership classes (steering committee, premium, participating, user, service provider members, etc.) they want to provide. According to their membership class the companies have different rights and duties (see Mueller 2014).

Group Charter

The group charter is a specification in addition to the general legal documents. It can be individualized according to the needs of a specific

Working Group as long as they are not in contravention to the bylaws and the IPR-policy (see Mueller 2014). The group charter registers the goals and the vision of the WG, the core domains represented in the WG, the utilized EF services, the different membership classes and their corresponding rights and benefits and the various organizational entities (Eclipse, Polarsys Working Group Charter 2012).

4.2.4 Financial Perspective

The funding of the Polarsys Working Group consists of two parts. The first part is the Eclipse membership fee the WG members are required to pay, as the minimum membership level for an Eclipse Working Group is the Eclipse solutions membership. This fee is between USD 5,000 and USD 20,000 and covers the basic-services provided by the Eclipse Foundation. In a further step, the Working Group decides on charging an additional fee for extended services. Extended services are understood as services that exceed the provided standard capacity of the EF basic services. Extended Services might be additional employees such as an extra product manager for a specific task. Further the Polarsys WG is highly interested in and reliant upon a long-term support infrastructure since they must maintain their software for more than 60 years. For this service an extra charge is claimed (see Mueller 2014).

4.2.5 Adapted Onion Model

According to the findings in this chapter the governance model of the Polarsys Working Group will be adapted to the adapted onion model developed in Chapter 3.5.6 (see Figure 10). The Eclipse Foundation including the provided services for the Working Groups and the legal necessary basics in order to run such an organization can be seen as the framework around the entire construct and therefore as the open source association. Further, the Eclipse Foundation members represent the open source community contributing and developing software components. In order to adapt the structure of the Eclipse Foundation to the model in Chapter 3.5.6 there is a need of bringing an additional layer into the onion - an Open Source Subcommunity. This subcommunity in the case of the

Eclipse Foundation is represented by the Working Groups whose members are part of the main community. Equally to the GA the open source projects are defined as the software development projects within the community. At this point, it might be argued to introduce two different kinds of projects, namely the Eclipse Foundation projects open to all Eclipse members and the Working Group projects open only to WG members. Since this thesis deals primarily with the Working Groups and not with the entire Eclipse Foundation, this will be omitted. The IT infrastructure, in the form of the Java platform, provided by the EF serves as the technical base of the entire association.

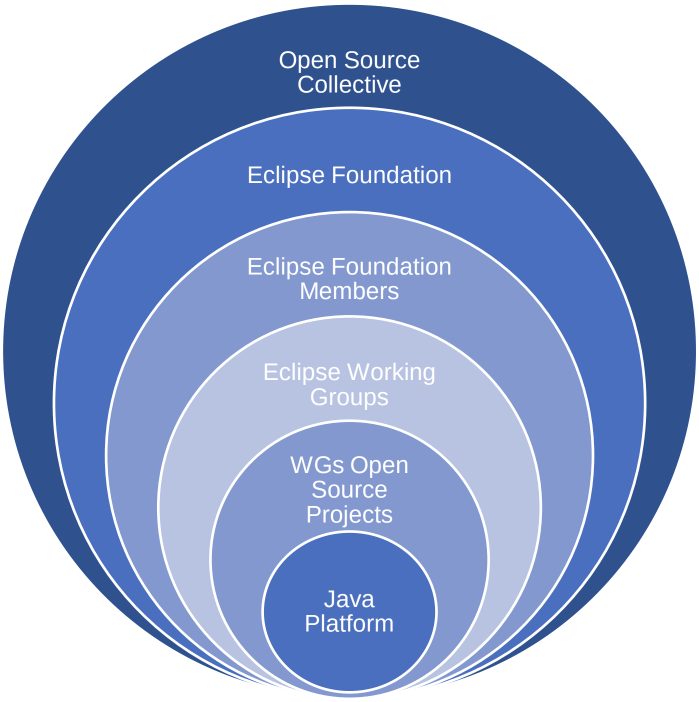


Figure 10: Onion Model Adapted to the Eclipse Foundation and the Polarsys Working Group.

5 Findings and Implications

Chapter 5 shall answer the second research question defined in Chapter 1.3: *What different types of collaborative open source software development organizations do currently exist and which criteria need to be considered in order to build a collaborative open source software development organization?*

In order to answer this question, a comparison between the GENIVI Alliance and the Polarsys Working Group is drawn. The overall aim as well as the governance model will be examined from an organizational, financial and legal perspective. In a second step, the key characteristics of the distinct types of collaborative OSS development organizations resulting from the comparative analysis are worked out. As a last part of Chapter 5, a framework consisting of recommendations and criteria for parties interested in building a collaborative OSS development organization is provided.

5.1 Comparative Analysis of the GENIVI Alliance and the Polarsys Working Group

In the subsequent chapter the presented collaborative OSS development organizations are compared. Based on important factors that accrue from the analysis of the GENIVI Alliance and the Polarsys Working Group performed in the preceding chapters, the common as well as the distinguishing characteristics are demonstrated (see Table 4).

<table>
<tr><th></th><th>GENIVI Alliance</th><th>Polarsys Working Group</th></tr>
<tr><td><b>Overall Aim</b></td><td>• Reduction of time to market
• Shortage of the development-cycle
• Reduction of overall costs
• Awareness-raising of open source</td><td>• Longevity and sustainability of software
• Reduction of overall costs
• Awareness-raising of open source</td></tr>
<tr><td></td><td colspan="2"><b>Organizational Perspective</b></td></tr>
<tr><td><b>Organizational Structure</b></td><td><i>One-level (see Figure 11)</i></td><td><i>Two-level (see Figure 11)</i></td></tr>
<tr><td><b>Application Process</b></td><td>One way: Signing of the participation agreement</td><td>Three different ways:
<table>
<tr><th>Member Classification</th><th>Agreements to sign</th></tr>
<tr><td>EF Solutions, Enterprise, Strategic Member</td><td>Polarsys Participation Agreement</td></tr>
<tr><td>EF Associate or Committer Member</td><td>1.) Membership Upgrade
2.) Polarsys Participation Agreement</td></tr>
<tr><td>Non-Member</td><td>1.) EF Membership Agreement
2.) Polarsys Participation Agreement</td></tr>
</table></td></tr>
<tr><td><b>Membership Classification</b></td><td>Three different member groups: The Founding Charter and Charter, the Core and the Associate Members</td><td>Four different member groups: The Steering Committee, the Participant, the Guest and the Committer Members</td></tr>
<tr><td></td><td colspan="2"><b>Legal Perspective</b></td></tr>
<tr><td><b>Open Source License</b></td><td>General Public License</td><td>Eclipse Public License</td></tr>
<tr><td></td><td colspan="2"><b>Financial Perspective</b></td></tr>
<tr><td><b>Revenues & Expenses</b></td><td>
<table>
<tr><th>Revenues</th><th>Expenses</th></tr>
<tr><td>Well-defined membership fees</td><td>Marketing activities</td></tr>
<tr><td></td><td>Contracted staff</td></tr>
</table></td><td>
<table>
<tr><th>Revenues</th><th>Expenses</th></tr>
<tr><td>Membership fees dependent on EF membership classification</td><td>Marketing activities</td></tr>
<tr><td></td><td>Contracted staff</td></tr>
</table></td></tr>
</table>

Table 4: Key Characteristics of the GENIVI Alliance and the Polarsys Working Group.

5.1.1 Overall Aim

The purpose and the incentives that led to the founding of the GENIVI Alliance and the Polarsys Working Group are similar. At the beginning of both organizations were companies that faced a problem the nature of their businesses brought along. In the case of the GA the problem was composed of increasing end customer needs based on rapidly changing technology of smartphones. In order to satisfy these customer needs the automotive companies were forced to speed up their development processes and reduce their time to market. Due to this, the idea to create a collaborative construct around automotive software development emerged (see Chapter 4.1.1; Crumb 2014). The founding of the Polarsys WG was based on similar reasons. However, in contrast to the GA, not the acceleration of the software development was the main focus but rather the longevity of the aerospace software. The initiating companies faced the challenge of how their software can be sustainably hosted and developed. As a result, a long-term strategy in the form of a collaborative software development organization has been determined (see Chapter 4.2.1). In economical terms a collaborative organization provides the opportunity to reduce overall development costs in sharing knowledge resources.

After achieving these primary goals, the GA as well as the Polarsys WG had strong incentives to promote and spread the idea of collaborative open source software development. In order to create a thriving and dynamic development ecosystem and to advance the vision of shared and faster development processes, as many other companies as possible needed to be convinced and acquired.

5.1.2 Organizational Perspective

There are three aspects to consider from the organizational perspective - the organizational structure and the allocation of responsibilities, the different steps of the application process and the member classification into different categories.

5.1.2.1 Organizational Structure

Compared to the GENIVI Alliance, the Polarsys Working Group has a rather small organizational structure (see Figure 11). This can be explained by the fact that the GA is an independent organization that needs to provide all its services by its own whereas Polarsys makes use of the defaulted services of the Eclipse Foundation (see Chapter 4.2.1). There is the provided and hosted IT infrastructure that in contrast is self-maintained by the GENIVI Alliance. Further, there is the ecosystem development service that organizes and takes care of various marketing activities and provides an opportunity to create awareness of the Working Groups. The GENIVI Alliance, however, accommodates a three-sided marketing department (see Chapter 4.1.2.1) that is responsible for representing the Alliance as widespread as possible. Furthermore, the Eclipse Foundation presets the basic governmental documents whereas the founding members of the GENIVI Alliance needed to develop these documents to a great extent from scratch. The Working Groups only have to create a group specific charter and a participation agreement. Due to the provided services, some departments of the GA automatically disappear in the Polarsys WG. The marketing team for example is similar to the ecosystem development service of the Eclipse Foundation and also the IT infrastructure team would be redundant within the Polarsys WG. In contrast to the GENIVI Alliance, however, the Eclipse Foundation has built up a long-term support infrastructure. Due to the long-term nature of the aerospace business the Polarsys Working Group is highly interested in this service.

Besides these differences in the governmental structure, there are also similarities. The highest ranked position of the Polarsys WG, the steering committee, resembles the board of directors of the GA in terms of its tasks and responsibilities. The slight difference is the supervisory authority. The GA has the operations subcommittee that controls and supports the board of directors. In contrast, the authority of the Polarsys WG is most likely represented by the EF itself. Technically speaking, the supervisory authority of the GENIVI Alliance is one level lower in the hierarchy but one level higher in the Polarsys WG. Further, both possess quality management entities.

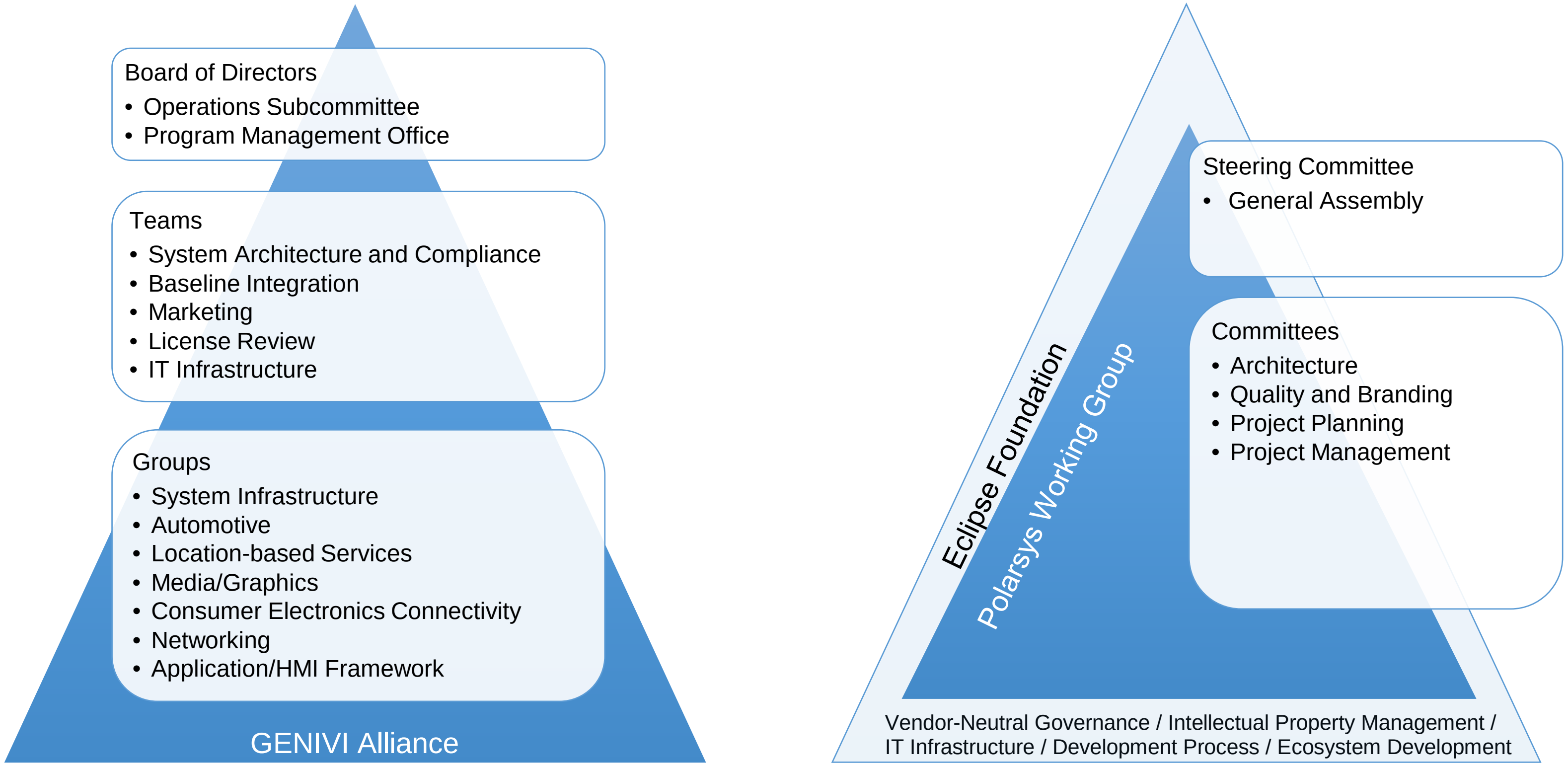


Figure 11: Comparison of the Organizational Structure.

One entity is responsible for the compliance of the technical requirements and one entity is responsible for the compliance of the licensing requirements. At the GENIVI Alliance those are the system architecture and compliance team in technical terms and the license review team in legal terms. At the Polarsys WG those are the architecture committee and the quality and branding committee respectively.

5.1.2.2 Application Process

There are three initial positions companies can hold when intending to join the Polarsys Working Group. First, they are already a member of the Eclipse Foundation on the level of a solutions member or higher. Second, they are already a member of the Eclipse Foundation but on a lower level than a solutions member. Or as a last option, they are not yet a member of the EF and need to enter the foundation first. The first is the ideal starting position in order to join a Working Group. Such members only have to complete the participation agreement. As shown before, option two and three are different starting positions. However, the process until admittance to the Working Group is the same. They first need to upgrade their membership status or apply for membership through the membership agreement. In a second step, it is required to complete the participation agreement of the particular Working Group. In comparison to this, the application process at the GENIVI Alliance is simpler and shorter. Companies interested in joining the Alliance solely need to fill in the participation agreement.

The membership agreement between the Eclipse Foundation and the Eclipse members and the participation agreement between the Working Group and the Working Group members is equal compared to the participation agreement between the GENIVI Alliance and its members. The agreements are similar and contain the same basic statements. For instance that the members agree with the intellectual property rights policy by signing the participation agreement.

5.1.2.3 Membership Classification

In order to get an overview of the different member groups, they were divided into three categories that are subsequently called gold, silver and bronze Member (see Table 5). The gold members mean the GENIVI's founding charter and charter members and the Polarsys' steering committee members. The silver members refer to the core (GA) and the participant members (Polarsys). The bronze members describe the associate members (GA) and the guests and committers (Polarsys). In order to ascertain the similarities and differences of the membership classes of the different organizations a summary of the main characteristics has been conducted (see Table 5).

In a first step the participation rights of the organizations' subgroups are analyzed. The subgroups, in the case of the GA are the teams and groups and in the case of Polarsys the committees. Within both organizations the gold members are automatically assigned to the highest authority. In the case of the GENIVI Alliance they are assigned to the board of directors (see Chapter 4.1.4.1) and in the case of Polarsys to the steering committee (see Chapter 4.2.4.1). However, while Polarsys' steering committee members are automatically members of all the subcommittees, the GENIVI founding charter and charter members need to be elected in order to lead or participate in teams and groups.

The silver members in both organizations must be elected in order to participate in a subgroup. Additionally, silver members of the GA in some cases need to meet some preconditions (see Table 5). Besides this, referring to all the other benefits and rights both the gold as well as the silver members get the same treatment (see Eclipse, Polarsys Working Group Charter 2012). Further, both organizations provide a broad range of marketing measures and activities for gold and silver members.

While the gold and silver members of the GA and the Polarsys WG do only differentiate to a minor degree, the bronze members differ fundamentally from each other. Compared to the other member categories, the associate members of the GA play a secondary role. There is no intention to say that they are not important, nevertheless, they often rather use the Alliance for marketing purposes than for contributing software (see Crumb 2014). This might be helpful to extend the whole ecosystem around the organization but it does not help advance the technology. As opposed to this, the committers of Polarsys play an essential role in the progress of the Working Group. Since they are the only group having write access to the open source code repositories, they are crucial to the success of the WG.

The committers can be compared to the project maintainers (see Foster 2014) within the GA, who are responsible for accepting and committing patches to the code of a particular open source project. The second category of bronze members within the Polarsys WG, the guests, at first sight does not seem as important as the committers. They neither support the WG financially nor have they relevant rights within the WG. Nonetheless, they take on an essential role; most of the guests are academic or research institutions that contribute new findings or methods. This might be highly conducive for the advancement of the Working Group.

Further, both organizations struggle with the free riding issue that can be considered from the negative as well as from the positive aspect. That more than half of the members do only participate in the collaborative OSS development organization for marketing or similar purposes and do not contribute ideas with innovative potential can be considered as the negative part of free riding. In contrast, each member that is part of the organization, regardless of whether active or passive, creates added value in terms of extending the network of the organization - the larger the network, the higher the probability that a new member that contributes essential components will join the organization (see Crumb 2014; Mueller 2014).

	Gold Members	**Silver Members**	**Bronze Members**
	GENIVI Alliance		
	Founding Charter and Charter Members	**Core Members**	**Associate Members**
Participation Rights	• Permanent board seat • Eligible for officer positions • Eligibility to participate in and lead teams and groups • Be part of several marketing measures & activities • Have various access rights	• Eligibility to participate in and lead teams and groups • Be part of several marketing measures & activities • Have various access rights • *Possible Preconditions:* Being an elected board member, an expert group lead or an architect	• Eligibility to participate in the marketing or license review team and groups • Eligibility to participate in expert groups • Be part of several marketing activities • Have some access rights
Membership Fee	*USD 150,000*	*USD 25,000*	*USD 5,000*
	Polarsys WG		
	Steering Committee Members	**Participant Members**	**Guest, Committer**
Main Benefits	• Definite member of each Working Group committee • Have all access rights • Be part of the branding process	• Possible elected member of Working Group committee • Have nearly all access rights • Be part of the branding process	• Guest: Possible elected member of Working Group committee • Committer: Write access to open source code repositories
Membership Fee	Polarsys: USD 20,000 - 30,000 EF Solutions Member: USD 5,000 - 20,000 *Total: USD 25,000 - 50,000* EF Enterprise Member: USD 125,000 *Total: USD 145,000 - 155,000* EF Strategic Member: USD 25,000 - 500'000 *Total: USD 25,000 - 530,000*	Polarsys: USD 1,500 - 10,000 EF Solutions Member: USD 1,500 - 20,000 *Total: USD 3,000 - 30,000* EF Enterprise Member: USD 125,000 *Total: USD 126,500 - 135,000* EF Strategic Member: USD 25,000 - 500'000 *Total: USD 3,000 - 510,000*	Polarsys: *USD 0*

Table 5: Overview of Membership Classifications.

5.1.3 Financial Perspective

The financial part of the GENIVI Alliance and the Polarsys Working Group can be split up into revenues and expenses (see Table 4). Both organizations generate their revenues through demanding a membership fee from all their members. The difference is that while the GENIVI Alliance has transparent and clearly defined amounts of membership fees, the fees of Polarsys strongly depend on the membership classification within the Eclipse Foundation. As described in Chapter 4.2.4 the membership fee of the Eclipse Working Groups is composed of the basic Eclipse Foundation membership fee plus the respective Working Group classification fee. Due to this two-level setup, members of an Eclipse Foundations WG actually pay twice. If registered as a solutions member at the Eclipse Foundation the membership fees are, compared to those of the GENIVI Alliance, significantly lower in terms of the gold member and rather similar in terms of the silver members (see Table 5). If an organization is active as an Eclipse enterprise member, the costs are the same compared to gold members of the GA, but tremendously higher compared to core members. The fees of the strategic members, which is the most common class of membership in the case of Polarsys, are not comparable to the GA membership fees. Since the fee is calculated on the basis of the annual revenue, the array of payable amounts ranges within a large number of possibilities. However, in comparison to the silver members of the GA, the membership fee of a strategic member in the participant category is most probably considerably higher (see Table 5).

The money is deployed in similar ways within both organizations. There are two main elements of expenditure. The first one is the funding of the various marketing activities that include conferences, marketing events or member meetings. The second main expenditure is labor. Despite the fact that the money in the end is deployed for more or less the same activities, there is a difference between the processes of payment. Like a traditional organization, the GA pursues a one-level process. This means that the GA earns and spends its money in order to fund its operational business. In contrast, the Polarsys WG pursues a two-tiered process. The Polarsys WG in a first step charges the membership fee and then passes it on to the Eclipse Foundation in a second step. The EF in turn uses these funds to ensure the provided services to the Working Groups.

One part of the revenues is spent on the organization of promotional events or conferences in order to increase the level of awareness and to extend the open source ecosystem in acquiring new members. Another part of the revenues are the salaries for non-member or additional employees. Although most of the employees at the GENIVI Alliance are members that are normally paid by their firms, there is also some externally contracted staff paid by the GENIVI Alliance (see Crumb 2014). Considering only the Working Group revenue, which is used for additional services, most of it is used for extra employees. Mueller uses the example of a Working Group that has an extraordinary high amount of code that needs to be checked by the Eclipse IP-management in order to be implemented. In this case the particular Working Group co-finance the additional labor cost for the IP department (see Mueller 2014).

5.1.4 Legal Perspective

Considering the intellectual property rights, both organizations pursue the same basic approach following the OSS principles (see Chapter 3.1 - 3.3). In other words the GENIVI Alliance as well as the Polarsys Working Group require the contributions' disclosure of their members through licensing their work. However, since the GENIVI Alliance is based on the General Public License (GPL) (see Chapter 3.3) and the Polarsys WG on the Eclipse Public License (EPL) (see Chapter 4.2.3), they differ in the severity of the copyleft principle (see Chapter 3.3). While the Eclipse Public License pursues a more permissive approach, the GPL makes no exception. While the EPL allows contributors to license add-ons in the form of separate module or not understood as derivative work under a different (open source or proprietary) than the Eclipse Public License, the GPL intends to transfer any copyleft right through licensing every add-on under the GPL (see Chapter 3.3).

5.2 Types of Collaborative Open Source Software Development Organizations

By reference to the comparative analysis in the preceding chapter, two distinct types of collaborative open source software development organizations can be determined. On the one hand is the GENIVI Alliance that operates as an independent, autonomous organization and on the other hand is the Polarsys Working Group that operates under the umbrella of the Eclipse Foundation as a partner of sorts. The type of the GENIVI Alliance will further be entitled as the *Autonomous Collaborative Open Source Software Development Organization* and the type of the Polarsys Working Group will be entitled as the *Affiliated Collaborative Open Source Software Development Organization*. In the subsequent subchapters the key characteristics as well as the benefits and pitfalls of both types of collaborative open source software development organization will be described in more detail (see Table 6).

Key Characteristics	
Autonomous Collaborative Open Source Software Development Organization	**Affiliated** Collaborative Open Source Software Development Organization
1. Starting from Scratch Development of the entire basic organizational, legal and financial structure.	*1. Two-tiered set up* Processes need to pass through the requirements of the parent company as well as the affiliated collaborative OSS development organization.
2. Build up of Reputation The awareness and reputation is non-existent and needs to be created.	*2. Out of the Box Principle* Basic services are provided by the parent company.
3. Coopetition Cooperation of competitors within one organization.	*3. (Inter)dependency* The parent company and the affiliated collaborative OSS development organization are interconnected and dependent on each other.

Table 6: Key Characteristics of an Autonomous and an Affiliated Collaborative Open Source Software Development Organization.

5.2.1 Autonomous Collaborative Open Source Software Development Organization

In this chapter the key characteristics of an autonomous collaborative open source software development organization and its advantages and disadvantages will be highlighted from the different perspectives of the involved parties.

1. Starting from Scratch

The crucial point of an autonomous collaborative open source software development organization is the completely new set up. The building of an autonomous collaborative OSS development organization is similar to a business formation. In a first step, the company that initiates the founding of a new organization needs to find like-minded people in companies, industries and organizations suitable for the intended purpose. If the search for partners interested in allying into an organization was successful, they either have to build up a new organization on their own or seek an expert team that supports them with its knowledge and experience (see Crumb 2014). After that, the actual founding and the setup of the organization follow. Due to the fact that there is more than one party plus a possible expert group involved in the founding of the organization, there is an enormous effort in collective terms. From the collaborative point of view, the major challenge is to find a common ground. First of all, it is essential to agree on the overall vision, goals and the purpose the new organization shall pursue. Further, they need to reach an agreement in organizational, financial as well as in legal terms. The involved parties have to define a governmental structure, must determine the way the organization will be financed and need a draft of the basic legal documents such as the bylaws, the intellectual property policy or a membership agreement. The entire setting-up of a collaborative organization is subject to many discussions, compromises and concessions that are highly time-consuming (see Mueller 2014). Moreover, significant human resources are required since various groups of each founding member, are intra- and interorganizationally involved in building a collaborative OSS development organization. The legal department for instance, is responsible for a preliminary version of the legal documents and the subsequent negotiations with the legal departments from the other partner companies (see Mueller 2014). Furthermore, most employees of the organization are under contract by members (see Crumb 2014). In other words, each member has to provide additional human resources if participating in an organization. All in all, building an organization from scratch is enormously cost-intensive and time-consuming and needs a high amount of expertise in several business domains. Therefore, an expert group might support the founding members with best practices or act as a mediator and steward who arbitrates between the different parties during the negotiations and the strategic planning. This support might mitigate the required effort of the involved parties.

In sum, building a new collaborative OSS development organization from scratch can be seen as both an advantage and a disadvantage - an advantage in the sense of being in charge and a disadvantage considering the significant investment in collaborative and financial terms.

2. Build up of Reputation

As already mentioned in the preceding paragraph, the autonomous collaborative OSS development organization is not bound to any specifications. Founding members are free to decide on their scope, their governmental structure, their technical basics (e.g. operating systems) and their target groups in terms of which businesses and industries they focus on. To be autonomous can implicate advantages as well as disadvantages. However, the previous paragraph deals with tangible resources whereas the reputation belongs to the intangible, ideational resources. Having no parent company does mean not to have any reputational foundation from the beginning. On the one hand this might be a benefit. Assuming the parent company gets into trouble or creates a scandal, all the affiliated collaborative OSS development organizations will be automatically involved although they have no responsibility for what has happened. On the other hand it might be a disadvantage. The organization first needs to establish itself and become known within its target industry. The entire reputation and the brand awareness must be built up in a long-term process.

3. Coopetition

Considering the fact that several different actors are involved in the successful continuation of the organization, the collaborative aspect and divergent opinions will always pose a challenge. However, due to the creation of all the documents in the beginning (see Paragraph 1.), the collaborative effort decreases to a certain degree as soon as the organization establishes itself. In this context the Executive Director Steve Crumb (2014) of the GENIVI Alliance uses the term *Coopetition*. The word Coopetition means that the participating members are cooperating as competitors. *“What we encourage the board members to think about is, that, when they are sitting at the board table at a GENIVI board meeting, they are there for the sake of the organization. Yes, they represent their member company but they are there to ensure that the organization is successful and accomplishes its mission and so forth”,* says Steve Crumb (2014, p. 91). This statement illustrates that there are different opinions, however, as long as the different parties come to a compromise regarding possible controversial subjects and concentrate on the essential points that led to this cooperation, the organization should not suffer but rather benefit from clashing of opinions.

To sum up, the positive aspects outweigh the negative aspects of Coopetition. The involved people have different educational backgrounds, careers in distinct business domains and as a result of this, diverging knowledge and experience. The fact that each of them holds a slightly differing view leads rather to a constructive discussion and this in turn to more innovation (see Crumb 2014).

It might be true that Coopetition as well occurs within the affiliated collaborative OSS development organization. However, it seems more fundamental to the autonomous collaborative OSS development organization. Since they are on their own rather than supported by a parent company, they need to be able to cope with possible divergences.

In summary, the autonomous collaborative open source software development organization is a completely new organization based on self-developed documents, regulations and guidelines that has to set up an environment where companies across various industries are able to work together as a unity and consequential build up a widespread awareness and common reputation (see Figure 12).

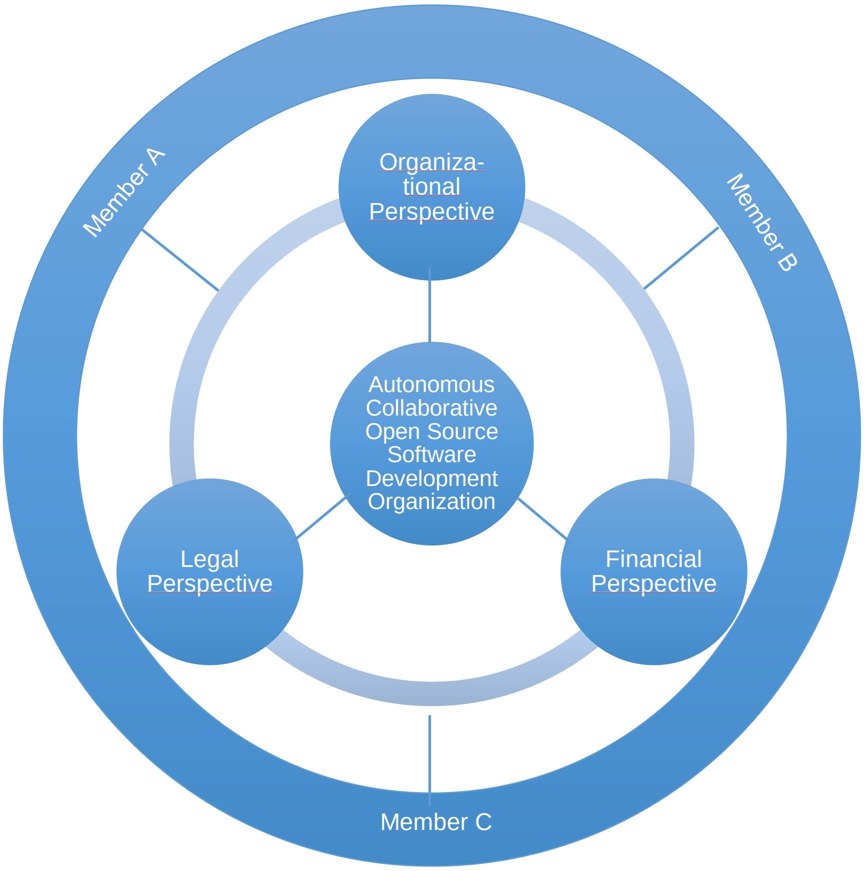

Member A
Member B
Member C
Organiza-tional Perspective
Autonomous Collaborative Open Source Software Development Organization
Legal Perspective
Financial Perspective

Figure 12: Autonomous Collaborative Open Source Software Development Organization Model.

5.2.2 Affiliated Collaborative Open Source Software Development Organization

In this chapter the key characteristics of an affiliated collaborative open source software development organization and its advantages and disadvantages will be highlighted from the different perspectives of the involved parties. In order to set these findings into a more general context, the Polarsys Working Group will further be replaced by *affiliated collaborative* *open source software development organization* and *parent company* will replace the Eclipse Foundation*.*

1. Two-tiered set up

What attracts attention and appears in each of the compared section is the two-tiered setup of the affiliated collaborative open source software development organization - be it the two-leveled stream of cash, the application process with its twofold membership or the split organizational structure.

From the perspective of a potential applicant the process of becoming a member of an affiliated collaborative OSS development organization is rather complicated. Since a potential applicant is not allowed to register for the affiliated collaborative OSS development organization until it signed up with the parent company, the hurdle to join might appear significant and may have a deterrent effect. It could be considered to create an option where companies get the possibility to register only for the specific purpose they want to participate. However, since the affiliated collaborative OSS development organization is highly dependent on the parent corporation from several perspectives (see Paragraph 3.), this is a hardly realizable proposition.

Affiliated collaborative OSS development organizations benefit from the two-tiered membership application process in two ways. First, they can assume that the applying party has already been accurately screened during the application process at the parent company. This reduces their own effort to recheck the suitability of the applicants regarding the affiliated collaborative OSS development organization’s purpose. Further, a company that is willing to pass through more than one application process shows its interest and it can be assumed that they are highly motivated in encouraging the affiliated collaborative OSS development organization.

The parent company benefits from the preceding registration from affiliated collaborative OSS development organization members in terms of automatically extending their membership base. The more members, the wider the network and this in turn leads to a broader awareness within the open source collective (see Mueller 2014). The disadvantage of a broad membership base might be the growing number of free riders (see Chapter 4.1.2.2).

Another aspect of the two-tiered set up is the divided money transfer. In a first step, the members need to deposit the membership fee according to their membership classification at the affiliated collaborative OSS development organization. In a second step, the working group pays the parent company to a certain extent depending on their obtained services.

All in all, comparing the two-tiered set up of the affiliated collaborative OSS development organization with the one-level governance structure of the autonomous collaborative OSS development organization, it appears that the two-tiered structure brings along a higher degree of consultation. Due to the divided allocation of responsibilities to the parent company and the affiliated collaborative OSS development organization, there is often a need of conferring with the parent company in order to carry out particular actions and processes.

2. Out of the Box Principle

A further key characteristic of an affiliated collaborative OSS development organization is the *Out of the Box Principle* (see Mueller 2014). The *Out of the Box Principle* in this case describes the provided services by the parent company that are modular and can be pieced together according to a particular purpose. The affiliated collaborative OSS development organizations are asked to make use of the provided services in order to avoid building a completely new organization from scratch. Considering the costs and the time-consuming effort the building of a collaborative OSS development organization brings along, this is an advantage from the perspective of companies that are interested in entering into the open source collaboration business. Despite the beneficial aspect of the prefabricated services, interested companies need to accept the given rules. They might adapt but not change the basic standards.

From the perspective of the parent company the *Out of the Box Principle* may be advantageous as well as disadvantageous. On the one hand, through providing several well-developed and sophisticated services, they signalize many years of experience what might appear attractive towards potential applicants. In addition, if a company decides on entering the open source business but is not willing or does not have the ability to spend a lot of resources, they rather join an existing organization than initiate a new one. This attitude of choosing the path of least resistance may lead to a competitive advantage towards the autonomous collaborative OSS development organization. However, on the other hand, the parent company puts itself under pressure since the expectations and requirements in relation to the provided services are high and members proceed on the assumption that no complications occur.

All in all, the *Out of the Box Principle* is sort of a resources recycling process that seems to be a positive aspect of the affiliated collaborative OSS development organization.

3. (Inter)dependency

At a first glance, it appears that the dependency is a clear drawback for the affiliated collaborative OSS development organization. The affiliated collaborative OSS development organization is reliant upon the support of the parent company - be it the technical, legal or marketing support in terms of providing the IT infrastructure, the legal documents or the branding activities. Without the integration of the collected knowledge and experience, the affiliated collaborative OSS development organization could not exist. Despite the fact that the affiliated collaborative OSS development organization cannot survive without the parent company but vice versa, there is a great interest from the parent company in accommodating affiliated collaborative OSS development organization. The VLTS (Very Long Term Support) sets a good example for this interplay. While some of the affiliated collaborative OSS development organizations rely heavily on an IT infrastructure that has a durability of several decades and have therefore a high willingness to pay, the parent company can benefit from obtaining funds for a new service which in turn leads to the acquisition of new industries interested in long term support. Consequently, the parent company again benefits from the needs of the affiliated collaborative OSS development organizations through gaining new members.

In sum, the described aspect is rather a reciprocal than a dependent relationship. Although the degree of dependency of the affiliated collaborative OSS development organization on the parent company is much higher than the dependence of the parent company on the affiliated collaborative OSS development organization, the parent company would lose an important part of its network and make a step backwards regarding possible innovative technologies.

In summary, the affiliated collaborative OSS development organization is an organization that benefits from provided services of a parent company and does not need to build up the basic structure by itself. This determines a high degree of interdependence between the different parties and a two-tiered set up where several processes need to pass through. It is an interplay between the overarching parent company, the affiliated collaborative OSS development organization and the members. Those are only parent company members, parent company and affiliated collaborative OSS development organization members or parent company members and members of different affiliated collaborative OSS development organizations (see Figure 13).

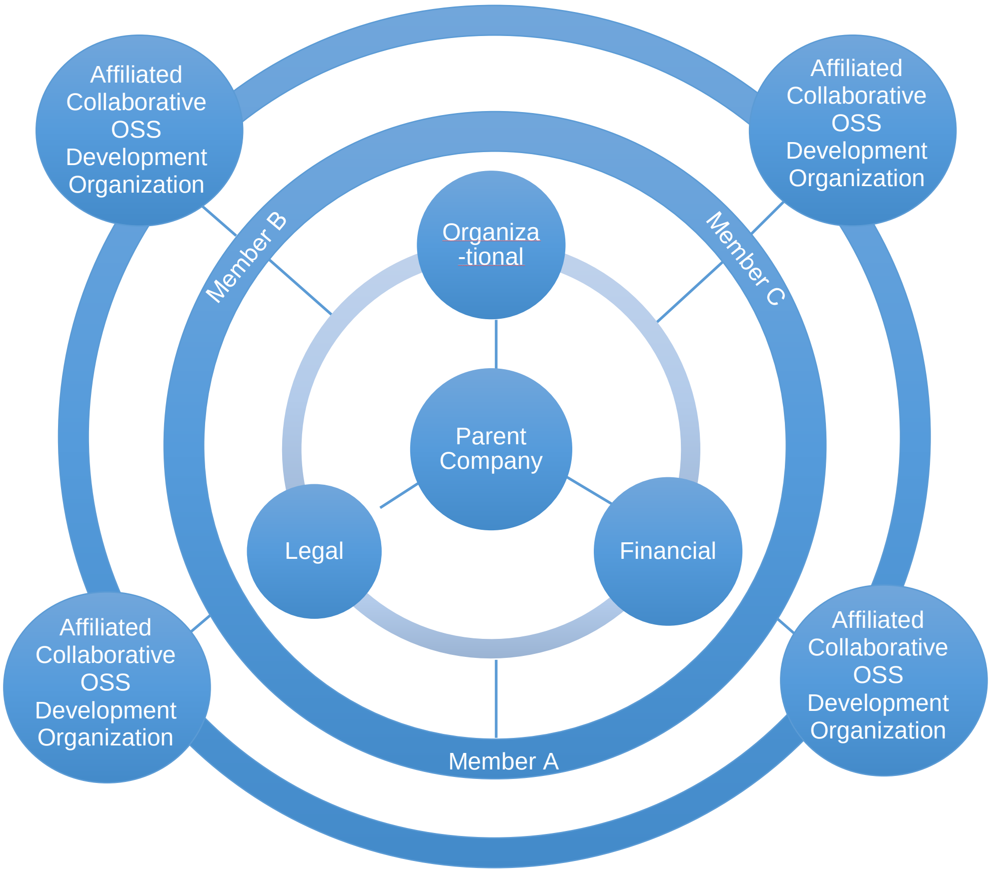

Affiliated Collaborative OSS Development Organization
Affiliated Collaborative OSS Development Organization
Member B
Member C
Organiza -tional
Parent Company
Legal
Financial
Affiliated Collaborative OSS Development Organization
Affiliated Collaborative OSS Development Organization
Member A

Figure 13: Affiliated Collaborative Open Source Software Development Organization Model.

In conclusion, the evaluation of the benefits, drawbacks and key characteristics of the two forms of collaborative OSS development organizations reveals that there is no better or worse way. Both forms carry risks as well as provide opportunities. Which form would be the most suitable for a company, is dependent on several factors described in this thesis. In order to examine these factors an evaluation of the particular company and how and to which degree the various factors are distinct is needed.

In due consideration of these findings it seems legitimate to filter out the fundamental common characteristics of both. By doing this, the companies are given the basic framework that reveals if they bring along all the required preconditions for entering the collaborative OSS development sector.

5.3 Framework for Building a Collaborative Software Development Organization

Based on the findings resulting from the evaluation and the comparative analysis in the preceding chapters and the answers of the expert interviews, a set of criteria that have to be considered by companies in order to build a collaborative OSS development organization will be developed (see Table 7).

Criteria	**Description**
1. Explicit Scope	A common interest that is consequently pursued by all members.
2. Long-Term and Sustainability Approach	The overall goal is to build a sustainable open source environment that persists for a long period of time.
3. Non-Differentiating Approach	There is no competition regarding the developed products.
4. Openness	Communication and information flows must be transparent.
5. Comfort Zone: Open Source	Members have to move on from the traditional proprietary software development approach and make the idea of open source their new guiding principle.
6. Crucial Roles	The executives, a mediating role, a community manager, a legal entity and a quality management entity are the most important roles.
7. Expert Council	A group of experienced people helps to find individual solutions for particular interests and needs and gives advice in precarious issues.
8. Continuous Acquisition of New Members	A high amount of members leads to a broad network and this in turn to a broader awareness of open source and the particular organization.

Table 7: Framework for Building a Collaborative Open Source Software Development Organization.

1. Explicit Scope

The first step to successfully create a collaborative OSS development organization is a common interest and as a consequence thereof, a vision. It is necessary to have a guiding principle that forms the base and determines the direction the organization will pursue in future. To create a more precise impression, specific goals and how to achieve them must be defined. Without a clearly determined focus, it will be challenging to hold together the different organization members. Considering the fact that they are still competitors, it is conducive to create transparency in communicating precise goals and their purposes. It is vital to create a feeling of solidarity and cohesiveness that serves as a motivational factor in order to create a strong organization and a thriving open source environment (see Crumb 2014; Mueller 2014; Wiedemann 2014).

2. Long-Term and Sustainability Approach

In order to build a sustainable environment, the purpose of a collaborative OSS development organization is to pursue long-term rather than short-term objectives. By means of collaborative OSS development organizations, software shall be made available and continuously refined and used over several decades. Considering this future-oriented approach, it is not recommendable to build a collaborative OSS development organization when a short-term software solution is needed. This neither helps the organization nor satisfies the demand. The involved parties' comprehension of this idea is fundamental in order to grow a functioning and stable open source ecosystem (see Mueller 2014).

3. Non-differentiating Approach

A further crucial characteristic of a collaborative OSS development organization is the non-differentiating approach (see Chapter 4.1.1). Since the involved parties are competitors and their relationship is basically dominated by rivalry, it is crucial that they work together in relation to a non-differentiating good (see Crumb 2014). The idea behind a collaborative OSS development organization is not to create an individual but a common competitive advantage. In other words, the organization as a whole can benefit towards non-members. An individual competitive advantage might arise from the products individually built upon the non-differentiating goods by the members. The Coopetition (see Chapter 5.2.1, 3.) goes together with the non-differentiating approach. Keeping in mind the fact that the participants work together in order to develop commodity software, they are able to collaborate in a neutral way although they are actually competitors (see Wiedemann 2014).

4. Openness

While the disclosed source code is the foundation of each and every collaborative OSS development organization, other factors in terms of openness have a significant influence on the successful growth of the organization. It is highly important that apart from the software the information base is openly accessible as well. In concrete terms, there is a need for open communication and open collaboration (see Wiedemann 2014). Information flows and processes need to be unveiled in order to prevent any unfairness issues or misunderstandings. While openness is important in any traditional enterprise, it plays an even more vital role in context with collaborative OSS development organization. Since the participants stay competitors and do only collaborate on the special issue dealt with within the organization, it is necessary to avoid any further platform with potential for conflicts.

5. Comfort Zone: Open Source

Many companies still struggle with the idea of giving away and making publicly available their in-house innovation and with that a possible competitive advantage and revenue (see Crumb 2014; Foster 2014). Therefore, it is essential to choose partners that are either already operating in the open source sector or are highly motivated in learning more about this new way of collaboratively developing software. Companies that are difficult to convince and cannot entirely identify with the basic idea of open source, only inhibit the growth and thriving of a collaborative OSS development organization. Steve Crumb (2014) says *“there has been a lot of education, a lot of transition and a lot of paradigm shift that has been necessary”* (see p. 93). According to this statement, the goal in acquiring new members is to make them comfortable in using open source. In order to achieve this goal there is a need of rethinking. It is not just a way of changing particular development tools, it is a basic attitude and philosophy. Therefore it is important to start a collaborative OSS development organization as a fully open source organization rather than as a proprietary organization that needs to be turned into an open source organization. The processes of transition and the remodeling of settled habits may force companies to face unpleasant challenges (see Crumb 2014; Foster 2014).

6. Crucial Roles

In order to make the organization as efficient and effective as possible, it is essential to fill positions with the most suitable people. Suitable people in this case can be defined as people that have been dealing with open source for a long time and have the knowledge and experience on various levels. On the social level for instance, they need to know how the open source collective thinks and how people within this business sector behave. Depending on the position, it is more or less important to have technical skills and knowledge.

What was accrued from the interviews is that the crucial roles within a collaborative OSS development organization are a mediating role, experienced executives, a community manager, a legal entity and a quality management entity. The setup and the governance of a collaborative OSS development organization is an interplay between managerial and technical skills. Because a collaborative OSS development organization has the characteristics of a traditional enterprise, it is crucial to have experienced executives. To implement the ideas and needs of the managers, capable software developers are an essential part of a collaborative OSS development organization as well. In managerial terms the roles of the executives and the mediating role are the most important parts of the organization. The executives have multiple tasks and support the organization in strategic, control and planning questions. The mediating role acts as a neutral entity between the various stakeholders and intervenes or gives advice if the members do not achieve a consensus (see Crumb 2014). The community manager operates in both functions, as a software developer and as a manager, and can be understood as the interface between the two sides. In technical terms the community manager is working together with the software developers and provides support and assistance for OSS projects. In managerial terms the community manager is collaborating with the executives (see Foster 2014).

A third crucial part of a collaborative OSS development organization is the legal department. To ensure that the organization acts within the legal boundaries, an entity with the required professional knowledge concerning the basic legal documents such as the bylaws or the intellectual property rights policy is needed. Additionally, an entity that ensures the quality and the compliance of submitted projects must be build up.

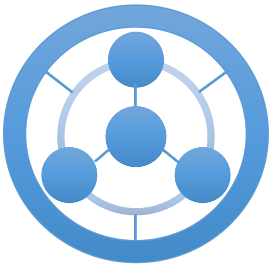
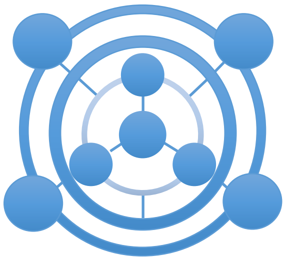
7. Expert Council

Considering the fact that this type of collaboration is a rather new phenomenon, it is pivotal to join forces with experienced people. Whether those are experts that are already part of an existing collaborative OSS development organization or an outside expert group specializing in establishing collaborative OSS development organizations, it is necessary to consult qualified people. They are familiar with possible pitfalls and might draw on case studies, best practices and lessons learned of previous customers. Success stories of other prosperous organizations might act as a motivating incentive (see Crumb 2014).

8. Continuous Acquisition of new Members

In order to grow and expand, it is vital to constantly acquire new members. New organization members go together with further know-how, other experiences and other perceptions and attitudes towards various aspects. The larger the mixture of people from different business domains, the higher the innovation potential. For this reason, a successful collaborative OSS development organization should never stop growing. Even though the probability of attracting free riders is high, they also contribute to more awareness of the organization. It is therefore not advisable to take measures in order to prevent possible free riders from joining, but rather benefit from the large network (see Mueller 2014).

Table 8 summarizes the key findings and key results of the preceding chapter.

Autonomous Collaborative OSS Development Organization	**Affiliated Collaborative OSS Development Organization**
	
Key Characteristics	
Autonomous Collaborative Open Source Software Development Organization	**Affiliated** Collaborative Open Source Software Development Organization
1. Starting from Scratch Development of the entire basic organizational, legal and financial structure.	*1. Two-tiered set up* Processes need to pass through the requirements of the parent company as well as the affiliated collaborative OSS development organization.
2. Build up of Reputation The awareness and reputation is non-existent and needs to be created.	*2. Out of the Box Principle* Basic services are provided by the parent company.
3. Coopetition Cooperation of competitors within one organization.	*3. (Inter)dependency* The parent company and the affiliated collaborative OSS development organization are interconnected and dependent on each other.

Criteria for Building a Collaborative OSS Development Organization	
1. Explicit Scope	A common interest that is consequently pursued by all members.
2. Long-Term and Sustainability Approach	The overall goal is to build a sustainable open source environment that persists for a long period of time.
3. Non-Differentiating Approach	There is no competition regarding the developed products.
4. Openness	Communication and information flows must be transparent.
5. Comfort Zone: Open Source	Members have to move on from the traditional proprietary software development approach and make the idea of open source their new guiding principle.
6. Crucial Roles	The executives, a mediating role, a community manager, a legal entity and a quality management entity are the most important roles.
7. Expert Council	A group of experienced people helps to find individual solutions for particular interests and needs and gives advice in precarious issues.
8. Continuous Acquisition of New Members	A high amount of members leads to a broad network and this in turn to a broader awareness of open source and the particular organization.

Table 8: Overview of Key Findings.

6 Conclusion

Taken all together, the four interviewed experts support the paradigm shift from in-house developed to collaboratively developed products ascertained earlier by Enkel et al. (2009) and Baldwin and von Hippel (2011) (see Chapter 3.4). They confirm the changing awareness of open source and the consequential changing way of cooperation across competing firms. Nevertheless, it needs to be considered that there are still many market players that up until now did not experience this change of mind, but rather pursue the traditional approach. It seems, that the process of remodeling the way of developing software is a large step - no matter in which industry. However, the ensuing step of merging and jointly developing and refining software appears to be the next logical stage and therefore easier. All in all, it can be said that this paradigm shift takes place and is establishing step-by-step, but needs high efforts at persuasion. Once companies decide to become part of the open source sector they have two options. Either they join an already existing OSS association as an affiliated collaborative OSS development organization or they establish their own autonomous collaborative OSS development organization. As shown in the preceding chapters, both forms have their advantages and their disadvantages. Therefore, it is crucial for companies intending to take part in the open source sector to evaluate their capabilities regarding the organizational, financial and legal perspective. In so doing, they are well-prepared to choose a solution appropriate to their abilities and needs.

6.1 Future of Collaborative Open Source Software Development

As mentioned above, the phenomenon of collaborative open source software development among firms is spreading continuously. For this reason, industries with high open source potential will be discussed in this chapter.

The *Internet of Things*[8] *(IoT) (*Rouse 2014*)* is an upcoming business segment that seems to be an interesting issue in relation to collaborative OSS development organizations (see Foster 2014; Mueller 2014; Wiedemann 2014). On the one hand IT experts attribute high potential to the new technology and predict a promising future (Gartner Hype Cycle Special Report 2014) and on the other hand most of the projects developing IoT technologies and standards have been initiated by the open source sector (Harvey 2014), such as the Internet of Things Working Group hosted by the Eclipse Foundation.

High growth potential in relation to collaborative open source software development is also expected in the research field, especially with large research projects, such as Horizon 2020[9] (What is Horizon 2020?). Providing open source services in order to sustainably make the research results available in the form of source code might be a highly attractive option for open source organizations (see Mueller 2014).

In terms of the research field not only the provision of hosting services for research projects but also the cooperation with research entities should be considered. Based on the model of the Eclipse Foundation, future open source organizations are well-advised involving research entities to a greater extent. Both, the entrepreneurial as well as the research entities would benefit in terms of knowledge exchange and varying approaches and methods (see Mueller 2014).

Furthermore, according to the results of the Future of Open Source Survey 2014 conducted by Black Duck Software (Black Duck Software 2014) the government, the education and the health medical industry are most impacted by open source over the next two to three years.

The United Nations (UN) for instance published the report *A World That Counts* authored by the Secretary-General's Independent Expert Advisory Group on a Data Revolution for Sustainable Development (IEAG) in

[8] The Internet of Things (IoT) is a scenario in which objects, animals or people are provided with unique identifiers and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction. IoT has evolved from the convergence of wireless technologies, micro-electromechanical systems (MEMS) and the Internet.

[9] The biggest EU Research and Innovation program ever with nearly €80 billion of funding available over 7 years (2014 to 2020).

November 2014 (Data Revolution Group 2014). The aim of this report was to propose ways to improve data for achieving and monitoring sustainable development according to the Sustainable Development Goals (SDGs) concluded at the Rio+20 United Nations Conference on Sustainable Development in 2012 (United Nations Department of Economic and Social Affairs 2012). However, as Mark Charmer (2015) criticizes in his article, this endeavor of gaining control over the mass of data faced by the entire world does not incorporate the open source movement. In his article he highlights the benefits an implementation of these goals with open source might bring along. Due to the fact, that not only companies but the entire world is involved in this project makes it highly interesting for the collaborative open source software development approach.

Further, the educational sector seems to be a promising supporter of open source. As examples the open source learning platform Moodle (Moodle 2014) that is used by educational institutions all over the world or the Swiss organization Parldigi (Parliamentary Group of Digital Sustainability) that fosters the application of open source in Swiss schools can be mentioned (Stürmer 2014). The increasing use of open source software applications in educational institutions might lead to an increased exchange of knowledge and experience among them and this in turn to the development of open source software in a collaborative way. Additionally, the young generation’s awareness of open source and the understanding that open source is not a niche product but rather an alternative to proprietary software can be raised. This might facilitate the entering of the students into the collaborative open source software development sector once they are in the world of work.

Regarding the health medical industry Bakar, Sheik and Sultan already stated in 2012 in their paper about health care information systems in developing countries that there is *“a shift from proprietary software to OSS”* (p. 443). Thus the health medical sector seems to be on course to use open source in a serious way, which is the ideal precondition for entering the collaborative OSS development organization sector. For example, the Open MRS platform which is currently rather an open source community than an open source organization meets the requirements to slowly transferring into a collaborative open source software development organization.

6.2 Limitations and Suggestions for Further Research

A limitation of this thesis is the small amount of discussed objects of investigation. The findings are solely based on two different organizations. Due to this, they are rather limited and might not be conclusive. In order to provide a broader information base, it is therefore recommended to analyze more collaborative OSS development organizations in further studies. By doing this, crucial points that lead to a successful and sustainable organization can be extracted more precisely and a higher amount of various types of organizations can be defined. Considering these findings, it might also be possible to provide more industry-specific information, such as which type of organization makes sense for which industry.

This thesis deliberately neglected the technical perspective of collaborative OSS development organizations. The more detailed examination of the technical basics and what kinds of possibilities exist in technical terms in order to run a collaborative OSS development organization might be an interesting research approach for future studies.

It would further be interesting to examine the incorporation of research entities into collaborative OSS development organizations. An interesting subject for a survey might be to analyze the type of collaboration, meaning how the more theoretical oriented academic side works together with the more practical business side. The actual outputs growing out of this cooperation or the incentives to work collaboratively with the respective other side might be further objects of future research.

Moreover, future research could focus on the merger of various existing collaborative OSS development organizations with similar interests to one overarching organization. Organizations with similar or complementary purposes and visions could be extracted and a common strategy of reuse and sustainable hosting developed. Along with this idea, the decoupling from a parent company and founding of an autonomous collaborative OSS development organizations or vice versa, the transfer from an autonomous to an affiliated collaborative OSS development organization and the consequential effects could be simulated.

In view of the discussed factors another point that would be worth questioning is the credibility of these collaborative OSS development organizations. It might be ask if they still represent the basic idea of open source in terms of software as a public good that is freely accessible to anyone, or if the current movement leads into a commercial direction where no longer the contribution but rather the monetary aspect is valued (see Foster 2014). It appears to be a trade-off between governance and determining rules in order to control and steer the organizations and the ideology of developing software in an open and accessible way.

In order to create a thriving worldwide open source environment, the author hopes that this thesis encourages companies across various industries to build collaborative open source software development organizations following the basic open source principles.

Appendix

In order to ensure complete transparency regarding the conducted interviews, the conversations with the four experts are in the following listed to their full extent.

Interview Steve Crumb (SC), Executive Director GENIVI Alliance, on October 23rd, 2014

RW: What is your educational background and how did you get in contact with open source and the GENIVI Alliance?

SC: I have a Bachelor of Science in Computer Science a long time ago. I have since been in the technology field in a number of different activities and jobs ranging from networking in the early days of the Internet, super computing applications and in the last about 11 years I have been in not-for-profit alliance management. The company that I actually work for is an alliance management organization. We manage about twenty different technology alliances of various types - GENIVI being one of those. How I came into contact with GENIVI was back in 2008. Intel, BMW and Windriver came to my organization and said: Hey, we have this idea to build a collaboration around an automotive software and we need help creating the organization and managing the organization once it's created. I was involved in helping get the organization to the point where it was launched from a legal point of view and grew the membership and I have been the Executive Director since.

RW: How did the GENIVI Alliance come into existence? What were the reasons of the founding?

SC: There are three primarily reasons. I don't know if you are familiar with the In-Vehicle Infotainment Systems that they're putting in their cars right now?

RW: No, not really.

SC: Let me give you a one-minute overview. In cars that are being delivered these days you don't just have a radio or a CD or DVD Player. You can have navigation and information systems that tell you what's happening with the car, review cameras. The amount of information that is available to a driver now, not just in the entertainment-side of things but also in the car-information side of things is just growing and growing. Basically, these infotainment systems are becoming personal computers in the car with all of the potential activity. Everything you can do on your smartphone, you can do in your car these days. Obviously what was happening to deliver all that functionality is that the amount of software that is going into these systems was growing and growing and it was getting more expensive to deliver them. But the cost to the driver, to the end-consumer, was not growing at that same level of speed. They didn't want to pay more than couple of hundred dollars, 400 dollars for the system. The automakers were facing the situation where they were being required to build a lot of software and to deliver a lot of functionality but not at significantly more cost for the car. They had to figure out a way to handle this software cheaper, faster and that sort of thing. The other problem was that everybody's got a smartphone these days and they want to be able to plug that into their car and have their own playlist, their own SMS and all the things that they have on their smartphones in the car. If you think about smartphones, they turn around every twelve to eighteen months. You get a new phone, but you don't get a new car every twelve to eighteen months, you get a new car once every five years or so. There had to be a delivery-cycle-reduction and the typical car program would take three to five years to complete. Whereas with these phones you'd have a new one in twelve to eighteen months and there had to be a reduction in the development period for the building of the software to make it aligned better with these consumer electronics devices delivery.

RW: Considering the membership benefits, GENIVI classifies the degree of participation and the corresponding annual fee into three categories: the Founding Charter & Charter, the Core and the Associate Members. What are the causes for this distinction and with reference to which criteria were these different degrees of participation determined?

SC: GENIVI membership is self-selecting in the sense that if you're an automotive organization, then you have a very high interest in what GENIVI is producing and you have budget what most OEMs, carmakers and large software suppliers have. Then you would probably be interested in joining at a more influential level which the higher tiers of membership. As you go up in tiers you get more influence, more ability to determine the strategy, the direction and the financial models - those sorts of things. At a very simplistic level the higher tiers have more influence, the lower tiers are more about visibility. A lot of organizations join at the associate level, simply because their new in the automotive market or maybe they've been around in the automotive market and they want to build out their business network in the system. There is kind of three different words that I use for the different levels. The founding charter & charter is all about influence, the core level is all about leadership in a sense of technical leadership and leadership in the delivery of GENIVIs output and then the associate level tends to be more about visibility and participation. So you get in there, you are visible in the network and that is important to you.

RW: What about the other automotive manufacturers? Brands like Mercedes Benz, Opel or Ford are not members of GENIVI. Is there an intentional selection of participants to explicitly differentiate from these non-members and to hold a competitive advantage towards them or is there a long-term strategy which aims at bring them all together as a big corporation?

SC: No, there is no selection. The organizations determine whether they want to be a member of GENIVI or not. In Mercedes case and in other cases they may not be a clear alignment of their organizational goals with the goals that GENIVI is attempting to meet - our mission of delivering an open source platform. The other thing is that a lot of organizations, Ford e.g., hasn't selected the LINUX operating system as the basis for their IVI-Systems and so they have less interest in participate, since we're a LINUX based software system.

RW: How do you assess the relevance of GENIVI within the automotive market? Are cost savings the crucial point or is it more like a marketing platform?

SC: I'll start with the associates and the marketing platform. It's true that GENIVI has built over the years a great business network. And organizations, quite frankly, join GENIVI just to be part of that business network. And that's fine with us. We would love for them to also participate in the technical delivery of things but if they just want to come and participate in business network, we're fine with that. We do want to encourage all members to contribute and participate in whatever way they can. It is important to us to give them that capability to show them how they can do that.

As far as relevance in the automotive market is concerned - The BMW in the end of last year launched the first GENIVI based IVI-system in their three, five and seven series models. That was the first instance of a market relevant production program being delivered. If you talk to BMW they would say it was a hard project, as is every hard project with a new technology. But they already launched a second and a third production program and they are reusing tons of codes that they used in their first program and they are far ahead in their second and third programs than they would have been if they would have started from scratch. I think the market relevance is there, it's been proven in the BMW model and it is beginning to be proven out in some of the other automakers.

RW: The board of GENIVI consists of representatives from different member groups (OEMs, first tier, software supplier). In the role of a GENIVI member they are acting as partners with their competitors. In the role of a CEO or of a first tier of an automotive manufacturer, *however, they try to maximize their profit at the expense of these partners. Does that influence their actions and behavior regarding GENIVI issues?*

SC: The right answer is absolutely yes. It influences, but what we encourage the board members to think about is that when they are sitting at the board table at a GENIVI board meeting, they are there for the sake of the organization. Yes, they represent their member company but they are there to ensure that the organization is successful and accomplishing its mission and so forth. There is, we often use the word coopetition, meaning we are cooperating as competitors and that's absolutely essential in an alliance like GENIVI. The one thing that helps us here is that GENIVI's scope of work is on what we call non-differentiating functionality. In other words, it is not the things that those organizations sitting around the board table really compete over, it is about commodity functionality that just has to work. It's overhead if multiple companies have to develop it. That helps that we have a scope of commodity functionality as opposed to trying to do things where those organizations actually compete.

RW: Is there something like a neutral instance?

SC: To some degree that is the role I play. I help the board come to decisions and at the end of the day we go by majority vote and so decisions are determined based on majority vote. When there is a very important decision and we are not making progress, then sometimes we pull back and give it some time for more discussion. But there is no permanent role of a mediator that is necessary because we work pretty well together.

RW: What are the strategic challenges and benefits of such a cooperation? Where do you see need for improvement?

SC: We can always improve. No organization is perfect. There is always opportunity for improvement. This goes back to your original topic, around a user-driven open source community. GENIVI was originally founded not as an open source community but has a member alliance who was producing open source software. I think over the years we've made several changes and evolved into an organization that is much closer to an open source community than we were at the beginning and frankly some of our legal formation documents that are still in place from five years ago hinder us from being that fully open community that we want to be. That is something that we're looking at perhaps improving.

RW: Is it rather an advantage or a disadvantage that there are so many people with different backgrounds and experience?

SC: It is an advantage. Open source is definitely an area where innovation can occur. In having different people at the table with different backgrounds and different experiences has been quite helpful. In fact, a number of our member organizations are heavily involved in open source software development for years whereas many of the automotive companies, the automakers and even some of the tier ones are relatively new to open source and so having that diversity around the table is actually a benefit.

RW: There is a functional organization chart on the GENIVI website which transparently reveals the task offices. Who is part of these groups, teams and offices? Are these primarily GENIVI members participating in these teams or are there also external employees?

SC: We have a limited number of contracted employees, such as myself. Playing leadership roles. There is myself and there is a program management lead. And some other specialty resources like a community manager that we contract. But that is a very limited number of those. The member volunteers do the bulk of the work, which is done in GENIVI. And they populate those teams and groups.

RW: In the FAQs on your homepage the question: "Will GENIVI license its technology to non-members?" is answered with: "This is under discussion." Are there any approaches how to proceed further in this matter?

SC: I need to go back and look at that FAQ. I am not sure which question that is specifically related to. GENIVI does two things. It delivers code which is always fully open source licensed, so there is no issue at all in making that work open to all interested parties through an open source license. The one thing that we do deliver that is not open to non-members is what we call our compliance specification. That specification is the basis for our compliance program, which is something that allows members to measure their products against the GENIVI specification. That program today is a member only benefit. We do not license or distribute the compliance specification outside the membership at this point. If that may be the one thing that is related in the FAQ that suggested GENIVI is not licensing all its deliverables but with the code it is all open source licensed and available publicly.

RW: *The GENIVI compliance specification program is a trademark, which is reserved for core projects that correspond to the required standards of the program. How did the compliance program come into existence? Were there any existing compliance programs you could use as a foundation?*

SC: No, it was not developed from scratch. It was based on a model that was used by the Carrier Grade Linux Working Group of the LINUX Foundation. This is another working group that is basically developing a distribution for equipment that is used in the networking space and they had a model that we felt like it worked for us in the first instance and so we adopted it. That was the beginning point for our compliance program.

RW: You mentioned that the associate members do not participate in development. According to "Join the GENIVI Alliance" every organization that confirms several agreements and pays the annual fee can join the GENIVI Alliance. This sounds like an invitation for free riding. What do you think of this phenomenon?

SC: I wouldn't say that associate members don't engage, I think that a number of them do engage. What I said earlier was, that a number of associates join to get into the network, to get visibility, to get their name and their organization recognized. Many of those associate members do actually engage in the organization and do produce really quality work and that GENIVI uses, but some don't. But like I said, we're fine with that. Be careful not to say that all associates are not really engaging, it's not true and I apologize if I made that communication incorrectly. Respective free riding - how do you define free riding?

RW: It means to me that someone is actually part of a community but only benefits and does not engage.

SC: I think that could be the case in certain cases. But when you think about building a business network, anytime you had any company to that business network, that company by the fact that is in the network is helpful and useful to that network. I wouldn't say that we have necessarily a lot of organizations who are just coming in to take and take and not give. But that is certainly possible with this organization to come in and just take.

RW: Besides the participation agreement, are there any contracts that "hold together" this consortium? If yes, who exactly concludes a contract with whom? Who is the principal, who is the agent?

SC: There is the participation agreement and then there is the Intellectual Property Rights Policy which governs how contributions are licensed within the membership. When you actually sign the participation agreement you are agreeing to the IPR policy and the bylaws of the organization like the articles of the association or the rules by which the organization manages to work. Really the participation agreement is the governing document in the relationship between the member and the organization and it references to the bylaws and the IPR policy.

RW: So there are only contracts between the organization and the members and not between the members additionally?

SC: No, not within the context with GENIVI. Obviously there is commercial arrangements that go on, but GENIVI is not with those commercial arrangements.

RW: Considering the fact that GENIVI is a non-profit organization. What exactly happens with the annual fees each member has to pay? Is this the only source of revenue? Which costs need to be covered with that money?

SC: As I mentioned before, we have some contracted staff. They perform key roles in the organization. There is that cost that has to be covered. There is the cost of operating the organization from a membership processing and financial management perspective. We provide a fairly comprehensive collaborative infrastructure with wikis and code repositories, issue trackers and so force. Those cost have to be covered. We run two all member meetings a year, that cost significant amount of money to put on. We also participate in a number of industry events throughout the year, that also have cost that have to be covered. Those are the main cost that have to be covered. Second part of that question was, is that the only source of revenue. We do charge limited amounts for participation in, what we call, member showcases or other marketing events. We may go to, for example, an industry tradeshow and GENIVI may buy a larger set of tables and then offer those back at a reduced cost to our members. That would be revenue that we realize from those members for those tables. But generally speaking, the membership makes up about 95% of the operating revenue.

RW: How do you assess the relevance of GENIVI as part of the open source movement? Can GENIVI be understood as an example for other business sectors?

SC: I think that GENIVI is learning, and we've been learning for about 5.5 years and will continue to learn. Open source was not something that the automotive sector was familiar with and it took a lot of effort on GENIVI's part. And still to this day we're still working very hard at helping automotive organizations become comfortable with open source development. There has been a lot of education, a lot of transition, a lot of paradigm shifts that have been necessary there and some automotive organizations are with it and understand it and some are still struggling with it because they are not used giving software away. That's how they make money - it's to sell software. There were some very large hurdles to get over and still some I had. I think we've done a good job helping the automotive organizations, the automotive ecosystem become more comfortable with open source. I think in that part we are probably highly relevant to the automotive ecosystem anyway. Whether we are producing at the speed with which the organizations or the ecosystem wants us to be in whether we do everything just as other industries had proven that it works I do not know that I can say that today, we are still learning and we have still some work to do there.

RW: What are the crucial points founders of a collaborative open source development organization should consider?

SC: I would say selecting the right scope of the development platform is essential to success. If your scope includes areas where there is a high degree of competition between your target participants, then it will not be successful. Picking the right scope is important. Educating and bringing the organizations to the point where they are comfortable using open source development techniques is essential. What we found particularly helpful was bringing in experts who have helped other industries or other communities get developed and give a sense of: You can do this! And here are some things you are going to need to do differently. So you know case studies or success stories, those sorts of things are things that I would say you should consider.

Interview Jeremiah Foster (JF), Community Manager GENIVI Alliance, on November 11th, 2014

RW: What is your educational background?

JF: Undergraduate work in Fine Art History and some graduate work in Computer Science.

RW: How did you get in contact with open source and with the GENIVI Alliance?

JF: Through open source by just living on the Internet and by being interested in the Internet. I wanted to program it. In the days that I was involved in the Internet in the late nineties, all that was pretty much powered by open source. Through GENIVI, I worked for other large companies. I was working for NOKIA for example and I was exposed to embedded operating systems. I got a job in a company that did that for automotive and they were GENIVI member. I guess you can say through business reasons that I got involved in GENIVI.

RW: So you are kind of a pioneer in open source?

JF: No, I am actually one of the latecomers, but I've been using it for a very long time. The people that I look to as a pioneer have really started in the eighties. The serious LINUX stuff, which is also part of it but not the whole thing, began in the early nineties and I did come along to in about late nineties.

RW: What exactly are your tasks and responsibilities as a community manager?

JF: Good question! It is hard to say, because a community manager is a relatively new title. It probably hasn't been around for much more than five years, maybe little more. The job description is not very well defined in general. At GENIVI it is quite well defined and specific. Most tasks are to represent GENIVI for example at Open Source Developer Conferences, to check the health of GENIVI open source projects, to provide support and assistance for projects moving at open source or want to enlarge at open source and to report and update GENIVI leadership on the status of GENIVI's open source projects. The community manager reports to GENIVI board as well as the system architect. It is both the technical and the management position.

RW: What exactly do you mean by "health of an open source project"?

JF: In open source you want a project to be alive, to be living. That means that the code is in a good shape. You don't want software with bugs obviously. You want to make sure that there is a maintainer who is taking a look at the code now and then, who is fixing the bugs if they come in, who is communicating with both the open source community as well as with GENIVI. That kind of thing.

RW: You actually play two roles. On the one side you are the community manager of GENIVI where you represent the interests of the GENIVI Alliance. And on the other side you represent the interests of Pelagicore as an open source technologists. Is there any potential for conflicts or can you manage to reconcile conflicting interests?

JF: There really is very little conflict. Simply because Pelagicore wouldn't contribute my services, if it didn't align already with their business goals. Since Pelagicore is an open source automotive company, GENIVI fits our business goals perfectly.

RW: Mr. Crumb told me about the so-called Coopetition in the GENIVI Alliance. How do you perceive this Coopetition?

JF: I perceive it as not functioning. It's the idea or it's the goal you strive for. You hope that competing companies or companies that compete in the market place will come together and contribute resources to, what is considered commodity software. But unfortunately, in reality the automotive industry is still not prepared for that step. While I believe that coopetition, should we say, the modus operandi, I don't believe we are there yet.

RW: Is it rather hindering than helpful to have people with so many different backgrounds and experiences in one organization or vice versa?

JF: I'm neutral on that question. If anything, it is helpful.

RW: Where do you see need for improvement?

JF: Well there are a lot of places where I would say that understanding how they can contribute and advance their own business interest. I don't think many companies understand that. I think they see open source rather conflicting with their business interests or if they see an alignment with their business interests they don't know how they can participate. While GENIVI has members representing surely thousands of software developers, we have maybe a handful that does much contribution.

RW: From your point of view, what are the crucial roles in an open source foundation like GENIVI?

JF: It is difficult to say. GENIVI would have its own idea what the crucial roles are and they would obviously be the CEO, the project management, the system architect lead and the community manager. However, in a real open source project you would have neither of those positions. In a real open source project you would have developers. If I am comparing it to open source I would say developers, if I am comparing it to the automotive industry I would say project management and system architect lead are probably the most important.

RW: You mentioned the "real" developer-driven open source projects. Debian is one of them right?

JF: Yes, it is the largest one, the most important one.

RW: What are the differences between a user-driven and a developer-driven open source community?

JF: It is difficult when you say user-driven. I am not really sure what user-driven means. Certainly not end-user driven. In other words the person driving the car. That is not the person involved in GENIVI.

RW: No.

JF: User in this case mean car companies.

RW: Yes, exactly.

JF: What would be the difference between GENIVI, which is a car company driven alliance and say Debian. I think pretty much everything. There is a fundamental difference and that probably with commercial indicators. In GENIVI, everything is built around commercial interests. Software is going to be built to a business plan and to sell more cars. Nothing in Debian is ever considered in that matter. Debian doesn't even accept money. They are fundamentally different and the commercial aspect is the key. The DNA of companies is competing and in Debian it is not competition that is the goal.

RW: What about the degree of participation?

JF: Vastly more in a developer-driven. Because your contribution is valued otherwise in a user-driven. Your company decides how much and what you contribute.

RW: So you actually do not have much decisive power in a user-driven?

JF: No, there is very little. There is very little that you are doing that you are able to decide. You have to follow a project plan. You have to follow a work package whereas in a developer-driven community the things that interest you what are you focus on. It's night and day.

RW: What do you think of the phenomenon of free riding? This means that some companies are actually part of the community but only benefit and do not really engage in the development of new software.

JF: It is a gigantic phenomenon and it is only getting worse. It is a huge issue. Companies like Valve, which is a very large gaming company uses Debian, Milky uses Debian, Google uses it, many and many of companies use it and don't contribute back at all. I don't know if there is a remedy and I think in the end it is probably good because if you have lots of users you get lots of bugs. And if you fix those bugs you'll have high quality software. That's what's allowed Debian to maintain its number one position for example as the number one webserver software system despite the fact that there is no commercial business behind it. I think this free riding has good sides and bad sides.

RW: The GENIVI compliance programs are currently for GENIVI members only, However there are GENIVI open source projects accessible for each and everyone. What exactly are the purposes of and the differences between these two types of projects?

JF: The purpose of the compliance program was essentially to thrive the usage of GENIVI software. Because that's very difficult to do with open source it turned pretty much into a marketing program and hasn't had a significant impact in open source in any case. It has been an affective tool for automotive companies to say that they are GENIVI compliant and so the compliance program is successful from a marketing standpoint but if you want to say use GENIVI software you probably have no reason to be GENIVI compliant and in fact many companies just come and use GENIVI software and are members of GENIVI. The two separate projects have two very different purposes. One is to build up a thriving ecosystem, those are the open source projects. And the other is to market this platform. That's the compliance program.

RW: There is also a third category Works with GENIVI Products. What is the purpose of this third category?

JF: I think there is some lack of clarity about that. Again it's about branding and about marketing. It's designed to be a more lightweight marketing approach for those that don't want full GENIVI compliance. So that's pretty much the purpose of that.

RW: The code behind the GENIVI platform comes from the following three different sources:

- Upstream projects where the required functionality is already built (e.g., kernel.org)

- Upstream projects where a good starting point for required functionality exists upon which GENIVI can build additional automotive functionality (e.g., connman.net)

- GENIVI member projects where IVI code is developed because nothing is available in the open source community as a good starting point.

What exactly means a "good starting point"? Which criteria need to be fulfilled in order that there is a "good starting point"?

JF: A good starting point is certainly a significant set of features. If you have a code of software that has those particular features, it's a very good start. Secondly, it has to have a certain quality. The code doesn't have to be bug free, but it does have to run without crashing frequently. Then it has to have additional things like a maintainer, somebody you can talk to. If you found a bug you really want to know that the maintainer is going to fix it or at least be open to your ideas to change it. It would also have to have other infrastructure around it - a way to file bugs or maybe a mailing list to get in touch with other users of the software. Then finally it probably has to have other users. There is a snowball effect. Once a number of people said: *Yeah, this software is pretty good. We used it for our purposes and we improved it.* That makes a lot of other companies to join in. There is sort of due diligence that one does in evaluating software projects before they are adopted in GENIVI.

RW: Which of these three options is the most common one? Projects where functionality is already built?

JF: Yes, overwhelmingly all the software the GENIVI needs is already built. GENIVI really has very few projects of its own. They are specific to an automotive.

RW: LINUX is the most prevalent open source kernel. GENIVI as well is based on the GNU/LINUX operating system. If building a new community, would you recommend the GNU/LINUX as a basis or are there any other operating systems, e.g. Debian, you would recommend?

JF: It depends on the use. But almost exclusively I would recommend GNU/Linux. In some cases I might recommend BSD. Unfortunately, they have suffered a great deal because Apple uses BSD a great deal and they hauled out that project with taking all the good developers and they don't contribute back. It doesn't move with the same paces. GNU/Linux is much more focused on, for example, web servers. It is not really suited for the embedded world ware. Yes almost exclusively Linux.

RW: Are there any other programs, systems or software that are crucial for successfully running an open source platform?
JF: There are some software tools that are essential I would say. But I think the fundamental pieces are the C-Library and the Linux Kernel. I think you can pretty much survive on those two things. Plus little bit more.
RW: What is C-Library?
JF. This is a set of functionality written in a specific programming language called C.
RW: On the website of Pelagicore you make the statement: „It's time to put traditional software solutions where they belong. Way behind us." Can you explain which message you exactly intend to convey by this statement?
JF: That statement is meant to convey that traditional software solutions are often bound up in traditional licensing and that is often very expensive and not flexible for our customers needs. By using open source we can use a more modern, perhaps nontraditional approach. But that approach is much more effective with customers needs.
RW: In a long-term prospect. Do you think that this model of shared software development does replace proprietary software or at least gain the upper hand in the software market?
JF: I think it already has. I think someone recently said that there's been no significant platform innovation done in the last decade that hasn't been in open source. I don't think that you can actually do very much not be open source. I think it already dominates. In fact that might be a problem. It might be too dominant.
RW: What means too dominant?
JF: There are a couple of things I think are problematic. For example, one, it is very difficult for open source projects to register as non-profit organization in the United States. That's a big problem, which means that commercial entities dominate open source projects nowadays. It used to be hobbyist and engineers doing stuff in their free time now there are very large corporations like Apple, Google, Microsoft etc. We don't have the kind of flexibility and freedom that we used to have. The commercial pressure the change on software is very great and well that's a benefit in some way. You can make money doing this, where you couldn't before. It is really changing our software and changes the system we built. There are some social implications that are some problems.
RW: How do you assess the relevance of GENIVI as part of the open source development? Can GENIVI be understood as an example for other business sectors? Do you think there is a need for more organizations like GENIVI?
JF: When you look at other business sectors, let's say automation or mobile or the Internet of Things, we find that they already have organizations. In many ways GENIVI is sort of the last missing piece in the puzzle. And as far as the influence goes, I think the goal of BMW and Intel, the two main people behind GENIVI, was to ensure that open source could come to the table as an equal partner, as a supplier to BMW and Intel. Then I think they've been very successful. About 200 companies had come into GENIVI and wanted to market software to the car companies based on open source. They had created a thriven ecosystem.
RW: What are the crucial points, founders of a new open source development platform should consider? What are your recommendations?
JF: My strongest recommendation is that you need to start as an open source project. Its very difficult if you start as a proprietary or just a commercial alliance and then try to become open source, which is what GENIVI did. If you start from the beginning as a completely open source project you have much more success.

Interview Ralph Mueller (RM), Managing Director Eclipse Foundation Europe GmbH, on November 20th, 2014

RM: Die erste Frage bezieht sich auf meinen Bildungsweg. Ich bin Maurer von Beruf. Das war ich aber nur für kurze Zeit. Mir wurde es dann zu kalt draussen. Ich habe ein ganz normales Informatikstudium abgeschlossen. Ich bin Diplominformatiker und habe an der Universität in Darmstadt studiert und anschliessend eine ganz normale Informatikkarriere gemacht. Vom Programmierer über Entwickler und Projektmanager bis hin zum Abteilungsleiter. Ich war bei den verschiedensten Firmen. Möglichst immer erst kleine Firmen, die etwas Interessantes gemacht haben und dann von Grossen aufgekauft wurden. Da war ich schnell wieder weg. Dann ging das wieder von vorne los. Stationen in meiner Karriere, die interessant sind und von welchen Sie die Namen vielleicht kennen, waren Siemens Nixdorf und eine kleine Firma Object Technology International in Kanada. Diese wurden schliesslich von IBM gekauft. 2004 habe ich beschlossen IBM zu verlassen und habe 2005 bei der Eclipse Foundation angeheuert.

RW: Wie hat sich das mit Open Source entwickelt? Wie sind Sie auf diese Schiene geraten?

RM: Eclipse ist ein Projekt, das am Anfang hauptsächlich von der IBM getrieben wurde. Eines der Teams, das die Basisentwicklung von Eclipse mitgemacht hatte, war das Team um Erich Gamma herum. Ich weiss nicht ob Ihnen der Name etwas sagt?

RW: Nein.

RM: Erich ist Schweizer wie Sie. Das heisst als ich noch bei dieser kleinen Firma OTI (Object Technology International) gearbeitet habe, haben wir in Zürich ein Labor eröffnet und dort Erich Gamma und sein Team angestellt. Erich ist ein ziemlich bekannter Mensch. Wenn Sie nach Ihrem Master noch Lust haben und ein bisschen Informatik machen wollen, um dort einen Doktor draufzusetzen, dann würden Sie eine ähnliche Karriere machen wie er. Er ist wirklich sehr bekannt. Er hat auch ein sehr interessantes und wichtiges Buch geschrieben. Das berühmte Patterns-Buch, in dem er Vorgehensweisen aus der Architektur ableitet, deren Basis auch häufig aus der Informatik stammt. Wenn Sie heute Patterns auf Google nachschauen, finden Sie 1000 Sachen. Da hat er im Prinzip angefangen. Ich war Leiter dieser Niederlassung in Zürich und habe bis 2001 mit diesem Team zusammengearbeitet. In dieser Zeit ist der erste Eclipse Code entstanden. Viele Leute, die an der ursprünglichen Version von Eclipse gearbeitet haben und aus diesem Unternehmen OTI kamen, kannte ich natürlich. Ein guter Bekannter, Mike Milinkovich, wurde 2004 der Executive Director der Eclipse Foundation. Mike hat mich 2005 gefragt, ob ich Lust habe für die Eclipse Foundation zu arbeiten. Das ist kein Standardweg wie ich dazu gekommen bin. Ich bin am Anfang damit in Berührung gewesen und später durch persönliche Kontakte dort eingestellt worden.

RW: Wie sehen Ihre Aufgaben bei der Eclipse Foundation genau aus?

RM: Da muss man vielleicht zuerst ein paar Worte über die Eclipse Foundation sagen. Die Eclipse Foundation ist eine unabhängige Organisation, eine sogenannte neutrale Organisation, die 2004 gegründet wurde, um IBM als den Führer dieses Projektes abzulösen. Mit dem Ziel, dass man auch andere grosse Unternehmen (bspw. Oracle, SAP, BEA) aus dem Java-Umfeld miteinbeziehen und damit eine neutrale Governance schaffen konnte, die nicht mehr nur von IBM sondern von mehreren getrieben wurde. Die Eclipse Foundation wurde basierend auf Erfahrungen, die alle diese Unternehmen im OS-Bereich also in Apache, Linux usw. gemacht haben, so ausgerichtet, dass sie einerseits neutral ist und andererseits verschiedene Services für die Eclipse Projekte und auch für das Ökosystem um diese Eclipse Projekte herum zur Verfügung stellt. Da kommen wir sicher später nochmals darauf zurück. Wichtig ist an dieser Stelle, dass eine der Aufgaben ist, dieses Ökosystem zu betreuen. Oft werden auch andere Wörter, wie Community, für dieses Ökosystem benutzt. Eine meiner zentralen Aufgaben ist es dieses Ökosystem in Europa zu betreuen und andererseits natürlich dafür zu sorgen, dass die EF nachhaltig operieren kann.

Das heisst wir sind in der Hauptsache durch Mitgliedsbeiträge finanziert. Deshalb ist offensichtlich eine meiner Hauptaufgaben auch das Business Development (Mitgliedergewinnung). Im Rahmen dieser Ökosystem-Betreuung sind wir auch aufgefordert dies, das und jenes zu tun. Da gehört auch die Veranstaltung von Konferenzen dazu. Meine Verantwortlichkeit ist es unsere beiden europäischen Konferenzen in Frankreich (Toulouse) und Deutschland (Ludwigsburg) zu managen. Dann gehören viele andere kleine Aufgaben dazu. Wie kann ich es schaffen, dass ich Mitgliederwertschöpfung generiere? Wie schaffe ich es interessierte Leute miteinander in Kontakt zu bringen? Eine weitere zentrale Aufgabe ist auch Technologie Akquise, d.h. bei Eclipse ging es am Anfang um das Java-Development Tool und um SprachIDEs, also Entwicklungsumgebungen. Im Laufe der Zeit hat sich dieser Technologierahmen, den wir heute mit unseren ca. 240 Projekten haben, verändert und ausgebreitet. Ich habe auch Verantwortung für meine Mitarbeiter, die ich betreue. Aber das ist wahrscheinlich selbstverständlich.

RW: Etwas spezifischer zu den Working Groups (WG). Die EF war ursprünglich eigentlich developer-driven, d.h durch IBM ins Leben gerufen worden. Diese Working Groups sind jedoch eher user-driven.

RM: Lassen Sie uns einen Schritt zurückgehen und es uns aus einer anderen Perspektive betrachten. Was heisst developer-driven?

RW: Dass es von Software-Firmen initiiert wurde. User-driven wäre wenn es von denjenigen, die die Software benutzen initiiert wird.

RM: Gut, diese Eclipse Foundation ist auch user-driven gewesen. Die Klientel für die Benutzung der Artefakte war jedoch die Software-Industrie. Keines dieser Unternehmen, das daran mitgearbeitet hatte, hat das zum Spass gemacht. Die wollten alle Resultate erzeugen, nämlich diese Java-Plugin-Development-Plattform, die sie später auch kommerzialisieren wollten. Das ist eine ganz wichtige Geschichte. Diese ganze Eclipse Foundation war von Anfang an sehr stark business-orientiert. Es sollte eine Plattform gebaut und betreut werden, welche die Software- oder Beratungsfirmen oder wer auch immer, kommerziell einsetzen konnten. Insbesondere haben IBM und andere das Ziel gehabt, diese Plattform aufzustellen gegen die .NET Plattform von Microsoft. Das war die Hauptmotivation, die IBM und die anderen hatten. Wenn Sie so wollen, hat man schon relativ früh gesehen wie eine solche user-driven Organisation funktionieren könnte, wobei die Hersteller und die Nutzer dieselben Firmen waren. IBM bspw. hat diese Plattform mit anderen zusammen hergestellt und hat dann das Thema für Produkte in IBM umgesetzt. IBM hat, Stand heute, eine grosse Anzahl von Produkten, die auf dieser Plattform basieren. Diese Working Groups waren auch nicht eine Idee, die wir selber hatten. Die Idee kam aus der Industrie. Genau genommen aus der Aerospace-Industrie. Dort hat Airbus irgendwann mal angefangen und hat sich gefragt, wie sie eine Entwicklungsplattform für die Aerospace-Software, also die Software im Flugzeug, durch einen ähnlichen Ansatz bekommen könnten. Die haben dann angefangen zu experimentieren und ihr eigenes loses Konsortium gegründet. Ich weiss nicht, ob Ihnen das Wort Topcased irgendwas sagt.

RW: Nein.

RM: Sie haben mit Partner zusammen und aus Research-Projekten finanziert diese Topcased.org, basierend auf der Eclipse Plattform, gegründet und haben angefangen kollaborativ Tools zu entwickeln. So sind wir ins Gespräch gekommen. Als diese Plattform dann reifer und einsatzbereiter wurde, hatte Airbus ein ähnliches Problem wie das seinerzeit IBM hatte. Die mussten sich überlegen wie sie das in eine unabhängige Organisation überführen. Da gab es drei Ansätze. Es einerseits so zu lassen, was nicht möglich war, zweitens eine eigene Organisation zu gründen, so wie GENIVI das später gemacht hat, oder mit uns zu sprechen, was sie dann auch getan haben. Wie wäre es denn, wenn wir von euch soviel wie möglich übernehmen, von dem was sich in den damals letzten sieben Jahren an Governance, Vorgehensweisen, Prozessen und an Services entwickelt hat und uns unter euch, der Eclipse Foundation, aufhängen. Das war eine Entwicklung, die wir am Anfang nicht voraus gesehen hatten und die dann in Diskussionen mit dieser Gruppe um Airbus herum entstanden ist. Das haben wir dann auch ausgearbeitet und die ersten Dokumente geschrieben. Damals ist auch dieser Working Group Process entstanden und das Ende war, dass sich die Polarsys Working Group innerhalb der Eclipse Foundation gegründet hat.

RW: Das war die erste Working Group?

RM: Es gab schon andere Ansätze, welche jedoch bei weitem nicht so fokussiert waren. Die waren mehr um eine bestimmte Technologie, als um eine bestimmte Problemlösung herum. Das ist vielleicht eine ganz wichtige Geschichte. Sie hatten später auch noch gefragt, wann sich Working Groups sinnvoll gründen. Meine Erfahrung ist heute, dass es ein Problem geben muss, dass man lösen will, an dem man dann fokussiert arbeitet. Kennen Sie die Elinor Ostrom?

RW: Nein.

RM: Das ist die einzige Frau in der BWL, die einen Nobelpreis gewonnen hat. Die Elinor Ostrom hat sich mit sog. Commons beschäftigt. Sagt ihnen der Begriff was?

RW: Nein.

RM: Kennen Sie das Allmendfeld?

RW: Ja.

RM: Das sind in der Vergangenheit die Allmendfelder gewesen und wurden von allen unterhalten. Alle haben einbezahlt und haben dadurch die Weide und den Hirtenbuben bekommen, der auf alle Tiere aufgepasst hat. Typischerweise war die Wiese im Besitz von allen oder vom Dorf. Das heisst, da wurden die Kühe immer auf die Allmend getrieben, der Hütejunge (also ich in dem Fall) hat auf die Kühe aufgepasst und abends haben sie ihre Kühe wieder abgeholt. Das nennt man die Commons. Im Englischen gibt es viele Diskussionen über Commons. In der libertärianischen Betrachtungsweise spricht man auch von der Strategy of the Commons. Das ist ein grosses Thema. Diese Strategie der Commons geht davon aus, dass Commons mit der Zeit sterben. Frau Ostrom hat in ihrer Nobelpreis gewürdigten Arbeit nachgewiesen, dass Commons durchaus Bestand haben können, solange sie fokussiert sind, solange es eine klare Richtlinie gibt wer denn was zu sagen hat, wie das ganze betrieben wird usw. Und das Commons eigentlich nur kaputt gehen, wenn entweder der Fokus verloren geht, die Finanzierung unklar ist oder kein Bedarf mehr für diese Common existiert. Irgendwann hat sich die Viehwirtschaft in der Schweiz sehr stark intensiviert. Ich weiss das alles von einer Freundin, die Architektin ist und sich mit der Intensivierung von Viehwirtschaft in der Schweiz beschäftigt. Die baut an allen möglichen Stellen hochmoderne Kuhställe. Und das ist dann wirklich dort passiert, d.h. für diese Commons gab es wenig Willen und es gab Stacheldraht und Elektrozaun, der den Hirtenbuben irgendwann überflüssig gemacht hat. Das war eine weite Ausschweifung. Diese user-driven Working Groups haben alle ein klares Problem. Im Fall von Airbus und Co. war das, dass man langfristig eine Toolkette weiterentwickeln musste, wo man sich nicht darauf verlassen konnte, dass die einer oder auch mehrere Hersteller alleine liefern können und dass diese Hersteller dann auch lange genug leben, um den Support für dies Toolchain weiter zu leisten. Deshalb ist man auf diese Idee mit einer OS-Working Group gekommen.

RW: Was sind Ihrer Meinung nach in der Foundation selbst und auch in den Working Groups die wichtigsten Rollen?

RM: Also wir in der Foundation haben alles verschiedene Aufgaben. Wenn Sie bei uns auf der Website unter diesem Working Group-Thema schauen, sehen sie auch eine Liste dieser wichtigen Aufgabe. In der Hauptsache ist das, dass wir die Einhaltung der Governance beaufsichtigen, also wir sind eine Art Aufsichtsbehörde und auf der anderen Seite, stellen wir noch andere Services bereit. Wir stellen den Hirten, also denjenigen, der das Ökosystem verbreiten hilft, wir stellen das sog. IP-Management zur Verfügung. Wir betreuen die Entwickler, sowohl im Entwicklungsprozess als auch mit der nötigen Infrastruktur dazu und wir helfen denen auch, dass sie entsprechend Marketing und vielleicht sogar Vertrieb machen können für ihre Themen.

RW: Das heisst die Abhängigkeit ist sehr gross? Ohne die Foundation müssten sie alles selbst aufbauen.

RM: Genau. Da wir das jetzt können und das mittlerweile über 13 Jahre geübt und verbessert haben, können wir diese Leistung kostengünstig auch zur Verfügung stellen. Das ist diese Economies of Scale. Anstatt, dass sich das jeder selber aufbauen muss, was kostenintensiv und auch intensiv im aushandeln der Bylaws und der Governance ist, können wir im Prinzip Out of the Box zur Verfügung stellen. Als Anbieter solcher Out of the Box Lösungen sind wir ganz gut aufgestellt. Wir haben zu Frage 5 noch eine kleine Thematik vergessen, nämlich die wichtigen Rollen in der Working Group. Das ist eigentlich ganz klar. Es muss in jeder Working Group jemanden geben, der das haben will und jemanden der das produziert. D.h. es ist immer ein Spiel zwischen dem Produzenten und dem Konsumenten, die durchaus manchmal dieselben sein können. Im Falle von IBM sind es auf der einen Seite die Produktverantwortlichen, die auf dieser Eclipse Plattform ihre Produkte liefern wollen und auf der anderen Seite die Entwickler, die diese Plattform entwickeln. So entstehen auch die Anforderungen an diese Plattformen. Wobei das dann nicht nur IBM ist, sondern die ganze Benutzercommunity.

RW: Wie nehmen Sie das Trittbrettfahrer-Problem im Open Source Business wahr?

RM: Das ist eine schwierige Frage. Wir sprechen eigentlich ungern von Trittbrettfahrern. Trittbrettfahrer wäre ja bspw. irgendjemand, der die Eclipse Plattform nimmt, daraus seine Produkte macht und wir hören und sehen von dem nichts mehr. Nehmen wir mal eine Schweizer Firma aus Bern. Da gibt es eine grosse Firma, die den Public Transport reguliert, die SBB. Wenn wir gemein wären, könnten wir sagen, dass die SBB ein Trittbrettfahrer ist. Die SBB liefert nichts an die Eclipse Foundation. Trotzdem sind wir froh, dass die SBB Eclipse benutzt, weil sich um die SBB herum dann auch wider ein kleines Ökosystem entwickelt. Dieses Ökosystem, das sich mit Public Transport beschäftigt und wo Lösungen im Bereich Public Transport entstehen, wo Arbeitsplätze im Bereich des Eclipse Ökosystems entstehen, wo diese Technologie nachhaltig eingesetzt wird. Und natürlich kommt dann in irgendeiner Art und Weise wieder Rückfluss in die Eclipse Foundation. Wir hatten die Eclipse Con Europe. Da waren drei Leute von der SBB und haben ihre Themen miteingebracht. Es gibt ein ganzes Ökosystem in und um Bern herum und vielleicht sogar noch weiter, wozu dann auch Firmen wie CSC gehören, die schliesslich mit Eclipse Geschäfts machen. Dadurch wird diese Plattform natürlich auch wieder beeinflusst und verbessert. Irgendwann fangen dies CSCs an Bug Reporters einzustellen. Natürlich gibt es auch Firmen, von denen wir eigentlich nie was hören oder noch nie etwas gehört haben und die auch keine Bug Reports machen. Aber was wir oft erleben ist, dass diese Firmen leiden. D.h. wenn sie keinen Rückfluss tätigen, verlieren sie eigentlich. Das heisst, je reifer die werden (wir nennen das OS-Maturity Model für Organisationen), desto mehr werden sie anfangen zu verstehen, dass der Rückfluss nicht nur aus Trittbrettfahrergründen, also aus altruistischen Gründen gut ist, sondern dass er ihnen auch wieder hilft Geld zu sparen oder vielleicht Geld zu verdienen. Wir sehen das Trittbrettproblem ziemlich gelassen. Wenn sie fragen wie gross die Zahl der Firmen ist, die nur aus Marketing oder ähnlichen Gründen beigetreten sind, würde ich schätzen, dass es mindestens die Hälfte der Mitglieder ist.

RW: Das ist ein grosser Anteil.

RM: Ja, diejenigen, die schliesslich tatsächlich produzieren, sind sogar in der Minderheit.

RW: Die Working Groups können alles von der Eclipse Foundation benutzen, wie auch die Bylaws und die IP-Policies. Zusätzlich gibt es noch die Group Charter.

RM: Die Group Charter ist nochmals eine Spezialisierung unterhalb der Bylaws. Man kann nichts, das in den Bylaws steht, ausser Kraft setzen, aber man kann sich noch spezielle Rollenverteilungen, Aufgaben oder Regeln geben, nach denen man vorgehen möchte. Dafür sind die Charters da. Abhängig davon wie die Working Group aufgestellt ist, werden die von Working Group zu Working Group ähnlich oder ganz anders aussehen.

RW: Diese drei Dokumente sind die wichtigsten rechtlichen Grundlagen?

RM: Ja, aber ich würde es nicht rechtliche Sicht, sondern Governance Sicht nennen. Die sind zum Teil natürlich rechtlich bindend. Wenn z.B. jemand den Eclipse Mitgliedschaftsbeitrag unterschreibt, ist er schon rechtlich bindend. Das ist ein ganz normaler Vertrag. Ein weiteres Dokument, das ausschlaggebend ist, sind die Participation Agreements, die jede Working Group selber definiert. Dort wird bspw. festgelegt, was die Mitglieder an Leistung in die Working Group miteinbringen müssen. Soweit dies definiert ist, was nicht bei jeder Working Group der Fall ist. Und was vielleicht auch an Geldmittel an die Eclipse Foundation für zusätzliche Services abgeführt werden muss und soll. Die Polarsys Gruppe ist natürlich sehr stark daran interessiert, dass eine Longterm-Support Infrastruktur innerhalb der Eclipse Foundation zur Verfügung gestellt wird, weil die ihre Software im Prinzip über 60 Jahre warten müssen. Da sind wir auch aufgefordert gewesen, eine Longterm-Support Infrastruktur zu bauen. Das war nicht nur für die Polarsys Gruppe. Da waren auch Mitglieder wie IBM, SAP oder Computer Associates sehr interessiert daran. Da leisten sie auch einen finanziellen Beitrag, um diese Infrastruktur erst mal zu bauen und dann zu betreuen und zu warten.

RW: Ist das diese Eclipse Solution Membership oder ist das nochmals was anderes?

RM: Wir gehen davon aus, dass jeder der in einer Working Group teilnehmen will zumindest Eclipse Solution Mitglied sein muss. Im Allgemeinen zahlen sie zumindest diesen Beitrag an die Eclipse Foundation. Für sehr grosse Unternehmen sind das USD 20,000 pro Jahr, im Falle von kleineren Unternehmen sind das USD 5,000. Dann vereinbaren sie untereinander, also innerhalb der Working Group, ob sie noch weiteres Geld für erweiterte Services an die Eclipse Foundation zahlen wollen. Die Polarsys Working Group wollte einen eigenen Produktmanager haben, den wir dann auch gestellt haben und der zur Hälfte aus Polarsys Geldern finanziert wird.

RW: Welche anderen Kosten werden durch diese Mitgliederbeiträge gedeckt?

RM: Die Basis-Services sind damit gedeckt. Das wird an uns überwiesen. Aber wie gesagt sind das nur die Mitgliedsbeiträge. Wenn sie erweiterte Services haben wollen, wie bspw. eine grosse Menge Code, die das IP-Management innerhalb des nächsten halben Jahres erledigt haben soll, haben wir natürlich ein Problem, das über unsere Standardkapazität hinaus geht. Da müssen wir eine Person einstellen, die irgendwie bezahlt werden muss. D.h. die würden sagen, wir vereinbaren für die nächsten zwei Jahre, dass wir eine halbe Person im IP-Management mitfinanzieren und das wird dann entsprechend angerechnet. Wenn sie sich in den verschiedenen Gruppen die Charters anschauen, sehen sie auch was für Extrabeitrage noch geleistet werden. Dieses Budget wird von uns genutzt. Da gibt es natürlich auch ein offenes Reporting, welches wir dann wirklich leisten.

RW: Es kann jede Working Group selbst festlegen wie viele und was für welche Mitgliederbeiträge sie machen will?

RM: Genau. In den stark user-getriebenen ist es z.B auch so, dass die Mitglieder, die aus dem Using Bereich kommen oft auch die Steering Members sind und andere Aufgaben, Pflichte und Rechte haben als zum Beispiel die Zulieferer.

RW: Diese Steering Mitglieder sind Firmen, die im gleichen Komitee sitzen. Gibt es da Interessenskonflikte oder löst sich das mit dem gemeinsamen Zweck der Organisation wieder auf?

RM: Bisher haben wir festgestellt, dass es natürlich Konflikte gibt. Diese lassen sich aber durch den gemeinsamen Fokus wieder auflösen.

RW: Gibt es so etwas wie eine neutrale Instanz, die in solchen Fällen vermitteln kann oder müssen sie das untereinander klären?

RM: Die neutrale Instanz sind wir. Das ist auch unsere Aufgabe. Wobei wir mehr vermitteln, als Direktiven ausgeben.

RW: Angenommen, die Mitglieder können sich nicht einigen. Hat die Eclipse Foundation so was wie ein Vetorecht?

RM: Ja, dann können wir eine Charter oder auch Beschlüsse vetoen. Stellen Sie sich vor, es gäbe eine Working Group, wo man sagen würde, wir sind eine Working Group der Medizingerätehersteller und in eine bestimmte Klasse dürfen nur die Medizingerätehersteller rein. Die dürfen bspw. nur ins Steering Committee. Dann würde wir das vetoen, weil das bzgl. Kartellrecht schon fraglich und bedenklich wäre. Es gelten weiterhin die ganzen OS Regeln, die auch die Grundlage für die Eclipse Public License sind, das sind die zehn commandments von der OSI. Also offen, transparent, nicht nur für die Medizinindustrie usw. Die kennen sie wahrscheinlich.

RW: Ja, die habe ich schon öfters angetroffen.

RM: Das sind so Grundregeln. Wenn sich eine Working Group darüber hinwegsetzen würde, würden wir vetoen.

RW: Es gibt bei verschiedenen Working Groups Zusammenarbeiten mit Academics.

RM: Das ist von den meisten gewünscht. Hochschulen und Forschungsinstitute haben eine spezielle Rolle. Die können oft kostenfrei mitarbeiten, weil es ja gewünscht ist, dass die sich dort mit ihrem Wissen und ihren Forschungsarbeiten einbringen. Die reifste Working Group ist sicher Polarsys. Wenn Sie sich da die Mitgliederliste anschauen, werden sie viele Hochschulen sehen.

RW: Die unterstützen quasi die praxisorientierten Unternehmen mit der nötigen wissenschaftlichen Basis?

RM: Oder auch mit Advanced Research, wie das bei der Polarsys der Fall ist. Dort werden neue Methoden und Verfahren miteingebracht, die frisch aus der Forschung kommen oder sogar als Auftragsarbeit direkt als Forschungsaufträge vergeben werden.

RW: Der Eclipse Working Group Process beschreibt sehr ausführlich, wie solch eine WG zustande kommen kann. Was sind die ausschlaggebenden Kriterien, welche es legitimieren, dass eine Working Group unter der Eclipse Foundation agieren kann? Gibt es auch solche, die abgelehnt werden?

RM: Der Prozess, dass eine Working Group zur Eclipse Foundation kommt oder dass sich eine Working Group formiert und mit uns diskutiert, ist nicht irgendwas wo wir einen Brief bekommen und die sagen, hallo wir würden gerne bei ihnen eine Working Group machen. Das ist typischerweise, ein im Schnitt 18 Monate dauernder Prozess indem man sich trifft und anfängt zu diskutieren. Man lernt sich kennen, indem man langsam, langsam diese Entscheidung trifft. Von daher, wenn wir mal diese 18 Monate im Schnitt durchgegangen sind, haben wir uns schon dort verständigt, dass wir kompatibel sind. D.h. es gibt da keine Ablehnungen in dem Sinn.

RW: Das heisst das würde sich schon in einer früheren Phase erledigen.

RM: Das würde innerhalb dieser 18 Monate schon passiert sein.

RW: Nebst der Eclipse Foundation gibt es auch noch die Linux Foundation. Inwiefern schätzen sie diese als Konkurrenz ein?

RM: Die schätzen wir schon als Konkurrenz ein. Wir haben auch oft Diskussionen, bei denen wir mit potentiellen Working Groups reden, welche dann entweder zu uns oder zu der Linux Foundation gehen. Da gibt es schon Paralleldiskussionen. Wobei wir glauben, dass wir dort aufgrund unserer Services, die wir anbieten, die die Linux Foundation übrigens nicht für ihre Working Groups anbietet, einen gewissen Vorteil haben. Diesen können wir aber nicht immer in guter Art demonstrieren oder sie werden nicht verstanden. Im Nachgang müssen die Working Groups bei der Linux Foundation das alles selber aufbauen.

RW: Aber es gibt schon solche Unternehmen, die sich an die Eclipse Foundation und gleichzeitig auch an die Linux Foundation wenden und dann wird um sie geworben?

RM: Wenn sie die Liste der Gruppen, die sich jetzt bei der Linux Foundation gegründet haben, anschauen, haben wir sicherlich mit der Hälfte auch diskutiert. Nur gibt es dort viele Entscheidungen, die getroffen werden. Zum Beispiel können wir keine GPL basierten Projekte annehmen. Diese werden dann mit einiger Wahrscheinlichkeit zur Linux Foundation wandern. Es gibt dort natürlich auch politische Entscheidungen. Das haben wir gerade erst erlebt. Firmen die schon sehr stark in der Linux Foundation engagiert sind, treiben in Richtung Linux Foundation. Und umgekehrt natürlich auch.

RW: Nebst der Eclipse Foundation und der Linux Foundation gibt es auch unabhängige Plattformen wie die GENIVI Alliance.

RM: Sie können da auch die openETCS dazu zählen, wenn Sie wollen. Die gehört sicher auch in den Rahmen rein. Die ist etwas weniger bekannt, kommt auch aus dem Public Transport Bereich. Das sind alles Organisationen, die sich selber gründen und versuchen diese Dinge mit mehr oder weniger Erfolg zu tun.

RW: Worin liegt der Anreiz selber eine Organisation zu gründen, wenn man sich doch einer Out of the Box Foundation wie der Eclipse Foundation anschliessen könnte?

RM: GENIVI ist 2008 entstanden und hat sich zu der Zeit konstituiert. Zu dieser Zeit war es so, dass wir noch überhaupt nicht in der Lage gewesen wären, das für GENIVI anzubieten. Wir hatten die Erfahrung noch nicht. Die hat sich erst ab 2008 aufgebaut und unsere erste ernsthafte WG haben wir 2011 mit Polarsys gegründet.

RW: Wäre es theoretisch auch möglich, dass sich die GENIVI Alliance auch jetzt noch im Nachhinein der EF anschliessen würde?

RM: Das ist durchaus denkbar. Es gab auch die einen oder anderen vorsichtigen Unterhaltungen, die typischerweise von Einzelpersonen getrieben waren. Sie wissen sicher, dass BMW bei GENIVI einer der Gründer gewesen ist. BMW ist mittlerweile bei uns in der Automotive Group und auch in der Open MDM Group organisiert und es kann durchaus sein, wobei ich keinen Hinweise darauf habe, dass die GENIVI Organisation irgendwann entscheidet, sich unter der Eclipse oder Linux Foundation aufzuhängen. Weil wir hier dreifache Kosten haben und Services, die wir Stand heute auch gar nicht anbieten.

RW: Wenn Sie sagen, dass BMW auch bei Ihnen engagiert ist. Gibt es da keine Konkurrenzkonflikte?

RM: Nein, weil GENIVI in der Hauptsache einen ganz anderen Fokus als die Gruppen, bei denen BMW bei uns mitarbeitet, hat. Bei GENIVI geht es ganz gezielt um die Entwicklung von IVIs während es bei uns in der Automobil Working Group darum geht Tools für Embedded Development zu entwickeln oder zu verbessern und in der Open MDM Gruppe um Messdatenerfassungssysteme. Das sind komplett verschiedene Bereiche innerhalb von BMW.

RW: Wo sehen Sie Vor- und Nachteile einer unter einer Dachorganisation eingegliederten Community wie den Working Groups unter der Eclipse Foundation und einer unabhängigen Organisation wie GENIVI?

RM: Ein Nachteil bei unabhängigen Organisation, was wir jetzt auch ein paar Mal erlebt haben, ist, dass sie sehr viel Zeit verlieren das zu tun oder aufzubauen, was wir schon können. Die haben unheimliche Kosten, da sie sich auf gemeinsame Bylaws einigen müssen. Stellen Sie sich vor, fünf grosse Firmen müssen sich zusammen Bylaws erarbeiten. D.h. Sie können davon ausgehen, dass dies ein dreijähriger Prozess ist, bis sich die Anwälte dieser verschiedenen Firmen geeinigt haben. Bei uns ist es so, dass die das vorgelegt bekommen und nichts gross daran ändern können. Die Zeit die dort von der ersten Diskussion bis zur Gründung vergeht, ist im besten Fall sechs Wochen. Die Kosten, die anfallen, um die initialen Diskussionen zu halten und die Anwälte untereinander können auf null gehen. Das ist sicherlich ein Vorteil. Der andere Vorteil ist und das haben wir bei dieser openETCS Gruppe, die ich jetzt doch namentlich erwähne, gesehen. Die haben im Prinzip unsere Bylaws kopiert und ein bisschen angepasst. Sie hatten und sie haben immer noch, enorme Schwierigkeiten die Services auf die Reihe zu kriegen, weil man das nicht einfach so von heute auf morgen machen kann. Da haben wir 14 Jahre Erfahrung und sind da ganz gut aufgestellt.

RW: Wie schätzen Sie die Relevanz der Eclipse Foundation als Teil der Open Source Entwicklung und der kollaborativen Softwareentwicklung ein?

RM: Relevanz verglichen mit wem?

RW: Zu Linux zum Beispiel?

RM: Ganz klar ist, dass die Linux Foundation eine erheblich finanziell besser ausgestattete Organisation ist als wir, die auf einem ganz anderen Level mit einem ganz anderen Verbreitungsgrad operiert. Die Linux Technologie zieht sich durch viel mehr Bereiche durch. Es ist eine Betriebssystem Plattform mit allen möglichen anderen Sachen dabei, wie in der Automatisierungstechnik, in der Autoindustrie, in der Flugzeugindustrie usw. Wenn sie dort alleine den Mitgliederstamm sehen, sehen Sie dass die weit grösser sind als wir und weit potenter. Im Bereich Tools sind wir wahrscheinlich mit Abstand die einflussreichste Foundation und wir sehen, dass wir uns auch in anderen Technologiebereichen vergrössern. Ein Beispiel ist das Thema Internet der Dinge, wo sich mittlerweile grosse und einflussreiche Firmen wie Bosch, die deutsche Telekom, IBM usw. aufstellen, um bei uns zu arbeiten. Wir haben schon eine gewisse Relevanz und wir haben uns wider Erwarten in den letzten Jahren auch in der Öffentlichkeit nicht nur gehalten sondern sehen einen stetigen Technologie- und Mitgliederzuwachs. Wir sind nicht die grösste und einflussreichste Foundation. Wenn sie sich in der Welt so umhören, sehen Sie da gibt es Linux, Mozilla und Apache. Das sind so die Grossen, die alle kennen. Eclipse hat einen erheblich niedrigeren Bekanntheitsgrad. Ich glaube aber, dass wir auch auf dem Wachstumspfad sind und dass wir das durch unsere Leistung machen, während andere Communities wie zum Beispiel OW2, die auch mal eine grosse war, eher aussterben. Da findet sicherlich eine Marktbereinigung statt, bei der die Guten ins Töpfchen und die Schlechten ins Kröpfchen wandern. Da gehören wir schon zu denen, die noch ein gutes Wachstumspotential haben und sicherlich noch bis zu meiner Rente überleben werden.

RW: Das heisst Sie haben das Gefühl, dass die Services, die die Eclipse Foundation anbietet auch noch Unternehmen animiert, mitzumachen?

RM: Ich habe aktuell Diskussionen mit zwei weiteren Organisationen oder Industriekonsortien, die sich gerne bei uns gründen würden. Meine Kollegen arbeiten sehr viel an bspw. Location basierten Services. Das ist ein Wachstumsfeld, das bei uns stattfindet. Das zweite Wachstumsfeld ist das Internet der Dinge. Da sind wir auch sehr gut aufgestellt und da haben wir auch einen sehr grossen Mitglieder- und Technologiezuwachs.

RW: Kommen die Industriekonsortien eher zu Ihnen oder sprechen Sie diese an?

RM: Es ist eine Mischung aus beidem. Auf der einen Seite machen wir natürlich aktiv Akquise und erzählen über uns. Das gehört auch zu meiner Rolle. Wir stellen auch diese Methodik vor. Auf der anderen Seite haben wir verschiedenste Organisationen, die mit uns zusammenarbeiten. Zum Beispiel die Open Source Business Foundation in Deutschland, die so eine Art Vermittlerrolle spielt und sagt, du wenn du da ein Problem hast, solltest du mal mit dem Ralph Mueller von der Eclipse Foundation reden. Das findet in einem grösseren internationalen Netzwerk statt.

RW: Was sind die essentiellen Punkte, die Gründer einer neuen Collaborative Open Source Development Organisation berücksichtigen sollten? Was sind Ihre Empfehlungen zur Gründung einer erfolgreichen Open Source Plattform? Sollte man unbedingt zur Eclipse Foundation kommen?

RM: Nicht unbedingt. Fragen, die sich bspw. stellen, sind: Wie willst du dein Ökosystem haben? Wie können Unternehmen dort nachhaltig Geld verdienen? Willst du ein GPL ähnliches Modell machen, also das rein auf Services basiert oder willst du ein Geschäft machen, das wirklich wie ein Ökosystem aussieht, indem du alle möglichen Nischen zulässt? Wie offen bist du da? Das ist ganz ein wichtiges Thema. Wir stellen uns vor, dass ein Ökosystem alle möglichen Services, Dienstleistungen, Produkte usw. auf der Plattform zulassen muss und soll. Wenn es da Probleme gibt, wenn man das sehr stark regulieren will, sind wir sicher nicht die Richtigen. Das wäre z.B. ein wichtiges Thema. Wenn dort nur etwas für die Bahnindustrie gemacht wird oder werden soll und man das auf Benutzer in der Bahnindustrie einschränken will, dann können und wollen wir leider nicht mitspielen. Das andere Thema ist die Nachhaltigkeit. Wie nachhaltig ist denn das? Ist das einfach mal eine Lösung, die man innerhalb von einem Jahr macht und dann vergisst oder soll das längerfristig betrieben werden und auch längerfristig weiterwachsen. Das ist für uns auch ein zentrales Thema. Was ich noch erwähnen will. Wir stellen unsere Services mittlerweile nicht nur solchen Konsortien zur Verfügung sondern versuchen auch immer stärker Dienstleistung im Bereich Forschungsprojekte bereit zu stellen. Sie kennen sicherlich das ganze Thema mit Horizon2020 oder ähnliche Grossforschungsprojekte, in denen viele Partner zusammenarbeiten. Da versuchen wir uns auch aufzustellen, indem wir versuchen diese Forschungsergebnisse nachhaltig zu machen und dafür zu sorgen, dass Forschungsergebnisse bzw. Codes, die aus der Forschung kommen langfristig als Open Source verfügbar bleibt.

Interview Claus-Peter Wiedemann (C-PW), Lead License Review Team GENIVI Alliance, on November 21st, 2014

RW: Wie sieht ihr Bildungsweg aus?

C-PW: Ich bin Informatiker, habe Informatik studiert und bin dort auch seit mehr als 24 Jahren im Beruf im Bereich Softwareentwicklung unterwegs in verschiedenen Positionen bei verschiedenen Firmen, z.B. bei Siemens und zuletzt seit 2009 bei BearingPoint in der Rolle Senior Manager für Open Source Software Management.

RW: Sind sie erst durch BearingPoint zu Open Source gekommen oder wie genau hat sich das entwickelt?

C-PW: Das ist eigentlich schon in der Historie. Ich habe lange Jahre im Beruf Open Source verwendet und auch selber Open Source entwickelt. Von daher gibt es eine längere Historie zu dem Thema.

RW: Welches sind Ihre Aufgaben, Verantwortlichkeiten und Kompetenzen als Lead des License Review Teams?

C-PW: Das License Review Team kümmert sich im Wesentlichen um das Thema License Compliance bei GENIVI, um sicherzustellen, dass die GENIVI Softwarebestandteile auch den Lizenzbedingungen genügen. Damit man genau weiss, wenn andere Teile, die nicht von GENIVI sind, dort eingebaut werden. Dass, das alles zusammen passt und dass die Lizenzbestimmungen von GENIVI eingehalten werden wenn die Plattform angeboten und weitergegeben wird. Das ist die grobe Aufgabe. Das Team ist verantwortlich für die Prozesse und Policies, die man dazu braucht, macht operativ Codeanalysen und steht für Fragen von Mitgliedern, die sich im Rahmen der GENIVI Tätigkeit ergeben, zur Verfügung.

RW: Aus wie vielen Leuten besteht das License Review Team und welchen beruflichen Hintergrund bringen diese mit (juristisch, technisch, betriebswirtschaftlich)?

C-PW: Das ist ein guter Mix, dessen Anzahl variiert. Es ist eine relativ offene Gemeinschaft, die wir da haben. Es sind fünf bis zehn Kernmitglieder plus weitere GENIVI-Mitglieder, die sich fallweise mit dazu schalten. Je nachdem welche Themen behandelt werden. Beruflich ist es gemischt. Hauptsächlich sind es Techniker. Wir haben aber auch solche mit juristischem Hintergrund mit dabei.

RW: Sie sind einerseits verantwortlich für das License Review Team bei GENIVI und vertreten deren Interessen, andererseits sind Sie Senior Manager bei BearingPoint. Gibt es da Konfliktpotential oder lösen sich allfällige Konflikte durch den gemeinsamen Fokus auf?

C-PW: Konflikte gibt es keine, weil wir als BearingPoint neutral sind. Wir haben kein kommerzielles Interesse an GENIVI Produkten oder ähnliches. Wir sind eine Beratungsfirma. Wir arbeiten sowohl für GENIVI wie auch für GENIVI Mitglieder. Das ist gut trennbar, weil das Team bestimmt. Ich habe keine Alleinherrschermacht. Es wird durchaus immer im Team entschieden und es gibt auch noch das Board of Directors der GENIVI, das für substantielle Entscheidungen noch darüber steht. Von daher gibt es da keinen Interessenskonflikt.

RW: Mr. Crumb hat mir von der sogenannten Coopetition erzählt. Wie nehmen Sie diese wahr?

C-PW: Interessenskonflikte im Sinne von GENIVI und ihrem eigenem Business?

RW: Nein, eher unter den verschiedenen Firmen, die involviert sind.

C-PW: GENIVI besteht aus circa 170 Mitgliedsfirmen, welche zum grossen Teil aus dem Automotive Space kommen und Wettbewerber im Tagesgeschäft sind. GENIVI bietet aber die Möglichkeit auf einer technischen Ebene als Plattform für diese Firmen zusammenzuarbeiten. Konzentriert auf die technischen Problemstellungen, die alle lösen müssen. So dass GENIVI sich um nicht-differenzierende Bestandteile kümmert, die in einem System drinnen sind, die jeder braucht, aber die nicht wettbewerbsdifferenzierend sind. Der Kunde bspw. wählt ein System nicht aus, weil eine Basiskomponente gut ist, sondern aus anderen Gründen. Von daher, klar jeder hat seine eigenen Interessen, es sind immer noch Firmen, aber es funktioniert sehr gut im GENIVI Umfeld habe ich den Eindruck.

RW: Dann ist es eher hilfreich als hindernd, wenn so viele Leute mit verschiedenen Erfahrungs-, und Wissenshintergründen aufeinander treffen?

C-PW: Ja, sicher. Wenn es von einigen wenigen getrieben würde, wäre das auch nicht akzeptiert in der Breite. So ergibt sich eine breite Basis und es ist durchaus ein Vorteil, dass man auch verschiedene Strömungen im Spiel hat.

RW: Was sind Ihrer Meinung nach die wichtigsten Rollen in einer Open Source Foundation wie der GENIVI Alliance?

C-PW: Es gibt ja immer die Organisation selbst und die Entwicklung, die dahinter steht. Das Wichtigste ist, dass sich die Community da drum rum bildet. Die Entwicklung und die Anforderungen weitertreibt und es zum Erfolg bringt. Von daher muss die Organisation diese Community fördern und dieser Community eine möglichst angenehme Arbeitsatmosphäre geben. Alles tun, damit die Zusammenarbeit dieser Community, die ja nicht nur aus Mitgliedern besteht sondern auch aus anderen Entwickler aus anderen Domänen. Von daher ist die Rolle des Community Managers in meinen Augen sehr wichtig. Da haben Sie ja mit dem Jeremiah Foster schon gesprochen. Das ist die wichtigste Rolle im operativen Bereich würde ich sagen, damit die Community läuft. Weiter das Strategische mit dem Board of Directors, das die strategische Stossrichtung der Alliance definiert und die Alliance selbst weiterbringt. Executive Director Steve Crumb, der das ganze Tagesgeschäft managet. Das ist auch eine sehr zentrale Rolle, damit alles gut funktioniert. Es gibt natürlich die Expert Groups, die inhaltlich die Themen treiben. Die Rollen sind eigentlich alle sehr wichtig und solch eine Organisation ist immer recht schlank aufgestellt. Da gibt es keine Rollen, die nicht wirklich benötigt werden, die nicht wirklich zum Erfolg beitragen. Es ist nicht so wie in einem grossen Unternehmen in denen es durchaus mal Schichten gibt, die vielleicht nicht unbedingt eine Bedeutung für den Geschäftserfolg haben, aber das hat man in der Regel in so einer Alliance nicht.

RW: Welches sind die wichtigsten Rechtsdokumente der GENIVI Alliance (Bylaws, Intellecutal Property Rights Policy, License Policy) und weshalb sind diese relevant? Welches sind die wichtigsten Grundaussagen dieser Dokumente?

C-PW: Die Bylaws sind die grundsätzlichen Regeln der Zusammenarbeit. Dazu gehört dann direkt die IP-Policiy, die Policy, die erklärt was mit den Intellectual Property passiert, welche im Rahmen der Alliance entsteht oder von den Mitgliedern eingebracht wird. Das sind die grundsätzlichen Regeln der Zusammenarbeit. Die sind auch öffentlich zugänglich auf der GENIVI Website. Das ist die Basis. Daneben gibt es noch die GENIVI License Policy, die festlegt welche Open Source Lizenzen bei GENIVI akzeptabel sind und welche Regeln es gibt für zum Beispiel neuen Code. Dies ist auch öffentlich und einsehbar.

RW: Auf welcher Basis wurden diese Dokumente erstellt? Gab es Vorlagen bereits bestehender ähnlicher Organisationen, welche benutzt werden konnten oder wurden diese von Grund auf neu erarbeitet?

C-PW: Ich kann nichts über die Bylaws und die IP-Policies sagen. Grundsätzlich entstehen diese Dokumente gemeinschaftlich. So wie die License Policy, die mit verschiedensten Review-Zyklen im Rahmen des License Review Teams entstanden ist. Wir haben das von Grund auf neu gemacht. Natürlich mit ein paar Vorlagen im Gedächtnis aber wir haben nichts genommen, das wir direkt angepasst haben.

RW: Ist es schon vorgekommen, dass Firmen aus der Alliance verwiesen wurden, weil sie in irgendeiner Form gegen das Regelwerk verstossen haben?

C-PW: Das ist mir nicht bekannt, dass es irgendwelche Ausschlüsse gegeben hat.

RW: GENIVI bietet 2 verschiedene Projektgruppen an. Open Source Projekte (für alle) und das Zertifizierungsprogramm mit GENIVI Compliant™ und Works with GENIVI™ (nur für Mitglieder). Was war die Ursache dieser Unterscheidung und wo liegen die essentiellen Unterschiede dieser zwei Projektarten?

C-PW: Die Unterscheidung ist eigentlich ganz natürlich, weil es drei verschiedene Dinge sind, um die es hier geht. Einmal die Open Source Projekte. Ganz klar dort entsteht der Kern von GENIVI, die GENIVI Plattform in einer kollaborativen Art und Weise wie bei allen Open Source Projekten. Bei dem GENIVI Compliance Program, das ist also nicht License Compliance, sondern auf der technischen Schiene. Da wird festgelegt, was ein GENIVI Compliance System für Anforderungen erfüllen muss. Im Sinne von welche Komponenten da vorhanden sein müssen und welche Schnittstellen dort sein müssen. Das ist eigentlich die Grundlage dafür und da gibt es die Möglichkeit für GENIVI Mitglieder eigene Produkte zertifizieren zu lassen, damit diese Produkte das Label GENIVI Compliant bekommen. Das ist etwas ganz wichtiges, weil es auch nach aussen hin eine Anforderung erfüllt, die vielfach auch von den Herstellern gefordert wird. Das ist das technische Compliance Programm. Das Work with GENIVI ist wieder ein bisschen anders. Das ist auch Art der Zertifizierung oder eine Art der Bestätigung, dass ein bestimmtes Produkt oder eine Komponente mit GENIVI Compliant Systemen zusammenarbeiten kann. Das ist quasi noch eine andere Dimension. Die haben alle ihre Berechtigung. Der Unterschied ist, dass bei den Open Source Projekten alle mitmachen können. Da gibt es keinerlei Beschränkungen. Das GENIVI Compliant und das Work with GENIVI muss man ganz klar trennen. Das GENIVI Compliance Programm ist für Mitglieder, das heisst um GENIVI Compliant Produkte anbieten zu können, müssen sie Mitglied sein. Das macht auch Sinn. Work with GENIVI kann jeder beantragen, muss nicht unbedingt GENIVI Mitglied sein. Da geht es darum, dass irgendein anderes Produkt auf technischer Ebene mit GENIVI zusammenarbeiten kann. Das ist durchaus sinnvoll und hat sich bis anhin auch so bewährt.

RW: Worin genau besteht der Unterschied zwischen den Workgroups und den Collaborative Projects der Linux Foundation?

C-PW: Diese Collaborative Projects sind meines Erachtens der Überbau und die Workgroups machen bestimmte Teile, die sie vertiefen. Ich bin da auch nicht so informiert bzgl. der Organisation der Linux Foundation. Ich bin im Automotive Grade Linux dabei. Das ist ein Projekt innerhalb der Linux Foundation und da gibt es dann wieder eine Gliederung in

Expert Groups usw. Das ist eine typische technische Gliederung innerhalb der Projekte. Aber was genau der Unterschied zwischen diesen beiden ist, kann ich nicht sagen.

RW: Sie arbeiten bei der LINUX Foundation für die Workgroup „Automotive Grade Linux". Steht diese in Konkurrenz zu GENIVI?

C-PW: Konkurrenz kann man in dem Sinne nicht sagen. Das sind zwei Ansätze in Richtung Linux basierte Systeme, wobei Automotive Grade Linux den Ansatz hat, eine komplette Distribution zu bauen, also ein komplettes Linux, das im Auto eingesetzt werden kann im Infotainment Bereich und auch in anderen Bereichen im Auto. GENIVI verfolgt da ein bisschen einen anderen Ansatz. GENIVI konzentriert sich auf Middleware, das heisst GENIVI arbeitet mit verschiedenen Linux Distributionen und ist davon mehr oder weniger unabhängig. GENIVI konzentriert sich auf diese Middleware, die Automobil spezifisch und Infotainment spezifisch ist. Deswegen gibt es da auch einen engen Dialog zwischen den beiden Organisationen, weil es natürlich Überschneidungen gibt in diesem Middelware-Bereich. Da versucht man die Dinge nicht zweimal zu erfinden. Aber es sind durchaus zwei getrennte Projekte mit zwei ähnlichen aber durchaus unterschiedlichen Ansätzen. Aber es ist nicht so, dass man nicht miteinander spricht.

RW: Wäre es rein hypothetisch möglich die GENIVI Alliance an die Linux Foundation anzugliedern?

C-PW: Möglich ist alles. Ich würde sagen, GENIVI ist eine eigenständige Organisation und hat auch den Brandname schon etabliert und das kann in alle Richtungen gehen. Solche Organisationen verändern sich, Strategien verändern sich, aber im Moment ist da nichts sichtbar. GENIVI hat eine Beziehung zu der Linux Foundation, weil die ganzen Projekte über die Linux Foundation gehostet sind. Da gibt es durchaus eine Nähe, aber es ist kein Linux Foundation Projekt.

RW: Es gibt noch die Eclipse Foundation. Sehen Sie da auch eine Konkurrenz zur Linux Foundation?

C-PW: Das ist in meinen Augen ein unterschiedlicher Fokus. Die Eclipse Foundation ist sozusagen die Eclipse Umgebung um die es da geht und die hat auch eine andere Zielsetzung. Sie ist sehr erfolgreich, gross und bekannt, aber die sind wesentlich breiter aufgestellt vom Abdeckungsgrad. Die sind nicht branchenspezifisch, sondern einfach im Sinne von Entwicklungsumgebung und Entwicklungsbibliotheken und –werkzeugen usw. Also unabhängig von der Domäne. Es ist eine Organisation, die sehr erfolgreich und gross ist, wo es auch Bindungspunkte gibt, aber die steht jetzt auch nicht im direkten Wettbewerb.

RW: Wie schätzen Sie die Relevanz der GENIVI Alliance als Teil der Open Source Entwicklung ein? Denken Sie, dass sich nebst der Automobil-Industrie auch andere Branchen zusammenschliessen sollten? Kann GENIVI als einer der Vorreiter und als glänzendes Beispiel für andere Branchen verstanden werden?

C-PW: Es kommt auf den Scope an. Wenn man sagt, insgesamt in Open Source. Da gibt es natürlich viele. Da gibt es Apache, Eclipse, Linux,.. und viele hunderte von Grundprojekten, die in Alliances oder auch ohne Alliances laufen. So ist GENIVI eine unter vielen. Im Automotive Umfeld ist GENIVI aber ein Pionier und trägt meines Erachtens sehr viel dazu bei, dass das Thema Open Source und auch die Art der Softwareentwicklung im Automotive Umfeld sich radikal ändert und sich in Richtung Offenheit und Zusammenarbeit entwickelt. Da leistet GENIVI einen sehr grossen Beitrag und auch Pionierarbeit. Das merkt man, weil die Automobilindustrie eine traditionelle Industrie ist. Historische Strukturen, bestimmte Arbeitsweisen, Open Source und diese ganze Offenheit und diese Zusammenarbeit war bis vor ein paar Jahren, bevor GENIVI da war, noch nicht so verbreitet. Von daher ist es durchaus eine Pionierarbeit, die in meinen Augen sehr erfolgreich ist.

RW: Was sind die essentiellen Punkte, die Gründer einer neuen Collaborative Open Source Development Plattform berücksichtigen sollten? Was sind Ihre Empfehlungen zur Gründung einer erfolgreichen Open Source Plattform?

Das ist eine gute Frage. Eine Alliance lebt eigentlich von der Community, wenn es im Open Source Bereich ist. Sich einfach zusammenzuschliessen und zu sagen, wir machen jetzt eine Alliance und entwickeln was zusammen, macht wenig Sinn. Es kommt sehr auf das Thema drauf an. Wenn man überlegt wie Android entstanden ist. Das war die Open Handset Alliance Datei mit dem Ziel eine Open Source Plattform für Mobilgeräte zu machen. Es gibt eine Alliance im Internet of Things Bereich, die neu ist. Es muss immer das Thema passen und es muss ein gemeinsames Interesse von Firmen da sein, die im täglichen Geschäft im Wettbewerb stehen, zusammenzuarbeiten und sich zusammenzuschliessen. Wenn man das macht, ist es wichtig die Alliance so aufzustellen, dass eine offene Kommunikation und eine offene Zusammenarbeit herrscht und sich so eine Community entwickeln kann, die das Ganze dann treibt. Das sind die wichtigsten Eckpunkte.

List of Figures

List of Tables

List of Abbreviations

CH = Switzerland

EF = Eclipse Foundation

EPL = Eclipse Public License

GA = GENIVI Alliance

GPL = General Public License

IP = Intellectual Property

IT = Information Technology

IVI = In-vehicle Infotainment

OSI = Open Source Initiative

OSS = Open Source Software

USA = United States of America

USD = United States Dollar

WG = Working Group

References

BAKAR, A., SHEIKH, Y., & SULTAN, B. (2012). Opportunities and challenges of open source software integration in developing countries: Case of Zanzibar health sector. *Journal of Health Informatics in Developing Countries*, *6*(2).

Baldwin, C., & Von Hippel, E. (2011). Modeling a paradigm shift: From producer innovation to user and open collaborative innovation. *Organization Science*, *22*(6), 1399-1417.

Beal, V. (2014), Webopedia, URL: http://www.webopedia.com/TERM/P/platform.html [Accessed: 2014-12-02].

Benkler, Y. (2002). Coase's Penguin, or, Linux and" The Nature of the Firm". *Yale Law Journal*, 369-446.

Betschon, S. (2014), NZZ Online, Der Siegeszug der Smartphones, Und plötzlich füllte sich der Bildschirm, URL: http://www.nzz.ch/wirtschaft/wirtschafts-und-finanzportal/und-ploetzlich-fuellte-sich-der-bildschirm-1.18259719 [Created: 2014-03-10; Accessed: 2014-11-08].

Blackduck, Future Open Source Survey (2014), URL: https://www.blackducksoftware.com/news/releases/2014-future-open-source-survey-results-revealed [Accessed: 2014-12-30].

Blackduck, Top 20 Open Source Licenses, URL: https://www.blackducksoftware.com/resources/data/top-20-open-source-licenses [Accessed: 2014-12-01].

Bonaccorsi, A., & Rossi, C. (2003). Why open source software can succeed. *Research policy*, *32*(7), 1243-1258.

Bosch, J. (2009, August). From software product lines to software ecosystems. In *Proceedings of the 13th International Software Product Line Conference* (pp. 111-119). Carnegie Mellon University.

Cambridge Dictionaries Online, Community (2014), URL: http://dictionary.cambridge.org/de/worterbuch/british/community?topic=ordinary-people [Accessed: 2014-11-09].

Charmer, M. (2015). 2015: The year the UN resets the count, URL: http://opensource.com/life/15/1/2015-year-un-resets-count [Created: 2015-01-07; Accessed: 2015-01-10].

Chesbrough, H., Vanhaverbeke, W., & West, J. (Eds.). (2006). *Open innovation: Researching a new paradigm*. *Chapter 1, Open Innovation: A New Paradigm for Understanding Industrial Innovation.* Oxford university press.

Crumb, S. (2014, October). GENIVI Alliance (Executive Director), Interview.

Data Revolution Group, A World That Counts (2014), URL: http://www.undatarevolution.org [Accessed: 2015-01-10].

Dedrick, J., & West, J. (2003, December). Why firms adopt open source platforms: a grounded theory of innovation and standards adoption. In *Proceedings of the workshop on standard making: A critical research frontier for information systems* (pp. 236-257). Seattle, WA.

Eclipse, About Us (2014), URL: http://www.eclipse.org/org/ [Accessed: 2014-11-04].

Eclipse, Eclipse Working Groups (2014), URL: http://eclipse.org/org/workinggroups/about.php#wg-neutral [Accessed: 2014-11-04].

Eclipse Foundation, Inc., Bylaws, (2014), URL: https://eclipse.org/org/documents/Eclipse%20BYLAWS%202003_11_10%20Final.pdf [Accessed: 2014-11-04].

Eclipse Foundation, Inc., Intellectual Property Policy, (2011), https://eclipse.org/org/documents/Eclipse_IP_Policy.pdf [Accessed: 2014-11-04].

Eclipse Foundation, Inc., Membership Agreement (2010), URL: https://eclipse.org/org/documents/Eclipse%20MEMBERSHIP%20AGMT%202010_01_05%20Final.pdf [Accessed: 2014-11-09].

Eclipse, Membership Rights (2014), URL: http://eclipse.org/membership/become_a_member/memberRights.php [Accessed: 2014-11-04].

Eclipse, Polarsys Working Group Charter (2012), URL: http://www.eclipse.org/org/workinggroups/polarsys_charter.php [Accessed: 2014-10-31].

Eclipse, Public License, FAQ (2014), URL: https://eclipse.org/legal/eplfaq.php [Accessed: 2014-11-13].

Eclipse, Public License – V 1.0 (2014), URL: https://eclipse.org/org/documents/epl-v10.php [Accessed: 2014-11-13].

Eclipse, Types of Membership (2014), URL: https://eclipse.org/membership/become_a_member/membershipTypes.php [Accessed: 2014-11-04].

Economides, N., & Katsamakas, E. (2006). Two-sided competition of proprietary vs. open source technology platforms and the implications for the software industry. *Management Science*, *52*(7), 1057-1071.

Enkel, E., Gassmann, O., & Chesbrough, H. (2009). Open R&D and open innovation: exploring the phenomenon. *R&d Management*, *39*(4), 311-316.

Fitzgerald, B. (2006). The transformation of open source software. *Mis Quarterly*, 587-598.

Foster, J. (2014, November). GENIVI Alliance (Community Manager), Interview.

Gartner (2014), Hype Cycle Special Report, URL: http://www.gartner.com/technology/research/hype-cycles/ [Accessed: 2014-12-21].

GENIVI, About GENIVI (2014), URL: http://www.genivi.org/about-genivi [Accessed: 2014-09-23].

GENIVI, Bylaws (2011), URL: http://www.genivi.org/sites/default/files/GENIVI_Bylaws_Amended_10October2011.pdf [Accessed: 2014-10-23].

GENIVI, Compliance Programs (2014), URL: http://genivi.org/genivi-compliance-program [Accessed: 2014-09-23].

GENIVI, FAQ (2014), URL: http://www.genivi.org/faq [Accessed: 2014-09-23].

GENIVI, Functional Organization Chart (2014), URL: http://www.genivi.org/functional-organization-chart [Accessed: 2014-09-23].

GENIVI, Instructions for Obtaining Membership (2012), URL: http://www.genivi.org/sites/default/files/GENIVI_Instruction_for_Obtaining_Membership_092012.pdf [Accessed: 2014-11-21].

GENIVI, Intellectual Property Rights Policy (2012), URL: http://www.genivi.org/sites/default/files/GENIVI_IPR_Policy_Amended_8Feb2012%5B1%5D.pdf

GENIVI, Join the GENIVI Alliance (2014), URL: http://genivi.org/join [Accessed: 2014-09-23].

GENIVI, Members (2014), URL: http://www.genivi.org/genivi-members [Accessed: 2014-09-23].

GENIVI, Membership Value (2014), URL: http://www.genivi.org/current-genivi-members [Accessed: 2014-11-12].

GENIVI, Open Source Projects (2014), URL: http://projects.genivi.org [Accessed: 2014-09-23].

GENIVI, Participation Agreement (2014), URL: http://www.genivi.org/sites/default/files/Genivi_Participant%20Agreement_Revised15Dec2014_2.pdf [Accessed: 2014-12-01].

GENIVI, Public Policy for GENIVI Licensing and Copyright Version 1.5 (2014), URL: http://docs.projects.genivi.org/License/Public_Policy_for_GENIVI_Licensing_and_Copyright_v1.5.pdf [Accessed: 2014-12-01].

GENIVI, Why Become a GENIVI Member (2014), URL: http://www.genivi.org/why-become-genivi-member [Accessed: 2014-10-20].

GNU Operating System, About GNU (2014), URL: https://www.gnu.org/gnu/gnu.html [Accessed: 2014-11-30].

GNU Operating System, Licenses (2014), URL: https://www.gnu.org/licenses/licenses.html [Accessed: 2014-11-30].

GNU Operating System, What is copyleft? (2014), URL: https://www.gnu.org/copyleft/ [Accessed: 2014-11-30].

Harvey, C. (2014), Datamation, 35 Open Source Tools for the Internet of Things, URL: http://www.datamation.com/open-source/35-open-source-tools-for-the-internet-of-things-1.html [Created: 2014-08-21; Accessed: 2014-12-21].

Hippel, E. V., & Krogh, G. V. (2003). Open source software and the “private-collective” innovation model: Issues for organization science. *Organization science*, *14*(2), 209-223.

Kilamo, T., Hammouda, I., Mikkonen, T., & Aaltonen, T. (2012). From proprietary to open source—Growing an open source ecosystem. *Journal of Systems and Software*, *85*(7), 1467-1478.

Kogut, B., & Metiu, A. (2001). Open‐ source software development and distributed innovation. *Oxford Review of Economic Policy*, *17*(2), 248-264.

Kuali, About the Kuali Foundation (2013), URL: http://www.kuali.org/about [Accessed: 2014-10-21].

Laurent, A. M. S. (2004). *Understanding open source and free software licensing*. " O'Reilly Media, Inc.".

Lerner, J., & Tirole, J. (2001). The open source movement: Key research questions. *European Economic Review*, *45*(4), 819-826.

Lerner, J., & Tirole, J. (2002). Some simple economics of open source. *The journal of industrial economics*, *50*(2), 197-234.

Linux Foundation, About Us (2014), URL: http://www.linuxfoundation.org/about [Accessed: 2014-11-01].

Linux Foundation (2014), Collaborative Development Trends Report

Liu, M., Hansen, S., & Tu, Q. (2014). The community source approach to software development and the Kuali experience. *Communications of the ACM*, *57*(5), 88-96.

Machbarkeitsstudie (2013). Konsortiale Softwareentwicklung auf der Basis von Open-Source-Software

Milinkovich, M. (2010), Wordpress, EPL / GPL Commentary, URL: http://mmilinkov.wordpress.com/2010/04/06/epl-gpl-commentary/ [Created: 2010-04-06; Accessed: 2014-11-26].

Moodle (2014), URL: https://moodle.org [Accessed: 2015-01-11].

Mueller, R. (2014, November). Managing Director Eclipse Foundation Europe GmbH, Interview.

Nakakoji, K., Yamamoto, Y., Nishinaka, Y., Kishida, K., & Ye, Y. (2002, May). Evolution patterns of open-source software systems and communities. In *Proceedings of the international workshop on Principles of software evolution* (pp. 76-85). ACM.

onegovbox (2014), URL: http://onegov.ch [Accessed: 2014-10-21].

onegovbox, OneGov Box (2014), URL: http://onegov.ch/onegov-box [Accessed: 2014-10-21].

onegovbox, OneGov GEVER (2014), URL: http://onegov.ch/gever [Accessed: 2014-10-21].

Open Source Initiative (2014), URL: http://opensource.org [Accessed: 2014-09-21].

Polarsys, About Us (2014), URL: http://www.polarsys.org/about-us [Accessed: 2014-10-21].

Polarsys Industry Working Group Participation Agreement (2012), URL: https://eclipse.org/org/foundation/boardminutes/2012_02_exhibits/ExhibitE.pdf [Accessed: 2014-11-04].

Polarsys Members (2014), URL: https://polarsys.org/members [Accessed: 2014-10-21].

Polarsys/TLPProposal (2012), URL: https://wiki.eclipse.org/Polarsys/TLPProposal [Accessed: 2014-12-12].

Raymond, E. S. (2001). *The Cathedral & the Bazaar: Musings on linux and open source by an accidental revolutionary*. " O'Reilly Media, Inc.".

Riehle, D. (2007). The economic motivation of open source software: Stakeholder perspectives. *Computer*, *40*(4), 25-32.

Riehle, D. (2010). The economic case for Open Source foundations. *Computer*, *43*(1), 0086-90.

Riehle, D. (2012). The single-vendor commercial open course business model. *Information Systems and e-Business Management*, *10*(1), 5-17.

Rouse, M. (2014), Internet of Things (IoT), Part of the Cloud computing glossary*,* URL: *http://whatis.techtarget.com/definition/Internet-of-Things)* [Accessed: 2015-01-05].

Sarokin, D., Demand Media, Difference Between Tier 1 & Tier 2 Companies, URL: http://smallbusiness.chron.com/difference-between-tier-1-tier-2-companies-25430.html [Accessed: 2015-01-05].

Schweizerische Eidgenossenschaft, Bundesgesetz über das Urheberrecht und verwandte Schutzrechte, Urheberrechtsgesetz (URG) (2011), URL: http://www.admin.ch/opc/de/classified-compilation/19920251/ [Accessed: 2014-12-02].

Schweizerisches Bundesgericht, OpenJustitia, URL: http://www.openjustitia.ch/DE/interne_Open_Justitia.html [Accessed: 2014-10-21].

Sharma, S., Sugumaran, V., & Rajagopalan, B. (2002). A framework for creating hybrid-open source software communities. *Information Systems Journal*, *12*(1), 7-25.

Stürmer, M. (2014), Parldigi (Parlamentarische Gruppe Digitale Nachhaltigkeit), URL: http://www.digitale-nachhaltigkeit.ch/2014/11/open-source-an-stadtberner-schulen/ [Created: 2014-11-07; Accessed: 2015-01-11].

United Nations Department of Economic and Social Affairs, Sustainable Development, Knowledge Platform (2012), URL:http://sustainabledevelopment.un.org/?menu=1300

Techopedia, URL: http://www.techopedia.com/definition/3411/platform [Accessed: 2014-12-02].

TechTerms.com, URL: http://www.techterms.com/definition/platform [Accessed: 2014-12-02].

Tiwana, A., Konsynski, B., & Bush, A. A. (2010). Research commentary-Platform evolution: Coevolution of platform architecture, governance, and environmental dynamics. *Information Systems Research*, *21*(4), 675-687.

Überhorst, S. (2009). Computerwoche, Lizenzen: GNU GPL versus EPL und Apache, URL: http://www.computerwoche.de/a/java-gemeinde-setzt-hohe-erwartungen-in-oracle,1897589,4 [Created: 2009-06-03; Accessed: 2014-11-25].

van den Berk, I., Jansen, S., & Luinenburg, L. (2010, August). Software ecosystems: a software ecosystem strategy assessment model. In *Proceedings of the Fourth European Conference on Software Architecture: Companion Volume* (pp. 127-134). ACM.

Webopedia (2014), URL: http://www.webopedia.com/TERM/I/in-vehicle-infotainment-ivi.html [Accessed: 2014-11-25].

West, J., & O'Mahony, S. (2008). The role of participation architecture in growing sponsored open source communities. *Industry and Innovation*, *15*(2), 145-168.

What is Horizon 2020, URL: http://ec.europa.eu/programmes/horizon2020/en/what-horizon-2020) [Accessed: 2015-01-05].

Wiedemann, C.-P. (2014, November). GENIVI Alliance (Lead License Review Team), Interview.

Statement of Autonomous and Independent Work

„Ich erkläre hiermit, dass ich diese Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen benutzt habe. Alle Stellen, die wörtlich oder sinngemäss aus Quellen entnommen wurden, habe ich als solche gekennzeichnet. Mir ist bekannt, dass andernfalls der Senat gemäss Artikel 36 Absatz 1 Buchstabe o des Gesetzes vom 5. September 1996 über die Universität zum Entzug des aufgrund dieser Arbeit verliehenen Titels berechtigt ist.“

Bern, 20.01.2015 Rahel Winkelmann

Publication of the Thesis

Hereby I give permission to make my thesis publically available at the Library of Information Systems.

Bern, 20.01.2015 Rahel Winkelmann