

# Challenges of the Open Source Component Marketplace in the Industry

Claudia Ayala<sup>1,2,\*</sup>, Øyvind Hauge<sup>1</sup>, Reidar Conradi<sup>1</sup>, Xavier Franch<sup>2</sup>,  
Jingyue Li<sup>1</sup> and Ketil Sandanger Velle<sup>1</sup>

1: Norwegian University of Science and Technology (NTNU)  
{ oyvind.hauge, reidar.conradi, jingyue.li, ketilsan }@idi.ntnu.no

2: Technical University of Catalunya (UPC)  
{ cayala, franch }@lsi.upc.edu

**Abstract.** The reuse of Open Source Software components available on the Internet is playing a major role in the development of Component Based Software Systems. Nevertheless, the special nature of the OSS marketplace has taken the “classical” concept of software reuse based on centralized repositories to a completely different arena based on massive reuse over Internet. In this paper we provide an overview of the actual state of the OSS marketplace, and report preliminary findings about how companies interact with this marketplace to reuse OSS components. Such data was gathered from interviews in software companies in Spain and Norway. Based on these results we identify some challenges aimed to improve the industrial reuse of OSS components.

## 1 Introduction

Nowadays, Component-Based Software Development (CBSD) is considered the standard way of developing software systems [3]. The main motivation behind this is reusability as “reusing components avoid reinventing the wheel”. This allows companies a faster technology adoption and innovation whilst reducing costs and time-to-market [18]. In particular, the existence of the Open Source Software (OSS) marketplace, consisting of tens of thousands of OSS components which are developed and freely available over the Internet, has greatly influenced the software reuse practices and the overall economy behind [8, 11].

In this huge diversity, one of the most influential activities in CBSD is the selection of components [7, 26]. Successful reuse of OSS components highly depends on being able to navigate in the OSS marketplace to identify and evaluate which component(s) may (best) fit the requirements [25]. In the industrial practice, the selection of OSS components is considered a highly risky activity as companies are confronted with incredibly large amounts of diverse, partial, and ephemeral,

\* Her work was carried out during the tenure of an ERCIM “Alain Bensoussan” Fellowship Programme.

information about OSS components. This information could be tacit and it is not always trustable. Therefore, it is not easy for companies to fully exploit the benefits of reusing OSS components [18].

To support the industry in this crucial task, we need to understand the industrial needs for selecting components and the state of the OSS marketplace as the place where components are found. To do so, we investigated: (1) the elements that constitute the OSS marketplace, and the current state of such infrastructure from the point of view of industrial reusers. (2) How reusers interact with the OSS marketplace to select OSS components. First, we assessed a variety of Internet resources and identified their main characteristics and utility. Second, we performed semi-structured interviews in companies in Spain and Norway. Based on these investigations we identify challenges aimed to improve the industrial reuse of OSS components.

## 2 State-of-the-Art

Systematic software reuse is an engineering strategy proposed to increase productivity and software quality, and lead to economic benefits [9]. It is based on the premise that for reuse to be effective, a proper infrastructure enabling reusers to find and understand the components that best fit their needs should exist [21]. Although systematic software reuse has been an active research area for more than a decade [9], the special nature of OSS has taken the original concept of systematic reuse based on centralized repositories into a completely different arena based on massive reuse over Internet. Therefore, the need of new approaches for effectively finding and understanding components has been widely recognized [2, 9].

Although research and practitioners have proposed a diverse set of methods and evaluation guidelines for supporting components selection (e.g., [16, 17, 19]), these proposals have not been widely adopted in the industrial practice [15]. The literature presents various possible explanations for this: that the proposed methods are failing to deal with identification of components and information for evaluating them [4, 6], and that it is impractical to perform complete evaluations in terms of time and cost [12].

The Internet as the infrastructure of the OSS marketplace constitute as we see an important part of both identification and evaluation of OSS components [25]. However, we know little about this OSS marketplace and how practitioners search, evaluate and choose components from this marketplace.

In order to envisage effective solutions for enabling successful reuse of OSS components, further empirical studies are needed to better understand how researchers may contribute to developing the marketplace and improving how reusers select OSS components. The work presented in this paper is trying to contribute to this fact.

### 3 Elements of the OSS Marketplace

The OSS reuse environment greatly differs to the “classical” reuse environment based on centralized repositories [21]. In this section, we describe the high-level elements that constitute the OSS marketplace in order to understand the new required needs for improving OSS components reusability.

Although the word marketplace may have different connotations [5, 14, 24], in the context of this paper, as OSS marketplace we refer to the self-organizing virtual place on the Internet that includes the exchange interactions between reusers and providers of OSS components as well as the actions of other actors that facilitate or promote such transactions.

The OSS marketplace implies diverse elements, relationships and interactions over internet: *Providers* offer OSS components through their own websites. *Reusers* use a *search mechanism* or *Intermediary* services to find and select components, whilst *Promoters* foster the OSS movement.

Although it is not our intention to further describe the marketplace interactions, as it implies a broad line of research, we briefly describe its main elements and provide some examples of the actors that actually cover these elements.

- **Reuser:** Refers to developers that search in the marketplace for components that may cover certain functionality. Such component(s) are intended to be integrated in a software system. The work described in this paper focus on the problems reusers face in their OSS selection practices.
- **Provider:** Refers to OSS communities or companies which develop and release OSS components. Currently there are thousands of OSS communities and therefore thousands of potential component providers. Examples are moodle.org, linux.org, eclipse.org, FreMed.org, Openmrs.org etc.
- **Search Mechanism:** Refers to the mechanism that allows navigation through the marketplace. General-purpose tools to navigate through the Internet as *Google* exist. But some specialized tools as *Google Code Search* or *Kooders* have been designed for indexing various open source repositories and to allow more focused component searches on the web.
- **Promoter:** Refers to individuals and organizations which main aim is to foster the OSS movement. Examples are the Open Source Technology Group (OSTG), Free Software Foundation (FSF), Apache Foundation, and personal blogs with useful resources. Practical research efforts from academia and/or industry can be also found, an example is the CeBASE repository that provides a "lessons learned" database.
- **Intermediary:** Refers to profit or non-profit organizations or individuals that index and/or distribute OSS components or other related products and services. Examples are companies selling support around certain components or domains as Forrester or Gartner; and *General-oriented or Domain specific portals* as SourceForge or TheServerSide respectively.

## 4 The Study

The aim of the study is to establish an empirical foundation of the challenges of the OSS marketplace when dealing with the needs of industrial *Reusers*. The study consisted of two parts: a) investigation of the actual state of the marketplace, b) investigation of how *Reusers* interact with the OSS marketplace to select OSS components.

### 4.1 Investigation of the OSS Marketplace

To better understand the state of the marketplace and the kind of resources it offers for supporting components selection, we further assessed more than 60 related sites and search mechanisms on the Internet. The elements of the marketplace introduced in Section 3 were identified throughout this analysis. The studied sites were identified from the answers of the respondents of the study described in Section 4.2, previous studies e.g. [10, 13, 15], research team's experience, and web investigation. Of course, we do not claim that we have reviewed all existing portals or search tools of the marketplace, indeed by the nature of the marketplace it is not realistic to think that this can be done. However, we think that the ones we have assessed are representative of the marketplace elements and their actual offerings. The focus of the assessment was on the factors affecting the selection of OSS components. This set of resources was analyzed between March and November 2008. Main results are briefly summarized in the context of Section 5.

### 4.2 Investigation of Reusers Interaction with the Marketplace

We performed an explorative study in Small and Medium Enterprises (SME). The study consisted of semi-structured interviews with people involved in CBSD using OSS components (i.e., *Reusers*). We asked about how they identify and evaluate OSS components, which resources they use and the problems they face with such resources.

With a basis in earlier studies e.g. [10, 13, 15], we developed and tested an interview guide. It focuses on the experience of industrial *Reusers* with a finished project in which one or more OSS components were used. The interview guide contained one part each about identification and evaluation, the use of internet and social resources in OSS selection, and finally demographic information. The interviews were performed in 5 companies from Spain (ES1-ES5) and 3 from Norway (NO1-NO3). See Table 1 for an overview of the respondents, their companies and the projects. We considered having data from different countries valuable to strengthen the external validity of the results.

**Table 1.** Some details of the study

<b>Id</b>	<b>Company Scope</b>	<b>No. Employees</b>	<b>Project</b>	<b>Experience with CBSD</b>
ES1	Web monitoring SW	10	Web-statistics, Ruby, 1 person/month, part of larger system	2 years, MSc
ES2	HW sales, add-on SW development	150-200	CMS/e-commerce, PHP, 4 person/months	6 years, MSc
ES3	SW development/consultancy	4	Web application, Java, 8 person/months	10 years, MSc
ES4	Organizational IT department	15	Web application, PHP 24 person/month	8 years, MSc
ES5	SW development/consultancy	70-100	e-Business application 10 person/month	11 years, MSc
NO1	SW development/consultancy	20	Web search, Java, 2 person/months, part of larger system	11 years, MSc
NO2	SW development/consultancy	200	Web application, Java, 6 person/years	4 years, MSc
NO3	SW house	12	Linguistic SW, .Net, 10 person/years	7 years, BSc

Each interview took around one hour, and was performed face to face by one or two researcher in the native language of the interviewees (Spanish or Norwegian). To establish the interviews' context and limitations, we began each interview by stating our motivation. We rigorously avoided suggesting any Internet resources during the interviews. Moreover, in cases when the interviewee did not remember the URL or location of a specific resource, they were asked to send us such information by e-mail. The resources mentioned by the interviewees were evaluated as described in Section 4.1. The semi-structured nature of the interview, allows to further inquiry in relevant areas and to get useful qualitative data. Each interview was recorded and transcribed. To perform the data analysis, the research team listened through the recorded interviews or read through the transcriptions. Then, as all authors speak English, we were able to analyze and discuss the obtained data in several meetings.

The following section relates the results obtained from these studies.

## 5 Resources Used to Select OSS Components in the Industry

In this section, we report our assessment of the existing resources in the market-place and how the interviewees used these resources to select OSS components.

Scenarios are used to describe our findings, followed by discussions of some of the problems the *Reusers* face when using resources in the marketplace.

It is important to remark that the objective of the scenarios is to show how *Reusers* used existing resources and not to explain the process they follow in detail. The scenarios may therefore be overlapping.

### 5.1 Searching

The search process departs from the need to find a component that may cover certain functionality in the final system. Further assessment of components is performed in the context of the subsequent evaluation and decision activities.

#### 5.1.1 Existing Resources for Performing OSS Searching

To enable navigation throughout the OSS marketplace, some *Search mechanisms* and a variety of portals issued by *Promoters* and/or *Intermediaries* exist (see Section 3). One of the main goals of these portals is to offer categorizations aimed to guide their users to find information, services and components themselves. From our assessment of several of these portals, we observed that understanding and use of the portal content is not an easy task, especially if the domain is absolutely un-known.

Furthermore, according to their topics, these portals range from general-oriented as SourceForge to domain-specific portals. Domains can be understood at different levels. For instance, domain technologies as presented in TheServerSide which is related to the Java technology or more specific ones as the health care domain as Openmrs.org. On the other hand, the collaborative and “open source” philosophy has also enabled the formation and explosion of open and collaborative portals. These are aimed to discuss and exchange experiences around specific domains. Examples are CMSMatrix and WikiMatrix in the content management system and wiki management system domains respectively.

To describe how interviewees used these resources, we identify three different situations described in terms of the following scenarios.

#### 5.1.2 Scenario S1 – *No Search is Required*

*Reusers* are aware of a component that may fit the functionality they are looking for, or someone (a colleague or member of the project team) has used and recommended the components. As one developers said “*if someone has experience [with a component] we normally select this one*” (ES2). *Reusers* do in other words quite often select components based on previous experience, even without looking for other candidates.

#### 5.1.2.1 Problems Related to Scenario S1

Although no problems with the use of resources were found in the context of this scenario (i.e., it does not imply any search), it highlights the influence previous experiences have on the outcome of the whole selection process.

#### 5.1.3 Scenario S2 – Regular Monitoring of the Marketplace

Experienced *Reusers* tend to be familiar with the domains they work with, and they typically monitor the marketplace to inform themselves of technologies and trends (even before they have a specific need). As a result, the component searching process is influenced by this familiarity, and it turns out to be a gradual process rather than a momentarily one.

In this context, when *Reusers* are looking for components, they review the portals they already know to see what components are being used by the community. “*We know or have bookmarks of several portals we usually review to be aware of the components and technologies that are being used by the community. In particular, the most representative portal for our work is TheServerSide*” (ES3). Portals are however not the only kind of resources used in this monitoring and *Reusers* benefit from a range of information sources. One respondent said that they read “*different private blogs where one basically picks up trends*” (NO2), another preferred information from printed source. “*We buy a lot of magazines and typically O’Really Books*” (NO3).

In addition, *Reusers* will tend to come back to the ones he knows and trust. “*There are a lot of portals about OSS and technologies, but I tend to use the ones I usually follow and trust*” (ES3). Reading a variety of news sites and portals could be time consuming and one respondent said that “*I am more depending on the RSS feeds which I subscribe to. I do not actively read as much on the page [as before]*” (NO1).

#### 5.1.3.1 Problems Related to Scenario S2

*Reusers* mentioned to be concerned about trust and contradictory information. “*Sometimes it is difficult to formulate an opinion from information contained in internet because some of the opinions are extremely contradictory, so it is not easy to decide if a component could be a candidate*” (ES2).

In general, *Reusers* considered that having more comprehensive search functionality and more flexible classification systems in portals could be valuable. Since search was used a lot it could simplify the identification of components.

#### 5.1.4 Scenario S3 – Open Search

When *Reusers* do not have strong experience in a domain, they usually do not know where to find components that may cover the functionality they are looking for. In these cases, they mostly mentioned to use general search engines (Google was mentioned the most). “*When we do not have a clear idea of the kind of components that may cover the functionality we are looking for, we directly go to Google*” (ES2).

From our assessment of the marketplace we observed that some specialized search engines as Google Code Search or Koders exist, but none of the interviewees

mentioned to use them. *“No, we do not know any specialized tool to find components, we always use Google”* (ES1).

Regarding the use of intermediary portals hosting hundreds of OSS components, a reuser said *“I know SourceForge, but in portals like this it is really difficult to navigate and find relevant components. It is better to find components in a specialized portal (i.e., domain-specific) and then go to SourceForge for down-load it”* (ES1)

#### 5.1.4.1 Problems Related to Scenario S3

*Reusers* recognize that using general search engines as Google, the number of returned hits is frequently large and many of these hits are irrelevant. *“We know we will have a lot of irrelevant hits but anyway it is easier to arrive to the component and its community by Google”* (ES5). In addition, the accuracy of the search relies heavily on how well they are able to identify the exact terms to describe the functionality *“It is more than anything Google search tactics”* (ES1).

## 5.2 Evaluation and Decision

The evaluation and decision process implies the investigation of the features of the candidate components in order to choose the one(s) that (best) fit the system requirements.

### 5.2.1 Existing Resources for Performing OSS Evaluation and Decision

Our assessment of the existing resources to perform such tasks leads us to confirm that each existing portal describes components following its own classification and description model. There are in other words no standard for describing components. Thus, very different kinds of information are included: from technical documentation, newsletters, and articles to information coming from social collaborative features, as wikis, chats, blogs, and forums.

The following scenarios describe how *Reusers* use existing resources when they come to perform the component evaluation and decision activities.

#### 5.2.2 Scenario E1 – Experience-Based Evaluation

*Reusers* usually do not have enough time to evaluate components as much as they would like. As a result, they tend to use components they already know. *“We usually cannot evaluate a component as much as we would like because it requires time we usually do not have, so we tend to use only components we know or some colleagues have recommended”* (ES4).

In some cases they did not even considered other candidates that could fit better to the required task. *“We prefer to use a component we already know than assuming the risks of using a new one, even when the new component could perform better”* (ES4).



### 5.2.2.1 Problems Related to Scenario E1

*Reusers* recognize that the evaluation and decision process is not performed properly and it is mainly biased for personal experiences that narrow the exploitation of reuse. We observed that the reuse of components is mainly seen in the form of fine-grained or commodity components that do not imply so much risk if they should be replaced.

### 5.2.3 Scenario E2 – Searching for Information for Evaluating Components

Although the formal documentation of the OSS component obtained from the *Provider* was considered important at the technical level, experience either personal or of others plays a crucial role when evaluating and deciding which component to use, as stated in Scenario E1.

If the *Reuser* does not have personal experience with a particular component he usually consults his personal network (i.e., colleagues) or uses Google to search for more information. “Another things which we almost always do is to read opinions ... and examine a bit the experiences other people have.” (ES3). These searches are usually performed by using the name of the component as a keyword, in order to find other portals, forums or blogs which can provide more information. The most valuable information is: experience reports of successful implementations, possible problems with the use of the component and solutions to these problems. If required, *Reusers* may for instance pose questions in forums or on mailing lists as well.

### 5.2.3.1 Problems Related to Scenario E2

Existing portals are facing serious problems for making information available in a suitable way [6]. Some reusers mentioned to be unhappy with the completeness and quality of the documentation. One developer complained over this quality saying “There are a lot [of OSS projects] which are not well documented” (NO3). Improving this documentation could simplify the evaluation of the components. “What would have made it [OSS selection] a lot easier was if more of those offering OSS libraries put a bit more work into the documentation because this is often insufficient” (NO1). However, others mentioned to be quite happy with the documentation of other components. “We have been using Hybernate for a long time and we are quite happy with the extensive documentation and community support we have” (ES3). Assessing the interviewees answers and the characteristics of their projects we realized that the level of maturity of the community where the component come from, seem to have a significant impact on the quality of the documentation. Mature communities tend to provide better documentation.

The general perception of existing collaborative mechanisms for sharing knowledge as forums and wikis was really appreciated, but *reusers* mentioned to have problems with dealing with the subjective nature of opinions and unstructured information. “There are a lot of subjective meanings out there” (NO1). Indeed, existing portals do not offer advanced features that could help reasoning about and structuring scattered and subjective information. Furthermore, personalization features are also rare, and as basic as configuration of the appearance, payment personalization and RSS advertisement.

## 6 Discussion of Results

Although the benefits of having OSS components available are well-known, their successful reuse implies many challenges. Based on the empirical observations from the studies presented in the previous sections, we discuss the main issues that should be faced to improve OSS components selection.

- *The rapid changes and growing nature of the OSS marketplace.* New components and technologies are continuously offered, but also new and improved versions of existing components are frequently released. One straightforward example of the difficulty to deal with the changing nature of the marketplace can be seen in the evolution of mail servers products, which at present could also provide instant messaging or even video-conferencing facilities. So, to effectively classify components in order that they can be easily found by reusers is not trivial [2]. In this context, existing resources for enabling finding of components are facing some problems. Although industrial reusers do not really bother about searching, they are aware that the narrow character of their current searching processes and the influence of previous experiences hamper the fully exploitation of OSS reuse.
- *The lack of standards for describing OSS and huge of information diversity.* The lack of standards for describing OSS and the huge amounts of diverse, partial and subjective information about components makes it tough for reusers: first, to trust on the information and second, to perform an informed evaluation and decision. This contributes to the fact that decisions are mostly based on experience and limited knowledge of available components. Hence, the need for decision support mechanisms has been recognized [19]. Several research proposals recognize this problem and have proposed solutions that range from developing a general ontology [6, 23], to the use of semantic web technologies [1] and description logics. However, the real applicability of most of these proposals have resulted scarce [15, 19]. At this respect, personalization and recommendation functionality in portals are perceived as desirable by industrial reusers (e.g. users that selected this component also selected this...) as this could help them to get relevant information.
- *The influence of Experience on the selection process.* As our results confirm, experience either personal or of others play a critical role on the selection of OSS components. Although, the OSS spirit encourages the free and collaborative production and sharing of knowledge, there is a demanding need to effectively deal with the inherent subjectivity of the information. Reputation mechanisms as used in other business domains as ebay.com could be really valuable to deal with the subjectivity of diverse opinions.

Table 2 summarizes and relates the identified practical problems of Reusers when selecting OSS components and the challenges related to the elements of the marketplace.

**Table 2.** Summary of Reuser Problems and Associated Challenge

	Reuser Problem	Marketplace Element	Challenge
Searching	Difficulty to navigate through diversity of portals	Search Mechanism	Advanced and configurable search engines.
	Difficulty to find Components	Intermediary	More flexible classification schemas able to efficiently represent and deal with volatility and growing size of the marketplace.
Evaluation & Decision	Poor documentation quality	Provider	To improve components documentation
	Difficulty to find relevant information and deal with its subjectivity	Provider All	More sophisticated knowledge portals with decision-making support Need of integral efforts for improvements based on <i>Reusers</i> needs.

## 7 Threads of Validity

The results presented here are preliminary and we are about to conduct further studies, including more interviews to verify our results and get more observations.

The companies in this study were selected by convenience and we had only limited control over the selection of the projects. However, the results presented here come from companies in several domains and of different size. The projects are also of different size and types, and the interviewees have different back-grounds, see Table 1. One limitation is that most interviewed companies were developing web applications. The web applications domain is normally not represented by critical systems and, it is a domain in which there are plenty of OSS components. Both facts have an impact on how components are selected. We could expect somewhat different results in critical systems and in domains where OSS components are rare.

Furthermore, the interview guide used was prepared based on previous experience with similar studies [10, 13, 15] and it was pretested both in Spain and in Norway. To make answering as easy as possible for the interviewees, the interviews were performed in their respective native tongues. Two members of the re-search team are native Spanish speakers and two are native Norwegian speakers, one of which also speaks Spanish. To avoid misunderstandings and to simplify the analysis and discussions of the interviews, we taped, transcribed and translated the interviews to English as all authors speak English.

## 8 Conclusions and Future Work

The study and the results presented in this paper are discussed from the perspective of industrial reusers of OSS components with a particular focus on their needs when performing OSS selection. The systematic software reuse theory is used as a background for the discussions of the needs. Traditional view of reuse as a centralized reuse databases is challenged by the Internet as a massive marketplace for OSS components.

The relevance of this study is twofold: First, it may serve as a solid basis for understanding the real needs of OSS industrial reusers when selecting components, and therefore to properly address research and industrial efforts from several arenas. Second, the challenges identified here could help to the OSS market-place elements to contribute to maturing the marketplace. A more mature marketplace would allow reusers to fully exploit the benefits of OSS components and thereby contribute to increasing the adoption of OSS in the industry.

As future work, our goal is to further explore the industrial OSS selection by performing more interviews in both Norway and Spain. We expect to collect data that could help to describe how reusers and to understand how reusers can maximize the benefit of the OSS marketplace.

## References

1. Ankolekar A, Herbsleb JD, Sycara K (2003) "Addressing Challenges to Open Source Collaboration With the Semantic Web". In Feller J, Fitzgerald B, Hissam SA, Lakhani KR (Editors), *Taking Stock of the Bazaar: 3rd Workshop on Open Source Software Engineering*, pp 9-14.
2. Ayala CP (2008) *Systematic Construction of Goal-Oriented COTS Taxonomies*. PhD Thesis. Technical University of Catalunya.
3. Basili VR, Boehm BW (2001) COTS-based Systems Top 10 List. *Computer*, 34(5):91-95.
4. Bertoa M, Troya JM, Vallecillo A (2003) A Survey on the Quality Information Provided by Software Component Vendors. In *QAOOSE'03 Proceedings of the 7th ECOOP Workshop on Quantitative Approaches in Object-Oriented Software Engineering*, pp. 25-30.
5. Brereton P, Linkman S, Thomas N, Bøegh J, De Panfilis S (2002) Software Components - Enabling a Mass Market. In *STEP'2002: Proceedings of the 10th International Workshop on Software Technology and Engineering Practice*, pp. 169-176. IEEE Computer Society.
6. Cechich A, Requile-Romanczuk A, Aguirre J, Luzuriaga JM (2006) Trends on COTS Component Identification. In *ICCBSS'2006: Proceedings of the Fifth International Conference on Commercial-Off-The-Shelf (COTS)-Based Software Systems*, pp. 90-99. IEEE Computer Society.
7. Clark J, Clarke C, De Panfilis S, Granatella G, Predonzani P, Sillitti A, Succi G, Vernazza T (2004) Selecting components in large COTS repositories. *Journal of Systems and Software*, 73(2):323-331.
8. Fitzgerald B (2006) The Transformation of Open Source Software. *MIS Quarterly*, 30(3): 587-598.

9. Frakes WB, Kang K (2005) Software Reuse Research: Status and Future. *IEEE Transactions on Software Engineering*, 31(7):529-536, 2005.
10. Gereá M (2007) Selection of Open Source Components - A Qualitative Survey in Norwegian IT Industry. Master's thesis, Norwegian University of Science and Technology.
11. Ghosh RA (2006) Study On the Economic Impact of Open Source Software on Innovation and the Competiveness of the Information and Communication Technologies (ICT) Sector in the EU. Technical report, UNU-MERIT.
12. Gorton I, Liu A, Brebner P (2003) Rigorous Evaluation of COTS Middleware Technology. *Computer*, 36(3):50-55.
13. Hauge Ø, Sørensen C-F, Conradi R (2008) Adoption of Open Source in the Software Industry. In Russo et al. [22], pp. 211-222.
14. Knudsen D (2003) B2B-Marketplace Value Creation, Conceptual Predictions and Empirical Findings. In NOFOMA'2003 Proceedings of the 15th Annual Conference for Nordic Researchers in Logistics, pp 318-331.
15. Li J, Conradi R, Bunse C, Torchiano M, Slyngstad OPN, Morisio M (2009) Development with Off-The-Shelf Components: 10 Facts. *IEEE Software*, March-April 2009.
16. Majchrowski M, Deprez J-C (2008) An Operational Approach for Selecting Open Source Components in a Software Development Project. In EuroSPI'2008 Proceedings of the 15th European Conference on Software Process Improvement. Springer - Volume 16 of Communications in Computer and Information Science, pp 176-188.
17. Merilinna J, Matinlassi M (2006) State of the Art and Practice of Open-Source Component Integration. In Proceedings of the 32nd EUROMICRO Conference on Software Engineering and Advanced Applications, pp. 170-177. IEEE Computer Society.
18. Mohagheghi P, Conradi R (2008) An Empirical Investigation of Software Reuse Benefits in a Large Telecom Product. *ACM Transactions of Software Engineering and Methodology*, Vol. 17, No. 3, Article 13, 30 pages.
19. Mohamed A, Ruhe G, Eberlein A (2007) COTS Selection: Past, Present, and Future. In Proceedings of the 14th Annual IEEE International Conference and Workshops on the Engineering of Computer-Based Systems, pages 103-114. IEEE Computer Society.
20. Norris JS (2004) Mission-Critical Development with Open Source Software: Lessons Learned. *IEEE Software*, 21(1):42-49.
21. Prieto-Díaz R, Freeman P (1987) Classifying Software for Reusability. *IEEE Software*. 4(1):6-16.
22. Russo B, Damiani E, Hissam SA, Lundell B, Succi C (2008) (Editors) Open Source Development Communities and Quality Working Group 2.3 on Open Source Software. IFIP International Federation for Information Processing. Springer.
23. Simmons G, Dillon T (2006) Towards An Ontology for Open Source Software Development. In Damiani E, Fitzgerald B, Scacchi W, Scotto M (Editors), Open Source Systems IFIP Working Group 2.13 Foundation on Open Source Software, pp 65-75. Springer.
24. Ulkuniemi P, Seppanen V (2004) COTS Component Acquisition in An Emerging Market. *IEEE Software*, 21(6):76-82, 2004.
25. Umarji M, Elliott-Sim S, Lopes C (2008) Archetypal Internet-Scale Source Code Searching. In Russo et al. [22], pp. 257-263.
26. Vitharana P, Zahedi F, Jain H (2003) Design, Retrieval, and Assembly in Component-based Software Development. *Communications of the ACM*, 46(11):97-102.