# An Investigation of the users' perception of OSS quality

Vieri del Bianco[1], Luigi Lavazza[2], Sandro Morasca[2], Davide Taibi[2] and
Davide Tosi[2]

1 University College Dublin, Systems Research Group, CASL,
Dublin, Ireland
vieri.delbianco@ucd.ie

2 Università degli Studi dell'Insubria, Dipartimento di Informatica e
Comunicazione,
Via Mazzini, 5 – 21100 Varese, Italy
{luigi.lavazza sandro.morasca, davide.taibi,
davide.tosi}@uninsubria.it,
WWW home page: http://www.dicom.uninsubria.it

**Abstract**. The quality of Open Source Software (OSS) is generally much debated. Some state that it is generally higher than closed-source counterparts, while others are more skeptical. The authors have collected the opinions of the users concerning the quality of 44 OSS products in a systematic manner, so that it is now possible to present the actual opinions of real users about the quality of OSS products. Among the results reported in the paper are: the distribution of trustworthiness of OSS based on our survey; a comparison of the trustworthiness of the surveyed products with respect to both open and closed-source competitors; the identification of the qualities that affect the perception of trustworthiness, based on rigorous statistical analysis.

**Keywords**: Open Source Software quality, perceived quality, trustworthiness.

## 1    Introduction

Quality is often an elusive concept in Software Engineering. First, many attributes exist that may be used to describe software quality. For instance, the ISO9126 standard [1] views software quality as a multi-attribute concept, and different people may place different emphasis on the same attribute, depending on their experience, goals, and software at hand. In addition, the actual quantification of even some specific attribute may be problematic, as measures for that attribute may not be mature enough to have reached a sufficient degree of consensus, or may provide inconclusive results. For instance, two measures for the same software attribute may rank two software applications A and B in a conflicting way, i.e., one measure may rank A better than B while the other measure may reverse the ranking.

So, software stakeholders often choose to adopt one application over another based on the quality they perceive, instead of an objective quality evaluation. To some degree, this happens with several different types of products, if not all. For

instance, a prospective buyer may choose one car over another based on his or her own perception of the overall quality of the car, or some of the car's characteristics, or even the characteristics of the car's manufacturer. At any rate, in the case of cars, a number of objective measures exist, like length, width, height, volume of the engine, maximum speed, number of miles per gallon, number of seconds needed to get to some specified speed, so a prospective buyer can make informed decisions. When it comes to software, however, the lack of consensus measures makes decisions even more based on perceptions. Perceptions may be even more important for Open Source Software (OSS) than for other types of software. OSS has often suffered from some kind of biased perception, probably based on the idea that OSS is built by amateur developers in their spare time. It took a few years and a few success cases to dispel at least some of these perceptions about OSS, but some of that stigma is believed to still taint the reputation of OSS vs. Closed Source Software (CSS) at least in some environments. So, it is important to study how various attributes of OSS are perceived, to check if those perceptions about OSS qualities are still valid today and which specific qualities are believed to need improving more than others.

In this paper, we report on an empirical study about the perception of OSS qualities. We carried out the study in the framework of the QualiPSo project [9], which is funded by the European Union in the 6th Framework Program. Trustworthiness is the main focus of the QualiPSo project as for OSS product evaluation. However, OSS trustworthiness itself is a broad concept. On the one hand, trustworthiness is closely related to the idea of overall OSS quality: an OSS product is adopted only if stakeholders have sufficient trust in its quality. On the other hand, as OSS trustworthiness is influenced by a number of diverse factors which may include product- and process-related ones, several concepts and sources of information may need to be taken into account when studying OSS trustworthiness.

In the QualiPSo project we investigated the factors that are *believed* to affect trustworthiness [3] by OSS stakeholders. Then, we defined a conceptual model that represents the dependence of trustworthiness on other qualities and characteristics of the software [4]. To prove the validity of such conceptual model and provide it with quantitative models of trustworthiness, we collected both subjective evaluations and objective measures of OSS. Specifically, the subjective evaluations concerned how users evaluate the trustworthiness and other qualities of OSS. These evaluations are here analyzed in a rigorous way to derive indications concerning the quality of OSS that are both quantitative and reliable, since they are rooted on a reasonably wide sample of users' opinions.

Our investigation has shown that the majority (56%) of OSS products are considered very trustworthy and that the surveyed OSS products are generally considered better than the competitors (both OSS and CSS) by their users. Finally, we discovered statistically significant models that quantitatively describe the dependence of trustworthiness on qualities like reliability, usability, interoperability, efficiency, and documentation.

The paper is organized as follows. Section 2 describes data collection. Section 3 reports the results of the analysis, while the threats to the validity of the results are discussed in Section 4. Section 5 discusses the related work, and Section 6 draws some conclusions and sketches future work.

## 2    The Investigation

We carried out a survey to collect OSS stakeholders' evaluations of several OSS products according to a number of qualities. We actually selected just a few of all the qualities identified in the GQM plan that defines the QualiPSo notion of trustworthiness [4], because we knew that users may not be able to evaluate many OSS products, and, for each OSS product, too many of its qualities. Thus, in addition to a few questions characterizing the users (including how familiar they were with the product), we asked them to evaluate the overall trustworthiness of the products and the following qualities, which are believed to be the ones that most affect trustworthiness, based on a previous survey that we carried out among OSS stakeholders [3]:

- Usability;
- Portability;
- Functional requirements satisfaction;
- Interoperability;
- Reliability;
- Security;
- Developer community utility;
- Efficiency;
- Documentation;
- Trustworthiness vs. OSS competitors;
- Trustworthiness vs. CSS competitors.

We used a questionnaire to ask our respondents how they would rate the qualities of up to 22 Java and 22 C++ OSS products. The list of products appears in Fig. 1. We used a 1 to 6 ordinal scale, where 1 was the worst evaluation and 6 the best evaluation for a specific quality of a product with the following possible answers:

1 = absolutely not;
2 = little;
3 = just enough;
4 = more than enough;
5 = very/a lot;
6 = completely.

For illustration's sake, one of the questions was "How usable is the product?" with reference to some specified product. All other questions about all other qualities were asked in a similar fashion.

Up to the end of August 2009, we collected 100 questionnaires, containing 722 product evaluations, of which about 36% concerned Java products, while the remaining ones concerned C++ products.

The questionnaires were collected at major international events, not necessarily dealing with OSS topics, as summarized in Table 1.

**Table 1.** Events where data where collected.

| Event | Date (in year 2009) and location | Collected questionnaires | Product evaluations |
|---|---|---|---|
| Apache Conference | March 24-27, Amsterdam, The Netherlands | 15 | 31 |
| OW2 Conference | April 1-2, Paris, France | 20 | 31 |
| XP 2009 | April 24-30, Pula, Italy | 12 | 95 |
| OSS 2009 | June 2-5, Skovde, Sweden | 2 | 5 |
| ICSE 2009 | May 15-20, Vancouver, Canada | 9 | 69 |
| CONFSL 2009 | June 12-13, Bologna, Italy | 3 | 27 |
| QualiPSo Meeting | July 1-2, Madrid, Spain | 6 | 38 |
| ESC | August 30-31, Venice, Italy | 31 | 411 |
| Others | | 2 | 15 |

We did not screen our respondents beforehand, so we used what is known as a convenience sample. The possible effects that this may have had on our empirical study are addressed in Section 4.
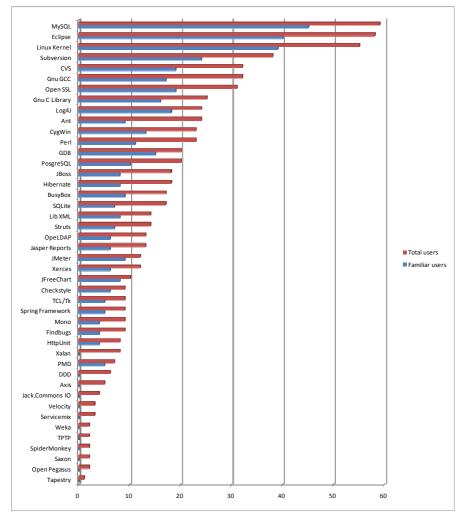
In the rest of the paper, we often call the respondents "users;" however, in addition to end-users, these "users" include also developers, managers, and stakeholders that are interested in OSS for various reasons.

# 3    The Results of the Investigation

## 3.1    The Popularity of the Products

A first result of our investigation concerns the popularity of the OSS products we selected. Since users were asked to answer about the products they knew well enough, the number of evaluations received by a product may be taken as a reasonable indication of its popularity.

Fig. 1 shows how many users evaluated each product, and how many of them answered that they have good familiarity with the product (the shorter bars report how many users rated their familiarity > 3).

**Fig. 1.** Number and familiarity of respondents per product.

The results reported in Fig. 1 were quite expected: MySQL, Eclipse and the Linux Kernel appear to be the most popular products.

We then proceeded to evaluate whether the popularity could be explained in terms of the type of the product (end-user oriented vs. programmer oriented, database management systems vs. configuration management systems vs. libraries, etc.). However, we found no such relationships. We consider this a good result, since it seems to indicate that users evaluated the actual qualities of the products, independent of their types and target users.

## 3.2    The Trustworthiness of the Products

We take the median of the evaluations as representative of the evaluations provided by the respondents (including end users and stakeholders) to the overall trustworthiness of OSS products. However, we noticed that some products had very low median grades because of only two or three respondents who had little familiarity with the product. As such evaluations may be deemed unreliable, we removed the evaluations of respondents with familiarity $\leq 3$ from the dataset. Also, we considered only products that were rated by at least 4 respondents.

Fig. 2 shows the median of users' evaluations for each product. On the x axis are the 32 products for which we collected enough data from users having sufficient familiarity. In other words, "Products" is a nominal variable.

It appears that users are generally very satisfied with the OSS products. Nevertheless, the facts that no product's median reached the maximum, that in several cases the median was only 4, and in one case even 3, shows that the users were not 'fanatic' of OSS. Rather, they seem to have provided well-balanced and reliable evaluations.
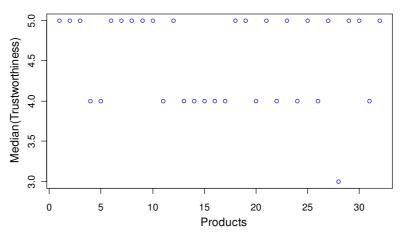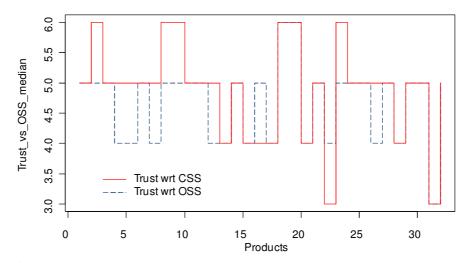


**Fig. 2.** Overall trustworthiness of the evaluated products (medians per product considered).

## 3.3    OSS vs. CSS (Closed-Source Software)

Users were asked to rate the trustworthiness of every product in comparison to similar OSS and CSS products. The medians of these ratings are illustrated in Fig. 3.

**Fig. 3.** Overall trustworthiness of the evaluated products compared to similar OSS and CSS products (medians per product considered).

A first observation suggested by the figure is that in general the evaluated projects are highly trusted with respect to both OSS and CSS competition. In fact, both scores are positioned mainly in the 4–5 range.

It is then possible to observe that in 21 out of 32 cases (65.6%) the CSS and OSS alternatives are considered of equivalent quality, though generally lower than the considered OSS product. This result seems to confirm that the choice of OSS products to evaluate was actually a good one, in that the considered products are generally considered very well positioned with respect to OSS and CSS alternatives.

In only 9 cases out of 32 (33%), the OSS alternatives are better than the CSS alternatives (this is the case whenever the red solid line is above the blue dashed line), while in only 2 cases out of 32 (6%) the CSS alternatives are considered better than the OSS ones.

The overall impression that is conveyed by Fig. 3 is that the OSS user community does trust OSS, but not in a fanatic manner, since the quality of CSS is also acknowledged, e.g., by considering OSS and CSS alternatives to leading OSS products as substantially equivalent.

### 3.4    The Quality of OSS Products

In the QualiPSo project, we have investigated the qualities that –according to OSS users– most affect the overall notion of OSS trustworthiness [3]. Following such indications, we have built a conceptual model of OSS trustworthiness that proposes an explanation of how OSS product sub-qualities (like as-is utility, exploitability in development, functionality, reliability) contribute to determining the

overall trustworthiness as perceived by users. Such model is defined via a GQM plan
[4], which involves also objectively measurable characteristics of OSS. The idea is
that, when enough data are available, we can build a quantitative model that explains
to what extent the subjectively perceived qualities of OSS depend on its internal
characteristics.

The data reported here are the result of the data collection (concerning
exclusively the subjective evaluations) performed as part of the execution of the
GQM plan.

Fig. 4 illustrates the distributions of the median evaluations of different products
for each surveyed quality. For this analysis, only the products evaluated by at least
10 users with sufficient familiarity have been considered. For each quality, the box
represents the range comprising half the population, the thick segment represents the
median, the dashed lines extend to the most extreme data point which is no more
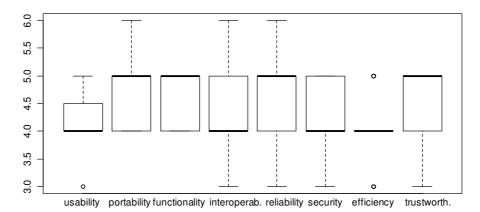than 1.5 times the interquartile range from the box, while the small circles indicate
outliers.



**Fig. 4.** Boxplots and medians for the qualities of the evaluated products.

A first observation suggested by Fig. 4 is that all qualities are given quite high
grades in general, as most evaluations are between 4.0 and 5.0. Also, while overall
trustworthiness is rated very well (see also Fig. 2), the users have been more critical
with other qualities –like usability, interoperability, security and efficiency– which
are given lower grades than trustworthiness. It is interesting to note that security,
which is usually very positively correlated with trustworthiness, is not rated
particularly well, even though most products are graded "more than enough secure."

Fig. 5 reports the box-plots that synthesize the distribution of grades across
products for every quality concerning the support to the end user, namely the
available documentation and the support from the developer community. This is a
rather relevant aspect, since users of OSS products often need to rely exclusively on

the available documentation or the support from the developer community in order to get information or resolve problems concerning OSS products.
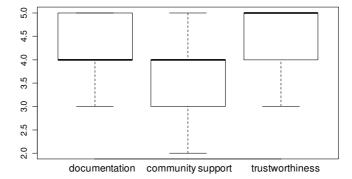


**Fig. 5.** User-support qualities (medians) of the evaluated products.

The median value of user support is rated "more than enough" in general. However, user evaluations are not aligned with conventional wisdom: documentation, which is often considered a weak point of OSS, is rated quite well, while the support by the developer community, which is generally believed to be a strong point of OSS is not rated very well for several products.
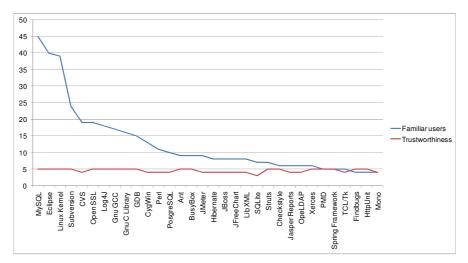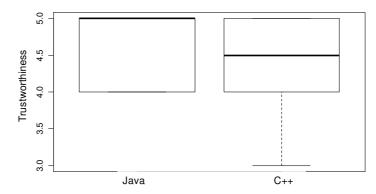


**Fig. 6.** Trustworthiness and popularity.

Fig. 6 shows that the four most popular products are also among those considered most trustworthy. Anyway, there is clearly no correlation between popularity and

trustworthiness: for instance, several products having relatively little popularity are considered very trustworthy.

## 3.5    Influence of the Implementation Language on the User-perceivable Trustworthiness

We collected users' opinions on products written in Java or C++. We investigated if the implementation language affects user-perceived trustworthiness. The box-plots in Fig. 7 summarize the distributions of the trustworthiness evaluations for C++ and Java programs



**Fig. 7.** Comparison of the overall (median) trustworthiness of the evaluated products depending on the implementation language.

While 64% of the Java products are rated "very trustworthy," only 50% of the C++ ones appear to get such rating. According to Fig. 7, Java programs appear to be slightly better than C++ ones. However, just one additional C++ project rated 5 would move the median trustworthiness of C++ programs to 5, thus making it equal to the one of Java programs. In conclusion, it is not possible to state that there is a dependence of trustworthiness on the implementation language.

## 3.6    Which Factors Affect OSS Trustworthiness?

Here, we investigate whether there is a statistically significant dependence of trustworthiness on other subjective qualities. To this end, we consider the fractions of users that are satisfied with the given qualities. For instance, the fraction of users satisfied with a product's trustworthiness is computed as the number of users that rated the product's trustworthiness above a given threshold, divided by the total number of users who evaluated the product's trustworthiness. The satisfaction

threshold was set at 4, i.e., the satisfied users are those who rated the product 5 (very good) or 6 (completely satisfactory).

To have reasonably significant fractions, we limited the analyses to product that were evaluated by at least 10 users having a good familiarity with the product.

The analyses reported below were conducted using Ordinary Least Squares (OLS) regression. The choice of OLS regressions is justified by the fact that both the dependent and independent variables are in the 0..1 range.

We used 0.05 as the statistical significance threshold, as is customarily done in empirical software engineering studies. Therefore, all the reported models have p-value < 0.05. The normality of the distribution of the residuals, which is a statistical requirement for safely applying OLS regression, was tested by means of the Shapiro-Wilk test [5]: consistent with our statistical significance threshold, p-values > 0.05 do not allow the rejection of the normality hypothesis.

A first result involves the dependence of trustworthiness on reliability. If we denote by *rrel* the fraction of users satisfied with reliability and *rtrust* the fraction of users satisfied with trustworthiness, we obtained the following statistically significant OLS regression model:
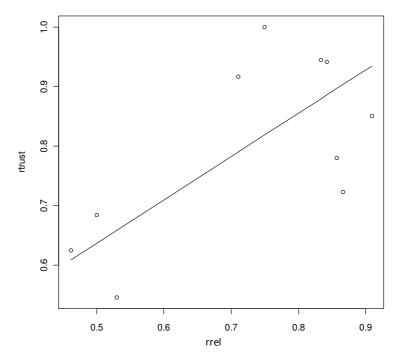
rtrust = 0.2726 + 0.7278 * rrel



**Fig. 8. Dependence of trustworthiness on reliability: the linear regression line.**

The residuals are normally distributed (i.e., the normality hypothesis cannot be rejected), and the determination coefficient is reasonably good ($R^2$ = 0.6391, adjusted $R^2$ = 0.594). So, around 60% of the variability in the degree of trustworthiness satisfaction is explained by the degree of reliability satisfaction.

The precision of the fitting is quite good: MMRE=9.2%, pred(25)=90%, with errors in the -13%..+25% range.

No valid OLS regression model could be found between trustworthiness and the satisfaction of functional requirements, portability, security, and the usefulness of the developers' community. On the contrary, we found statistically significant models of the dependence of trustworthiness on usability, interoperability, efficiency and documentation. These results are summarized in Table 2. In the "Line equation" column, the dependent variable is always *rtrust*, while the independent variable is the fraction of users that were satisfied with the quality reported in the 'Quality' column.

**Table 2.** Correlations found.

| Quality | Line equation | $R^2$ | MMRE | Pred(25) | Error range |
|---|---|---|---|---|---|
| Usability | y= 0.3686 + 0.7252 x | 0.4634 | 11.8% | 90% | -16% .. +26% |
| Interoperability | y = 0.1370 * 0.9479 x | 0.6817 | 9.2% | 100% | -16% .. +19% |
| Efficiency | y = 0.3642 * 0.7172 x | 0.6579 | 8.6% | 100% | -21% .. +14% |
| Documentation | y = 0.3712 * 0.7289 x | 0.6256 | 8.9% | 100% | -18% .. +18% |

## 4    Threats to Validity

A number of threats may exist to the validity of a correlational study like ours. We now examine some of the most relevant ones.

### 4.1    Internal Validity

We checked whether variables are normally distributed when carrying out OLS regressions, as required by the theory of OLS regression. Consistent with the literature, we used a 0.05 statistical significance threshold, the same we used for all statistical tests in our paper. The vast majority of statistical tests we carried out to this end provided quite strong evidence that the variables are indeed normally distributed, with the exception of Reliability and Usability, for which the p-values obtained with the Shapiro-Wilk normality test are 0.0612 and 0.09, respectively. These values are close to the 0.05 statistical significance threshold, but, based on these values, we could still not reject the hypotheses that Reliability and Usability are normally distributed, so we could carry out OLS regression. At any rate, the statistical tests used in OLS regression are somewhat robust and they can be practically used even when the variables' distributions are not that close to normal.

**4.2   External Validity**

Like with any other correlational study, the threats to the external validity of our study need to be identified and assessed. The most important issue is about the fact that our sample may not be fully "balanced," and that may have somewhat influenced the results. While this may be true, the following points need to be taken into account.

- It was not possible to interview several additional people that could have made our sample more "balanced," because they were not available or had no or little interest in answering our questionnaire.
- No reliable demographic information about the overall population of OSS "users" is available, so it would be impossible to know if a sample is "balanced" in any way.
- Like in many correlational studies, we used a so-called "convenience sample," composed of respondents who agreed to answer our questions. We collected information about the respondents' experience, application field, etc., but we did not make any screening. Excluding respondents based on some criteria, which must have been perforce subjective, may have resulted in an "unbalanced" sample, which may have biased the results.
- We dealt with motivated interviewees, so this ensured a good level for the quality of responses.
- There is no researcher's bias in our survey, since we simply wanted to collect and analyze data from the field, and not provide evidence supporting or refuting some theory.

**4.3   Construct Validity**

An additional threat concerns the fact that the measures used to quantify the relevant factors may not be adequate. This paper deals with trustworthiness, which is an intrinsically subjective quality, so the only way to measure it is to carry out a survey. As for the the other qualities, we are interested in stakeholders' evaluations and not in objective measures (which do not exist anyway), so, again, a survey is adequate to collect information about them.

## 5   Related Work

Several attempts were made to address the issue of software quality assessment in general, and within OSS in particular (see for instance the seminal papers by Audris Mockus et al. [12][14][15]). The online communication platforms and tools used in the development process (Concurrent Versioning System – such as CVS– Mailing Lists, Bug Tracking Systems – such as Bugzilla – and online discussion forums) contain a considerable amount of evaluations and data about the quality of

the software project. Therefore, such repositories have often been used for extracting data concerning the quality of OSS.

In [6] Tawileh et al. define a new approach for quality assessment of F/OSS projects based on social networking. They exploit the use of social networks of users formed around F/OSS projects, in order to collect data about the perception of OSS projects' quality and to make recommendations according to user preferences. Unfortunately, no data about the user' perception of the quality of OSS projects are reported in [6]. In our work, we collect data through a more controlled environment than social networks (i.e., via questionnaires dispensed to OSS users); we statistically analyze the collected data and we report how users perceive the quality of a representative sample of Java and C++ OSS products.

The TOSSAD portal stores an extensive set of surveys about the adoption of OSS products in target countries. For example in [7] OSS users are asked about their perception of quality about OSS in general. In our survey, we investigate the perceived quality of specific OSS products instead of the perceived quality of OSS in general.

In 2009, the Eclipse foundation conducted a deep survey about the overall quality of their IDE as perceived by users [8]. The survey is focalized only on their development environment; moreover, it does not take into account the specific qualities of Eclipse. In our work, we ask OSS users about several specific aspects of quality, thus ranking not only the general perception of trustworthiness, but also the perception of reliability, interoperability, efficiency, usability, and documentation of OSS products.


## 6    Conclusions and Future Work

The evaluation of the trustworthiness of OSS is important because of OSS ever increasing importance in software development and practical applications. However, lacking objective measures, OSS users and stakeholders rely on their own somewhat subjective evaluations when deciding to adopt an OSS product.

We carried out a survey to study the users' perception of trustworthiness and a number of other qualities of OSS products. We selected 22 Java and 22 C++ products, and we studied their popularity, the influence of the implementation language on trustworthiness, and whether OSS products are rated better than CSS products.

In addition, our results seem to provide evidence in favor of the existence of a few relationships between the user evaluations of a number of OSS qualities and trustworthiness. So, it is possible to have an idea of the impact that the evaluations of these qualities have on trustworthiness. Some trustworthiness evaluation methods have been proposed to let potential users assess the quality of OSS products before possibly adopt them. Such methods –like the OpenBQR [2] and the other similar approaches [10][11][12]– face typically two problems: what are the factors that should be taken into consideration, and what is the relative importance of such

factors? Generally these decisions are left to the user, who has to choose the qualities in a usually long list and assign weights. So, the work reported here improves our knowledge of the user-perceived qualities and trustworthiness of OSS products and of trustworthiness models.

Our future work will include the following activities.
- Collecting additional data about users' evaluations of OSS.
- Collecting data about additional qualities that may be of interest.
- Carrying out studies to check whether there exist relationships between some structural characteristics of OSS (e.g., size, structural complexity) and the external, user-related qualities we study in this paper.
- Using the profiling information about respondents to build more precise models for specific classes of OSS stakeholders.

The survey of users' opinions is going on through an on-line questionnaire (the Trustworthy Products Questionnaire is accessible via the QualiPSo web page http://qualipso.dscpi.uninsubria.it/limesurvey/index.php?sid=58332&newtest=Y&lang=en) and we invite all interested readers to fill out the questionnaire and contribute to this study).

## Acknowlegments

## References

[1]   ISO/IEC 9126-1:2001 "Software Engineering—Product Quality—Part 1: Quality model", June 2001.
[2]   D. Taibi, L. Lavazza, S. Morasca, "OpenBQR: a framework for the assessment of OSS", Open Source Software 2007, Limerick, June 2007.
[3]   V.del Bianco, M. Chinosi, L. Lavazza, S. Morasca, D. Taibi, "How European software industry perceives OSS trustworthiness and what are the specific criteria to establish trust in OSS", QualiPSo report, October 2008.
[4]   V. del Bianco, L. Lavazza, S. Morasca and D. Taibi, "Quality of Open Source Software: the QualiPSo Trustworthiness Model", The 5th International Conference on Open Source Systems OSS09, 3-6 June 2009, Skövde, Sweden.
[5]   S. S. Shapiro, M. B.Wilk, "An analysis of variance test for normality (complete samples)". Biometrika 52 (3-4), 1965.

[6]     A. Tawileh, O. Rana and S. McIntosh, "A Social Networking Approach to F/OSS Quality Assessment", Proceeding of the International Conference on Computer Mediated Social Networking (ICCMSN'08), Dunedin, New Zealand, June 2008.

[7]     TOSSAD, "Annex5 – Survey Report". Web published: www.tossad.org/publications/. Accessed: 14/12/2009.

[8]     Eclipse, "The Open Source Developer Report". Web published: http://www.eclipse.org/org/press-release/Eclipse_Survey_2009_final.pdf. May 2009. Accessed: 14/12/2009.

[9]     QualiPSo project portal. http://www.qualipso.org.

[10]   Atos Origin, "Method for Qualification and Selection of Open Source software (QSOS), version 1.6", http://www.qsos.org/download/qsos-1.6-en.pdf

[11]   "Business Readiness Rating for Open Source - A Proposed Open Standard to Facilitate Assessment and Adoption of Open Source Software", BRR 2005 - RFC 1, http://www.openbrr.org.

[12]   "Making Open Source Ready for the Enterprise: The Open Source Maturity Model", from "Succeeding with Open Source" by Bernard Golden, Addison-Wesley, 2005, available form http://www.navicasoft.com

[13]   A. Mockus, R. T. Fielding, J. D. Herbsleb "Two case studies of open source software development: Apache and Mozilla", ACM Trans. Softw. Eng. Methodol., vol. 11, n. 3, 2002.

[14]   A. Mockus, D. Weiss, "Interval quality: relating customer-perceived quality to process quality, International Conference on Software Engineering 2008, Leipzig, Germany, May 2008.

[15]   A. Mockus, Ping Zhang, Paul Luo Li, "Predictors of customer perceived software quality", International Conference on Software Engineering 2005, St. Louis, May 2005.