

An Exploratory Study on the Two New Trends in Open Source Software: End-Users and Service

Namjoo Choi
College of Computing and Information
University at Albany, SUNY
nc236879@albany.edu

Indushobha Chengalur-Smith
School of Business
University at Albany, SUNY
shobha@albany.edu

Abstract

Many have been envisaging the emergence of Open Source Software (OSS) for general end-users and the enhancements in providing services and support, as the most critical factors for OSS success, and at the same time, the most critical issues which are holding back the OSS movement. While these two distinct waves in OSS evolution have become more observable, researchers have not yet explored the characteristics of these two distinct new waves. The current study found evidence for these two waves and further explored the two waves by empirically examining two hundred projects hosted in Sourceforge.net. We compared the characteristics of OSS projects that are intended for two disparate audiences: developers and end-users and found that projects for end-users supported more languages but also had more restrictive licenses as compared to projects for developers.

1. Introduction

Over the last several decades, Open Source Software (OSS) emerged as one of the most interesting and debated phenomena in the software industry. As an alternative to proprietary software, it has gained the attention of businesses, governments and researchers. Despite the appealing and promising advantages of OSS, however, proprietary software still prevails in today's software market. In an effort to better understand OSS and maximize its benefits, researchers have examined various aspects of OSS. Major research streams in this area include (1) identifying motivations for developers and users to voluntarily participate in OSS projects [e.g., 16, 24, 39], (2) comparing and contrasting the organization and control of OSS development to conventional software development [e.g., 14, 19], (3) investigating success factors for OSS development [e.g., 7, 8, 30, 31, 37].

One of the most unique characteristics of OSS is its evolving nature. As Raymond [35] pointed out, OSS

has evolved in different waves. In the seventies, most of the time and effort was devoted to hackers' personal interests, in the eighties, interest shifted toward the use of Internet tools, and in the nineties, operating systems became popular. For the next wave, practitioners and researchers have been predicting the advent of OSS for general end-users, as well as enhancements in providing services and support. Both have been considered the most critical factors for OSS success, and at the same time, the most critical issues which are holding back the OSS movement [e.g., 5, 6, 12, 26, 27, 29, 34, 36, 38, 41, 42]. Now, these two distinct waves in OSS evolution have become more observable and thus there is a need for researchers to turn their attention to more closely examining these two new promising waves.

The most obvious trend in the OSS movement over time is that the number of OSS projects has been growing as well as the number of OSS developers. Of special account in this trend, is the fact that the users of software have become more heterogeneous. Unlike the early stages in the OSS evolution, when most of the software was made for technically sophisticated users, there is a growing body of OSS projects geared toward general end-users [5, 6, 33]. Yet, most of the general end-users still use proprietary software as their main applications [33]. In this regard itself, it is clearly worth exploring and examining this new wave.

On the other hand, the uncertainty in service and support has long been referred to as a major hurdle to OSS adoption [22, 41, 42]. However, efforts to mitigate this are underway. For example, RedHat, a major distributor of the open-source operating system, Linux, generates its revenue by providing relevant services and support. Also, SourceForge.net, the biggest free web-based hosting system for OSS projects, recently implemented a system called Marketplace solely devoted to facilitate selling and buying services and support. We believe that this second new wave, separate from the first new wave, is also worth exploring to better understand the current stage of OSS evolution.

So far, a few studies [e.g., 1, 15, 29] have addressed the issues related to OSS adoption, but, to the best of our knowledge, no-one has explored or specified both of these two distinct new waves in the OSS evolution. To call for more attention to be directed toward them, the current study explores these two new waves by comparing OSS projects that are intended for developers to those that are intended for end-users, and presents an exploratory empirical analysis from the perspectives of these two new waves.

The next section provides a literature review and our research hypotheses. The data collection and the research method employed to test our hypotheses is described in the following section. We then present our results and outline the implications, limitations, and contributions of this research in the discussions and conclusions section.

2. Literature review and research hypotheses

2.1. The first wave: more OSS projects geared toward general end-users

Up to the late nineties, OSS was mostly by, of and for developers who have relatively strong technical skills. In other words, for most OSS, users were indeed developers [28]. The well-known statement by Raymond [34], “Every good work of software starts by scratching a developer's personal itch,” also demonstrates indirectly the above phenomena. It implies that most OSS was initiated and used by technically sophisticated developers. Most OSS starts out with an initiator(s) who wants to solve his or her own particular problem and as the project attracts more developers and users who have a similar problem, its source code is improved. To participate in an OSS project, one might share the same issue, or understand the source code or at least, have a basic programming background or relevant experience to the issue. Developers may be motivated to join an OSS project by the opportunity to add advanced features or improve the quality of the existing features.

However, given more awareness among the general public bolstered by the recent popularity of OSS such as Linux and Firefox, the user base of OSS started expanding to embrace more general end-users [5, 6, 33]. To explore this emerging wave, the current study first focuses on empirically verifying whether the OSS projects toward more general end-users have been actually increasing. We therefore hypothesize:

H1: OSS projects geared towards end-users have been increasing.

Although it is assumed that there are a growing number of the OSS projects geared toward the general public, proprietary software still dominates the market [33]. Possible reasons for the situation have been discussed by several authors and one of the major obstacles identified is that most of the project participants, whether they are developers or users, are essentially software developers who usually do not quite understand the issues that ordinary users often face [34]. In conjunction with this problem, usability has been discussed as one of the most problematic issues for OSS [e.g., 2, 10, 28, 33, 40]. Fortunately, however, there is an exemplary case that shows us how to deal with the usability issue of OSS. Firefox has not only been actively focusing on raising the general public's awareness of it, but also actively seeking their feedback to improve its usability. Now, many use Firefox as their primary web-browser and its user base seems to be constantly growing.

As usability started receiving attention as a problematic issue along with related issues such as user support and compatibility, we expect that the OSS projects geared toward end-users will care more about usability than the OSS projects geared toward developers. We therefore hypothesize:

H2: OSS projects geared toward end-users care more about usability than those geared toward developers.

In addition to usability, the number of languages supported by a project can be a good determinant of OSS projects that are geared toward end-users since they usually focus on universal software rather than local-specific software [6, 27]. Although there are many projects geared toward developers that support multiple languages, it can be argued that if equally successful, the OSS projects geared toward end-users are more likely to support multiple languages than the OSS projects geared toward developers. The rationale is that developers across the globe are more likely to be better educated than general end-users, and hence more likely to be comfortable in English. Since the user base for OSS projects intended for end-users can be anyone in the world who has access to the Internet, but is not necessarily fluent in English, in order to truly appeal to such a potentially diverse audience, the project must support multiple languages. We therefore hypothesize:

H3: OSS projects geared toward end-users are more likely to support multiple languages than those geared toward developers.

One of the unique characteristics of OSS is their licensing agreements. There are multiple types of

licensing structures with varying restrictions. Several studies have empirically found that the level of restrictiveness of a project's license type significantly differs by its intended audience type [e.g., 6, 27]. By employing the Sourceforge.net database, a compilation of nearly 40,000 open source projects, Lerner and Tirole [27] found that restrictive licenses (e.g., GPL) are more common for OSS projects geared toward end-users and system administrators, but significantly less common for those geared toward technical users and developers, and that the OSS that is consumer oriented (e.g., desktop tools and games) are significantly more likely to have restrictive licenses. Comino et al. [6] also found that restrictive licenses are more common for OSS projects geared towards end-users, but significantly less common for those geared toward developers. By limiting the sample to the OSS projects solely geared towards end-users and solely towards developers, we aim to further verify the findings from the previous studies [6, 27]. We therefore hypothesize:

H4: OSS projects geared toward end-users are more likely to have more restrictive licenses than those geared toward developers.

2.2. The second wave: providing service and support

As mentioned in the introduction section, the uncertainty in service and support has long been referred to as a major hurdle to OSS adoption [e.g., 12, 29, 35, 36, 38, 42]. More than twenty years ago, Stallman [38] suggested for developers' sake that OSS can generate revenue in complementary areas such as service and support. However, the efforts in this regard have been neglected despite the improved level of adoption for many OSS. Consequently, many organizations have been concerned about the difficulties in obtaining the necessary service and support, the lack of which often results in the abandonment of the organization's OSS adoption [42]. Reflecting this concern, however, there now exists a growing number of support services (e.g., RedHat, SpikeSource, Sourceforge.net). While it is encouraging to both developers and organizations, these practices are still at an early stage and needs to be further improved, relative to proprietary software [36].

From an economic perspective, several studies have clearly emphasized the significance of services and support in OSS success [e.g., 12, 26, 36, 42]. The discussions in these studies, however, are mainly focused on the organizational adoption of OSS, not the general public. While both organizations and the general public need services and support in their use of

OSS, the lack of focus on the general public is explicable considering that most OSS for organizations involve more complicated services and support, from implementation to maintenance, and therefore the availability of services and support is a more critical issue for adoption. It can be argued that OSS for the general public requires relatively less complicated services and support. Also, another potential explanation from OSS developers' perspective is that generating monetary compensation by providing services and support is more likely in the OSS aimed towards organizations rather than those for the general public.

Therefore, it might be possible to infer that there is a difference in the availability of services and support offered by developers between the OSS projects geared toward developers and end-users. Although the focus of the current study is on the introduction of the new wave, it might be worth exploring this inference. Therefore, we hypothesize that providing services and support for monetary compensation is more likely in OSS projects geared toward developers than end-users.

H5: The availability of services and support for monetary compensation is higher in OSS projects geared toward developers than end-users

Instead of generating monetary compensation by providing services and support, there has been another way of getting funded for OSS projects, viz. donation. Developers can solicit donations for their work and often use an opt-in feature to do so. By extending the inference discussed in hypothesis 5, it could also be inferred that the developers of OSS for organizations are more likely to opt in to receive donations due to their perception of organizations' willingness to offer monetary compensation for necessary services and support. Organizations are also more likely to have a culture of compensation as well as a pay structure in place for services and support. We speculate that donations to developers of OSS projects for the general public are more likely to be motivated by the personal generosity of the end-user.

Based on this discussion, we explore whether there is a difference in the number of developers who opt to receive donations between the OSS projects geared toward developers and end-users. This could suggest different approaches to success for different OSS projects: those geared toward developers and those geared toward end-users. Therefore, we hypothesize that OSS projects geared toward developers will have more developers who opt in to receive donations than OSS projects geared toward end-users.

H6: OSS projects geared toward developers have more developers who opt in to receive donations than those geared toward end-users.

Although developers can signal their interest in receiving donations, this measures only one side of the relationship, viz. the openness to commercial involvement on the part of the developers. On the other side is the reciprocal interest on the part of consumers to compensate developers for their efforts. Thus, hypothesis 6 is primarily concerned with developers' potential interest in generating revenue, and a complementary variable is the actual revenue generated, which can be measured by both the frequency and magnitude of donations. For completeness, we consider both sides of the equation, the proclivity of the consumers to pay as well as the tendency of the providers to facilitate payment.

It might be further inferred that, at a project level, the number and amount of donations received also differ across OSS projects intended for developers versus those intended for end users. We speculate that developers who are interested in furthering OSS are more likely to support it in kind, viz. by volunteering their expertise and time, rather than through monetary donations. On the other hand, end-users who may not be as skilled, may prefer to support OSS through monetary compensation. We therefore hypothesize:

H7: OSS projects geared toward end-users have more donations than those geared toward developers.

3. Research methodology

3.1. Data collection

Most studies of OSS have adopted case study methodology to focus on large, well-known projects such as Linux and Firefox [e.g., 31, 32]. As pointed out by Stewart and Gosain [39], however, they have a limitation that they may not be representative of the majority of OSS projects. Along with quantitative methodologies [e.g., 1, 6, 7, 8, 39], some studies started focusing on the data from various OSS hosting websites (e.g., Sourceforge.net, Freshmeat.net). Although there are still some pitfalls in using these data [21], their clear advantage is a large data set of a diversity of OSS projects.

For the current exploratory study, we compiled data from Sourceforge.net (<http://www.sourceforge.net/>). Sourceforge.net is the biggest (179,606 registered projects as of June 15, 2008) free web-based hosting system for OSS projects. Since it provides the same

environment with a standard technology toolset for all the registered projects, we reduce variance among the projects by eliminating differences in technology or setting. Sourceforge.net categorizes OSS projects by status based on their stage of product development: planning, pre-alpha, alpha, beta, production/stable, and mature. We created our dataset by restricting our attention to projects that have production/stable and mature development status, thus reducing the variance that could be introduced by the issues resulting from product immaturity or project abandonment [23]. Sourceforge.net also categorizes OSS projects into five categories based on their intended audience type: advanced end users, developers, end-users/desktop, quality engineers, and system administrators. As mentioned in section two, our interests are in comparing OSS for organizations vs. OSS for general public end-users. Therefore, we limited our sample to two categories: developers and end-users/desktop. We chose the end-users/desktop category because we believe that it represents more general public end-users than the advanced end-users category. Likewise the quality engineers and system administrators categories can be included in the broader developers category.

We manually compiled a total of 200 projects, 100 each of the most downloaded projects from developers and end-users/desktop categories during the period of March 4 to March 23, 2008. Projects that fell into more than one category were eliminated, so only those categorized as intended for only developers or only end-users/desktop, with production/stable or mature status, were chosen. Since we are interested in exploring the differences between the OSS projects geared toward developers and end-users from the perspectives of the two new waves, we believe that clear isolation better represents the characteristics of each category.

3.2. Measurement

End-user OSS projects increase: To verify that the OSS projects geared toward end-users has been increasing, monthly frequencies and percentages of the OSS projects that categorize themselves as geared towards end-users/desktop were taken from the database provided by the University of Notre Dame (https://zerlot.cse.nd.edu/mywiki/index.php?title=Main_Page). Since 2003, Sourceforge.net has licensed researchers at the University of Notre Dame (UND) to make the Sourceforge.net data available for academic and scholarly research. Although the previous literature [5, 6, 33] assumes that the OSS projects geared toward end-users has been increasing since the late nineties, we could only obtain data since 2003 (Jan. 2003, Nov. Dec. 2004, and regularly every month since Feb. 2005

up to May 2008) because the data is available since then.

Usability: Most OSS projects hosted by Sourceforge.net allow developers to describe themselves using job titles chosen from a pre-set group of titles including administrator, developer, porter, web designer, translator, doc writer, and graphic/other designer. As an indicator of how a project regards the significance of usability, we counted the number of positions related to usability in each project. The positions we designated as related to usability are web designer, graphic/other designer, editorial/content writer, doc writer, and translator.

Language: As mentioned earlier, the number of languages supported can be a good determinant of the OSS projects geared toward end-users. Each project in Sourceforge.net indicates the languages supported. For our measure the number of languages supported in each project was counted.

License: Projects in Sourceforge.net are currently grouped into more than 40 different license types. To classify them, we adopted the scheme suggested in [26] and verified in [6]. The scheme classifies licenses into three different categories according to the degree of restrictions imposed on the use of the software: highly restrictive licenses with copyleft and viral restrictions such as GPL, restrictive with only copyleft such as LGPL, and unrestrictive licenses such as BSD [6]. Since many projects list more than one license type, we further adopted the index also suggested by Lerner and Tirole [26]: “the index takes on the value 4 if all licenses are highly restrictive; 3 if some are highly restrictive; 2 if all licenses are restrictive but none are highly restrictive; 1 if some are restrictive but none are highly restrictive; and 0 otherwise”.

Service and support: To examine whether there are some differences in providing services and support for monetary compensation between the OSS projects geared toward developers and end-users, we counted how many developers indicate that they provide service and support in each project. This information in each project is also linked to a newly developed system titled, “Marketplace,” where further information about the services and support with cost are provided (<http://sourceforge.net/services/buy/index.php>).

Opt in for donation: Developers with the dollar sign attached to their ID in Sourceforge.net designates that this developer has opted-in to receive donations through the Sourceforge.net donation system. For our measure, we counted the number of developers with this dollar sign attached to their ID in each project for the measurement.

Donation received: Each project that has received a donation lists the donor’s ID accompanied by a gear sign. Five different colors are used to signify the size

of donation made to the project. Therefore, we summed all these values (1-5) in each project to measure the size of donation. Then, we divided these values by the number of downloads and scaled the final values for the measure. We used the number of downloads as a proxy for the user population to address a concern that the user population of OSS projects geared toward end-users is most likely to overwhelm that of OSS projects geared toward developers.

Intended audience: The OSS projects geared toward developers were coded as 1, and the OSS projects geared toward end-users were coded as 0.

3.3. Data analysis

In order to test the first hypothesis (OSS projects geared toward end-users have been increasing), a frequency analysis was conducted. For hypotheses 2 to 7, data were analyzed in two stages. First, a t-test for each variable was conducted to examine the mean difference between the two groups: the OSS projects geared towards developers and the OSS projects geared towards end-users. However, since we suspect that the number of developers on a project is correlated with some of the hypothesized variables (e. g., usability, service, opt in for donation) we decided to include the number of developers as a control variable in a multivariate analysis. Given that the dependent variable (the OSS projects geared toward developers vs. the OSS projects geared toward end-users) is binary, logistic regression analysis was conducted rather than multiple regression analysis [4, 20]. Therefore, beyond a t-test examining the mean difference between the two groups the logistic regression analysis allows us to assess the contribution of each variable in the presence of the other variables [9]. Considering that the research in this field is still in its infancy and the main purpose of the current study is to explore potential indicators that might represent each of the two groups, additional information obtained by performing the logistic regression analysis may be worthwhile. (SPSS 16.0 was employed to undertake the analysis.)

4. Results

Descriptive statistics and correlations among variables are shown in Table 1. As we suspect, it can be seen that the number of developers on a project is correlated with several variables (i.e., usability, language, service, opt-in for donation, intended audience). Thus we use the number of developers as a control variable in our logistic regression analysis.

For hypothesis 1, the result of the frequency analysis conducted indicates that the OSS projects geared toward end-users have been increasing, (see Figures 1 and 2), both in absolute numbers as well as in terms of a proportion of the total number of projects in Sourceforge.net. Even though the support found is limited to the projects in Sourceforge.net and the data was only available beginning in 2003, considering that

Sourceforge.net is currently the biggest OSS hosting website we believe that potential bias does not affect the result much. With regard to the outliers in Figure 1 and Figure 2 (e.g., 2007-05, 2003-01), we suspect possible systematic errors such as cleaning up non-active projects or data restructuring errors that we cannot detect over the Sourceforge.net database.

Table 1: Descriptive statistics and correlations among variables

	1	2	3	4	5	6	7	8	9	10
Mean	11.10	5.12	3.82E6	0.70	3.34	2.78	0.16	0.98	0.17	0.50
Std. Dev.	13.27	0.33	2.833E7	2.09	6.34	1.52	0.64	1.73	0.43	0.50
1. Developers	1.000									
2. Development Status	0.043	1.000								
3. Downloads	-0.011	-0.022	1.000							
4. Usability	0.328**	0.202**	-0.004	1.000						
5. Language	0.162*	0.041	0.392**	0.436**	1.000					
6. License	-0.138	-0.099	0.080	0.129	0.229**	1.000				
7. Service	0.233**	0.052	-0.006	0.022	0.086	-0.033	1.000			
8. Opt in for donation	0.579**	0.067	-0.025	0.266**	0.188**	0.003	0.312**	1.000		
9. Donation received	0.068	-0.107	-0.044	0.070	0.036**	0.029	-0.057	0.115	1.000	
10. Intended audience	0.168*	0.185**	-0.102	-0.151*	-0.244**	-0.586**	0.094	0.162*	0.051	1.000

** . Correlation is significant at the 0.01 level, * . Correlation is significant at the 0.05 level

Figure 1: End-Users (Frequency)

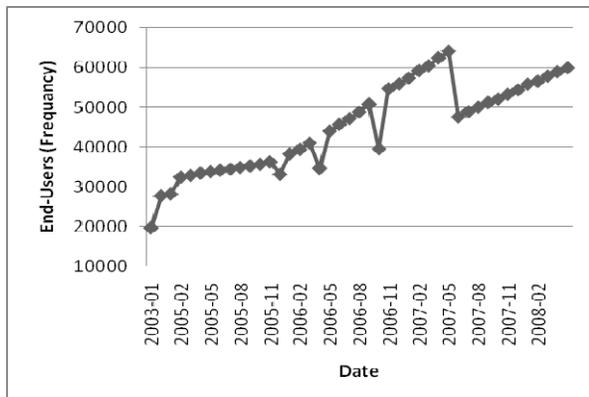
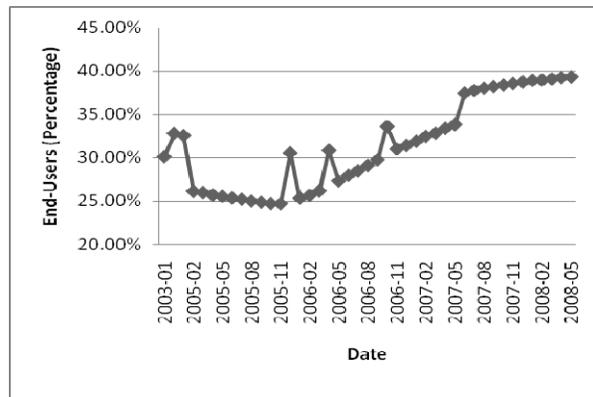


Figure 2: End-Users (Percentage)



To examine the mean differences in the independent variables by the two groups: the OSS projects geared toward developers and the OSS projects geared toward end-users, t-tests were conducted. The results (see Table 2) indicate that the independent variables analyzed have a significant

mean difference by intended audience for hypotheses 2 – 6 as hypothesized except for hypothesis 7. The most significant differences exist when comparing licenses and languages across the two groups. These results will be further examined in the discussions and conclusions section.

Table 2: T-tests results

	Intended Audience	Mean	Std. Dev.	T	Sig.
H2: Usability	Developers	0.38	1.42	-2.149	0.017**
	End-Users	1.01	2.56		
H3: Language	Developers	1.80	2.94	-3.535	0.000***
	End-Users	4.88	8.20		
H4: License	Developers	1.75	1.49	-10.183	0.000***
	End-Users	3.60	1.04		
H5: Service	Developers	0.22	0.63	1.334	0.092*
	End-Users	0.10	0.64		
H6: Opt in for donation	Developers	1.26	2.01	2.309	0.011**
	End-Users	0.70	1.34		
H7: Donation received	Developers	0.19	0.52	0.723	0.235
	End-Users	0.15	0.31		

***p<0.01, **p<0.05, *p<0.10

Finally, logistic regression analysis was conducted for hypotheses 2 to 7 [20]. The results are shown in Table 3. There was no statistically significant difference ($p = 0.533$) between the two groups (i. e., developers and end-users) by the number of developers, which was used as a control variable in the analysis. The significance of the regression coefficients of the other independent variables was examined to determine support for the hypotheses. The Wald chi-square statistic [17] used in the significance test. As for individual independent variables, the coefficients of language ($p < 0.05$) and license ($p < 0.01$) are significantly from zero. The coefficients of usability and opt in for donations shows only marginal significance at the 0.10 level (respectively, 0.099 and 0.059). The coefficients of the other variables (i.e., service and donation received) are not significant. These results are generally consistent with, though

weaker than, those from the individual t-tests performed earlier (e.g., service was only marginally significant in the t-test results). The results will be further examined in the discussions and conclusions section. While the focus of the analysis is not on building and suggesting a model, we provide the results regarding the model for additional information as follows. The goodness-of-fit statistic (chi-square (8) = 16.549; $p = 0.035$) indicates that the logistic regression model is not significantly different from a perfect model which correctly classifies all projects into their respective groups: developers and end-users. The discriminating power of the model was also assessed. As there are 100 developers and 100 end-users projects, the random choice is 50% accurate. The result indicates a classification accuracy of 83%, which is better than by random choice.

Table 3: Logistic regression analyses results

Dependent Variable is Intended Audience (1 = Developers, 0 = End-Users) Number of observation =200 (Developers = 100, End-Users = 100) -2 Log likelihood : Chi-square (7) = 98.803, p = 0.000 Goodness of fit: Chi-square (8) = 16.549, p = 0.035			
Independent Variable	Coefficient	Wald statistic	Significance.
No. of Developers	0.014	0.389	0.533
H2: Usability	-0.252	2.725	0.099*
H3: Language	-0.119	4.143	0.042**
H4: License	-0.978	38.326	0.000***
H5: Service	0.122	0.297	0.683
H6: Opt in for donation	0.312	3.571	0.059*
H7: Donation received	0.803	2.442	0.118
Constant	2.650	23.651	0.000***

***p<0.01, **p<0.05, *p<0.10

5. Discussions and conclusions

Our study offers some contributions to research as well as implications for practice, yet with several limitations. First and most importantly, the current study opens up new research into the two new distinct waves: end-users and service that has long been claimed as most critical for OSS success. To facilitate the study, some possible indicators were introduced that had not been investigated extensively in previous studies. However, considering that this is an exploratory study and, to the best of our knowledge, the first study of its kind, more potential indicators certainly need to be explored for these new distinct waves: end-users and service. Especially since the variables for hypotheses 5 to 7 (service and donation) were drawn from the authors' inferences rather than theoretical support from the literature, they need to be further examined.

Second, the study provides an exploratory empirical analysis. Most studies of OSS have adopted case study methodology to focus on large and well-known projects, whereas the current study fills the gap by offering an empirical quantitative analysis. The results from this exploratory analysis show that there are some differences in characteristics between OSS projects geared towards developers and those

geared towards end-users. The results support findings from previous studies [6, 26] that OSS projects geared toward end-users tend to have more restrictive license types than projects for developers. In addition, hypothesis 4, with regard to the difference in languages provided between OSS projects geared towards developers and those geared towards end-users was supported.

It is interesting to note that although usability has been pointed out as one of the most problematic issues for OSS by some studies [e.g., 2, 10, 28, 33, 40], the results indicate only marginal significance in difference, as compared to the differences between language and license. It is possible that OSS projects geared towards developers and those geared towards end-users are essentially on par in terms of usability. Our study used a single measure of usability, whereas usability may be a more complex, multi-dimensional construct. This could explain why it was significant in the t-test results but only marginally significant in the logistic regression analysis. Thus our measures of usability need to be refined to be able to capture more subtle differences.

While we believe that OSS projects geared toward developers are more likely to be adopted by organizations, many OSS focused on end-user requirements have also been adopted by

organizations. In the former case, it could be argued that the developers of those projects are more likely to provide services and support and solicit donations upon their perception of organizations' willingness to offer monetary compensation for necessary services and support. On the other hand, developers are expected to be able to better solve their problems when adopting a piece of OSS than end-users. The more pressing need to provide support for end-users may be balanced out by the potential for compensation from supporting organizations, and this trade-off may contribute to the relative lack of significance for hypothesis 5. Another possible explanation is that the feature called Marketplace in Sourceforge.net is relatively new (less than one year old), and hence the results for hypothesis 5 may be biased due to sample immaturity.

There has been a clear shift towards providing more services and support in recent years but what is unclear is how sustainable those efforts are. We anticipate that the response from the consumers will be crucial in determining the fate of this new trend.

The importance of understanding the current trends and exploring its characteristics, in any field, cannot be overemphasized. In this sense, the study provides a preliminary base to determine not only the current trends, but also the future directions in the software market. For example, as an increasing number of OSS projects geared toward end-users become more available, with better enhanced usability and other critical factors such as interoperability, many can benefit from using these OSS over the proprietary software that currently prevails in today's software market.

6. References

- [1] S. A. Ajila, and D. Wu, "Empirical study of the effects of open source adoption on software development economics", *The Journal of Systems and Software*, vol. 80, 2007, pp. 1517-1529.
- [2] M. S. Andreasen, H. V. Nielsen, S. O. Schröder, and J. Stage, "Usability in open source software development: opinions and practice", *Information Technology and Control*, vol. 35, no.3, 2006.
- [3] Berlecon, "Free/Libre and Open Source Software (FLOSS): Survey and Study", *International Institute of Infonomics*, University of Maastricht and Berlecon Research, 2002.
- [4] P. Y. K. Chau and K. Y. Tam, "Factors Affecting the Adoption of Open Systems: An Exploratory Study", *MIS Quarterly*, vol. 21, no. 1, 1997, pp. 1-24.
- [5] S. Comino, and F. M. Manenti, "Government policies supporting open source software for the mass market", *Review of Industrial Organization*, vol. 26, no. 2, 2005, pp. 217-240.
- [6] S. Comino, F. M. Manenti, and M. L. Parisi, "From planning to mature: on the determinants of open source take off", working paper 0517, Department of Economics, University of Trento, Italia, 2005.
- [7] K. Crowston, H. Annabi, and J. Howison, "Defining Open Source Software project success", *Proceedings of the International Conference on Information Systems*, Association for Information Systems, Seattle, WA, USA, December 14 - 17, 2003, pp. 327-340.
- [8] K. Crowston, J. Howison, and H. Annabi, "Information systems success in free and open source software development: theory and measures", *Software Process: Improvement and Practice*, vol. 11, no. 2, 2006, pp.123-148.
- [9] S. J. Curry, A. K. Sporer, O. Pugach, R. T. Campbell, S. Emery, "Use of Tobacco Cessation Treatments Among Young Adult Smokers: 2005 National Health Interview Survey", *American Journal of Public Health*, vol. 97, no. 8, 2007.
- [10] S. Eklund, M. Feldman, M. Trombley, and R. Sinha, "Improving the Usability of Open Source Software", Usability Testing of StarOffice, calc., 2001.
- [11] J. Feller and B. Fitzgerald, "A Framework Analysis of the Open Source Software Development Paradigm", *Proceedings of the International Conference on Information Systems*, Association for Information Systems, 2005, pp. 58-69.
- [12] B. Fitzgerald, "The Transformation of Open Source Software", *MIS Quarterly*, vol 30, no. 3, 2006, pp. 587-598.
- [13] N. Frishberg, A.M. Dirks, C. Benson, S. Nickell, and S. Smith, "Getting to Know You: Open Source Development Meets Usability", CHI '02 extended abstracts, ACM Press, 2002, pp. 932-933.
- [14] M. J. Gallivan, "Striking a Balance between Trust and Control in a Virtual Organization: A Content Analysis of Open Source Software Case Studies", *Information Systems Journal*, vol. 11, no. 4, 2001, pp. 277-304.
- [15] D. G. Gance, J. Kerr, and A. Reid, "Factors affecting the use of open source software in tertiary education institutions", *First Monday*, vol. 9, no. 2, 2004.
- [16] A. Hars and S. Ou, "Working for free?: motivations of participating in open source projects", *Proceedings of the*

34th Annual Hawaii International Conference on Systems Sciences, 2001.

[17] W. W. Hauch and A. Donner, “Wald’s test as applied to hypotheses in logit analysis”, *Journal of the American Statistical Association*, vol. 72, 1977, pp. 851-853.

[18] G. Hertel, S. Niedner, and S. Herrmann, “Motivation of software developers in Open Source projects: an Internet-based survey of contributors to the Linux kernel”, *Research Policy*, vol. 32, 2003, pp. 1159–1177.

[19] E. von Hippel, “Innovation by User Communities: Learning from Open Source Software”, *Sloan Management Review*, 2001.

[20] D. W. Hosmer and S. Lemeshow, *Applied Logistic Regression, 2nd ed.*, Chichester Wiley, New York, 2004.

[21] J. Howison and K. Crowston, “The perils and pitfalls of mining SourceForge”, *In Proceedings of Mining Software Repositories Workshop, International Conference on Software Engineering (ICSE 2004)*, Edinburgh, Scotland, 2004.

[22] A. Khalak, “Economic Model for Impact of Open Source Software”, working paper, Sloan School of Management, Massachusetts Institute of Technology, 2000.

[23] S. Krishnamurthy, “Cave or Community?: An Empirical Examination of 100 Mature Open Source Projects”, *First Monday*, vol. 7, no. 6, 2002.

[24] G.von Krogh, S. Spaeth, and K. R. Lakhani, “Community, Joining, and Specialization in Open Source Software Innovation: A Case Study”, *Research Policy*, vol. 32, no. 7, 2003, pp. 1217-1242.

[25] K. R. Lakhani and E.V. Hippel, “How Open Source Software Works: “Free” User-To-User Assistance”, *Research Policy*, vol. 32, 2003, pp. 923–943.

[26] J. Lerner, and J. Tirole, “Some Simple Economics of Open Source”, *Journal of Industrial Economics*, vol. 52, 2002.

[27] J. Lerner and J. Tirole, “The Scope of Open Source Licensing”, *Journal of Law, Economics, and Organization*, Vol. 21, No. 1, pp. 20-56, 2005

[28] M. Levesque, “Fundamental issues with open source software development”, *First Monday*, vol. 9, no. 4, 2004.

[29] Y. Li, C. Tan, H. Teo, and A. Siow, “A human capital perspective of organizational intention to adopt open source software”, *Proceedings of the International Conference on Information Systems*, Association for Information Systems, 2005.

[30] M. Michlmayr, “Software Process Maturity and the Success of Free Software Projects”, *Software Engineering: Evolution and Emerging Technologies*, vol. 3, no. 14, 2005.

[31] A. Mockus, R. T. Fielding, and J. D. Herbsleb, “Two Case Studies of Open Source Software Development: Apache and Mozilla”, *ACM Transactions on Software Engineering and Methodology*, vol. 11, no. 3, July 2002, pp. 309–346.

[32] J. Y. Moon and L. Sproull, “Essence of Distributed Work: The Case of the Linux Kernel”, *First Monday*, vol. 5, no. 11, 2000.

[33] D. M. Nichols and M. B. Twidale, “The Usability of Open Source Software”, *First Monday*, vol. 8, no. 1, 2003.

[34] E. S. Raymond, “The Cathedral and the Bazaar”, *First Monday*, vol. 3, no. 3, 1998.

[35] E. S. Raymond, “Homesteading the Noosphere”, *First Monday*, vol. 3, no. 10, 1998.

[36] R. Sen, A Strategic Analysis of Competition between Open Source and Proprietary Software”, *Journal of Management Information Systems*, vol. 24, n. 1, Summer 2007, pp. 233–257.

[37] W. Scacchi, “Understanding the requirements for developing Open Source Software Systems”, *IEE Proceedings Software*, vol. 149, no. 1, 2002, pp. 24–39.

[38] R. Stallman, “The GNU Manifesto”, 1985, (available online at <http://www.gnu.org/gnu/manifesto.html>).

[39] K. Stewart and S. Gosain, “The impact of ideology on effectiveness in open source software development teams”, *Management Information Systems Quarterly*, vol. 30, no. 2, 2006, pp. 291–314.

[40] P. Trudelle, “Shall We Dance? Ten Lessons Learned from Netscape’s Flirtation with Open Source UI Development”, *The Open Source Meets Usability Workshop, Conference on Human Factors in Computer Systems*, 2002.

[41] R. T. Watson, D. Wynn, and M. Boudreau, “The Evolution of Professional Open Source Software”, *MIS Quarterly Executive*, vol 4, no. 3, 2005, pp. 329–341.

[42] R. T. Watson, M. Boudreau, P. T. York, M. E. Greiner, and D. Wynn, “The business of open source software”, *Communications of the ACM*, vol. 51, no. 4, 2008.