A Preliminary Analysis of the Influences of Licensing and Organizational Sponsorship on Success in Open Source Projects

Katherine J. Stewart  
University of Maryland  
kstewart@rhsmith.umd.edu

Anthony P. Ammeter  
University of Mississippi  
tammeter@bus.olemiss.edu

Likoebe M. Maruping  
University of Maryland  
lmarupin@rhsmith.umd.edu

Abstract

This paper develops and tests a model of the impact of licensing restrictiveness and organizational sponsorship on the popularity and vitality of open source software (OSS) development projects. Using data gathered from Freshmeat.net and OSS project homepages the main conclusions derived from the analysis are that organizational sponsorship has a positive effect on project popularity by easing user concerns about cost and quality and that license restrictiveness may have a negative effect on popularity by reducing the perceived utility of open source software. Theoretical and practical implications are discussed, and the paper outlines several avenues for future research.

1. Introduction

Why do some OSS projects succeed and others fail? Answering this question is important because both private and government organizations are becoming more dependent on OSS (cf. [4, 19, 31]), because OSS represents a competing model for software development that proponents argue has the potential to create better software cheaper than closed development models (cf. [28]), and because doing so could help predict the possible effects of legal and policy decisions, such as those posed in the SCO-IBM lawsuit (see [27] or www.groklaw.net) on the survival of OSS projects. To address the research question, this paper focuses on developing and testing theory regarding two factors that have been argued to be crucial in determining OSS success: developer motivation and user utility.

The motivation of OSS developers has been the subject of much discussion amongst both researchers and professionals concerned with OSS (cf. [14, 17, 18, 21]). The essential argument developed in this paper is that for a project to be successful it must attract the input of developers, and to do that, characteristics of the project must be aligned with developer motivations.

A subject of similar interest has been the benefits and drawbacks of using OSS software, and these discussions have generally focused on the overall utility of OSS to users, including some consideration of costs, quality, and support concerns (cf. [24, 31]). Based on these discussions, this paper develops the argument that to be successful, characteristics of OSS projects must signal to potential users that the project is likely to provide a high level of overall utility.

Prior work has suggested several project characteristics that may be important to OSS success including project age, project development status, programming language, intended audience, reputation of participants, the type of software developed by the project, licensing issues, and organizational involvement in the project ([5, 7]). This research focuses on license choice (i.e., how restrictive is the license) and organizational sponsorship (i.e., is the project affiliated with an organization) as antecedents to success. These two factors are examined for three reasons. First, these factors are tied to perceptions that existing theoretical perspectives imply will influence the success outcomes of interest, in particular user perceptions of cost and quality and developer perceptions of the outcomes they are likely to experience as a result of working on the project. Second, prior work on OSS implies but has not tested the importance of licensing and organizational sponsorship in influencing developers and users ([21]), and current debates about OSS center around issues of ownership, which are closely related to licensing and organizational sponsorship (e.g., [27]). Third, license choice and organizational affiliation may be within the control of project leaders or administrators and understanding their impact may therefore be of practical usefulness.

The remainder of the paper is organized as follows. The next section develops a more detailed definition of what OSS software is and what success means in the context of OSS development. This is followed by the development of the theoretical basis for hypotheses regarding how and why license restrictiveness and organizational sponsorship influence developer motivations and user perceptions of software utility thereby leading to greater or lesser levels of success. The final section before the discussion of methods considers the inter-relatedness of our two success outcomes. The methods section describes the operationalization of the constructs using data collected at two points in time from two sources: Freshmeat.net (a site that hosts several thousand OSS projects) and the homepages of projects in the sample. The analysis and results section describes the outcomes of hypothesis tests, and the discussion section explains how the results support and extend the
2.0 Defining features of OSS software, development processes, and success

There is sometimes confusion among those not intimately familiar with OSS as to exactly what qualifies as OSS and what does not. It is not simply a matter of access to source code, else software obtained under Microsoft’s shared source initiative might be considered open source, and it most definitely is not, according to OSS leaders ([33]). Nor is it a matter of the software being developed by volunteers, as Linux is developed by individual volunteers as well as individuals paid by their employers to work on it. Perhaps the simplest way to distinguish what is open source software from what is not open source software is by the definition of the Open Source Initiative (OSI). Software is open source if it is released under a license approved by the OSI. OSI standards for open source software licenses include the following provisions: the source code must be available at little or no charge; redistribution of the program, in source code or other form, must be allowed without fee; distribution of modified versions must be allowed without discrimination against groups of users or types of uses; and distributions of modified versions must be allowed on the same terms as the original program.

While we take the position here that licensing is the crucial feature to distinguish OSS from other software, it is important to recognize that OSS often has other unique features as well. OSS development is different from traditional software development in that it often depends on volunteer labor pools coordinating their effort without a common organizational superstructure, and the end product is generally provided freely ([13]). Because of these differences, the success indicators often applied to commercial software projects – e.g., being on-time, on-budget, and meeting specifications – may not be readily applied in the OSS setting. In this setting, there often is no a priori budget, timeline, or set of specifications ([29]). Nonetheless, there are many ways to define success in the context of OSS development (cf. [6]).

Market penetration is often cited when discussing the success of well-known open source software, such as Linux and Apache (e.g., [13]). A similar indicator for less well-known projects is the popularity of the project among potential users, as indicated by the level of interest displayed in the project (e.g., traffic on the project website, downloads of the code, and the number of people who monitor announcements and new releases regarding a project are some indicators of user interest). A second type of success relevant in this setting is the success of the project in attracting interest and input from the development community. Since open source projects rely on voluntary input, attracting and motivating contributors is a key factor in project success, and one way of assessing that success is by looking at the level of activity on a project – e.g., how frequently are bugs fixed, support requests answered, or new releases of the software posted. We refer to this kind of development-oriented success as project vitality. Thus we focus on two types of success in OSS development: project popularity and project vitality. Popularity refers to how much interest and attention the project generates among current and potential users, and vitality refers to the level of activity by developers on the project.

3.0 Antecedents to OSS Success

Our study focuses on two potential antecedents to project success: the restrictiveness of the license chosen and organizational sponsorship. License choice refers to which of the many available OSI certified licenses a project uses, and restrictiveness will be explained below. Organizational sponsorship refers to the fact that some but not all OSS projects are affiliated with a formal organization such as a for-profit company or a University. We use the term organizational sponsorship to indicate a publicly displayed affiliation between an OSS project and an organization.

In the next section we focus on popularity as an outcome. To understand project popularity, we must consider why users may choose one software solution over another. We thus focus our hypothesis development around how license choice and organizational sponsorship may impact user perceptions that prior theoretical work has shown are critical to such choices. After developing hypotheses regarding popularity, the next section focuses on how licensing and sponsorship may influence vitality. As project vitality depends on the input of developers, we develop hypotheses regarding vitality based on the emerging research on OSS developer motivation.

3.1 Means by which License and Sponsorship may affect the popularity of OSS Projects

As popularity is a function of user interest in and adoption of OSS, many theoretical perspectives may be
brought to bear in considering how license choice and organizational sponsorship may influence popularity including work on consumer behavior (e.g., [40]), software acquisition (e.g., [15]), and technology acceptance (e.g., [36]). While each of these streams of work focuses on somewhat different antecedents and outcomes, they all argue for the importance of the user’s (or adopter’s or consumer’s) perceptions about the object (in this case OSS) that is under consideration. Further, perceptions that are argued to be important across these different streams of work are very similar. In the next paragraph we summarize these similarities and in the remainder of this section we focus on consumer research to provide the underlying logic for our hypotheses.

Research on consumer behavior attributes customer purchase decisions to perceived value [40]. Customers choose to purchase those products that offer the highest perceived value, which is assessed based on both product cost and product quality ([12]). Similarly, assessments of value discussed in literature on software acquisition (e.g., [15]) are also based on cost, which may include both price and other factors such as estimates of the installation and maintenance requirements, and quality, which may be seen in terms of the extent to which the software meets an organization’s needs and the impact it may have on firm operations. Software acquisition literature tends to focus on organizations as the acquiring entity. The technology acceptance literature focuses on individuals’ use of software, and provides consistent conclusions about important factors. Perceptions of cost and quality are reflected in the TAM literature’s focus on perceived usefulness and perceived ease of use as key determinants of behavioral intention to use and actual use of technology [9, 35, 36]. Davis (1989: 320) defines perceived ease of use as “the degree to which a person believes that using a particular system would be free of effort,” and effort may clearly be seen as a cost associated with use. In their work integrating the Technology Acceptance Model with other similar user acceptance models [36] concluded that perceived usefulness was a relevant indicator of job related performance expectations. Performance expectations refer to a potential user’s impressions of how use of a technology will positively impact his or her productivity at work and might be seen as an aspect of quality.

In keeping with the conclusion, drawn from the theories discussed above, that perceptions of cost and quality are likely to be crucial in determining popularity of an OSS, quality and cost factors have been cited as major concerns among potential open source users. A survey of 260 IT corporate managers [31] indicated that 41% viewed the inability to hold someone responsible for software breakdown – a quality concern – as one of the major reasons that open source software had not been widely adopted in their companies. In this same survey 59% of IT managers cited support concerns – which are dependent on quality and impact costs – as a factor reducing their companies’ usage of OSS.

3.2 Cues to cost and quality

Perceptions of value may be developed from individuals’ use of extrinsic cues in the formation of product quality and cost perceptions [12]. Brand name, price and store name are some of the most widely studied extrinsic cues in the marketing literature [1, 2]. Consumers make product quality inferences based on cues such as brand and store name through a process called affect-referral [40]. Using affect-referral, customers simplify their decision making process by basing their judgments on summary information (i.e., brand attitudes) rather than on product attribute information [39]. This process suggests that individuals do not always use product attribute information when choosing amongst competing products.

Drawing on the concept of affect-referral, organizational sponsorship and license choice can be viewed as salient extrinsic cues for evaluating OSS projects’ products (i.e. their software releases). Although code is available for inspection, users may not have the necessary background knowledge to evaluate the inner workings and features of the software program before they install it, or even if they do have the requisite skill, they may seek to minimize the cognitive effort involved in evaluation by relying instead on more easily interpreted cues. When selecting among different possible OSS (and non-OSS) solutions, individuals may draw conclusions about likely costs and quality based on these cues as described below.

3.3 Impact of OSS License Choice on Popularity
Within the OSI framework, there is room for variance across many dimensions of a license. One such dimension that others [21, 22, 30] have suggested as important is the restrictiveness of the license. For example, the most widely used OSI license, the GNU General Public License (GPL), has two restrictions that many other popular licenses (e.g. the BSD) lack: it requires that modified versions of the software also be open (often referred to as a “copyleft” provision) and it requires that the code be combined only with other programs distributed under licenses that share the first requirement (often referred to as a “viral” provision). Below, we use the term restrictive to refer to licenses that have both of these provisions and non-restrictive to refer to licenses that do not (such licenses may have the first provision but not the second).

While requirements to open modified versions and combine the software only with similarly licensed software act to maintain the ‘openness’ of the code, they are restrictive in the sense that they limit what a user can do with the software. These restrictions may constrain commercialization of OSS applications [8, 21, 38]. In doing so, they may significantly reduce the perceived usefulness of the software among one category of potential users – those who are seeking to advance commercial interests. Perceived usefulness may be further reduced by restrictive licenses in that such licenses limit potential users’ ability to employ the code in conjunction with software that is distributed under a less restrictive license. For example, someone working on a project using a non-restrictive license would not be able to use software with a restricted license, which, if it did not have such a license, might otherwise have been combined in a synergistic fashion with the non-restricted software.

Hypothesis 1: OSS projects that use a non-restrictive license will become more popular over time than those that use a restrictive license.

3.4 Impact of OSS Sponsorship on Popularity.

Building on the above discussion of cues, we suggest that organizational sponsorship may also be used as a form of summary information for potential adopters making a decision on whether to use a particular software product. At the most basic level, projects may be distinguished by either having a sponsor or not. The majority of projects hosted on sites like SourceForge have no apparent organizational sponsor, but rather are created and maintained by private individuals. However, many different kinds of organizations may be affiliated with an OSS project. For example, TaskGuide Viewer, an XML-based tool for creating wizards, is an IBM-sponsored open source software development project, and C-Kermit is a communication utility that is sponsored by Columbia University.

Organizational sponsorship information may be analogous to brand or store name in terms of its use as an extrinsic cue. Sponsorship information, like brand or store information, may enable potential users to make judgments about the quality of OSS without experientially evaluating it. Perceived costs associated with using software do not only include price (which for most OSS is close to zero), but also such factors as maintenance time and effort expended in using the product in the future. Potential users’ perceptions of software quality may be shaped by a sponsoring entity based on the users’ expectations of the sponsor’s ability to meet conformance (a software vendor’s ability to deliver the right product), service (the vendor’s ability to customize the product to specific user needs), and innovation (the vendor’s ability to provide continuous feature enhancements through future upgrades) requirements in the future ([26]).

Organizational sponsorship may imply the availability of technical support, upgrades, and other resources that may be needed over the long term by consumers of software products. A greater degree of uncertainty may exist regarding the availability of such services for non-sponsored OSS projects. Others ([32]) have argued that consumers make judgments about uncertainties and the potential future losses that could stem from such uncertainties, and a recent survey of IT managers demonstrates that this is applicable in the case of OSS ([31]). Hence, organizational sponsorship may act as a cue to the likely future costs and quality associated with an OSS product.

Hypothesis 2: Sponsored OSS projects will become more popular over time than non-sponsored OSS projects.

3.5 Antecedents to Vitality in OSS Projects

While popularity is an indicator of success among OSS users, vitality is an indicator of success among OSS developers. A project will have a high level of vitality if developers devote their time to making contributions and developing enhanced software versions. Contrary to some popular conceptions of OSS development as drawing from an infinite pool of talent (e.g., [34]), OSS work requires specific skills and there is a limited pool of people with the knowledge to be able to productively contribute, leading to potential competition among projects to attract developer efforts. For example, [20] cites the Orbiten Free Software Survey ([16]), which indicates that the 100 most prolific OSS contributors contribute to 1,886 distinct projects, a contributor to project ratio of approximately 1 to 19. Other analyses
such that more vital projects are also more popular. We may expect vitality and popularity to be correlated, as the product they develop, therefore, at any point in time, may experience higher levels of vitality than those using a non-restrictive license.

Hypothesis 3: OSS projects using a restrictive license will experience higher levels of vitality than those using a non-restrictive license.

3.7 The Relationship between Vitality and Popularity

[25] point out that OSS developers are often users of the product they develop, therefore, at any point in time, we may expect vitality and popularity to be correlated such that more vital projects are also more popular. [20] provides some evidence of such a relationship, reporting significant correlations between number of developers and page views and downloads. However, developers and users are not completely overlapping sets. [25] suggest that successful projects require many more users than developers. Based on our discussion of developer motivation above, we suggest that popularity will influence vitality over time. The more popular a project is, the wider the audience for individual contributions and therefore the more visible the efforts of contributors. Hence there may be greater potential reputation benefits from working on more popular projects ([22]), and we might expect such projects to attract more activity from developers. Further, an active user base will generate defect reports and support requests ([25]), providing greater opportunities for developers to hone their skills on a variety of tasks and thereby stimulating more development work.

Hypothesis 4: OSS project popularity will have a positive effect on OSS project vitality.

4.0 Methods

Researchers have examined OSS from different levels of analysis including viewing OSS as a phenomenon at the community (e.g., [3]) organization (e.g., [23]) and team or group levels (e.g., [25]). The focus of this research is limited to understanding OSS success at the project level. Thus dimensions of and antecedents to success discussed above have been conceptualized at the level of the project itself and all measures discussed below are similarly at the project level. Publicly available data on open source projects registered on the Freshmeat website (www.freshmeat.net) was used to test the hypotheses. Data was collected from each project’s Freshmeat website at the start and end of an eight month period (March - December 2002).

4.1 Sample

We used a stratified random sampling technique to select projects to be included in the study. We first selected three project categories from which to draw our sample. These were utilities, software development, and games. These categories were chosen to represent different kinds of software and because they contained large numbers of projects. To be eligible to be included in our sample, a project had to list exactly one of these three categories. Within these categories we further differentiated between new projects, which had been registered on the site within the two weeks prior to our first data collection point and older projects that had been registered more than two weeks prior to our initial data registration.
collection. This stratification was done to ensure ample variance across projects in terms of their development stage and to minimize the possibility of selecting only well-established, successful projects for analysis. The selection of three project categories and the distinction between older and newer projects created six distinct groups across which we randomly selected a total of 218 projects. Of these, forty-nine projects disappeared from the Freshmeat website during the period of our data collection and therefore could not be included in the analysis. Similarly, because some projects may list on Freshmeat as a means of advertising but do not provide releases through Freshmet, we checked project homepages for any release history of projects that had zero releases posted on Freshmeat during our observation period and removed these projects from the sample. The final sample size for analysis was 147 projects.

4.2 Dependent Variables

We measured project popularity using the total number of subscribers associated with a project, as reported on Freshmeat.net. A subscriber on Freshmeat is someone who has registered to receive email announcements about a project (such as notices of new releases). The identity of subscribers is not available to project administrators or developers; the subscriber list is separate from development email groups. Because we are concerned with how our independent variables influence popularity over time, we used the change in subscribers over the period of our observation as the dependent variable. This was calculated as the number of subscribers at the second observation point minus the number of subscribers at the first observation point. We counted the number of new releases of the project software on Freshmeat.net over the observation period as an indicator of project vitality. Square root transformations were performed on both dependent variables to reduce skewness and better approximate a normal distribution.

4.3 Independent Variables

The independent variables in our model included license restrictiveness (H1, H3), whether or not a project was sponsored (H2), and initial popularity (H4). Sponsorship was coded based on the description of the project provided on Freshmeat and by visiting the project home page. A project was categorized as having a sponsor if it stated an affiliation with an organization (e.g., a company or university) in the project description or if the project homepage was hosted by an organization. Projects that neither stated an affiliation with an organization nor maintained project pages on an organization’s website were categorized as non-sponsored. This included projects that were hosted on SourceForge or other OSS community websites and those that were hosted on individual’s websites.

A graduate assistant who was not informed of the hypotheses coded the sponsorship for all projects. To assess the reliability of the coding, the first author later classified a randomly chosen subset of 18 of the projects in the sample using the same classification criteria described above. The level of agreement between the coders was .89 and in the two cases of disagreement discussion among the two coders and the second author determined that the original coding was justified.

The license associated with each project is reported on Freshmeat.net. We used the definitions and categorization provided by ([21]) to determine whether a project had a non-restrictive or a restrictive license. Ninety of the projects in the final sample used the GNU GPL license and were thus classified as non-restrictive. The remaining 57 used other licenses that did not include a viral provision and were thus classified as non-restrictive. These percentages seem to reflect well the overall distribution of license use in the population of OSS projects as reported by [21]) (in their sample, 69% used the GNU GPL).

We recorded the number of subscribers that were associated with each project at the first measurement point (initial subscribers) as our measure of initial popularity to allow testing of hypothesis 4.

4.4 Control Variables

Project category information for each project is self-reported on Freshmeat.net. As noted above, we included three project categories in our sample: software development, games, and utilities. Because projects in these categories could systematically differ in ways important to our dependent variables (e.g., each category may draw a different kind of user), we controlled for project category in our analysis.

The age of a project may serve as a proxy for several factors that could be important to success including the experience of the development group in working together, the entrenchment of the software in the user community, and the development status of the project. Thus age, measured as the number of days each project had been registered on the Freshmeat website at our second observation point, was used as a control variable. Development status itself was not used as a control variable because many of the projects in the sample did not list a development status. In addition to serving as an independent variable predicting new releases, the initial subscribers also served as a control variable for the change in subscribers over the observation period because projects with larger initial subscriber bases might be
expected to experience greater positive effects of word of mouth recommendations.

5.0 Analysis and Results

Our data includes both categorical (license restrictiveness, sponsorship, and project category) and continuous (project age, initial subscribers) independent and control variables, and an expected correlation between the two continuous dependent variables (change in subscribers and new releases). We therefore used multivariate analysis of covariance (MANCOVA) to test the hypotheses. Table 1 presents the results of the MANCOVA.
Table 1 MANCOVA Predicting Popularity and Vitality

<table>
<thead>
<tr>
<th>Variable</th>
<th>Multivariate F</th>
<th>df</th>
<th>Univariate F</th>
<th>df</th>
<th>Popularity</th>
<th>Vitality (df)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project Age</td>
<td>6.34**</td>
<td>2</td>
<td>9.67**</td>
<td>9.51**</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Project Category</td>
<td>1.06</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sponsorship</td>
<td>3.88*</td>
<td>2</td>
<td>7.71**</td>
<td>2.81†</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>License</td>
<td>2.85†</td>
<td>2</td>
<td>3.05†</td>
<td>0.29(1)</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Initial Subscriber Base</td>
<td>30.89***</td>
<td>2</td>
<td>62.09***</td>
<td>12.74***</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Sponsorship X Project Category</td>
<td>1.86</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>License X Project Category</td>
<td>1.47</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sponsorship X License</td>
<td>1.88</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sponsorship X License X Project Category</td>
<td>3.60**</td>
<td>4</td>
<td>7.51***</td>
<td>1.12(2)</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>

As table 1 illustrates, the MANCOVA yielded a significant multivariate main effects for sponsorship (p < .001) and a marginally significant effect for license restrictiveness (p = .06) on change in subscribers, and the model explained 51% of the variance. Hypothesis 1 suggested that projects with a non-restrictive license would experience greater increases in subscribers than projects with a restrictive license. The univariate F-tests revealed a marginally significant main effect for license restrictiveness on change in subscribers (p = .08) with the difference in the expected direction, lending some support to H1. The univariate F-test for the effect of sponsorship on change in subscribers was significant (p < .01) and in the expected direction, supporting H2, which suggested that sponsored projects would become more popular than non-sponsored projects.

Hypothesis 3 suggested that projects using a restrictive license would experience greater vitality over time than projects using a non-restrictive license. The MANCOVA explained 26% of the variance in new releases, our measure of project vitality. However, the univariate effect for license restrictiveness was not significant therefore hypothesis 3 was not supported.

Hypothesis 4 predicted that higher levels of subscriber interest in a project would generate greater developer attention and activity on the project over time. The MANCOVA shows a significant main effect for initial subscribers (p < .001) on new releases, supporting hypothesis four.

Finally, there was a significant 3-way interaction effect on change in subscribers (p < .001). However, closer inspection of the data revealed one cell contained a single project. We re-ran the analysis excluding the single observation, and the 3-way interaction was no longer significant, thus we concluded that the effect was spurious and we do not seek to interpret it further below.

6.0 Discussion

The goal of this research was to explore the antecedents to success in OSS projects. We developed and tested hypotheses regarding the effects of two important OSS project characteristics, license restrictiveness and organizational sponsorship, on two key outcomes in OSS projects, project vitality and the popularity of the project among users and potential users. Results, summarized in Table 2, were generally supportive of the arguments made in the paper, indicating that licensing decisions and organizational affiliations are crucial to OSS project outcomes both in terms of harnessing the efforts of the development community (the model explained 26% of the variance in new releases, our measure of project vitality) and attracting the interest of the user community (the model explained 51% of the variance in the change in subscribers, our measure of popularity).

Table 2 Results of Hypothesis Tests

<table>
<thead>
<tr>
<th>Hypothesis</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hypothesis 1: OSS projects that use a non-restrictive license will become more popular over time than those that use a restrictive license.</td>
<td>Marginally supported</td>
</tr>
<tr>
<td>Hypothesis 2: Sponsored OSS projects will become more popular over time than non-sponsored OSS projects.</td>
<td>Supported</td>
</tr>
<tr>
<td>Hypothesis 3: OSS projects using a restrictive license will experience higher levels of vitality than those using a non-restrictive license</td>
<td>Not supported</td>
</tr>
<tr>
<td>Hypothesis 4: OSS project popularity will have a positive effect on OSS project vitality.</td>
<td>Supported</td>
</tr>
</tbody>
</table>

6.1 Theoretical Implications, Limitations, and Future Research

The empirical results generally supported the theoretical reasoning put forth suggesting that potential users of OSS projects are influenced by the type of
license under which software is released and whether the software project is associated with an organizational sponsor. Specifically, users were shown to prefer software projects that had an affiliation with a formal organization, supporting our arguments that such affiliations signal to users the availability of technical support and other desirable software characteristics. Similarly, results indicated users preferred software with less restrictive licenses. This supported our reasoning that users see such restrictions as limiting the potential future usefulness of the software, though the result must be interpreted with caution given the borderline statistical significance of the test. The effect of initial popularity on vitality lent strong support to our theoretical argument that having an audience enhances developers’ motivation to work on a project and thereby increases vitality.

Altogether, the results of the preliminary study described here indicate that developer motivation and user utility are promising avenues for understanding the means by which project characteristics may influence success in OSS projects. While the model tested in this paper explained significant percentages of the variance in project vitality and popularity, a more fine-grained consideration of project characteristics may increase the predictive power, particularly for vitality. For example, considering characteristics of licenses beyond just restrictiveness, or considering more than two levels of restrictiveness may enhance understanding of how licensing choices influence developers and users. The relatively constrained variance in our measure of restrictiveness could be one reason that no effect of license choice on vitality was found. Similarly, a more fine-grained approach to categorizing organizational sponsors seems like a promising avenue to pursue based on the results uncovered in this study.

In addition to the constraints imposed on this study by a relatively coarse-grained measurement of project characteristics, there are important limitations that should be addressed in future work in order to rule out possible alternate explanations for findings and generate a more detailed understanding of the OSS phenomenon. Perhaps the most significant limitation of the current study is that we were not able to measure the suggested mediating mechanisms, hence it is possible that there are other factors responsible for the effects we observe. In particular, it is impossible to empirically distinguish between restrictiveness and familiarity of the GPL as the cause of the effect for licensing. As one of the oldest and most widely used open source licenses, the GPL could have high name recognition, which could impact the appeal of software among users and developers. However, there are several reasons why we believe restrictiveness is the more compelling explanation. First, we find restrictiveness the more compelling explanation because it is more directly related to the incentives of all users, who may or may not be familiar with the GPL based on its relatively long history. In other words, the institutional legitimacy of the GPL may have effects on a subset of users who have a long history of involvement with OSS and are therefore aware of that history and familiar with the GPL. However, the wider audience of potential OSS users does not have that experience and may only have become interested in OSS recently, at a time when many licenses are widely used. Similarly, all licenses employed in our sample had the legitimacy granted by OSI certification, further diluting possible perceived differences of the GPL versus others. Finally, the non-restrictive license category included the LGPL license, which would share some of the “name recognition” of the GPL and the BSD, which also has a long history and should therefore share some of the benefits of familiarity. Nonetheless, future research should seek to collect more subjective data to confirm the importance of the theoretical mechanisms that are argued to cause the effects found in this study.

Other important limitations of the study impact the generalizability of our conclusions. We limited our sample to three software categories, thus our ability to generalize across all categories of software is constrained. While we see no reason that results should not generalize to other categories of software, this remains an empirical question. Similarly, our selection of projects that had registered on Freshmeat places a constraint on generalizability. Such projects could differ in important ways from OSS projects that are not registered on Freshmeat. For example, because Freshmeat is targeted at a user population, the administrators of these projects may, on the whole, be more motivated to popularize their software than the administrators of non-Freshmeat projects. These limitations on generalizability may be addressed in future work by expanding the sample of projects studied.

6.2 Practical implications

The independent variables in this study were selected in part because of their practical relevance. Licensing and organizational affiliations are factors that are under the control of the individuals or organizations that start and run OSS projects. Results imply that both users and developers are influenced by these project characteristics. It is to the benefit of project administrators to recognize this fact, and it may be possible for project administrators to either enhance or reduce the effects observed in the study by communicating to user and developer audiences the reasons for their decisions regarding license choice or sponsorship arrangements rather than allowing these cues to speak for themselves.

Overall, sponsorship generally had a positive effect on the projects in our sample. Perhaps one of the most
interesting practical implications of these results is that, in contrast to some popular conceptions of volunteers banding together to create viable competitive commercial software alternatives, some organizational involvement may be crucial to move OSS into the realms of success experienced by products such as Linux. While further research is needed to confirm this suggestion, if it is supported there may be important policy implications for fostering the development of OSS.

6.3 Conclusion

The main contribution of this research has been in developing and empirically testing a theoretical basis for the effects of key project characteristics – license choice and organizational sponsorship – on two different kinds of OSS project success – attracting developer input and attracting user attention to a project. The research also complements prior work by providing an analysis of smaller, younger projects than have been the focus of most prior studies, by studying these projects over time rather than relying on a cross-sectional snapshot, and by going beyond the SourceForge data, which has been the basis of the most closely related prior OSS studies on success (cf. [5, 7, 21]). Taken as a whole, the results support an interesting theoretical picture of the positive and negative effects of licensing and organizational sponsorship on OSS project success, and while the basic premises put forth in the paper are supported the results also serve to highlight many avenues for future research to expand our understanding of the unique context of open source software development.

7.0 References


