The Journal of Systems and Software 85 (2012) 895–905

Contents lists available at SciVerse ScienceDirect

The Journal of Systems and Software

journal homepage: www.elsevier.com/locate/jss

Factors affecting the success of Open Source Software

Vishal Midha [a,*], Prashant Palvia [b,1]

[a] The University of Texas – Pan American, 1201 W. University Drive, Edinburg, TX 78539, United States
[b] The University of North Carolina at Greensboro, 1000 Spring Garden St, Greensboro, NC 27402, United States

ARTICLE INFO

Article history:
Received 20 July 2010
Received in revised form 3 November 2011
Accepted 8 November 2011
Available online 17 November 2011

Keywords:
Open Source Software
System success
Longitudinal effects
Intrinsic cues
Extrinsic cues

ABSTRACT

With the rapid rise in the use of Open Source Software (OSS) in all types of applications, it is important to know which factors can lead to OSS success. OSS projects evolve and transform over time; therefore success must be examined longitudinally over a period of time. In this research, we examine two measures of project success: project popularity and developer activity, of 283 OSS projects over a span of 3 years, in order to observe changes over time. A comprehensive research model of OSS success is developed which includes both extrinsic and intrinsic attributes. Results show that while many of the hypothesized relationships are supported, there were marked differences in some of the relationships at different points in time lending support to the notion that different factors need to be emphasized as the OSS project unfolds over time.

1. Introduction

Over the last two decades, the practice of community based software development and distribution, combined with novel uses of intellectual property law, has emerged. This practice has come to be known as "Open Source Software" (OSS) development. The success of community based model of software development has extended this paradigm into new arenas such as pharmaceutical development, knowledge repositories, space exploration, and education as well as to unexpected domains such as developing recipes for open-cola, coffee and beer (Lawton, 2002). The hype of open source phenomenon has also led to an exponential growth in the number of such projects. However, except for a few, the majority of the projects at open source hosting sites have been unsuccessful (Chengalur-Smith and Sidorova, 2003). This apparent paradox has generated immense interest among open source researchers in identifying the factors that affect open source project success.

A typical open source project starts when an individual (or group) feels the need for a new feature or entirely new software, and someone in that group, eventually writes one. In order to share it with others who have similar needs, the software is released under a license that allows the community not only to use it but also to see the source code and modify it to meet local needs and improve the product by fixing bugs (Crowston et al., 2006; Feller and Fitzgerald, 2002; Midha et al., 2010). Making software available widely on an open network, e.g., the Internet, allows developers around the world to contribute code, add new features, improve the present code, report bugs, and submit fixes to the current version. The developers of the project incorporate the features and fixes into the main source code and a new version of the software is made available to the public. This process of improvement and customization through code contribution and bug fixing is continued in an iterative manner (Midha et al., 2010). Clearly, as OSS projects are continually in development, the project characteristics change through the life of the project as different versions are released. In addition, even the interests of its user and developer base can change. It is, therefore, important to understand the changing dynamics of the relationships between project success measures and their antecedents over time.

The purpose of this research is to understand the impact of various factors, categorized as intrinsic and extrinsic factors, on OSS project success over the first three years of its life. A longitudinal analysis of these factors is conducted at various stages in the OSS life cycle. This analysis provides unique insights into various project management decisions, such as the escalation of errors and design flaws leading to substantial maintenance costs. As a result, OSS project administrators can develop a deeper understanding of the metrics for open source development that can help them to manage the escalation of costly flaws in these projects. Ultimately, the understanding of the antecedents could lead to a project's success both in terms of market penetration and technical achievements over time.

It may be noted that some of the relationships in this research, such as impact of complexity on technical success (Feller and Fitzgerald, 2002), and impact of license type on market success

* Corresponding author. Tel.: +1 956 381 2801; fax: +1 956 381 3367.
E-mail addresses: vmidha@utpa.edu (V. Midha), pcpalvia@uncg.edu (P. Palvia).
[1] Tel.: +1 336 334 4818; fax: +1 336 334 4550.

0164-1212/$ – see front matter © 2011 Elsevier Inc. All rights reserved.
doi:10.1016/j.jss.2011.11.010

(Stewart et al., 2005), have already been investigated in the OSS literature. They are, however, still considered in this study for two reasons: (i) completeness of the model, and (ii) to study the impact of these variables in a longitudinal fashion, which to the best of our knowledge has received very limited attention. One example is that of Subramaniam et al. (2009) who modeled time-variant and time-invariant variables to study their impact over time. Their study could be considered the first quantitative longitudinal study to investigate factors that impact the success of OSS projects. Our study builds on Subramaniam et al's (2009) work to propose a comprehensive model by including various other factors, such as complexity, modularity, responsibility assignment, and language translations.

The remainder of the article is organized as follows: the next section presents theoretical foundations leading to the hypotheses and model development. It is followed by a description of methods and measures. The following sections present the evaluation of the model and discussion of the results. The article concludes by acknowledging its limitations and highlighting its contributions to both research and practice.

2. Theory and research hypotheses

2.1. OSS success

The meaning of success in the software development context is subjective. Some researchers define success of OSS projects in terms of the size of consumer base, whereas others have defined it as the number of free contributions from its developer community (Feller and Fitzgerald, 2002; Stewart et al., 2005; Crowston et al., 2006). For example, Raymond (2001) points that OSS projects success is characterized by a continuing process of volunteer developers fixing bugs, adding features and releasing software "often and early". Supporting Raymond's definition, Markus et al. (2000) suggested that it is critical to attract contributors on an on-going basis to keep the project sustainable. English and Schweik (2007) grouped projects in successful and failures based on initiation and growth in terms of number of releases and downloads. Feller and Fitzgerald (2002) observed that, for well known open source projects such as Linux and Apache, market penetration could be used as a success measure. However, this measure is not applicable to less known projects. For lesser known projects, success is indicated by the popularity of the project among its current and potential users (Stewart et al., 2005).

In congruence with Crowston et al. (2006) suggestion that success is a multi-dimensional construct that needs to be assessed from multiple perspectives, Grewal et al. (2006) pointed out that measuring success of OSS projects in terms of technical achievements or market success represents an incomplete picture of success. They further pointed that a comprehensive picture of OSS success should encompass both the developers' technical achievements as well as indicators of market success. This pair of criteria for project success is consistent with the literature on software success (Rai et al., 2002) and new product development (Mansfield and Wagner, 1975). Following these recommendations, we focus our attention on both commercial and technical measures of success. '*Market Success*' of an OSS project, a measure of project popularity, is defined as the level of interest displayed in the project by its consumers, whereas '*Technical Success*' is defined in terms of developer activity, i.e., the level of effort expended by developers of the project.

As there are over 300,000 OSS projects listed on SourceForge alone (October 2011), then the question arises: which projects achieve market and technical success? What factors govern the two success criteria? These need to be evaluated from the consumer's point of view. There are two types of consumers: users and developers (users simply use OSS products, whereas developers use and contribute to the development of OSS products). More specifically, (i) which projects users display interest in, (ii) which projects OSS developers contribute to, and (iii) as most OSS projects are continually in development, implying that the project characteristics change over time, then how does this impact OSS users affinity and OSS developers activity in the project. As OSS users and developers have different motivations to be associated with OSS projects, we argue that two different sets of factors exist that impact the two consumer types. We borrow the concepts of intrinsic and extrinsic cues from Cue Utilization Theory (CUT) to explain the two sets of factors, and how they impact the success of OSS projects.

2.2. Cue utilization theory and Open Source Software

As per cue utilization theory (CUT), products are conceived to consist of several cues. Each cue provides a basis for developing various impressions of the product, and is used by consumers in their product selection decision (Olson, 1972). Cues can be classified as extrinsic or intrinsic. *Extrinsic cues* are product-related attributes which are not part of the physical product. Conversely, *intrinsic cues* represent product-related attributes which cannot be manipulated without also altering physical properties of the product. Intrinsic cues play a major role in the product selection decision-making process. Nevertheless, consumers depend more on extrinsic cues as "surrogates" for risk reduction, as intrinsic cues require more time and effort (Zeithaml, 1988).

Careful evaluation of SourceForge, the biggest OSS repository, and the OSS literature reveal a select yet diverse set of cues which are available to consumers. Following Olson's (1972) definition, the set of cues was grouped into six extrinsic and two intrinsic cues. The product-related attributes which are not part of the physical product, such as license type, available language translations, size of existing user, size of developer base, and responsibility assignment of work were categorized as extrinsic cues; whereas product-related attributes which represent the properties of the product, such as complexity and modularity of the source code were categorized as intrinsic cues.

According to CUT, consumers arrive at judgments by evaluating intrinsic and extrinsic cues. However, Wright (1975) pointed out that consumers tend to simplify their decision making process by basing their judgments on summary information available through external cues, rather than on the total set of project's attributes, including both internal and external cues. This is most likely for choice behavior where consumers prefer to engage in minimal processing. Although, in the case of OSS projects, the complete source code is available for inspection to all consumers, the OSS users may not possess the necessary skills and knowledge to evaluate the intrinsic cues accessible through the source code. Even if they have the necessary skills, they may rely on more easily interpretable extrinsic attributes than spending time and effort on understanding and evaluating the source-code. Consequently, extrinsic cues may form the basis of selection of OSS projects by its consumers.

Conversely, knowledgeable consumers such as OSS developers are less likely to use extrinsic cues alone in their decision making process (McConnell, 1968). Due to their well-developed cognitive structures, experts prefer to use both extrinsic and intrinsic cues to form their decision (Olson, 1972). The OSS developers not only use OSS products, but also contribute to their source code development. Such consumers have the necessary skills and knowledge to evaluate the intrinsic cues of OSS projects. Therefore, both intrinsic and extrinsic cues form the basis of project's technical success. Combining these arguments with H1 (as discussed in the next section) leads to the development of our comprehensive theoretical OSS success model (Fig. 1).