# City Research Online

## City St George's, University of London

**Citation:** Krogh, G. V., Haefliger, S., Spaeth, S. & Wallin, M. W. (2012). Carrots and Rainbows: Motivation and Social Practice in Open Source Software Development. MIS Quarterly, 36(2), pp. 649-676. doi: 10.2307/41703471

This is the published version of the paper.

This version of the publication may differ from the final published version. To cite this item please consult the publisher's version.

**Permanent repository link:** https://openaccess.city.ac.uk/id/eprint/5955/

**Link to published version:** https://doi.org/10.2307/41703471

**Copyright and Reuse:** Copyright and Moral Rights remain with the author(s) and/or copyright holders. Copies of full items can be used for personal research or study, educational, or not-for-profit purposes without prior permission or charge, unless otherwise indicated, provided that the authors, title and full bibliographic details are credited, a hyperlink and/or URL is given for the original metadata page and the content is not changed in any way. For full details of reuse please refer to City Research Online policy.

City Research Online: http://openaccess.city.ac.uk/ publications@citystgeorges.ac.uk

MIS Quarterly

THEORY AND REVIEW

# CARROTS AND RAINBOWS: MOTIVATION AND SOCIAL PRACTICE IN OPEN SOURCE SOFTWARE DEVELOPMENT[1]

**Georg von Krogh, Stefan Haefliger, Sebastian Spaeth, and Martin W. Wallin**
ETH Zurich, SWITZERLAND {gvkrogh@ethz.ch} {shaefliger@ethz.ch} {sspaeth@ethz.ch} {mwallin@ethz.ch}

*Open source software (OSS) is a social and economic phenomenon that raises fundamental questions about the motivations of contributors to information systems development. Some developers are unpaid volunteers who seek to solve their own technical problems, while others create OSS as part of their employment contract. For the past 10 years, a substantial amount of academic work has theorized about and empirically examined developer motivations. We review this work and suggest considering motivation in terms of the values of the social practice in which developers participate. Based on the social philosophy of Alasdair MacIntyre, we construct a theoretical framework that expands our assumptions about individual motivation to include the idea of a long-term, value-informed quest beyond short-term rewards. This motivation–practice framework depicts how the social practice and its supporting institutions mediate between individual motivation and outcome. The framework contains three theoretical conjectures that seek to explain how collectively elaborated standards of excellence prompt developers to produce high-quality software, change institutions, and sustain OSS development. From the framework, we derive six concrete propositions and suggest a new research agenda on motivation in OSS.*

**Keywords**: Free software, incentives, MacIntyre, motivation, open source software, innovation, social practice

## Introduction

Over the last 20 years, open source software (OSS) products have made successful inroads into many information systems (IS) segments, attracting millions of users. OSS, broadly defined, is software where users can inspect the source code, modify it, and redistribute modified or unmodified versions for others to use. Today, firms are both heavy users of OSS products and contributors to their development. As a result, IS managers are increasingly dependent on development resources outside their direct control, giving them reason to be concerned about what motivates developers outside the firm, many as volunteers, to participate in the creation of OSS. For example, if a firm decides to invest millions of dollars to migrate its servers to a Linux system, managers will want to know to what extent Linux will continue to receive contributions from individuals and companies, how the software will evolve, and if the OSS projects involved will manage to release new and improved versions of the software regularly. Thus, IS managers should be interested in the question formulated by Lerner and Tirole (2002) which points to the very existence of the OSS phenomenon: "Why would thousands of top-notch software developers contribute for free to the creation of a public good?" A public good, here, is defined by its non-excludability and non-rivalry in consumption, which applies to software published and licensed under an open source license. Lerner and Tirole's question poses huge challenges for scholars studying IS development within firms that systematically rely on pay and career incentives. Answering this question explains a phenomenon of high academic and practical interest and much research on motivation in OSS has already shed considerable light on critical

[1]M. Lynne Markus was the accepting senior editor for this paper. Jan Pries-Heje served as the associate editor.

The appendices for this paper are located in the "Online Supplements" section of the *MIS Quarterly*'s website (http://www.misq.org).

issues regarding contributions to OSS projects and the emergence and growth of OSS projects and their organization (e.g., Hertel et al. 2003; Markus 2007; Ulhøi 2004; West and O'Mahony 2005).

In this paper, we review the literature on motivation to contribute to the development of OSS. The review shows that the existing literature does not provide satisfactory answers to three differentiated questions as to why this phenomenon exists (see Lerner and Tirole 2002): First, why do OSS developers produce high-quality software when they do? This question is warranted because software quality is critical for attracting interest in OSS, as high-quality OSS systems make headlines due to their developers' achievements in terms of reliability, speed, accessibility, and more. For example, the world's largest stock exchanges run Linux systems while racing for trading speed records (King 2010; Vaughn-Nichols 2009) and 75 percent of the world's websites are served by OSS web servers. However, as our review will show, while the existing literature informs us well about how developers are intrinsically and extrinsically motivated to contribute by means of time, effort, and code contributions, little is known about why they develop high-quality OSS when they do.

Second, why do OSS developers change institutions? A significant impact of the phenomenon on business and public administration stems from the nature of the free and OSS licenses that define and govern their potential use.[2] Public administrators perceive value in the free accessibility of OSS (Comino et al. 2010; Maldonado 2010) and firms have long used and contributed to OSS (Bonaccorsi et al. 2006; Dahlander and Wallin 2006; Henkel 2006; Rolandsson et al. 2011; Stam 2009) as part of a model of innovation that has become known as private–collective (von Hippel and von Krogh 2003). Thus, it is critical to explain why OSS developers set up OS licenses, organizations, and foundations and thus create new institutions that house what they do and make. However, while existing academic work explains how institutions constrain developers' motivation, it does not address more broadly how developers are motivated to change institutions.

Third, why do developers sustain OSS development? This question is critical because the answer indicates whether firms and individual users can expect future development of the software products in use today.[3] In this regard, volunteering OSS developers may constitute an unstable development resource as they can stop development or leave the project anytime. Developers may also suddenly and opportunistically alter their sharing behavior (Osterloh and Rota 2007). However, so far, theory and research have not investigated in much detail why OSS developers engage in activities that guarantee the survival and sustainment of the OSS development practice.

In order to explore the three questions, we first review existing literature on motivation to contribute to OSS development and subsequently advance a new theoretical framework that deepens, enriches, and reinvigorates research on motivation in OSS. We show that the concepts of individual motivation and social practice are mutually constitutive, and argue that theory and research should incorporate a social practice perspective that focuses on action to explain how shared beliefs are created through shared work (MacIntyre 1981; Orlikowski 2005; Rouse 2007), rather than reducing motivation studies to a conventional model that relies on short-term intrinsic and extrinsic motivation (e.g., Hars and Ou 2002). A simple analogy illustrates our argument: Humans often act to satisfy their immediate needs and might fall for the dangling carrot. Occasionally, humans also make elaborate detours, strive for bigger things in life, and undertake long voyages to find the gold at the end of the rainbow (even though they know it is not there). They value the journey, perceive unity in the sum of their efforts, and manage their life so that they can reflect upon it as well lived in a social practice. The new theoretical framework we put forward is based on the social philosophy of Alasdair MacIntyre. MacIntyre's perspective of social practice and motivation rests on a belief that human behavior cannot be decoupled from ethical considerations about what people strive for and the narratives they construct about their life. His work provides three important conceptual building blocks—goods, institutions, and social practice—that assist in the analysis of motivation to act. While much is known about the direct link between individual motivation and output in the form of, for example, submitting new or modified code to OSS projects, much less is known about how motivation is intertwined with forms of cooperative human activity (which we analyze through MacIntyre's concepts of *social practice* and *institution*) and a broader set of related outputs, including high-quality technology for public use (which we analyze through his concept of *good*).

With regard to motivation, OSS differs from traditional software development in firms along three dimensions: incentives, control, and coordination mechanisms. First, to motivate software development, firms rely on pay and career incentives, and other benefits stipulated as part of employ-

[2]For information about the OSS initiative and licenses see http://www.opensource.org/.

[3]See *The Economist*, May 16, 2006, "Open, but Not as Usual" (http://www.economist.com/node/5624944?story_id=5624944).

ment contracts (e.g., Peters 2003). The presence of a significant number of volunteers in OSS development (Hars and Ou 2002; Lakhani and Wolf 2005) limits the effectiveness of incentives to firm employees; volunteers find motivation elsewhere. Second, and related to the use of incentives, software development firms implement control mechanisms, including behavior- or output-based control (see Ouchi 1978). The lack of formal governance and mandatory or formal membership by volunteers excludes managers from exerting control in an OSS project following these traditional ways (Markus 2007). OSS is predominantly characterized by *clan control*, which is based on common values and beliefs (Ouchi 1980), or *self-control*, based on self-monitoring (Kirsch 1996; Ouchi 1979). Third, firms tend to coordinate software development projects through organizational hierarchy and centralized planning (Cusumano and Selby 1995). In contrast, coordinating the work of distributed OSS developers hinges on their ability to exchange information over the Internet at low cost (Bonaccorsi and Rossi 2003). A set of rather simple tools, such as concurrent versioning systems and mailing lists, are used for coordination at the level of information exchange (Lee and Cole 2003).

These three dimensions illustrate differences that can be expected between the social practice of OSS development and what is normally associated with software development in firms and software engineering (Scacci 2002). Moreover, the social practice of OSS development is saturated with ethical aspects. Historically, open source software grew out of the Free Software movement founded by Richard M. Stallman, which was primarily driven by ethical considerations; running software that a user cannot inspect, modify, and share is considered immoral. Developers who identify themselves with the Free, or Libre, Software movement are often motivated by the desire to allow users to control the software they require (Stewart and Gosain 2006). Thus, motivation of OSS developers should also be studied from the perspective of the social practice in which the development work collectively unfolds, evolves, and heeds ethical considerations (Rouse 2007). In the following, we will argue that MacIntyre's theory accomplishes precisely this task.

Next, we describe the method used for the review and examine existing theory and research, which predominately covers the link between motivation and contribution and, to a far lesser extent, the link between motivation, quality, institutions, and sustainable practice. Subsequently, we introduce and critique MacIntyre's theory, which is suited to further exploration of our three questions pertaining to motivation. In a fourth section, we develop a new motivation–practice framework that allows us to answer these three questions. The framework contains a set of theoretical conjectures from which we derive concrete propositions and suggest new research opportunities. The last section concludes the paper.

## Research on Open Source and Motivation

### Review Method

We performed a literature review of publications pertaining to motivation in OSS. The review aimed to (1) reveal in what way and to what degree literature has addressed the three questions we posed in the "Introduction"; (2) organize and classify received literature according to topics covered and their theoretical underpinnings, focusing on motivational aspects; and (3) identify gaps in the current literature that justify the creation of a new framework.

The review stages conducted correspond to those recommended by Cooper (1998): problem formulation, data collection, data evaluation, analysis, and presentation. In doing so, we were able to take advantage of the suggestions and potential pitfalls associated with each stage of the review. The reviewed papers originated in different disciplines, including organization and management theory, anthropology, economics, law, psychology, and sociology, and applied a variety of qualitative and quantitative research designs. We therefore report the data and methods used for each paper reviewed (See Appendix A).

We adopted a broad hierarchical search strategy to capture high-quality and relevant articles starting with the most reliable sources and, in subsequent searches, adding articles not previously identified. The search strategy followed four main steps (the first three were carried out on June 22, 2009, while the papers in the fourth category were added on a continuous basis). First, we identified all articles listed in the ISI Web of Knowledge database published by Thomson Reuters containing the terms *open source*, *motivation*, or *incentive(s)* in the title, abstract, or keywords. This index is the most critical source of published material since it contains published and peer-reviewed articles. The initial search yielded 214 articles; we excluded those that did not specifically deal with the relationship between motivation and OSS. Search results that described software that happened to be open source were also excluded, as were articles that focused exclusively on technical issues.

Our review focused on motivation in the practice of software development under open source licenses and excluded other

activities carried out in electronic networks of practice (Wasko and Faraj 2005), online communities (Wiertz and de Ruyter 2007), and social networks sites more generally (Boyd and Ellison 2007). While motivation patterns might be similar across various activities, we have no inclination to combine in one review motivation studies in different practices simply because they might occur online or share volunteer contributions. In total, we included 26 articles from the ISI Web of Knowledge database. Second, we browsed the paper repository at http://opensource.mit.edu (no longer online) to identify articles that matched the ISI search criteria. This online paper repository was a major source of unpublished papers and work in progress dealing with OSS. We included eight articles from this source.

Third, in order to be more comprehensive, we searched Google Scholar with the combined search terms *open source* and *motivation*. Google Scholar searches the scholarly literature and identifies articles from multiple disciplines and sources: peer-reviewed papers, theses, books, abstracts, and articles, from academic publishers, professional societies, preprint repositories, universities, and other scholarly organizations. The initial search yielded 38,900 (often duplicated) articles. Because of the vast number of articles identified, we opted to retain only the top 200 articles. Google Scholar ranks articles

> the way researchers do, weighing the full text of each article, the author, the publication in which the article appears, and how often the piece has been cited in other scholarly literature. The most relevant results will always appear on the first page.[4]

In line with the two previous searches we excluded articles not specifically dealing with the relationship between motivation and OSS. We included one additional article that appeared only in Google Scholar. In addition, we searched Google Scholar using slightly different search phrases, such as *incentive* and *open source software*. However, this search failed to yield any relevant articles to add to the full sample. Fourth, we included topical conference papers and book chapters that were known to us and colleagues in the field (and which had escaped the previous identification). This yielded six additional articles.

Finally, we only included English-language documents, which led to the exclusion of one ISI paper. The search strategy led to 40 articles in our review sample, with a comprehensive inclusion of empirical contributions but a selective inclusion of purely theoretical contributions. The sample included all articles that empirically investigate the relationship between motivation and OSS development but excluded nonempirical, non-ISI articles that do not develop new theoretical categories or explanations. Conceptual papers were examined for motivational factors that were used to create theory. Motivational factors that proved relevant in empirical papers were also included in the review.

We coded all papers according to motivational aspects covered, as well as to how they regarded institutional context. For example, surveys proposing "I am contributing software because I learn from receiving feedback on my work" led to the creation of a "learning" dimension of individuals' motivations to contribute. If a study used a different terminology, but the specific motivation seemed sufficiently close to an existing one in the taxonomy, it was merged into the existing category. After a first round, we convened and performed a triage of all studies and motivational dimensions, merging them when they seemed to express the same type of motivation. This led to a final typology of 10 clusters related to individual motivation. As a final step, we decided whether each cluster covered intrinsic motivation, internalized extrinsic motivation, or extrinsic motivation.

While reviewing the articles, we also categorized topics relating to institutions (which are outside of the direct control of a single individual), leading to a list of five dimensions, as discussed later. A complete list of papers included, with a brief description of methods and data used, is included in the Appendix A.

### The Motivation to Contribute

Studies of individual motivation to contribute to OSS development were grouped into 10 motivational categories. We also examined the frameworks used to justify and categorize motivations. While a wide variety has been used, the most frequent framework by far has been the distinction between intrinsic and extrinsic motivation in self-determination theory (SDT) (see Deci and Ryan 1985; Gagné and Deci 2005). This distinction is based on different reasons that bring about human action. An action is extrinsically motivated when it is performed to obtain some separable outcome, whereas an action is intrinsically motivated when it is done for the inherent interest or joy of performing it (Deci and Ryan 1985). A number of empirical studies have shown that OSS developers have both intrinsic and extrinsic motivations for contributing to its development (Hars and Ou 2002; Lakhani and Wolf 2005; Roberts et al. 2006; Wu et al. 2007),

[4]Http://scholar.google.com/intl/en/scholar/about.html, retrieved June 22, 2009.

and we summarize how some of the works make use of SDT. Following Lindenberg (2001) and Lakhani and Wolf (2005), Osterloh and Rota (2007) distinguished between enjoyment-based intrinsic motivation and obligation-based or community-based intrinsic motivation. Their paper provided a theoretical overview, whereas Lakhani and Wolf presented survey data showing that both types of intrinsic motivation, as well as extrinsic motivation, impacted on work effort. Wu et al. (2007) also used the SDT framework to explain the continued intention to contribute to OSS projects.

Most of the studies using SDT focused on intrinsic motivations. For instance, Hars and Ou (2002) suggested that intrinsically motivated developers spend more time and effort in open source projects than extrinsically motivated developers, but did not examine this empirically. Other empirical studies concentrated on intrinsic motivation rather than extrinsic motivation. For example, Lakhani and von Hippel (2003) linked feelings of competence and fun to willingness to help other developers. Authors have also viewed motivation in relation to reciprocity, for example, giving software patches as "gifts" to the community (Bergquist and Ljungberg 2001; Wu et al. 2007); reciprocal helping behavior (i.e., helping because of having been helped or expecting to be helped) (Lakhani and von Hippel 2003); or status motivation (Roberts et al. 2006). This work stands in contrast to Lerner and Tirole's (2002) early explanation of contribution, purely based on extrinsic motivation. Few of the works on intrinsic motivation looked at how institutional context shapes motivations (and incentivizes to create or change institutions).

While alternative frameworks have been proposed, they are often closely related to Deci and Ryan's original framework of extrinsic and intrinsic motivation. For example, Bonaccorsi et al. (2006) distinguished between economic, social, and technological motivation, building on a taxonomy proposed by Feller and Fitzgerald (2002). Economic motivation is similar to extrinsic motivation, while social motivation roughly corresponds to intrinsic motivation. The authors also suggested a third type—*technological motivation*—that includes benefits from learning and working with a "bleeding-edge" technology.

Attempts to develop a broader and integrative framework can be found in Hemetsberger (2004) and Hertel et al. (2003). Hemetsberger viewed motivation as "self-interest" and "others-orientation." Self-interest was further divided into task- and product-related motivation (corresponding to intrinsic motivation); others-orientation, including long-term utilitarian goals and social significance (corresponding to extrinsic motivation), was divided into internalized group goals and values, and socio-emotional relationships. Hertel et al. extended a model of voluntary action in social movements proposed by Klandermans (1997). The authors included collective, norm-oriented, reward, and identification motives, and combined these with the "VIST-model" (Hertel 2002) that explains individual motivation in virtual teams.[5] These frameworks go beyond self-determination theory in specific aspects but do not answer any of the three questions we developed in the previous section.

Most of the work on motivation based on the intrinsic/ extrinsic framework can be grouped into intrinsic motivation, internalized extrinsic motivation, or extrinsic motivation (see Table 1 for an overview of the resulting dimensions). Some motivations are by definition extrinsic but developers could internalize them, so that they are perceived as self-regulating behavior rather than external impositions (Deci and Ryan 1987; Roberts et al. 2006). These internalized extrinsic motivations include reputation, reciprocity, learning, and own-use value. Pure extrinsic motivations include careers and pay. The specific dimensions with key empirical findings of the papers in our sample are summarized in Appendices B, C, and D.

Interaction among motivation factors has been given less attention, with the notable exception of crowding theory (Frey and Jegen 2001), which predicts mutual adjustments between intrinsic and extrinsic motivation; introducing pecuniary incentives may not always increase the supply of a public good (Frey and Jegen 2001). Osterloh and Rota (2007) argued that extrinsic motivation might crowd out voluntary sharing of software and other knowledge. Several surveys have shown that around 40 percent of the contributions to OSS emanate from paid working time (Hars and Ou 2002; Hertel et al. 2003; Lakhani and Wolf 2005; Luthiger and Jungwirth 2007). However, crowding out of intrinsic motivations cannot be detected in empirical studies so far. Lakhani and Wolf found that intrinsic and extrinsic motivations coexist, and Roberts et al. (2006) detected no crowding out of intrinsic motivation by extrinsic motivation. Osterloh and Rota argued that this might be due to a balance between intrinsic and extrinsic motivations and the "pro-social intrinsic motivation of a sufficient number of participants to contribute [enforcing the rules of cooperation]" (p. 196). Increased reputation and career concerns (status motivation) even enhance developers' intrinsic motivations, although extrinsic motivation does crowd out "own-use value" motivation (a form of internalized extrinsic motivation) (Roberts et al. 2006).

[5]VIST stands for valence (the subjective evaluation of goals), instrumentality (the perceived importance of one's own contribution), self-efficacy (team members' perceived ability to perform the required activities for the team task), and trust (the expectation of reciprocity rather than exploitation).

**Table 1. Intrinsic and Extrinsic Motivations**

| | Intrinsic | | | | Internalized Extrinsic | | | | Extrinsic | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Ideology | Altruism | Kinship | Fun | Reputation | Reciprocity | Learning | Own-use | Career | Pay |
| Alexy and Leitner 2007 | | | | | | | | | | ✓ |
| Baldwin and Clark 2006 | | | | | | | | | | |
| Benkler 2002 | | | | ✓ | | | | | | ✓ |
| Berquist and Ljungberg 2001 | | | | | | ✓ | | | | |
| Bitzer et al. 2007 | | ✓ | | | | | | ✓ | | |
| David and Shapiro 2008 | ✓ | | ✓ | | | ✓ | ✓ | ✓ | | |
| David et al. 2003 | ✓ | | | | | ✓ | ✓ | ✓ | | |
| Fershtman and Gandal 2004 | | | | | | | | | | |
| Ghosh 2005 | ✓ | ✓ | | | ✓ | | ✓ | ✓ | ✓ | ✓ |
| Hars and Ou 2002 | | ✓ | ✓ | | ✓ | | ✓ | ✓ | ✓ | |
| Haruvy et al. 2003 | | ✓ | | | | | | | | |
| Hemetsberger 2004 | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | |
| Hertel et al. 2003 | ✓ | | ✓ | ✓ | ✓ | | | ✓ | ✓ | ✓ |
| Ke and Zhang 2008 | | ✓ | | | | | | | | |
| Lakhani and von Hippel 2003 | ✓ | | | ✓ | ✓ | ✓ | | ✓ | | |
| Lakhani and Wolf 2005 | ✓ | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Lattemann and Stieglitz 2005 | | | | | ✓ | | | ✓ | | ✓ |
| Lee and Cole 2003 | | | | | | | | | | |
| Lerner and Tirole 2002 | | | | | ✓ | | | | ✓ | |
| Luthiger and Jungwirth 2007 | | | | ✓ | | | | | | ✓ |
| Markus 2007 | | | | | | | | | | |
| Okoli and Oh 2007 | | | | | ✓ | | | | | |
| O'Mahony and Ferraro 2007 | | | | | | | | | | |
| Oreg and Nov 2008 | | ✓ | | | ✓ | | ✓ | | | |
| Osterloh and Rota 2007 | | ✓ | | | | | | ✓ | | |
| Riehle 2007 | | | | | | | | | ✓ | |
| Roberts et al. 2006 | | | | ✓ | ✓ | | ✓ | ✓ | ✓ | ✓ |
| Rullani 2007 | | | | | | | | | | |
| Schofield and Cooper 2006 | | ✓ | ✓ | | | | | ✓ | | |
| Shah 2006 | ✓ | | | ✓ | | ✓ | ✓ | ✓ | | |
| Spaeth et al. 2008 | | | | | ✓ | | ✓ | | | |
| Stewart et al. 2006 | | | | | | | | | | |
| Stewart and Gosain 2006 | ✓ | ✓ | | | ✓ | | ✓ | | | |
| von Hippel and von Krogh 2003 | | | | ✓ | | | ✓ | ✓ | | |
| von Krogh et al. 2003 | | | | | | | | | | |
| Wu et al. 2007 | | ✓ | | | | | ✓ | ✓ | ✓ | |
| Xu et al. 2009 | ✓ | | | ✓ | ✓ | | ✓ | ✓ | | |
| Ye and Kishida 2003 | | | | | | | ✓ | | | |
| Yu et al. 2007 | ✓ | ✓ | | | ✓ | ✓ | ✓ | | ✓ | |
| Zeitlyn 2003 | | | ✓ | | | | | | | |

To conclude, while research on motivation in OSS generated a clear link between extrinsic and intrinsic motives and contributions, it did not relate individual motivation to the quality of the contributions made. This leaves our first question (How and why do OSS developers produce high-quality software when they do?) unanswered.

## Motivation, Institutions, and Social Practice

OSS development is a form of collective action that shapes institutions, and in turn enables individuals to contribute (von Hippel and von Krogh 2003). Decades of research into other forms of collective action, ranging from lobbying and preservation of natural resources, to money collection for a good cause, have shown that institutions and individual motivations are interrelated (e.g., Morris and Mueller 1992). As a result, we believe that it is important to investigate both the individual level and the social context of development in order to understand individual behavior in a social practice. This section reports on our findings on the relations between motivation, institutions, and social practice in the literature reviewed.

The research reviewed here mostly investigates why developers are moved to contribute to OSS development. However, it has disregarded potential external influences and interferences, and focuses instead on the underlying needs, wishes, and goal orientations of individuals. Early on, though, Deci and Ryan (1985) pointed to the link between a context (the external environment) and individual motivation. In self-determination theory, extrinsic and intrinsic motivations are both predictors and outcomes of institutional arrangements such as governance or norms, depending on "the nature of the study and the way and time in which self-determined motivation is measured" (Sheldon and Krieger 2007, p. 885). Accordingly, some OSS scholars have recently moved beyond the direct link between motivation and contribution, and investigated factors that enable and constrain motivation and contribution to OSS. These factors are external to individuals and often beyond their direct control, impacting indirectly on OSS contributions. They do so either by influencing individual motivation or moderating its effect. While some authors discuss the interrelationship between the motivation of OSS developers and contextual factors that impact on development, most of this work is recent and difficult to categorize (e.g., Shah 2006; von Hippel and von Krogh 2003). Yet, our review was able to identify five kinds of contextual factors of relevance to our second question (Why do OSS developers change institutions?) and the third (Why do developers sustain OSS development?).

First, a few studies related motivation to institutions in terms of governance, community sponsorship, the provision of rewards, and license restriction. Markus (2007, p. 152) defined OSS governance as

> the means of achieving the direction, control, and coordination of wholly or partially autonomous individuals and organizations on behalf of an OSS development project to which they jointly contribute.

While governance in OSS has been described in terms of structure, practices, rules, and norms, it leaves unanswered an important question relating directly to motivation: What is the source of control in OSS development communities (Markus 2007, p. 153)? As well as exerting control, the creation of routines and rules impacts on developers' voluntary engagement and motivation. Markus's question also points to the organizational sponsorship of OSS development. Shah (2006) distinguished between "open" and "gated" source communities, where gated refers to developers' limited access to the development process, due to firm sponsorship and control. She found that, in the long run, developers who are mainly motivated by use value tend to contribute to gated source communities, whereas developers mainly motivated by enjoyment contribute to open source communities. Stewart et al. (2006) investigated the role of community sponsorship and distinguished between market (e.g., firm) and nonmarket (e.g., university) sponsors and concluded that developers pick up cues as to the project's future from the type of sponsor, impacting incentives to contribute. For example, they found that projects with a nonmarket sponsor attracted greater development activity than projects without a sponsor. How contributions are rewarded also seems to matter. Alexy and Leitner (2011) criticized existing OSS literature for assuming that intrinsic motivation and extrinsic financial rewards have a one-dimensionally positive effect on the motivation of individual developers; they claimed that payment norms moderate the effect on intrinsic and total motivation. It has also been found that license restriction impacts on developers' motivations, in the sense that less restrictive licenses tend to attract more development activity to a project (Fershtman and Gandal 2007; Stewart et al. 2006). The literature reviewed thus far considers institutions as constraints to motivation and has studied the interaction and interference that governance, community sponsorship, provision of rewards, and license restrictions inflict on individual motivation. That is, causality has been established between the way institutions impact on motivation and human behavior. However, the opposite causal direction has not been investigated although it is known that OSS developers set up governance structure

(O'Mahony and Ferraro 2007). This means that Question 2 (Why do OSS developers change institutions?) remains unanswered.

Second, a number of studies related motivation to norms and exposure to a specific community. The literature reviewed identifies social and technical exposure to a community that over time creates opportunities, in terms of advancement within the community's social structure (Rullani 2007; von Krogh et al. 2003) and of insights that can lead to more challenging tasks (Shah 2006). To get their work accepted by established developers and to be able to exert influence on the technical design in the project, newcomers have to adhere to behavioral scripts for joining the community. Learning about and following "joining scripts" takes time for developers, but this time is essential if they are to advance to community leadership or other central positions (von Krogh et al. 2003). Although this literature approaches the idea of a social practice, Question 3 (Why do OSS developers sustain the social practice of OSS development?) remains unanswered.

Reviewing state-of-the-art literature, we identified five dimensions (governance, community sponsorship, provision of rewards, license restrictions, and social and technical exposure to the community) associated with institution and social practice that impact on motivation to contribute source code (see Table 2). However, little is yet known about what moves OSS developers to (1) produce high-quality work when they do, (2) engage in institutional change, or (3) sustain OSS development. In order to address these issues, we suggest research to take a social practice perspective on OSS development.

To summarize, individual motivation rooted in people's search for immediate outcomes is important but does not suffice to explain critical facets of the OSS phenomenon. The OSS phenomenon initially triggered research on contributors' motivation from a self-determination perspective, but to get more differentiated explanations we may need to look beyond this dominating perspective. Some of the work reviewed represents a first step toward taking into account institutions and practice; yet it seems that the emergence of institutions through social practice, and their interactions in OSS development, are currently not well understood. A perspective that explicitly engages with these interactions can be found in the social philosophy of Alastair MacIntyre, which we introduce in the following section. Subsequently, we develop a framework that contains theoretical conjectures from which we derive concrete propositions. The framework fills the identified research gap and thus answers the three questions posed in the "Introduction."

## OSS Development: Conceptual Building Blocks

In recent years, scholars have increasingly shown interest in the influence of social practices on the evolution of economy and society, often referred to as the "practice turn" in the social sciences (Schatzki et al. 2001). The practice turn has also influenced theorizing in IS development and use (e.g., Kellogg et al. 2006; Orlikowski 2000; Suchman et al. 1999) and stimulated theory development on human motivation that takes into account the relationship of passionate individuals engaging in their social practices and the role individuals play in the institutionalizing power of social practices (Brown and Duguid 1991). The practice turn describes the work of social scientists who attempt to articulate relationships between individual activities, structures, and matter (for example, rules or physical bodies) that have been separated by various dualisms (such as structuralism or methodological individualism). Alasdair MacIntyre's seminal work *After Virtue* (1981) presents a theory that extends and links aspects from the social practice perspective. The theory includes an ethical dimension to the discussion of quality in the goods produced by motivated individuals in a social practice that is either enabled or constrained by institutions (Knight 1998).

MacIntyre's work has had wide-ranging implications for several fields, including sociology, political science, education, religious studies, and business ethics (for a discussion and introduction, see Knight 1998). His perspective on human motivation is rooted in moral philosophy, in particular virtue ethics, which concerns the knowledge of how to live a good life (MacIntyre 1981). As we showed in the last section, motivation is often considered a psychological reaction aroused in human beings that make us act in the direction of a specific goal. However, according to MacIntyre, understanding what people do (or do not do) and why they do it cannot be decoupled from an analysis of ethical considerations, such as what they strive for in life, the narratives they construct about their life, and why these are worthwhile to them and others. The theory builds on, and criticizes, Aristotelean value ethics by arguing from a more relativistic and process-based point of view. Furthermore, he takes a dynamic and historical perspective that is rare in mainstream motivation studies and explores the role of institutions and social practices in shaping virtues.

More specifically for our purpose, MacIntyre's theory is useful for analyzing motivation in the context of OSS development because it raises questions of ethics that guide the activities of people, institutions, and social practices. As shown by Stewart and Gosain (2006), OSS developers often

**Table 2. Institutions and Social Practice**

| | |
|---|---|
| **Governance** | Governance refers to "the means of achieving the direction, control, and coordination of wholly or partially autonomous individuals and organizations on behalf of an OSS development project to which they jointly contribute" (Markus 2007, p. 152).<br><br>*Key empirical findings:*<br>• Benkler (2002) argued that peer production has a relative advantage over firm- or market-based production because of the highly variable nature of human capital and lower costs of coordination and communication.<br>• Lattemann and Stieglitz (2005) argued that the adequacy of governance tools is related to the motivational preferences of participants and that behavioral and output control should be regarded as secondary to social control in the form of morals and cultural rules.<br>• Lerner and Tirole (2002) argued that the success of an OSS project is dependent on its ability to break it into distinct components as well as the presence of a credible leader or leadership. They suggest that code modularity enables the former and exemplifies the latter with projects characterized by authority or consensus.<br>• Markus (2007) suggested three purposes of OSS governance: Solving collective action problems, solving coordination problems, and creating a better climate for contributors. Her review also suggested a framework for future comparative and case study research on OSS governance.<br>• O'Mahony and Ferraro (2007) concluded that, contrary to common belief, the contributions the community valued the most were not purely technical. The study showed that over time a governance structure emerged that valued the informal work of coordinating individual efforts and linking them to community goals. In doing so, the members developed a shared basis of formal authority but limited it with democratic mechanisms.<br>• Shah (2006) distinguished between open and gated software communities, performing a qualitative study of both, and found that the difference in governance structure affects the participation choices of volunteer OSS developers. In addition, she found that in the long run, developers who were mainly motivated by use-value tended to contribute to gated source communities, whereas developers mainly motivated by enjoyment contributed to open source communities. Considerations of fairness and reciprocity explain this behavior. The developers were aware of the property rights situation in the gated source communities and suspected firms of acting "strategically," that is, neglecting the needs of the community and pursuing their economic interests. Thus, developers in gated source communities contributed only if they derived direct use-value from their work. |
| **Community sponsorship** | Sponsorship refers to control by an organization, such as a firm, over the development process, source code accessibility, or code ownership.<br><br>*Key empirical findings:*<br>• Shah (2006) studied open and gated software communities and argued that firms may encounter difficulties if they seek to construct hybrid arrangements that balance community-based value creation with private value appropriation.<br>• Stewart et al. (2006) found that community sponsorship and licensing address complementary developer motivations, so that the influence of licensing on development activity depends on what kind of organizational sponsor a project has. In addition, OSS projects with sponsors—non-market sponsors in particular—attract more attention. |

| Table 2. Institutions and Social Practice (Continued) | |
|---|---|
| **Provision of rewards** | The provision of rewards refers to incentive structure, i.e., how someone other than the developer is rewarded for participation in OSS development.<br><br>*Key empirical findings*:<br>• Alexy and Leitner (2011) found that payment norms moderate the effect on motivation. In particular they studied the impact of rewards and payment norms within a community on developers' intentions to contribute to an OSS project. They found that payment exhibited a positive effect on developers' total motivation when they were offered a monetary reward on completion as long as payment was not strongly expected. At the same time intrinsic motivation only decreased for that share of the sample that received payment, which also perceived the existence of social norms refuting payment.<br>• Benkler (2002) distinguished between, and related, monetary rewards, intrinsic hedonic rewards, and social-psychological rewards (such as social associations and status perception). He formalized the possible effects of increased monetary rewards. In particular, he discussed the situation when one agent is jealous of another's rewards (which he called *jalt*), such as when some people are paid and others not.<br>• Lerner and Tirole (2002) drew on labor economics when they analyzed the economics of OSS and concluded that OSS contributors could be directly rewarded though employment or rewarded by signaling and subsequently gaining employment.<br>• Roberts et al. (2006) found that paid participation lead to above-average contribution levels. In addition, they found that being paid to contribute was positively related to developers' status motivations but negatively related to their use-value motivations. No evidence of diminished intrinsic motivation in the presence of extrinsic motivations was found. |
| **License restriction** | License restriction is usually thought to limit the commercial exploitation of the source code, i.e., the option to combine OSS with proprietary software for sale.<br><br>*Key empirical findings*:<br>• Fershtman and Gandal (2007) found that output per contributor is much higher when licenses are less restrictive and that the number of contributors is higher when licenses are restrictive.<br>• Lerner and Tirole (2002) described changes toward less restrictive OSS licenses that opened up for bundling OSS with proprietary software. They discussed the firm advantages of releasing existing proprietary code.<br>• Osterloh and Rota (2007) claimed that OSS can flourish even after the advent of a dominant design, when knowledge sharing is no longer supported by great learning potential, low opportunity costs, and selective benefits. They argued that this is because OSS is better at solving first- and second-order social dilemmas where OSS licenses hinder the exploitation of voluntary donors.<br>• Stewart et al. (2006) found that community sponsorship and licensing address complementary developer motivations, so that the influence of licensing on development activity depends on the kind of organizational sponsor a project has. For example, "the presence of a non-market sponsor may alleviate concerns as to the project's future in the same way as a restrictive license would, in the sense that the restrictive license is not perceived as necessary to protect the developers' interests" (2006, p. 141). In addition, OSS projects that use a nonrestrictive license will attract greater user interest over time than those that use a restrictive license. |
| **Social and technical exposure to the community** | Refers to effects from exposure that, over time, create opportunities for advancement and work on more challenging tasks that require deep insights into the community's workings.<br><br>*Key empirical findings:*<br>• Rullani (2007) concluded that exposure to a community increases developers' contributions independently of their predetermined preferences.<br>• Shah (2006) maintained that the developers' long-term accumulation of knowledge provided opportunities to engage in more challenging tasks, or tasks that require broader knowledge of the code base (multiple modules and components).<br>• Spaeth et al. (2008) found that contributions impacted on developers' positioning in the community, which in turn provided some developers with private benefits, such as reputation gain, influence of the technical agenda, and learning opportunities.<br>• von Krogh et al. (2003) found that newcomers who wanted to get their work accepted by established developers and to be able to exert influence on the technical design in the project, have to adhere to behavioral scripts for joining the community. |

report that their work is guided by strong ideologies (which may be expressions of ethics), such as writing software that should be free for all to download and use. A review and framework of motivations in the context of the OSS development phenomenon, therefore, needs a theoretical foundation that allows such ethical and ideological issues to be addressed.

Further, MacIntyre is concerned with the implications of so-called "incommensurable moral premises" rather than approaches to reach a consensus on values, norms, and decisions often found in conventional (analytical) moral philosophy (e.g., Habermas (1988) on rational dialogue and communicative action; see Mingers and Walsham (2010) for an application to IS). The theory is based on the assumption that people from different cultures, practices, and institutions may "subscribe" to various ethics and thus may also argue from different moral premises that would be impractical or impossible to reconcile. Incommensurable moral premises are relevant for understanding motivation in OSS development, too. While commercial firms provide pecuniary and career incentives to motivate developers, many (but not all) OSS developers work for free (Lakhani and Wolf 2005). According to field accounts, some developers even find it "immoral" to work for a commercial software company (Stallman 1999).

Trying to change developers' moral premises may be difficult and impractical. Firm participation in OSS development and gated communities (Roberts et al. 2006; Shah 2006) raises the question of implications for developers' motivation, given the coexistence and interaction of institutions and social practices that build on different moral premises. People who join a social practice gradually adopt shared values and strive for standards of excellence defined within it (MacIntyre 1981). Thus, the analysis of moral premises in a social practice is essential in order to understand what motivates people to perform and improve a craft. As Feller and Fitzgerald (2002) argue, software development is a craft that has evolved its own professional standards, which constitute the context in which software development is done. It is reasonable to assume that, in contemporary software development, people are differently motivated and conduct their work based on vastly different (perhaps incommensurable) moral premises. An analysis of motivation in this context thus needs a framework that can capture these moral premises.

There are three concepts from MacIntyre's theory that are central to our analysis: social practice, goods, and institutions. Human activity is a holistic expression of the narrative of a human life embedded in some social traditions that give rise to social practices and the pursuit of quality. MacIntyre defines a (*social*) *practice* as

> any coherent and complex form of socially established cooperative human activity through which goods internal to that form of activity are realized in the course of trying to achieve those standards of excellence which are appropriate to, and partly definitive of, that form of activity, with the result that human powers to achieve excellence, and human conceptions of the ends and goods involved, are systematically extended (MacIntyre 1981, p. 187).

This definition can be applied to a range of professions and crafts, such as architecture, medicine, journalism, science, and the arts, with the precondition that a social practice should have wide and positive effects for humankind.[6]

Because of its ubiquitous presence and the wide-ranging impact of IS in all aspects of contemporary human life, software development, like many other areas of engineering and technology (Latour 1996; van der Burg and van Gorp 2005), should be considered a social practice in MacIntyre's sense of the concept. Scacchi (2002), though, has written about the differences between traditional software engineering and OSS development practices (see also Rolandsson et al. 2011). He notes that

> [OSS developers] enact teamwork structures and relatively flat, peer-oriented, decentralized community forms that reduce/supplant functional organizational forms inherent in traditional [software engineering] techniques that increased bureaucratic tendencies. [OSS] avoids reliance on formal project management techniques and administrative structures that pervade industrial [software engineering] projects (p. 3).

[6]A social practice is the basis for making decisions about which virtues are called for in particular circumstances and the best way of enacting them (Fowers 2003; Noel 1999; Rämö 2004). Dunne (1992) refers to "ethical knowledge" that directs "ethical action." Aristotle's discussion of the "good" refers to man's practice "of his soul's faculties in conformity with excellence or virtue" (McKeon 1941, p. 33). The good becomes a metaphysical goal, like truth, justice, and beauty, toward which people strive by adjusting their life and actions. However, in a modern world "good" may be contested by people who pursue different goals. The very standards that define what is good may be subject to different interests and, therefore, judgment itself will be judged as more or less virtuous (whether something is judged good or bad, right or wrong). MacIntyre argues we should rather understand the common good as internal to a social practice, as a goal to be achieved by its practitioners (Knight 1998).

Moreover, the social practice is intertwined with the technical object (see the discussion in Orlikowski and Scott 2008). MacKenzie (2005) suggests that the OSS code itself, with modular, functional, and transparent objects, gives rise to a social practice with its own ethics. The technical object of the software code requires developers to behave in a specific way when creating and maintaining it, for example, modularizing, reusing, keeping to the API specification, or taking great care to document (Baldwin and Clark 2006). Thus, "to be an OSS developer" means to engage in a social practice and to adhere to its specific rules of conduct, because these are believed to enable the creation of a high-quality product for its users (von Hippel 2001; von Krogh and von Hippel 2006). At the same time, according to Scacchi, it also means abandoning other rules related to institutions of software engineering prevalent in industry.

When individuals act, MacIntyre's theory proposes that social practices together with institutions create two types of good: external and internal. *External goods* (external to the social practice) include capital, status, or power, which are the property of individuals and/or institutions. *Internal goods* are defined by the social practice and are public goods that benefit all participants in the social practice and the wider community. This is why, as noted above, MacIntyre (1981) chooses to speak of a social practice in relation to collective activity that produces wide-ranging positive effects for humankind. The difference between external and internal goods can be exemplified in the case of science. Scientific knowledge can be considered an internal good of science, of benefit to the scientific community and humanity at large. The creation of scientific knowledge adheres to the highest methodological standards set by the scientific community. The status and salary bestowed on individual scientists, and the power that follows from their expertise, are the external goods of science. However, there is no short cut to obtain these external rewards, which illustrates the importance of practice-based virtues of practitioners extending beyond their personal preferences. One first needs to work hard on developing the skills necessary for being a good scientist (as an unpaid student, for example) in the view of the scientific community, before personal economic benefits can be realized. Analogously, internal and external goods exist in OSS development. Internal goods are, for instance, the resulting software code, which—under an appropriate license—has public good characteristics. Additional internal goods to the practice are the joy of collaborating with similar-minded peers and the spreading of quality software, empowering software users. Given the wide range of developers' skills, the code quality nevertheless varies across projects. Individual reputation, the opportunity for developers to signal their value to potential employers through their code, and the solution to one's private technical problems are external goods that are accessible by one individual.

MacIntyre's theory emanates from a criticism of Aristotle's work on ethics and virtues in political leadership. He notes that practitioners achieve excellence of character or virtue in pursuing internal goods. To act in a virtuous manner is to emulate the rules of morality rather than simply abiding by them because one is in one way or another commanded to do so. A social practice, therefore, is a "school of virtue," where practitioners learn aspects of the internal good, such as ethical reasoning, argumentation, criteria for excellence and product quality, rules of communication, and so forth. Justice, courage, truthfulness, and, above all, love for the social practice, are cultivated through practitioners' participation. Practitioners discover and commit to goals that lie beyond their own selfish, short-term needs and desires. They realize that they can only achieve the internal goods that are of value to themselves, their social practice, and wider society when they emulate the standards of excellence already established within the practice. Pursuing internal goods (excellence) is synonymous with cultivating virtues by subordinating oneself and one's relations with others to the reasoning that is internal to the social practice.[7]

As Feldman and Orlikowski (2011, p. 11) argue, "in focusing on practice theory, we understand the mutually constitutive ways in which agency is shaped by but also produces, reinforces, and changes its structural conditions." We refer to these structural conditions, which include organizations (e.g., hierarchy, community), rules (e.g., coordination), and routines, as institutions. More specifically, in MacIntyre's theory *institutions* are "characteristically and necessarily concerned with [the provision of] external goods" (1981, p. 194), and can be understood as sustainable forms of human cooperation, governed by these organizations, rules, and routines, which exist beyond the presence and efforts of each individual. Thus, medicine, chess, and software development are

[7]For there to be an identifiable common and internal good, MacIntyre (1994, p. 35) suggests that "there must be identifiable structures of community, so that one can understand how the parts which different individuals contribute are contributing to a common goal." However, as Blackledge (2009, p. 870) argues in his analysis of MacIntyre and social practices,"It is…only in small-scale communities that politics can escape from the compartmentalization that is endemic in the modern world." MacIntyre believes that it is this compartmentalization that hinders the "flourishing of local communities" (1998b, p. 248) and that it is only in local communities that "cooperation as a common good" (1999, p. 114) can emerge spontaneously. However, the Internet allows even large global communities with hundreds of developers and users to follow each other's work, making Blackledge's insistence on "local" less relevant for analyzing OSS development.

examples of social practices, whereas a hospital, a chess club, or a software firm are institutions.

Institutions such as governments, NGOs, and firms are a prerequisite for the organization and sustenance of social practices. In the words of Beadle and Moore (2006), "institutions house the social practice." A social practice cannot exist and be sustained without the supporting structures that provide rules for human cooperation. The relationship between institutions and social practices is so intimate that they

> form a single causal order in which the ideals and the creativity of the practice are always vulnerable to the acquisitiveness of the institution, in which the cooperative care for common goods of the practice is always vulnerable to the competitiveness of the institution" (MacIntyre 1981, p. 194).

At the same time as institutions enable social practices to produce internal goods, they distribute external goods in the form of power, status, and financial rewards. Thus, internal goods and external goods are always produced concurrently, giving rise to unavoidable tensions between social practices and institutions to be discussed below. To summarize, the conceptual building blocks discussed here form a dynamic and mutually dependent complex that helps us to understand aspects of human behavior in general and cooperative behavior, such as OSS development, in particular.

## MacIntyre's Critique of Firms and Managers

MacIntyre's theory has been used in organization studies (e.g., Beadle and Moore 2006) but, as Dawson and Bartholomew (2003) note, applying it is not unproblematic since MacIntyre criticizes heavily contemporary notions of management and firms as institutions. While this concern must be taken seriously, we will show that the strength of the conceptual building blocks is precisely that they derive explanatory power from highlighting distinct differences between social practices and institutions.

Whereas institutions provide practitioners with external goods that satisfy their need for compensation, they may also corrupt or fail to support social practices by overshadowing or conflicting with motives to develop a craft to adhere to the social practice's ethics. Inevitably, much of MacIntyre's (1981) critique of institutions is directed at management. However, Moore (2008) points out that this critique is part of a general critique of modernity; we live in an emotivist culture in which moral choices and actions are expressions of people's preferences and emotional states. In MacIntyre's highly controversial view, managers' preferences are to seek external goods as ends, using other individuals as means. They justify their power and monetary compensation by implementing techniques and systems for social change. However, because of the complexity of organizations, techniques and systems seldom lead to predictable outcomes in terms of enhancing effectiveness or efficiency, and so he claims the basis for justification must be false. Furthermore, the implementation of planned social change can do more harm than good to a social practice's capacity for producing internal goods. This critique echoes Robey and Markus's (1984) much-cited analysis of the unpredictable outcomes of the design of management information systems (MIS). Moreover, because managers are often influenced by monetary rewards and seek to create and appropriate excessive external goods, their actions may undermine practitioners' motivations to improve their social practice, and ultimately destroy its social fabric, including relationships between practitioners.[8]

Analyzing the "morality of management," Moore attempts to find a solution to this conundrum. In his view, the business organization is a social practice–institution combination where the presence of virtuous agents at both levels may result in a "virtuous business organization." The manager partakes in both the "core" social practice of, for example, software development and the social practice geared at building and changing institutions (e.g., running a software firm). MacIntyre repeatedly warns that because institutional goals may conflict with the internal goods of a social practice, institutions may constrain or corrupt social practices and demotivate or demoralize practitioners. This is the case when institutions pursue limited goals aimed at external goods (e.g., excessive profits and competition) at the expense of internal goods that motivate practitioners toward the goals of improving the social practice, such as achieving excellence in a craft (making money on watches is not the same as perfecting the skills of watchmaking). One strength of MacIntyre's work is to highlight such distinct differences between social practices and institutions. The role of virtue ethics also be-

[8]MacIntyre (1998a) also raises doubts that management can be considered a social practice because this presumes attention to the creation of internal goods and the well-being of humanity. Managers' concerns are with techniques and systems for transforming raw materials into products, unskilled labor into skilled labor, and investment into profit (p. 30). Managers decide and act to achieve a desired end state through social change, but are blind to other concerns, such as the wider effect of their actions on humanity. Thus, MacIntyre concludes they are not able to engage in moral debates about their own actions. MacIntyre's harsh critique is unwarranted. It neglects the very fact that managers are mostly concerned with the consequences of their actions and repeatedly engage in a wider discourse with society about the nature of their activities and the purpose of the institutions they run. A similar point is made by Dawson and Bartholomew (2003).

comes clear: Without virtues, such as justice, care, courage, and truthfulness, social practices could not resist what he calls the "corrupting power" of institutions (MacIntyre 1981, p. 194).

## Critique and Exegesis

We believe there are great merits to MacIntyre's theory when analyzing human motivation in relation to social practices and institutions. Yet, his writings have received fundamental criticism, which we will discuss briefly here. The following section thus completes our review of MacIntyre's theory with a critique and exegesis. This critical interpretation aims to uncover the significance of the theory for OSS development, and in particular how it informs the three questions we posed in the "Introduction."

First, MacIntyre has been criticized for his claim that habitual patterns of behavior associated with ethical values can no longer be found in contemporary society in the way that Aristotle conceived of them in his idea of *polis* (Bender 1998). MacIntyre gives too much credit to an historical account of virtues, and how they can be achieved by individuals seeking to obtain "unity of life." In so doing, he takes an overly optimistic view of what social practices, and the people within them, can achieve in terms of striving for excellence. Yet writers on OSS remind us that there are strong ethical values at work in the community of developers (Stallman 1999; Zeitlyn 2003). For example, people who are suspected of unlawfully commercializing the software, or of introducing code from a commercial environment, are likely to be "flamed" through verbal attacks and have their privileges (such as access to the formal version of the source code) revoked, because this behavior is not considered consistent with being an OSS developer (Shah 2006). MacIntyre's theory links human behavior and standards of excellence within the social practice to outputs in the form of internal goods. One main concern for the software industry in general, and OSS in particular, is the output of high-quality software (see Aberdour 2007; Gillies 1992; Kan 2002), which is captured by Question 1: Why do OSS developers produce high-quality software when they do?

Second, Macintyre speaks broadly and confidently about managers and firms without seeming to care much about the details of what they do and how they work. While he takes a negative view of institutions (e.g., firms), at the same time he acknowledges "no practices could survive for any length of time unsustained by institutions" (MacIntyre 1981, p. 194). MacIntyre's critique of institutions, in particular firms, is rooted in an overly critical view of modern society. Claiming that firms universally demoralize practitioners, who otherwise seek to do good for their craft and social practice, is naïve and does not take into account the variety of motivations and institutions in existence (see also Brewer 1997). Instead, it is more fruitful to point to the risk that institutions may destroy social practices. For MacIntyre, managers act on behalf of institutions and are primarily concerned with increasing external goods. Yet, the pursuit of external goods clearly does not preclude some kind of balance between internal and external goods (Halliday and Johnsson 2010). In our reading, MacIntyre is overly pessimistic about management's ability to achieve this balance. As we mentioned above, one solution to the problem is to understand managers as belonging to a type of social practice geared at founding and changing institutions (Moore 2008). However, accepting this perspective risks omitting a potentially important inquiry into why OSS developers (rather than managers) contribute to institutional change. What is clear from MacIntyre's analysis is that institutions and social practices are mutually dependent for the production of goods, and it follows that changes in institutions may trigger changes in social practices and vice versa. The mechanisms by which this change functions, and the role of OSS developers, remain unknown but the link between institutions and social practice provides a theoretical basis for approaching Question 2: Why do OSS developers change institutions?

Third, MacIntyre's view of virtue ethics as related to social practices and, more broadly, culture can be criticized from a utilitarian view of moral philosophy. Proponents of the latter view would seek to find universal rules that can guide moral behavior (Adams 1976; Cornman et al. 1982; Mill et al. 1987). For example, one person will help another in distress because helping is a universal rule that provides utility to the person being helped and brings the helper a deep sense of satisfaction. MacInytre's relativistic position neglects the search for such universal rules. In addition, from a utilitarian perspective, the theory defines an ideal unity of life bound to the social practice and the virtues it develops, but lacks objective standards. However, it would also be problematic to ignore entirely that moral premise and actions in everyday life may be shaped by the social context. Indeed, OSS development is a collective undertaking, whether in terms of reuse of software code or adhering to collectively established rules of conduct. Thus, researchers may acknowledge the social practice as the relevant context for OSS development and study rules that might depart from universal rules (say, of the Free Software movement or commercial software development). In studying OSS development as a heterogeneous

phenomenon[9] that contrasts with commercial software development institutions (such as software firms and intellectual property rules), MacIntyre's theory offers a way to think about motivation with a complementary set of assumptions about how and why people act in a particular context (see the next section). This is particularly relevant because of the long-term perspective on the end result of making a contribution to OSS and the unclear rewards associated with engaging in other practice-enhancing activities (see Shah 2006). The shift in the conception of motivation as primarily reward-oriented (as, for example, in self-determination theory) to practice-oriented may guide research into answering Question 3: Why do developers sustain OSS development?

## Toward a New Research Framework and Agenda

The research framework and agenda that we present in this section grows out of the incomplete match between the nature of OSS development and the theoretical underpinnings of reviewed literature on motivation to contribute to OSS development. OSS development is characterized by significant voluntary contributions, self-selection of tasks among developers, clan- and self-control, community-type organization, and strong ethical considerations. Our review identified the dominant role played by self-determination theory in explaining developers' contributions. However, as we discovered, self-determination theory cannot account for several of the intricacies that characterize the OSS development phenomenon. As a result, the domination of the self-determination perspective that runs through the existing motivation literature may compromise our field's ability to adequately address the three research questions posed initially. There is no doubt that self-determination theory does provide very valuable guidance and answers to many important research questions. However, a fundamentally different and complementary approach is required to account for gaps we identified; research on motivation in OSS development neglected important aspects of the social practice as a context for motivation, including the ethics and virtues that guide the work of engineers and software developers (Friedman and Kahn 1994; Latour 1999; Martin 2000, 2002). In short, people are moved to do many things, but whether or not this is constituted by and for the social practice of OSS development, is an entirely different matter.

In this section, we develop a theoretical framework to explain how virtues and ethics—integral elements of a social practice—influence developers' motivations to contribute to OSS. Grounded in the conceptual building blocks discussed in the previous section, we suggest an alternative and complementary set of assumptions about the individual OSS developer, as outlined in Table 3. In particular we emphasize a logic of *unity of life*. Following this logic, individuals do not necessarily act to achieve some immediate reward, because they want to maximize use-value or gain favors (compare this with the self-determination view). Instead, they act to achieve unity of life; they want to reach or maintain consistency of action throughout their life—an ambition that values personal development and contextual events and points beyond the attainment of specific and immediate rewards. We use the metaphor of the journey to the end of the rainbow to describe the idea of *unity of life* because a quest harbors value in itself and can be questioned, interpreted, and reconstructed self-referentially. The journey might be considered worthwhile, exhausting, beautiful, pointless, fulfilling, and many other things—but as a journey it follows a sequence of events limited by the individual life experience and subject to reflection, memory, and expectation. In MacIntyre's (1981) terms, human lives possess narrative structures because they are embedded in social practices. It is from this link that we draw our framework: Individuals are motivated because, through participatory exposure to social practices, they learn what it makes sense to do; and, vice versa, by reflecting on the *unity of life* they shape social practices. This interdependence lies at the core of MacIntyre's (1998a) virtue ethics as it allows for a characterization of the virtues in the interplay of individual life and social practice.

Following this logic, OSS developers are motivated by a consistent stream of actions that we call *unity of life*. Here, it is not the immediate and isolated outcome that matters (the carrot), but how the individual subjectively holds outcomes and actions to be consistent over time (the journey toward the end of the rainbow). We contrast the alternative set of assumptions with the assumptions observed in the reviewed literature. The dominant role of self-determination theory informs the logic of what we call the *self-determination view*.

The reviewed literature on OSS development motivation has not accounted broadly for the insight that social practices strongly relate to people's motivation (Morgeson and Humphrey 2006). As a response, we suggest a *social practice view* of OSS development, where individuals take a long-term, frequently developmental rather than situational per-

[9]There can be major differences, factions, and guilds among OSS developers. As one reviewer helpfully pointed out, there is an "XML guild," for example. Developers often campaign for the "right and best" text editor, programming language, or software license.

**Table 3. Assumptions About the Individual OSS Developer**

| | Self-determination View | Social Practice View |
|---|---|---|
| Output | Product | Good |
| Incentive | Reward | Unity of life, moral obligation |
| Interaction with peers and tasks | Situational, next step, solution-oriented | Developmental, sequential, quest-oriented |
| Quality perception | Use-value | Standards of excellence |
| Time perspective | Short- to mid-term | Long-term, sense for the right time |

spective on social interaction, and pay attention to the importance an activity assumes for unity of life. Living well, in the Aristotelian sense, aims at the right timing of actions, attention to particulars, and aesthetic values (MacIntyre 1981; Nussbaum 1985). The individual views actions, outcomes, and interactions through a "temporal lens" in order to achieve a sequence of events that supports a life well lived. A good life refers to a moral position that results from the accumulation of individual perception and more general principles (Nussbaum 1985, p. 524). The general principles (standards of excellence are of particular interest here) are shared by members of the social practice, and, following MacIntyre, the individual gradually adapts standards of excellence developed in the social practice, changing the institution to become supportive rather than constraining, and producing a "good" in terms of these standards, rather than a product measured in, for example, features or hours spent working. To be clear, a good can only result from work, effort, and activities carried out in a social practice. Goods are collective in production even when only one person carries out coding activities, because coding follows standards of excellence that are defined by the collective (i.e., the social practice). In a social practice view, the individual's ideas of quality extend beyond use-value—the value for users in resolving their own technical problem—to account for what the social practice values and defines as the current state of the art.

To become an OSS developer means to acquire an identity related to a social practice. This means, contrary to the self-determination view, that an individual does not become an OSS developer simply by submitting code to a software repository. Martin (2002, p. 556) suggests:

> These (social) practices enter centrally into defining a way of life, with technological development entering even more centrally into the ways of life of engineers. (The technologies are parts of or aspects of ways of life.)

Identification instills a sense of moral obligation in individuals to support and further develop the social practice. For example, a "moral obligation" in OSS is the early release of software patches to the public, an integral part of the social practice of OSS development (Raymond 1999).

Personal commitment goes beyond commitment to the software product or external goods like profit but extends to the social practice and its broader social effects. From this perspective, OSS development embedded in a social practice becomes meaningful and a way of life with its own pleasure, challenges, and other benefits. Several authors have suggested that self-reflection helps shape practitioners' character (Flyvbjerg 2001; Habermas 1988; MacIntyre 1981). Once developers contribute to the social practice, are motivated by it, and reflect on their actions (Calhoun 1983), they also gain insights about their values in relation to that social practice, for example, whether or not the virtues it values fit their own personal virtues.

An orientation toward a sequence of tasks mirrors MacIntyre's idea about the individual's search for a narrative structure in life, which means that the sum and choice of our activities form a consistent story of our life or the role we see ourselves fulfilling. This can mean that a developer follows a quest that extends beyond the next step in a workflow or the solution to a specific task and accepts high goal ambiguity. Individuals are important actors who participate in building institutions that structure and govern the social practice (Geiger 2009; Whittington 2006) and whose motivations range widely from seeking autonomy, freedom from conventions, creativity, to the wish to change their world (Ketchen et al. 2007; Rindova et al. 2009).[10]

In the social practice view, motivation can be directed at the good—which includes software code produced following standards of excellence defined by the social practice—or at institutional change aimed at supporting the social practice. In either case it is not directly (short-term) reward-oriented but induced through the social practice. Hence, motivation in

[10]Many authors on entrepreneurship therefore understand motivation and social practice as inseparable (Steyaert 2007, p. 467; see also Schatzki 2005).

the social practice view differs from motivation in the self-determination view. According to the social practice view, motivation is intimately linked with a developer's experience of being a member of the social practice of OSS. One important result of this is that if people sense that the social practices they value have become corrupted in some way, for example by institutional goals, they often seek to change institutions.

These two views of motivation are complementary in the study of OSS development. Starting with the premise that the social practice view accounts for at least some individuals' behavior, the framework's theoretical conjectures can be used to formulate concrete propositions about OSS development and contribute to an agenda for future work. We begin the next section by illustrating this with an account of the emergence of the Free Software Foundation.

## On the Emergence of an Institution

In the 1960s and 1970s, much software development was carried out by scientists and engineers in academic and corporate research laboratories. It was a normal part of developers' social practice to give and exchange software they had written, in order to modify and build upon each other's software, both individually and collectively, and in turn to make their modifications freely available. Virtues such as the openness to sharing knowledge actively and intensely were considered important for learning, efficient code development, better bug-free products, and, overall, the development of the software engineering profession.

The emphasis on the virtues of sharing work in software development was very strong among a group of developers at MIT's Artificial Intelligence Laboratory during this period (Levy 1984). The first conflicts between the institutional goals and the virtues of individuals valued by the social practice can be traced back to the 1980s. At this time, MIT decided to license some of the code created by this group to a commercial firm. In accordance with its commercial interests (external goods), the firm restricted access to the source code for that software to exclude the original MIT developers, creating a great deal of frustration and irritation among them. This incident is a good illustration of the kind of conflicts to which MacIntyre draws our attention. Whereas software developers at the time considered virtues such as openness, learning, and knowledge-sharing a prerequisite for the social practice's creation of internal goods (including product excellence, professional development, and benefits for others), they felt institutional concerns for the constraint of external goods, which they feared would eventually harm the practice. Richard Stallman, at the time a programmer at MIT's Artificial Intelligence Laboratory, was distressed by the institutional pressure to restrict access to source code and sell software through licensing. He believed it would harm the software engineering profession and hinder "humanity's rapidly growing need for better and better technologies." Stallman viewed these practices as "morally wrong" impingements upon the rights of software users to learn and create freely. In his own words, faced with the collapse of his community's social practice, he had to make "a stark moral choice" (Stallman 1999, p. 19):

> With my community gone, to continue as before was impossible.…The easy choice was to join the proprietary software world, signing non-disclosure agreements and promising not to help my fellow hackers. Most likely I would also be developing software that was released under a non-disclosure agreement, thus adding to the pressure on other people to betray their fellows too. I could have made money this way, and perhaps amused myself writing code. But I knew at the end of my career, I would look back on years of building walls to divide people, and I feel I had spent my life making the world a worse place....So I looked for a way a programmer could do something for the good. I asked myself, was there a program or programs I could write so as to make the community possible again?

Stallman's response to this challenge was to create an institution as an alternative to the firm, called the Free Software Foundation. The purpose of the foundation was to preserve free access to software developed by people who shared the virtues valued by the social practice. The legal mechanism he developed to support this idea was the GNU General Public License, which can be affixed to a piece of software by a developer, and which guarantees a number of basic rights to all future developers and users. These include the right to download for free, study, and modify the source code, and the right to redistribute to others modified or unmodified versions of the software for free. Stallman firmly believed that this license and the new institution of the Free Software Foundation would support the social practice of software development and eventually help create excellent products of benefit to society. His institutional alternative was created to preserve the social practice's capacity to create internal goods alongside external goods pursued by the software industry. Later observers refer to Stallman's institutional alternative as a "free software" ideology that motivates developers to join and contribute to OSS (Stewart and Gosain 2006). However, the origin of this collective action was the social practice of

software development that motivated Stallman to create an institution. To paraphrase an insight from MacIntyre's theory, anything that does not promote the public good, such as the appropriation of collectively developed software code, may not be properly regarded as social practice.

An internal good can be a practitioner's pursuit of excellence in software development; it is against this backdrop we should understand Stallman's moves. Thus, the new institution of free software sprang out of practitioners' moral considerations, and these motives are shaped by the social practice of software development. Rather than ascribing to the view that the new institution originates in an overarching ideology that motivates people to act, we conclude that the concern for product quality, work, and the wider implications of Free and Open Source software gives rise to new institutions. This conclusion differs from other accounts of OSS, like Moody's *Rebel Code* (2001), or social movements that confront the establishment of the software industry on ideological grounds (see the discussion of ideology and collective action in von Hippel and von Krogh 2003; Stewart and Gosain 2006). Finally, consistent with the theory we develop, if the institutions of free and open source software fail to support social practices that create internal and external goods and enable developers' pursuit of excellence or unity of life, institutional alternatives will emerge that do the job better.

## A Social Practice View on Motivation in OSS Development and Issues for IS Research

In the previous two sections we discussed, first, an alternative set of assumptions about individual OSS developers and, second, how the emergence of new institutions are linked to values central to the social practice of OSS development. Based on this, we now develop a framework that links MacIntyre's conceptual building blocks with motivation. In this motivation–practice framework, we consider OSS development the social practice under investigation. Thus, the motivation to contribute to OSS development can be rephrased as the motivation to contribute to the social practice of OSS development. However, in the MacIntyrian perspective adopted here, it is not meaningful to think of a social practice absent from institutions and goods. Strictly, a social practice cannot exist without producing internal goods; and it cannot exist without supporting institutions that protect its standards of excellence and enable the creation of external goods. Such is the nature of the social practice. Rather, a separation of the social practice from institutions and goods can only be made for purely analytical reasons, a fact we shall profit from when theorizing from our conceptual building blocks.

We postulate that individual-level motivation to *act* is *directly* related to either active participation in the social practice or through attempts to change the supporting institutions of the social practice. Thus, goods (as outcomes and rewards) are not directly impacted by motivation. Rather, the individual is moved by and through the social practice to contribute to the good or to change the institutions that support the social practice. Figure 1 depicts these relationships and shows three theoretical conjectures discussed in detail below. In its most basic form, the framework illustrates that researchers should not implicitly assume a single unidirectional link between motivation, action, and reward. Thus, using the framework, researchers can ask questions previously not considered to belong to the domain of motivation.

The integration of motivation with MacIntyre's conceptual building blocks creates relationships that suggest answers to the three questions stated at the outset of the paper, in the form of theoretical conjectures and concrete propositions. Each of the three questions corresponds to one of the relationships in the framework (numbered in Figure 1). Next, we discuss corresponding conjectures and propositions as well as issues for future research.

### Theoretical Conjecture One

> ***Question 1:*** *How and why do OSS developers produce high-quality software (goods) when they do?*
>
> ***Theoretical Conjecture 1****: OSS developers contribute to the production of internal goods (e.g. high-quality software) when their actions follow the standards of excellence of the social practice. At the same time, the internal good produced by the social practice impacts on standards of excellence pursued in the social practice.*
>
> ***Proposition 1****: Developers in a social practice create a sense of timing and developmental interactions with peers that improves the social practice.*
>
> ***Proposition 2****: The software product impacts on the standards of excellence in OSS development.*

First, quality is an essential element in understanding social practices because its members learn, internalize, and eventually improve the standards of excellence. Developers need to acquire an understanding and appreciation of quality by participating (for some time) in the social practice. In the framework, quality is not globally assessed or measured but is understood as internal to the social practice: it is defined as

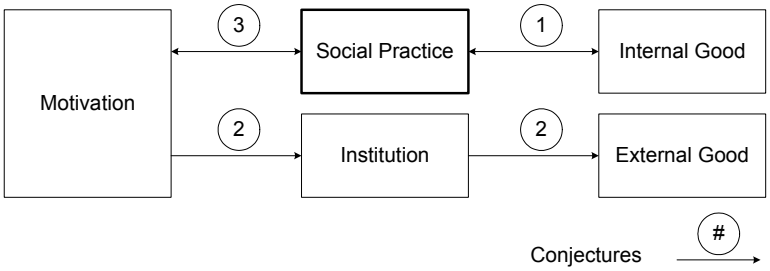


**Figure 1. Relationships Between Motivation and Social Practice, Institution, and Goods**

a characteristic of internal goods that adhere to the standards of excellence in the social practice. Revisiting the assumptions about the individual developer, Proposition 1 fits with the observation that participation in a social practice requires engagement with the practice's goals, a developmental understanding of social relations within the practice that leads to a broader set of activities (e.g., including helping behavior; see Lakhani and von Hippel 2003), and an appreciation of incentives (their timing and moral and aesthetic value) beyond the immediate rewards of an activity. This is why OSS developers devote considerable effort to making code "beautiful," helping other developers to understand and appreciate the nature of "beautiful code," or releasing a new version of software at symbolic points in time. One OSS project, for example, chooses release version numbers that asymptotically approximate the number $\pi$.

In addition, Proposition 2 fits with the observation that OSS developers often receive immediate user feedback on installing, systems compatibility, bugs, license restrictions, etc. (Francalanci and Merlo 2008; Lee and Cole 2003). When developers see how the software product performs on their own and other users' computers, or compares in efficiency with competing products, they may choose to maintain or adjust the standards of excellence in the social practice, as part of learning to practice better. In OSS development, software users often communicate directly with software developers, exemplifying their intent with code patches. The issue tracker of the Firefox web browser, for example, is filled with feature requests that are often accompanied with source code patches implementing these.

The theoretical conjecture also entails a research agenda: What activities are considered good (e.g., code production, training of other developers, improving governance structures, or communication tools)? Collective work also involves mundane tasks that, nevertheless, are taken up voluntarily if need be (Shah 2006). In our perspective, what needs to be done to achieve quality is determined collectively by individuals who "hold one another accountable to what is at issue and at stake in ongoing practices" (Rouse 2007, p. 54). Yet, by which other mechanisms are quality standards collectively determined? Furthermore, what does the perception of quality, informed by standards of excellence, mean for firm–community collaboration? What are the sources of heterogeneity in standards of excellence across OSS communities? These two questions are related because individual contributors (such as the representative of a software firm) help define standards of excellence in their respective communities. In turn, the quality standards in an OSS community might have repercussions within firms that participate in collective development. The extent to which these repercussions lead to conflict or to fruitful interaction between firms and communities is largely unexplored.

### Theoretical Conjecture Two

***Question 2**: Why do OSS developers change institutions?*

***Theoretical Conjecture 2**: OSS developers change institutions when and where these institutions no longer protect sufficiently the standards of excellence of the social practice. In addition, institutions are changed in order to provide external goods that support the social practice.*

***Proposition 3**: Institutions that offer external goods to OSS developers (such as firms participating in OSS and hiring developers) are judged as supportive or constraining by the developers according to the institutions' adherence to the standards of excellence defined in the social practice.*

***Proposition 4**: Under certain conditions, developers of OSS are prepared to sacrifice potential rewards (external goods offered by a current institution) in*

*order to make an institution compatible with the standards of excellence defined by the social practice of OSS.*

Second, motivation may trigger institutional change. The earlier example of Richard Stallman founding the Free Software Foundation illustrated a form of motivation geared not at immediate rewards but at the protection of a social practice that achieves internal good. This can be labeled *ethical* or *ideological* and makes perfect sense when viewed from the perspective of the social practice. Institutional change was necessary to protect the social practice from extinction, given that newly formed software companies hired away Stallman's community of peers and threatened to change radically the way software was developed. The changes Stallman and his peers implemented in the way software could be licensed represented an institutional change. The license regime regulates the exchange of software among developers. The exchange is part of the social practice and free exchange is considered good practice in OSS development. MacInytre focuses on virtues that extend beyond the short-term interests of individuals giving social practices an enduring character worthy of analysis in their own rights (see also Halliday and Johnsson 2010). In brief, we propose that these enduring characteristics of the social practice give rise to institutional change via the motivation of developers.

When institutions, such as firms, cannot sustain social practices sought by virtuous software developers, the latter will seek to change or develop new institutions that serve better the internal goods of the social practice. Proposition 3 builds on the argument that practitioners will judge institutions by their adherence to the standards of excellence of the social practice because a firm, for example, that contributes to OSS development engages in the social practice and offers external goods (such as salaries and career opportunities). As an output of institutions, external goods need to enable standards of excellence of the social practice that the institution supports. This could mean that salaries or promotions within the firm need to be perceived as fair according to the standards of excellence in OSS development.

In the motivation–practice framework, institutions serve the social practice and not the other way around. The priority of the social practice over institutions implies that individuals will sacrifice potential rewards offered by an institution (such as a firm hiring OSS developers and offering a salary) in favor of adhering to standards of excellence, as Proposition 4 states. Thus, the "good and right thing to do," according to the social practice's standards of excellence, may be to change the institution and risk losing external goods in favor of internal goods if standards of excellence are threatened, as the example of Stallman illustrates. This could mean that a developer would lobby for policy changes, implement corporate restructuring, or leave a well-paid job in a software firm.

Scholars currently cannot explain when institutional change is called for in more general and systematic terms, which opens a number of related research questions. For example: How are conflicts between extrinsic motivation and moral obligations resolved when current institutions impose a choice between the two (as in Stallman's case)? Under what conditions do the assumptions about motivation from the social practice view hold and account for institutional change? How can organizations accommodate individuals prepared to engage in institutional change in order to pursue standards of excellence? And related to this, how can organizations drive institutional change with the motivation of developers?

## Theoretical Conjecture Three

***Question 3**: Why do developers sustain the social practice of OSS development?*

***Theoretical Conjecture 3**: OSS developers sustain the social practice of OSS development because social practice instills the motivation to uphold its standards of excellence over time.*

***Proposition 5**: The motivation to contribute becomes stronger during developers' tenure in an OSS community and by their contribution to the social practice.*

***Proposition 6**: Through sustained contributions to the social practice, developers become motivated to contribute beyond code patches to educate and help others, and take on tasks that support the internal good of the social practice.*

Third, how does a social practice influence motivation and vice versa? Perhaps surprisingly, this question does not appear to be prominent in MacIntyre's work. Virtues are transmitted to individuals through socialization and collective work and exercised within social practices by individuals who choose activities that appear right in their life. They choose to sustain practices that cultivate internal goods that match their individual sense of the common good and the life they wish to live. Entry and exit represent the dynamics of these individual choices. Organizations continuously face the entry and exit of individuals who choose freely how and where to enact their profession. It follows from our argument that motivations change with the context of the social practice. A

first entry into a community by making a contribution to OSS is motivated differently from a sustained contribution to the same practice: Shah (2006) observes that mundane tasks tend to be accomplished by long-term members of communities. This can be interpreted, in line with our Propositions 5 and 6, as a growing understanding of the internal goods of the social practice and a stronger motivation to sustain contributions. Developers gradually develop an understanding and appreciation of the larger needs of the practice and are motivated to contribute to necessary tasks that might appear mundane to the outside observer. Tasks such as bug fixing, maintenance of software, or helping new users are, however, essential for the social practice to function and produce high-quality software, perhaps understood even better by seasoned developers than newcomers. A flourishing OSS community succeeds in creating this understanding with its members and in motivating them to see these activities as core to the production of internal goods.

Future research should uncover more detail about mutual influence. The motivation to contribute to a social practice may change in quality as well as in direction and open questions relate to both possibilities. For example, could a loss of interest in the internal good and subsequent exit be triggered by conflicts over incommensurable moral premises? The emergence of a proprietary software industry, as witnessed by Stallman, made him the architect of a new software license that regulates the free exchange of software. When do competing views of what constitutes a good practice jeopardize developers' long-term motivation? Under what conditions does an initial interest grow or wane over time, given exposure to the social practice, learning, social interaction, moral premises, help received, or a sense of reciprocity in contributions to OSS?

One individual's mark on the social practice may turn its course of action or significantly extend its standards of excellence. According to the definition we discussed above, a social practice is coherent and complex, two characteristics that might be explained by the content of the work and the history of the internal discussions of developers about the good it achieves. However, the social practice depends on individuals who carry it forward, understand its complex history and possibly incoherent tendencies, and correct and sustain the social practice going forward. This is easier said than done. So far, research on OSS has not contributed much to the understanding of individual motivation beyond the contribution of code itself, which raises a number of future research questions. For example, why do individuals found new or fork (split up) existing OSS communities? Why do we observe a wide heterogeneity of communities and philosophies within OSS (Himanen 2001; Moody 2001)? There are, for example, distinct differences in ideology and values between pragmatic OSS enthusiasts and the Free Software Foundation (Stewart and Gosain 2006). The wealth of licenses and approaches to similar problems calls for more research on why individuals choose different paths and subtle changes over the established wisdom found in some social practice (developers are motivated to build institutions, not only code). Given the chance, OSS developers may diverge, follow their own ideas, and realize similar if not identical solutions in different ways. Thus, what are the proper time intervals to study motivation if one is to consider motivation in which developers seek *unity of life*? Another interesting topic worth exploring is the differences between paid and unpaid developers. In the motivation–practice framework, paid developers partly receive monetary compensation provided by institutions, whereas unpaid developers need to seek other means of compensation. Future research could, for example, explore relationships between monetary compensation and behavior within the social practice in order to identify potential threats to the sustenance of the social practice, beyond the contributions of individual developers. Finally, our review of motivation to contribute to OSS reveals a bias toward assumptions originating in self-determination theory, and prompts the development of a complimentary set of assumptions that we associate with a social practice view, building on the theory of MacIntyre. The assumptions underlying the social practice view raise the empirical question of the actual distribution of motivation types in a population. Future research may study the distribution of the two "ideal types" of individual in a distribution of OSS developers.[11]

## Conclusion

The objective of this paper is to provide a state-of-the-art review of the study of motivation to contribute to OSS development, and reinvigorate the research field by providing a new theoretical framework with propositions. We identify a large body of work examining types of motivation that lead developers to contribute their time and effort to the development of OSS. However, we argue that both the antecedents and consequences of motivation are more extensive and complex than the present level of theorizing and empirical research has suggested.

[11]Methodologically, this could be done analogously to the study of social preferences in behavioral economics—for example, as Fehr and Schmidt (1999) have done in their experimental studies to determine the distribution of inequity-averse individuals in a population of students.

OSS development differs from conventional software development along three dimensions (incentives, control, and coordination mechanisms), which in turn are reflective of a distinct social practice in which ethics plays a central role. Because of these differences, we argued that individual motivation should not be looked at in isolation. Instead, scholars should expand theory building and research to cover the interplay with institutions, goods, and the social practice. Following a trend to explore the role of social practices in the IS field (Kellogg et al. 2006; Orlikowski 2000), and the social sciences more generally (Schatzki 2005), we develop a theoretical framework around the conceptual building blocks of social practice, institutions, and goods. This new motivation–practice framework is based on the theory of Alisdair MacIntyre (1981, 1998a) and includes ethical considerations of social practice. The framework's set of theoretical conjectures and accompanying propositions provide answers to the three questions identified in the beginning of the paper and point to future research opportunities.

While the main contribution of the paper is the new motivation–practice framework, we also (1) formulate implications for IS management, (2) discuss the relationship between institutional change and social practice, and (3) tie implications from the framework back to the self-determination perspective in motivation studies.

Several implications for IS management can be extracted. First, standards of excellence emerging in global communities of software developers can gain broad endorsement and impact quality standards expected by users and customers. For example, peer review and quick feedback loops in OSS are said to lead to better quality. Installing a culture of "doing things properly," rather than quickly hacking around bugs, may ultimately result in better code quality. Finally, established tools developed for and within OSS can be employed by any software development project. Hence, close observation of the standards of excellence that characterize OSS development may pay off also for software firms that do not actively participate in OSS development.

Second, software developers collectively account for and evolve the virtues in social practice. MacIntyre (1981) refers to the social practice as a "school of virtue," because individual learning is shaped by and informs the collective appropriate conduct regarding ways of developing software. Participating in OSS development could thus provide a valuable training ground for software engineers, in so far as the firm sees fit and is prepared to comply with the ethical considerations in the social practice. In this regard, the new framework informs managers about the relationship between OSS and proprietary software development. OSS does not necessarily exist in competition to proprietary software, but rather complements it; it secures the social practices through which certain standards of excellence in software development can be further nurtured. Software firms are actively looking for developers who have "cut their teeth" in OSS communities; firms such as Intel or Red Hat recruit developers whose skills and standards of excellence stand out.

Third, as shown in the framework, incommensurable moral premises may collide within a firm when developers adhere to different traditions yet collaborate on the same projects. In a social practice perspective, one may identify and preemptively solve looming conflicts over technology, standards, rules, and routines in software firms. However, the activity of software development often builds on and refers to earlier work and developers and the firms they represent may be held accountable by other OSS developers for the work they submit to OSS projects. Questions of compatibility and compliance with OSS development play an increasingly important role for those firms that develop both proprietary and open source software.

Fourth, a related issue is that the use of incentives and control in software firms may need careful tailoring to fit with developers' motivations to contribute to OSS development. Monetary incentives are, of course, compatible with OSS development (Roberts et al. 2006), but social practices can instill loyalty and lead to motivation for institutional changes that perhaps question the efficiency of such incentives.

Fifth, developers can be motivated to change institutions in order to support their practices. This may apply to firms, too. For example, IBM founded the nonprofit Eclipse Foundation as a neutral steward for technology that IBM had initially developed, and created formal membership roles in which IBM was only one among equals, in order to facilitate external contributions of other firms and individuals. Intel and Oracle have founded institutions such as the Open Innovation Network, which grants mutual licenses and a non-litigation agreement for certain OSS technologies to all of its members.

The theoretical framework developed here also opens up a new view on the relationship between institutional change and social practices. Other perspectives on the emergence of institutions (for a review, see Hargrave and Van de Ven 2006) argue convincingly that the struggle between factions and social movements brings about change and institutional innovation. However, framing and political struggles may capture neither the full diversity of people's motivations to contribute nor the role technology might play in this process. As the case of the Free Software Foundation demonstrates, notions of quality that drive social practice to generate and

maintain its standards of excellence can be powerful forces for change. These views are compatible, in that a social practice of OSS development can become a social movement (Hertel et al. 2003; von Hippel and von Krogh 2003). They are also complementary, in that collective action emphasizes the "struggle over meanings of new issues and technologies" (Hargrave and Van de Ven 2006, p. 884) across social movements, whereas social practice focuses on the quest for higher quality products, more profound knowledge, and improved collaboration in creating internal goods.

Collectively, we have merely begun to scratch the surface of a full exploration of developers' motivations for contributing to OSS development. There are vast research opportunities in all areas covered in this theory and review piece. The motivations across OSS projects could, for example, be investigated productively using self-determination theory. Three topics in particular could stimulate interesting research from a self-determination perspective. First, future research can incorporate ethical considerations in self-determination models, since ethics is integral to the OSS phenomenon per se. Self-determination theory would suggest the existence of both intrinsic and extrinsic ethical motives. Disentangling these ethical motives and linking them to types and levels of involvement would be an important contribution. The motivation–practice framework, in turn, suggests that ethical considerations are geared toward the virtues and standards of excellence of the social practice. For example, self-determination research could fruitfully explore how developers are intrinsically motivated to maintain standards of excellence. Again, the motivation–practice framework is complementary since it explains how standards of excellence are formed collectively through active participation in the social practice over time.

Second, our argument that social practice moves individuals to act may inform future applications of self-determination theory in two ways. Ethical considerations may not only directly motivate individuals to act, but also condition what is self-determined. Cultural analysis in social sciences recently shifted to treating culture "as constitutive of a wide range of social processes rather than a regulative that works against other forces, such as interests or rationality" (Weber and Dacin 2011, p. 287; see also Boltanski and Thévenot 2006). Motivation studies using self-determination theory should explore if and to what extent a contextual and cultural substrate impacts the confines of what individuals experience as self-determined action. Further, the social practice perspective suggests a carefully paced integration of temporal aspects to context when designing studies using self-determination theory. In other words, the perspective points to potential changes in motivation over time triggered by exposure to and integration in a specific context. Self-determination models in the reviewed literature treated context as developer-external and static and thus not accountable for motivational dynamics. For example, self-determination research could investigate motivational differences between groups of developers with different lengths and types of contextual exposure to OSS projects.

Third, the understanding of how OSS institutions (e.g., standards, licenses, governance, copyright, and IP assignments) and supporting umbrella organizations relate to developers' self-determined motivations is only beginning to emerge and many questions remain unanswered. With the growth of social networking sites and online communities (Wiertz and de Ruyter 2007; Wasko and Faraj 2005), countless other practitioners apart from software developers rely on volunteer contributions and institutions such as open content licenses. Obviously, motivation research on the OSS phenomenon may inform research designs in other domains, and vice versa.

To conclude, while there is ample room to investigate motivation in OSS from a multitude of perspectives and methodological approaches, we believe the greatest research opportunities lie in those questions found at the intersection between social practices and institutions, against which individual motivations can and should be understood. Figuring out what moves people, we should start with the assertion that people's pursuit of visible carrots is at times interrupted by the larger quest for the invisible gold at the end of the rainbow.

## Acknowledgments

The authors would like to thank the editorial team of *MIS Quarterly* for guidance and developmental feedback that helped tremendously in improving the manuscript. Conference audiences at the University of Gothenburg, Harvard Business School, Copenhagen Business School, Grenoble Ecole de Management, Politecnico di Milano, Bocconi University, and Università degli Studi di Trento offered valuable comments. This research project received support from the Swiss National Science Foundation (Grant 100014_125513). All authors contributed equally.

## References

Aberdour, M. 2007. "Achieving Quality in Open Source Software," *IEEE Software* (24:1), pp. 58-64.

Adams, R. M. 1976. "Motive Utilitarianism," *The Journal of Philosophy* (73:14), pp. 467-481.

Alexy, O., and Leitner, M. 2011. "A Fistful of Dollars: Are Financial Rewards a Suitable Management Practice for Dis-

tributed Models of Innovation?," *European Management Review* (8:3), pp. 165-185.

Baldwin, C. Y., and Clark, K. B. 2006. "The Architecture of Participation: Does Code Architecture Mitigate Free Riding in the Open Source Development Model?," *Management Science* (52:7), pp. 1116-1127.

Beadle, R., and Moore, G. 2006. "MacIntyre on Virtue and Organization," *Organization Studies* (27:3), pp. 323-340.

Bender, R. 1998. "The Aesthetics of Ethical Reflection and the Ethical Significance of Aesthetic Experience: A Critique of Alasdair MacIntyre and Martha Nussbaum," *Erfurt Electronic Studies in English* (http://webdoc.gwdg.de/edoc/ia/eese/).

Benkler, Y. 2002. "Coase's Penguin, or Linux and the Nature of the Firm," *Yale Law Journal* (112:3), pp. 369-447.

Bergquist, M., and Ljungberg, J. 2001. "The Power of Gifts: Organizing Social Relationships in Open Source Communities," *Information Systems Journal* (11:4), pp. 305-320.

Bitzer, J., Schrettl, W., and Schröder, P. J. 2007. "Intrinsic Motivation in Open Source Software Development," *Journal of Comparative Economics* (35:1), pp. 160-169.

Blackledge, P. 2009. "Alasdair MacIntyre: Social Practices, Marxism and Ethical Anti-Capitalism," *Political Studies* (57:4), pp. 866-884.

Boltanski, L., and Thévenot, L. 2006. *On Justification: Economies of Worth*, Princeton, NJ: Princeton University Press.

Bonaccorsi, A., Giannangeli, S., and Rossi, C. 2006. "Entry Strategies Under Competing Standards: Hybrid Business Models in the Open Source Software Industry," *Management Science* (52:7), pp. 1085-1098.

Bonaccorsi, A., and Rossi, C. 2003. "Why Open Source Software Can Succeed," *Research Policy* (32:7), pp. 1243-1258.

Bonaccorsi, A., and Rossi, C. 2006. "Comparing Motivations of Individual Programmers and Firms to Take Part in the Open Source Movement: From Community to Business," *Knowledge, Technology, and Policy* (18:4), pp. 40-64.

Boyd, D., and Ellison, N. 2007. "Social Network Sites: Definition, History, and Scholarship," *Journal of Computer-Mediated Communication* (13:10), pp. 210-230.

Brewer, K. B. 1997. "Management as a Practice: A Response to Alasdair MacIntyre," *Journal of Business Ethics* (16:8), pp. 825-833.

Brown, J. S., and Duguid, P. 1991. "Organizational Learning and Communities-of-Practice," *Organization Science* (2), pp. 40-57.

Calhoun, C. J. 1983. "The Radicalism of Tradition: Community Strength or Venerable Disguise and Borrowed Language?," *American Journal of Sociology* (88:5), pp. 886-914.

Comino, S., Rossi, A., and Manenti, F. M. 2010. "Public Intervention for Free/Open Source Software," available at SSRN: http://ssrn.com/abstract=1628566.

Cooper, H. 1998. *Synthesizing Research* (3rd ed.), Thousand Oaks, CA: Sage Publications.

Cornman, J. W., Lehrer, K., and Pappas, G. S. 1982. *Philosophical Problems and Arguments: An Introduction* (3rd ed.), Indianapolis, IN: Hackett Publishing.

Cusumano, M. A., and Selby, R. W. 1995. *Microsoft Secrets: How the World's Most Powerful Software Company Creates Technology, Shapes Markets, and Manages People*, New York: Free Press.

Dahlander, L., and Wallin, M. W. 2006. "A Man on the Inside: Unlocking Communities as Complementary Assets," *Research Policy* (35:8), pp. 1243-1259.

David, P. A., and Shapiro, J. S. 2008. "Community-Based Production of Open-Source Software: What Do We Know About the Developers Who Participate?," *Information Economics and Policy* (20:4), pp. 364-398.

David, P. A., Waterman, A., and Arora, S. 2003. *FLOSS-US: The Free/Libre/Open Source Software Survey for 2003*, Stanford Institute for Economic Policy Research, Stanford University, Stanford, CA (http://www.stanford.edu/group/floss-us/report/FLOSS-US-Report.pdf).

Dawson, D., and Bartholomew, C. 2003. "Virtues, Managers and Business People: Finding a Place for MacIntyre in a Business Context," *Journal of Business Ethics* (48:2), pp. 127-138.

Deci, E. L., and Ryan, R. M. 1985. *Intrinsic Motivation and Self-Determination in Human Behavior*, New York: Plenum Press.

Deci, E. L., and Ryan, R. M. 1987. "The Support of Autonomy and the Control of Behavior," *Journal of Personality and Social Psychology* (53:6), pp. 1024-1037.

Dunne, J. 1992. *Back To the Rough Ground: "Phronesis" and "Techne" in Modern Philosophy and in Aristotle* (1st ed.), Notre Dame, IN: University of Notre Dame Press.

Fehr, E., and Schmidt, K. M. 1999. "A Theory of Fairness, Competition, and Cooperation," *Quarterly Journal of Economics* (114:3), pp. 817-868.

Feldman, M. S., and Orlikowski, W. J. 2011. "Theorizing Practice and Practicing Theory," *Organization Science* (22:5), pp. 1240-1253.

Feller, J., and Fitzgerald, B. 2002. *Understanding Open Source Software Development*, Boston: Addison-Wesley.

Fershtman, C., and Gandal, N. 2007. "Open Source Software: Motivation and Restrictive Licensing," *International Economics and Economic Policy* (4:2), pp. 209-225.

Flyvbjerg, B. 2001. *Making Social Science Matter* (1st ed,), New York: Cambridge University Press.

Fowers, B. J. 2003. "Reason and Human Finitude: In Praise of Practical Wisdom," *American Behavioral Scientist* (47:4), pp. 415-426.

Francalanci, C., and Merlo, F. 2008. "Empirical Analysis of the Bug Fixing Process in Open Source Projects," in *Open Source Development, Communities and Quality*, B. Russo, E. Damiani, S. Hissam, B. Lundell, and G. Succi (eds.), Boston, MA: Springer, pp. 187-196.

Frey, B. S., and Jegen, R. 2001. "Motivation Crowding Theory," *Journal of Economic Surveys* (15), pp. 589-610.

Friedman, B., and Kahn, P. H. 1994. "Educating Computer Scientists: Linking the Social and the Technical," *Communications of the ACM* (37:1), pp. 64-70.

Gagné, M., and Deci, E. L. 2005. "Self-Determination Theory and Work Motivation," *Journal of Organizational Behavior* (26:4), pp. 331-362.

Geiger, D. 2009. "Revisiting the Concept of Practice: Toward an Argumentative Understanding of Practicing," *Management Learning* (40:2), pp. 129-144.

Ghosh, R. A. 2005. "Understanding Free Software Developers: Findings from the FLOSS Study," in *Perspectives on Free and Open Source Software*, J. Feller, B. Fitzgerald, S. A. Hissam, and K. R. Lakhani (eds.), Cambridge, MA: MIT Press, pp. 23-46.

Gillies, A. C. 1992. *Software Quality: Theory and Management*, London: Chapman & Hall.

Habermas, J. 1988. *On the Logic of the Social Sciences*, S. Weber Nicholsen and J. A. Stark (trans.), Cambridge, MA: MIT Press.

Halliday, J., and Johnsson, M. 2010. "A MacIntyrian Perspective on Organizational Learning," *Management Learning* (41:1), pp. 37-51.

Hargrave, T., and Van de Ven, A. H. 2006. "A Collective Action Model of Institutional Innovation," *Academy of Management Review* (31:4), pp. 864-888.

Hars, A., and Ou, S. 2002. "Working for Free? Motivations for Participating in Open-Source Projects," *International Journal of Electronic Commerce* (6), pp. 25-39.

Haruvy, E., Prasad, A., and Sethi, S. 2003. "Harvesting Altruism in Open-Source Software Development," *Journal of Optimization Theory and Applications* (118:2), pp. 381-416.

Hemetsberger, A. 2004. "Fostering Cooperation on the Internet: Social Exchange Processes in Innovative Virtual Consumer Communities," extended abstract published in *Advances in Consumer Research* (Vol. 29), S. M. Broniarczyk and K. Nakamoto (eds.), Valdosta, GA: Association for Consumer Research, pp. 354-356.

Henkel, J. 2006. "Selective Revealing in Open Innovation Processes: The Case of Embedded Linux," *Research Policy* (35:7), pp. 953-969.

Hertel, G. 2002. "Management Virtueller Teams auf der Basis Sozialpsychologischer Modelle," in *Sozialpsychologie Wirtschaftlicher Prozess*, E. H. Witt (ed.), Lengerich, Germany: Pabst Publishers, pp. 172-202.

Hertel, G., Niedner, S., and Herrmann, S. 2003. "Motivation of Software Developers in Open Source Projects: An Internet-Based Survey of Contributors to the Linux Kernel," *Research Policy* (32:7), pp. 1159-1177.

Himanen, P. 2001. *The Hacker Ethic and the Spirit of the Information Age*, London: Secker & Warburg.

Kan, S. 2002. *Metrics and Models in Software Quality Engineering*, Boston: Addison-Wesley.

Ke, W., and Zhang, P. 2008. "Motivations for Participating in Open Source Software Communities: Roles of Psychological Needs and Altruism," in *Proceedings of the 12th Pacific Asia Conference on Information Systems*, Suzhou, China, pp. 136-148.

Kellogg, K. C., Orlikowski, W. J., and Yates, J. 2006. "Life in the Trading Zone: Structuring Coordination Across Boundaries in Postbureaucratic Organizations," *Organization Science* (17:1), pp. 22-44.

Ketchen Jr., D. J., Ireland, R. D., and Snow, C. C. 2007. "Strategic Entrepreneurship, Collaborative Innovation and Wealth Creation," *Journal of Strategic Entrepreneurship* (1), pp. 371-385.

King, L. 2010. "London Stock Exchange Linux Record Breaking System Faces New Challengers," *Computerworld UK*, November 10 (available at: http://www.computerworlduk.com/in-depth/open-source/3246835/london-stock-exchange-linux-record-breaking-system-faces-new-challengers/; accessed March 30, 2011).

Kirsch, L. J. 1996. "The Management of Complex Tasks in Organizations: Controlling the Systems Development Process," *Organization Science* (7:1), pp. 1-21.

Klandermans, B. 1997. *The Social Psychology of Protest*, Oxford, UK: Basil Blackwell.

Knight, K. 1998. *The MacIntyre Reader*, Notre Dame, IN: University of Notre Dame Press.

Lakhani, K. R., and von Hippel, E. 2003. "How Open Source Software Works: 'Free' User-to-User Assistance," *Research Policy* (32:6), pp. 923-943.

Lakhani, K. R., and Wolf, R. G. 2005. "Why Hackers Do What They Do: Understanding Motivation and Effort in Free/Open Source Software Projects," in *Perspectives on Free and Open Source Software*, J. Feller, B. Fitzgerald, S. Hissam, and K. R. Lakhani (eds.), Cambridge, MA: MIT Press, pp. 3-22.

Latour, B. 1996. *Aramis, or the Love of Technology*, Boston: Harvard University Press.

Latour, B. 1999. *Pandora's Hope: Essays on the Reality of Science Studies* (1st ed.), Boston: Harvard University Press.

Lattemann, C., and Stieglitz, S. 2005. "Framework for Governance in Open Source Communities," in *Proceedings of the 38th Annual Hawaii International Conference on System Sciences*, Los Alamitos, CA: IEEE Computer Society Press, p. 192b.

Lee, G. K., and Cole, R. E. 2003. "From a Firm-Based to a Community-Based Model of Knowledge Creation: The Case of the Linux Kernel Development," *Organization Science* (14:6), pp. 633-649.

Lerner, J., and Tirole, J. 2002. "Some Simple Economics of Open Source," *Journal of Industrial Economics* (50:2), pp. 197-234.

Levy, S. 1984. *Hackers: Heroes of the Computer Revolution*, New York: Anchor Press/Doubleday.

Lindenberg, S. M. 2001. "Intrinsic Motivation in a New Light," *Kyklos* (54:2/3), pp. 317-342.

Luthiger, B., and Jungwirth, C. 2007. "Pervasive Fun," *First Monday* (12:1), January (available at http://firstmonday.org/htbin/cgiwrap/bin/ojs/index.php/fm/article/view/1422).

MacIntyre, A. C. 1981. *After Virtue: A Study in Moral Theory* (1st ed.), Notre Dame, IN: University of Notre Dame Press.

MacIntyre, A.C. 1994. "Interview with Professor Alasdair MacIntyre," *Kinesis* (20), pp. 34-47.

MacIntyre, A. C. 1998a. "The Claims of After Virtue," in *The Macintyre Reader*, K. Knight (ed.), Notre Dame, IN: University of Notre Dame Press, pp. 69-72.

MacIntyre, A. C. 1998b. "Politics, Philosophy and the Common Good," K. Knight (ed.), Notre Dame, IN: University of Notre Dame Press, pp. 235-252.

MacIntyre, A. C. 1999. *Dependent Rational Animals: Why Human Beings Need the Virtues*, Chicago: Open Court Publishing Company.

Mackenzie, A. 2005. "The Performativity of Code: Software and Cultures of Circulation," *Theory, Culture & Society* (22:1), pp. 71-92.

Maldonado, E. 2010. "The Process of Introducing FLOSS in the Public Administration: The Case of Venezuela." *Journal of the Association for Information Systems* (11:11), pp. 756-783.

Markus, M. L. 2007. "The Governance of Free/Open Source Software Projects: Monolithic, Multidimensional, or Configurational?," *Journal of Management and Governance* (11:2), pp. 151-163.

Martin, M. W. 2000. *Meaningful Work: Rethinking Professional Ethics*, Oxford, UK: Oxford University Press.

Martin, M. W. 2002. "Personal Meaning and Ethics in Engineering," *Science and Engineering Ethics* (8:4), pp. 545-560.

McKeon, R. (ed.). 1941. *The Basic Works of Aristotle*, New York: Random House.

Mill, J. S., Bentham, J., and Ryan, A. 1987. *Utilitarianism and Other Essays*, New York: Penguin Classics.

Mingers, J., and Walsham, G. 2010. "Toward Ethical Information Systems: The Contribution of Discourse Ethics." *MIS Quarterly* (34:4), pp. 833-854.

Moody, G. 2001. *Rebel Code: Linux and the Open Source Revolution*, New York: Penguin.

Moore, G. 2008. "Re-imagining the Morality of Management: A Modern Virtue Ethics Approach," *Business Ethics Quarterly* (18:4), pp. 483-511.

Morgeson, F. P., and Humphrey, S. E. 2006. "The Work Design Questionnaire (WDQ): Developing and Validating a Comprehensive Measure for Assessing Job Design and the Nature of Work," *Journal of Applied Psychology* (91:6), pp. 1321-1339.

Morris, A. D., and Mueller, C. M. 1992. *Frontiers in Social Movement Theory*, New Haven, CT: Yale University Press.

Noel, J. 1999. "Phronesis and Phantasia: Teaching with Wisdom and Imagination," *Journal of the Philosophy of Education* (33:2), pp. 277-286.

Nussbaum, M. 1985. "Finely Aware and Richly Responsible: Moral Attention and the Moral Task of Literature," *The Journal of Philosophy* (82:10), pp. 516-529.

Okoli, C., and Oh, W. 2007. "Investigating Recognition-Based Performance in an Open Content Community: A Social Capital Perspective," *Information & Management* (44:3), pp. 240-252.

O'Mahony, S. C., and Ferraro, F. 2007. "The Emergence of Governance in an Open Source Community," *Academy of Management Journal* (50:5), pp. 1079-1106.

Oreg, S., and Nov, O. 2008. "Exploring Motivations for Contributing to Open Source Initiatives: The Roles of Contribution Context and Personal Values," *Computers in Human Behavior* (24:5), pp. 2055-2073.

Orlikowski, W. J. 2000. "Using Technology and Constituting Structures: A Practice Lens for Studying Technology in Organizations," *Organization Science* (11:4), pp. 404-428.

Orlikowski, W. J. 2005. "Material Works: Exploring the Situated Entanglement of Technological Performativity and Human Agency," *Scandinavian Journal of Information Systems* (17:1), pp. 183-186.

Orlikowski, W., and Scott, S. 2008. "Sociomateriality: Challenging the Separation of Technology, Work and Organization," *The Academy of Management Annals* (2), pp. 433-474.

Osterloh, M., and Rota, S. G. 2007. "Open Source Software Development—Just Another Case of Collective Invention?," *Research Policy* (36:2), pp. 157-171.

Ouchi, W. G. 1978. "The Transmission of Control through Organizational Hierarchy," *The Academy of Management Journal* (21:2), pp. 173-192.

Ouchi, W. G. 1979. "A Conceptual Framework for the Design of Organizational Control Mechanisms," *Management Science* (25:9), pp. 833-848.

Ouchi, W. G. 1980. "Markets, Bureaucracies, and Clans," *Administrative Science Quarterly* (25:1), pp. 129-141.

Peters, L. 2003. "Managing Software Professionals," in *Proceedings of the 2003 Engineering Management Conference: Managing Technologically Driven Organizations: The Human Side of Innovation and Change*, Troy, NY, November 2-4, pp. 61-66.

Rämö, H. 2004. "Spatio-Temporal Notions and Organized Environmental Issues: An Axiology of Action," *Organization* (11), pp. 849-872.

Raymond, E. S. 1999. "The Cathedral and the Bazaar," in *The Cathedral & the Bazaar*, Sebastapol, CA: O'Reilly & Associates, Inc., pp. 19-64.

Riehle, D. 2007. "The Economic Motivation of Open Source Software: Stakeholder Perspectives," *Computer* (40:4), pp. 25-32.

Rindova, V., Barry, D., and Ketchen Jr., D. J. 2009. "Entrepreneuring as Emancipation," *Academy of Management Review* (34:3), pp. 477-491.

Roberts, J. A., Hann, I., and Slaughter, S. A. 2006. "Understanding the Motivations, Participation, and Performance of Open Source Software Developers: A Longitudinal Study of the Apache Projects," *Management Science* (52:7), pp. 984-999.

Robey, D., and Markus, M. L. 1984. "Rituals in Information System Design," *MIS Quarterly* (8:1), pp. 5-15.

Rolandsson, B, Bergquist, M., and Ljungberg, J. 2011. "Open Source in the Firm: Opening Up Professional Practices of Software Development," *Research Policy* (40:4), pp. 576-587.

Rouse, J. 2007. "Social Practices and Normativity," *Philosophy of the Social Sciences* (37:1), pop. 46-56.

Rullani, F. 2007. "Dragging Developers Towards the Core: How the Free/Libre/Open Source Software Community Enhances Developers' Contribution," IFIP Working Group 2.13 (http://ifipwg213.org/system/files/rullani_socenv.pdf).

Scacchi, W. 2002. "Understanding Requirements for Developing Open Source Software Systems," *IEEE Proceedings - Software* (149:1), pp. 24-39.

Schatzki, T. R. 2005. "Peripheral Vision: The Sites of Organizations," *Organization Studies* (26:3), pp. 465-484.

Schatzki, T. R., Knorr-Cetina, K., and Savigny, E. V. 2001. *The Practice Turn in Contemporary Theory* (1st ed.), London: Routledge.

Schofield, A., and Cooper, G. S. 2006. "Participation in Free and Open Source Communities: An Empirical Study of Community Members' Perceptions." in *Open Source Systems* (IFIP Working Group 2.13), Berlin: Springer, pp. 221-231.

Shah, S. K. 2006. "Motivation, Governance, and the Viability of Hybrid Forms in Open Source Software Development," *Management Science* (52:7), pp. 1000-1014.

Sheldon, K. M., and Krieger, L. S. 2007. "Understanding the Negative Effects of Legal Education on Law Students: A Longitudinal Test of Self-Determination Theory," *Personality and Social Psychology Bulletin* (33:6), pp. 883-897.

Spaeth, S., Haefliger, S., von Krogh, G., and Renzl, B. 2008. "Communal Resources in Open Source Software Development," *Information Research* (13:1) (http://informationr.net/ir/13-1/paper332.html).

Stallman, R. 1999. "The GNU Operating System and the Free Software Movement," in *Open Sources: Voices from the Open Source Revolution*, C. DiBona, S. Ockman, and M. Stone (eds.), Sebastopol, CA: O'Reilly & Associates, Inc., pp. 53-70.

Stam, W. 2009. "When Does Community Participation Enhance the Performance of Open Source Software Companies?," *Research Policy* (38:8), pp. 1288-1299.

Stewart, K. J., Ammeter, A. P., and Maruping, L. M. 2006. "Impacts of License Choice and Organizational Sponsorship on User Interest and Development Activity in Open Source Software Projects," *Information Systems Research* (17:2), pp. 126-144.

Stewart, K. J., and Gosain, S. 2006. "The Impact of Ideology on Effectiveness in Open Source Software Development Teams," *MIS Quarterly* (30:2), pp. 291-314.

Steyaert, C. 2007. "'Entrepreneuring' as a Conceptual Attractor? A Review of Process Theories in 20 Years of Entrepreneurship Studies," *Entrepreneurship & Regional Development: An International Journal* (19:6), pp. 453-477.

Suchman, L., Blomberg, J., Orr, J., and Trigg, R. 1999. "Reconstructing Technologies as Social Practice," *American Behavioral Scientist* (43:3), pp. 392-408.

Ulhøi, J. P. 2004. "Open Source Development: A Hybrid in Innovation and Management Theory," *Management Decision* (42:9), pp. 1095-1114.

van der Burg, S., and van Gorp, A. 2005. "Understanding Moral Responsibility in the Design of Trailers," *Science and Engineering Ethics* (11:2), pp. 235-256.

Vaughn-Nichols, S. J. 2009. "Linux Runs the Stock Exchange," *PCWorld Business Center*, September 6 (http://www.pcworld.com/businesscenter/article/171523/linux_runs_the_stock_exchange.html).

von Hippel, E. 2001. "Innovation by User Communities: Learning from Open-Source Software," *Sloan Management Review* (42:4), pp. 82-86.

von Hippel, E., and von Krogh, G. 2003. "Open Source Software and the 'Private–Collective' Innovation Model: Issues for Organization Science," *Organization Science* (14:2), pp. 209-223.

von Krogh, G., Spaeth, S., and Lakhani, K. 2003. "Community, Joining, and Specialization in Open Source Software Innovation: A Case Study," *Research Policy* (32:7), pp. 1217-1241.

von Krogh, G., and von Hippel, E. 2006. "The Promise of Research on Open Source Software," *Management Science* (52:7), pp. 975-983.

Wasko, M. M., and Faraj, S. 2005. "Why Should I Share? Examining Social Capital and Knowledge Contribution in Electronic Networks of Practice," *MIS Quarterly* (29:1), pp. 35-37.

Weber, K., and Dacin, M. T. 2011. "The Cultural Construction of Organizational Life: Introduction to the Special Issue," *Organization Science* (22:2), pp. 287-298.

West, J., and O'Mahony, S. C. 2005. "Contrasting Community Building in Sponsored and Community Founded Open Source Projects," in *Proceedings of the 38th Annual Hawaii International Conference on System Sciences*, Los Alamitos, CA: IEEE Computer Society Press.

Whittington, R. 2006. "Completing the Practice Turn in Strategy Research," *Studies* (27:5), pp. 613-634.

Wiertz, C., and de Ruyter, K. 2007. "Beyond the Call of Duty: Why Customers Contribute to Firm-Hosted Commercial Online Communities," *Organization Studies* (28:3), pp. 347-376.

Wu, C., Gerlach, J. H., and Young, C. E. 2007. "An Empirical Analysis of Open Source Software Developers' Motivations and Continuance Intentions," *Information & Management* (44:3), pp. 253-262.

Xu, B., Jones, D. R., and Shao, B. 2009. "Volunteers' Involvement in Online Community Based Software Development," *Information & Management* (46:3), pp. 151-158.

Ye, Y., and Kishida, K. 2003. "Toward an Understanding of the Motivation of Open Source Software Developers," in *Proceedings of 2003 International Conference on Software Engineering*, Portland, OR, May 3-10, pp. 419-429.

Yu, J., Jiang, Z., and Chan, H. C. 2007. "Knowledge Contribution in Problem Solving Virtual Communities: The Mediating Role of Individual Motivations," in *Proceedings of the 2007 ACM SIGMIS CPR Conference on Computer Personnel Doctoral Consortium and Research Conference*, St. Louis, MO, April 19-21, pp. 144-152.

Zeitlyn, D. 2003. "Gift Economies in the Development of Open Source Software: Anthropological Reflections," *Research Policy* (32:7), pp. 1287-1291.

## About the Authors

**Georg von Krogh** (Ph.D., Norwegian University of Technology and Natural Science) is a professor at ETH Zurich where he holds the Chair of Strategic Management and Innovation. Georg specializes in competitive strategy, technological innovation, and knowledge management. His work has been published in leading journals including *Management Science*, *Organization Science*, *Research Policy*, *Strategic Management Journal*, and *Harvard Business Review*. He is a senior editor for *Organization Studies*, and an editorial board member of a number of journals.

**Stefan Haefliger** (Dr.oec., University of St.Gallen) works as a researcher and lecturer at the Department of Management, Technology, and Economics at ETH Zurich. His research and teaching focuses on cocreation strategies as well as knowledge reuse,

creation, and design in innovation processes. Stefan serves as an associate editor for *Long Range Planning* and his research has appeared in *Management Science, Research Policy*, and *Journal of Strategic Information Systems*.

**Sebastian Spaeth** (Dr.oec., University of St.Gallen) works as a researcher and lecturer at the Department of Management, Technology, and Economics at ETH Zurich, Switzerland. He is interested in collaborative innovation models and open business models. Sebastian has published articles in *Management Science*, *Research Policy*, *R&D Management*, and *Journal of Strategic Information Systems*.

**Martin W. Wallin** (Ph.D., Chalmers University of Technology, Sweden) works as a researcher and lecturer at the Department of Management, Technology, and Economics at ETH Zurich, Switzerland. His research focuses on the organizational and motivational implications of open innovation and his work has appeared in *Research Policy*, *Technovation*, *R&D Management*, and *Organizational Dynamics*.

MIS Quarterly

THEORY AND REVIEW

# CARROTS AND RAINBOWS: MOTIVATION AND SOCIAL PRACTICE IN OPEN SOURCE SOFTWARE DEVELOPMENT

**Georg von Krogh, Stefan Haefliger, Sebastian Spaeth, and Martin W. Wallin**
ETH Zurich, SWITZERLAND {gvkrogh@ethz.ch} {shaefliger@ethz.ch} {sspaeth@ethz.ch} {mwallin@ethz.ch}

# Appendix A

## Review Sample

| Article | Data/Method |
|---|---|
| Alexy and Leitner 2011 | Scenario experiment with 229 European computer science students testing the effect of monetary rewards on intrinsic motivation of OSS developers. |
| Baldwin and Clark 2006 | Models relationship between modular code bases and developers' incentives to join and remain involved in OSS development based on option value and game theory. |
| Benkler 2002 | Theory paper trying to explain advantages of commons-based peer-production. |
| Berquist and Ljungberg 2001 | Virtual ethnography using archival data from news groups and discussion lists trying to explain the power of gifts in OSS development. |
| Bitzer et al. 2007 | Formal modeling aiming to explain the importance of intrinsic motivation in OSS development. |
| David and Shapiro 2008 | Cluster analysis of data from David et al. (2003) to create motivation profiles. |
| David et al. 2003 | Global online survey of 1,588 self-selected OSS developers. |
| Fershtman and Gandal 2007 | Investigates the relationship between output and license restrictiveness using a sample of the 71 most active projects hosted on SourceForge. |
| Ghosh 2005 | Survey with 2,700 respondents investigating e.g., demographics, motivations, and contributions. |
| Hars and Ou 2002 | Online survey of 81 OSS developers aiming to explain participation. |
| Haruvy et al. 2003 | Formal modeling aiming to explain how non-pecuniary benefits impact on OSS contribution. |
| Hemetsberger 2004 | Content analysis of online survey responses of 1,139 OSS developers and users aiming to explain contribution. |
| Hertel et al. 2003 | Online survey of 141 OSS developers of the Linux kernel. |
| Ke and Zhang 2008 | Online survey of 204 OSS participants aiming to explain effort intensity. |
| Lakhani and von Hippel 2003 | Online survey of 336 contributors to the Apache web server software. |
| Lakhani and Wolf 2005 | Online survey of 684 SourceForge developers. |
| Lattemann and Stieglitz 2005 | Reviews literature with the aim to identify factors that sustain motivation over the life cycle of an open source project. |
| Lee and Cole 2003 | Uses archival data analyses, online research publications, and observations of how the Linux technology has evolved to create a community-based model of knowledge creation. |

| Article | Data/Method |
|---|---|
| Lerner and Tirole 2002 | Uses four cases and descriptive statistics to highlight the extent to which economics can explain OSS participation ("career concerns"). |
| Luthiger and Jungwirth 2007 | Two surveys of 1,330 open source developers and 114 closed source developers to explore motivation from a "flow theory" perspective. |
| Markus 2007 | Review and synthesis of OS governance literature. |
| Okoli and Oh 2007 | Examines the impact of network closure and structural holes on social capital by using a sample of 465 Wikipedia participants. |
| O'Mahony and Ferraro 2007 | Inductive ethnography exploring the evolution of governance structure and a logit regression model testing factors that increase the likelihood of becoming a leader. |
| Oreg and Nov 2008 | Survey of 185 SourceForge users and 115 Wikipedia participants investigating individual motivation. |
| Osterloh and Rota 2007 | Conceptual article exploring OSS characteristics that enable low-cost contributions and lower barriers to entry, and provide intrinsic motivation. |
| Riehle 2007 | Examines firms' and employees' motivations to contribute to OSS development. |
| Roberts et al. 2006 | Develops a model of motivation to participate in OSS development and tests it using data from an email survey of 288 Apache developers. |
| Rullani 2007 | Uses SourceForge project data to model how various variables explain "contribution." |
| Schofield and Cooper 2006 | Survey of 145 members of Linux user groups. |
| Shah 2006 | Inductively develops a framework of changing motivation over time depending on governance structure using data from 88 qualitative interviews and archival data. |
| Spaeth et al. 2008 | Case study examining private benefits of contributing to a public good. |
| Stewart et al. 2006 | Investigates the effect of license restrictiveness and organizational sponsorship on developer activity by examining 138 OSS projects on www.freshmeat.net. |
| Stewart and Gosain 2006 | Assesses a PLS model on effectiveness in OSS development based on OSS beliefs, values, and norms using survey data from 67 project administrators and SourceForge project data. |
| von Hippel and von Krogh 2003 | Conceptual article proposing the private-collective innovation model. |
| von Krogh et al. 2003 | Multi-source, grounded approach examining [un]successful joining and specialization process into www.freenetproject.org. |
| Wu et al. 2007 | Develops a structural equation model to test the effect of motivations on satisfaction and intention for continued participation using a SourceForge web survey with 148 responses (12% response rate). |
| Xu et al. 2009 | Develops a structural equation model to test the effect of motivation and community factors on voluntary involvement in OSS projects using a SourceForge web survey with 172 responses (17% response rate). |
| Ye and Kishida 2003 | Investigates the importance of learning for motivation and provides descriptive statistics of the GIMP project's mailing list and code contribution behavior. |
| Yu et al. 2007 | Propose eight motivations that drive volunteering individuals' motivation by creating hypotheses from a model. |
| Zeitlyn 2003 | Conceptual article proposing the concept of "kinship amity" in understanding OSS development. |

# Appendix B

## Intrinsic Motivation

| | |
|---|---|
| **Ideology** | Ideology has been quoted as a major reason for starting the GNU project, one of the earliest coherent attempts to write software under an explicitly open license (Stallman 1999). The extent to which contributors adhere to ideology is usually captured by items such as "software should be free for all," "free to modify and redistribute," or that "open source code should replace proprietary software."<br><br>*Key empirical findings:*<br>• Support for ideological motives found in developer surveys (David and Shapiro 2008; David et al. 2003; Ghosh 2005; Lakhani and Wolf 2005).<br>• Weak support by Hemetsberger (2004).<br>• Hertel et al. (2003) found a positive, significant relationship between social and political motives, and accepted source code patches and lines of code contributions.<br>• Stewart and Gosain (2006) found that open source developers' adherence to the community ideology (defined as "open source" norms, values, and beliefs) impacts on team effectiveness.<br>• Yu et al. (2007) identified "moral obligation" and "advancement of virtual community motive" as individual motivations in a literature review. |
| **Altruism** | Altruism is the selfless concern for the welfare of others. A typical altruistic act consists of three characteristics: "a) it is an end in itself; it is not directed at gain, b) is emitted voluntarily, and c) does good" (Heider 1958 in Krebs 1970, p. 259). Due to the self-containment of an altruistic act, it fits well with the category of intrinsic motivation, and several authors have used the concept of altruism to explain code contribution of OSS developers.<br><br>*Key empirical findings:*<br>• Osterloh and Rota (2007) suggested that altruistic behavior caused by "pro-social motives" influences developers to contribute to OSS development. The "pro-social motive" is a type of intrinsic motivation (Lindenberg 2001, quoted in Osterloh and Rota 2007), which the authors link to open source contributions.<br>• Haruvy et al. (2003) point out that companies need to manage contributors' motivations so as not to crowd out their altruistic motives.<br>• Wu et al. (2007) investigated the intention of OSS developers to continue their involvement in future projects. Their structural equation model shows that altruism in the form of helping behavior influences developers' continuance only if mediated by their satisfaction.<br>• Hemetsberger (2004) reported that 22% of developers ranked altruism as a motivational factor to contribute. Hemetsberger also attempted to differentiate between types of developer and found that the importance of altruism in explaining contributions is stronger for people who contribute a lot (30.7%), compared to medium (23.9%) and low contributors (6.5%).<br>• Hars and Ou (2002) reported that altruism motivated developers to contribute: 16.5% of the survey participants rated high on altruism. Student and hobby programmers rated altruism the highest at 24.2%, followed by salaried and contract programmers at 11.1%, whereas only 7.7% of the programmers paid for open source development were driven by altruistic motivations.<br>• Ghosh's (2005) survey finds selfish behavior to such a degree as to rule out altruistic behavior as an important characteristic of OSS development, although altruism is a driver for some individuals.<br>• Bitzer et al. (2007) identify the "desire to give a present to the programmer community" as a crucial pattern in OSS literature and include it in a model. |

| | |
|---|---|
| **Kinship amity** | The concept of kinship amity (Fortes 1969) has been related to the concept of the gift economy (Zeitlyn 2003). However, kinship amity differs from the gift economy, since the former does not assume reciprocity in social relations. For example, there is no calculated economic relationship in families (kin). Kinship amity thus also differs from altruism as a motive to contribute, because it is restricted to the group to which one belongs, such as the OSS community. In our review, some equivalent constructs are subsumed under kinship amity, for example, the frequently cited motivation for OSS contributors, community identification (e.g., Hars and Ou 2002). Community identification instills a feeling of belonging to a certain group, and urges people to help others in that group.<br><br>*Key empirical findings:*<br>• Zeitlyn (2003) first suggested kinship amity as a motive in OSS and as an explanation for why people contribute to OSS.<br>• Hemetsberger (2004), using concepts such as "group boundaries" and "group bonds," found a weak relationship between kinship amity and developers' level of contribution.<br>• Lakhani and Wolf's (2005) survey identified kinship amity as an important motive for contributing, and showed it is an important determinant of the effort invested (hours per week). Other surveys (David and Shapiro 2008; Hars and Ou 2002) find similar results. Hars and Ou (2002) studied "community identification," finding a correlation between kinship amity and the number of hours per week spent on OSS contribution.<br>• Hertel et al. (2003) tested the relationship between kinship amity and the number of accepted patches and lines of code, and found it to be positive and significant. |
| **Enjoyment and fun** | Enjoyment and fun have been said to motivate contributors to open source projects. One of the main drivers of the so called "hacker culture" emerging during the 1980s was for developers to enjoy the playfulness and experimentation with hardware and software (Levy 1984; Torvalds and Diamond 2002).<br><br>*Key empirical findings:*<br>• Benkler (2002) and Osterloh and Rota (2007) suggested that enjoyment plays an important role in OSS.<br>• Lakhani and von Hippel (2003) showed in their survey that developers considered enjoyment and fun important when conducting technically challenging tasks, whereas mundane tasks, such as helping users to install software, required different motives.<br>• Luthiger and Jungwirth (2007) conducted the most comprehensive study focusing exclusively on fun and enjoyment motivations. Their survey of 1,330 open source developers revealed that the fun factor had a significantly positive effect on both the number of hours spent on a project as well as on developers' intention to participate in the future. Fun accounted for 28% of the effort in terms of number of hours dedicated to projects.<br>• Lakhani and Wolf (2005) found that developers deemed enjoyment-based motivation an important source of motivation. In their research, high levels of enjoyment also increased the hours per week that developers spent on a project.<br>• Hertel et al. (2003), measuring the number of accepted patches and lines of code in the Linux project, also found a significant positive impact of fun and enjoyment.<br>• Hemetsberger's (2004) survey identified modest positive impact of enjoyment on contributions to projects.<br>• Shah (2006) showed that own-use value often formed the initial reason to join development, but over time fun and enjoyment increasingly form the sustaining motivation to long-term contribution.<br>• Roberts et al. (2006) could not identify a significant link between intrinsic enjoyment and the number of accepted patches and lines of code. |

# Appendix C

## Internalized Extrinsic Motivation

| | |
|---|---|
| **Reputation** | Reputation can be classified as "peer reputation" and "outside reputation." Peer reputation is usually targeted at community insiders (peers, or kin) and potential employers who perceive peer reputation to signal talent. Very few studies have considered reputation directed outside the community and not targeted toward potential future employers. Outside reputation is concerned with anticipated reactions to the contributors by significant others, such as friends and relatives, and prestige awarded.<br><br>*Key empirical findings:*<br>• Raymond's (1998) essay "Homesteading the Noosphere" linked reputation to reciprocity in the gift economy and described it as the "major motivation" for developers.<br>• Lerner and Tirole (2002) proposed peer reputation as a fundamental motivation.<br>• Osterloh and Rota (2007) termed this motivation "ego gratification," which could easily be confused with intrinsically motivated self-determination, the happiness of having/being able to achieve something (see Deci and Ryan 1987). However, the authors classify it as an extrinsic signaling incentive, aimed at increasing one's own labor market value.<br>• Lakhani and von Hippel (2003) differentiated peer reputation further. They proposed that peer reputation motivates "gratifying" technical tasks, while it fails to motivate the "necessary but mundane tasks" that are an inherent part of each software project.<br>• Lattemann and Stieglitz (2005) proposed that contributors' roles are related to motivations. In their view, programmers (rather than bug fixers, or managers) were motivated through peer reputation.<br>• Spaeth et al. (2008) argued that some motives are formed as by-products of contributions. In their empirical study of the Freenet project, the authors found that higher levels of contributions provided more peer reputation, such as positive mentioning in e-mail lists.<br>• The surveys by Ghosh (2005), Hars and Ou (2002), Hemetsberger (2004), and Lakhani and Wolf (2005) reported peer reputation as a driver for participation.<br>• Hars and Ou (2002) identified a weak but existing relationship between reputation and the number of hours invested.<br>• Lakhani and Wolf (2005) found peer reputation to be the fourth biggest determinant of invested effort.<br>• Roberts et al. (2006) measured the accepted lines of code. They identified a significant positive relationship between peer reputation motives and accepted code in the Apache project.<br>• Hemetsberger (2004) found a weak relationship between outside reputation and participation.<br>• Hertel et al. (2003) tested the impact of outside reputation on the number of accepted patches and lines of code. *Ceteris paribus* they found a significant positive impact of outside motivation on accepted code. |
| **Gift economy/ Reciprocity** | Originally a concept from anthropology (Mauss 1959), several authors discussed the logic of gift-giving in the context of OSS development (Bergquist and Ljungberg 2001; Raymond 1999; Zeitlyn 2003). Viewing OSS development as a gift economy asserts that developers give code to others expecting gifts in return. The corresponding internalized, extrinsic motivation can be termed "reciprocity."<br><br>*Key empirical findings:*<br>• Bergquist and Ljungberg (2001) suggested reciprocity as a motivation for contributions to OSS.<br>• Hemetsberger (2004) and Lakhani and Wolf (2005) confirmed reciprocity in empirical studies that found moderate support, while David et al. (2003) found strong support.<br>• In their survey, Lakhani and von Hippel (2003) found that reciprocity motivated developers to perform mundane tasks. It seems that people who have been helped by other contributors in the past are more inclined to reciprocate as they gain experience and knowledge. |

| | |
|---|---|
| **Learning** | The motive to acquire new skills or to learn through OSS development appears in almost every contribution to the review sample. However, the definition of learning was often vague and referred to survey items such as "improve programming skills" (the opportunity to learn from the experience of writing software and the feedback provided by the peers who tested, integrated, and commented upon the software published).<br><br>*Key empirical findings:*<br>• von Hippel and von Krogh (2003) proposed that learning poses a private benefit derived from the contributions to OSS development (also proposed by Yu et al. 2007). Spaeth et al. (2008) confirmed in an empirical study that learning through feedback is a driver for participation.<br>• Survey studies confirmed that "learning" motivated individuals to participate (Ghosh 2005; Hemetsberger 2004; Lakhani and Wolf 2005; Oreg and Nov 2008; Roberts et al. 2006), particularly in the surveys by David et al. (2003) and Hars and Ou (2002).<br>• Roberts et al. (2006) additionally show that accepted patches and lines of code written were positively impacted by learning.<br>• Wu et al. (2007) found that learning motives led to a higher intention to participate.<br>• Ye and Kishida (2003) suggested the consideration of legitimate peripheral learning based on the work by Lave and Wenger (1991) to explain increasing levels of participation over time (see also Rullani 2007).<br>• Xu et al. (2009) also find "skill development" to be a driver, although they refer to future work opportunities, rather than learning as a goal in itself.<br>• Stewart and Gosain (2006) see "learning as a value in itself" as a dimension of the "OSS values" construct that impacts effort positively via "affective trust." |
| **Own-use value** | Own-use value refers to internalized extrinsic motives to create OSS for contributors' personal use.<br><br>*Key empirical findings:*<br>• Lakhani and von Hippel (2003), Osterloh and Rota (2007), and Raymond (1999) suggested developers of OSS "scratch their itch" by developing software they find useful, by fixing bugs, and by adding features they need.<br>• Surveys by David et al. (2003), Ghosh (2005), Hars and Ou (2002), Lakhani and Wolf (2005), as well as Hemetsberger (2004) identified own-use value as a motive for participating in the development of OSS.<br>• Wu et al. (2007) found own-use value was connected to the intention to participate.<br>• Lakhani and von Hippel (2003) identified own-use value as a motive for taking on mundane tasks.<br><br>• Hars and Ou (2002) reported that developers attributed a high score to own-use value as their motive regarding actual effort measured in hours spent per week.<br>• Hertel et al. (2003) reported that own-use value had a significant effect on accepted patches and lines of code contributed.<br>• Roberts et al. (2006) reported that own-use value exerted a significant negative impact on the level of participation in the Apache project, also measured in accepted patches and lines of code. One explanation offered is that developers motivated by own-use value worked "eclectically": they would fix bugs that annoyed them and then leave the development again, rather than remaining as long-term developers. This behavior would result in relatively low total contributions to one project.<br>• Lattemann and Stieglitz (2005) proposed that own-use value might impact on OSS development via the roles individuals assume in communities. Contributors who mainly fix bugs may be particularly motivated by own-use value, whereas others such as managers (maintainers), might be more motivated by pay. |

# Appendix D

## Extrinsic Motivation

| | |
|---|---|
| **Career** | Lerner and Tirole (2002) first suggested studying the signaling behavior of OSS developers. Their proposition, derived from economic literature, stated that individual developers would be motivated by career concerns when developing OSS. By publishing software that was free for all to inspect, they could signal their talent to potential employers and thus increase their value in the labor market.<br><br>*Key empirical findings:*<br>• Has been proposed by Lerner and Tirole (2002), Riehle (2007), Yu et al. (2007) as motivation.<br>• Some surveys including Lakhani and Wolf (2005), Hemetsberger (2004) found weak support for career concerns as motivation.<br>• Hars and Ou (2002) and Ghosh (2005) found more substantial support for career concerns as motivation.<br>• Wu et al. (2007) found career concerns related to intended participation.<br>• Hars and Ou (2002) found strong support of the motivation for efforts measured in hours per week spent on OSS development. They also report that career concerns played a more important role for paid participation in OSS development than for unpaid participation.<br>• Roberts et al. (2006) and Hertel et al. (2003) documented a positive and significant relationship between career concerns, accepted code patches and lines of code.<br>• Xu et al. (2009) use a single construct—"Reputation and Skill Gaining"—which they state "may help the developer's future work opportunities" (2009, p. 153), driving involvement. |
| **Pay** | A significant minority (approximately 40%) of contributors is paid to participate in OSS projects (Lakhani and Wolf 2005). An examination of contributions to the Linux kernel found only 9% of the developers involved worked in their own time (Kroah-Hartman et al. 2009).<br><br>*Key empirical findings:*<br>• Lakhani and Wolf (2005) examined the degree of participation and its link to financial motives. They concluded that the financial subsidy of these projects was substantial. For example, paid contributors dedicated 17.7 hours per week on all FOSS projects they participated in, while volunteers contributed 11.7 hours per week. As programmers often participate in several projects simultaneously, Lakhani and Wolf reported the results for the focal project of the programmers as well. These results showed a similar pattern: 10.3 hour per week for the paid contributor and 5.7 hours per week for the volunteer. The differences between the groups were found to be significant.<br><br>• Surveys by Hars and Ou (2002), Hertel et al. (2003), and Luthiger and Jungwirth (2007) reported findings consistent with Lakhani and Wolf (2005). |

## References


Alexy, O., and Leitner, M. 2011. "A Fistful of Dollars: Are Financial Rewards a Suitable Management Practice for Distributed Models of Innovation?," *European Management Review* (8:3), pp. 165-185.

Baldwin, C. Y., and Clark, K. B. 2006. "The Architecture of Participation: Does Code Architecture Mitigate Free Riding in the Open Source Development Model?," *Management Science* (52:7), pp. 1116-1127.

Benkler, Y. 2002. "Coase's Penguin, or Linux and the Nature of the Firm," *Yale Law Journal* (112:3), pp. 369-447.

Bergquist, M., and Ljungberg, J. 2001. "The Power of Gifts: Organizing Social Relationships in Open Source Communities," *Information Systems Journal* (11:4), pp. 305-320.

Bitzer, J., Schrettl, W., and Schröder, P. J. 2007. "Intrinsic Motivation in Open Source Software Development," *Journal of Comparative Economics* (35:1), pp. 160-169.

David, P. A., and Shapiro, J. S. 2008. "Community-Based Production of Open-Source Ssoftware: What Do We Know About the Developers Who Participate?," *Information Economics and Policy* (20:4), pp. 364-398.

David, P. A., Waterman, A., and Arora, S. 2003. *FLOSS-US: The Free/Libre/Open Source Software Survey for 2003*, Stanford Institute for Economic Policy Research, Stanford University, Stanford, CA (http://www.stanford.edu/group/floss-us/report/ FLOSS-US-Report.pdf).

Deci, E. L., and Ryan, R. M. 1987. "The Support of Autonomy and the Control of Behavior," *Journal of Personality and Social Psychology* (53:6), pp. 1024-1037.

Fershtman, C., and Gandal, N. 2007. "Open Source Software: Motivation and Restrictive Licensing," *International Economics and Economic Policy* (4:2), pp. 209-225.

Fortes, M. 1969. *Kinship and the Social Order: The Legacy of Lewis Henry Morgan*, Chicago: Aldine.

Ghosh, R. A. 2005. "Understanding Free Software Developers: Findings from the FLOSS Study," in *Perspectives on Free and Open Source Software*, J. Feller, B. Fitzgerald, S. A. Hissam, and K. R. Lakhani (eds.), Cambridge, MA: MIT Press, pp. 23-46.

Hars, A., and Ou, S. 2002. "Working for Free? Motivations for Participating in Open-Source Projects," *International Journal of Electronic Commerce* (6), pp. 25-39.

Haruvy, E., Prasad, A., and Sethi, S. 2003. "Harvesting Altruism in Open-Source Software Development," *Journal of Optimization Theory and Applications* (118:2), pp. 381-416.

Heider, F. 1958. *The Psychology of Interpersonal Relations*, New York: Wiley.

Hemetsberger, A. 2004. "Fostering Cooperation on the Internet: Social Exchange Processes in Innovative Virtual Consumer Communities," extended abstract published in *Advances in Consumer Research* (Vol. 29), S. M. Broniarczyk and K. Nakamoto (eds.), Valdosta, GA: Association for Consumer Research, pp. 354-356.

Hertel, G., Niedner, S., and Herrmann, S. 2003. "Motivation of Software Developers in Open Source Projects: An Internet-Based Survey of Contributors to the Linux Kernel," *Research Policy* (32:7), pp. 1159-1177.

Ke, W., and Zhang, P. 2008. "Motivations for Participating in Open Source Software Communities: Roles of Psychological Needs and Altruism," in *Proceedings of the 12th Pacific Asia Conference on Information Systems*, Suzhou, China, pp. 136-148.

Krebs, D. L. 1970. "Altruism - Examination of Concept and a Review of Literature," *Psychological Bulletin* (73:4), pp. 258-302.

Kroah-Hartman, G., Corbet, J., and McPherson, A. 2009. *Linux Kernel Development: How Fast It Is Going, Who Is Doing It, What They Are Doing, and Who Is Sponsoring It: An August 2009 Update*, The Linux Foundation (http://www.linuxfoundation.org/).

Lakhani, K. R., and von Hippel, E. 2003. "How Open Source Software Works: 'Free' User-to-User Assistance," *Research Policy* (32:6), pp. 923-943.

Lakhani, K. R., and Wolf, R. G. 2005. "Why Hackers Do What They Do: Understanding Motivation and Effort in Free/Open Source Software Projects," in *Perspectives on Free and Open Source Software*, J. Feller, B. Fitzgerald, S. Hissam, and K. R. Lakhani (eds.), Cambridge, MA: MIT Press, pp. 3-22.

Lattemann, C., and Stieglitz, S. 2005. "Framework for Governance in Open Source Communities," in *Proceedings of the 38th Annual Hawaii International Conference on System Sciences*, Los Alamitos, CA: IEEE Computer Society Press, pp. 192b.

Lave, J., and Wenger, E. 1991. *Situated Learning: Legitimate Peripheral Participation*, Cambridge, UK: Cambridge University Press.

Lee, G. K., and Cole, R. E. 2003. "From a Firm-Based to a Community-Based Model of Knowledge Creation: The Case of the Linux Kernel Development," *Organization Science* (14:6), pp. 633-649.

Lerner, J., and Tirole, J. 2002. "Some Simple Economics of Open Source," *Journal of Industrial Economics* (50:2), pp. 197-234.

Levy, S. 1984. *Hackers: Heroes of the Computer Revolution*, New York: Anchor Press/Doubleday.

Lindenberg, S. M. 2001. "Intrinsic Motivation in a New Light," *Kyklos* (54:2/3), pp. 317-342.

Luthiger, B., and Jungwirth, C. 2007. "Pervasive Fun," *First Monday* (12:1), January (available at http://firstmonday.org/htbin/cgiwrap/bin/ojs/index.php/fm/article/view/1422).

Markus, M. L. 2007. "The Governance of Free/Open Source Software Projects: Monolithic, Multidimensional, or Configurational?," *Journal of Management and Governance* (11:2), pp. 151-163.

Mauss, M. 1959. *The Gift: The Form and the Reason for Exchange in Archaic Societies*, London: Routledge.

Okoli, C., and Oh, W. 2007. "Investigating Recognition-Based Performance in an Open Content Community: A Social Capital Perspective," *Information & Management* (44:3), pp. 240-252.

O'Mahony, S. C., and Ferraro, F. 2007. "The Emergence of Governance in an Open Source Community," *Academy of Management Journal* (50:5), pp. 1079-1106.

Oreg, S., and Nov, O. 2008. "Exploring Motivations for Contributing to Open Source Initiatives: The Roles of Contribution Context and Personal Values," *Computers in Human Behavior* (24:5), pp. 2055-2073.

Osterloh, M., and Rota, S. G. 2007. "Open Source Software Development—Just Another Case of Collective Invention?," *Research Policy* (36:2), pp. 157-171.

Raymond, E. S. 1998. "Homesteading the Noosphere," *FirstMonday* (3:10), November 10 http://firstmonday.org/htbin/cgiwrap/bin/ojs/index.php/fm/article/view/1474/1389).

Raymond, E. S. 1999. "The Cathedral & the Bazaar," in *The Cathedral & the Bazaar,* Sebastapol, CA: O'Reilly & Associates, Inc., pp. 19-64.

Riehle, D. 2007. "The Economic Motivation of Open Source Software: Stakeholder Perspectives," *Computer* (40:4), pp. 25-32.

Roberts, J. A., Hann, I., and Slaughter, S. A. 2006. "Understanding the Motivations, Participation, and Performance of Open Source Software Developers: A Longitudinal Study of the Apache Projects," *Management Science* (52:7), pp. 984-999.

Rullani, F. 2007. "Dragging Developers Towards the Core: How the Free/Libre/Open Source Software Community Enhances Developers' Contribution," IFIP Working Group 2.13 (http://ifipwg213.org/system/files/rullani_socenv.pdf).

Schofield, A. and Cooper, G. S. 2006. "Participation in Free and Open Source Communities: An Empirical Study of Community Members' Perceptions." in *Open Source Systems* (IFIP Working Group 2.13), Berlin: Springer, pp. 221-231.

Shah, S. K. 2006. "Motivation, Governance, and the Viability of Hybrid Forms in Open Source Software Development," *Management Science* (52:7), pp. 1000-1014.

Spaeth, S., Haefliger, S., von Krogh, G., and Renzl, B. 2008. "Communal Resources in Open Source Software Development," *Information Research* (13:1) (http://informationr.net/ir/13-1/ paper332.html).

Stallman, R. 1999. "The GNU Operating System and the Free Software Movement," in *Open Sources: Voices from the Open Source Revolution*, C. DiBona, S. Ockman, and M. Stone (eds.), Sebastopol, CA: O'Reilly & Associates, Inc., pp. 53-70.

Stewart, K. J., Ammeter, A. P., and Maruping, L. M. 2006. "Impacts of License Choice and Organizational Sponsorship on User Interest and Development Activity in Open Source Software Projects," *Information Systems Research* (17:2), pp. 126-144.

Stewart, K. J., and Gosain, S. 2006. "The Impact of Ideology on Effectiveness in Open Source Software Development Teams," *MIS Quarterly* (30:2), pp. 291-314.

Torvalds, L., and Diamond, D. 2002. *Just for Fun: The Story of an Accidental Revolutionary*, New York: HarperCollins

von Hippel, E., and von Krogh, G. 2003. "Open Source Software and the 'Private–Collective' Innovation Model: Issues for Organization Science," *Organization Science* (14:2), pp. 209-223.

von Krogh, G., Spaeth, S., and Lakhani, K. 2003. "Community, Joining, and Specialization in Open Source Software Innovation: A Case Study," *Research Policy* (32:7), pp. 1217-1241.

Wu, C., Gerlach, J. H., and Young, C. E. 2007. "An Empirical Analysis of Open Source Software Developers' Motivations and Continuance Intentions," *Information & Management* (44:3), pp. 253-262.

Xu, B., Jones, D. R., and Shao, B. 2009. "Volunteers' Involvement in Online Community Bbased Software Development," *Information & Management* (46:3), pp. 151-158.

Ye, Y., and Kishida, K. 2003. "Toward an Understanding of the Motivation of Open Source Software Developers," in *Proceedings of 2003 International Conference on Software Engineering*, Portland, OR, May 3-10, pp. 419-429.

Yu, J., Jiang, Z., and Chan, H. C. 2007. "Knowledge Contribution in Problem Solving Virtual Communities: The Mediating Role of Individual Motivations," in *Proceedings of the 2007 ACM SIGMIS CPR Conference on Computer Personnel Doctoral Consortium and Research Conference*, St. Louis, MO, April 19-21, pp. 144-152.

Zeitlyn, D. 2003. "Gift Economies in the Development of Open Source Software: Anthropological Reflections," *Research Policy* (32:7), pp. 1287-1291.

Copyright of MIS Quarterly is the property of MIS Quarterly & The Society for Information Management and its content may not be copied or emailed to multiple sites or posted to a listserv without the copyright holder's express written permission. However, users may print, download, or email articles for individual use.