DEVELOPER CENTRALITY AND THE IMPACT OF VALUE CONGRUENCE AND INCONGRUENCE ON COMMITMENT AND CODE CONTRIBUTION ACTIVITY IN OPEN SOURCE SOFTWARE COMMUNITIES

Sherae L. Daniel

University of Cincinnati

Likoebe M. Maruping

Georgia State University

Marcelo Cataldo

Dell EMC

Developer Centrality and the Impact of Value Congruence and Incongruence on Commitment and Code Contribution Activity in Open Source Software Communities

ABSTRACT

Open source software (OSS) communities enjoy success that comes from the contributions of developers, who in many cases, never meet face-to-face and collaborate primarily through technology-enabled means. With their fluid membership, such communities often rely on engaging the commitment of developers to their cause. The values associated with the open source movement represent one key mechanism that entices developers and research has found such values to affect their attitudes and behavior. Much of the prior research that examines the impact of OSS values has typically focused on either the extent to which the developer adheres to these values or the extent to which the community adheres to these values, neglecting their joint impact. Moreover, an implicit assumption in the prior work is that developers are all equally embedded within the community. In this research we challenge these assumptions by examining the role of congruence and incongruence in developer and OSS community values in affecting the commitment and associated code contribution activity of developers who have varying degrees of centrality within the community. Drawing on uncertainty reduction theory, person-environment (PE) fit, and social network theory, we argue that the extent to which congruence and incongruence in OSS values matter in shaping commitment differs for developers who are highly central within the community social network versus those who are less central. Using polynomial regression analysis, we analyze survey and archival data from 410 developers in an OSS community. Results suggest that developer centrality moderates the impact of congruence and incongruence in OSS values on commitment. Moreover, commitment fully mediates the impact of OSS value congruence and incongruence on developer contribution activity. We discuss the implications of our findings for research and practice.

INTRODUCTION

Open source software (OSS) development has grown in popularity from a fledgling movement to a highly attractive alternative to closed source development. The attractiveness of this model has only grown with the success of OSS projects such as Linux, Apache, Eclipse, mySQL, R, and Open Office (Dahlander and O'Mahony 2011, Kogut and Metiu 2001, Lee and Cole 2003, Mallapragada et al. 2012, Shah 2006). A major source of the attraction to the OSS approach to developing software is that it leverages the voluntary contributions of a multitude of developers who produce software that comes at least as robust and creative as closed source, if not more so (Kuan 2002, Paulson et al. 2004). In many ways, the OSS approach has revolutionized the way we manage innovation (Singh et al. 2011, Von Krogh et al. 2012). It is no wonder, then, that there has been a change in the nature of OSS communities due to the large influx of investment from a bevy of major software firms and venture capitalists seeking to support OSS development initiatives (Dwoskin 2016). For example, Oracle Corporation participates in more than 700 OSS community projects (Aksulu and Wade 2010).

As successful as the OSS approach has been, many OSS projects fail (Chengalur-Smith et al. 2010, English and Schweik 2007) because they cannot attract developer code contributions (Fang and Neufeld 2009, Robles and Gonzalez-Barahona 2006). Developer contributions to the code base mainly take the form of *commits*—implemented changes or additions to the software code (Singh, et al. 2011). This is the means by which OSS projects create new features and functionalities, fix bugs, and improve general robustness in a software application. Without these developer contributions, OSS projects remain stagnant and can become less attractive in the marketplace because they fail to keep pace with user demands and may become costly to maintain as the code decays (Chengalur-Smith, et al. 2010). At the same time the project will not attract developers looking to learn innovative software development techniques (Dahlander and O'Mahony 2011). A key challenge that OSS projects face is in attracting the commitment of developer contributions, particularly because (1) developers choose among myriad other projects to which to direct their attention and (2) they are under no contractual obligation to any particular project (Robles and Gonzalez-Barahona 2006, Seidel and Stewart 2011).

Developing software code is demanding on time and cognitive effort. As such it behooves OSS communities to make the collaboration process as frictionless as possible, lest developers take their time and talents elsewhere. Because of the distributed and digitally-mediated nature of OSS collaboration (Agerfalk and Fitzgerald 2008, Howison and Crowston 2014, Singh, et al. 2011, Stewart and Gosain 2006), developers rely on a common understanding of the practices for contributing to a project. In its early stages, the OSS movement enshrined these practices in OSS values that emphasized an approach to software development that was inimical to the closed source approach dominant at the time (Ljungberg 2000, Raymond 2001). OSS values created a level of certainty for developers about what would be expected of them when contributing to OSS projects (Stewart and Gosain 2006). The increased participation of profit-oriented firms and paid developers (Spaeth et al. 2015) with vastly different interests has created less clarity about how developers should engage in OSS projects. For example, when Sun Microsystems—which is heavily involved in the Netbeans.org community—changed the project's licensing scheme (that regulates the development process), it created uncertainty for developers and resulted in collaboration breakdowns regarding who would do what (Jensen and Scacchi 2005). Some developers in the community believed Sun did not follow the OSS values.

As the above example illustrates, developers care deeply about how OSS communities orchestrate contributions. Yet, researchers and practitioners understand little about the ramifications of inadequate attention to such matters. OSS community managers then must grapple to understand how to channel developer resources in a way that can make or break their projects. In particular, how can OSS community managers maintain developer commitment when there is uncertainty around how developers should make contributions?

To elaborate the theoretical mechanisms linking developer uncertainty to individual attitudes and behaviors, we leverage uncertainty reduction theory (Berger and Calabrese 1975, Hogg et al. 2010). Our core thesis is that as certainty around the process of contribution increases, developer commitment rises and, consequently, their contribution to the OSS community grows. Leveraging uncertainty reduction theory, we posit that this relationship occurs because of a person's natural aversion to the unpredictability of the behavior of those around them and the uncertainty that this creates about how they themselves should behave (Berger and Calabrese 1975, Hogg 2000). Given the collaborative nature of OSS communities, we identify two relational mechanisms that can minimize developer uncertainty. In particular, we integrate the person-environment (PE) fit perspective and social network theory as alternative relational mechanisms to minimize developer uncertainty by providing clarity about how to engage within the community. We draw upon these two theoretical perspectives because both are relational in nature and offer complementary perspectives in their treatment of individuals *within* their environment. The PE fit perspective explicitly recognizes an individual's attitudes and behavior as being shaped by characteristics of the individual relative to the environment in which they engage (Kristof-Brown et al. 2005, Kristof-Brown and Guay 2011). It represents a more passive mechanism of reducing uncertainty. However, the PE fit perspective says little about how embedded individuals are—by way of social connections—within their environments.

Social network theory enables us to further elaborate this influence by considering the individual's embeddedness within their environment via their social connectedness within the community (Pollack et al. 2012, Vardaman et al. 2015). A developer's position within the community's communication network can provide an alternative avenue for managing uncertainty by serving as a pipeline for the flow of, and access to, information about how to engage within the community (Podolny 2001). Compared to a PE fit perspective, an individual's pattern of accessing information is more interactive. As such, it enables us to conceptualize how different individuals within the same environment might react to similar person-environment relations—i.e., who is more or less likely to anchor their commitment to the community on congruence or incongruence in values. The potential impact of such social connectedness on developer reaction remains unclear. On the one hand a highly connected developer might experience visceral concerns about congruence and incongruence in OSS values, whereas those with limited connections might not care much given their relative isolation from community members. On the other hand, by virtue of the social connections to the community, a highly connected developer may have sufficient anchoring in the community to render congruence and incongruence less relevant to their experience of participation. More isolated developers might care more because congruence and incongruence affect their ability to make code contributions. Hence, we seek to understand whether a developer's connectedness within a community's communication network shapes their reaction to congruence and incongruence in OSS values. In so doing, this research addresses the following questions: *How do OSS value congruence and incongruence between developers and community affect developers' commitment and code contribution to the community? How salient is the influence of this congruence and incongruence among developers with different degrees of connectedness within the community?*

THEORETICAL BACKGROUND

Commitment in OSS Communities

With well over 100,000 OSS projects currently in existence and digital repositories (e.g., Github, Sourceforge) making it easy to find and access projects of interest, developers have a plethora of choices about where to devote their time and effort. With numerous choices and such ease of movement, developers often make one-off contributions to OSS communities (Pham et al. 2013, Pinto et al. 2016, Zhou and Mockus 2012). However, a significant number of developers—who are not project owners—make sustained contributions to a particular OSS community (Fang and Neufeld 2009, Qureshi and Fang 2010). The sustained contribution begs the question of why developers do this and what OSS communities might do to attract and retain them.

A significant amount of research has provided useful insights about why developers voluntarily contribute to OSS communities (e.g., Ke and Zhang 2010, Lakhani and Wolf 2005, Roberts et al. 2006). While this research helps us understand why developers would generally volunteer their time and effort to OSS communities, the literature remains less clear about why developers would devote themselves to contributing to a *particular* OSS community. To seek better understanding of this, we draw upon the concept of organizational commitment.

Organizational commitment represents a psychological bond that individuals form with an organization; that stabilizes their behaviors toward the organization (Mathieu and Zajac 1990, Meyer and Allen 1991). It has been used as a basis for understanding a variety of individual attitudes and behavior toward organizations including job performance (e.g., Meyer et al. 1989), satisfaction (e.g., Dishon-Berkovits and Koslowski 2002), and prosocial behavior (e.g., Meyer and Herscovitch 2001). We focus on the *affective* form of organizational commitment, which emphasizes the emotional attachment that individuals feel toward an organization (Meyer and Allen 1991). In a digital environment where developers have numerous OSS communities to which they can contribute, emotional attachment has the greatest potential to retain the interest and contribution of participants. Emotional attachment is particularly important in digital environments where participants generally do not interact with each other face-to-face (Seidel and Stewart 2011, Stewart and Gosain 2006). Additionally, affective commitment has been found to be the only form of commitment that prompts participants in online communities to post replies—a behavior that is more effortful than reading posts (Bateman et al. 2011). Commitment likely plays a similar role in influencing developers to make code commits—a particularly effortful form of contribution to OSS communities.

There are three primary reasons why organizational commitment is particularly well-suited for our research. First, although it has received a significant amount of empirical and theoretical attention within employment contexts, organizational commitment traces its roots to efforts to understand why volunteers in non-profits exhibit differences in their level of dedication (Becker 1960). This speaks directly to our phenomenon of interest with regard to developers volunteering their time and effort to contributing to an OSS community. Second, a focus on organizational commitment provides a locally situated explanation for individual behavior within a particular organizational context, by recognizing both the individual and the organization (Allen and Meyer 1997, Bateman, et al. 2011). This is relevant because it can provide insight into the code contribution behaviors (e.g., commits) of developers in OSS communities. Third, as Bateman et al. (2011) point out in the context of online communities, a focus on commitment helps managers explain why such individuals "engage in particular activities in *their* community," (2011, p. 842, emphasis in original) not in online communities in general. This is relevant given our theoretical interest in understanding why developers choose to commit to a particular OSS community.

A person does not develop emotional attachments at first sight—or at first encounter in the OSS case. Before they can form an emotional attachment to an OSS community, a developer must experience contributing to that community. In attempting to contribute code, developers can experience frustration and impatience with issues that make the process of contributing more difficult. Indeed, recent research suggests that a major reason that developers turn away from contributing to particular OSS communities is not because software coding is difficult, but because the *process* of working in that environment is cumbersome or unclear (Steinmacher et al. 2015). Next, we elaborate this barrier to contributing to OSS communities and explain why

OSS values and communication networks can serve as valuable mechanisms for reducing uncertainty around the process of working together.

Uncertainty about Participation Processes in OSS Communities

Steinmacher et al. (2015) identify several social barriers to participation in OSS communities. They include a lack of community responsiveness to developer communications and cultural differences. With regard to lack of community responsiveness, they found that many developers who turned away did so after not receiving answers to their posts to the community or after continuously receiving delayed responses that made it difficult to complete tasks in a timely manner. A lack of response can result in duplicated code and effort which can frustrate developers. Other studies also found that community responsiveness drives some developers continued participation (Zhang et al. 2013, Zhou and Mockus 2015). Some developers further reported receiving rude responses from the community as a major factor. In terms of cultural differences, some developers reported difficulty dealing with the communication styles of other community members. One Brazilian developer interpreted a German developer’s directness as being rude (Steinmacher, et al. 2015). Daniel et al. (2013) similarly found that cultural diversity in OSS communities decreased the amount of community engagement.

From a technical standpoint, the main barriers identified by Steinmacher et al. (2015) include trouble with orienting themselves within the community and inadequate documentation. Developers noted experiencing trouble orienting themselves and their role within the community in terms of knowing on which tasks to begin work, finding the appropriate artifacts to use in fixing an issue, knowing how to contribute, setting up the local environment, and understanding the contribution flow. In addition, inadequate documentation made it difficult for developers to understand the architecture of the software, how to setup the workspace, and how to write-up the necessary code documentation. Notably, one developer in their study lamented "*I think I will have to look at all the documentation. I have no idea of where the code that I need to change is*" (Steinmacher et al. 2016, p. 6). These types of barriers made it difficult for developers to engage in the process of contributing code (Steinmacher et al. 2016).

As this discussion suggests, developers face challenges in the process of contributing to OSS projects. These barriers create uncertainty for developers about how to use their time and skills to interact with community members and to make code contributions. Uncertainty can erode any attachment that a developer might otherwise feel toward an OSS community. Next, we briefly discuss uncertainty reduction theory as a lens for understanding why developers are less likely to commit to OSS communities when there are no mechanisms in place to provide guidance about how to contribute. We then discuss the importance of the values that underlie the OSS ideology and position in the communication network as just such mechanisms.

Uncertainty Reduction Theory

Uncertainty emerges when a person cannot predict the behaviors of others or the outcomes of his own actions (Berger and Calabrese 1975, Hogg et al. 2010, Hogg and Terry 2000). It also emerges when experiences do not match expectations and when relationships change (Kramer 1993). Uncertainty often leads to anxiety and stress (Hogg 2000, Miller and Monge 1985, Schweiger and Denisi 1991) and within organizational settings, uncertain situations—such as secrecy during mergers and job uncertainty—are associated with lower job satisfaction and lower commitment (Ashford et al. 1989, Bastien 1987, Bordia et al. 2004). Consequently, people hold a core motivation to reduce uncertainty (Hogg 2000, Miller and Monge 1985, Schweiger and Denisi 1991).

People generally employ three strategies to reduce uncertainty: passive, active, and interactive (Berger and Calabrese 1975, Ramirez et al. 2002). Passive approaches involve information acquisition about how others act and what they expect through inconspicuous means such as behavior observation (Ramirez et al. 2002). Behavior observation is one of the main approaches that people use to gather information in general (Berger and Calabrese 1975, Knobloch 2015). Further, passive observation, or lurking, is the primary form of information seeking on discussion lists (Nonnecke and Preece 2000) and online communities (Smith 1993). Digital traces of contribution activity can enable OSS developers to passively infer OSS community member values through observation. Active approaches involve searching for information about the target from third parties.

Finally, interactive strategies encompass information acquisition through direct interaction with the target. This can be accomplished through posing questions to the target. Indeed, Berger and Calabrese (1975) observed that people communicated as a way to reduce uncertainty about strangers and subsequent research found that communication patterns often mirror the degree of uncertainty in the task environment in technical work (Tushman 1979, Tushman 1979). OSS developers can also communicate as a way to reduce uncertainty. These mechanisms can reduce the friction involved in engaging with the OSS community, enabling them to focus on the more enjoyable work of writing code and solving technical problems (Weinberg 1998). Next, we highlight OSS values and communication networks as two mechanisms for managing OSS developer uncertainty.

OSS Values and Communication Networks as Relational Mechanisms

During the early stages of the OSS movement, the OSS ideology emerged as a way to socially codify an approach to developing software that departed from the predominant closed source approach (Ljungberg 2000, Raymond 2001, Stewart and Gosain 2006). The OSS ideology included a set of values. Broadly speaking, values represent a preference for a particular set of behaviors or outcomes (Trice and Beyer 1993). Values are a vital component of one's self-concept (Lydon 1996) and include a relatively enduring conviction that a specific mode of conduct or end state is preferable to its opposite (Chatman 1991, Rokeach 1973). For these reasons, values guide an individual's decisions and actions. *OSS values* place a significant emphasis on sharing information, helping each other in development efforts, and voluntarily cooperating with others as preferable modes of conduct and underscore the pursuit of the development of technical knowledge, continuously learning new skills, and development of reputation through participation as desirable end states (Raymond 2001, Stallman 1992, Stewart and Gosain 2006). As such, OSS values codify the expectations around how developers conduct themselves when contributing to an OSS community. In this way, OSS values can facilitate a passive mechanism through which developers can minimize uncertainty.

Although the focus has not been relational in nature, prior research has focused on the influence of OSS values. One body of work has focused exclusively on the influence of OSS values with regard to developers and their impact on attitudes and contributions (Benbya and Belbaly 2010, Henkel 2008, Xu et al. 2009. A different body of work focuses exclusively on the influence of OSS values with regard to the community and the impact on developers' attitudes and contributions (e.g., Chou and He 2011, Stewart and Gosain 2006). In one of the earliest studied to consider both developer and OSS communities in general, Ke and Zhang (2009) conceptualize ideological conviction as the extent to which a developer identifies with the OSS values. They argue that when developers feel that their values are aligned with those of the OSS movement, they are likely to be energized in their community contributions. Similarly, Ke and

Zhang (2010) conceptualize integrated motivation as a developer's conviction with the ideological values of the OSS movement. They argue that the greater the conviction developers feel the greater their task effort should be. Both of these studies primarily focus on the motivational role of OSS values. A summary of the research on OSS values and its impact on developers is included in Appendix A.

Research on a variety of phenomena has shown the theoretical insights that we can gain by isolating the effects of person characteristics (i.e. developer), environment characteristics (i.e. OSS community), the similarities between them, and the differences when one exceeds the other (Kristof-Brown et al. 2005). We follow this work and leverage the PE fit perspective because it provides an explicit treatment of individuals within their environments (Kristof-Brown, et al. 2005). PE fit represents the compatibility between a person and their environment on the basis of some characteristic (Kristof-Brown, et al. 2005, Kristof-Brown and Guay 2011). Consistent with this perspective, we focus on the extent to which a developer's OSS values match his perception of those of the rest of the OSS community. *Congruence* is the amount of similarity between the characteristics of an individual and the characteristics of the environment in which that individual operates (Hoffman and Woehr 2006). We also examine *incongruence* which is the extent to which a developer's OSS values are discrepant from the perception of those of the OSS community in which he is embedded.

As noted earlier, in some communities, the opportunity to leverage passive mechanismsmay not necessarily be obvious or present. Yet, in such conditions we still see developers willing to commit to contributing to the community. Communication networks offer an opportunity to enact interactive approaches for attaining understanding of how to engage within the community. By considering the communication network we address a limitation of the

PE fit perspective. While PE fit explicitly accounts for person-environment relations with regard to *characteristics*, it does not account for the individual's *connectedness* within that environment. As such, PE fit itself does not fully account for person-environment relations. This is theoretically significant because social connections represent a vital part of organizational and community life (Vardaman et al. 2015), particularly in digitally-enabled settings (Zhang et al. 2013). Reviews of PE fit have recognized this limitation and called for research to incorporate considerations of individuals' local context within their environment (Kristof-Brown et al. 2005, Kristof-Brown and Guay 2011). Consequently, we leverage social network theory to understand the local context of developers in relation to other community members by examining the degree to which a developer's centrality in the communication network shapes the effects of value congruence and incongruence on commitment.

When knowledge work is distributed, individuals engage primarily in discussion threads in order to share knowledge and learn from each other (Orlikowski and Yates 1994). In OSS communities, mailing lists are a leading communication venue for developers (Jensen et al. 2011, Kidane and Gloor 2007). Through mailing lists developers communicate about topics including design and implementation decisions, developer roles on the project, decision making processes and coordination (Jensen, et al. 2011, Kidane and Gloor 2007). The relative informality of OSS community organization coupled with fluid participation creates coordination needs that are typically handled through mailing lists (Dahlander and O'Mahony 2011). Coordination in OSS projects includes prioritizing the timing of code contributions and managing their sequencing.

Through such communication networks, developers typically receive highly contextualized information that they can apply to their specific needs or interests. Potential developers, who are learning about the OSS community culture before they start making code contributions, find these communications especially useful (von Krogh et al. 2003). von Krogh et al. (2003) found that, on average, developers posted 23.4 messages before making code contributions and the average time between the first post and first CVS commit was 40.8 days (von Krogh, et al. 2003). Consistent with social network theory (Freeman 1978), we believe that a developer's position within the communication network holds important implications for their ability to directly access information about how to contribute.

Broadly, social network theory posits that one's position within a network of relationships facilitates access to resources (Freeman 1978). Accessing these resources can allow an individual to perform better or improve other career outcomes (Ahuja et al. 2003). Some of the most commonly studied structural properties of a person's position within a social network include centrality (Ahuja, et al. 2003, Freeman 1978), closure (Coleman 1988), brokerage (Freeman 1978), and structural holes (Burt 2004). In this research, we focus on degree centrality. Degree centrality is the number of people to whom a person is directly connected. In a communication network, that is the number of people with whom they have directly communicated. We focus on degree centrality for several reasons. First, centrality in the communication network reflects a developer's linkages with other OSS community members (Sarker et al. 2011) and echoes the notion of communication networks as pipes through which information flows (Podolny 2001). This fits well with our interest in communication networks as an avenue for developers to enact interactive information seeking approaches to understand how to contribute within an OSS community. Further, degree centrality, compared to betweenness or closeness centrality, is appropriate when the context is related to communication (Freeman and Spyridakis 2004). In contrast, some of the other structural features of social networks such as brokerage and bridging of structural holes speak to the power and advantage that one can attain by having privileged access to information that is unevenly distributed or connecting different people in a network who might otherwise not be connected (Freeman 1978).

Second, centrality effectively reflects the extent to which a person sits at the nexus of information flow within the communication network (Freeman 1978). As such it provides an ideal lens for characterizing a developer's ability to understand contribution behavior through the communication that occurs between many different participants. Centrality within a communication network can enable a developer to learn about dynamic rapidly changing technical factors including awareness of other developers' work, their style of working as well as their skills (Dabbish et al. 2012). Finally, centrality is appropriately used when communication structure is less strongly dictated by formal structures compared to traditional organizations (Rice and Aydin 1991).

Taken together, we believe that congruence and incongruence in OSS values, provide opportunities to enact passive information seeking , and that communication network centrality provides opportunities to enact interactive information seeking. These serve as important relational strategies developers can use to gain certainty about how to make OSS development code contributions and such information is likely to engender greater commitment to the community. Next, we introduce and develop the logic behind our research model. We begin by arguing for the importance of considering the developer's OSS values within the context of the OSS community within which they are embedded. We then consider the implications of OSS values for commitment among developers who are not centrally located within the communication network versus those who hold more central positions. We ultimately link the influence of these factors on commitment to the code contributions made by developers.

HYPOTHESIS DEVELOPMENT

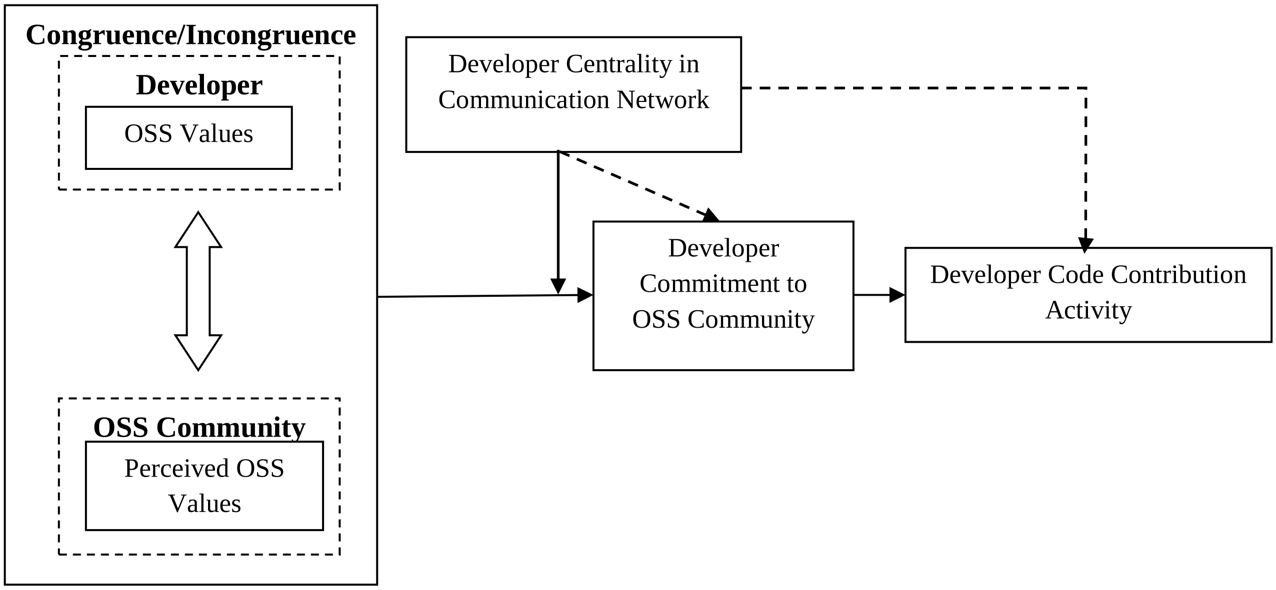

Figure 1. Research Model

The proposed model is shown in Figure 1. Previous empirical research supports the relationships indicated by the dotted lines connecting centrality to activity (Ehrlich and Cataldo 2012, Hinds and McGrath 2006, Sparrowe et al. 2001) and to commitment (Lee and Kim 2011, Siciliano and Thompson 2015) and are beyond the scope of this research. We focus on centrality as a moderator in the model for several reasons. First, prior studies demonstrate that psychological and network variables jointly impact outcomes (Vardaman, et al. 2015) and centrality, in particular, can alter the way negative attitudes impact outcomes (Pollack, et al. 2012, Vardaman, et al. 2015). For example, Vardaman et al. (2015) recently showed that an employee's centrality in the workplace social network weakened the relationship between their turnover intentions and actual turnover. Similarly, Pollack et al. (2012) provided evidence that centrality in a social network attenuated the positive relationship between economic stress and depressed affect.[1] Consistent with these and other prior works that treat social network position as a moderator, we are particularly interested in advancing understanding of how developer centrality moderates the influence of congruence and incongruence in person and environment OSS values. Second, PE fit and social network theory represent complementary views about how individuals are related to their environment. As such, we believe that an individual's social connectedness to their environment acts as an important boundary condition that shapes their reactions to congruence and incongruence in values espoused by the environment vis-à-vis commitment. Next we discuss the impact of OSS value congruence on commitment and then we describe how centrality in the communication network shapes the relationship.

Influence of OSS Value Congruence

As noted earlier, the OSS movement is defined by a sense of community and common purpose (Ljungberg 2000) and, in fact, Bagozzi and Dholakia (2006) suggest that OSS communities tend to be defined by a sense of "we-"ness. This sense of community, fueled by congruence between the developer and community values (i.e., seeing helping, sharing, cooperation and learning as important), evokes a sense of certainty about how to collaborate and interact with others in the community. When a developer's preferred way of working in the OSS environment matches that of the community, the developer feels he can predict how others will behave. This allows the developer to focus his energy on the joy of making technical contributions (Ke and Zhang 2010).

[1] Although we have not come across any empirical studies of this, we acknowledge the possibility that commitment can influence developer centrality—i.e., more committed developers may take actions that increase their centrality within the community. However, there are many circumstances in the OSS context in which the two maybe unrelated in this way. For instance, other developers may connect with a particular developer because she possesses a valued technical skill or because she is employed by a popular company (e.g., Google). This developer may not necessarily have formed an affective commitment to the OSS community. In any case, we empirically account for potential endogeneity between commitment and centrality in our robustness analysis.

Although an overarching set of OSS values that define the open source movement exists, not all OSS communities adhere to those values (Ljungberg 2000). Indeed, Ljungberg (2000) and others (e.g., Stewart and Gosain 2006) note that some communities include zealots who see these OSS values as a way of life while other communities view the OSS model as simply a means to an end—a way of developing software. Some of the differences in the extent to which communities adhere to OSS values may be attributable to the entry and active participation of organizations with commercial interests (Spaeth, et al. 2015, Stewart et al. 2006). In much the same way, individual developers differ in the extent to which they embrace the OSS values. Some developers participate in OSS communities for purely instrumental reasons (e.g., they need the software that is under construction), some are paid for their contributions, and others participate because they believe in the OSS movement (Fang and Neufeld 2009). These motivations for joining may not correlate with different values in that a developer who participates for instrumental reasons may not value cooperation, sharing and learning. Conversely, a person who participates for ideological reasons might value cooperation, sharing and learning (Ke and Zhang 2009, 2010).

Value congruence reflects a passive information seeking opportunity in that contribution actions taken by a developer may be reinforced by the community, giving the developer some sense that the behavior is acceptable. Observation of such positive feedback (and even just a lack of negative feedback) is consistent with passive information seeking approaches when there is uncertainty (Gibbs et al. 2015, Ramirez et al. 2002). By reinforcing the developer’s preferred approach to doing things—i.e., the developer’s preferred approach to contributing is also the community’s preferred approach—this validation provides clarity to the developer about how to engage (Byrne 1997). Developers are more likely to form an emotional attachment to an OSS community that shares a similar value system to their own—i.e., greater commitment (Arthur et al. 2006). Ke and Zhang (2010) argue that developers who identify with the values of an OSS project are likely to become more energized in their work on the project. Ke and Zhang (2009) note that OSS developers who feel their values are embraced in the community find the work more rewarding and worthwhile, suggesting commitment should be highest among developers who embrace OSS values and feel that the OSS community adheres to the same values.

Congruence in OSS values also suggests that developers and the community of which they are a part, share certain aspects of cognitive processing, fostering comparable methods of classifying and interpreting environment events, and a common system of communication (Meglino et al. 1989). Successful collaborative activities require these qualities because they reduce or eliminate uncertainty between participants (Fisher and Gitelson 1983, Schein 1985). Congruence in OSS values enables developers to know what to expect of others and to more accurately predict behavior (Kluckhohn 1951). Greater certainty about the process of contributing enables developers to enjoy participation in the community, increasing commitment.

OSS Value Congruence and Developer Centrality in the Communication Network

As the preceding discussion suggests, we anticipate congruence in OSS values will positively influence developer commitment to the community. We expect this relationship to be more salient among developers who are less central in the community's communication network. Connections in the communication network represent the pipelines through which information about ongoing community work flows (Zhang et al. 2013). Developers with limited connections in the communication network have few channels through which to assess the appropriate methods and procedures for contributing to the community (Steinmacher et al. 2015, 2016). Their opportunities for enacting interactive approaches is limited. OSS value congruence offers a passive approach to attain validation and reassurance of their preferred approach. Lower centrality in the communication network suggests less access to the flow of information that might otherwise help provide such validation. This makes OSS value congruence particularly salient in shaping less central developers' commitment to the community. It serves as the main channel through which they feel they fit in.

In contrast, developers who hold central positions in the communication network rely less on OSS value congruence in shaping their level of commitment to the community. Two main reasons underlie this belief. First, connections in the community's communication network can offer an awareness of the community's practices (Ahuja et al. 2003, Zhang et al. 2013). Highly central developers in the communication network sit at the nexus of the community's information flow. This provides an alternative, interactive channel for them to understand the contribution practices of the community. Interactive information seeking through direct communication has been argued to be more efficient and yield more accurate feedback in reducing uncertainty compared to passive approaches (Antheunis et al. 2010). Second, prior research suggests that centrality in social networks leaves individuals more embedded in their social contexts and less likely to voluntarily leave for fear of forfeiting their social capital (Borgatti and Halgin 2011, Lee and Kim 2011, Lee et al. 2004). As suggested earlier, these social connections themselves endear highly central developers to a community. Consequently, OSS value congruence, while important in influencing developers' commitment to the community, is likely to play a lesser role among developers who are highly central in the communication network.

H1: OSS value congruence will have a stronger positive relationship with commitment among less central developers than among highly central developers.

Influence of OSS Value Incongruence

In the same way that we expect low centrality to enhance the importance of OSS value congruence, we anticipate low centrality will enhance the influence of two forms of OSS value incongruence. In one form of OSS value incongruence, a developer perceives that the OSS community adheres to the OSS values more than he does. That is, the developer may not be in the community because of the values of knowledge sharing, learning, helping, cooperation, and reputation; but nevertheless, the community values these things. Incongruence also emerges when a developer embraces the OSS values more than the OSS community does. For instance, the developer may participate in a community in which knowledge sharing, learning, helping, cooperation, and reputation are not valued as highly as he values them. While incongruence typically negatively impacts outcomes, recent empirical research suggests that the two scenarios can lead to distinct impacts on outcomes (Edwards 1996, Edwards and Rothbard 1999, Jansen and Kristof-Brown 2005). We expect the two different type of incongruence to lead to distinct outcomes in the OSS context. We first consider incongruence when the OSS community embraces OSS values more than the OSS developer does and then we consider incongruence when the developer embraces the OSS values more than the community does. In both cases we theorize how the influence of incongruence on commitment is more potent for developers who are less central in the communication network.

Developer Centrality when the Community Embraces OSS Values More Than the Developer

Many developers expect that OSS communities embrace OSS values, even if they themselves do not. When the community embraces the OSS values, it affords developers a measure of predictability both in terms of what to expect from other participants in the community and in terms of the practices the community expects of them. According to uncertainty reduction theory, such predictability makes for a desirable environment as it reduces anxiety and stress

(Hogg et al. 2010, Hogg and Terry 2000). Given that the environment is desirable we expect developers who perceive the OSS community to adhere to OSS values more than them to be committed to the community. Edwards and Rothbard's (1999) idea of carryover reinforces this expectation. They argue that an individual who feels the environment provides more of something (in this case adherence to the OSS values) than he needs is likely to exhibit pro-social behaviors toward the organization. Consistent with this idea, such environments have been found to promote greater commitment (Ashford et al. 1989).

The positive impact of incongruence when the developer perceives the community adheres more to OSS values than he does is important for a developer with low centrality in the communication network because he may have few other avenues for more actively ascertaining acceptable behavior in the community (Steinmacher et al. 2015). For these developers the practices that underlie OSS values can substitute for the collaboration information to which they are not privy in the communication network. In digitally-enabled environments, such as OSS communities, developers can observe the behaviors dictated by community OSS values. The observation represents one of the few avenues for gathering information about appropriate ways of engaging for those who are not otherwise well-connected within the community (Lindberg et al. 2016). The OSS values manifest through digital traces of community member activities as they go about interacting with other community members and making code contributions. Digital repositories provide the requisite degree of permanence to these behaviors that allow developers to observe engagements over time. Indeed, many developers will spend a month or so observing how people contribute and behave in a community before making their first contribution (von Krogh et al. 2003). For such developers, code contributions are influenced by the commitment that is formed once they have had an opportunity to observe and understand the behaviors embedded in the values of the community. This process is consistent with a passive information seeking process to reduce uncertainty. Communities that embrace OSS values enact the underlying practices—i.e., they help, cooperate, and share knowledge—making them responsive to developer needs (Zhang et al. 2013). Such welcoming gestures should instill greater commitment on the developer's part, even if he does not embrace the same values.

In contrast to developers with low centrality in the communication network, developers with high centrality can actively use their connections to attain a measure of understanding and predictability with regard to the community's way of collaborating. Since they can use communication to access information and thereby lower uncertainty (Brashers 2007), a developer with high centrality in the communication network can get personalized responses to her queries about the process. Communication patterns help determine a person's ability to process information and as the complexity of the task increases the communication patterns needed to address the task change (Tushman 1979). Because of the complex nature of software development and the distribution of knowledge across developers, access to multiple developers helps a developer feel certain about how to do his work. For instance, there may be many analytic solutions to a programming problem requiring the developer to select the best solution. The developer could find information about the skills of co-developers useful in this decision making process. This makes highly central developers less reliant on the OSS values in shaping their ease of interaction with the contribution process and associated commitment to the community. Therefore, the impact of incongruence, where the OSS community embraces OSS values more than the developer does, is likely to be quite muted at best. The preceding arguments suggest that when the OSS community adheres to OSS values more than the developer does, the positive effect on commitment will be stronger among less central developers than among highly central developers.

H2a: When OSS community OSS values exceed developer OSS values, the effect of value incongruence on commitment will be more positive among less central developers than among highly central developers.

Developer Centrality when the Developer Embraces OSS Values More Than the Community

Next we consider the second case of value incongruence; when a developer embraces OSS values more than the community. We expect the impact of such incongruence on commitment to be stronger among developers who are less central in the communication network. A less central developer who embraces OSS values more than the OSS community may experience uncertainty. Instead of having his approach for developing software reinforced—as would be the case if the values were congruent—the OSS community's differing approach may cause him to question his approach or feel tension related to the discrepancy. Further, unlike the case where the OSS community strongly embraces the OSS values, in this case the developer cannot predict the community's processes, placing the developer in a situation of uncertainty. Edwards and Rothbard (1999) would describe this scenario as one of the environment not providing enough of a desired characteristic (in this case OSS values) for the developer and would expect anti-social attitudes to result. In line with this logic, Elliott and Scacchi (2003) document two cases where an OSS community participant experiences discomfort because he senses the rest of the community does not embrace OSS values enough. In the first case the developer speaks out against the method (non-free Adobe Photoshop software) used to produce a graphical representation of the system architecture on the GNU website. Over the next couple of days or so the argument about this practice continued and tension between the developer and the community rose. In the second case, a developer expressed his disagreement with the use of non-free tools to develop GNU documentation. The discussion lasted three days. This kind of tension could lead to frustration and lower commitment.

As the community battles about what is, or is not "OSS enough", time is lost on making code contributions. In these examples, the OSS community is not consistent with the OSS values leaving developers unsure about what to expect and what is acceptable. Uncertainty reduction theory suggests that people use communication when their sense of certainty has been compromised (Berger and Calabrese 1975). Unfortunately, limited social ties mean that developers with low centrality in the communication network may not have interactive information seeking as an option, creating further discomfort. Steers (1977) found that individuals who perceived their organizations to be undependable in carrying out their commitments to employees were, in turn, less committed to their organizations. Such a situation can make the community appear less welcoming to developers with low centrality; especially because they already have limited social ties to the community—making them feel isolated and not belonging (Vardaman, et al. 2015, Williams 2007). Their low embeddedness in the community, coupled with unpredictability regarding acceptable practices, are likely to strongly erode commitment.

In contrast, for highly central developers in the community's communication network, we expect this form of value incongruence to minimally impact commitment. First, connected developers can resolve any uncertainty that might arise from such incongruence through their connections in the communication network. Responses from social connections within the community can clarify acceptable practices and offer material support for any ramifications (Ehrlich and Cataldo 2012, Wagstrom 2009). Such connections provide highly central developers with insight into the embedded practices of the community, enabling them to make meaningful contributions with less trouble (Steinmacher et al. 2015). Second, highly central developers are embedded within the community. Such embeddedness affords them significant social capital within the community. Vardaman et al. (2015) argue that highly central individuals are generally reluctant to forgo the social capital they have built even when the environment pushes them to consider leaving. In much the same way, we expect that highly central developers in the communication network are unlikely to forgo their social capital on account of discrepancies in preferred practice in the community. Consequently, OSS value incongruence is less likely to impact the commitment of highly central developers. Taken together, the preceding arguments suggest that, in situations where the developer embraces OSS values more than the OSS community does, OSS value incongruence should have a stronger negative influence on commitment among less central developers than among highly central developers.

> *H2b: When developer OSS values exceed OSS community OSS values, the effect of value incongruence on commitment will be more negative among less central developers than among highly central developers.*

Developer Commitment and Contribution Activity

A developer's commitment to the community is expected to mediate the relationship between OSS value congruence and incongruence and code contributions to the community. Uncertainty reduction theory indicates that in relational situations, reduction in uncertainty fosters greater psychological commitment to the relationship in a way that guides future behavior (Knobloch 2015, Knobloch et al. 2010). This suggests that commitment plays an important role in connecting uncertainty reduction activities, such as information seeking (Knobloch 2015), to subsequent relational behaviors such as conforming to behavioral norms. Per uncertainty reduction theory, people are naturally drawn to predictability in behavior as it provides greater certainty about how to act within the relationship (Berger and Calabrese 1975). As elaborated in

H1 and H2, we expect OSS value congruence and incongruence to affect commitment—i.e., the emotional bond that developers feel toward the community. We expect commitment, in turn, to promote greater code contribution to the development process. The relevance of commitment in promoting pro-social behaviors in organizations is well established in the literature (Mathieu and Zajac 1990) and recent research on online communities identifies commitment as a chief antecedent of ongoing contribution behavior (Bateman, et al. 2011). Similarly, Ke and Zhang (2009) find that goal commitment in OSS communities is positively associated with developers' self-reported task performance. Drawing on these findings, we expect that when developers are committed to an OSS community, they care about its viability. Consequently, they are likely to make more contributions to the community to ensure its success (von Krogh et al. 2012). We expect OSS value congruence and incongruence to influence developers' code contribution activity through their impact on commitment.

H3: Commitment will mediate the relationship between OSS value congruence and incongruence and developer code contribution activity.

Building on the mediating logic outlined in H3 above, we expect the mediating role of commitment to be stronger among developers who are less central in the community's communication network and weaker among highly central developers in the community's communication network. Two main considerations provide a foundation for the rationale explaining the differences in the strength of the mediating role of commitment. First, as discussed earlier, interactive and passive relational uncertainty reduction mechanisms are two main alternatives that can be leveraged in digitally-enabled environments (Ramirez et al. 2002). Second, given the choice between the two uncertainty reduction mechanisms, interactive approaches—such as that represented in the communication network—are likely to be preferred by developers who are highly central in the communication network. Less well-connected developers may not have such an avenue.

Less central developers primarily have passive approaches at their disposal, making OSS value congruence and incongruence a main driver of commitment and concomitant code contribution. In contrast, highly central developers have two alternative avenues for gathering information about appropriate ways of engaging within the community. The passive approach reflected in observation of the activities of community members represents one avenue. The interactive approach to information seeking through their communication network connections is an alternative avenue that such developers can leverage. Uncertainty reduction theory identifies information seeking through direct communication as an important mechanism that people use to reduce uncertainty (Knobloch 2015). Given the choice between these two avenues, an interactive approach through the communication network is more preferable because developers can receive clear, direct, and immediate answers to their questions rather than having to observe and infer what constitutes acceptable forms of engaging within the community (Ramirez, et al. 2002). Their high centrality within the community's communication network affords them such access to information. Indeed, Antheunis et al. (2010) find that interactive approaches are the most effective in reducing uncertainty. Consequently, as noted in H1 and H2, highly central developers rely less on OSS values to guide their formation of commitment to the community and their code contributions are driven less by the mediated influence of value congruence and incongruence. As such, commitment should play less of a mediating role in linking OSS value congruence and incongruence to code contributions among such developers.

> *H4: Commitment will have a stronger mediating effect in the relationship between OSS value congruence and incongruence and developer code contribution activity among less central developers than among highly central developers.*

METHODOLOGY

Research Setting

We examined the hypotheses using data collected from the GNOME OSS community. Formed in 1997, the GNOME community creates a graphical user interface platform for Unix operating systems such as Linux. Over the following decades, volunteers and paid contributors from around the world contributed in order to create a freely available desktop platform and many applications. Red Hat, Cisco and Novell make contributions to GNOME and the number of paid and volunteer contributors continues to increase (Dahlander and Wallin 2006). GNOME project members utilize two tools to enable their work. Mailing lists represent the main mechanism for communication among project members (e.g., developers) as well as between project members and community members (e.g., developers in other projects). Source code repositories enable developers to make changes to the system and also allow them access to the history of the development activities.

Sample and Procedure

We used a combination of archival and survey data to test our model. The GNOME source code repositories allowed us to identify the collection of contributors across the various projects in GNOME as well as to extract developer contribution activity, developer degree centrality and control variables used in our analyses. From its inception in 1997 until 2010, numerous GNOME projects surfaced. As of August 2010, the GNOME community had 734 different projects. Since the requirements to create a project in the GNOME source code repository system remain relatively easy to meet, projects tend to differ significantly in their development activity, size, and participation rate. Building on criteria used in past research (Crowston et al. 2003), we only considered active projects that satisfied the following criteria: (a) continuity of development activity (at least one year), (b) amount of development activity (at least 100 commits)[2], (c) attractiveness of project for developers (at least 10 committers), and (d) user interest to participate (at least one community hosted mailing list). Use of projects that satisfied these criteria provided more complete information about email communication networks and developer contribution activity over time. Ninety-one projects satisfied these criteria and included 27 projects that had been active in the community from as early as 1998. These 91 projects are all of the projects that are part of the official GNOME distribution.

We identified a total of 2,576 developers who contributed at least one commit to the development effort from 1998 until December of 2010. 2,341 out of those 2,576 developers made contributions to the 91 projects identified as active given Crowston et al (2003)'s criteria. However, 553 developers did not make any contributions beyond 2005 and so we did not consider them in our analyses. The remaining 1,788 developers contributed 91.6% of the commits (595,327 out of 649,526) to those 91 projects throughout the period covered by our data. Furthermore, those 1,788 developers contributed 90.2% (596,320 out of 661,109) of all development activity across all 734 projects. Therefore, we consider the 1,788 developers to be a representative subgroup of the entire population of developers in the GNOME community.

We invited the 1,788 contributors to complete our survey. An initial email message went to the developers, asking for participation. One week later a follow-up reminder went out and three weeks after that a final reminder was sent. As an incentive, each participant who completed the survey earned entry into a drawing for a monetary prize. Two participants, randomly selected through the drawing, received $200.00 each. Five hundred and sixty-two individuals responded

[2] Source code repositories manage the software development process by keeping track of changes: what was changed, when it was changed, who made the change. A commit occurs when a developer uploads altered code.

to the survey (31% response rate) and 410 provided usable responses (filled in all survey questions used in analysis). The 410 respondents represented 55 countries distributed across North America (69), Europe (248), South America (20), Asia (45), Africa (4), and Oceania (13). Eleven participants did not indicate their location. The sample of respondents included 93.7% males. Mean age for the respondents was 29.7 years, with a standard deviation of 8.3 years. Three hundred and thirty respondents were employed. One respondent explained that he was blind and we surveyed him over the phone. Of the 410 respondents, 127 indicated that they were employed by a traditional organization and that their organization has assigned them to work on the GNOME project. Our analysis showed that our sample of 410 participants did not differ significantly in their patterns of contribution activity from the other 152 individuals who did not provide complete responses to the survey in terms of the number of projects in which the individuals participated ($z = -0.51$, $p = 0.61$), the number of commits made ($z = -0.40$, $p = 0.69$), the number of lines of software code contributed ($z = 1.70$, $p = 0.09$), the number of messages contributed to the mailing lists ($z = -1.41$, $p = 0.16$) and the number of messages posted on the defect tracking system of the GNOME community ($z = -1.30$, $p = 0.19$). Additionally, analysis showed that our sample of 410 participants did not differ significantly from the 1,226 developers who did not participate in the survey in terms of the number of projects in which the individuals participated ($z = -0.05$, $p = 0.96$), number of commits made ($z = -1.10$, $p = 0.27$), number of lines of software code contributed ($z = -0.95$, $p = 0.36$), and number of messages contributed to the mailing lists ($z = -0.78$, $p = 0.44$). The two groups differed in terms of the number of messages posted on the defect tracking system of the GNOME community ($z = -4.46$, $p < 0.00$).[3]

[3] Since we had pre- and post-survey data on network position, communication activity, and coding activity for all 1,788 developers, we conducted a Heckman two-step procedure to ensure that there was no selection bias in the sample that participated in the survey (Heckman 1979). The results of the Heckman selection model indicate that selection bias was not a concern in our models.

Measurement

Dependent Variables

Developer contribution activity (Number of commits). We counted the number of commits on all GNOME projects over the 9 months following the survey administration (between January 2011 and September 2011) to measure each developer's contribution activity. Since the main task of OSS development communities involves creating and extending software, we use the total number of CVS commits. A CVS commit occurs when a contributor uploads some new or modified software code to a project. CVS commits have been used to measure OSS productivity because CVS commits are similar to completed modification requests (MRs) in commercial development environments (Grewal et al. 2006, Midha and Bhattacherjee 2012, Mockus et al. 2002, Singh, et al. 2011).

Mediator Variable

Developer commitment. We modified the organizational commitment scale items to make GNOME the referent by substituting the word GNOME for organization. Sample questions include "I show by my actions that I really care about the fate of GNOME" and "I am extremely glad to have chosen GNOME to work for over other projects." The Cronbach's alpha reliability score for the scale is .79.

Independent Variables

Developer OSS values. We modified the values scale used by Stewart and Gosain (2006) to measure developer OSS values. Sample items include "I believe in helping others" and "I value the reputation I gain by participating in open source projects." Reliability for scale is .72.

Perceived OSS community values. In keeping with the P-E fit literature's use of commensurate measures, the items from the OSS values scale were modified to reference the

OSS community. Sample questions include: “Members of this OSS community believe in helping others” and “Members of this OSS community value the reputation gained by participating in open source projects.” The reliability of the scale is .87.

Moderator Variable

Developer degree centrality. Our measure of degree centrality represents the potential number of paths through which information—in particular project-related information (e.g. norms of conduct, technical aspects of the system, etc.)—spreads. The degree centrality measure (Freeman 1978) introduced guided the measure of degree centrality used here. We measure degree centrality as the number of developer *i*’s (direct) ties, normalized by the size of the network. This is given by N_i/N where N_i is the number of ties that developer *i* has and N is the total number of developers in the network.

We created dyads based on email senders and receivers. Mailing lists embody the main forum for posting questions as well as comments and responding to them. The topics range from specific technical discussions to broad discussions about processes, norms or strategy in the community. An individual was uniquely identified by determining the one or more email addresses that he used for his contributions in the mailing lists as well as in the commits (if they had any) based on an email address. The links are an interaction and the interaction is a discussion over email. For every email, we extracted from the email header the message identifier, the sender, the sent time, and the identifier of the message (if any) to which this message was a reply. The origin of an email message can be determined by the email address of the sender. The responder is more difficult to identify because when a person sends an email it can be read by anyone who subscribes to the list. But deciphering if a person reads the email remains difficult. So, we only create an interaction dyad based on the people who explicitly replied to an email. We assume that the responder read the message. When a “reply-to” header was found, the responder found the initial message of interest; and so the responder was marked as a recipient of the original message. Any person who responds to the email is connected to the sender of the email (whether the person is the first or 31st responder). The sample includes emails sent during the 12-month period (December 2009 and December 2010). The GNOME community product release cycle, which occurred once every 6 months, informed the selection of the 12-month period used. The releases typically take place in March and September. Considering CVS commit and communication data over 12 months allows us to capture the typical nature of the work around two releases.

Control Variables

Using data collected through the survey, we controlled for gender, age, education, and volunteer status (i.e., whether the developer was paid to contribute or was volunteering) because these factors may impact commitment (Brierley and Cowton 2000, Hrebiniak and Alutto 1972). All models include these control variables.

Results

In order to assess the validity of the measurement scales, we conducted a confirmatory factor analysis using varimax rotation (Fornell and Larcker 1981). Results of the factor analysis are shown in Appendix B. With the exception of two items, all items had loadings above .66 on their expected constructs and cross-loadings lower than .31, thus demonstrating adequate convergent validity and discriminant validity. The two items with low loadings were removed from further analysis. In addition, the square root of the average variance extracted for each construct was higher than the correlation between that construct and all other constructs, further demonstrating the discriminant validity of the measures (Fornell and Larcker 1981). Table 1 displays descriptive statistics and correlations for all measures.

Table 1. Correlations and Descriptive Statistics.

Variable	Mean	SD	1	2	3	4	5	6	7	8
1. Developer contribution activity	30.21	57.59								
2. Developer commitment	4.77	1.15	.23***	(.89)						
3. Developer values	6.34	0.51	.04	.27***	(.71)					
4. Perceived OSS community values	5.78	0.86	.10*	.51***	.33***	(.86)				
5. Developer centrality	0.01	0.05	.29***	.10*	.02	.01				
6. Age	29.7	8.26	-.02	-.06	-.01	-.01	.02			
7. Gender	NA	NA	.06	.07	.09†	.08†	.05	-.04		
8. Education	NA	NA	.03	-.09†	.07	.01	.05	.20***	.10*	
9. Volunteer status	NA	NA	-.14**	-.06	.11*	-.03	.06	.01	-.05	.04

Notes: n = 410.

1. For the purpose of interpretability, the non-transformed means and standard deviations for number of commits are shown.
2. Gender = dummy variable (0 = women, 1 = men), volunteer status (0 = not paid, 1 = paid).

† $p < .10$, * $p < .05$, ** $p < .01$, *** $p < .001$.

Polynomial Regression Analysis

The hypotheses in the research model emphasize the effects of congruence and incongruence in values between developers and their perceptions of the environment (the OSS community). Prior research on congruence often relies on single-index measures such as difference scores, direct measurement, or profile similarity indices (Edwards 1993, Klein et al. 2009). However, due to limitations related to conceptual ambiguity, discarded information, and unrealistically restrictive constraints these approaches received criticism (Edwards and Parry 1993). In contrast, polynomial regression analysis incorporates separate person and environment ratings on commensurate measures and permits a high degree of precision in specifying and testing relationships about congruence and incongruence (Brown et al. 2012, Brown et al. 2014, Edwards 2002, Klein, et al. 2009). For these reasons, polynomial regression is well-suited for our examination of the effects of OSS value congruence and incongruence on commitment.

Polynomial regression analysis involves the estimation of coefficients for higher-order terms (e.g., quadratic terms) and such terms are especially susceptible to outliers in the data (Edwards 2002). To determine the extent to which this might raise concern in our sample, we screened the data for outliers by examining Cook's D and the standardized residuals from the polynomial regression equations (Bollen and Jackman 1990). Although three cases were flagged as potential outliers, no major outliers emerged from the analysis. Estimating the models with and without these observations did not meaningfully affect the results. Two or three cases rarely severely influence observed patterns of relationships in samples of this size (Bollen and Jackman 1990). Finally, we mean-centered the variables prior to computing the interaction and quadratic terms in order to reduce the potential for non-essential multicollinearity and facilitate better interpretation of the response surface plots (Aiken and West 1991, Dalal and Zickar 2012).

The regression coefficients from a polynomial regression equation enable the generation of three-dimensional response surface graphs to depict the relationship between two commensurate measures (e.g., developer values and perceived OSS community values) and their effect on an outcome (e.g., developer commitment). These graphs facilitate better interpretation of the precise nature of congruence and incongruence relationships (e.g., Brown, et al. 2012, Brown, et al. 2014). Given our focus on the influence of value congruence and incongruence on developer commitment among developers with different levels of degree centrality, we used moderated polynomial regression to test the hypotheses and the response surface methodology to interpret the results. In polynomial regression analysis, value congruence is operationalized by all values at which developer OSS values (X) are equal to perceived OSS community values (Y)—i.e., all cases where X = Y. Tests of the relationship between OSS value congruence and commitment involve the use of non-parametric techniques to determine if the slope of the relationship is statistically significantly different from 0. Value incongruence is operationalized by all values at which developer OSS values are different from perceived OSS community values. This involves cases where developer OSS values exceed perceived OSS community values (X > Y) as well as cases where perceived OSS community values exceed developer OSS values (X < Y). Tests of the relationship between value incongruence and commitment employ non-parametric tests to determine if their effects are statistically significantly different from 0.

In polynomial regression analysis, simple regression models are tested and then progressively higher-order terms enter the equation. The simpler model is rejected when a higher-order model explains statistically significantly greater additional variance and when the constraints imposed by the simpler model are rejected (Edwards 2002). Following convention in polynomial regression analysis, we estimated the following equations to represent the theoretical model (Edwards and Parry 1993):

$Z = b_0 + b_1X + b_2Y + e$ (1)

$Z = b_0 + b_1X + b_2Y + b_3X^2 + b_4XY + b_5Y^2 + e$ (2)

$Z = b_0 + b_1X + b_2Y + b_3X^2 + b_4XY + b_5Y^2 + b_6W + b_7WX + b_8WY\ b_9WX^2 + b_{10}WXY + b_{11}WY^2 + e$ (3)

where Z = developer commitment, X = the developer's values, Y = perceptions about the OSS community's values, and W = developer's degree centrality within the OSS community.

Confirmatory Polynomial Regression Analysis

Polynomial regression analysis also makes it possible to empirically test specific constraints that might be imposed on the specification of the regression model by a theoretical view point. With few exceptions, much of the extant literature on the impact of OSS values has implicitly assumed that either only the OSS values of the developer matter or only the OSS values of OSS community matter in predicting developer outcomes such as commitment. Polynomial regression analysis can empirically test these constraints (Brown et al. 2014, Edwards 2002). If developer OSS values alone matter in predicting developer commitment, then as, Brown et al. (2014) suggest, the following constraints would be imposed on Equation (1):

Constraint 1: $b_1 > 0$ (i.e., the coefficient on developer OSS values should be positive)

Constraint 2: $b_2 = 0$ (i.e., the coefficient on OSS community values should be non-significant)

Conversely, if OSS community OSS values alone matter in predicting developer commitment, then the following constraints would be placed on Equation (1):

Constraint 1: $b_1 = 0$ (i.e., the coefficient on developer OSS values should be non-significant)

Constraint 2: $b_2 > 0$ (i.e., the coefficient on OSS community values should be positive)

Table 2 shows the results of the polynomial regression predicting OSS commitment. In model 1, we tested Equation (1). As the results for model 1 show, the linear terms for developers' OSS values and perceptions of OSS community values explained 27 percent of the variance in developer commitment. The coefficients on developer OSS values and perceived OSS community values are positive and significant. Thus, the constraints implicitly imposed in prior research are rejected. This suggests that there is value in considering the joint effects of developer OSS values and perceived OSS community values on developer commitment. In light of the constraints being rejected, we proceeded to perform an exploratory polynomial regression analysis in which the constraints were relaxed.

Table 2. Results of Moderated Polynomial Regression Analysis Predicting Developer Commitment

Variable	Coefficient	1	2	3
Age		-.03	-.04	-.04
Gender		.03	.03	.03
Education		-.10*	-.10*	-.10*
Volunteer status		-.05	-.05	-.05
Developer values	b_1	.16***	.16*	.15*
OSS community values	b_2	.45***	.50***	.56***
Developer values-squared	b_3		.13*	.13*
Developer values x OSS community values	b_4		-.14*	-.09†
OSS community values-squared	b_5		.15*	.21**
Developer centrality	b_6			.15**
Developer values x developer centrality	b_7			.07
OSS community values x developer centrality	b_8			-.16**
Developer values-squared x developer centrality	b_9			.00
Developer values x OSS community values x developer centrality	b_{10}			.07
OSS community values-squared x developer centrality	b_{11}			.16**
Lowest VIF		1.023	1.025	1.030
Highest VIF		1.098	3.303	4.435
Adjusted R^2		.27***	.30***	.33***
ΔR^2			.03***	.03***

Notes: n = 410.
* p < .05, ** p < .01, *** p < .001.

Exploratory Polynomial Regression Analysis

In model 2, we tested Equation (2). The results show that the quadratic model (model 2) explained statistically significantly greater variance in developer commitment than did the linear model (model 1) ($R^2 = .30$, $\Delta R^2 = .03$, $p < .05$) resulting in a medium effect size of .43 (Cohen et al. (2003). Therefore, per the guidelines of Edwards (2002), we reject the linear model in favor of the quadratic model. In model 3, we tested for the moderating effect of developer degree centrality. This model explained statistically significantly greater variance in developer commitment than did the quadratic equation with no moderation (i.e., model 2) ($R^2 = .33$, $\Delta R^2 = .03$, $p < .05$) resulting a medium effect size of .49.[4]

Response Surface Methodology

The polynomial regression equations represent an important first step in modeling the joint effects of commensurate measures on the outcome of interest. To meaningfully interpret the results we graphically plot these relationships and examine their key features (Edwards 2002). The response surface methodology provides the necessary analytical tools to conduct hypothesis tests of various features of the response surface. Given our focus on the effects of OSS value congruence and incongruence on developer commitment, we tested the slope of the surface along these specific lines of interest. In light of the significant moderating effect of developer social network centrality, we graphed the response surface at high (i.e., one standard deviation above the mean) and low (i.e., one standard deviation below the mean) levels of developer centrality.

[4] We also estimated a cubic model to see if the moderated quadratic model would be rejected. The cubic model did not explain any additional variance in OSS commitment. Therefore, model 3 was retained for the purposes of the response surface analysis.

Testing the significance of the slopes along the line of congruence (i.e., the line along which developer OSS values equal perceived OSS community values) and incongruence (i.e., the line along which developer OSS values and perceived OSS community values are unequal) at high versus low levels of developer centrality requires non-parametric techniques. Following the recommendation of Edwards (2002), we used a bootstrapping procedure to test the significance of the slope of the response surface along the line of congruence and incongruence (Efron and Tibshirani 1993). Bootstrapping is generally preferred over jackknifing for small sample sizes such as that used in this study (Efron and Tibshirani 1993). Using the bootstrapping approach, we constructed bias-corrected confidence intervals around the estimates of the slopes of the response surface based on equation (4) below (Edwards 2002).

$$Z = (b_0 + b_6W) + (b_1 + b_7W)X + (b_2 + b_8W)Y + (b_3 + b_9W)X^2 + (b_4 + b_{10}W)XY + (b_5 + b_{11}W)Y^2 + e \quad (4)$$

where values of W at one standard deviation above versus below the mean for developer centrality were used to compute the simple coefficients for the response surface.[5]

Table 3. Results of Tests of Slopes Along Lines of Congruence and Incongruence.

	Surface along line of congruence H1 (X = Y)		Surface along line of incongruence H2 (X = -Y)	
Level of centrality (W)	Linear slope (a_x)	Curvature (a_x^2)	Linear slope (a_y)	Curvature (a_y^2)
High centrality	.69***	.31	-.35	.45
Low centrality	.73***	.19	-.47*	.41

Notes:

1. The linear slope of the response surface along the line of congruence is given by $a_x = (b_1 + b_2 + [b_7 + b_8]W)$ and the curvature of the slope along this line is given by $a_x^2 = (b_3 + b_4 + b_5 + [b_9 + b_{10} + b_{11}]W)$
2. The linear slope of the response surface along the line of incongruence is given by $a_y = (b_1 - b_2 + [b_7 - b_8]W)$ and the curvature of the slope along this line is given by $a_y^2 = (b_3 - b_4 + b_5 + [b_9 - b_{10} + b_{11}]W)$
3. Significance levels of estimated coefficients are based on bias-corrected confidence intervals constructed from 10,000 bootstraps at low and high centrality.
4. * $p < .05$, ** $p < .01$, *** $p < .001$.

[5] The simplified polynomial regression equations at low versus high values of developer centrality are shown in the Appendix C.

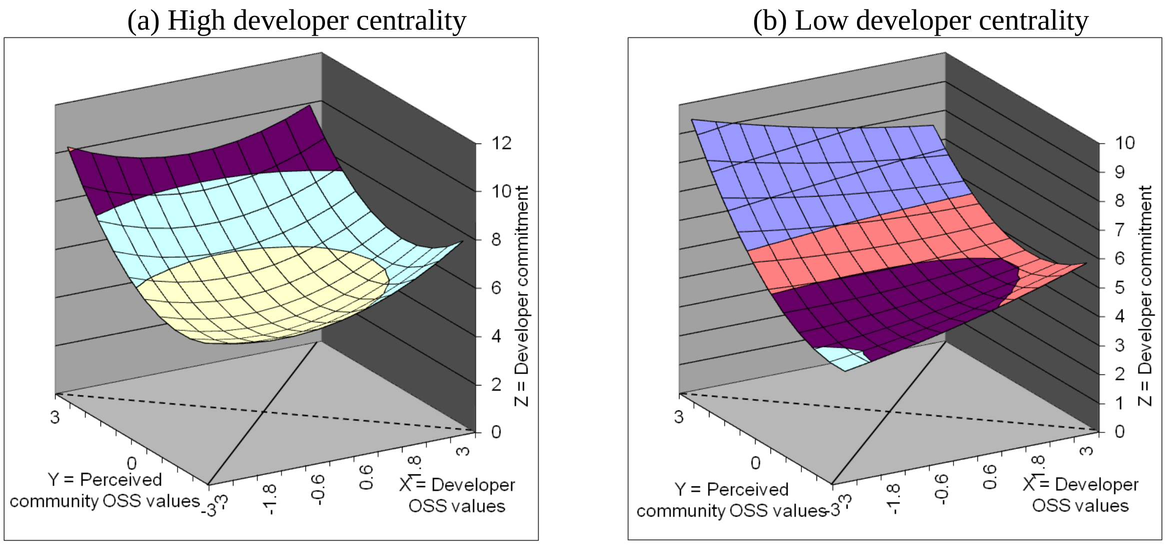

Figure 2. Graphical Plots of the Response Surfaces for High versus Low Developer Centrality

Tests of Hypotheses Regarding Effects of OSS Value Congruence and Incongruence

Table 3 presents the results of the tests of slopes along the line of congruence and incongruence and Figure 2 shows the graphical plots of the response surfaces. In H1, we posited that OSS value congruence would have a stronger positive effect on commitment among less central developers than among highly central developers. As the results in Table 3 show, when developer centrality is high, the slope of the surface is positive ($a_x = .69$, $p < .001$). When developer centrality is low, the slope of the surface along the line of congruence is positive ($a_x = .73$, $p < .001$). The linear slope of the response surface along the line of congruence is stronger when developer centrality is low than when developer centrality is high. This supports H1.

H2a predicted that, as OSS community values exceed developer OSS values the effects on commitment would be more positive among less central developers than among highly central developers. H2b posited that, as developer OSS values exceed OSS community values the effects on commitment would be more negative among less central developers than among highly central developers. For these hypotheses to be supported, the linear slope of the response surface along the line of incongruence should be more negative among less central developers than among highly central developers. As the results in Table 3 show, the linear slope of the response surface along the line of incongruence is negative and significant among less central developers ($a_y = -.47$, $p < .05$) and negative but non-significant among highly central developers ($a_y = -.35$, $p > .10$). This provides support for H2a and H2b. This suggests that, among highly central developers, the effects on developer commitment are the same whether $X > Y$ or $X < Y$. The significant negative linear slope of the response surface for less central developers also suggests that commitment increases as OSS community values exceed developer OSS values; commitment decreases as developer OSS values exceed OSS community values.

Test of Mediation Hypothesis

In H3 we posited that the relationship between value congruence and incongruence on developer code contributions would be mediated by commitment. To test this hypothesis, we conducted mediation tests using bootstrapping (Edwards and Lambert 2007, Preacher et al. 2007), which makes no assumptions about the shape of the sampling distribution for the mediation effects and, thus, is robust to non-normal distributions (MacKinnon et al. 2004, Preacher and Hayes 2004).

Before commencing with the mediation tests, we computed a block variable for value congruence and incongruence using equation (2) to account for the interactive, linear and non-linear effects of the two variables (developers' values and perceived OSS community values). Klein, et al. (2009) recommend this approach for testing the effects of polynomial regression coefficients in path models such as mediation and moderated-mediation models. Following the procedure outlined by Cable and Edwards (2004), we regressed the dependent variable (number of commits) on the five value congruence terms using negative binomial regression analysis.[6]

---

[6] Negative binomial regression is robust to the non-normal distribution reflected in count data such as number of commits.

The coefficients were then used to compute the block variable. This block variable was used as the independent variable in the mediation tests. To estimate the mediation (indirect) effects and the confidence intervals for the differences between estimates at high versus low levels of the moderator (developer centrality), we drew 1,000 bootstrap samples. To account for any potential difference between the product of coefficients from the full sample and the median estimate from the bootstrap samples, we computed bias-corrected confidence intervals (Edwards and Lambert 2007, Efron and Tibshirani 1993, Stine 1989). Table 4 shows the results of the analysis (see the Appendix for coefficient estimates for the moderated regression models).

Table 4. Results of Mediation (H3) and Moderated-Mediation (H4) Tests on Number of Commits.

Hypothesis	Level of developer centrality	Direct effect	Indirect effect	95% bias-corrected CI		R^2
				Lower bound	Upper bound	
H3		.55 (.31)	.32 (.14)	.08	.63	.21***
H4	High	.41 (.33)	.39 (.19)	.02	.97	.30***
	Low		.58 (.19)	.28	.99	

Notes: For H4, value congruence and incongruence block variable is the independent variable, developer commitment is the mediator, developer centrality is the moderator; control variables were included as covariates in the analysis; standard errors are in parentheses; dependent variable was log-transformed prior to analysis
*** $p < .001$.

The results in Table 5 show that value congruence and incongruence does not have a significant direct effect on number of commits (.55, $p > .10$); instead the mediated effect through commitment is significant as indicated by the bias-corrected confidence intervals (indirect effect: .32, CI: .08, .63). This provides support for H3.

In H4, we posited that the mediation effect of commitment in the relationship between value congruence and incongruence and developer code contributions would be stronger among less central developers than among highly central developers in the communication network. Thus, we tested the extent to which the mediation effect in H3 was moderated by developer centrality. We drew 1,000 bootstrap samples to estimate the confidence intervals for the differences between mediation effects at high versus low levels of the moderator (developer centrality). The mediated effect of value congruence and incongruence on number of commits was significant at high (indirect effect = .39, CI: .02, .97) and low (indirect effect = .58, CI: .28, .99) levels of developer centrality. Tests of differences in the mediated effects at high versus low developer centrality were non-significant (.39 - .58 = -.19, $p > .10$), suggesting that there was no difference in the indirect effect of value congruence and incongruence on developer code contributions. H4 is not supported, suggesting that the moderating effect of developer centrality is mediated by developer commitment.

Robustness Analysis

We conducted additional analyses to assess the robustness of our results. First, in order to ensure that our results were not an artifact of the timeframe upon which our degree centrality measure was based, we conducted the moderated polynomial regression analysis using degree centrality based on the past 24 months of email communication. The pattern of results was consistent across this alternative operationalization as shown in Appendix E. Second, in order to ensure that the results were not an artifact of the use of email communication as the basis for computing degree centrality, we computed degree centrality based on developers' collaboration on the same projects. We examined degree centrality based on collaboration on the same projects in the past 12 months and the past 24 months. As the results show, the pattern of results was similar to that of the analysis using degree centrality based on email communication. Third, an alternative argument could be made that developers who have greater commitment to the OSS community are more likely to be highly central in the communication network.[7] Empirically, our study design does not lend itself to testing the influence of commitment on centrality since our data

[7] We thank an anonymous reviewer for drawing our attention to this possibility.

were time lagged. That is, the developer centrality variable is based on email communication *before* commitment was measured. However, we acknowledge that this does not preclude the possibility of there being endogeneity between the two variables. Consequently, we estimated the polynomial regression models using an instrumental variables two-stage least squares (2SLS) analysis. The instrumental variables used include the number of messages previously posted to the listserv, the number of replies previously posted to the listserv and the number of previous commits made by a developer. As shown in Appendix F, the pattern of results from the 2SLS are similar to those of our main analysis. This provides additional confidence in the robustness of the model specification and results.

Finally, we wanted to ensure that the results of the mediation analysis were not simply an artifact of the specific operationalization of developer contribution activity as number of commits. Therefore, we repeated the analysis using two alternative operationalizations: number of lines of code added/deleted and number of files changed. As the results in Appendix G show, the mediating role of commitment is stronger among less central developers than among highly central developers in predicting number of lines of code added and deleted (test of differences: $.75 - .50 = .25$, $p < .05$) and number of files changed (test of differences: $.48 - .32 = .16$, $p < .05$). Specifically, we find that the mediating effect of commitment is non-significant among highly central developers and is significant among less central developers, providing support for H4. It is possible that number of commits is a less granular measure of the volume of work compared to the actual lines of code and files changed.

DISCUSSION

The OSS approach to developing software—which has traditionally been enshrined in the OSS values—has gained success over the years in large part by attracting the participation of developers. Given the increasing corporate presence in OSS communities, practices have changed to varying degrees (Spaeth et al. 2015, Stewart et al. 2006, von Krogh et al. 2012) and researchers and practitioners need to understand what happens when a developer grapples with issues of gaining certainty about how to participate in the community. A mismatch in values can fuel uncertainty and prior research suggests negative outcomes related to value incongruence (e.g. lower engagement and job performance) (Kristof-Brown, et al. 2005). However, OSS communities with corporate influence desire to retain the ongoing contributions of developers with different values. After all, corporate entities participate to reap the benefits of volunteer developer efforts. To advance our theoretical understanding of how developer commitment and ongoing contributions can be fostered in OSS communities, we focused on the role of OSS value congruence, incongruence, and centrality in the communication network as opportunities for passive and interactive information seeking to reduce uncertainty about how to participate in code contribution. That is, we integrated the PE fit perspective with social network theory. Our results indicate that the impact of OSS value congruence and incongruence is more salient among developers who are less central in the community's communication network. Next, we discuss the theoretical implications of our findings.

Theoretical Contributions

Implications for OSS Community Research

This research makes several contributions to the literature. From a theoretical standpoint, this research leverages uncertainty reduction theory to advance understanding of the roles of OSS values and communication networks as uncertainty reduction mechanisms. Prior research in this domain has not considered OSS values and communication networks in this light. By leveraging uncertainty reduction theory, we explain that these constitute alternative *relational* information seeking mechanisms that facilitate developers' ability to meaningfully participate in making code contributions by clarifying appropriate collaboration practices. Using this theoretical lens, we gain insight into the role of OSS values as a relational mechanism that facilitates developers' passive information seeking approaches to uncertainty reduction; and communication networks as a relational mechanism that facilitates interactive information seeking approaches to uncertainty reduction. These avenues represent an important means for enabling developers to contend with some of the common barriers to participation in digitally-enabled communities such as OSS (Steinmacher et al. 2015, 2016). Prior research has not considered the substitutive nature of the relationship between these two relational aspects of developers within their OSS communities. This is theoretically significant because it points to a degree of equifinality in fostering developer commitment to an OSS community. By using this theoretical perspective we add to the current body of work that has used theories including self-determination theory (Ke and Zhang 2009) and legitimate peripheral learning to understand code contributions in OSS development (Fang and Neufeld 2009).

We contribute to theoretical understanding of OSS values and their impact on OSS development. As our review suggests, this topic has received considerable attention in the IS literature since the seminal work by Stewart and Gosain (2006). Much of the prior research on this topic theorized the impact of OSS values in isolation, focusing only on developers or only on OSS communities. Consequently, mixed, and often contradictory, conclusions were reached about the impact of OSS values on developer attitudes and behavior based on an incomplete view of developers and their community environment. Our research shifts the conversation in this research stream by demonstrating the importance of understanding developers *within* their community environment. This repesents an important reorientation in the focus of research on

OSS values because it adopts a more holistic and relational view of developers and the communities in which they participate vis-à-vis congruence and incongruence. The results of our research clearly demonstrate that the OSS values of the developer *relative* to their perceptions of the OSS values of the community shape developers' commitment. This finding enables future research to explicitly account for developers within the context of their community environment.

This research contributes by outlining a *developer-within-community-environment* (i.e., OSS value congruence and incongruence, and social connectedness) → *attitudes* (i.e., commitment) → *contribution behavior* (i.e., code commits) chain. While myriad factors drive code contributions (Zhou and Mockus 2015), our findings indicate that the emotional attachments reflected in commitment represent an important mechanism that informs developers' code contributions to an OSS project. This mediating mechanism is particularly relevant when considering developers within the context of the OSS community environment. Our model augments prior research that links commitment to participation behavior in online communities (e.g., Bateman et al. 2011) by considering the influence of the environment. The view reflected in our model explicitly recognizes that individuals in online communities do not operate in a vacuum; the community of which they are a part has its own agency that shapes their attitudes and behavior.

This research contextualizes uncertainty reduction theory to digitally-enabled forms of organizing. Although uncertainty reduction theory has been applied to a variety of digital domains including online dating (e.g. Gibbs et al. 2015), social networking (Anthuenis et al. 2010), and access to e-government services (Venkatesh et al. 2016); OSS communities present a different set of domain-related problems. Chief among these is the lack of clarity around norms for contributing to a collaborative effort. By integrating insights from PE fit, social networks and uncertainty reduction theory to the OSS domain, we shed light on the relational mechanisms that facilitate uncertainty reduction in digitally-enabled collaborative environments. In such large-scale collectives, we highlight the important role played by digital traces of other developer contributions and interaction behavior as providing observability of appropriate approaches to engage within a community. These provide the basis for passive information seeking approaches to reducing uncertainty for developers about how to contribute to the collaborative effort. We also identify communication networks as a prime relational mechanism for interactive approaches to uncertainty reduction. We contextualize this to digitally-enabled collectives by harnessing the power of centrality within these communication networks. Prior uncertainty reduction research has not identified such theoretical mechanisms in digitally-enabled large-scale collectives such as OSS communities.

Finally, from an empirical standpoint we contribute to the literature through our moderated polynomial regression model. To the best of our knowledge, this is among the first empirical studies to examine moderation effects of polynomial regression within the information systems literature. With the growing number of studies employing this analytical method, this research provides a useful blueprint for testing boundary conditions that might affect the topology of response surfaces.

Implications for the PE Fit Perspective

This research makes a theoretical contribution to the PE fit perspective. Specifically, by incorporating the moderating role of centrality in the communication network, we advance the PE fit perspective by explicitly recognizing that the relationship between individuals and their environment is characterized by two layers. The first layer comprises the foundation upon which the PE fit perspective is built—namely the relation between an individual and their environment with regard to a specific characteristic (e.g., OSS values). The second relational layer which we incorporate into the PE fit perspective concerns the individual's connectedness to the environment. This second relational layer gives meaning to the salience of congruence and/or incongruence in the first relational layer. This is the first study to theoretically combine these two relational views to provide a holistic appreciation for individuals *within* their environments. It answers a call by prior reviews and meta-analyses of PE fit research to develop a more theoretically nuanced understanding of the relationship between people and the environments in which they work (Kristof-Brown and Guay 2011). This research suggests that, in order to develop a better understanding of phenomena, future PE fit research needs to account for the connectedness of the "person" aspect of PE fit within the environment being studied. After all, social connectedness has been shown to attenuate the impact of negative individual attitudes on actual negative behavior in organizational settings (see Vardaman et al. 2015 for an example).

Our examination of the impact of value congruence and incongruence in the context of an OSS community contributes to the broader PE fit literature by considering an understudied context (Johns 2006). Johns (2006) underscored the value of theorizing how well-studied phenomena unfold in different contexts. Open collaboration environments such as OSS communities rely on a different set of governance mechanisms than the traditional organization settings in which PE fit has typically been studied (O'Mahony and Bechky 2008, O'Mahony and Ferraro 2007, Seidel and Stewart 2011). As a form of virtual organization, these communities are heavily reliant on the collaborative effort of a voluntary and paid workforce. This context brings two key elements to the foreground that are not as salient in typical PE fit study settings. First, in OSS communities, the values come to embody preferred approaches to collaborating. This is significant because the membership of OSS communities as virtual organizations is fluid (Faraj et al. 2011). As such it is vital that acceptable practices be enshrined in the values of the community because newcomers enter frequently and are likely not familiar with acceptable behaviors (Steinmacher et al. 2015). One implication of this contextual difference is that, counter to meta-analytic findings in traditional organizational settings, excess values—i.e., conditions where the OSS community embraces the values more than the developer does—results in greater commitment among developers. Kristof-Brown et al.'s (2005) meta-analytic review states that "excess E conditions have little negative effect on attitudes, whereas excess P conditions accompany dramatic decreases in attitudes" (p. 313). Drawing on uncertainty reduction theory, we argue that the nature of OSS communities makes clarity around practices especially important and desirable even when they are at odds with an individual's own preferences. This is particularly salient for socially isolated developers within the community's communication network. In virtual organizations, such as OSS communities, digital communications serve as the primary channel for understanding how to contribute within the community.

According to the seminal attraction-similarity-attrition (ASA) framework, when employees do not fit with their employer they leave the organization (Schneider 1987). However, recent work suggests that value incongruence does not always lead to turnover (Kristof-Brown, et al. 2005). We follow recent work that seeks to understand the factors that suppress the negative effects of incongruence (Soltis 2012, Vogel et al. 2015). Like prior research we assume value exists in maintaining participation from individuals who may not hold the same values as the organization because they can bring alternate view points and increased creativity. By showing that communication centrality alters the relationship proposed by Schneider (1987) we contribute to the ASA model.

Implications for Practice

We stated at the outset that a major challenge for OSS communities rests in attracting the commitment of developer contributions. This challenge has its roots at least partially in the shifting make-up of OSS communities as commercial firms increasingly devote their own resources (including their own developers) to such communities (Aksulu and Wade 2010, Dwoskin 2016, Spaeth, et al. 2015, Stewart, et al. 2006). While OSS community managers obviously hold little control over who contributes and what values they embrace, the research findings suggest that it is better to err in favor of greater clarity regarding the acceptable practices of the community. This clarity holds unique importance for gaining the commitment of less well connected developers who operate on the periphery of the community. Prior empirical evidence suggests that such developers stand out in their ability to promote the popularity and quality of OSS projects (Setia et al. 2012). Developers face real barriers (Steinmacher et al. 2015) and OSS communities can ill-afford to lose such a vital resource. Embracing the OSS values to emphasize cooperation, helping, and the pursuit of technical knowledge clearly increases the commitment of such developers, even if they do not embrace those values.

As hybrid forms of software development become increasingly common (Aksulu and Wade 2010), OSS community managers must look to alternative ways of endearing themselves to developers in a manner that is robust to the fact that OSS values surely differ. In this regard, the research findings suggest that having more social connections within the communication network is one such mechanism for maintaining developer commitment. Importantly, a high degree of connectedness within the community’s communication network mitigates the negative influence of OSS value incongruence. OSS community managers may benefit from efforts to ensure that developers on the periphery of the community can access a pathway to becoming more connected to other members of the community. For instance, community managers can embed gamification incentives—e.g., badges, leaderboards—that reward establishment of social connections to less well-connected developers. Such efforts might encourage developers to reach out to less well-connected members of the community and explicitly bring them into the fold.

Limitations and Future Research

As with any field research, our study has several limitations that others should take into account when interpreting the results. A couple limitations exist because of the developers we sampled. First, observing GNOME developers allowed us to naturally control for unobserved between-community differences that might impact the results. However, at the same time it also limits the generalizability of the conclusions. GNOME remains much larger than many other OSS communities and probably receives more contributions from corporate sponsors and so generalizing to smaller communities must be done with caution. However, it is worth noting that GNOME represents how an increasing number of OSS communities evolve as corporate sponsors increase their engagement (Aksulu and Wade 2010, Dwoskin 2016).

An associated limitation related to the sample of GNOME developers is that the sample of GNOME developers may not represent all developers in the GNOME community. To minimize concern related to this issue we verified that on key variables (e.g., the number of concurrent versions system commits, the number of emails) the developers in our sample do not differ significantly from the larger set of developers in the GNOME community. Further, the results of our Heckman (1979) selection model indicated that selection bias was not a concern. The results of this analysis suggest that the sample likely represents the broader community.

One other limitation related to the sample is that the developers were predominantly male. Although the literature suggests that males make most code contributions to OSS development (Ghosh 2002, Hars and Ou 2002, Jensen, et al. 2011, Ke and Zhang 2009, Kuechler et al. 2012, Nafus 2012, Robles et al. 2016), one must still question the degree to which gender may shape our results. Prior work suggests men tend to be more instrumental in their interactions with technology while women tend to be more relational (Venkatesh and Morris 2000, Venkatesh et al. 2000). This distinction raises an important question about how the findings might differ in the event of a greater representation of women in such environments. Of particular interest is whether and how the combination of instrumental (e.g., executing coding tasks) and relational (e.g., sharing knowledge, communicating with others) factors might impact their commitment to a particular OSS community. If prior research provides a guide, it is possible that, among women, the relational mechanisms studied here are likely leveraged to establish social connections to community as opposed to being driven by purely instrumental ends. Future research, especially work around women, may want to consider methods to reduce uncertainty that are less instrumental and more relationship focused. For paid OSS developers, there is also the important question of which entity dictates their commitment and contribution—the community values or the employer's values? We did not consider this question given our focus on all developers (paid and volunteer) in our sample. In addition, future research could usefully consider the percentage of developers in the community that are assigned to the community by a corporate organization and how that relates to the degree to which the community exhibits OSS values.

Several research opportunities exist related to considering other virtual communities. Given that, the mechanisms used to govern contributor efforts differ across various OSS communities (O'Mahony and Bechky 2008), exploring the model developed here in OSS communities that work using different governing bodies represents one such opportunity.

Further, future research should explore this model in communities that create products other than software.

Value congruence and incongruence reflect one expression of a person's fit with his environment. As an illustration, a developer may possess a complementary set of technical skills that the project needs. In this case the developer exhibits a complementary fit with the project. Exploring a developer's complementary fit with an OSS project provides a useful avenue for future research.

Each OSS community developer participates in many networks. These networks could be defined by participating in multiple OSS development efforts, conference attendance, telephone calls, emails or sequential contributions of source code. Each of these network connections could represent an alternative route through which a developer could learn about OSS values and what matters in an OSS community. This research considers both email communication and contributions to the same project. Future research should consider other potentially relevant networks in this context. In particular, observing the spillover of unique values from one OSS development effort to another may represent a useful avenue for future research. Finally, we explore value congruence and incongruence, and commitment at one point in time. Commitment often varies over time (Solinger et al. 2013). Thus exploring how commitment to an OSS community changes over time may offer deeper insight into activity patterns. For instance, Solinger et al. (2013) reports a scenario they name "Honeymoon Hangover," where commitment starts high and quickly drops off. In addition, observing value congruence and incongruence, and commitment at the same time limits the ability to infer causality. Now that we empirically demonstrated relationships between them, future research is necessary to identify the nature of causation.

CONCLUSION

OSS communities leverage resources from a globally distributed network of skilled volunteers and paid contributors. To understand the factors that motivate and enable developers to contribute to these communities requires an analysis of both developer and OSS community values, as well as the developers' structural network position within the community. The research presented here leverages uncertainty reduction theory to provide nuanced insights into how value congruence and incongruence impact commitment and contribution activity in an OSS community among developers who are well-connected within the communication network versus those who are less well-connected.

REFERENCES

Agerfalk, P. and Fitzgerald, B. 2008. "Outsourcing to an Unknown Workforce: Exploring Opensourcing as a Global Sourcing Strategy," *Management Information Systems Quarterly* (32:2), pp. 385-409.

Ahuja, M.K., Galletta, D.F. and Carley, K.M. 2003. "Individual centrality and performance in virtual R&D groups: An empirical study," *Management Science* (49:1), pp. 21-38.

Aiken, L.S. and West, S.G. 1991. "*Multiple regression: Testing and interpreting interactions,"* Newbury Park, CA: Sage Publications.

Aksulu, A. and Wade, M. 2010. "A comprehensive review and synthesis of open source research," *Journal of the Association for Information Systems* (11:11), pp. 576-656.

Allen, N.J. and Meyer, J.P. 1997. "*Commitment in the workplace: Theory, research and application,"* Thousand Oaks, CA: Sage Publications.

Antheunis, M.L., Valkenburg, P.M. and Peter, J. 2010. "Getting acquainted through social network sites: Testing a model of online uncertainty reduction and social attraction," *Computers in Human Behavior* (26:1), pp. 100-109.

Arthur, W.J., Bell, S.T., Villado, A.J. and Doverspike, D. 2006. "The use of person-organization fit in employment decision making: An assessment of its criterion-related validity," *Journal of Applied Psychology* (91:4), pp. 786-801.

Ashford, S.J., Lee, C. and Bobko, P. 1989. "Content, cause, and consequences of job insecurity: A theory-based measure and substantive test," *Academy of Management Journal* (32:4), pp. 803-829.

Bastien, D.T. 1987. "Common patterns of behavior and communication in corporate mergers and acquisitions," *Human Resource Management Journal* (26:1), pp. 17-33.

Bateman, P., Gray, P. and Butler, B. 2011. "The Impact of Community Commitment on Participation in Online Communities," *Information Systems Research* (22:4), pp. 841-854.

Becker, H.S. 1960. "Notes on the Concept of Commitment," *American Journal of Sociology* (66:1), pp. 32-40.

Benbya, H. and Belbaly, N. 2010. "Understanding developers' motives in open source projects: a multi-theoretical framework," *Communications of the AIS* (27:1), pp. 589-610.

Berger, C.R. and Calabrese, R.J. 1975. "Some explorations in initial interaction and beyond: Toward a developmental theory of interpersonal communication," *Human Communication Research* (1:2), pp. 99-112.

Bollen, K.A. and Jackman, R.W. 1990. "*Regression diagnostics: An expository treatment of outliers and influential cases,"* Newbury Park, CA: Sage Publications.

Bordia, P., Hunt, E., Paulsen, N., Tourish, D. and DiFonzo, N. 2004. "Uncertainty during organizational change: Is it all about control?," *European Journal of Work and Organizational Psychology* (13:3), pp. 345-365.

Borgatti, S.P. and Halgin, D.S. 2011. "On Network Theory," *Organization Science* (22:5), pp. 1168-1181.

Brashers, D.E. 2007. "*Communication and Uncertainty Management,"* Mahwah, New Jersey: Lawrence Erlbaum Associates, Inc.

Brierley, J. and Cowton, C. 2000. "Putting Meta-Analysis to Work: Accountants' Organizational-Professional Conflict," *Journal of Business Ethics* (24:4), pp. 343 - 353.

Brown, S.A., Venkatesh, V. and Goyal, S. 2012. "Expectation confirmation in technology use," *Information Systems Research* (23:2), pp. 474-487.

Brown, S.A., Venkatesh, V. and Goyal, S. 2014. "Expectation confirmation in IS research: A test of six competing models," *Management Information Systems Quarterly* (38:3), pp. 729-756.

Burt, R. 2004. "Structural holes and good ideas," *American Journal of Sociology* (110:2), pp. 349-399.

Byrne, D. 1997. "An overview (and underview) of research and theory within the attraction paradigm," *Journal of Social and Personal Relationships* (14:3), pp. 417-431.

Cable, D.M. and Edwards, J.R. 2004. "Complementary and supplementary fit: a theoretical and empirical integration," *Journal of Applied Psychology* (89:5), pp. 822-834.

Chatman, J. 1991. "Matching People and Organizations: Selection and Socialization in Public Accounting Firms," *Administrative Science Quarterly* (36:3), pp. 459-484.

Chengalur-Smith, I., Sidorova, A. and Daniel, S. 2010. "Sustainability of Free/Libre Open Source Projects: A Longitudinal Study," *Journal of the Association for Information Systems* (11:11), pp. 657-683.

Chou, S.-W. and He, M.-Y. 2011. "Understanding OSS development in communities: the perspectives of ideology and knowledge sharing," *Behaviour & Information Technology* (30:3), pp. 325-337.

Cohen, J., Cohen, P., West, S.G. and Aiken, L.S. 2003. "*Applied multiple regression/correlation analysis for the behavioral sciences,"* Mahwah, NJ: Erlbaum.

Coleman, J. 1988. "Social Capital in the Creation of Human Capital," *American Journal of Sociology* (94:Supplement), pp. 95-120.

Crowston, K., Annabi, H. and Howison, J. "Defining Open Source Software Project Success," 2003. *International Conference on Information Systems*, Seattle, WA.

Dabbish, L., Stuart, C., Tsay, J. and Herbsleb, J. "Social coding in github: transparency and collaboration in an open software repository," 2012. *Conference on Computer Supported Cooperative Work*, Seattle, Washington.

Dahlander, L. and O'Mahony, S. 2011. "Progressing to the Center: Coordinating Project Work," *Organization Science* (22:4), pp. 961-979.

Dahlander, L. and Wallin, M. 2006. "A man on the inside: Unlocking communities as complementary assets," *Research Policy* (35:8), pp. 1243–1259.

Dalal, D.K. and Zickar, M.J. 2012. "Some common myths about centering predictor variables in moderated multiple regression and polynomial regression," *Organizational Research Methods* (15:3), pp. 339-362.

Daniel, S., Agarwal, R. and Stewart, K.J. 2013. "The Effects of Diversity in Global, Distributed Collectives: A Study of Open Source Project Success," *Information Systems Research* (24:2), pp. 312-333.

Dishon-Berkovits, M. and Koslowski, M. 2002. "Determinants of Employee Punctuality," *The Journal of Social Psychology* (142:6), pp. 723-739.

Dwoskin, E. 2016. "*Tech Firms Grapple with How to Make Open Source Pay,"* City.

Edwards, J. 1993. "Problems with the use of profile similarity indicies in the study of congruence in organizational research," *Personnel Psychology* (46:3), pp. 641-665.

Edwards, J.R. 1996. "An examination of competing versions of the person–environment fit approach to stress," *Academy of Management Journal* (39:2), pp. 292–339.
Edwards, J.R. 2002. "*Alternatives to difference scores: Polynomial regression and response surface methodology,"* San Francisco, CA.
Edwards, J.R. and Lambert, L.S. 2007. "Methods for integrating moderation and mediation: a general analytical framework using moderated path analysis," *Psychological Methods* (12:1), pp. 1-22.
Edwards, J.R. and Parry, M.E. 1993. "On the use of polynomial regression equations as an alternative to difference scores in organizational research," *Academy of Management Journal* (36:6), pp. 1577-1613.
Edwards, J.R. and Rothbard, N.P. 1999. "Work and family stress and well-being: An examination of person–environment fit in the work and family domains," *Organizational Behavior and Human Decision Processes* (77:2), pp. 85–129.
Efron, B. and Tibshirani, R.J. 1993. "*An introduction to the bootstrap,"* New York.
Ehrlich, K. and Cataldo, M. "All-for-one and One-for-all? A Multilevel Analysis of Communication Patterns and Individual Performance in Geographically Distributed Software Development," 2012. *Conference on Computer Supported Cooperative Work*, Seattle, WA.
Elliott, M.S. and Scacchi, W. "Free Software Developers as an Occupational Community: Resolving Conflicts and Fostering Collaboration," 2003. *International ACM SIGGROUP Conference on Supporting Group Work*, Sanibel Island, Florida.
English, R. and Schweik, C. "Identifying Success and Tragedy of FLOSS Commons: A Preliminary Classification of Sourceforge.net Projects," 2007. *First International Workshop on FLOSS: Emerging Trends in FLOSS Research and Development*, Minneapolis, MN.
Fang, Y. and Neufeld, D. 2009. "Understanding Sustained Participation in Open Source Software Projects," *Journal of Management Information Systems* (25:4), pp. 9 - 50.
Faraj, S., Jarvenpaa, S.L. and Majchrzak, A. 2011. "Knowledge Collaboration in Online Communities," *Organization Science* (22:5), pp. 1224-1239.
Fisher, C.D. and Gitelson, R. 1983. "A meta-analysis of the correlates of role conflict and ambiguity," *Journal of Applied Psychology* (68:2), pp. 320.
Fornell, C. and Larcker, D. 1981. "Evaluating Structural Equation Models with Unobservable Variables and Measurement Error," *Journal of Marketing Research* (18:1), pp. 39-50.
Freeman, K.S. and Spyridakis, J.H. 2004. "An Examination of Factors That Affect the Credibility of Online Health Information," *Technical Communication* (51:2), pp. 239-263.
Freeman, L.C. 1978. "Centrality in social networks conceptual clarification," *Social Networks* (1:3), pp. 215-239.
Ghosh, R.A. 2002. "Free/Libre Open Source Software: Survey and Study," University.
Gibbs, J.L., Ellison, N.B. and Lai, C.-H. 2015. "First Comes Love, Then Comes Google: An Investigation of Uncertainty Reduction Strategies and Self-Disclosure in Online Dating," *Communication Research* (38:1), pp. 70-100.
Grewal, R., Lilien, G.L. and Mallapragada, G. 2006. "Location, Location, Location: How Network Embeddedness Affects Project Success in Open Source Systems," *Management Science* (52:7), pp. 1043-1056.
Hars, A. and Ou, S. 2002. "Working for Free? Motivations for Participating in Open Source Projects," *International Journal of Electronic Commerce* (6:3), pp. 25-39.
Heckman, J. 1979. "Sample Selection as a Specification Error," *Econometrica* (47:1), pp. 153–161.
Henkel, J. 2008. "Champions of revealing—the role of open source developers in commercial firms," *Industrial and Corporate Change* (18:3), pp. 435–471.
Hinds, P. and McGrath, C. "Structures that work: social structure, work structure and coordination ease in geographically distributed teams," 2006. *Conference on Computer Supported Cooperative Work*, Alberta, Canada.
Hoffman, B.J. and Woehr, D.J. 2006. "A quantitative review of the relationship between person–organization fit and behavioral outcomes," *Journal of Vocational Behavior* (68:3), pp. 389–399.

Hogg, M.A. 2000. "Subjective uncertainty reduction through self-categorization: A motivational theory of social identity processes," *European Review of Social Psychology* (11:1), pp. 223-255.

Hogg, M.A., Meehan, C. and Farquharson, J. 2010. "The solace of radicalism: Self-uncertainty and group identification in the face of threat," *Journal of Experimental Social Psychology* (46:6), pp. 1061-1066.

Hogg, M.A. and Terry, D.J. 2000. "Social Identity and Self-Categorization Processes in Organizational Contexts," *Academy of Management Review* (25:1), pp. 121-140.

Howison, J. and Crowston, K. 2014. "Collaboration Through Open Superposition," *Management Information Systems Quarterly* (38:1), pp. 29-50.

Hrebiniak, L. and Alutto, J. 1972. "Personal and Role-Related Factors in the Development of Organizational Commitment " *Administrative Science Quarterly* (17:4), pp. 555-573.

Jansen, K.J. and Kristof-Brown, A.L. 2005. "Marching to the beat of a different drummer: Examining the impact of pacing congruence," *Organizational Behavior and Human Decision Processes* (97:2), pp. 93-105.

Jensen, C., King, S. and Kuechler, V. "Joining Free/Open source software communities: An analysis of newbies' first interactions on project mailing lists," 2011. *44th Hawaii International Conference on System Sciences (HICSS)*, Kauai, HI.

Jensen, C. and Scacchi, W. "Collaboration, leadership, control, and conflict negotiation and the netbeans. org open source software development community," 2005. *38th Annual Hawaii International Conference In System Sciences*, Big Island, HI.

Johns, G. 2006. "The essential impact of context on organizational behavior," *Academy of Management Review* (31:2), pp. 396–408.

Ke, W. and Zhang, P. 2009. "Motivations in open source software communities: The mediating role of effort intensity and goal commitment," *International Journal of Electronic Commerce* (13:4), pp. 39-66.

Ke, W. and Zhang, P. 2010. "The Effects of Extrinsic Motivations and Satisfaction in Open Source Software Development," *Journal of the Association for Information Systems* (11:12), pp. 784-808.

Kidane, Y.H. and Gloor, P.A. 2007. "Correlating temporal communication patterns of the Eclipse open source community with performance and creativity," *Computational and Mathematical Organization Theory* (13:1), pp. 17-27.

Klein, G., Jiang, J.J. and Cheney, P. 2009. "Resolving difference score issues in information systems research," *Management Information Systems Quarterly* (33:4), pp. 811-826.

Kluckhohn, C. 1951. "*Values and value-orientations in the theory of action: An exploration in definition and classification,"* Cambridge, MA: Harvard University Press.

Knobloch, L.K. 2015. "Uncertainty Reduction Theory," *International Encyclopedia of Interpersonal Communication* pp. 1-9.

Knobloch, L.K., Satterlee, K.L. and DiDomenico, S.M. 2010. "Relational Uncertainty Predicting Appraisals of Face Threat in Courtship: Integrating Uncertainty Reduction Theory and Politeness Theory," *Communication Research* (37:3), pp. 303-334.

Kogut, B. and Metiu, A. 2001. "Open source software development and distributed innovation," *Oxford Review of Economic Policy* (17:2), pp. 248-264.

Kramer, M.W. 1993. "Communication and uncertainty reduction during job transfers: Leaving and joining processes," *Communications Monographs* (60:2), pp. 178-198.

Kristof-Brown, A., Zimmerman, R. and Johnson, E. 2005. "Consequences of individuals' fit at work: A meta-analysis of person-job, person-organization, person-group and person-supervisor fit," *Personnel Psychology* (58:2), pp. 281-342.

Kristof-Brown, A.L. and Guay, R.P. 2011. "*Person-Environment Fit,"* Washington D.C.: American Psychological Association.

Kuan, J. "OSS as Lead User's Make or By Decision: A Study of Open and Closed Source Quality," 2002. *Open Source Software : Economics, Law and Policy*, Toulouse, France.

Kuechler, V., Gilbertson, C. and Jensen, C. "Gender differences in early free and open source software joining process," 2012. *IFIP International Conference on Open Source Systems*, Berlin Heidelberg.

Lakhani, K. and Wolf, R. 2005. "*Why Hackers Do What They Do: Understanding Effort and Motivation in Open Source Projects,"* Cambridge, Mass: MIT Press.

Lee, G. and Cole, R. 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.

Lee, J. and Kim, S. 2011. "Exploring the role of social networks in affective organizational commitment: Network centrality, strength of ties, and structural holes," *The American Review of Public Administration* (41:2), pp. 205-223.

Lee, T.W., Mitchell, T.R., Sablynski, C.J., Burton, J.P. and Holtom, B.C. 2004. "The Effects of Job Embeddedness on Organizational Citizenship, Job Performance, Volitional Absences, and Voluntary Turnover," *Academy of Management Journal* (47:5), pp. 711-722.

Lindberg, A., Berente, N., Gaskin, J. and Lyytinen, K. 2016. "Coordinating interdependencies in online communities: A study of an open source software project," *Information Systems Research* pp.

Ljungberg, J. 2000. "Open source movements as a model for organizing," *European Journal of Information Systems* (9:4), pp. 208-216.

Lydon, J. 1996. "*Toward a theory of commitment,"* Hillsdale, NJ: Lawrence Erlbaum Associates.

MacKinnon, D.P., Lockwood, C.M. and Williams, J. 2004. "Confidence limits for the indirect effect: Distribution of the product and resampling methods," *Multivariate Behavioral Research* (39:1), pp. 99-128.

Mallapragada, G., Grewal, R. and Lilien, G. 2012. "User Generated Open Source Products: Founder's Social Capital and Time to Market," *Marketing Science* (31:3), pp. 474-492.

Mathieu, J.E. and Zajac, D.M. 1990. "A review and meta-analysis of the antecedents, correlates, and consequences of organizational commitment," *Psychological Bulletin* (108:2), pp. 171-194.

Meglino, B.M., Ravlin, E.C. and Adkins, C.L. 1989. "A work values approach to corporate culture: A field test of the value congruence process and its relationship to individual outcomes," *Journal of Applied Psychology* (74:3), pp. 424-432.

Meyer, J.P. and Allen, N.J. 1991. "A three-component conceptualization of organizational commitment," *Human Resource Management Review* (1:1), pp. 61-89.

Meyer, J.P. and Herscovitch, L. 2001. "Commitment in the workplace: Toward a general model," *Human Resource Management Review* (11:3), pp. 299-326.

Meyer, J.P., Paunonen, S.V., Gellatly, I.R., Goffin, R.D. and Jackson, D.N. 1989. "Organizational commitment and job performance: It's the nature of the commitment that counts," *Journal of Applied Psychology* (74:1), pp. 152-156.

Midha, V. and Bhattacherjee, A. 2012. "Governance Practices and Software Maintenance: A study of open source projects," *Decision Support Systems* (54:1), pp. 23-32.

Miller, K.I. and Monge, P.R. 1985. "Social information and employee anxiety about organizational change," *Human Communication Research* (11:3), pp. 365-386.

Mockus, A., Fielding, R.T. and Herbsleb, J.D. 2002. "Two Case Studies of Open Source Software Development: Apache and Mozilla," *ACM Transactions on Software Engineering and Methodology* (11:3), pp. 309-346.

Nafus, D. 2012. "Patches don't have gender': What is not open in open source software," *New Media & Society* (14:4), pp. 669-683.

Nonnecke, B. and Preece, J. "Lurker demographics: Counting the silent.," 2000. *Proceedings of CHI 2000 Conference*, The Hague, Netherlands.

O'Mahony, S. and Bechky, B.A. 2008. "Boundary organizations: Enabling collaboration among unexpected allies," *Administrative Science Quarterly* (53:2), pp. 422-459.

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

Orlikowski, W.J. and Yates, J. 1994. "Genre Repertoire: The Structuring of Communicative Practices in Organizations," *Administrative Science Quarterly* (39:541), pp. 541-574.

Paulson, J., Succi, G. and Eberlein, A. 2004. "An Empirical Study of Open-Source and Closed-Source Software Products," *IEEE Transactions on Software Engineering* (30:4), pp. 246-256.

Pham, R., Singer, L. and Schneider, K. "Building test suites in social coding sites by leveraging drive-by commits," 2013. *International Conference on Software Engineering*, San Francisco, CA.

Pinto, G., Steinmacher, I. and Aur, M. "More Common Than You Think: An In-Depth Study of Casual Contributors," 2016. *IEEE 23rd International Conference on Software Analysis, Evolution, and Reengineering (SANER)*, Osaka, Japan.

Podolny, J.M. 2001. "Networks as the Pipes and Prisms of the Market," *American journal of sociology* (107:1), pp. 33-60.

Pollack, J.M., Vanepps, E.M. and Hayes, A.F. 2012. "The moderating role of social ties on entrepreneurs' depressed affect and withdrawal intentions in response to economic stress," *Journal of Organizational Behavior* (33:6), pp. 789-810.

Preacher, K.J. and Hayes, A.F. 2004. "SPSS and SAS procedures for estimating indirect effects in simple mediation models," *Behavior Research Methods* (36:4), pp. 717-731.

Preacher, K.J., Rucker, D.D. and Hayes, A.F. 2007. "Addressing moderated mediation hypotheses: Theory, methods, and prescriptions," *Multivariate Behavioral Research* (42:1), pp. 185-227.

Qureshi, I. and Fang, Y. 2010. "Socialization in Open Source Software Projects: A Growth Mixture Modeling Approach," *Organizational Research Methods* (14:1), pp. 208-238.

Ramirez, A., Walther, J.B., Burgoon, J.K. and Sunnafrank, M. 2002. "Information-Seeking Strategies, Uncertainty, and Computer-Mediated Communication: Toward a Conceptual Model," *Human Communication Research* (28: pp. 213-228.

Raymond, E.S. 2001. "*The cathedral and the Bazaar: Musing on Linux and Open Source by an Accidental Revolutionary,"* Sabastopol, CA: O'Reilly.

Rice, R.E. and Aydin, C. 1991. "Attitudes toward new organizational technology: Network proximity as a mechanism for social information processing," *Administrative Science Quarterly* (36:2), pp. 219-244.

Roberts, J.A., Hann, I.-H. 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. 984-1000.

Robles, G. and Gonzalez-Barahona, J. 2006. "*Contributor Turnover in Libre Software Projects,"* Boston: Springer.

Robles, G., Reina, L.A., González-Barahona, J.M. and Domínguez, S.D. "Women in Free/Libre/Open Source Software: The Situation in the 2010s," 2016. *IFIP International Conference on Open Source Systems*,

Rokeach, M. 1973. "*The nature of human values,"* New York: Free Press.

Sarker, S., Ahuja, M., Sarker, S. and Kirkeby, S. 2011. "The Role of Communication and Trust in Global Virtual Teams: A Social Network Perspective," *Journal of Management Information Systems* (28:1), pp. 273 - 310.

Schein, E.H. 1985. "*Organisational culture and leadership: A dynamic view,"* San Francisco: Jossey-Bass.

Schneider, B. 1987. "E= f (P, B): The road to a radical approach to person-environment fit," *Journal of Vocational Behavior* (31:3), pp. 353-361.

Schweiger, D.M. and Denisi, A.S. 1991. "Communication with employees following a merger: A longitudinal field experiment," *Academy of Management Journal* (34:1), pp. 110-135.

Seidel, M.-D. and Stewart, K.J. 2011. "An Initial Description of the C-Form," *Research in the Sociology of Organizations* (33:November), pp. 37 - 72.

Setia, P., Rajagopalan, B., Sambamurthy, V. and Calantone, R. 2012. "How Peripheral Developers Contribute to Open-Source Software Development," *Information Systems Research* (23:1), pp. 144-163.

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

Siciliano, M.D. and Thompson, J.R. 2015. "If You Are Committed, Then so Am I The Role of Social Networks and Social Influence on Organizational Commitment," *Administration & Society* pp. 1-31.

Singh, P., Tan, Y. and Mookerjee, V. 2011. "Network Effects: The Influence of Structural Social Capital on Open Source Project Success," *Management Information Systems Quarterly* (35:4), pp. 813-829.

Smith, M.A. 1993. "*Voices from the WELL: The logic of the virtual commons,"* City: University of California, Los Angeles.

Solinger, O.N., Olffen, W.v., Roe, R.A. and Hofmans, J. 2013. "On Becoming (Un) Committed: A Taxonomy and Test of Newcomer Onboarding Scenarios," *Organization Science* (24:6), pp. 1640-1661.

Soltis, S. 2012. "Who you are and who you know: The infuence of person envrionment fit and social network centrality on individual performance," Management. University of Kentucky.

Spaeth, S., von Krogh, G. and He, F. 2015. "Research Note—Perceived Firm Attributes and Intrinsic Motivation in Sponsored Open Source Software Projects," *Information Systems Research* (26:1), pp. 224-237.

Sparrowe, R., Liden, R.C., Wayne, S.J. and Kraimer, M.L. 2001. "Social Networks and the Performance of Individuals and Groups," *Academy of Management Journal* (44:2), pp. 316-325.

Stallman. 1992. "Why Software should be free," Retrieved March 1, 2003, University.

Steers, R.M. 1977. "Antecedents and outcomes of organizational commitment," *Administrative science quarterly* (22:1), pp. 46-56.

Steinmacher, I., Conte, T., Gerosa, M.A. and Redmiles, D. "Social barriers faced by newcomers placing their first contribution in open source software projects*,"* 2015. *Proceedings of the 18th ACM conference on Computer Supported Cooperative Work & Social Computing*, Vancouver, BC, Canada.

Steinmacher, I., Conte, T.U., Treude, C. and Gerosa, M.A. "Overcoming open source project entry barriers with a portal for newcomers*,"* 2016. *38th International Conference on Software Engineering*, Austin, TX.

Stewart, K., 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. 126-145.

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

Stine, R. 1989. "An Introduction to Bootstrap Methods Examples and Ideas," *Sociological Methods and Research* (18:2), pp. 243-291.

Trice, H.M. and Beyer, J.M. 1993. "*The Cultures of Work Organizations,"* Englewood Cliffs, NJ: Prentice Hall.

Tushman, M.L. 1979. "Impacts of perceived environmental variability on patterns of work related communication," *Academy of Management Journal* (22:3), pp. 482-500.

Tushman, M.L. 1979. "Work characteristics and subunit communication structure: A contingency analysis," *Administrative Science Quarterly* pp. 82-98.

Vardaman, J.M., Taylor, S.G., Allen, D.G., Gondo, M.B. and Amis, J.M. 2015. "Translating intentions to behavior: The interaction of network structure and behavioral intentions in understanding employee turnover," *Organization Science* (26:4), pp. 1177-1191.

Vogel, R., Rodell, J.B. and Lynch, J. 2015. "Engaged and productive misfits: How job crafting and leisure activity mitigate the negative effects of value incongruence," *Academy of Management Journal* pp. NA.

Von Krogh, G., Haefliger, S., Spaeth, S. and Wallin, M.W. 2012. "Carrots and Rainbows: Motivation and Social Practice in Open Source Software Development," *MIS Quarterly* (36:2), pp. 649-676.

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.

Wagstrom, P. 2009. "Vertical Communication in Open Software Engineering Communities," Carnegie Institute of Technology. Carnegie Mellon University. Pittsburgh. CMU-ISR-09-103.

Weinberg, G. 1998. "*The Psychology of Computer Programming (Silver Anniversary Edition),"* New York: Van Nostrand Reinhold.

Williams, K. 2007. "Ostracism," *Annual Review of Psychology* (58:1), pp. 425-452.

Xu, B., Jones, D.R. and Shao, B. 2009. "Volunteers' involvement in online community based software development," *Information and Management* (46:3), pp. 151-158.

Zhang, C., Hahn, J. and De, P. 2013. "Continued Participation in Online Innovation Communities: Does Community Response Matter Equally for Everyone?," *Information Systems Research* (24:4), pp. 1112-1130.

Zhou, M. and Mockus, A. "What make long term contributors: Willingness and opportunity in oss community," 2012. *International Conference on Software Engineering*, Zurich, Switzerland.

Zhou, M. and Mockus, A. 2015. "Who Will Stay in the FLOSS Community? Modelling Participant's Initial Behaviour," *IEEE Transactions on Software Engineering* (41:1), pp. 82-99.

MATERIALS FOR ONLINE APPENDIX

APPENDIX A: Summary of Empirical Research on Impact of OSS Values on Developers

A growing body of empirical research has examined the role of OSS values in affecting developers (Benbya and Belbaly 2010, Chou and He 2011, Stewart and Gosain 2006). The research to date has typically focused on either the community (or OSS team) OSS values or the developer OSS values. To the best of our knowledge, only two studies consider both. In their seminal study, Stewart and Gosain (2006) find that when OSS teams embrace OSS values, there is a positive influence on communication quality and affective trust in OSS teams. Surprisingly, they also find that embracing OSS values negatively influences task completion. They explain their findings by suggesting that teams who embrace OSS values that are geared towards collaboration tend to prioritize consensus over completing tasks. Chou and He (2011) find that a team embracing OSS values positively impacts its collaborative elaboration, and communication decoding and encoding competence. Taken together, this research suggests that the community (or team) OSS values can influence team attitudes and activity level. A looming open issue in this research is that researchers do not consider the role of the individual developer OSS values. As such, it is possible that a developer may not necessarily support the OSS values in the same way as the community.

Other researchers focus on the individual developer OSS values as a determinant. For instance, researchers find that a developer who embraces such values is likely to expend a greater amount of time and effort contributing to OSS initiatives (Benbya and Belbaly 2010) and is likely to report being more involved in OSS communities (Xu et al. 2009). In contrast to these findings, Henkel (2008) finds that OSS values play no role in affecting developers' contribution to OSS communities. In, sum, this stream of research suggests that the developer OSS values can influence developer contribution to OSS communities. However, this research overlooks the role of the OSS community's values in affecting developer attitudes and contribution behavior. As such, it provides an incomplete picture of the role of OSS values in shaping developers' attitudes and behavior in OSS communities.

Table 1. Summary of Empirical Research Findings on OSS Values

Article \ Referent	Focus of OSS Values		Key Finding
	Developer	OSS Community	
Xu et al. 2009	√		+ impact on developer involvement
Henkel 2008	√		n.s. developer contribution
Benbya and Belbaly 2010	√		+ impact on developer participation type + impact on developer effort
Ke and Zhang 2009	√	√	- developer goal commitment n.s. developer effort intensity
Ke and Zhang 2010	√	√	- developer task effort
Chow and He 2011		√	+ collaborative elaboration + communication decoding competence + communication encoding competence
Stewart and Gosain 2006		√	+ impact on communication quality + impact on affective trust - task completion
+ denotes values antecedent has positive impact on listed outcome, - denotes values antecedent has negative impact on listed outcome, n.s. denotes values antecedent has non-significant impact on listed outcome			

APPENDIX B: Results of Confirmatory Factor Analysis

	Factors		
Items	1	2	3
As a software developer…			
Developer values1: I value sharing knowledge.	***.32***	.01	.24
Developer values2: I believe in helping others.	**.76**	.17	.06
Developer values3: I place great value on technical knowledge.	**.70**	.23	.13
Developer values4: I am driven by a desire to learn new things.	**.70**	.03	.05
Developer values5: I think cooperation is important.	**.75**	.08	.11
Developer values6: I value the reputation I gain by participating in open source.	**.67**	.17	.09
In my view, members of this OSS community…			
Perception of OSS values1: value sharing knowledge.	.13	***.24***	.28
Perception of OSS values2: believe in helping others.	.20	**.82**	.22
Perception of OSS values3: place great value on technical knowledge.	.13	**.83**	.17
Perception of OSS values4: are driven by a desire to learn new things.	.17	**.69**	.22
Perception of OSS values5: think cooperation is important.	.18	**.73**	.18
Perception of OSS values6: value the reputation gained by participating in open	.08	**.78**	.23
Commitment1: I am willing to put in effort beyond the norm for the success of GNOME.	.15	.18	**.79**
Commitment2: For me, this is the best of all possible OSS projects for which to work.	.10	.21	**.85**
Commitment3: I am extremely glad to have chosen GNOME to work for over other projects.	.11	.23	**.86**
Commitment4: GNOME inspires me to be my best technical work.	.09	.30	**.79**
Commitment5: I show by my actions that I really care about the fate of GNOME.	.10	.30	**.77**

Note: Items with italicized loadings were dropped

APPENDIX C: Coefficients for Response Surface Analysis Predicting OSS Commitment at Low versus High Developer Centrality

Variable	Coefficient	Low developer centrality	High developer centrality
(Intercept)	b_0	4.98***	5.06***
Developer values	b_1	.13	.17*
OSS values	b_2	.60***	.52***
Developer values-squared	b_3	.13	.13
Developer values x OSS values	b_4	-.11*	-.07
OSS values-squared	b_5	.17	.25**

Notes: Significance levels are based on bias-corrected confidence intervals generated from bootstrap estimates.
p < .05, ** p < .01, *** p < .001.

APPENDIX D: Results of Moderation-Mediation Analysis

		Developer contribution activity		
	OSS commitment	Number of commits	Lines of code added/deleted	Number of files changed
(Intercept)	5.42***	-4.99***	-4.28***	-4.35***
Age	-.01	-.02	-.03	-.02
Gender	.35	2.42*	3.76*	2.33*
Education	-.14*	-.24	-.57	-.36
Volunteer status	-.08	-1.04*	-1.42*	-.94*
OSS Value (in)congruence	.69***	.41	.67	.41
Developer centrality	.46*			
OSS commitment		.71***	.90**	.58**
Value congruence x developer centrality	-.52*			
R^2	.46***	.30***	.29***	.28***

Notes: n = 410

1. Value congruence is computed from block variable using the predicted value from the polynomial regression equations.
2. Pattern of results is similar when using different measures of developer centrality.

* p < .05, ** p < .01, *** p < .001.

APPENDIX E: Results of Polynomial Regression Analysis Using Alternative Measures of Developer Centrality

Variable	Coefficient	1	2 Centrality: 12-month email	3 Centrality: 24-month email	4 Centrality: 12-month project	5 Centrality: 24-month project
Age		-.04	-.04	-.03	-.02	.00
Gender		.03	.03	.04	.03	.02
Education		-.10*	-.10*	-.10*	-.10*	-.10*
Volunteer status		-.05	-.05	-.06	-.03	-.06
Developer values	b_1	.16*	.15*	.18*	.19**	.20**
OSS values	b_2	.50***	.56***	.53***	.41***	.43***
Developer values-squared	b_3	.13*	.13*	.14*	.15*	.15*
Developer values x OSS values	b_4	-.14*	-.09†	-.10*	-.14*	-.15*
OSS values-squared	b_5	.15*	.21**	.11*	.10*	.13*
Developer centrality	b_6		.15**	.18*	.30***	.31***
Developer values x developer centrality	b_7		.07	.08	.06	.07
OSS values x developer centrality	b_8		-.16**	-.21**	-.12*	-.19*
Developer values-squared x developer centrality	b_9		.00	.03	.06	.03
Developer values x OSS values x developer centrality	b_{10}		.07	.13	.01	-.01
OSS values-squared x developer centrality	b_{11}		.16**	.14*	.10*	.16*
Adjusted R^2		.30***	.33***	.33***	.34***	.36***
ΔR^2			.03**	.03**	.04***	.06***

Notes: n = 410.

1. Developer centrality in model 2 is measured on email communication activity over prior 12 months, developer centrality in model 3 is measured on email communication activity over prior 24 months, developer centrality in model 4 is measured on project collaboration activity over prior 12 months, developer centrality in model 5 is measured on project collaboration activity over prior 24 months.
2. ΔR^2 for models 2 through 5 represent change in variance explained over and above model 1 (i.e., the polynomial regression model without moderation).

† $p < .10$, * $p < .05$, ** $p < .01$, *** $p < .001$.

APPENDIX F: Results of Instrumental Variables Two-Stage Least Squares Analysis with Robust Standard Errors

		OSS commitment		
Variable		1	2	3
Age		.00 (.00)	.00 (.01)	.00 (.00)
Gender		.06 (.18)	.03 (.17)	.12 (.19)
Education		-.10* (.05)	-.10† (.06)	-.14* (.06)
Volunteer status		.09 (.12)	.07 (.12)	.07 (.13)
Developer values	b_1	1.71* (.73)	2.15* (1.00)	2.40* (1.05)
OSS community values	b_2	2.36*** (.25)	2.62*** (.45)	2.06*** (.51)
Developer values-squared	b_3		8.91** (2.66)	5.94* (2.93)
Developer values x OSS community values	b_4		-9.09* (3.83)	-7.60† (4.02)
OSS community values-squared	b_5		.86** (.28)	1.21* (.60)
Developer centrality	b_6			2.59** (.91)
Developer values x developer centrality	b_7			.06 (.08)
OSS community values x developer centrality	b_8			-1.41** (.52)
Developer values-squared x developer centrality	b_9			.00 (.05)
Developer values x OSS community values x developer centrality	b_{10}			4.49 (3.66)
OSS community values-squared x developer centrality	b_{11}			2.49** (1.05)
Wald χ^2		150.20***	212.67***	220.292***
Adjusted R^2		.28	.30	.32

Notes: n = 410; standard errors are in parentheses; number of commits made (pre-survey), total number of messages posted to listserv (pre-survey), total number of projects (pre-survey), and number of replies posted to listserv (pre-survey) were used as instruments.

† $p < .10$, * $p < .05$, ** $p < .01$, *** $p < .001$.

APPENDIX G: Results of Mediation (H3) and Moderated-Mediation (H4) Tests on Number of Lines of Code Added and Deleted and Number of Files Changed

Level of developer centrality	Direct effect	Indirect effect	95% bias-corrected CI		R^2	Dependent variable
			Lower bound	Upper bound		
	.68 (.46)	.39 (.20)	.04	.86	.23***	Number of lines of code added/deleted
High	.68 (.49)	.50 (.31)	-.01	.97	.29***	
Low		.75 (.27)	.29	.99		
	.46 (.30)	.33 (.14)	.07	.64	.19***	Number of files changed
High	.41 (.32)	.32 (.20)	-.01	.62	.28***	
Low		.48 (.18)	.18	.89		

Notes:

1. Value congruence and incongruence block variable is the independent variable, developer commitment is the mediator, developer centrality is the moderator; control variables were included as covariates in the analysis.
2. Standard errors are in parentheses.
3. Dependent variables were log-transformed prior to analysis.
4. The pattern of results is the same when using degree centrality based on email communication in past 24 months, collaboration activity on projects in past 12 months and past 24 months.

*** $p < .001$.

APPENDIX H: Results of Model Estimation Using Heckman Two-Step Selection Procedure

Variable		Commitment
Age		.00 (.01)
Gender		.16 (.26)
Education		-.14* (.06)
Volunteer status		.02 (.10)
Developer values	b_1	2.29* (1.01)
OSS community values	b_2	2.60*** (.40)
Developer values-squared	b_3	6.99* (3.41)
Developer values x OSS community values	b_4	-7.45† (3.93)
OSS community values-squared	b_5	1.62* (.60)
Developer centrality	b_6	.28*** (.06)
Developer values x developer centrality	b_7	.93 (.66)
OSS community values x developer centrality	b_8	-.99** (.45)
Developer values-squared x developer centrality	b_9	.01 (.05)
Developer values x OSS community values x developer centrality	b_{10}	3.97 (3.61)
OSS community values-squared x developer centrality	b_{11}	.78* (.35)
Inverse Mills ratio		.10 (.52)
Wald χ^2		234.98***

Notes: n = 410; standard errors are in parentheses; developer degree centrality in the communication network (pre-survey), number of commits (pre-survey), number of different projects (pre-survey), and number of messages and replies posted to the listserv (pre-survey) were used as determinants in the estimation of the first stage probit model.

† p < .10, * p < .05, ** p < .01, *** p < .001