DETERMINANTS OF SUCCESS OF THE OPEN SOURCE SELECTIVE REVEALING STRATEGY: SOLUTION KNOWLEDGE EMERGENCE

MEKKI MACAULAY ABDELWAHAB

A DISSERTATION SUBMITTED TO
THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

GRADUATE PROGRAM IN ADMINISTRATION
YORK UNIVERSITY
TORONTO, ONTARIO

June 2018

ABSTRACT

Recent research suggests that firms may be able to create a competitive advantage by deliberately revealing specific problem knowledge beyond firm boundaries to open source meta-organisations such that new solution knowledge is created that benefits the focal firm more than its competitors (Alexy, George, & Salter, 2013). Yet, not all firms that use knowledge revealing strategies are successful in inducing the emergence of solution knowledge. The extant literature has as of yet not explained this heterogeneity in success of knowledge revealing strategies. Using a longitudinal database spanning the period from 1998 to end 2012 with more than 2 billion data points that was obtained from the Mozilla Foundation, one of the top open source meta-organisations, this dissertation identifies and measures the antecedent factors affecting successful solution knowledge emergence. The results reveal 35 antecedent factors that affect solution knowledge emergence in different ways across three levels of analysis. The numerous contributions to theory and practice that follow from the results are discussed.

DEDICATION

This dissertation is dedicated to those people with power who use that power to support those who do not have power, enabling them to achieve their potential and more.

ACKNOWLEDGEMENTS

This research would not have been possible without the energy and endless support of Ellen Auster. I am tremendously grateful for her tireless advice, her dedication to seeing students through this challenging process, and her boundless positivity.

My thanks goes also to Christine Oliver for believing in me, helping me understand the contextual history of this research, and challenging me to enter and contribute to some of the oldest strategic management conversations. With her guidance, I am privileged to have been able to stand upon the shoulders of giants, as the saying goes.

I am thankful to Yuval Deutsch for always being available for advice, in-depth discussion, and help refining ideas. With his guidance and suggestions, the methodological rigor and relevance of this research were greatly improved.

Thank you to Eileen Fischer for being blunt when I needed bluntness, for understanding and being patient with my engineer's mind, and for aiming me towards long term academic success.

Thank you to Jean-Baptiste Litrico for championing my academic path and putting our research collaborations on hold while this dissertation took center stage.

I am grateful to my colleagues in the PhD program at the Schulich School of Business. Your experience, insights, diverse backgrounds, mentorship, friendship, and emotional support improved both my quality of life and the quality of my research during my time in the program.

Thank you to the support staff at the Schulich School of Business, including, in particular, Stephanie Allen, Clara Kan, JoAnne Stein, Teresa Back, and Tammy Tam, for helping me cross all my t's and dot all my i's.

Thank you to the Mozilla Foundation for providing access to the Mozilla Bugzilla database for this research and for the time of your employees and volunteers who helped me understand the many complexities in the data and transform them into a form suitable for the analysis of questions relevant to strategic management.

This research was supported in part by a Community Investment Program grant from the Canadian Internet Registration Authority (CIRA). Thank you to CIRA for its support.

This research was supported in part by a grant from the Ontario Ministry of Training, Colleges and Universities. Thank you for supporting innovative Canadian research.

Finally, thank you to my family for the ongoing support and for understanding my distraction and focus. Thank you to Marcus Saad and Andressa Geburt for watching over me during the rollercoaster of this process. Thank you to Alex Mckenzie for keeping me sane during technical problems. Thank you to Yasmine Benatti for keeping me organized. Thank you to Kedim, my PhD cat, and Pluto, Dora, and Nanners for warm snuggles whenever I was stressed. And, thank you to Sarah Zandbergen for always being there for me, no matter what I needed.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER ONE: INTRODUCTION

Summary

For nearly two decades, many Fortune 500 companies have invested financial and human capital in the development of knowledge resources that are freely available to all, including their competitors (O'Mahony & Ferraro, 2007). They collectively participate in knowledge production in meta-organisations (Gulati, Puranam, & Tushman, 2012), such as the Mozilla Foundation, which is best known for the development of the popular Firefox web browser. Until recently, this open source strategy (MacAulay, 2013) had defied conventional strategic management analysis as it appeared, at first glance, to undermine the competitive advantage of participating firms. Alexy, George, and Salter (2013) propose that if a firm deliberately reveals certain knowledge outside its boundaries, as is the case when engaging in knowledge production activities in meta-organisations, useful complementary knowledge might emerge, knowledge that could not otherwise be efficiently created by the firm itself exclusively within its boundaries (Goldman & Gabriel, 2005). Yet, not all revelations of knowledge outside firm boundaries result in the emergence of useful knowledge. To date, no study has explained this heterogeneity in the success of knowledge revealing strategies. This dissertation addresses this gap in the literature.

Grounded in the Knowledge-Based View of the firm (KBV) this dissertation investigates the antecedent factors that determine the successful emergence of useful complementary knowledge in response to deliberate reveals of knowledge by participants in meta-organisations. This research topic is important because it builds upon the emerging literature stream (c.f. Yang, Phelps, & Steensma, 2010; Agarwal, Anand, Bercovitz, & Croson, 2012; Alexy, George, & Salter, 2013) that challenges the conventional notion (Peteraf, 1993; Grant, 1996a, Dyer &

Nobeoka, 2000) that the knowledge spillovers necessarily undermine competitive advantage by identifying the factors that connect deliberate knowledge reveals to the emergence of useful knowledge. While the emergent literature has suggested that firms may develop open source strategy based competitive advantages by more efficiently getting complementary knowledge returns for their knowledge revelations than other firms (Alexy, George, & Salter, 2013; MacAulay, 2013), this dissertation fills a gap in the present literature by examining the specific antecedent factors that connect knowledge revelation to the emergence of useful complementary knowledge, the specifics of which have not previously been considered in management research. The results of this dissertation provide a potential mechanism for explaining how firms might attain a competitive advantage using an open source strategy.

More specifically, this research analyzes longitudinal data spanning the period from 1998 to end of 2012 covering all the knowledge production activities of the Mozilla Foundation meta-organisation during that period to identify the factors that promote knowledge emergence. Mozilla is one of the largest and most active open source meta-organisations with a high number of participating firms. Its database of the interactions between participating actors engaging in collective knowledge production includes some of the largest Fortune 500 companies (O'Mahony & Ferraro, 2007). These data exhaustively document all of the revealed and emergent knowledge since the inception of the Mozilla meta-organisation, for all of its projects, including the popular Firefox web browser. To my knowledge, this extensive longitudinal data source that I negotiated access to with the Mozilla Foundation for the purpose of this study, which contains more than 2 billion data points, has not previously been analyzed in strategic management research.

Driving question

This dissertation aims to refine the model developed by Alexy et al. (2013), which describes the factors leading up to the decision to use a knowledge revealing strategy and the resulting competitive advantages derived from the emergent useful knowledge, by connecting these two end-points with the answer to the research question, "*What are the factors driving successful solution knowledge emergence?*" Given that not all firms that use knowledge revealing strategies succeed in getting a return of useful knowledge, the answer to this research question is necessary before the research stream on open source strategies can begin to test the theory that competitive advantage can result from the use of such strategies (Alexy, et al., 2013).

Importance of the topic

This section highlights the contributions of the dissertation thesis and the importance of the research on this topic to gaps in the open source literature and two streams of literature in strategy: the KBV and organisational forms of production.

Open source

The open source literature has long examined the motivations of firms that use open source strategies (Lerner & Tirole, 2002; Benkler, 2002; Bonaccorsi & Rossi, 2003; Grand, von Krogh, Leonard, & Swap, 2004; Demil & Lecocq, 2006; Chesbrough & Appleyard, 2007; Dahlander & Frederiksen, 2012). However, the factors that affect the emergence of useful knowledge for the focal firm once it chooses to undertake an open source strategy are not well understood (See Autio, Dahlander, & Frederiksen, 2013, for a recent exception that examines the entrepreneurial firm context). The open source literature focuses on issues of relevance to technology and innovation management, such as measures of innovative output, leaving questions of central relevance to strategic management, such as which factors should firms focus on when seeking to achieve competitive advantage through participation in open source meta-organisations, unaddressed. This dissertation bridges that gap by examining the nature and extent of factors that determine when knowledge revelation translates into knowledge emergence for focal actors. It further triangulates these factors across multiple levels of analysis, yielding an explanation for the disparities reported in the open source literature by authors of studies that did not split levels of analysis.

Strategy – The knowledge-based view of the firm

The traditional KBV perspective holds that firms exist because they are an organisational form that is more effective and efficient at the generation and application of knowledge resources than alternative organisational forms such as markets or networks (Grant, 1996a). Whereas from this perspective, knowledge resources must be carefully protected to avoid loss of competitive advantage due to imitation (Peteraf, 1993; Dyer & Nobeoka, 2000; McEvily & Chakravarthy, 2002), from more recent KBV refinements, arguments have emerged suggesting that knowledge spillovers may be beneficial to firms in some cases (Yang, Phelps, & Steensma, 2010; Agarwal, Anand, Bercovitz, & Croson, 2012; Alexy, George, & Salter, 2013). These refinements recognize that knowledge resources are different from traditional physical resources and that "the economics of scarce resources does not hold in the digital age where inventories do not deplete by application of the [knowledge] to the production of a finished product." (Kanevsky & Housel, 1998: 269). While these studies have offered important insights on motivations to engage in knowledge-revealing strategies, little is known about the factors that determine successful outcomes. The KBV literature describes a broad range of factors associated with knowledge resources and knowledge production processes its actors, yet these factors have typically been considered within firms who protect their internal knowledge resources and seek to develop them internally (c.f. Dewar & Dutton, 1986; Damanpour, 1996; Kessler & Chakrabarti, 1996; Sanchez & Mahoney, 1996; Matusik & Hill, 1998; Matusik, 2002; Matusik & Heeley, 2005; Wang, He, & Mahoney, 2009; Crossan & Apaydin, 2010) or from the perspective of the balance between exploration and exploitation of knowledge (c.f. March, 1991; Benner & Tushman, 2003; Jansen, Van Den Bosch, & Volberda, 2006; Miller, Zhao, & Calantone, 2006; Uotila, Maula, Keil, & Zahra, 2009; Hoang & Rothaermel, 2010). Yet, given that meta-organisations are a novel form of organisational production (Gulati, Puranam, & Tushman, 2012) and given that knowledge-revealing strategies are deliberate releases of knowledge rather than accidental spillovers (Alexy, George, & Salter, 2013), neither of these research streams fit the context of the phenomenon under investigation. This dissertation contributes to the KBV literature by empirically examining factors affecting knowledge resources and knowledge production in a novel context.

Further, the multilevel nature of the many factors in the KBV literature theorized to affect knowledge production processes has resulted in a lack of clarity on the locus of effects in the many, inconsistent empirical examinations, resulting in a call for more research that specifically considers both individual level factors and the relationship between individual and organisation level factors affecting knowledge production efforts (Pisano, 1994; von Hippel, 1994; Volberda, Foss, & Lyles, 2010). Both of these issues are considered in this study.

Strategy – Transaction cost economics and organisational forms of production

Organisational forms of production have been a central topic in strategic management research since the emergence of the field as distinct from its economics and sociological roots (cf. Williamson, 1975). From the perspective of transaction cost economics, open source meta-organisations are a distinct form of production, being neither markets nor hierarchies nor networks (Demil & Lecocq, 2006). They have low stratification and low barriers to entry (Gulati, Puranam & Tushman, 2012), allowing many firms to participate in the knowledge production effort to varying degrees. The traditional research on organisational forms of production assumes that open source strategies are similar to strategies used in the bottled water business: bundle a free product (water) with something else, like a brand, guarantee, or service agreement (Hecker, 1999; West & Dedrick, 2001; Lerner & Tirole, 2002). The value produced is assumed to relate purely to the non-free product in the bundle. Yet, over the past 20 years, open source strategies have evolved from simple value-added packaging to a distinct mode of production of knowledge resources. While the literature examines the distinct dimensions of different open source organisational forms (Chesbrough & Appleyard, 2007; O'Mahony & Bechky, 2008; Lakhani, Lifshitz-Assaf, & Tushman, 2012), little is known about the antecedent factors that lead to effective participation in such organisational forms, although it is clear that failure to pay attention to these factors may lead to organisations failing to derive benefit from their use of open source strategies (Anand, Glick, & Manz, 2002; West & Wood, 2014). While the organisational forms literature has a long history of factors that affect the success of production strategies in traditional forms such as markets, hierarchies, and networks, it is not clear the degree to which these factors apply to meta-organisational forms of production.

The purpose of this dissertation is to address these gaps in the open source literature, the strategy literature, and the literature on organisation forms of production, informing both research and practice.

Limitations and key assumptions

In order to promote the tractability and manageability of the dissertation, several key assumptions are made that result in necessary limitations to the generalisability and scope of interpretation of the research's results.

First, the Mozilla Foundation is assumed to be a representative open source meta-organisation. While this focus limits the generalizability of the findings, as different open source meta-organisations may have different antecedent factors for successful knowledge emergence, the choice is justified by the breadth of the organisation and the availability of the data. Thus, the focus on the single meta-organisation ensured that the dissertation remained manageable and primes future research that compares the findings across different meta-organisations.

Second, the participation of organisations in the Mozilla meta-organisation is assumed to be a deliberate strategic action, or "profit-oriented behavior…which implies that the focal actor does not reveal out of principle but rather as a result of weighing the commercial pros and cons" (Henkel, Schöberl, & Alexy, 2014: 880). As a result of this assumption there is an inherent limitation in separating organisational actors from other actors who may not be using deliberate management strategies in their engagement in the knowledge production process. To address this limitation, careful operationalisation procedures are used to compare conservatively refined samples of the data that are more readily attributable to organisational action and intent.

Third, this dissertation deliberately focuses exclusively on the value creation side of knowledge revealing strategies, making the assumption that these outcomes are valuable. However, there is undoubtedly a cost to participation in meta-organisations that is beyond the scope of this research. The resulting limitation is that the net value for firms is not measured in the present study. This limitation is reasonable because value creation and costs are typically considered separately in the strategy literature and the examination of value creation typically precedes the measuring of costs.

Fourth, this dissertation deliberately constrains its analytical focus to the data contained within the Mozilla Foundation's Bugzilla database, spanning from 1998 to end 2012 for the purpose of identifying antecedent factors that affect solution knowledge emergence. However, other data sources exist that may content relevant factors that affect outcomes, limiting the range of factors that are considered and measured. This limitation is justified because past research has suggested that matching up these disparate data sources cohesively is prohibitively difficult (Ayari, Meshkinfam, Antoniol, & Di Penta, 2007) and the focus on a single longitudinal data source promotes the manageability and the tractability of the research.

Contributions to the literature

This dissertation contributes to the KBV by providing the first specific model that identifies the antecedent factors linking knowledge revealing strategies and the emergence of novel knowledge that is of use to the initial knowledge revealer, extending the work of Alexy, George, and Salter (2013) and answering the call for such research in non-traditional organisational forms by Foss, Husted, and Michailova (2010).

This dissertation also contributes to the KBV by improving the extant understanding of the factors involved in knowledge resources that affect the knowledge production process in organisations, including absorptive capacity, codifiability, knowledge paradigms, knowledge flow processes, knowledge stakeholders, and knowledge value, by providing novel empirical evidence of the involvement of these factors in knowledge production in meta organisations. It also contributes by considering multiple levels of analysis of knowledge production, allowing the identification of experience effects involved in the knowledge creation processes, informing the debate on the influence of second order knowledge resource on lower order knowledge resource emergence (Dierickx & Cool, 1989; Arend, 2006; Bingham & Eisenhardt, 2011). Whereas previous research has struggled to identify higher order effects due to the data sources and research designs of most management research (Priem & Butler, 2001), this research uses a deliberate methodological design and accompanying analytical procedures to enable the localisation of the level of contributing effects of the identified factors on outcomes of interest.

Third, this research contributes to the KBV literature by providing novel methods and operationalizations for investigating traditional knowledge constructs. These approaches promote the replicability of the study and offer novel means of investigating other questions that are pertinent to the KBV literature.

Fourth, with the rise of meta-organisations and interest in the strategic management literature (See *Strategic Management Journal* special issue on organisational forms edited by Gulati, Puranam, & Tushman, 2012) the dimensions of different forms of meta-organisations are starting to be examined (Chesbrough & Appleyard, 2007; O'Mahony & Bechky, 2008; Lakhani, Lifshitz-Assaf, & Tushman, 2012). Yet, little is known about how firms can leverage meta-organisations to their advantage. This dissertation contributes to organisational form research by considering the success factors of participation in the specific subset of open source meta-organisations, which are described as low stratification and low boundary meta-organisations in the taxonomy of Gulati, Puranam, and Tushman. (2012). It provides an improved understanding of the factors organisations must consider after extending the knowledge-based boundaries of the firm, answering the call for such research by Bogers, Afuah, and Bastian (2010). It further provides the groundwork for future research on the success factors in meta-organisations that have different stratification and boundary characteristics or different dimensions of organisational production and participation factors altogether.

This dissertation contributes to strategy theory by bridging the open source, KBV, and organisational forms of production literatures in a manner that informs conversations in each literature stream. It extends our understanding of the factors that affect the outcomes of open source strategies as forms of production distinct from traditional markets, hierarchies, or networks (Demil & Lecocq, 2006). It also develops methods for analyzing databases that have previously only been considered in computer science from a strategic management lens, enabling future strategy research to tap into these rich data.

Finally, this research contributes to strategic management practice by offering firms a better understanding of factors on which to focus when in open source meta-organisations. It adds depth and breadth to the extant guidance on managing knowledge-sharing relationships outside firm boundaries, particularly with competitors, developed by von Hippel and von Krogh (2003), with more recent empirical evidence that reflects the ongoing evolution of open source meta-organisations (West & Wood, 2014). The factors identified in this research provide guidance on resource allocation for firms wishing to use knowledge revealing open source strategies to maximize their return on investment.

CHAPTER TWO: LITERATURE REVIEW

Meta-organisations as non-traditional forms of production

Research on organisational forms of production has been prominent in strategic management since its inception as a field that is distinct from its economics and sociology roots. Transaction cost economics (TCE) emerged in the late 1970s as one of the distinguished theories of strategy. Building on the economic roots of industrial organisation and equilibrium theorizing, it jumped into prominence because it showed that firm profitability could result from the economizing of transaction costs of varying forms of production alone and did not necessarily need to result from collusion, monopolies, or other strategies that were thought to damage social welfare (Williamson, 1975). One of Williamson's (1975) core arguments is that forms of production (which he called governance forms of economic activity) necessarily need to be comparative in nature. He argues that analyzing a single form of production (such as the traditional hierarchical firm), in isolation, provided no context for absolute assessments. Rather, he proposes that the task of strategic managers is to compare the different available forms of production and select the one that is best suited to a given strategic situation. The TCE's original formulation only compared firms and markets as forms of production. Later, Williamson (1991) added "hybrids" as intermediate forms that he describes as between the extremes of firms and markets on the same dimension. More recent research (Makadok & Coff, 2009) suggests that markets and hierarchies don't lie on opposite ends of a single dimension, but rather that there are three key dimensions: strength of incentives, strength of authority, and nature of ownership. Firms have low incentive strength, high authority, and high ownership and markets have high incentives, low authority and low ownership. Using these extended dimensions, it became clear that there are likely many more distinct forms of organisational production than had been traditionally considered.

Demil and Lecocq (2006) were the first to recognize that open source organisations are distinct forms of production. They attributed the distinction to the novel form of contract used to govern open source production arrangements, called “copyleft” agreements. They built on Williamson’s (1985) work where he argues that governance forms of production could be explained by their institutional context, by which he meant the legal structure in which they operate.

Traditional markets, he claims, are governed by classical (sales) contracts, that are well defined and absolute. Hybrids depend on neo-classical contracts, which do not attempt to foresee all possible outcomes (as the costs would be too high), and rather are flexible, but with intended goals and rewards for outcomes. Firms, by contrast, he argues, are governed by the legal principal of forbearance, where the courts would refuse to get involved in intra-firm disagreements, leading to them being resolved by fiat and other internal mechanisms. As Demil and Lecocq (2006) point out, open source production efforts use a novel legal contract mechanism that is distinct from all three forms described by Williamson. It takes the neo-classical contract as a starting point, but rather than attempting to describe at least some of the contingencies, it reverses the assumptions of the legal system of property rights. Traditional contracts take the assumption that parties may do absolutely nothing except what is explicitly permitted in the contract (license). Copyleft contracts take the exact opposite position. They guarantee that parties to the contract may do absolutely anything they wish except that which is prohibited by the license. They focus on ensuring that all parties have rights that cannot be restricted in the future, rather than granting temporary rights that may default back to the control of the firm in the future. It is this form of copyleft contract that enables the novel production method of the firm as distinct from markets, hierarchies and hybrids.

As interest grew, strategy researchers began to consider non-traditional organisational forms of production in more depth. Gulati, Puranam, and Tushman (2012) edited a special issue of *Strategic Management Journal* dedicated to fleshing out issues of non-traditional organisations, which they aggregated into the concept of "meta-organisational forms". Meta-organisational forms "comprise networks of firms or individuals not bound by authority based on employment relationships" (Gulati, Puranam, & Tushman, 2012:573). While they use the term "networks" in their definition of meta-organisations, the connotations used in the special issue and subsequent work are related to diverse interconnections and interactions, in the manner formulated by Demil and Lecocq (2006), not the formal networks described in the network theory literature (e.g. Powell, 1990). Gulati, Puranam, and Tushman suggested that meta-organisations can be classified according to two major factors: the degree of stratification and the nature of the boundaries to membership in the meta-organisation. The resulting 2 X 2 taxonomy, adapted from Gulati, Puranam, and Tushman. (2012) appears in Table 1.

	Low-stratification/hierarchy	**High-stratification/hierarchy**
Closed boundaries / membership	Consortia; standards committees	Franchising; supplier networks; extended enterprise
Open boundaries / membership	Open source organisations	Managed ecosystems; open innovation; contests

Table 1: Taxonomy of meta-organisations (Adapted from Gulati, Puranam, &Tushman, 2012)

The form of production that is open source organisations, such as the Mozilla Foundation, is distinct from other forms of meta-organisations in that it has open boundaries (anyone can become a member) and participation is peer-based, with low-stratification and an absence of hierarchical controls. Participants self-select to tasks and (largely) laterally review and support one another. Firms that participate in open source meta-organisations cannot directly exclude external participation such as by competing firms. They must exert strategic control in other ways. By contrast, standards communities also have low-stratification, but membership is tightly controlled with conditions for entry, participation, and consequences for exit. On the right side of Table 1, the level of stratification contrasts open source strategy from open innovation (terms that are frequently confounded). In the latter, firms exert a form of hierarchical control over the way the innovative effort or problem solving is done, to ensure that they keep control on the direction and outcomes.

Other authors have documented other dimensions of open source organisational forms as distinct from other forms of production. O'Mahony and Bechky (2008) added four additional dimensions: how the open source organisation is initiated (by a firm or by individuals); who owns the intellectual property that results from the production effort (the firm or those who contribute); who has the right to use the created resources after production is done (just the focal firm or everyone, including competitors); and, the nature of decision making (controlled by a firm or collective decision making through community governance mechanisms). While the nature of decision making factor overlaps slightly with degree of stratification discussed earlier, it extends the former concept by including factors such as design direction, conflict resolution, feature inclusion, quality metrics, and related factors that can have distinct approaches.

Lakhani, Lifshitz-Assaf, and Tushman (2012) added two additional dimensions for classifying non-traditional forms of production: the degree of task decomposability (high/modular or low/integrated) and the distribution of necessary problem solving knowledge (high/broad or low/narrow). They argued that in open source meta-organisations the ability to decompose the task that the production effort is targeting is moderately high to high and the available problem solving knowledge is broad. Open source meta-organisations as forms of production are hence best suited to fairly modular problems that do not require tight integration to solve (which pure firm hierarchies might be better suited to), and where the knowledge required to solve the problems is broadly distributed "out there" and may not require in-depth specialization (which pure markets might be able to solve better).

Finally, Chesbrough and Appleyard (2007) added two more dimensions: the locus of value creation (within a focal firm vs. in the broader open source community) and the appropriator of the majority of the produced value (a single firm, or the open source community as a whole). These two dimensions are particularly salient to open source organisations as distinct forms as the locus of value creation is typically in the community and the appropriator of the majority of the produced value is typically the open source community as a whole, outcomes that are contrary to many forms of production. As a result, firms need to carefully manage their participation in open source meta-organisations in order to leverage the valuable resources that are created therein.

In summary, the taxonomy of meta-organisations is described by at least 10 factors, with open source meta-organisations featuring prominently as a distinct form from traditional forms in the strategic management literature. One downside of some of the factors in the literature is that they assume that value production and capture is a zero-sum game. Some authors have begun to question this assumption (cf. Etzkowitz, 1997; Lado, Boyd, & Hanlon, 1997; Fey & Birkinshaw, 2005). Rather, open source meta-organisations may be a prominent example of the creation of valuable resources that can be simultaneously appropriated by multiple competing parties. This notion is well-matched to the KBV literature because a fundamental distinction of knowledge resources, as opposed to more traditional, physical resources, is that knowledge resources are not consumed when they are used.

Knowledge-based view

The knowledge-based view of the firm emerged as a distinct stream from the traditional resource-based view of the firm (RBV) (Wernerfelt, 1984; Barney, 1986, 1991; Dierickx & Cool, 1989; Peteraf, 1993; Peteraf & Barney, 2003) when researchers began to identify properties of knowledge resources that were distinct from other types of resources. In particular, knowledge resources are intangible resources that are not consumed when they are used (Grant, 1996a). Knowledge resources are also developed and improved by using them, as firms can learn from using their knowledge resources, inverting the traditional perspective that the use of resources leads to their depletion and suggesting that simply using knowledge resources periodically can prevent their depreciation over time, whereas traditional resources require directed effort to develop and replenish (Dierickx & Cool, 1989). This difference can be attributed to firm experience effects, which some authors suggest are the central distinction of the KBV in strategy (Kogut & Zander, 1992, 1996; Eisenhardt & Santos, 2002).

The KBV literature explored the factors that affect knowledge resources and firm learning, arguing that knowledge may be the most important resource to attain and sustain competitive advantage and superior performance (Grant; 1996; Winter & Szulanski, 2001). A key factor is the degree of tacitness of the knowledge, or the degree to which it is embedded in individuals and organisations, through learning and experience, and cannot be readily transmitted (Nonaka, 1994). Tacit knowledge is "rooted in action, procedures, routines, commitment, ideals, values, and emotions" (Nonaka & von Krogh, 2009: 636). The opposite of tacitness is explicitness, which is the property of knowledge that has been transformed into an artifact such as speech, text, depictions, or demonstration through a process known as codification (Zander & Kogut, 1995). Explicit knowledge can readily be transmitted from one person to another. Knowledge complexity is another factor (Zander & Kogut, 1995) that moderates the codifiability of tacit knowledge. Given the complex relationships between these factors, some researchers have used more aggregate constructs to describe properties of knowledge such as ambiguity (Szulanski, 1996). Further, at the individual level, the learner of a given knowledge resource must have sufficient absorptive capacity (Szulanski, 1996) to make effective use of it. At the firm level, firms must be sufficiently flexible and have appropriate organisational values in place to permit the uptake of useful knowledge (Leonard-Barton, 1992). Despite the range of factors, a consistent focus in the literature has been on how these factors affect the usefulness of the knowledge to the focal firm. Six of these factors are discussed in more detail in the next section and tied into the present research.

Some proponents of the KBV propose that, beyond novel properties of resources, it may represent a novel theory of the firm. From this perspective, the core strategy question about the nature of the firm is answered by the proposition that firms exist as a form of production because they are better able to create and apply knowledge resources and manage the transmission and retention of knowledge than other forms of production (Grant, 1996; Nonaka, Toyama, &

Nagata, 2000). Other researchers contend that to inform the theory of the firm, knowledge is better conceptualized as a process rather than a resource, as processes better describe the observed firm learning (Spender, 1996). This latter perspective builds on the KBV as a theory of the firm by arguing that the nature of firm boundaries is determined in part by knowledge flows, rather than traditional legal boundaries. The strategic interactions that take place between firms (Kuk, 2006) in meta-organisations have reopened old strategic management debates about the boundaries of the firm (c.f. Coase, 1937; Williamson, 1975) by suggesting that knowledge-based firm boundaries may be porous and mobile. In particular, by participating in open source meta-organisations to produce knowledge assets, firms are making a choice to extend their knowledge-based boundaries, which can lead to knowledge spillovers to competitors.

The traditional perspective on knowledge spillovers (Dyer & Nobeoka, 2000) is that they undermine one of the cornerstones of competitive advantage by allowing imitation of the knowledge resource (Peteraf, 1993). Recent research highlights the benefits of extending firm knowledge boundaries in terms of improvement of product development quality (Matusik, 2002) and improvement in a firm's ability to internally transmit knowledge (Kogut & Zander, 1992). It may also give a firm access to knowledge that it might not have been able to create on its own (Goldman & Gabriel, 2005). In short, the assessment of extending firm knowledge boundaries and participating in meta-organisations must consider more than the imitation by competitors that might result from knowledge spillovers. It must also weigh the value that firms accrue from the participation (Casadesus-Masanell & Llanes, 2011). Some authors have even begun to question if the assumption of knowledge spillovers as necessarily bad for competitive advantage holds in all situations (Yang, Phelps, & Steensma, 2010; Alexy, George, & Salter, 2013).

Yang, Phelps, & Steensma (2010) demonstrated that when knowledge spills over a firm boundary, which they argue may be unavoidable in certain contexts (such as when participating in open source meta-organisations), the knowledge gets recombined with spillover knowledge from other firms, creating a "spillover knowledge pool" that the focal firm can draw from. Firms benefit from this novel knowledge pool because it contains complementary knowledge that was previously within other firms' boundaries and not accessible to the focal firm. They suggest that knowledge spillovers should be reconceptualised as potentially valuable learning opportunities for firms.

Alexy, George, and Salter (2013) develop this argument further, suggesting that firms can benefit from knowledge spillovers by turning them into deliberate strategies that they call "selective-revealing strategies" (271). They argue that selective-revealing strategies are particularly effective when traditional forms of collaborative production are not suitable due to high partner uncertainty, high coordination costs, or when potential collaborators are concerned about unequal value acquisition (285). This argument matches well with the factors that identify open source meta-organisations as a distinct form of production in cases where traditional forms are not suitable, relating selective-revealing strategies to the choice to participate in meta-organisations. Alexy, George, and Salter (2013) also argue that firms can gain a competitive advantage through the use of selective-revealing strategies in two different ways. First, by revealing knowledge that relates to a problem that the firm cannot resolve on its own, knowledge that provides a solution to the problem may emerge from the meta-organisation. A firm that is better at revealing "problem knowledge" (Afuah & Tucci, 2012; Alexy, George, & Salter, 2013) in a manner that confirms to the institutional and social norms of specific meta-organisations may be able to extract more frequent and/or more valuable solution knowledge through participation than other firms, giving the focal firm a competitive advantage based on efficiency and effectiveness of its open source meta-organisation participation. Second, by selectively revealing knowledge to a meta-organisation a firm can reshape both the deliberate and the passive collaborative behaviours of other participants. It can shape the path-dependency of the future activities of other participating firms, making their future spillovers more valuable to the focal firm. The result is a subtle form of competitive manipulation and an exercise of power called "induced isomorphism", which they define as "deliberate strategic action to induce other [participants] to become more similar to the focal firm, particularly with respect to the production of knowledge" (272). The focal firm can gain a competitive advantage by binding other firms to specific technologies that the focal firm developed, or more generally, reshaping the content and structural compatibility of the knowledge in the meta-organisation such that it favours the focal firm over competitors by making it more complementary to the focal firm's proprietary assets. Selective-revealing strategies may even allow less traditionally powerful firms to exert influence on powerful firms over time by slowly binding them to a path that favours the focal firm. Further, open source knowledge production processes that frequently reuse software code for efficiency may be particularly conducive to deliberately inducing the adoption of knowledge by competitors (Haefliger, von Krogh, & Spaeth, 2008).

These properties of knowledge-revealing strategies lay the foundation for the theoretical framework of this dissertation. Specifically, this dissertation addresses the gap relating to the factors that influence successful knowledge emergence for firms that participate in open source meta-organisation, creating a bridge of antecedent factors between the decision to engage in a knowledge-revealing strategy and the emergence of valuable knowledge.

Open source

Nearly every Fortune 500 company depends on open source software to run its business and the impact of open source production on organisations worldwide is in the hundreds of billions of dollars range (MERIT, 2006). At the time of writing, there are over 800,000 registered open source projects, each with distinct characteristics, communities, norms, needs, and types of participants (Sourceforge, 2013). Most projects start small with just one or a few contributors. Over time, in some cases over a decade or more, they can grow into large meta-organisations such as the Mozilla Foundation. It is no surprise then that open source meta-organisations have been a source of curiosity for strategy scholars as, at first glance, it would seem that they are driven primarily by the altruistic intentions of individuals who volunteer their time to develop a collective good without expectation of direct financial compensation. The reality is far more complex.

Researchers have been investigating the phenomenon of open source for more than a decade. Raymond's (1999a) classic book *The Cathedral and the Bazaar* explained open source as an alternative software development method to the proprietary methods used by large corporations such as Microsoft. At that time, most major software projects were developed by a few select firms, behind closed doors, to a specification that was a tightly held secret. It was a long and laborious but clean process that Raymond equated to the construction of a cathedral. Open source, by contrast, was described as chaotic, like a bazaar, where there are many different participants, each with different skills, goals, and ways of participating. It is a fast-paced environment where software gets released quickly and often, regardless of how many defects a given version of the software might have. It promotes an incremental improvement model as opposed to a do-it-in-one-shot model. Raymond proposed that one of the major advantages of open source as a form of production was that "given enough eyeballs, all bugs are shallow" (1999b:29), by which he meant that it is easier to identify and fix defects when you have a large number of people working on a product than when you have a small number of people. When Raymond formulated this principle, he focused on the programmer perspective and the measured outcome of bug identification and resolution, an approach that is adapted to strategy research in this dissertation. More recently, it has been recognized that the open source form of production's distinction, with a diversity of perspectives, skills, and approaches, ultimately leads to a better product that addresses a broader array of individual and organisational needs (MacAulay, 2010ab). Mature projects already have solid code bases that will no longer see large improvements from the contributions of programmers alone. In such cases, Raymond's mantra may need to be updated to "With enough eyes, all open source project issues, technical and non-technical, are shallow", implying that skills other than programming are important for continued participation. From a strategic perspective, learning these diverse skills may promote favorable outcomes.

Much research explores the motivations of both individuals and firms to participate in open source. Lerner and Tirole (2002) first described what seemed like a crazy scenario of individuals and for-profit firms working on creating a valuable resource in order to freely share it with the public, including with competing firms. They explained that individuals participate in open source meta-organisations because it may help them address challenges they encounter in their jobs, as is the case when a systems administrator helps resolve a persistent problem that is common to his office environment and that of other organisations. Individuals also participate in order to develop a reputation in the community, to improve their career prospects through a portfolio of contributions and signaling effects, to entice participation in their personal projects, and because they identify as members of a community (Hertel, Niedner, & Herrmann, 2003; Bagozzi & Dholakia, 2006). These incentives can often be stronger than hierarchical work incentives such as salaried employment or market incentives such as contract work (Lerner & Tirole, 2002).

Firm participation in open source was thought to be even stranger given the presumed negative effects on competitive advantage. Yet, a recent attempt to measure firm participation by venture capital firm North Bridge with a survey of more than 1000 firms in 65 countries suggested that more than 65% of them are using knowledge revealing strategies in open source meta-organisations (North Bridge, 2016). Firms may use a variety of means for interacting with and attempting to influence the open source meta-organisation depending on their goals and their abilities to effectively learn about the relevant success factors (Dahlander & Magnusson, 2005). Firms sometimes create their own sponsored open source meta-organisations in an effort to balance control and growth surrounding an open source project (West & O'Mahony, 2008). At a first glance, such efforts were thought to be fruitless for the firm itself, while everyone else, including its competitors, could leverage the result. Yet a closer look has shown that firms that engage in open source production are not actually producing a purely collective good, but rather are using a "private-collective" (von Hippel & von Krogh, 2003) form of production that yields firm benefits in a range of ways. These benefits include learning and knowledge development (Lakhani & von Hippel, 2003), transaction cost reduction (Foss & Foss, 2005), access to resources that the firm might not otherwise be able to leverage (Dahlander & Magnusson, 2005; Goldman & Gabriel, 2005), promoting faster adoption of products and standards (Bonaccorsi & Rossi, 2006), shifting the locus of value in the competitive ecosystem away from the strengths of competitors (Chesbrough & Appleyard, 2007), increasing the sales of complementary assets

(West & Gallagher, 2006), and inducing isomorphism in competitors (Alexy, George, and Salter, 2013).

More recently, researchers are conceptualizing participation in open source meta-organisations as a deliberate strategic action. Such open source strategies, which are strategies that are built around and dependent upon a system of production that brings together participants from both within and outside the firm to produce a valuable good that remains readily available to all (Lakhani, 2012; Levine & Prietula, 2012; von Hippel, 2005), are a specific form of knowledge-revealing strategy (Alexy, George, and Salter, 2013) in the context of open source meta-organisations. As such, I argue that open source strategies are distinct from traditional strategies in that they focus on non-traditional forms of production, do not assume that value creation and capture is a zero-sum game (matching the emergent KBV perspective), and relax the assumption that knowledge-spillovers beyond firm boundaries (or the extension of the knowledge-based boundaries of the firm) are necessarily bad for competitive advantage. Instead, open source strategies leverage experience effects to improve the effectiveness and efficiency of the extraction of knowledge that is more useful to the focal firm than to competitors from open source meta-organisations. This dissertation examines the factors that affect the success of knowledge-revealing strategies. In particular, it focuses on the factors that improve the efficiency and effectiveness of solution knowledge emergence subsequent to the revelation of knowledge by a focal actor to a meta-organisation.

Success factors of knowledge-revealing strategies in open source meta-organisations

The foundation linking the KBV and open source literatures has been building for the past decade. It is becoming increasingly clear that "opens-source contribution structures for the production of [knowledge] resources increase the opportunities for… knowledge exchange" (Powell, 2012: 692). There have been studies with formal economic modeling of the performance of open source meta-organisations (Levine & Prietula, 2013), optimal business model design (Belenzon & Schankerman, 2015), limitations to firm size and diversification (Colombo, Piva, & Rossi-Lamastra, 2014), organisational structure design to efficiently utilize knowledge from outside the firm (Foss, Husted, & Michailova, 2010; Foss, Lyngsie, & Zahra, 2013), organisational practices for effective engagement with open source meta-organisations (Salter, Criscuolo, & Ter Wal, 2014) management of partnerships (Du, Leten, & Vanhavenbeke, 2014), power dynamics between meta-organisational participants engaging in knowledge creation (Gambardella & Panico, 2014), sources of knowledge and concerns about competitive imitation (Giarratana & Mariani, 2014), the commercial pros and cons of the use of knowledge revealing strategies (Henkel, Schöberl, & Alexy, 2014), and, of particular relevance to this dissertation, problem solving strategies (Felin & Zenger, 2014).

In their special issue of *Research Policy,* looking forward towards the next decade of research, West, Salter, Vanhavenbeke, and Chesbrough (2014) called for research "linking [open source strategy] research to the management and economics literature" as well as "better measurement" (805). To those ends, a comprehensive examination of the factors theorized to affect success in open source meta-organisations is compiled from the open source literature and then mapped to the corresponding themes in the KBV literature, linking the two, and identifying the factors that are operationalized for empirical measured in this research.

Success factors in open source literature

A comprehensive search of the open source literature revealed a large number of factors that are associated with open source meta-organisations. These factors can be organized into two streams: ways of measuring success, and antecedents to success.

Measuring success in open source meta-organisations

The primary measure of success in most open source meta-organisations is whether or not solution knowledge, often termed “fix”, emerges subsequent to the reveal of problem knowledge, often termed “bug”, to the meta-organisation (Anvik, Hiew, & Murphy, 2006; Antoniol, Ayari, Khomh, & Guéhéneuc, 2008; Baysal, Kononenko, Holmes, & Godfrey, 2013). The roots of this measure lie in the software development history of many open source meta-organisations (Raymond, 1999), where the goal was to identify defects in software code, known as “bugs”, and to “fix” them by updating the software code with a solution to the problem causing the “bug”. Since those early days, bug reporting systems, like Bugzilla (Serrano & Ciordia, 2005), have adapted to serve not only for tracking of software defects, but also to track entire strategic planning and community collaboration and the allocation of knowledge production effort in meta-organisations (Reagle Jr., 2007; Rahman, Ruhe, & Zimmermann, 2009; Lanubile, Ebert, Prikladnicki, & Bizcaino, 2010; Gheorghe, 2012; Hosseini, Nguyen, & Godfrey, 2012; Pereira, Gonçalves, von Wangenheim, & Buglione, 2013). From a KBV perspective, the “fixing” of problems represents the emergence of knowledge that organisations sought by engaging in the knowledge-revealing strategy by revealing the problem knowledge to the meta-organisation (Alexy, George, & Salter, 2013). It can be thought of the “return” on the “investment” of extending the knowledge-based boundaries of the firm.

A second complementary measure considers the circumstance when a fix to a problem emerges with an accompanied “patch”, which is typically a piece of software code that was produced to address the problem and a part or the whole of the solution knowledge that emerges as a result of the production effort. Given that not all solution knowledge emerges in the form of software code, the situation of “fix with patch” is handled as a separate success factor and measured independently of problems that are fixed without patches (Antoniol, et al., 2008).

A third set of measures considers timing factors related to the knowledge production process. Ceteris paribus, faster completely of knowledge creation is better for focal actors. Three time-related success factors are considered: the overall resolution time, independent of the actual resolution as “fixed” or “not fixed” (Huntley, 2003; Dalle, et al., 2008; Ahmed & Gokhale, 2009; Au et al., 2009); the time between the submission of new problem knowledge to the meta-organisation and its assignment to a solution knowledge producer who will work on creating the associated solution knowledge (Baysal, et al., 2013); and, the time it takes for a solution to be developed, which is the difference between resolution time and assignment time, often referred to as development time, following the software development lingo in use in many open source meta-organisations (Sharma, Sugumaran, & Rajagopalan, 2002; Fitzgerald, 2004; Haefliger, von Krogh, & Spaeth, 2008; Colazo & Fang, 2009).

A fourth set of measures considers the directness of the knowledge production process (Koponen, 2006; Wang & Zhang, 2012). As a complementary measure to timing-related factors, the directness with which a problem proceeds through the knowledge production process is a desirable factor. Loops in the process are characterised by “reopening” of problems, which happens when a solution emerges that does not address the problem it is purported to resolve

(Guo, et al., 2010), and "reassigning" of problems, which happens when a problem is assigned to a developer who is unable or unwilling to produce the required solution, resulting in a new developer being identified instead (Guo, et al., 2010; Guo, et al., 2011), both of which are considered negative success factors. The lack of reopening and reassigning implies a directness in the knowledge production process that provides an alternate measure to time that isn't biased based on the size of the problem. An associated measure, "confirmation", represents the state in the knowledge production process whereby a problem has been investigated by a knowledgeable actor in the meta-organisation and validated as suitable to proceed to solution knowledge production. As such, confirmation is generally considered positive and a desired success factor that is also independent of problem size (Panjer, 2007). As discussed in more detail in Chapter Four: Research Method, these success factors can be operationalized across three levels of analysis, each one resulting in a distinct outcome measurement.

Antecedent factors for success in open source literature

The open source literature is rife with factors purported to affect the success measures described in the previous section. A comprehensive review of the literature identified more than 150 ways to measure more than 50 different factors spanning several levels of analysis. As discussed in more detail in Chapter Four: Research Method, at the outset, 86 antecedent factors were operationalised across three levels of analysis. The core factors described in the literature used to create those measurements are as follows.

The number of other open problems is a factor often reported in the literature as affecting the subsequent production of knowledge in the meta-organisation. It is suggested that the number of unresolved problems draws attention away from novel problems, representing load on the meta-organisation's production effort. It has been suggested that the absolute number of open bugs is not the best factor, but rather the number of open bugs in similar knowledge categories as a focal new bug is sometimes reported as the most salient antecedent success factor (Anvik, Hiew, & Murphy, 2006). Other studies have suggested that the number of other open bugs submitted recently, or bugs resolved recently, from a time-frame rather than knowledge-similarity perspective, are the salient factors (Hooimeijer & Weimer, 2007; Shihab, et al., 2010).

With respect to time, the timing of submission of new problem knowledge has been extensively investigated in the literature (Fershtman & Gandal, 2004; Francalanci & Merlo, 2008) suggesting that social factors such as day of week, day of month, month, or year, or meta-organisational cycle factors, such as proximity to release schedules, may affect success factors. The amount of time a problem has remained "open" and no solution knowledge emerges may also be an antecedent factor for the likelihood of solution knowledge emergence (Giger, Pinzger & Gall, 2010).

With respect to the type of the problem knowledge submitted to the meta-organisation, the open source literature has considered the sufficiency of the information contained therein in fields such as "description" (Ahmed & Gokhale, 2009; Bettenburg, et al., 2008; Guo, et al., 2010; Guo, et al., 2011), the content and clarity of that information for different stakeholders (Canfora & Cerulo, 2006; WeiB et al., 2007; Chilana, Ko, & Wobbrock, 2010), and the redundancy of the information relative to information already in the meta-organisation (Sandusky, Gasser, & Ripoche, 2004; Zimmermann, et al., 2010). It has also considered the sufficiency and content of emergent information subsequent to the problem knowledge submission but prior to solution knowledge production such as “comments” (WeiB et al., 2007; Shihab, et al, 2010; Zhang, et al., 2012).

The extant literature has also assessed the ways meta-organisations categorize knowledge as antecedents to success (Panjer, 2007; Ahmed & Gokhale, 2009; Au et al., 2009; Bougie et al., 2010; Shihab, et al, 2010; Zhang, et al., 2012). In the case of the Mozilla meta-organisation, these categories include “platform”, which refers to the underlying computer hardware paradigm upon which computer programs have historically been written (Bresnahan & Greenstein, 1999); “operating system”, which refers to the interface between the computer hardware and software which manages the allocation of computing resources such as memory, storage, and processing power (Tanenbaum & Bos, 2014); “product”, which refers to the software program that performs a task that is useful to its user (Ruffin & Ebert, 2004); “component”, which refers to a piece of software code that implements a useful task that is useful in many different software programs, such as basic calculations, clocks, visual layouts, which are all independent of the purpose of the software program that aggregates these components to perform a task (Ajila & Wu, 2007; Haefliger, von Krogh, & Spaeth, 2008); and, “classification”, which refers to the software program design paradigm used to design the performance of the task the software is used for, such as the client vs. server communication paradigm (Lewis, 1995).

Another measure commonly reported in the open source literature as an antecedent for success is the prioritization of problems in meta-organisations. Commonly reported factors for this measure include the dependencies between related problems (Sandusky, Gasser, & Ripoche, 2004), the severity level assigned to a problem (Panjer, 2007; Herraiz, 2008; Shihab, et al, 2010; Guo, et al., 2011; Zhang, et al., 2012), the priority level assigned to a problem (Bougie et al., 2010; Giger, Pinzger, & Gall, 2010; Shihab, et al, 2010), and the perceived impact of the problem and its solution on stakeholders in the meta-organisation (Guo, et al., 2010).

The extant literature has considered a number of factors that affect the knowledge production process including the entry points, exit points, and states of the process itself (Baysal, et al., 2012b); the magnitude and nature of the engagement of actors during the different states and transitions in the knowledge production process (Hooimeijer & Weimer, 2007; Guo, et al., 2010); and, the degree to which the formal process is respected (Koponen, 2006). The directness success measure is also theorized to affect other success measures, making reopening and reassigning contextually antecedent factors with respect to certain other success measures (Guo et al., 2010, 2011).

The actors who engage in the knowledge production process have been considered as antecedent factors for success. The roles in the knowledge production process that these actors play are a primary antecedent, particularly the roles of problem knowledge producer, the actor who creates the initial problem knowledge submitted to the meta-organisation; the solution knowledge producer, the actor who creates the solution to the problem; and, the solution knowledge verifier, the actor who verifies that the solution matches the problem. Other secondary roles include the triager, the actor who confirms problems and assigns them to appropriate solution knowledge producers; commenters, actors who provide emergent knowledge to assist in the solution production process; and, influencers, actors who are peripherally involved with the knowledge production process through signalling mechanisms such as voting or watching (Anvik, Hiew, & Murphy, 2006; Panjer, 2007; Au et al., 2009; Giger, Pinzger & Gall, 2010; Shihab, et al, 2010; Zimmermann, et al., 2012).

Lastly, not all actors in the meta-organisation are equal. The extant literature has considered how actor heterogeneity antecedent factors affect success. Commonly reported factors include the popularity, visibility, reputation, skills, experience, and relationships between actors, measured in numerous different ways (Mockus, 2002; Sandusky, Gasser, & Ripoche, 2004; Hooimeijer & Weimer, 2007; Kidane & Gloor, 2007; Panjer, 2007; Au, et al., 2009; Guo, et al., 2010; Ko & Chilana, 2010; Shihab, et al., 2010; Baysal, et al., 2012ab). It has also considered how actor involvement affects prioritization of production effort and inclusion and/or exclusion of other individual (Lakhani & von Hippel, 2003; Bagozzi & Dholakia, 2006; Dahlander & O'Mahony. 2011; Dahlander & Frederiksen, 2012) or organisational (Dahlander & Magnusson, 2005; Bonaccorsi & Rossi, 2006; West & O'Mahony, 2008; West & Wood, 2014) stakeholders, and the complexities of their relationships (Mockus, 2002; Guo, et al., 2010; Baysal, et al., 2012a, 2013).

Collectively, these many measures in the open source literature relate closely to several research streams in the KBV strategy literature and offer novel ways of operationalising factors that bridge both literatures' research conversations. In the following section, six major areas of the KBV literature are discussed and linked to these measures from the open source literature.

Factors affecting knowledge production efforts in KBV literature

The KBV literature is ripe with research on factors affecting the production and utilization of knowledge in organisations. Six major research streams are discussed and related to the open source factors considered in this study.

Absorptive capacity

Absorptive capacity is the ability to “recognize the value of new, external [knowledge], assimilate it, and apply it” (Cohen & Levinthal, 1990: 128; Lane & Lubatkin, 1998: 461). It has both individual and organisation level representations that are distinct and a function of heterogeneous expertise. (Cohen & Levinthal, 1990) and heterogeneous knowledge bases (Lane & Lubatkin, 1998) between the producer and consumer of the knowledge. The potential absorptive capacity of an individual or organisation may not be wholly fulfilled in terms of realized absorptive capacity in specific knowledge consumption and application circumstances (Zahra & George, 2002). A large number of factors have been theorized to affect absorptive capacity, such that the literature has had some difficulty in standardizing the constructs for empirical examination (Lane, Koka, & Pathak, 2006; Todorova & Durisin, 2007), especially the locus of effects given many antecedents are multilevel in nature (Pisano, 1994; von Hippel, 1994; Volberda, Foss, & Lyles, 2010).

One of the common themes in the absorptive capacity literature is the factors that affect absorptive capacity load—the notion that individuals and organisations that attempt to juggle too many balls at once may not have the ability to effectively take on new knowledge-based tasks. (Cohen & Levinthal, 1990; Zahra & George, 2002; Jansen, van den Bosch, & Volberda, 2005; Todorova & Durisin, 2007). Whereas previously the amounts of knowledge that were handled in organisations were manageable, “what is happening today is that there has been a qualitative change in the way in which vast amounts of data can be collected and communicated. The risk is information overload” (Quintas, Lefrere, & Jones, 1997: 322). This factor relates closely to the “number of open problems” factor discussed in the open source literature. In both cases, the primary factor is the degree to which an actor is already burdened and therefore unable to effectively process new knowledge in knowledge production efforts.

This load is not necessarily even for all types of knowledge or actors and may rather be related to the way the knowledge is represented as well as the prior knowledge of producers and consumers of the knowledge (Cohen & Levinthal, 1990; Lane & Lubatkin, 1998; Zahra & George, 2002; Schmidt, 2010; Spithoven, Clarysse, & Knockaert, 2011). This factor maps closely to the way open source meta-organisations classify knowledge into platforms, operating systems, products, components, and classifications.

Absorptive capacity load may also be temporal in nature (Cohen & Levinthal, 1990; Lane & Lubatkin, 1998; van den Bosch, Volberda, & Boer, 1999; Tu, Vonderembse, Ragu-Nathan, & Sharkey, 2006) and have different effects based on the breadth and depth of knowledge (van Wijk, van den Bosch, & Volberda, 2011) both of which are related to its experience effects through activities (Lane & Lubatkin, 1998; Lane, Salk, & Lyles, 2001; Lichtenthaler, 2009). Similarly, the open source literature is concerned with the timing of new knowledge release, the activities and focus of actors in the meta-organisation, and their learning over time and involvement.

Social cycles (Haas, 2006) and organisational processes and structure are also theorized to affect absorptive capacity (Jansen, van den Bosch, & Volberda, 2005; Lichtenthaler, 2009). The open source literature has similar concerns about timing relative to cycles and processes in the meta-organisation and the effects on success factors.

These conceptualizations of absorptive capacity as an antecedent factor for success are connected to their counterparts in the open source literature and operationalized into the variables tested empirically in this study, as illustrated in Figure 12, Figure 20, and Figure 27.

Codifiability

Codifiability "refers to the ability of the firm to structure knowledge into a set of identifiable rules and relationships that can be easily communicated" (Kogut & Zander, 1992: 387). Not all knowledge is "codifiable", particularly when then knowledge is dependent on particular innate skills or knowhow. It is also possible that certain types of knowledge cannot be broken down by virtue of the knowledge itself due to a complexity in the properties of the knowledge such that there is causal ambiguity surrounding the properties of the knowledge making it unclear which properties are the most salient for its observed effects when applied to practice. "[As] dimensions [they] are not independent. Codifiability and complexity are related, though not identical." (Kogut & Zander, 1992: 387). These concepts are central to the KBV, with the literature suggesting that firms exist in part because of their ability to more efficiently codify complex knowledge than other forms of production (Kogut & Zander, 1992, 1993; Lam, 1997; Cowan, 2001; Levi, Kleindorfer & Wu, 2003; Reagans & McEvily, 2003; Turner & Makhija, 2006; van den Berg, 2013).

The complexity dimension of codifiability matches closely to the complexity factor of bug reports described in the open source literature, as represented by length of information as well as readability. Contextual and corroboratory factors, including the processes used in organisations (Schulz & Jobe, 2001), the distributed tacit knowledge amongst participants (), and the ways knowledge is represented (Nonaka & Konno, 1998), also match closely with counterpart concepts in the open source literature such as the knowledge and skills of actors in the mete-organisation, the categorization of knowledge representations as platform, operating system, etc., and the experience and involvement levels of different actors, resulting in different sets of tacit knowledge.

These conceptualizations of codifiability as an antecedent factor for success are connected to their counterparts in the open source literature and operationalized into the variables tested empirically in this study, as illustrated in Figure 14, Figure 21, and Figure 28.

Dominant knowledge paradigm

The KBV literature has examined the degree to which the ways of representing knowledge and the popularity of those representations affect outcomes of interest (Grant, 1996a, 1996b; Szulanski, 1996). This factor represents the intersection of the properties of knowledge and the social factors that govern its use in organisations (Lam, 1997, 2000; Hassard & Kelemen, 2002; Girard, 2015). The popularity of a given representation of knowledge acts as a form of path dependency for both future representations of knowledge and its applicability to new knowledge creation (Teece, Pisano, & Shuen, 1997). Dominant knowledge paradigms may also be a function of the information systems use to store and disseminate knowledge throughout organisations, with such systems shaping how the knowledge is encoded to be stored in the system and how it is encoded to be retrieved, independent of the properties of the knowledge itself (Nemati, Steiger, Iyer, & Herschel, 2002), creating another form of path-dependency.

These path dependencies based on the representation and storage/retrieval of knowledge assets may present challenges in periods of paradigm shift, both internally and externally (Allarakhia & Walsh, 2011, 2012) such as the paradigm shift from server computing to personal computers in the 1980s and 1990s (Cusumano & Selby, 1995), the shift to open source collaborative innovation in the 2000s (Baldwin & von Hippel, 2011), the shift to mobile computing in recent years (Dittrich & Duysters, 2007; West & Wood, 2014), the emerging field big data computing (LaValle, Lesser, Schokley, Hopkins, & Kruschwitz, 2011; Davenport, Barth, & Bean, 2012; McAfee & Brynjolfsson, 2012; Chang, Kauffman, & Kwon, 2014; George, Haas, & Pentland, 2014). These paradigm shifts have significant implications for knowledge creation and management in organisations (Nonaka, Umemoto, & Senoo, 1996), as do the dominant knowledge paradigms before and after the shifts (Prahalad & Bettis, 1986).

The paradigms used for representing knowledge are such influential factors that the field of "knowledge management" emerged as a distinct field that "builds on theoretical foundations from information economics, strategic management, organizational culture, organizational behaviour, organisational structure, artificial intelligence, quality management, and organizational performance measurement … [to] provide a rationale for managing knowledge, defining the process of managing knowledge, and enabling [the] evaluation of the results of this process." (Carlucci, Marr, & Schiuma, 2004; Baskerville & Dulipovici, 2006: 83).

There is significant overlap between the concepts of dominant knowledge paradigms in the KBV literature and the representations and storage of knowledge in open source meta-organisations. In particular, the choice to categorize knowledge according to the foundational computer platform, operating system, product, component, and classification results in organisational segmentation of knowledge. The Bugzilla knowledge repository from which the data in this study were drawn is a knowledge management system designed specifically to facilitate the representation of problem knowledge and the production of solution knowledge in open source meta-organisations (Serrano & Ciordia, 2005).

These conceptualizations of dominant knowledge paradigm as an antecedent factor for success are connected to their counterparts in the open source literature and operationalized into the variables tested empirically in this study, as illustrated in Figure 15, Figure 22, and Figure 29.

Knowledge flow impediments

The management literature has done considerable research on knowledge flows—the process or life cycle through which knowledge proceeds from creation to utilization. Studies have examined the internal organisational structure factors creating and influencing knowledge flows (Gupta & Govindarajan, 1991, 2000; Schulz, 2001; Garrett Jr. & Covin, 2015), the reciprocity of influence of knowledge flows on structural factors in organisations (Birkinshaw, Nobel, & Ridderstråle, 2002; Macpherson & Holt, 2007), and knowledge flows beyond organisational boundaries (Appleyard, 1996; Carlile, 2004; Malhotra, Gosain, & El Sawy, 2005; Singh, 2005; Sorenson, Rivkin, & Fleming, 2006; Bell & Zaheer, 2007; Zucker, Darby, Furner, Liu, & Ma, 2007).

Given the considerable wealth of studies and factors thought to affect the flow of knowledge, a stream of management practitioner-focused literature has emerged, describing the optimal ways to design knowledge flows and things to avoid in order to improve outcomes (Fahey & Prusak, 1998; Birkinshaw & Sheehan, 2002; Maier & Remus, 2003; Garud & Kamaraswamy, 2005). One of the most prevalent challenges is addressing knowledge flow impediments, which are often related to motivating actors to participate in the knowledge flow as it is designed (Starbuck, 1992; Carayannis, Alexander, & Ioannidis, 2000; Schulz, 2003; Garud, 2005; Kärreman, 2009). Another issue is the challenge associated with classifying disparate but contingent types of knowledge effectively in the knowledge flow process (Cheung, Lee, & Wang, 2005; Kulkarni, Ravindran, & Freeze, 2006), which can often be a function of the knowledge management systems used in organisations (Alavi & Leidner, 2001).

The open source literature's life cycle processes overlap considerably with the knowledge management literature for optimal knowledge flow design. Both share an interest in identifying antecedent factors affecting knowledge flows. The activities of different actors in open source meta-organisations match closely to the actions of different actors within and between organisations. In fact, meta-organisations have been conceptualized as a form of loose alliance with different characteristics from the traditional inter-firm alliances described in the KBV literature, as well as heralded as having different organisational structure characteristics than traditional organisations (Demil & Lecocq, 2006; Gulati, Puranam, & Tushman, 2012). As such, the study of knowledge flows in open source meta-organisation is a rich opportunity to contribute to the knowledge flow literature an examination in a novel organisational structure.

These conceptualizations of knowledge flow impediments as an antecedent factor for success are connected to their counterparts in the open source literature and operationalized into the variables tested empirically in this study, as illustrated in Figure 16, Figure 23, and Figure 30.

Knowledge stakeholder influence

The KBV literature has examined both individual and organisation level factors that influence the knowledge production process. Bill Starbuck's classic qualitative study (1992) set the stage for detailed examinations of the relationship of the power and influence of actors and knowledge production work over the subsequent decades (Kärreman, 2010), challenging the traditional “functional view” and introducing concepts such as institutional factors and rhetorical discourses (Alvesson, 1993). A social perspective on the production and consumption of knowledge emerged (Anand, Glick, & Manz, 2002), where the interconnections of knowledge stakeholders were found to be just as important as the independent knowledge production activities and the properties of the resulting knowledge (Bell & Zaheer, 2007). From this perspective, “knowledge can be seen as a product of power relations … [by] recogniz[ing] that knowledge is aa process or set of relationships” (Quintas, Lefere, & Jones, 1997: 322).

The human, organizational, and social capital factors of knowledge management have also been considered (Subramaniam & Youndt, 2005; Alavi, Kayworth, & Leidner, 2014). In particular, localizing the effects of knowledge stakeholder influence antecedents to appropriate level of analysis has been an ongoing challenge (Argote, McEvily, & Reagans, 2003). The knowledge stakeholder influence literature also bridges into the alliances literature by considering the effects of heterogeneous power dynamics on the direction and nature of knowledge flows between allied organisations (Inkpen & Dinur, 1998). It further bridges into the networks literature, considering both network structure and degree of distribution of knowledge on outcomes of interest (Rodan & Galunic, 2004; Dantas & Bell, 2009).

Given that much of the knowledge in organisations is tacit to individuals (Nonaka & Konno, 1998; Lam, 2000; Nonaka & von Krogh, 2009), it is unsurprising that the specific knowledge that resides in individuals is heterogeneous (Rodan & Galunic, 2004), sometimes conceptualized as differing subject matter expertise (Ardichvili, Page, & Wentling, 2003). This separation of knowledge can be both a facilitator and a barrier to knowledge production depending on the actor (Franke & von Hippel, 2003). Actors with different skills tend to engage in different roles in the knowledge production process (Brown & Duguid, 1991; Ardichvili, Page, & Wentling, 2003; Zahra & Filatotchev, 2004), which can exacerbate the knowledge differences and result in certain roles having stronger effective power in organisations by virtue of their knowledge.

The differential knowledge between individuals and organisations matches closely to the open source literature examinations of actor reputation, skill, and experience in meta-organisations. Often these discussions separate “developers” and related roles, which are simply actors that have different subject matter expertise, different needs, and different degrees of influence on the knowledge production process (Franke & von Hippel, 2003). The power and influence of management versus employees in traditional firms that use fiat-based decision making is the counterpart to actor centrality and the resulting supposedly “lateral” decision making influence in meta-organisations (Dahlander & O’Mahony, 2011; Dahlander & Frederiksen, 2012; Gulati, Puranam, & Tushman, 2012). The observation of other actors in meta-organisations is the counterpart to traditional social network maps in the KBV literature.

These conceptualizations of knowledge stakeholder influence as an antecedent factor for success are connected to their counterparts in the open source literature and operationalized into the variables tested empirically in this study, as illustrated in Figure 17, Figure 24, and Figure 31.

Solution knowledge value

“Solution knowledge” is knowledge on how to solve a certain problem, addressing a certain need or providing a certain function; it is the counterpart to “problem knowledge”, which is knowledge about current or anticipated technological problems for which the firm seeks others' support (Jeppesen & Laursen, 2009; Afuah & Tucci, 2012; Alexy, George, & Salter, 2013).

The value of a given set of solution knowledge is a frequently discussed factor in the literature that is theorized to dramatically affect the knowledge production process. Different organisations may derive value differently from the same set of solution knowledge (Davis & Botkin, 1994; Alexy, George, & Salter, 2013). This heterogeneity of value measurement amongst organisations and the individuals who are members of those organisations may be due to network position (Kogut, 2000), existing stocks of knowledge (Dierickx & Cool, 1989; Decarolis & Dees, 1999) or complementary intellectual capital assets (Wiig, 1997), team configurations (Lewis, 2004), the geographic local and related environment support factors (Cooke, 2005), the systems and processes used to manage knowledge (Swan, Newell, Scarbrough, & Hislop, 1999), the organisations' dynamic capabilities to reconfigure knowledge resources (Teece, Pisano, & Shuen, 1997; Eisenhardt & Martin, 2000; Easterby-Smith & Prieto, 2008), cultural differences (Cohen, 1998), the degree of intangibility of the knowledge (Tomas & Hult, 2003), the configuration of firm-user interactions (Dahlander & Magnusson, 2005; Bagozzi & Dholakia, 2006; Marh & Lievens, 2012), the transient uses of the knowledge (Bozeman & Rogers, 2002), the accounting principles and other metrics used to measure knowledge (Kanevsky & Housel, 1998; Martin, 2004; Xy & Bernard, 2011; Massingham, 2016), and the functional dependencies and knowledge asymmetries of knowledge producers and knowledge consumers (Das, 2003; Majchrzak, More, & Faraj, 2011).

Most of these factors overlap closely with those factors considered in the open source literature. In particular, the different value of knowledge between alliance partners considered at length in the KBV and alliances literature (c.f. Inkpen, 2000) matches closely with the representation of open meta-organisations as loosely organised alliances (Gulati, Puranam, & Tushman, 2012). And, the metrics used to attempt to represent knowledge value and the processes used to prioritize the production of certain sets of knowledge match closely to the categorization and resource allocation processes in open source meta-organisations. In fact, the signalling artefacts and processes used in open source meta-organisations contributes to the literature on ways of measuring knowledge value by evaluating the effectiveness of the knowledge value measurements used in this novel context.

These conceptualizations of solution knowledge value as an antecedent factor for success are connected to their counterparts in the open source literature and operationalized into the variables tested empirically in this study, as illustrated in Figure 18, Figure 25, and Figure 32.

CHAPTER THREE: CONCEPTUAL FRAMEWORK

Research positioning

In their seminal paper, Alexy, George, and Salter (2013) argued that by revealing "problem knowledge" to an open source meta-organisation, firms get back "solution knowledge" that solves their problem (Jeppesen & Laursen, 2009; Afuah & Tucci, 2012; Alexy, George, & Salter, 2013). More efficient use of this "knowledge-revealing strategy" by the firms relative to its competitors is said to lead to competitive advantage by creating "solution knowledge" that is more relevant and useful to the focal organisation than its competitors. In essence, Alexy, George, and Salter. (2013) provide a partial answer to the question "Why participate in open source?" using the knowledge-based view of the firm. Figure 1 depicts the framework adapted from Alexy, George, and Salter (2013) upon which the present study builds. The definitions of "problem knowledge" and "solution knowledge" that are adapted from Alexy, George, & Salter (2013) in the present study were derived in their study from the work of Afuah and Tucci (2012) and Jeppesen and Laursen (2009) on interactions between firms and outside sources of knowledge development such as open source meta-organisations, making these salient definitions particularly suitable for the present research context.

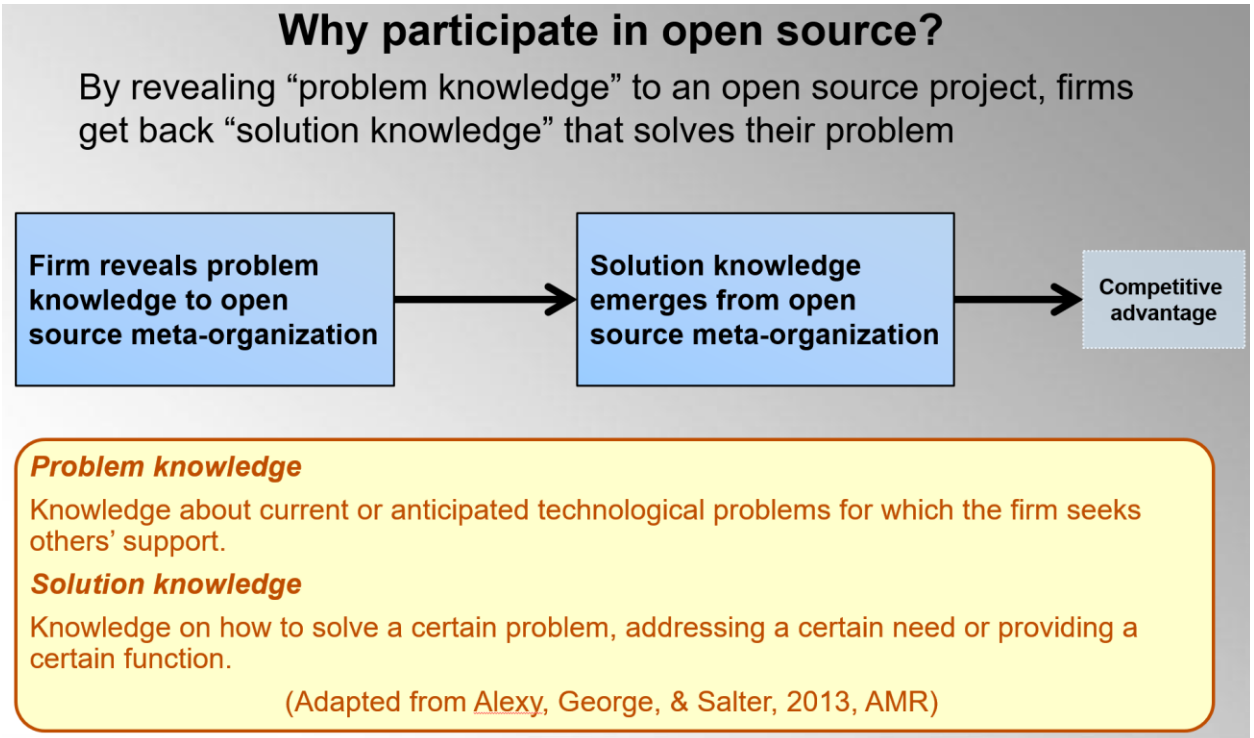

Figure 1: Knowledge revealing strategy

This study extends the model of Alexy, George, and Salter (2013) by examining the antecedent factors that affect solution knowledge emergence subsequent to the use of a knowledge revealing strategy by organisations. Whereas the focus of their propositional exposition was the factors involved in the decision to use a knowledge-revealing strategy and the expected positive outcomes for the firm, not all uses of a knowledge-revealing strategy lead to positive outcomes. No study to date has examined why. This study fills that gap by identifying the antecedent factors that increase the probability and/or magnitude of positive outcomes for organisations using knowledge-revealing strategies, as depicted in Figure 2.

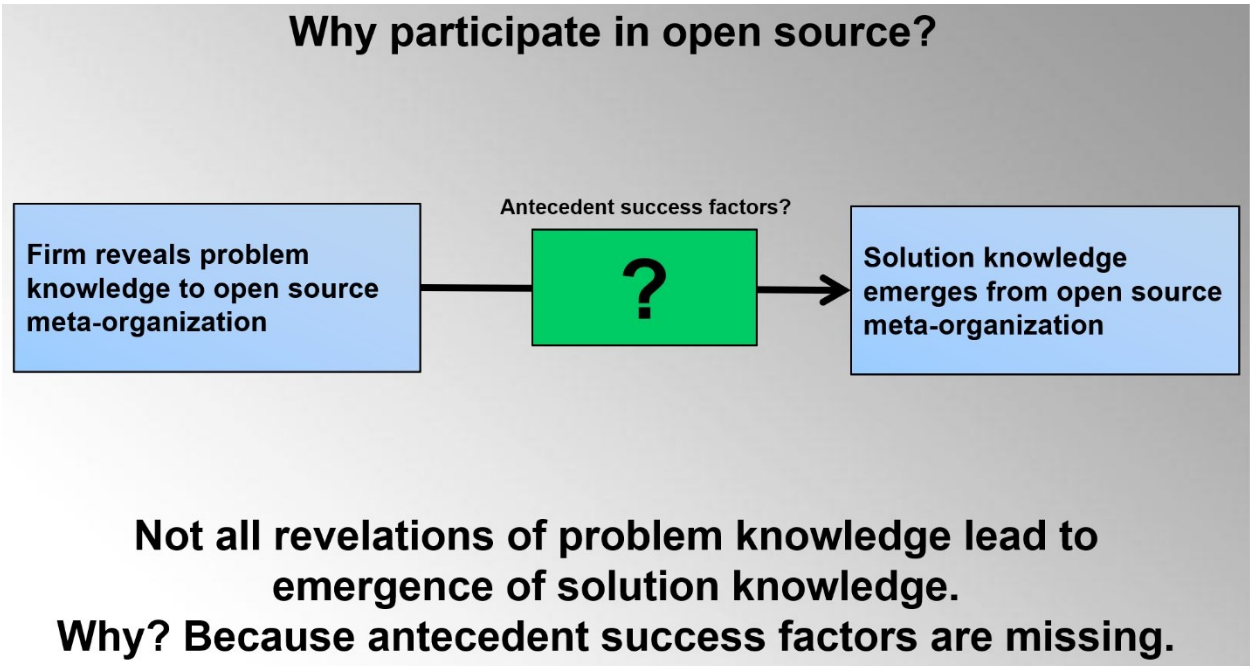

Figure 2: Gap filled by present study

In the context of extant research, this study is positioned as depicted in Figure 3. The left side of the figure portrays the work of Alexy, George, and Salter (2013). Their model suggests strategic factors surrounding a focal firm, namely benefits, drivers, and collaboration needs, lead to the decision to use a knowledge revealing strategy. The authors suggest that those firms which choose to use a knowledge revealing strategy will gain a competitive advantage by virtue of the solution knowledge that emerges as a consequence of the use of that knowledge revealing strategy. Yet, not all uses of knowledge revealing strategies result in solution knowledge emergence. The present research fills that gap by identifying, contextualizing, and measuring the contingency factors that affect solution knowledge emergence once a firm engages in a knowledge revealing strategy.

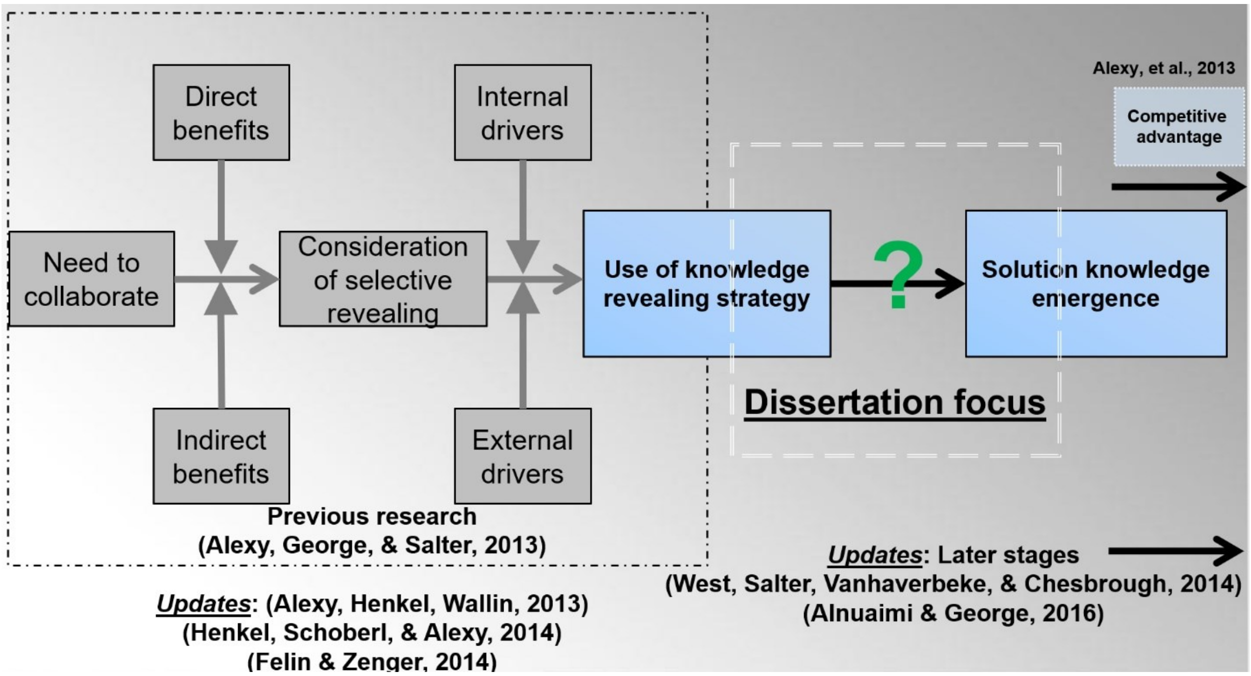

Figure 3: Positioning of dissertation relative to extant research

Antecedent factors impacting solution knowledge emergence

Categorization

Since this study focuses on examining antecedent factors affecting organisations that have already decided to use a knowledge-revealing strategy—Alexy, George, and Salter (2013) having already examined why firms may decide to not use the strategy—the antecedent factors are conceptualized as independent variable influencing the dependent variable of interest, namely solution knowledge emergence. The literature review, linking the open source and KBV literatures, revealed 86 potential antecedent measurements and 21 potential outcome measurements, spanning three levels of analysis. The antecedents were organized into the conceptual KBV categories identified in the literature review, resulting in the conceptual framework depicted in Figure 4.

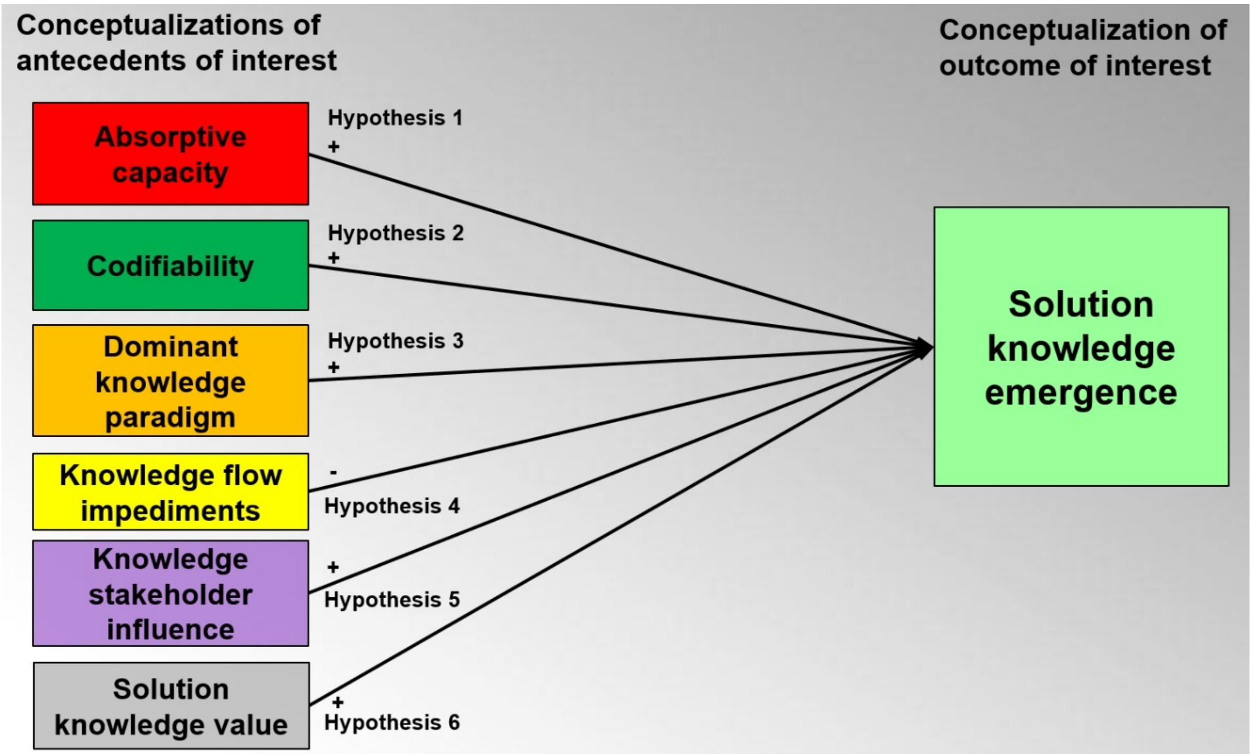

Figure 4: Conceptual framework

Hypotheses

The potential antecedent factors reported in the literature were categorized resulting in six hypotheses crafted to attempt to answer the research question, "*What are the factors driving successful solution knowledge emergence?*" Table 2 presents the formulation of the hypotheses.

Number	Hypothesis	Direction
H1	*The absorptive capacity of the meta-organisation is positively correlated with solution knowledge emergence*	+
H2	*The codifiability of the problem knowledge revealed to the meta-organisation is positively correlated with solution knowledge emergence*	+
H3	*The similarity of the problem knowledge revealed to the meta-organisation to the dominant knowledge paradigm in the meta-organisation is positively correlated with solution knowledge emergence*	+
H4	*Knowledge flow impediments are negatively correlated with solution knowledge emergence*	-
H5	*Knowledge stakeholder influence is positively correlated with solution knowledge emergence*	+
H6	*Solution knowledge value is positively correlated with solution knowledge emergence*	+

Table 2: Hypothesis formulation

CHAPTER FOUR: RESEARCH METHOD

Research paradigm and methodology choices

This research uses a post-positivist paradigm and a theory testing methodology to evaluate the hypotheses developed from the literature. This approach is justified for three reasons.

First, the traditional challenge of expressing social science phenomena in concise quantitative terms (Popper, 1957) is mitigated by the historical roots of open source meta-organisation processes in the mathematical and physical sciences, resulting in atomic, discrete data representations that are amenable to quantitative analysis (Lerner & Tirole, 2002; Fischer, Pinzger, & Gall, 2003).

Second, it ensures that "quantification and the use of sophisticated statistical methods and mathematical models" are not "taken as a sufficient and necessary basis for the production of valid empirical evidence" alone and rather are complemented by "a theoretically relevant interpretation of this evidence" in an "integrated and deliberative methodological approach" (Adam, 2014: 6).

Third, the theory testing methodology is appropriate when incrementally building upon a normal science research stream where phenomena are well described in the literature and parsimonious data are available as is the present case (Kuhn, 1970). Further, theory testing is well suited to the available archival data source because measurement of the variables of interest is nonreactive (Singleton & Straits, 2005: 354) and possible alternate outcomes (dependent variable measurement) can be constrained according to the theoretical formulation (independent variables) using the variability inherent in the data (Roberts & Pashler, 2000).

Data

Access to an archival data source was negotiated with the Mozilla Foundation, one of the largest open source meta-organisations, best known for the development of the Firefox web browser. The data, which reside in a relational database known as "Bugzilla" (a portmanteau for "bug" and "Mozilla"), "describe interesting aspects of the evolutionary changes of a project" making them "a valuable source for retrospective analysis techniques" which "enable reasoning about the past and anticipating future evolution of software projects" (Fischer, Pinzger, & Gall, 2003: 23). It is also a prominent example of a "virtual lead user community", which are theorized to be environments where the proactive creation of solution knowledge is more common than in other configurations of cross-organisational-boundaries knowledge creation efforts (Mahr & Lievens, 2012). The choice of the context of open source meta-organisations responds to numerous calls for research on collective knowledge production environments (Henkel & von Hippel, 2005; Benkler, 2006; Dahlander & Wallin, 2006; Jeppesen & Frederiksen, 2006; Shah & Tripsas, 2007; Bogers, Afuah, & Bastian, 2010;

The Bugzilla system itself is not exclusive to Mozilla and is used by many other major organisations including IBM, Google, and Eclipse to assist in their software development projects. The Mozilla database was selected for this research as it has been in continuous usage since 1998, leading to more than 1 billion data points for use in this research, which may be the largest such database in existence. Further, while the Mozilla database has been examined by information technology researchers, its knowledge-based strategic insights have never been considered in management research, making its examination in this study novel.

While the research focuses on a single open source meta-organisation, the Mozilla Foundation, the intention is to conservatively generalize (or at least lay the foundation for future research that relates) to the population of all open source meta-organisations. This focused sample is logistically justified as I have negotiated access to the database. The database is sufficiently large to triangulate the research question from multiple angles, promoting validity and permitting reliability testing to take place in future research by contrasting other similar databases in other open source meta-organisations. The intention of the present study is to lay the foundation for a long-term research program that builds upon these preliminary insights.

Data access and ethical considerations

The Mozilla Bugzilla database is publicly accessible through a web portal that allows searching with specific queries, similar to other databases used in strategic management such as COMPUSTAT (See: http://bugzilla.mozilla.org). However, this interface is not suitable for large scale retrieval and analysis. As such, a complete offline copy of the database was requested from the Mozilla Foundation for research purposes. The maintainers of the database evaluated and approved the request and provided a complete copy of the database, with all entries up to the end of 2012. Because the database is public, it is well understood by participants that any actions that they take will be documented indefinitely in this database and such data may be used for any purpose. The maintainers of the database reviewed a description of this dissertation's intended research focus against the guideline used to assess similar requests from previous academic researchers in other fields and agreed that no specific additional ethical considerations were required. The use of the database and this research have also been discussed with several employees of the Mozilla Foundation who believe that it is an appropriate use of the data and that the outcomes will be of benefit to the organisation and its participants. The use of these data has also received formal research ethics clearance by York University's Faculty of Graduate Studies.

Levels of analysis

The Bugzilla database contains dozens of cross-linked tables of data that were organized into three levels of analysis: problem, individual, organisation. These levels of analysis were concisely delineated in order to more cleanly identify the level at which the antecedent knowledge factors reside, an issue that has been challenging in past KBV and related research streams (Priem & Butler, 2001; Lakhani, Lifshitz-Assaf, & Tushman, 2012).

Problem level of analysis

The problem level of analysis examines the factors that are part of or related to the problem knowledge revealed by an organisation participating in the meta-organisation. All of the initial and emergent problem knowledge resides at this level of analysis. The unit of analysis is "a bug". The term "bug" is an artefact of the roots of most open source meta-organisations in software development. The term "bug" was used to describe "an error, flaw, failure, or fault in a computer program or system that produces an incorrect or unexpected result, or causes it to behave in unintended ways" (Wikipedia, 2017). In the present context, the term "bug" is extended to include problems of all types, including those that might related to errors in the software, while also including problems that might be related to developmental philosophy, features, enhancements, marketing, branding, support, and the broad array of related elements that are of strategic interest to organisations using knowledge-revealing strategies. Each “bug” is treated as a discrete knowledge-revealing act by participating organisations, providing a clear definition and boundary for the problem knowledge that is revealed and the factors that interact with it.

id	name	description
19	bug_file_loc	URL
21	keywords	Keywords
22	status_whiteboard	Status Whiteboard
23	bug_id	Bug #
24	short_desc	Summary
25	product	Product
26	version	Version
27	rep_platform	Platform
28	op_sys	OS/Version
29	bug_status	Status
30	resolution	Resolution
31	bug_severity	Severity
32	priority	Priority
33	component	Component
34	assigned_to	AssignedTo
35	reporter	ReportedBy
36	qa_contact	QAContact
37	cc	CC
38	dependson	Depends on
39	blocked	Blocks
40	target_milestone	Target Milestone
41	votes	Votes
42	longdesc	Comment
43	delta_ts	Last changed date
44	days_elapsed	Days since bug changed
45	attachments.description	Attachment description
47	attachments.mimetype	Attachment mime type

id	name	description
45	attachments.description	Attachment description
47	attachments.mimetype	Attachment mime type
48	attachments.ispatch	Attachment is patch
49	everconfirmed	Ever Confirmed
50	assignee_accessible	assignee_accessible
51	cclist_accessible	CC Accessible
52	qacontact_accessible	qacontact_accessible
53	reporter_accessible	Reporter Accessible
54	attachments.isobsolete	Attachment is obsolete
56	alias	Alias
64	attachments.filename	Attachment filename
65	attachments.isprivate	Attachment is private
66	bug_group	Group
67	estimated_time	Estimated Hours
68	remaining_time	Remaining Hours
69	flagtypes.name	Flags
71	setters.login_name	Flag Setter
72	work_time	Hours Worked
73	percentage_complete	Percentage Complete
74	content	Content
75	classification	Classification
76	requestees.login_name	Flag Requestee
77	owner_idle_time	Time Since Assignee T...
78	deadline	Deadline
79	commenter	Commenter
80	attachments.submitter	Attachment creator
81	creation_ts	Creation date

id	name	description
81	creation_ts	Creation date
82	longdescs.isprivate	Comment is private
83	attach_data.thedata	Attachment data
84	attachments.isurl	Attachment is a URL
86	attachment_cf_fixed_...	Attachment Fixed In
88	cf_blocking_fennec	tracking-fennec
89	cf_blocking_191	blocking1.9.1
90	cf_status_191	status1.9.1
92	cf_status_192	status1.9.2
93	cf_blocking_20	blocking2.0
94	see_also	See Also
95	cf_blocking_thunderb...	blocking-thunderbird3.0
96	cf_status_thunderbird...	status-thunderbird3.0
98	cf_blocking_192	blocking1.9.2
99	cf_blocking_thunderb...	blocking-thunderbird3.1
100	cf_status_thunderbird...	status-thunderbird3.1
101	cf_status_20	status2.0
103	cf_status_seamonkey...	status-seamonkey2.1
104	cf_blocking_seamon...	blocking-seamonkey2.1
105	cf_blocking_thunderb...	blocking-thunderbird3.2
106	cf_status_thunderbird...	status-thunderbird3.2
109	cf_blocking_thunderb...	blocking-thunderbird5.0
110	cf_status_thunderbird...	status-thunderbird5.0
112	cf_blocking_fx	blocking-fx
113	cf_tracking_firefox5	tracking-firefox5
114	cf_status_firefox5	status-firefox5
115	cf_tracking_firefox6	tracking-firefox6

Figure 5: Initial problem level data fields (excerpt)

The database contained more than 900,000 bug units at the problem level (prior to imputation). Each entry had more than 100 variables associated with it directly and thousands of variables associated with it through cross-referencing with other tables in the database. Many of these variables were not relevant to the present research, but certain variables such as status, priority, severity, and description were instrumental for hypothesis testing. Figure 5 depicts an excerpt of the initial problem level data fields as they appeared in the database.

Individual level of analysis

The individual level of analysis examines the actors who are involved in the knowledge production process in the open source meta-organisation. The unit of analysis is the "profile". Each profile has a unique identification number in the database that enables the tracking of each

userid	login_name	domain	realname	comment_count	creation_ts	first_patch_bug_id	first_patch_approved_id
138037	K.Sidiropoulos@web.de	web.de	Dr. K. Sidiropoulos	0	2011-04-23 07:05:38	NULL	NULL
296320	galar71@gmail.com	gmail.com	Ӈÿyvind	21	2007-12-20 02:36:34	NULL	NULL
60234	hasse@jasajudeju.se	jasajudeju.se	Hasse	1120	2002-07-25 09:34:09	226342	135999
193523	mozilla-bugzilla@iansealy.com	iansealy.com	Ian Sealy	1	2005-03-31 06:21:49	NULL	NULL
331594	himskim@msn.com	msn.com		0	2008-11-25 04:15:04	NULL	NULL
410793	vrac@an16.fr	an16.fr		2	2011-03-25 03:04:52	NULL	NULL
306192	lukebenes@hotmail.com	hotmail.com	Luke	5	2008-04-06 11:21:57	NULL	NULL
249622	tom@polemian.org	polemian.org	psyched_weels	2	2006-07-04 14:33:06	NULL	NULL
132266	k_f_e1@hotmail.com	hotmail.com	king	0	2011-04-23 07:05:38	NULL	NULL
56695	bradnixon@rogers.com	rogers.com	Brad Nixon	3	2002-06-11 18:15:00	NULL	NULL
267372	noefi1@hotmail.com	hotmail.com	gsxr600	0	2011-04-23 07:05:38	NULL	NULL
169432	chardt.mozdev@gmx.net	gmx.net	Christian Hardt	0	2011-04-23 07:05:38	NULL	NULL
229835	mozilla@bleys.spb.ru	bleys.spb.ru	Alex	1	2005-12-27 04:30:38	NULL	NULL
24217	jeffrey.phillips@staffeon.com	staffeon.com	Jeffrey Phillips	11	2001-05-08 11:28:05	NULL	NULL
371667	josip@kastela.org	kastela.org	jkrokar	0	2010-01-28 14:13:08	NULL	NULL
373173	jothi.sivathanupillai@ferguson.com	ferguson.com	Jothi	0	2010-02-09 08:51:29	NULL	NULL
48578	its_my_mailbox@gmx.net	gmx.net		0	2011-04-23 07:05:38	NULL	NULL
118668	hvac2805@cs.com	cs.com	Raymond Peters	0	2011-04-23 07:05:38	NULL	NULL
210788	gdarby@uiuc.edu	uiuc.edu	Sean	1	2005-07-29 11:55:22	NULL	NULL

Figure 6: Initial individual level data fields (excerpt)

individual's actions over time. Profiles are wholly contained descriptors of the participants in the open source meta-organisation. There were over 1 million unique profiles in the database (prior to imputation). Figure 6 depicts an excerpt of the initial individual level data fields as they appeared in the database.

By cross-referencing the initial problem level (bug) and individual level (profile) tables over time, the result is a longitudinal account of all the actions each individual participant has taken over the course of the knowledge generation process. The resulting table, referred to as the "activity" table in the database, links the problem and individual levels of analysis over time. This table is crucial for testing hypotheses related to absorptive capacity and knowledge flow.

There were over 10 million activities recorded from 1998 to the end of 2012 in the database (prior to imputation). While activities could have been treated as a separate level of analysis, the choice was made to handle those activities components that most relate to the bug component of the activity at the problem level of analysis and those activity components that relate most to the profile component of the activity at the individual level of analysis. A useful extension of the present research would consider the triadic nature of activities (bug, profile, time) in a longitudinal manner that is beyond the scope of the present study. Figure 7 depicts an excerpt of the activity data fields as they appeared in the database.

By examining the nature of the activities in which a given profile engages with respect to the bugs allows the identification of individual actor roles related to common activities. These roles organize the profiles at the individual level of analysis into three non-mutually exclusive categories, each of which participates in the knowledge production process in a distinct way. These roles represent a propensity of engaging in the knowledge flor process in a particular way that is comparable to strategic choices, enabling analysis of the factors related to individual level action in the hypotheses. To distinguish participant actions in a given role, a constraint was imposed that a profile must engage in a given role 4 or more times to be classified in that role. In this manner, one-off or fewer than 4 actions on problems by individuals are handled in aggregate at the problem level of analysis rather than at the individual level of analysis ensuring the levels remain conceptually distinct and theoretically concise.

bug_id	who	bug_when	fieldid	removed	added	attach_id
471685	233280	2008-12-31 08:56:13	69		blocking1.9.1?	NULL
471685	297995	2008-12-31 17:13:24	37		highmind63@gmail.com	NULL
471685	33840	2009-01-01 01:50:12	37		mozilla@Weilbacher.org	NULL
471685	233280	2009-01-05 12:51:46	29	NEW	ASSIGNED	NULL
471685	233280	2009-01-05 12:51:46	34	nobody@mozilla.org	sdwilsh@forerunnerdesigns.com	NULL
471685	233280	2009-01-05 12:51:46	69		review?(bugmail@asutherland.org)	355441
471685	91159	2009-01-07 08:15:53	32	--	P2	NULL
471685	91159	2009-01-07 08:15:53	69	blocking1.9.1?	blocking1.9.1+	NULL
471685	151407	2009-01-07 23:21:32	69	review?(bugmail@asutherland.org)	review+	355441
471685	233280	2009-01-08 06:34:12	22		[can land]	NULL
471685	306867	2009-01-08 08:49:43	37		drh@sqlite.org	NULL
471685	233280	2009-01-08 08:56:52	22	[can land]		NULL
471685	233280	2009-01-08 08:56:52	29	ASSIGNED	RESOLVED	NULL
471685	233280	2009-01-08 08:56:52	30		FIXED	NULL
471685	233280	2009-01-08 08:56:52	40	---	mozilla1.9.2a1	NULL
471685	233280	2009-01-08 08:56:52	184		2009-01-08 08:56:52	NULL
471685	233280	2009-01-09 10:37:54	21		fixed1.9.1	NULL
471685	233280	2009-01-09 13:23:59	21	fixed1.9.1		NULL
471685	233280	2009-01-09 13:23:59	29	RESOLVED	REOPENED	NULL
471685	233280	2009-01-09 13:23:59	30	FIXED		NULL
471685	5189	2009-01-09 15:38:38	37		kairo@kairo.at	NULL
471685	192994	2009-01-10 06:28:56	37		pablo@fliagreco.com.ar	NULL
471685	233280	2009-01-11 17:09:55	29	REOPENED	ASSIGNED	NULL
471685	233280	2009-01-13 09:30:58	29	ASSIGNED	RESOLVED	NULL
471685	233280	2009-01-13 09:30:58	30		FIXED	NULL
471685	233280	2009-01-13 09:30:58	184	2009-01-08 08:56:52	2009-01-13 09:30:58	NULL
471685	233280	2009-01-16 11:25:53	21		fixed1.9.1	NULL
471685	5189	2009-01-23 16:12:35	39		475111	NULL
471685	263812	2009-04-08 13:29:08	21	fixed1.9.1	verified1.9.1	NULL
471685	263812	2009-04-08 13:29:08	29	RESOLVED	VERIFIED	NULL

Figure 7: Initial activity table data fields (excerpt)

Individuals that frequently engage in the submission of new problem knowledge are classified in the role of “problem knowledge producer”, designated by the “reporter” field in the bugs table. Problem knowledge producers generate and disseminate the initial problem knowledge and provide additional, emergent problem knowledge during the knowledge production process as necessary.

“Solution knowledge producers” are those individuals who craft the solution that addresses the problem submitted by the problem knowledge producer. Typically, the solution involves the creation of software code that addresses the problem by resolving a bug, adding a feature, or updating information. The solution knowledge producer may collaborate with other individuals for the creation of solution knowledge that has non-trivial dependencies on knowledge residing elsewhere in the meta-organisation, either in problem or solution knowledge residing in bugs beyond the focal bug. Or, key solution knowledge may reside in tacit knowledge residing in individuals that have not yet acted on the focal bug. The solution knowledge producer role is designated by the “assigned_to” field in the bugs table.

The least common individual role is the “solution knowledge verifier”, designated by the “QA_contact” field in the bugs table. This role is in charge of verifying that the emergent solution knowledge does, in fact, resolve the initial problem knowledge that was submitted to the open source meta-organisation. The solution knowledge verifier liaises between the problem knowledge producers and solution knowledge producers to ensure that all facets of the problem have been addressed. The solution knowledge verifier may redirect incompletely solved problems back into the knowledge production process to be revisited and may identify alternate solution knowledge producers who can assist in resolving the incomplete portions of a partially resolved problem. Given the complexity of this role, typically it is engaged in only by the most experienced and involved participants in the ecosystem. Figure 8 summarizes these three types of knowledge actor roles in which individual level actors engage as described in the data.

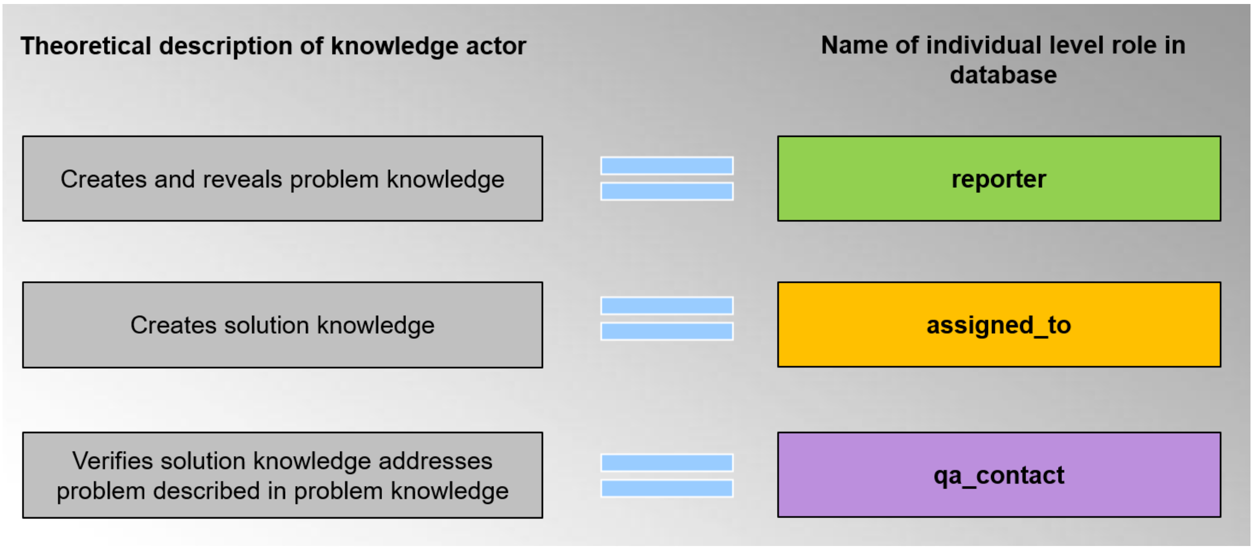

Figure 8: Individual level knowledge actor roles

Organisation level of analysis

The organisation level of analysis examines the organisations involved in the knowledge production process in the open source meta-organisation. The unit of analysis is "an organisation". The organisation level of analysis organizes the profiles in the individual level of analysis in a manner that reflects their nested nature according to organisational membership of each individual partipant. The email suffix of each profile's registered email address is used to identify the organisation in which that profile is nested. For example, the registered email address of one profile may be "john.doe@microsoft.com". Another profile might have the email address "peter.smith@microsoft.com", and so on. These addresses all share the suffix "microsoft.com", suggesting a high probability that these profiles are those of Microsoft employees. Email addresses of other major companies such RedHat, Google, IBM, and many others, all appear in the database.

Because many profiles are registered with personal email addresses instead of organisational email addresses, even when the individual actor is doing work as a member of the organisation, a conservative subset of what constitutes "an organisation" was necessary to ensure that the organisation level of analysis is not largely a restatement of the individual level of analysis. Three additional constraints were imposed to address this issue. First, all email suffixes were compared to an aggregate list assembled from six databases of known personal email provider domains and all such domains were excluded from the organisation level of analysis. For example, the domain "hotmail.com" is known to be a personal email service whose profiles are not necessarily employees of Microsoft, the parent company. All such entries were excluded from consideration at the organisation level of analysis.

Second, each organisation unit at the organisation level was only considered distinct from the individual level if at least 3 profiles existed in the database with the identifying domain name. While this constraint unduly excludes small organisations that may only have one or two people participating in the open source meta-organisation, the cutoff was selected in order to ensure that the remaining conservative sample of organisations was conceptually distinct and could be analyzed for aggregate strategic action, leaving the small organisations' actions to be analyzed at the individual level.

Third, the nested nature of the individual level of action into organisations allows the nesting of indivual level actor roles into organisation level actor roles. As such, when examing organisational actions they were also classified into problem knowledge producer, solution knowledge producer, and solution knowledge verifier. The same constraint was imposed on the inclusion in a role category as at the individual level, namely that the organisation acted at least four times in the given role to be conceptually distinct from both problem and invidual levels of analysis, maintaining theoretical conciseness.

The initial amount of unconstrained "organisations" identified in the database was over 100,000. Applying these constraints reduced the number to a conservative sample of 6,547 organisations at the organisation level of analysis. Manual inspection of the retained organisations revealed numerous reputable organisations that were known to participate in open source meta-organisations strategically, including ACM, Adobe, Dreamhost, IBM, Nokia, Oracle, PHP, Pixar, Qualcomm, Redhat, Sun Microsystems, Intel, and AMD, validating that the constraints were effective in conceptually distinguishing organisations in the database.

Community influence

The literature review revealed that the concept of community influence in meta-organisations, which is frequently mentioned across literature streams, does not have an agreed upon definition. Instead, it is frequently a component of factors at one or more of the three levels of analysis under consideration in this study. As a result, in this study, for the purposes of clarity and parsimony, the choice was made to consider the influence of community in the operationalization of the particular factors rather than as a separate level of analysis. The examination of community-level factors for a given definition of "community" would be a useful future research extension that is beyond the scope of the present study. The levels and units of analysis examined in this study are depicted in Figure 9.

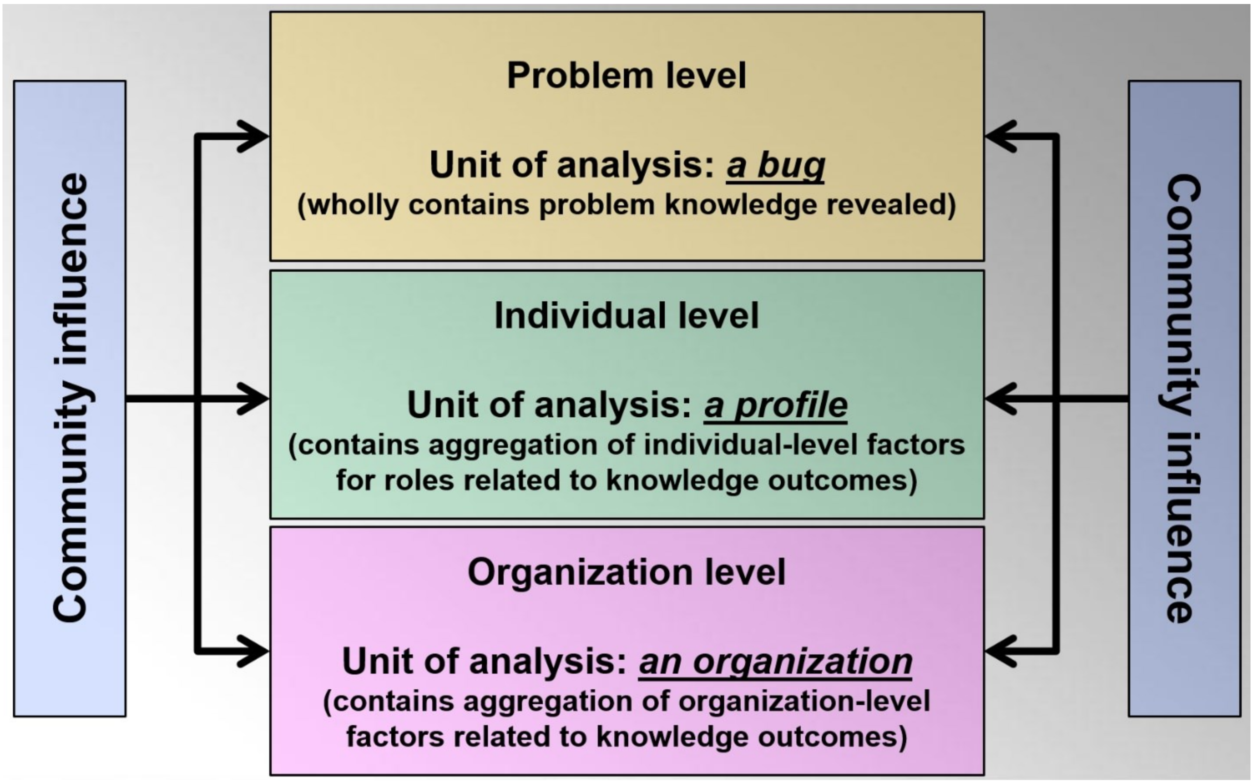

Figure 9: Levels and units of analysis

Operationalization

The independent and dependent variables in each of the hypotheses were operationalized to representations in the data according to level of analysis. Separating the operationalizations by level of analysis ensured that the locus of influence of each contingent factor could be clearly identified. Cross-level nesting effects were handled analytically rather than operationally in order to maintain consistent and concise definitions for each factor.

Problem level operationalization: Dependent variables

At the problem level, the dependent outcome of interest, solution knowledge emergence, was operationalized using seven types of measurement derived from the literature that were measured or calculated in the database. Each of these measures is described in the literature as a desired outcome of knowledge revealing strategies—that is to say that factors that improve these outcomes are of strategic relevance to organisations participating in open source meta-organisations, as per the theoretical framework of this study.

The first measure of the dependent outcome of interest, the most commonly reported in the open source literature and the most frequent focus of research (c.f. Anvik, Hiew, & Murphy, 2006; Antoniol, et al., 2008; Baysal, et al., 2013), is whether or not a focal bug was fixed. The concept of "fixed" is operationalized as equivalent to the emergence of solution knowledge from the meta-organisation—the primarily desired outcome of organisations using knowledge revealing strategies. At the problem level of analysis, each bug unit will eventually have an outcome of "fixed" or "not fixed".

The concept of the outcome "fixed / not fixed" is represented in the database with two variables, "status" and "resolution". Because the knowledge production process takes place over time, the outcome necessarily implies a window of observation for the determination of outcome. For example, any bug that is not yet fixed at a given time may be fixed at a future time. This ongoing process was taken into account by a classification process that organizes the bugs in the database from the beginning in 1998 to the time of the last entry in the database at the end of 2012. The goal was to only consider bugs that have reached an end point to ensure that type 1 errors are minimized. The "status" and "resolution" variables in the database denote the progress of the bug unit through the knowledge production process, known as bug life cycle, as depicted in Figure 10.

Bugs that were at a "status" stage in the knowledge flow depicted in green in Figure 10 were considered at an "end point" for classification, whereas bugs in any other stage were considered "pending" and excluded from consideration for the purpose of this variable. While in theory a bug at the stage “resolved” should eventually move on to “verified” and then “closed” stages before its “final” state is reached, in practice, examination of the data revealed that a very large number of bugs in the database remained permanently at “resolved” status as their final state.

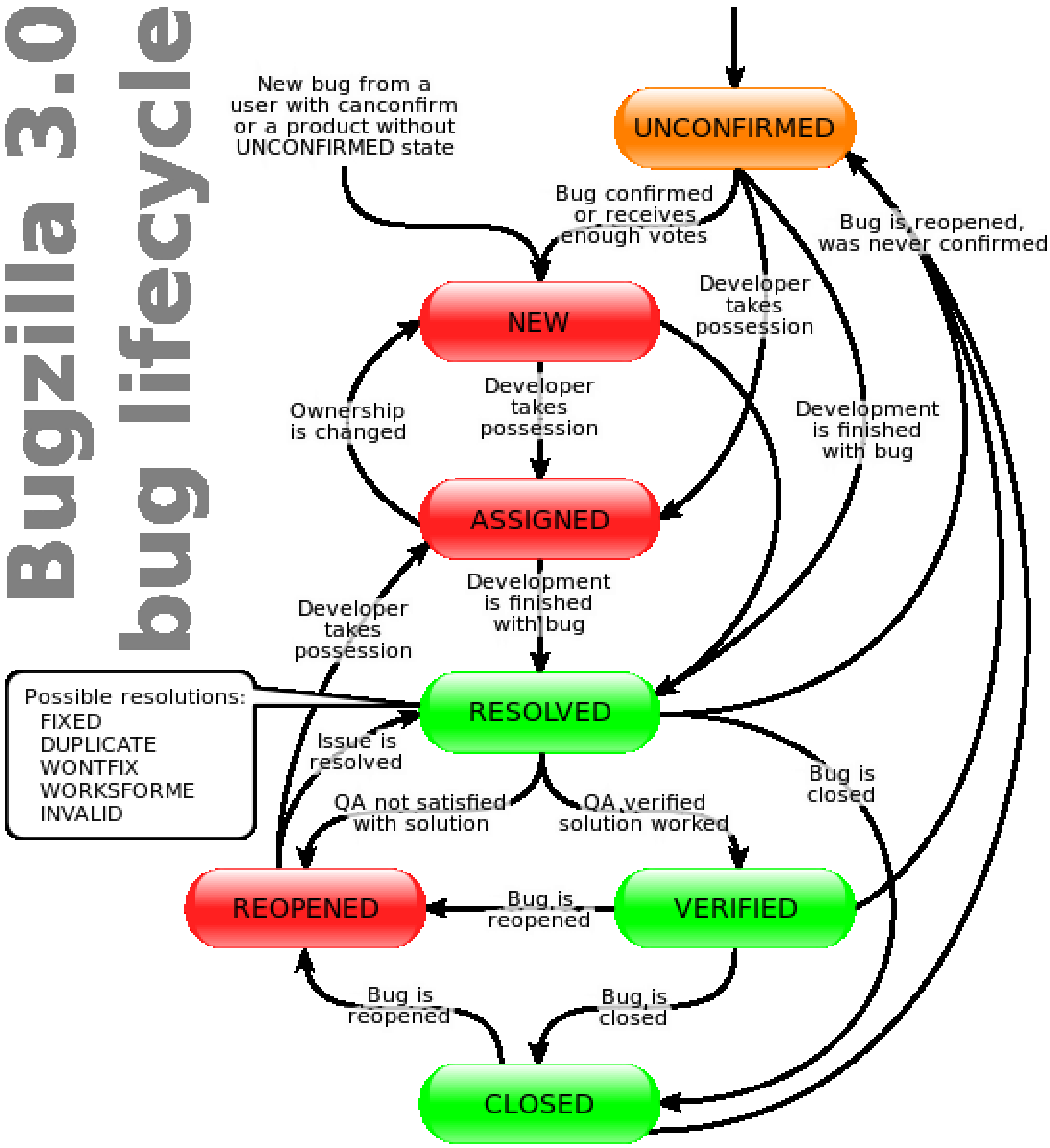

Figure 10: Knowledge flow at problem level of analysis

Once at an “end point” status, bugs were then classified based on the nature of the “resolution” at the conclusion of the knowledge flow. Of the seven possible “resolution” types in the database, only “fixed” was selected to mean emergence of solution knowledge. The remaining “resolution” types, namely “invalid”, “will not fix”, “duplicate”, “works for me”, “expired”, and “incomplete”, each of which representing a reason for which solution knowledge did not emerge, resulted in a bug being classified as “not fixed”. The result of this classification process was a single logical outcome variable for each of the retained bugs after the exclusion of those with “pending” classification. Table 3 summarizes the mapping of the status and resolution variables to the resulting solution knowledge emergence outcome measurement variable at the problem level of analysis.

Status	Resolution	Classification
UNCONFIRMED	ALL	PENDING
NEW	ALL	PENDING
ASSIGNED	ALL	PENDING
REOPENED	ALL	PENDING
CLOSED	FIXED	FIXED
CLOSED	INVALID	NOT FIXED
CLOSED	WONTFIX	NOT FIXED
CLOSED	DUPLICATE	NOT FIXED
CLOSED	WORKSFORME	NOT FIXED
CLOSED	EXPIRED	NOT FIXED
CLOSED	INCOMPLETE	NOT FIXED
RESOLVED	FIXED	FIXED
RESOLVED	INVALID	NOT FIXED
RESOLVED	WONTFIX	NOT FIXED
RESOLVED	DUPLICATE	NOT FIXED
RESOLVED	WORKSFORME	NOT FIXED
RESOLVED	EXPIRED	NOT FIXED
RESOLVED	INCOMPLETE	NOT FIXED
RESOLVED	MOVED	NOT FIXED
VERIFIED	FIXED	FIXED
VERIFIED	INVALID	NOT FIXED
VERIFIED	WONTFIX	NOT FIXED
VERIFIED	DUPLICATE	NOT FIXED
VERIFIED	WORKSFORME	NOT FIXED
VERIFIED	EXPIRED	NOT FIXED
VERIFIED	INCOMPLETE	NOT FIXED
VERIFIED	MOVED	NOT FIXED

Table 3: Classification of solution knowledge emergence outcome measurement

The second measure of the dependent outcome of interest at the problem level was operationalized to account for the distinction in the literature of cases where a bug is resolved with and without a software patch being issued (c.f. Antoniol, et al., 2008). Bugs fixed with patches are sometimes considered to have more value and are treated as conceptually distinct to bugs fixed without patches. While a detailed comparison of bug fixes with and without patches is beyond the scope of the present study, the fundamental distinction comes down to the tangibility of the emergent solution knowledge and its representation as software code (patch) or as something else (no patch).

For this measure, initial classification of the bugs at the knowledge level of analysis was conducted in the same manner as the first measure, using the “status” and “resolution” variables. Subsequently, a third logical variable “patch” allows further refinement of the classification into those bugs that “fixed with patch” and those that were not. The decision was made to create a logical outcome variable “fixed with patch” or “not fixed with patch” rather than a trinary variable “fixed with patch, fixed without patch, and not fixed” to maintain the conceptual distinctions between this measure and the first measure in a manner consistent with the literature’s conceptual categories. At the analysis stage, this choice enabled more powerful logistical analysis of two logical variables rather than a single less powerful multivariate analysis of a single trinary variable. The “fixed with patch” variable was measured independently of the “fixed” variable in a manner that controlled for the obvious collinearities given that the second variable depends on the first being true. Preliminary analysis using both of these approaches revealed the former to be provide more useful insights, validating this choice of operationalization.

The third measure of dependent outcome of interest at the problem level operationalizes the directness of the knowledge flow. The literature suggests that it is preferable if a bug proceeds as directly as possible through the knowledge production process (c.f. Koponen, 2006), suggesting that this directness is a useful outcome that is related to but conceptually distinct from the emergence of the solution knowledge itself. As depicted in Figure 10, there are numerous ways the knowledge production process can loop back upon itself. Three variables were selected to designate a loop in the knowledge production process: whether a bug was "reopened", whether a bug was "reassigned", and whether a bug was "even confirmed". While the third variable was its own field in the database, the "reopened" and "reassigned" variables were calculated by cross referencing the "activity" table to the "bugs" table in the database and separating those bugs that ever had their status set to "reopened" and those who had more than one profile set as "assigned_to" from those that had not over the course of their progress through the knowledge flow. The choice was made to operationalize these variables as a logical status rather than count of number of times a bug was reopened or reassigned. Preliminary analysis revealed that bugs reopened or reassigned more than once where extremely rare and of insufficient statistical power to be meaningfully conceptually different.

The fourth measure of the dependent outcome of interest at the problem level captured the argument in the literature that faster resolutions were preferable to slower resolutions (c.f. Huntley, 2003; Dalle, et al., 2008; Ahmed & Gokhale, 2009; Au et al., 2009). It may seem to be an obvious implication that solution knowledge that emerges faster is of more immediate use to the organisation that submitted the corresponding problem knowledge. However, given that the first measure of the dependent variable already considers the emergence of problem knowledge proper, this measure was operationalized to exclusively consider the amount of time until any "end point" resolution was reached in the knowledge production process. This separation keeps the measures conceptually distinct and isolates the effects of each one independently.

The amount of time until resolution was calculated in two ways. First, those bugs with a classification of "pending" were excluded as they have no measurable time to resolution. This step was followed by subtracting the creation time stamp of each bug from the time stamp of the last activity recorded in the activity table cross-referenced with each bug. The result is each bug's status assigned to one of the green end point statuses depicted in Figure 10. The choice was made to measure the "time to resolution" as the last "resolution" rather than possible earlier resolutions that were later deemed insufficient (by, for example, the solution knowledge verifier). This operationalization most closely matches the factor described in the literature which accounts for the total time until the knowledge production process ceases entirely, regardless of outcome. While the time covered may include one or more reopenings or reassignments, which are examined in their own logical measure variables, the nature of the present variable being an amount of time rather than a state ensures that these variables remain conceptually distinct. Preliminary analysis using different ways of measuring "time to resolution" revealed the present approach to be the most reliable and consistent.

The fifth measure of the dependent outcome of interest at the problem level of analysis was time until assignment. This measure was operationalized to account for reports in the literature that the more quickly a solution knowledge producer is identified and tasked (often self-tasked) with resolving the problem knowledge, the better the outcome for the organisations who submitted the problem knowledge (c.f. Baysal, et al., 2013). Time to assignment was measured in a manner similar to time to resolution by subtracting the creation time stamp of the bug from the first time the "assigned_to" field was populated as tracked in the activity table cross-referenced with the bugs table. The choice was made to stop the counting of time at the first assignment rather than the last assignment as this operationalization most closely matches the definition of the factor in the literature. This choice also keeps the time to assignment variable conceptually distinct from the development time variable discussed below. Given that only a subset of bugs is ever assigned at all, it was necessary to exclude all bugs that were never assigned as they have no meaningful value for this time variable.

The sixth measure of the dependent outcome of interest was development time. The literature argued that faster emergence of solution knowledge is of superior benefit to the organisation who submitted the problem knowledge than slower emergence (c.f. Dalle, et al., 2008; Ahmed & Gokhale, 2009; Baysal, et al., 2013; Alexy, George, & Salter, 2013). This variable separates the amount of time from the identification of a solution knowledge producer to the final resolution for the bug, regardless of the final outcome. Given that the action "development" can take on many forms and the outcome measurement is handled by the first measurement, the choice was made to not separate development time based on outcome to ensure this variable is conceptually distinct. This variable also accounts for the amount of time between first assignment, which was the end time point of the previous variable, and the final identification of the solution knowledge producer in the case that multiple people are assigned over time. This process is often conceptualized as part of the factor termed "development" in the literature making this operationalization the closest equivalent. In addition to the constraint to only bugs that are ever assigned, as per the previous variable, this measure also excludes bugs that have not yet reached an outcome ("pending") as they have no meaningful time value for this measure.

The seventh measure of the dependent outcome of interest examines the times to resolution, assignment, and end of development based on quantile-identified thresholds. This measure operationalizes the notion in the literature that the time measures have an endogeneity to the open source ecosystem itself based on its processes (c.f., Giger, Pinzger & Gall, 2010, Baysal, et al., 2013). Therefore, in order to compare the time-based outcome within this single ecosystem, it is necessary to compare each bug's time relative to the other bugs in the ecosystem, effectively controlling for ecosystem endogeneity. These relative measures complement the absolute measures for each of the time-related outcomes, triangulating the operationalizations to improve the validity of the measures.

In order to determine the appropriate thresholds for comparison, the frequency of occurrence of the duration of each time variable, in days, was graphed to identify inflection points in each variable. The result was a non-linear "s-type" curve that denoted inflection points around certain time values. These values were used to create logical variables for which "bucket" quantile in which each bug fell for each time variable. Each bucket was given a label that reflects its "speed" relative to other bugs for that time variable. The buckets are not numerically even but rather reflect the logarithmic equivalence of the non-linear s-curve shape of the quantile distribution, making them a better match for "relative" comparison by focusing on differences at the scale of the data. Table 4 summarizes the inflection point thresholds used to create each of the variables for these measures. Using these thresholds, 21 logical variables were created that could each independently be measured as the outcome variable of logistic regression models with easily interpretable and intuitively understandable results.

Relative speed category	Time to resolution thresholds (days)	Time to first assignment thresholds (days)	Time for development thresholds (days)
Extremely fast	X <= 0.4	X <= 0.05	X <= 0.5
Very fast	0.4 < X <= 1.0	0.05 < X <= 0.4	0.5 < X <= 2.0
Fast	1.0 < X <= 8.0	0.4 < X <= 2.0	2.0 < X <= 10
Average	8.0 < X <= 216	2.0 < X <= 20	10 < X <= 60
Slow	216 < X <= 300	20 < X <= 50	60 < X <= 180
Very slow	300 < X <= 800	50 < X <= 160	180 < X <= 500
Extremely slow	800 < X	160 < X	500 < X

Table 4: Thresholds for comparative time measure variables:

Taken collectively, the operationalizations of these seven conceptual measures provide a detailed triangulation of the concept of solution knowledge emergence, incorporating a broad range of definitions described in the literature. Figure 11 summarises the operationalizations of the measures of the dependent variable of interest, solution knowledge emergence, at the problem level.

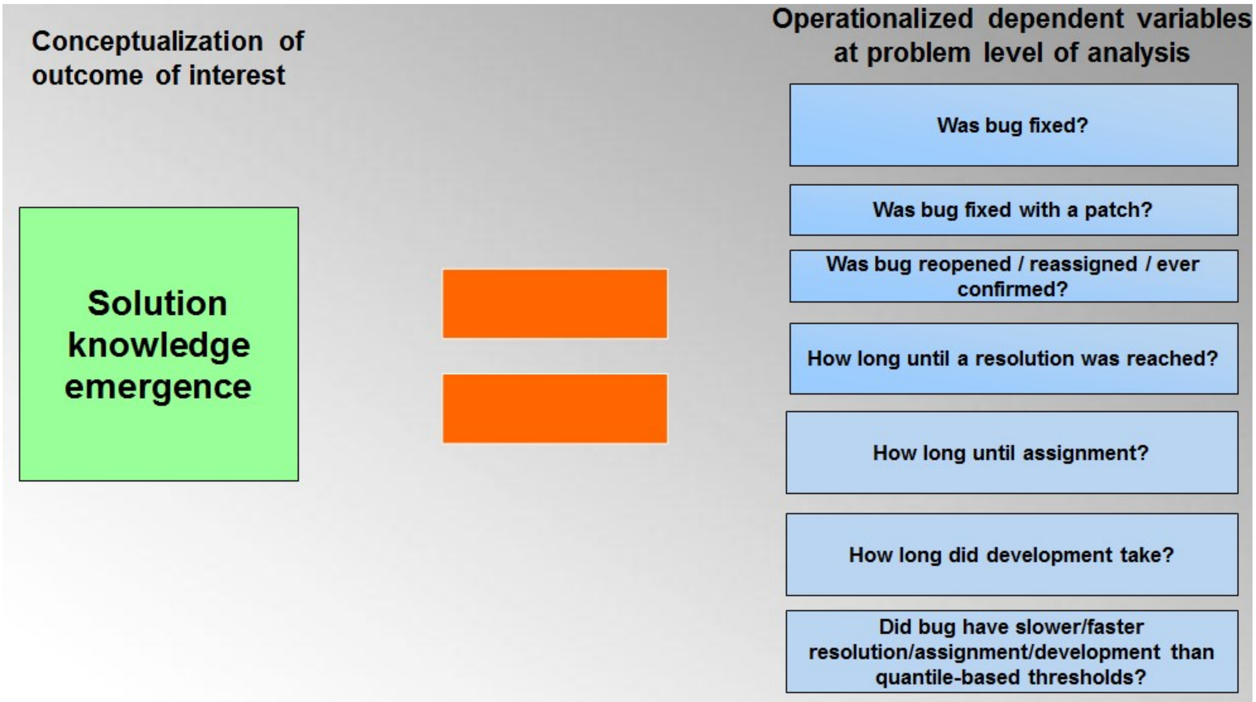

Figure 11: Operationalizations of measures of dependent variable of interest at problem level

Problem level operationalization: Independent variables

The independent variables were operationalized at the problem level in line with each of the six hypotheses that were formulated for the conceptualizations of the antecedents of interest. Each operationalization can be conceptualised as a distinct measure used to triangulate the overall conceptualisation derived from the KBV and open source literatures as well as a testable sub-hypothesis with each of the above-discussed measures as antecedent independent variables.

Absorptive capacity

The first antecedent of interest is absorptive capacity. It was triangulated with six measures derived from the open source and KBV literatures. In the hypothesis formulation, absorptive capacity is theorized to be positively correlated with solution knowledge emergence, i.e., the more absorptive capacity a given actor or the open source meta-organisation as a whole has, the better solution knowledge emergence as the actors are able to recognize, assimilate, and apply knowledge to solving problems. By contrast, many of the operationalised measures of absorptive capacity as operationalized are negative, i.e., they act as a load on absorptive capacity, reducing the remaining capacity that can be applied to knowledge recognition, assimilation, and application to problem solving. As such, these measures should be thought of as the factors that change the amount of absorptive capacity rather than representing an absolute absorptive capacity value directly.

The first measure was number of unresolved problems at the time new problem knowledge is revealed to the open source meta-organisation (c.f. Anvik, Hiew, & Murphy, 2006). The relationship between this independent variable and solution knowledge emergence is hypothesized to be negative as the greater the number of unresolved problems pending resolution, the worse the tendency of solution knowledge emergence for a subsequently revealed set of problem knowledge. This variable was calculated by examining the cross section of all bugs in the database at each time stamp of creation of a new bug and counting the number of bugs that had the status "pending" at each of those times. This variable matches well to the concept of juggling many balls at once and hence not being able to take on any new knowledge-based tasks as described in the KBV literature (c.f. Cohen & Levinthal, 1990; Zahra & George, 2002; Jansen, Van Den Bosch, & Volberda, 2005; Todorova & Durisin, 2007).

The second measure of absorptive capacity consists of variables that organize the number of unresolved problems according to the same platform, operating system, classification, product, or component as the focal problem at the time its problem knowledge was revealed to the meta-organisation. The variables were created in a manner similar to the previous measure. For each bug, at the cross-section of its creation time stamp, the subset of other bugs with the same platform in the database were reviewed at that time in order to count the number of bugs that had the status “pending”. The process was repeated for operating system, classification, product, and component separately, resulting in 5 variables that reach reflect a different scope of absorptive capacity. This measure allows more precise identification of the locus of absorptive capacity challenges. This measure also complements the previous measure in a manner that reflects the suggestion in the KBV literature that absorptive capacity can be localized at different levels in different categories of the knowledge production process (Pisano, 1994; von Hippel, 1994; Lane & Lubatkin, 1998; Zahra & George, 2002; Schmidt, 2010; Volberda, Foss, & Lyles, 2010; Spithoven, Clarysse, & Knockaert, 2011).

The third measure of absorptive capacity consists of variables that organize the number of newly revealed problems proximal to each focal problem’s revelation to the meta-organisation. This measure reflects the notion in the literature that absorptive capacity can vary based on points in time in the knowledge production process independently of different levels of categories of knowledge (c.f. Fershtman & Gandal, 2004; Francalanci & Merlo, 2008; Giger, Pinzger & Gall, 2010). The variables were created in a manner similar to the previous measures. For each bug, at the cross-section of its creation time stamp, all other bugs whose creation time stamp was within various intervals of time prior to the focal bug’s creation time stamp were summed. The time intervals used to represent “created in the past X amount of time” were 1 day, 3 days, 7 days, 30 days, 90 days, 180 days, 1 year, and 2 years prior. These intervals were selected based on manual experimentation with a wide variety of ranges and most closely map the data to the short term, medium term, and long-term inflections in absorptive capacity described in the literature, providing a triangulation of different varieties of time-based measures to more comprehensively measure absorptive capacity.

The fourth measure of absorptive capacity consists of variables that organize the number of newly solved problems proximal to each focal problem's revelation to the meta-organisation. This measure complements the previous measure by reflecting the notion in the literature that absorptive capacity can vary based on both the number of newly identified problems and the number of newly resolved problems over time (c.f. Hooimeijer & Weimer, 2007; Shihab, et al., 2010). The former identifies potential draws for absorptive capacity instead of the focal problem whereas the latter, the present measure, identifies actual draws of absorptive capacity in the form of work done on other problems recently. As with the previous measure, experimentation revealed that the ideal intervals for the variables of this measure used to represent "resolved in the past X amount of time" were 1 day, 3 days, 7 days, 30 days, 90 days, 180 days, 1 year, and 2 years prior. Taken together, these measures triangulate the time-based draws of absorptive capacity that are theorised to affect solution knowledge emergence.

The fifth measure of absorptive capacity consists of variables that consider the timing of the revelation of the problem knowledge to the meta-organisation relative to institutional schedules. Using the time stamp of the creation of each bug in the database, the timing was classified into variables for year, month, day of month, and weekday. Each of these variables provides a measure of timing that represents unobserved temporal scheduling factors in the meta-organisation as per the literature. For example, in many organisations, the absorptive capacity is higher at the beginning of the work week than at the end of the work week when organisation members are tired and in need of a break over the weekend. A similar situation takes place over the course of months, particularly around socially scheduled holidays, tax season, or regularly scheduled organisation deliverable periods. These variables aim to capture those factors' influence on solution knowledge emergence.

The sixth and final measure of absorptive capacity considers the amount of time that it takes for a given problem to reach a resolved status. This measure is the same as the fourth measure of the dependent outcome variable of interest, alternatively considered as an antecedent. Clearly, this measure cannot be both dependent and independent variable at the same time. As a result, it is only assessed when it can be theoretically separated from itself and evaluated with other orthogonal measures of solution knowledge emergence. For example, the amount of time a given bug was open is orthogonal to the eventual resolution status of "fixed" or "not fixed". In this case, it is appropriate to consider the former as IV and the latter as DV without confounding the measures. Considering this measure as both IV and DV separately also serves as a verification of the independence of the different measures of solution knowledge emergence, reducing issues of cross-correlation confounding results. Its theoretical importance as potential antecedent, in addition to outcome, reflects the notion in the literature that the knowledge production process might become "stale" and absorptive capacity might drop as a result as time progresses and relevant knowledge loses value (c.f. Au et al., 2009; Giger, Pinzger & Gall, 2010).

Figure 12 summarises the operationalizations of the measures of absorptive capacity at the problem level as well as their theorised direction of influence as independent variables on the dependent outcome of interest solution knowledge emergence.

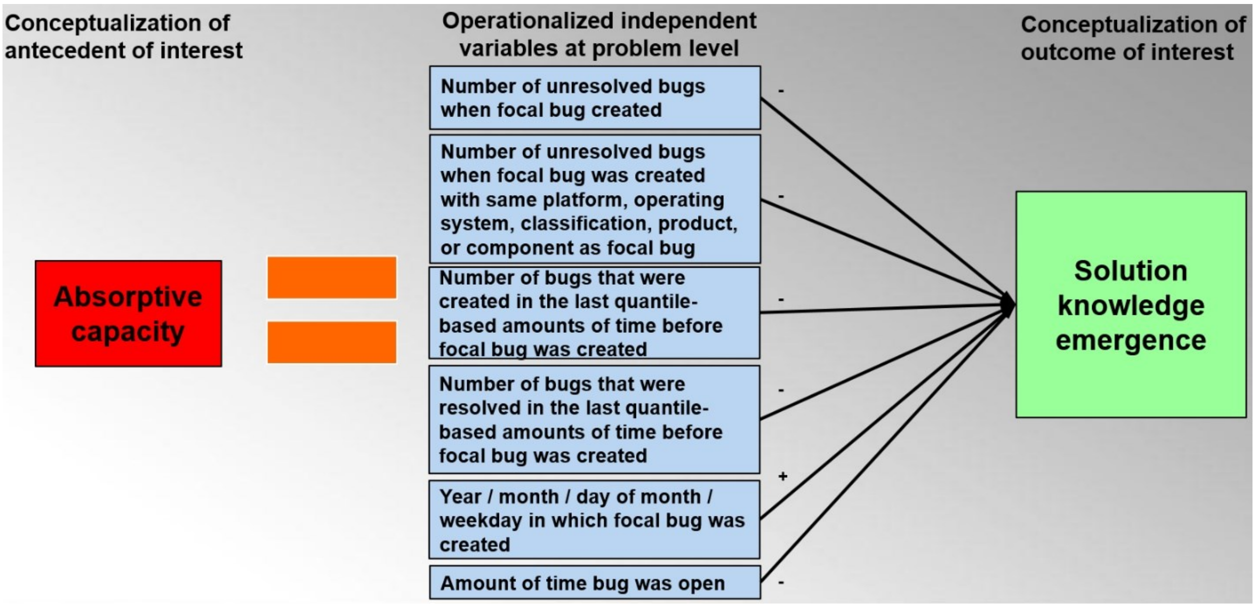

Figure 12: Operationalizations of measures of absorptive capacity at problem level

Codifiability

The second antecedent of interest is codifiability. It was triangulated with seven measures derived from the open source and KBV literatures. The first measure was the length of the title and description of each of the problem knowledge sets. These variables were calculated by counting the number of characters in the title and description of each bug. Preliminary analysis on the quantile distribution of the range of description lengths suggested a non-linear distribution that could not be readily transformed to linear with conventional transformations. Instead, the distribution revealed that there were two major classes of “description lengths” whose inflection point was at approximately 10,000 characters in length. As a result, rather than a non-linear pure length variable, a logical “shorter than” and “longer than” 10,000 characters variable was created. The use of a logical variable maximizes the power available for detecting large-scale effects. A more nuanced measure using an expanded database with a more evenly distributed range of description lengths would be a useful future extension beyond the scope of the present research.

These measures reflect the notion in the literature that codifiability is a function of complexity and overall length is a basic measure of complexity (c.f. Bettenburg, et al., 2008; Ahmed & Gokhale, 2009; Guo, et al., 2010). It is theorized that excessively long titles and descriptions have a negative effect on solution knowledge emergence because they reflect difficulty of codifiability, resulting in hindered knowledge flow within the meta-organisation (c.f. Kogut & Zander, 1992; Cowan, 2001). Whereas it would be ideal to examine the potential for a u-shaped relationship, where both too long and too short titles and descriptions were theorized to be negative, as per the theory in the extant codifiability literature, the data are not conducive to this sort of analysis. As a result, the choice was made to examine the effects too much complexity rather than too little. Further research with data more suited to quadratic analysis may wish to examine this relationship in more detail.

The second measure of codifiability was the readability of the description. This measure reflects the notion in the literature that many descriptions of problems suffer from readability problems due to poor grammar, poor choice of words, sub-optimal punctuation, and so on (Canfora & Cerulo, 2006; Guo, et al., 2010; Masmoudi, 2012). This issue is particularly prevalent in meta-organisations that have a technical focus, as in the present case, as some problem descriptions can be overly technical, reducing their codifiability for those not expert in the narrow technical field required to understand and communicate the problem clearly. As a result, description readability is expected to positively relate to solution knowledge emergence. The variable representing "readability" was calculated using the Flesch Reading Ease

Readability Formula (Flesch, 1948; Kincaid, Aagard, O'Hara, & Cottrell, 1981). This particular formula was selected over other available formulae as it has successfully been used as a readability measure for technical manuals making it a better fit to a technically-focused meta-organisation than readability measures more applicable to the realms of literature and education (Smith & Kincaid 1970). The application of the Flesch formula to calculate the readability of each bug in the database was done using the koRpus R package (Michalke, 2012-2017) that translates the formula to R (R Foundation, 2017) code for statistical analysis resulting in a single variable measure suitable for hypothesis testing.

The third measure of codifiability was the presence and type of attachments in the problem knowledge revealed to the meta-organisation. Attachments are optional appendices to problem description that provide additional details and contextualisation related to the problem. The literature suggests that contextual and corroboratory knowledge are factors in the codifiability of problem knowledge, suggesting their existence and nature may improve solution knowledge emergence (Bettenburg, et al., 2008; Guo, et al., 2010; Guo, et al., 2011). In theory, attachments can be of many different types including textual, source code, images, logical code, audio, video, and structural models. In practice, in this particular open source meta-organisation, the image attachment type is used disproportionally as compared to other attachment types. As a result, preliminary analysis revealed that there was insufficient power to examine all the attachment types as a single categorical variable. Instead, two logical variables were created for this measure, with the first representing the presence or absence of an attachment and the second indicating whether or not the attachment type is "image". This choice was justified in that it represents a conservative measure of the effects of attachment type given the low power of the full range of types. While it may result in missing smaller effect related to the specific other types, the goal in this study is to capture large scale effects in a thorough analysis. Future research with an extended database could usefully consider the effects of non-image attachment types, if any.

The fourth measure of codifiability is the similarity of the title and description of each problem knowledge from the set of problem knowledge reveals that resulted in a “fixed” resolution. This measure reflects the notion in the literature that some elements of codifiability are inherent to that which is being codified and are best described by comparison to desired categories (Sandusky, Gasser, & Ripoche, 2004; Zimmermann, et al., 2010). In this case, the outcome of interest being solution knowledge emergence, a prototypical model of the title and description of each problem knowledge reveal that led to solution knowledge emergence was created and each individual bug’s title and description were compared to that prototype to determine similarity and difference. The prototypes for “titles and descriptions of fixed bugs” were created using a concept derived in the field of linguistics known as an “n-gram profile” (Armstrong-Warwick, Thompson, McKelvie, & Pettitpierre, 1994, Hornik, Rauch, Buchta, & Feinerer, 2013).

“An n-gram is an n-character slice of a longer string” (Cavnar & Trenkle, 1994: 2). Examining the frequency of these slices and their proximity to spaces, indicating the boundaries of a word, allow the creation of probability models of similarity. Initially, these models were used to deal with “noisy” transmission channels to correct loss of data during transmission of textual information (Cavnar & Trenkle, 1994). More recently these models are used for classification and prediction, as in the present case (Hornik, et al., 2013). As Cavnar & Trenkle (1994: 2) explain, “If we count n-grams that are common to two [or more] strings, we get a measure of their similarity that is resistant to a wide variety of textual errors [or non-error similarities such as synonyms]”. In particular, the frequency of occurrence of any given set of n-grams in a document can be modelled using ranking algorithms, “implying that if we are comparing documents from the same category, they should have similar n-gram frequency distributions” (Cavnar & Trenkle, 1994: 3). The present measures were created using the Textcat R package (Hornik, et al., 2013) that applies these principles to build a text categorisation system. The top part of Figure 13 depicts the knowledge categorisation process. In the present case, the category sample (only one category is used in the present measure) consists of all bugs with the status “fixed”, the desired outcome. The “new document” is the focal bug with the “new” title and description. Profiles of each are generated to calculate the frequency distributions of n-grams. The distance between the n-gram statistical distributions of the category of “fixed” bugs and new bug is then calculated to create a “distance” measure.

That “distance” measure can be calculated in different ways. Based on preliminary analysis with the 7 most common measures, it was found that the Kullback-Leibler Jeffreys divergence measure for multivariate skew-normal distributions (Kullback & Leibler, 1951; Contreras-Reyes & Arellano-Valle, 2012) and the n-gram ranks comparison measure (Hornik, et al., 2013) produced variables that were most readily comparable between bugs on a linear scale suitable for hypothesis testing. The KLJ measure was chosen for the present “distance” measure. The ranks comparison measure was used in the following measure, discussed in the next paragraph.

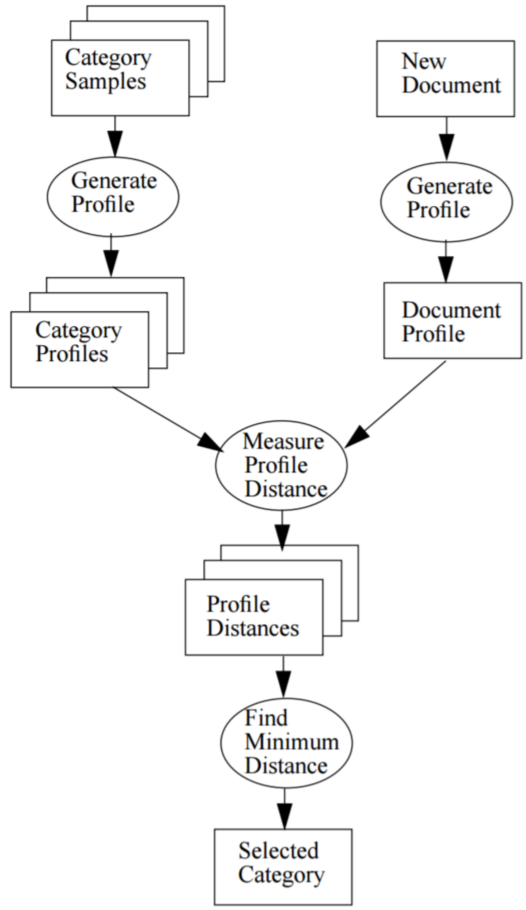

Figure 13: N-gram based text categorisation knowledge flow (Adapted from Cavnar & Trenkle, 1994)

The fifth measure of codifiability is the outcome of automatic classification based on n-gram profile comparison to the categories of "fixed" and "not_fixed" for previously revealed problem knowledge. This measure complements the previous measure by operationalizing the bottom portion of Figure 13 by applying algorithms to figure out the minimal distance between each bug and the n-gram profiles created from the categories of all bugs previously classified as "fixed" or "not fixed". The result is a logical variable of "predicted fixed" or "predicted not fixed".

As with the previous measure, the top portion of the knowledge flow uses samples of previously resolved bugs to generate profiles. In the present measure, as opposed to the previous measure, this time two profile categories are created using separate sets of samples for bugs previously resolved as "fixed" and as "not fixed". As a result, two profile distances are calculated for each bug's title and description n-gram profiles: one distance from the set of "fixed" profiles and one distance from the set of "not fixed" profiles.

The algorithm simply chooses the lower distance to "automatically" classify each bug. As with the previous measure, different "distance" algorithms can be used. Whereas the intent in the previous measure was to examine the relationship between the actual distance measures and the outcome of interest, the intent in the present measure is to examine the relationship between the relative distance from two measures and the outcome of interest, trading relative distance from one category for net difference of distance from two categories. As such, a different measure of the distance, the Cavnar & Trenkle aggregate absolute difference of ranks of the combined n-grams in the two profiles measure (Cavnar & Trenkle, 1994) was chosen as it focuses more specifically on logical classification rather than an absolute numeric measure as in the previous measure's case.

Along with the previous measure, this measure reflects the notion in the literature that codifiability partially consists of readily identifiable patterns (c.f. Anvik, Hiew, & Murphy, 2006; Breu, et al., 2010). It is therefore hypothesized that the identification of these patterns and their ability to be classified algorithmically positively correlates with solution knowledge emergence. The use of two distinct n-gram profile distance measures further triangulates these operationalizations of codifiability to strengthen the overall validity of the measures.

The sixth measure of codifiability is the redundancy of the submitted problem knowledge. One of the variables in the database designates that a knowledge actor has flagged the submitted problem knowledge as redundant to previously submitted problem knowledge in a dyadic manner. Each bug can potentially be a duplicate and can potentially have duplicates. In the former case, being a duplicate suggests that the problem knowledge is not new and may have previously led to solution knowledge emergence. As a result, new solution knowledge is unlikely to emerge in a manner associated with the duplicate problem knowledge reveal. Instead, in the latter case, having a duplicate suggests that the focal duplicated problem knowledge has been codified anew, possibly in a more detailed way, increasing the likelihood that the duplicated problem knowledge will lead to solution knowledge emergence. The two variables, "is a duplicate" and "has a duplicate" triangulate both sides of the dyadic relationship implied by this measure, following the open source literature (Sandusky, Gasser, & Ripoche, 2004).

The seventh and final measure of codifiability is the number and length of comments appended to the problem knowledge revealed to the meta-organisation. Comments are conceptualised as a form of emergent problem knowledge that complements the initial problem knowledge by providing more details, answering questions by knowledge actors, and linking the problem knowledge to a reproducible context. The number and length of comments represent the available additional emergent problem knowledge available for codification and are therefore hypothesized to relate to solution knowledge emergence in a u-shape, in a manner similar to title and description lengths. Too many comments and/or comments that are too long may represent a degree of complexity that begins to compromise the codifiability.

Preliminary analysis of the quantiles of the range of number of comments revealed that the number of comments were not linearly distributed across bugs. Standard transformations did not successfully induce linearity. Instead, an inflection point was apparent at more than 50 comments. As a result, rather than a non-linear pure count variable, a logical “fewer than” and “more than” 50 comments variable was created. As in the case of description length, the use of a logical variable maximizes the power available for detecting large-scale effects. A more nuanced measure using an expanded database with a more evenly distributed range of comment counts would be a useful future extension beyond the scope of the present research.

In the case of comment length, given that, unlike for a bug’s description, there are typically multiple comments, the problem-level measure that is most readily comparable across bugs is the mean comment length rather than the absolute sum of comment lengths which is highly non-linearly skewed and not readily comparable even with standard transformations. These two variables triangulate the measures of emergent problem knowledge, mapping to this component of codifiability as described in the literature (WeiB et al., 2007; Zhang, et al., 2012; Alexy, George, & Salter, 2013), enhancing the validity of the overall measure.

Figure 14 summarises the operationalizations of the measures of codifiability at the problem level as well as their theorised direction of influence as independent variables on the dependent outcome of interest solution knowledge emergence.

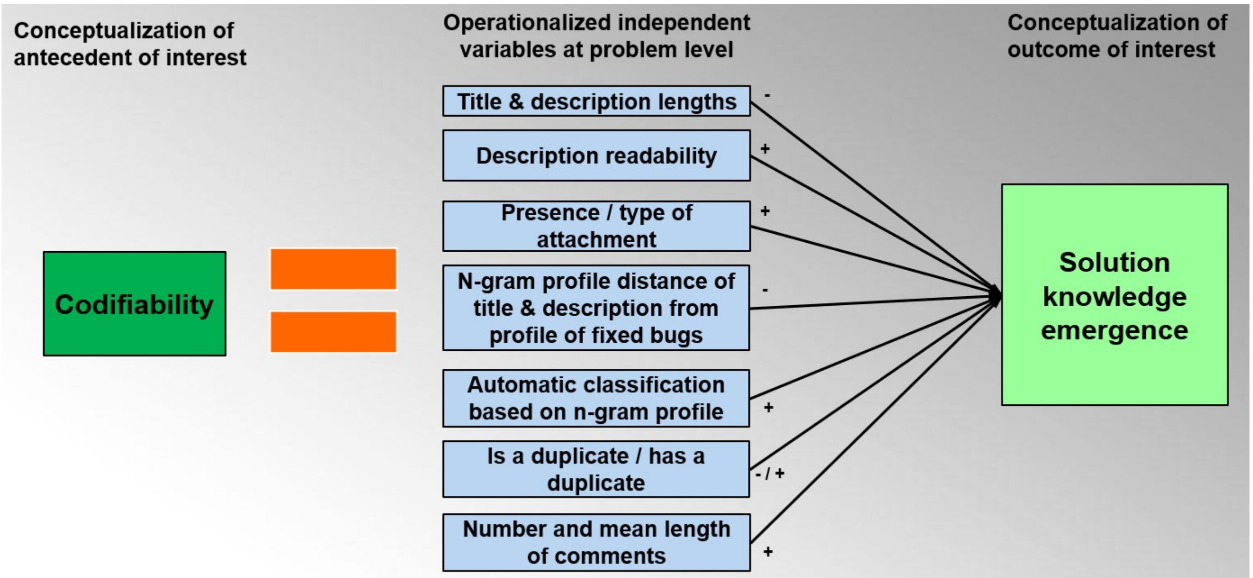

Figure 14: Operationalizations of measures of codifiability at problem level

Dominant knowledge paradigm

The third antecedent of interest is dominant knowledge paradigm. It was triangulated with five measures derived from the open source and KBV literatures (Panjer, 2007; Ahmed & Gokhale, 2009; Au et al., 2009; Bougie et al., 2010; Shihab, et al, 2010; Zhang, et al., 2012). The KBV literature suggests that certain knowledge paradigms, such as technical standards, will go through phases of popularity in organisations independent of their actual utility as relates to the specific nature of the knowledge (Grant, 1996a, 1996b; Szulanski, 1996). The present measures organize the problem knowledge into knowledge paradigms related to open source meta-organisations. The measures are used in the meta-organisation to identify solution knowledge producers whose skillset includes the knowledge paradigms related to the operating system of a focal piece of problem knowledge as knowledge is often unevenly distributed within a meta-organisation with only partial overlap amongst participants (Gulati, Puranam, & Tushman, 2012), making these measures a good representation of the theoretical concept in the literature.

The first measure of dominant knowledge paradigm was platform type. Computer platforms form of technical standard that represent particular architecture implementations of different possible arrangements of components that map to different engineering and computer science fundamental principles. In the early days of computing, platforms were often unique to the firm that developed a given computer. In the 1970s, a range of standard platforms emerged, led by IBM (cf. Breshnahan & Greenstein, 1999; Gawer & Cusumano, 2008), with dominant platforms waxing and waning and new platforms emerging over time. Given the longitudinal nature of the database used in this study, ranging from 1998 to end of 2012, it is unsurprising that a range of platforms are represented in the database, which the present measure captures in a single categorical variable. Platform types include more well-known platforms such as “x86” and “x86 64-bit” which are used for a large portion of personal computers and servers, “PowerPC”, which was used for many Apple computers for many years, and “ARM”, which is used in many mobile computing devices. Lesser well-known platforms including “DEC”, “SGI”, “Sun”, “HP”, and “Scale” which are used primarily for specialised server devices are also represented. As the measure is meant to organize the nature of the problem knowledge, it also includes types for “all” and “other” in the case that the problem knowledge relates to more than one platform or platforms so rare that they don’t have their own designation in the database. In the case of dummy variable regression models, the “all” category is held as the reference category that determines the relative dummy variable coefficient directions.

The second measure of dominant knowledge paradigm was classification type. The field “classification” in the database attempts to organize the problem knowledge according to the knowledge creation priorities of the open source meta-organisation. In the Mozilla meta-organisation, 5 classifications are used: “client software”, “server software”, “component”, "other" and "graveyard". Client software refers to development priorities related to software that will be used by end users such as the Firefox web browser. Server software refers to software that is primary run on enterprise servers used by more than one user at the same time for broader organisational development purposes, such as the Bugzilla bug tracking software that created the database used in this study. Components refer to pieces of computer software that are reused across different client and server software. For example, both Firefox and Bugzilla must present a user-interface. Their user interface rendering is done by a component known as Gecko which is part of both software products and has its own development priorities. The other classification is used for categorisation of problem knowledge that doesn't fit in any of these categories such as documentation, strategy, consumer outreach, standards development, support, and marketing. The last classification, graveyard, is used to denote obsolete knowledge creation priorities that are no longer of relevance to the meta-organisation's knowledge production activities going forward. Periodically, designated knowledge flows are "retired" and moved to the graveyard classification and retained there indefinitely for retrospective and post-mortem analysis. Such a retirement typically represents the end of the dominance of a given knowledge paradigm in the open source meta-organisation, making it particularly well suited for the present measure. In the dummy variable regression models, the classification "client software" is held as the reference category to determine relative dummy variable coefficient directions.

The third measure of dominant knowledge paradigm was operating system type. Operating systems are a software layer of abstraction that interacts with the hardware platform thereby creating a programming standard for the development of applications that is independent of hardware specifics. Hundreds of operating systems have been developed since the invention of computers. This categorical measure has categories designating the 48 most common operating systems in use between 1998 and 2012, including most versions of Microsoft Windows (e.g., 95, 98, ME, 2000, NT, XP, 2003, 2008, Vista, 7, 8), most MacOS versions (e.g., 7.5-9.x and OS X), Android and iOS smartphone operating systems, Linux, FreeBSD, and so on. As in the case of the classification measure, the 49th category of the operating system type is “all”, indicating problem knowledge that applies to more than one operating system type, and the 50th category is “other”, referring to issues that apply to one of the less common operating system types not reflected by its own category in the variable. Given the large number of type categories in this variable, during analysis, in order to increase the statistical power of the measure, it was converted to a continuous numerical variable from 1 to 50 to represent the operating system type. This approach has been shown to be appropriate for statistical analysis when a variable has more than 7 categories as the use of a large number of dummy variables significantly increases computational complexity without yielding a significant variance in results (Rhemtulla, Brosseau-Liar, & Savalei, 2012).

The fourth measure of dominant knowledge paradigm is product type. Products are software applications that provide features to conduct computational tasks. Products can be both single user and multi-user in focus and generally focus on addressing a particular set of needs related to organisational tasks. There are 85 product types designated in the database, including “all” and “other” types similar to those used in the previous measures. Common products include the Firefox web browser, the Bugzilla bug tracking system, and the Thunderbird email client. “Product” types are also sometimes used in the meta-organisation to indicate non-software related knowledge production activities such as in the case of the types “documentation”, “Marketing”, “Finance”, “tech evangelism”, and “websites”. As with the previous measure, given the large number of type categories in this variable, during analysis it was converted to a continuous numerical variable from 1 to 85 to represent the product type and avoid the excessive use of dummy variables.

The fifth and final measure of dominant knowledge paradigm is component type. Components are typically reusable portions of software code that have applicability to multiple products. There are 1253 designated component types including "all" and "other" used for problem knowledge categorisation. Examples of general categories include "themes", "localisation", "view source", "extensions", and more specific categories include languages such as "Estonian" and "Catalan", and "dictionaries" and "sidebars" used in many different products developed in the meta-organisation. As with the previous measures, this categorical type variable was converted to a continuous numerical variable from 1 to 1253 during analysis.

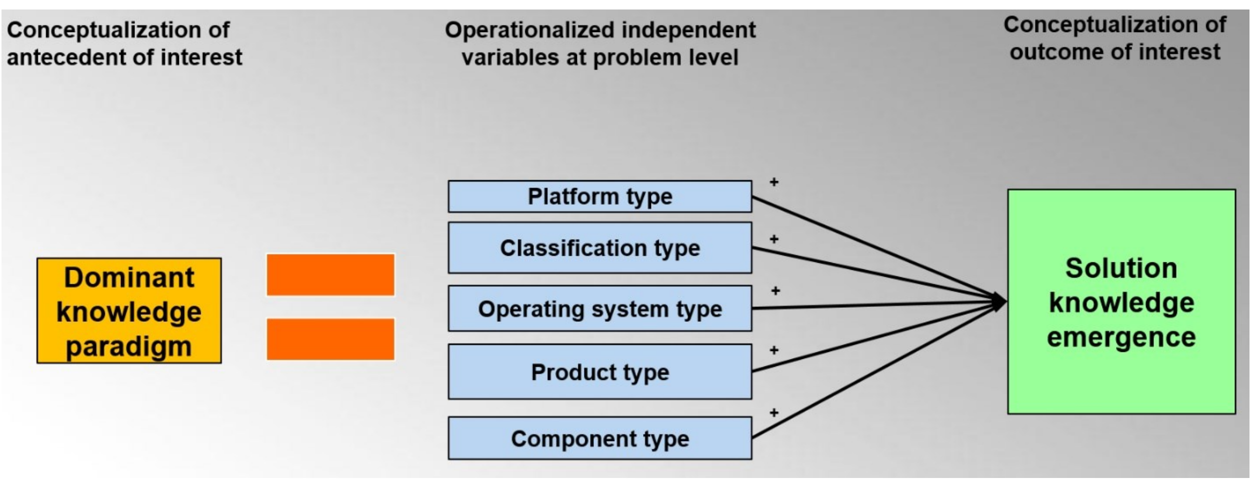

Figure 15: Operationalizations of measures of dominant knowledge paradigm at problem level

Taken collectively, these measures triangulate many different types of knowledge paradigms that have different levels of dominance as represented in the database. It is hypothesized for each one that the more dominant the knowledge paradigm of each problem knowledge reveal in the meta-organisation, the more likely solution knowledge emergence because there is likely to be more available solution knowledge in the meta-organisation amongst knowledge producers. Figure 15 summarises the operationalizations of the measures of dominant knowledge paradigm at the problem level as well as their theorised direction of influence as independent variables on the dependent outcome of interest solution knowledge emergence.

Knowledge flow impediments

The fourth antecedent of interest is knowledge flow impediments. It was triangulated with six measures derived from the open source and KBV literatures. These measures map to the descriptions in the literature of factors that interrupt the knowledge flow from problem knowledge to solution knowledge in organisations.

The first measure of knowledge flow impediments is the timing of the release of the problem knowledge to the meta-organisation relative to a formal decision to change the knowledge production process for all knowledge production activities. Historically, Mozilla's product release strategy was to release a new version of a given piece of software whenever a sufficient number of new features and bug fixes resulted in a substantial difference from the previous version. The period between releases of major products like the Firefox web browser could sometimes be a year or more. In 2010, Google began rapidly releasing new versions of its Chrome web browser, a major competitor to Firefox for market share, with releases as often as once a week. In order to keep pace with Google, on April 12, 2011, Mozilla formally adopted a "rapid release" strategy for the Firefox web browser and several other products. As a result, the period from April 12, 2011 to December 31, 2012 in the database reflects a deliberate change in the knowledge production process with wide-ranging implications for solution knowledge emergence distinct from the period of January 1, 1998 to April 11, 2011. The present measure is represented as a simple logical variable, "bug was created before/after rapid-release strategy", and it is hypothesized that knowledge reveals taking place after the rapid-release strategy was implemented are more likely to lead to solution knowledge emergence as that was one of the stated goals of using the strategy. This measure maps to the concept in the literature that deliberate knowledge flow strategies can be used to influence knowledge production (Gupta & Govindarajan, 1991; Appleyard, 1996; Fahey & Prusak, 1998; Alexy, George, & Salter, 2013; Baysal, et al., 2012b; Garrett Jr., & Covin, 2015).

The second measure of knowledge flow impediments was knowledge production activity quantities and timing. Activities, tracked in the database by cross-referencing the "activity" table with the "bugs" table, describe in detail the actions taken by knowledge actors to move problem knowledge along the knowledge production process towards generating solution knowledge. The literature suggests that the frequency and timing of these activities are correlated with the outcome of the knowledge production process (Hooimeijer & Weimer, 2007; Guo, et al., 2010). Activities immediately after the reveal of the problem knowledge to the meta-organisation are hypothesised to be positively correlated with solution knowledge emergence, whereas activities a long time after creation are hypothesised to be negatively correlated with solution knowledge emergence. The timing of these activities reflects knowledge actor engagement in the knowledge flow, with earlier being better and later suggesting that the problem knowledge was initially ignored and revisited as an afterthought, at which point it may have lost relevance. Activity count overall is hypothesised to be positively correlated with solution knowledge emergence as it reflects engagement of solution knowledge producers in resolving the problem.

Preliminary analysis of activity patterns revealed that the continuous distributions across time thresholds are non-linear and not readily transformed to linear with standard transformations. Instead, logical variables were created around frequency inflection points of time since bug was created in order to capture time frames that are relevant to the measure without sacrificing statistical power. Similarly, preliminary analysis of the quantile distribution of the range of number of activities amongst all the bugs revealed a non-linear pattern with an inflection point at 20 activities total. Comparing the time and quantity quantile distributions also revealed an inflection point for more than 20 activities occurring more than 2 years after creation, suggesting a distinct effect for the subset of bugs where that's the case. The result of this preliminary analysis was the creation of 15 logical variables that, collectively, categorise each bug relative to the inflection points observed in the frequency distributions. Experimentation with continuous variables and differing time and quantity thresholds revealed these measures to be the best balance between statistical power and theoretical validity. Table 5 summarises the thresholds of activities relative to bug creation and their hypothesised direction of influence on solution knowledge emergence. Table 6 summarises the quantity-based thresholds for activities as well as the inflection point of the interaction of quantity and timing in the case of more than 20 activities occurring more than 2 years after bug creation which reverses the hypothesised direction of influence relative to the occurrence of fewer activities than 20 in the period beyond 2 years after creation. Taken collectively, these variables triangulate the measure of knowledge flow impediments related to activities both temporally and in terms of quantity, improving the validity of the measure.

Time after creation thresholds for activities	Hypothesised direction of influence on outcome
0 hours < activity <= 3 hours	+ (Strongest)
3 hours < activity <= 6 hours	+
6 hours < activity <= 12 hours	+
12 hours < activity <= 24 hours	+
1 day < activity <= 3 days	+
3 days < activity <= 7 days	+ (Weakest)
7 days < activity <= 15 days	- (Weakest)
15 days < activity <= 45 days	-
45 days < activity <= 90 days	-
90 days < activity <= 180 days	-
180 days < activity <= 365 days	-
1 year < activity <= 2 years	-
2 years < activity	- (Strongest)

Table 5: Activity timing threshold measures of knowledge flow impediments

Activity quantity thresholds	Hypothesised direction of influence on outcome
20 < activities total	+
20 < activities later than 2 years after creation	+ (Reverses direction of influence of timing alone)

Table 6: Activity quantity threshold measures of knowledge flow impediments

The third measure of knowledge flow impediments consists of whether the knowledge flow was rerouted through reopening or reassigning activities, as previously described and depicted in Figure 10. These variables are the same as those used as the third measure of the dependent outcome variable of interest, alternatively considered as an antecedent. Similar to the case of the measure of time to outcome being considered separately as independent and dependent variable, this measure cannot be both dependent and independent variable at the same time. As a result, it is only assessed when it can be theoretically separated from itself and evaluated with other orthogonal measures of solution knowledge emergence such as the nature or timing of outcome. The choice to separately consider reopening and reassigning as antecedents and outcomes in different statistical analyses was made as an alternative to multivariate regression that promotes interpretability and statistical power for identification of effects and their relationship to one another across models rather than within a single model. Further, the dual nature of these measures reflects their descriptions in the open source literature which considers them alternatively as antecedents and outcomes depending on the frame of reference, which is the present case (Guo, et al., 2010; Guo, et al., 2011).

The fourth measure of knowledge flow impediments is the changing of knowledge flow signalling artefacts in the database during the course of the knowledge flow of each set of problem knowledge. Knowledge actors who engage in various triaging roles will periodically change signals attached to bugs during their knowledge flow. There are four main signals: keywords, flags, whiteboard, and target milestone. Keywords reflect identifiers associated with the problem knowledge that enables categorization based on known verbal tokens. Keywords are selected from a pre-approved list of relevant keywords that is periodically updated by senior members of the meta-organisation. Flags are custom identifiers with a positive or negative signal that can be changed over the course of the knowledge flow. These flags could be partial completion milestones for the problem production. They could also be used for coordination across different but related pieces of problem knowledge to synchronize their knowledge flow. The whiteboard is a scratch space used to track textual descriptions of status related to the problem knowledge by the knowledge producers. It complements the keywords field by allowing the use of any text rather than pre-approved keywords. Frequently used text in the whiteboard may be periodically moved to the list of usable keywords in the keyword field making their use related but distinct. Target milestone is a field used to synchronize the release of solution knowledge in a single version of a product. For example, several sets of problem knowledge related to new desired features may have a target milestone signal set to “Version 5”, indicating that even if the solution knowledge is completed before hand it will be released simultaneously with other solution knowledge that has this same target milestone. This practice is common in software development communities. Taken collectively, these four signalling artefacts change the knowledge flow by triaging or coordinating the parallel production of knowledge in a manner that may be different from how each individual knowledge flow would proceed in isolation.

Preliminary examination of the distribution of the count of each of these signalling variables revealed a non-linear frequency distribution amongst bugs that could not readily be transformed to linear form. Instead, a logical variable was created to differentiate problem knowledge whose initial keywords, flags, whiteboard, and target milestone set during initial problem knowledge reveal remained unchanged during the course of the knowledge flow from those sets of problem knowledge whose initial keywords, flags, whiteboard, or target milestone were changed subsequent to the initial problem knowledge reveal. Examination of the frequency distribution quantiles suggested that these two categories were the only major knowledge flow impediment factors and that a continuous variable would not properly represent the actual frequency distributions, diluting statistical power. As the goal is to identify large scale effects in this study, future research may usefully consider a more nuanced count of changes in signalling variables over time using a database that has suitable frequency distributions for such analysis which is beyond the scope of the present study.

The fifth measure of knowledge flow impediments was whether or not the knowledge flow life cycle was violated. The knowledge flow life cycle is a designated flow agreed upon by participants in the meta-organisation that is updated from time to time to reflect attempts to improve the flow's efficiency and effectiveness based on up-to-date knowledge production practices. The life cycle process used for the knowledge production captured in the database is depicted in Figure 10. Problem knowledge enters the flow at the top and exits the flow when it reaches one of the "green" statuses at the bottom. The arrows between states in designate valid paths in the knowledge flow life cycle. The present measure examines the case where states are "jumped", i.e., a transition takes place that is not indicated by one of the arrows. For example, there is no valid knowledge flow life cycle link between the "verified" and "assigned" states. Therefore, if a bug were to transition from "verified" to "assigned" without first going to the "reopened" state, it would be said to have violated the bug life cycle. For the present measure, all valid state transitions, indicated by the arrows in the diagram, were mapped and all state transitions for all bugs in the database were examined in the "activity" table. The logical variable "violated bug lifecycle" was set to "true" for all bugs that proceeded through a knowledge flow life cycle transition that was not a valid transition designated by an arrow. This measure maps to the concept in the literature that agreed upon knowledge flows improve outcomes if they are adhered to (Fahey & Prusak, 1998; Birkinshaw & Sheehan, 2002; Maier & Remus, 2003; Koponen, 2006; Zucker, et al., 2007). As such it is theorised that violation of the bug life cycle is negatively correlated with solution knowledge outcome measures.

The sixth measure of knowledge flow impediments is the dependency relationship between sets of problem knowledge. Two variables reflect complementary sides of this measure, reflection the notion in the literature that knowledge flows may be impeded by cross-problem effects that are independent from a given set of problem knowledge considered in isolation (Birkinshaw, Nobel, & Ridderstråle, 2002; Sandusky, Gasser, & Ripoche, 2004; Garud & Kumaraswamy, 2005). The first variable identifies the case when a focal set of problem knowledge reveal depends on another set of problem knowledge first being solved before a solution can be created to address the focal problem. This situation is known as a bug being "blocked by" another bug. The term "blocked" is meant to convey the sense of a physical impediment in the knowledge production process that indicates a solution ordering dependency between bugs. The second variable identifies the reciprocal side of this dyadic relationship between bugs, i.e., when a bug is "blocking". In this case, the focal problem must be solved before another problem can be addressed. As a result, each bug can, independently, be blocking one or more bugs and blocked by one or more bugs. Preliminary frequency analysis revealed that the number of bugs that each bug is blocking or blocked by is non-linear and best represented by the logical variables of "blocking one or more" and "blocked by one or more". As with previous measures, given that the goal in this study is to capture large scale effects, future research may usefully examine the effect of the count of blocking or blocked by relationships using network theory modelled with structural equation models and a database that has data with suitable frequency distributions, which is beyond the scope of the present study.

Taken collectively, these measures triangulate the many different types impediments that may delay or reroute the flow of knowledge production. Figure 16 summarises the operationalizations of the measures of knowledge flow impediments at the problem level as well as their theorised direction of influence as independent variables on the dependent outcome of interest solution knowledge emergence.

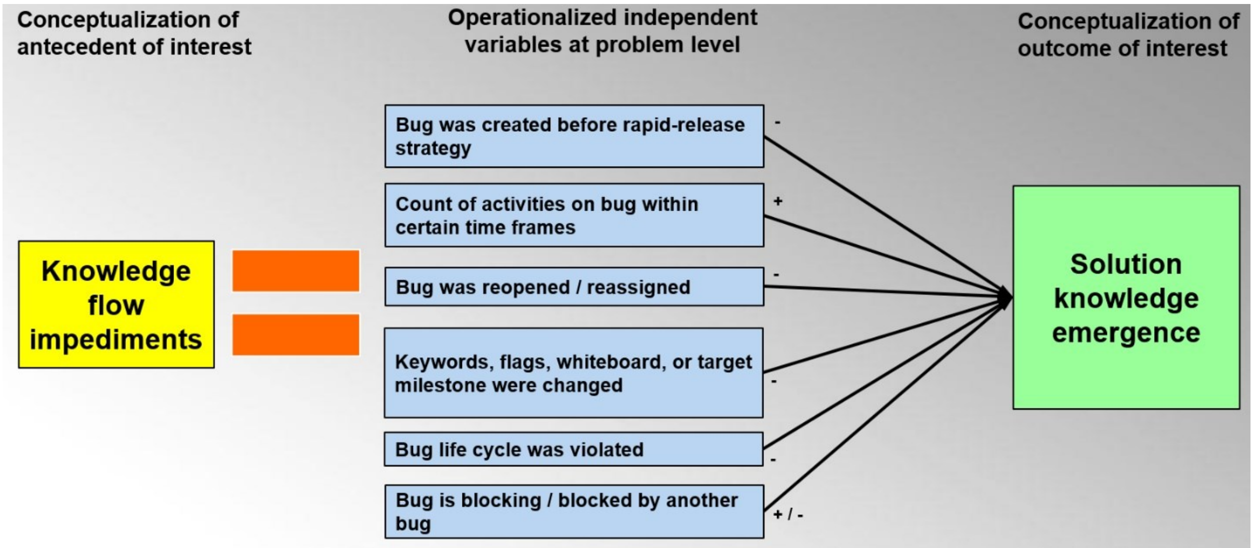

Figure 16: Operationalizations of measures of knowledge flow impediments at problem level

Knowledge stakeholder influence

The fifth antecedent of interest is knowledge stakeholder influence. The concept of "stakeholder" is defined broadly in the present study to include all actors, individual and organisational, that are involved in the knowledge production process itself or have a stake in its outcome. In this sense, the major stakeholders include the actor that creates and submits the problem knowledge, the actor who creates the solution knowledge, and the actor who verifies the solution knowledge. Other stakeholders include the broader open source community, which includes the users of the products that are produced as part of the collective effort.

The knowledge stakeholder influence measure was triangulated with five measures derived from the open source and KBV literatures. These measures map to the descriptions in the literature of factors that relate to the influence of knowledge stakeholders in meta-organisations that influence knowledge production.

The first measure of knowledge stakeholder influence is the stakeholder who revealed the initial problem knowledge to the meta-organisation. Reputation is a commonly reported factor in the literature that is theorised to affect solution knowledge emergence as certain stakeholders have sway over the knowledge production process and, as result, the meta-organisation will prioritise problem knowledge submitted by certain stakeholders over other problem knowledge independent of the contents of the problem knowledge (c.f. Anvik, Hiew, & Murphy, 2006; Panjer, 2007; Au et al., 2009; Giger, Pinzger & Gall, 2010; Zimmermann, et al., 2012). This measure is represented by a variable that uniquely identifies the profile of the stakeholder who submitted the problem knowledge. As there are hundreds of thousands of unique profiles in the database, while theoretically this variable is categorical in nature, as with previous variables with a large number of categories, it was treated as a continuous numerical variable for the purpose of analysis.

The second measure of knowledge stakeholder influence is the nature of the role of the stakeholder who revealed the initial problem knowledge. Whereas the former measure considers each individual problem knowledge revealing stakeholder, the present measure separates the bug reporting stakeholders into those who are “core actors” and those who are not. Core actors are those knowledge actors who have certain abilities to influence the knowledge production process that other knowledge actors do not. These abilities are typically reserved for senior members of the meta-organisation and are earned over time, resulting in a form of knowledge stakeholder influence hierarchy. Knowledge stakeholders were classified as “core knowledge actors” if their profile had one or more of the abilities to edit parameters in bugs, create/modify groups of profiles, create/modify components, create/modify keywords, modify user profiles, confirm bug reports, create/modify classifications, create/modify profile notifications, or, was listed as a designated component owner, quality assurance contact or default recipient of notifications. Only approximately 0.23% of profiles in the database had one or more of these abilities, suggesting that “core knowledge actors” are an exclusive type of knowledge stakeholder with disproportionate influence as compared to other knowledge stakeholders, which is a close match to the description of the role the present measure is attempting to capture in the literature (Dalle, Besten, & Masmoudi, 2008; Dalle, et al., 2008; Guo, et al., 2010; Masmoudi, 2012). At the problem level, the variable retained for the present measure was a logical variable indicating whether or not the stakeholder who revealed the problem knowledge as a core knowledge actor.

The third measure of knowledge stakeholder influence is the number of stakeholders designated as core or peripheral knowledge actors who follow, vote for, and/or comment on each set of problem knowledge. Whereas core knowledge actors are defined as in the previous measure, peripheral knowledge actors are defined as those knowledge actors who are not core actors, i.e., do not have any of the abilities set on their profile as described for the previous measure, and, who also have never submitted problem knowledge, have never submitted an attachment to a set of problem knowledge (reporter), have never been a designated solution knowledge producer (assigned_to), and, have never been a designated solution knowledge verifier (QA_contact). These actors reflect the notion in the open source literature that there are participants in open source meta organisations who do not affect the knowledge production activities directly but instead enact an influence through peripheral participation such quantifiable observation, voting, or commenting. A third theoretical category of actors, referred to as “knowledge flow participant” actors captures those stakeholders who do not have any of the core knowledge actor abilities but have directly participated in the knowledge production in one of the roles described above. This category of stakeholders is the most common amongst profiles. As a result, it is held as the reference category in the analysis, with the core and peripheral stakeholders' relative influence measured directly.

Preliminary examination of the frequency distributions of the following, voting, and commenting activities of core and peripheral stakeholders revealed that only the following activity of core stakeholders had a linear frequency distribution suitable for analysis with a single count variable. The other five variables, namely votes by core stakeholders, comments by core stakeholders, following by peripheral stakeholders, votes by peripheral stakeholders, and comments by peripheral stakeholders, all had non-linear frequency distributions not readily transformable using standard transformations. Instead, each one was represented by a logical variable at the problem level which captured whether each set of problem knowledge had one ore more votes by a core stakeholder, one or more comments by a core stakeholder, one or more follows by a peripheral stakeholder, one or more votes by a peripheral stakeholder, and, one or more comments by a peripheral stakeholder. Preliminary analysis revealed that these logical variables captured the majority of the variance of these factors in the database, ensuring the power would be sufficient to capture major effects. As with previous measures, in future research with a database with a more suitable frequency distribution, examination of the effect of count variables could usefully be conducted, though such analysis is beyond the scope of the present study. Collectively these six variables triangulate the description in the literature of knowledge stakeholders influencing the knowledge production process both directly and indirectly (Mockus, 2002; Sandusky, Gasser, & Ripoche, 2004; Kidane & Gloor, 2007).

The fourth measure of knowledge stakeholder influence is the domain of the profile that submitted each set of problem knowledge. Much like the first measure, organisations that participate in meta-organisations are theorised in the literature to have reputations that manifest as an influence over the knowledge production process (Au et al., 2009; Baysal, et al., 2013). At the problem-level of analysis this measure is represented by the variable that captures the domain of the registered email address that is associated to the profile of the submitter of the problem knowledge. Each bug will have a single "reporter" domain. Given that there are tens of thousands of distinct domains in the database, much like the first measure, while theoretically this variable is categorical in nature, it was transformed into a continuous numerical variable for the purpose of analysis.

The fifth measure of knowledge stakeholder influence is whether or not the domain of the profile that submitted each set of problem knowledge was a known webmail domain. This measure complements the fourth measure by teasing apart the stakeholder influence effects of each domain based on whether or not each domain can be classified as an "organisation". As discussed previously, to be considered an "organisation", domains that appear on lists of known webmail domains, i.e., domains that are known to be usable by anyone, regardless of whether or not they are a member of the organisation associated with that domain, are excluded. It is theorised in the literature that "organisations" hold higher knowledge stakeholder influence than non-organisations (Baysal, et al., 2013a). The present operationalization of this concept allows a direct testing of this hypothesis in a manner distinct from the representation of organisations as all domains of registered profiles. It further allows a closer examination of the use of domains as a proxy for organisations to ensure the validity of the measure. A single logical variable captured this measure for each set of problem knowledge, separating those bugs that were reported by profiles with a domain that was a known webmail domain from those that were from other domains. The former was theorised to negatively correlate with solution knowledge emergence whereas the latter was theorised to positively correlate with solution knowledge emergence.

Taken collectively, these five measures triangulate the notion of knowledge stakeholder influence affecting the knowledge production process as described in the literature. Figure 17 summarises the operationalizations of the measures of knowledge stakeholder influence at the problem level as well as their theorised direction of influence as independent variables on the dependent outcome of solution knowledge emergence.

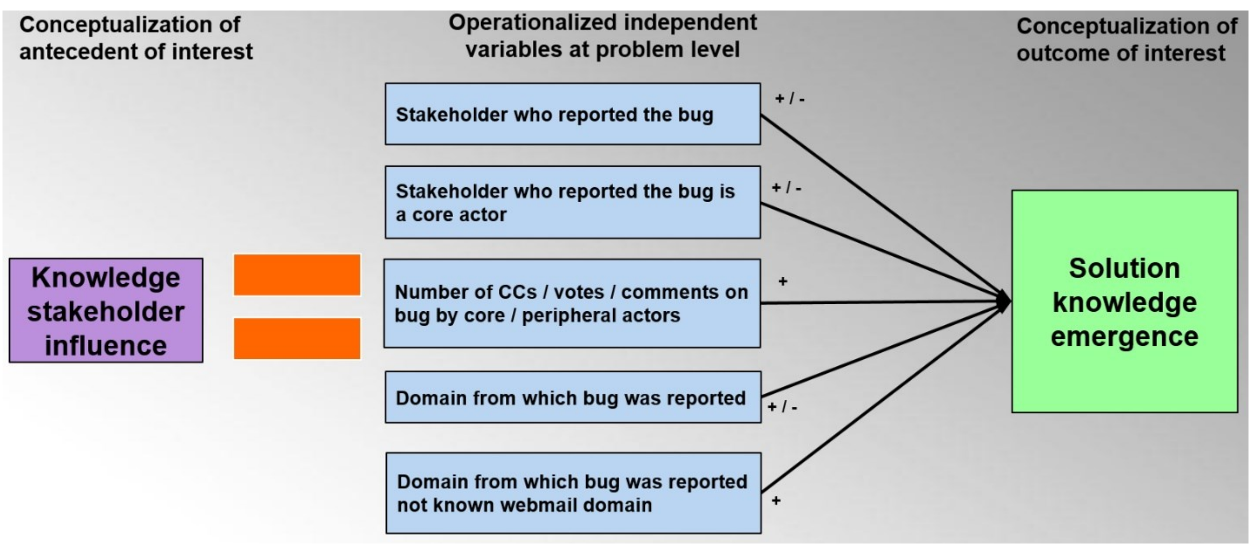

Figure 17: Operationalizations of measures of knowledge stakeholder influence at problem level

Solution knowledge value

The sixth antecedent of interest is solution knowledge value. It was triangulated with four measures derived from the open source and KBV literatures. These measures map to the descriptions in the literature of factors that relate to the value of the solution knowledge as measured both inherently to the problem and solution knowledge pair and as measured by differing value to different stakeholders in the open source meta-organisation.

The first measure of solution knowledge value is the stated importance of resolving each set of problem knowledge as specified by the problem knowledge producer and the community. The measure is represented by two variables: severity and priority. The severity variable is set by the problem knowledge producer at the time the problem knowledge is revealed to the meta-organisation. It is typically interpreted as the value of the sought solution knowledge to the focal profile and the organisation to which it is associated. It can also be interpreted as a proposed impact of the problem on other participants of the meta-organisation and a signalling mechanism to encourage solution knowledge producers to assist in resolving the problem to everyone's benefit. There are seven standard severity levels used in the database. Table 7 describes each severity level used to classify sets of problem knowledge.

Severity level	Description
Blocker	Major issue that is preventing (blocking) a product release
Critical	Problem relates to program crashes, loss of data, severe memory leaks, etc.
Major	Major loss of functionality of program
Normal	Some loss of functionality of program under specific circumstances
Minor	Minor loss of function where workaround is possible
Trivial	Cosmetic problems that don't affect functionality
Enhancement	Request for new feature or enhancement of existing feature

Table 7: Severity levels associated with sets of problem knowledge

The "enhancement" severity level, by definition, isn't so much a severity of problem as it is a desire for a new feature. It reflects the notion that problem knowledge submissions also include things beside actual problems and include all phases of product development including new feature development and more. As a result, enhancement type knowledge reveals are also classified for solution knowledge value using the priority variable to complement the severity variable. The priority variable represents the perceived usefulness of the enhancement request to the meta-organisation as a whole as judged by one of the core knowledge actors who has earned the ability to set priority on bugs through seniority. There are six priority levels, described in Table 8.

Priority level	Description
Not set	Not yet reviewed by core knowledge actor or not an enhancement request
P1	Definitely wanted by the community; useful to everybody in meta-organisation
P2	Wanted by the community
P3	May be useful to community in the future; may be useful for certain groups
P4	Not broadly useful to community but may accept if solution submitted
P5	Not useful to community; unlikely to accept even if solution submitted

Table 8: Priority levels associated with enhancement sets of problem knowledge

Both severity and priority ordered categorical variables and are hypothesised to be positively correlated with solution knowledge emergence (Ahmed & Gokhale, 2009; Bougie et al., 2010; Giger, Pinzger, & Gall, 2010; Guo, et al., 2011).

The second measure of solution knowledge value is number of changes in severity and priority subsequent to the initial reveal of problem knowledge to the meta-organisation. While ideally the initial setting of the severity variable in the bug report would be sufficient to classify solution knowledge value, in practice, the individuals who submit bugs tend to misclassify the severity either because they overestimate the usefulness of the solution to other participants in the ecosystem or because they underestimate the full impact of the problem they have described (Herraiz, 2008; Saha, Lawall, Khurshid, & Perry, 2015). As a result, core knowledge actors who have the ability to modify problem knowledge reports based on seniority in the meta-organisation effectively act as preliminary reviewers or triagers of the problem knowledge's severity and adjust it as necessary to fit the descriptions in Table 7. Likewise, problem knowledge triagers may periodically revisit the priority of feature enhancement problem knowledge reports to adjust them according to the community's updated priorities.

While changes to severity and priority could be in either direction, either increasing or decreasing the severity or priority attached to a given set of problem knowledge, the very fact of the change reflects an interesting in the community, even if it may be one of disagreement, which suggests some level of value to at least some members in the community. By contrast, a lack of severity or priority change may indicate that the initially set severity and priority (if any) were accurate but it may also indicate a lack of interest on the part of the community to engage with the problem at all. For the present research, it is hypothesized that the number of changes to severity and priority are positively correlated with solution knowledge emergence. Future research may wish to examine more closely whether a lack of change in severity or priority is more related to accuracy of initial levels or lack of interest, using a database that allows the distinction of the two cases which isn't possible in the present case.

The third measure of solution knowledge value is the popularity of the keywords associated with each set of revealed problem knowledge. Keywords are both chosen by the problem knowledge producer at the time of submission of the bug and added by core actors in the community as the bug is triaged. Keywords are used to identify potential solution knowledge producers and reflect a tag-based valuation of community priorities (Guo, et al., 2010). While there are thousands of keywords in the database, preliminary analysis revealed that certain keywords appear with far greater frequency than other keywords, suggesting they describe sets of problem knowledge that are of higher value than others. Examination of the frequency distribution quantiles of keyword popularity revealed inflection points at the top 3, top 10, top

25, and top 50 keywords. Logical variables were created for each of these inflection points such that each bug has a logical status variable for each threshold, i.e., "has one or more top 3 keywords", "has one or more top 10 keywords", "has one or more top 25 keywords" and, "has one or more top 50 keywords". The association of a "top" keyword to a set of problem knowledge is hypothesized to have a positive relationship to solution knowledge emergence, with higher tier keywords, i.e., "top 3" having a stronger positive relationship than lower tier keywords, i.e., "top 50".

The fourth measure of solution knowledge value is the number of community members following and voting for each set of problem knowledge. Each bug can be "followed" by

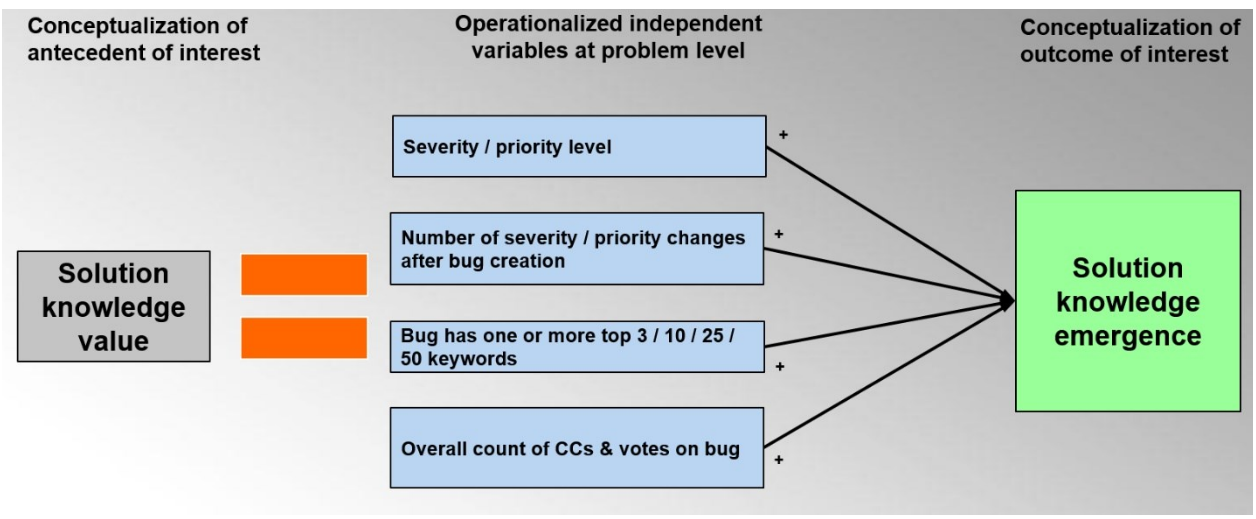

Figure 18: Operationalizations of measures of solution knowledge value at problem level

participants in the meta-organisation, which means that the following participants are notified whenever a change takes place related to the bug, including status changes, comments, and other activities. Given that it is a proactive choice for a participant to follow a bug and that no special permission is required to do so, the count of followers provides a rough measure of the value that the community, as a whole, places on each individual bug. Likewise, participants can directly "vote" for particular bugs, signaling to potential solution knowledge producers that the resolution of the problem associated with that bug is valuable to them. As a result, each bug has a count of number of participants following it with a variable known as "CC", the abbreviation for "carbon copy", and a count of the votes that have been cast for it. These variables collectively reflect the concept in the literature of solution knowledge value as expressed by the meta-organisation's community (Hooimeijer & Weimer, 2007; Panjer, 2007; Ahmed & Gokhale, 2009; Shihab, et al, 2010).

Figure 18 summarises the operationalizations of the measures of solution knowledge value at the problem level as well as their theorised direction of influence as independent variables on the dependent outcome of solution knowledge emergence.

Individual level operationalization: Dependent variables

At the individual level, the dependent outcome of interest, solution knowledge emergence, was operationalized using seven types of measurement derived from the literature that were measured or calculated in the database. Each of these measures is described in the literature as a desired outcome of knowledge revealing strategies—that is to say that factors that improve these outcomes are of strategic relevance to organisations participating in open source meta-organisations, as per the theoretical framework of this study. These measures are distinct from the dependent variable measures at the problem level in that they relate to each of the knowledge actor roles at the individual level of analysis. As a result, the measures are related to the individual level unit of analysis, the "profile", rather than the problem level unit of analysis, the "bug". Each of the seven measures is separately assessed on the subset of the profiles in the database that is classified in each of the roles described in the previous section on levels of analysis, creating, effectively, twenty-one measures for solution knowledge emergence for each hypothesis.

The first measure of solution knowledge emergence at the individual level is the percent of bugs acted upon in each role that achieved a final status of "fixed". At the problem level this measure's counterpart is the outcome of the problem, i.e., did it ever reach status "fixed" with resolution "resolved". At the individual level, this measure reflects the notion in the literature that individual actors have different success rates that are attributable to the individual rather than the problems alone. Therefore, a percentage of "fixed" vs. "not_fixed" problems can be calculated for each individual in terms of each role in which they act upon problems.

For example, suppose a focal individual has acted as problem knowledge producer 100 times and therefore is listed as "reporter" on 100 different bug reports. And, further assume that at the time of analysis 23 of those bugs have a status "pending". Of the remaining 77 bugs, 33 had an outcome of "fixed" and 44 had an outcome of "not fixed". Therefore, the percentage of bugs fixed for the focal individual in the role of problem knowledge producer is 33/77 = ~43%. This process is repeated for the same individual considering only the bugs upon which the individual acted as solution knowledge producer (assigned_to) and repeated again for the role of solution knowledge verifier (QA_contact). The result is 3 percentage variables for each profile that collectively constitute this measure at the individual level. The separation of this measure into the three roles reflects the notion in the literature that individuals may have differing abilities, different strengths and weaknesses, and, as a result, may have different solution knowledge outcome success rates depending on the roles they play in the knowledge production process.

The second measure of solution knowledge emergence at the individual level is the counterpart to the second measure at the problem level namely percent of bugs acted upon in each role that achieved a final status of “fixed” with a patch attached. This measure is calculated similarly to the previous measure except the percentages result are calculated using number of bugs “fixed with patch” versus all other results, including “fixed without a patch” for each role. As a result, this second measure will necessarily always result in a percentage that is lower than the first measure. This measure reflects the notion in the literature that individuals may have abilities related to outcomes that involve patches independently from their resolution alone and that this measure is a sought-after outcome representing a distinct from of solution knowledge emergence. As there is clear correlation between the two measures, they are never analyzed together in a single regression model as it would violate assumptions of orthogonality. Instead they are always analyzed in a complementary and comparative manner using separate regression models.

The third and fourth measures of solution knowledge emergence at the individual level capture the reopening tendencies of bugs upon which each profile acts in each role. Preliminary analysis of the frequency distribution of bug reopenings per profile revealed a heavily skewed non-linear distribution that was not readily transformable to linearity with standard functions. Instead, the reopening tendencies for each profile were split into two measures, with the first being three logical variables that capture whether or not at least one bug acted upon in each role was reopened for each profile, and the second being three non-zero percentage variables that measure the percentage of bugs that were reopened in each role, where each retained profile acted upon at least one bug that was reopened. The first measure separates the skewed frequency distribution of reopening tendencies into profile-roles that have any reopenings vs. those profile-roles that do not have any reopenings. The second measure focusses in on the subset of profile-roles with reopenings and examines the comparative percentage distribution amongst them. Taken together, these measures triangulate individual level reopening tendencies in a manner that is best suited to the observed frequency distributions in the data. Table 9 summarises the definitions of the variables that constitute the third and fourth measures of solution knowledge emergence at the individual level, collectively referred to as "reopening tendencies".

Reopening tendencies captured	Variable type	Measure
For each profile, was at least one bug acted upon in the role "reporter" reopened?	Logical	3rd
For each profile, was at least one bug acted upon in the role "assigned_to" reopened?	Logical	3rd
For each profile, was at least one bug acted upon in the role "qa_contact" reopened?	Logical	3rd
For each profile in the subset of profiles where at least one bug acted upon in the role "reporter" was reopened, what was the percentage of bugs that were reopened that were acted upon in the "reporter" role?	Non-zero percentage	4th
For each profile in the subset of profiles where at least one bug acted upon in the role "assigned_to" was reopened, what was the percentage of bugs that were reopened that were acted upon in the "assigned_to" role?	Non-zero percentage	4th
For each profile in the subset of profiles where at least one bug acted upon in the role "qa_contact" was reopened, what was the percentage of bugs that were reopened that were acted upon in the "qa_contact" role?	Non-zero percentage	4th

Table 9: Variables constituting reopening tendencies, the third and fourth measures of solution knowledge emergence at individual level

The fifth and sixth measures of solution knowledge emergence at the individual level capture the reassigning tendencies of bugs upon which each profile acts in each role. During preliminary analysis of the frequency distributions, which proved to be non-linear, it was found that the best representation of these measures was six variables that triangulate the measures in a manner similar to measures three and four. The result was three logical variables separating the rare cases of reassignments occurring at all and three non-zero percentage variables distinguishing amongst those profile-roles that had reassignments. Table 10 summarises the definitions of the variables that constitute the fifth and sixth measures of solution knowledge emergence at the individual level, collectively referred to as "reassigning tendencies".

Reassigning tendencies captured	Variable type	Measure
For each profile, was at least one bug acted upon in the role "reporter" reassigned?	Logical	5th
For each profile, was at least one bug acted upon in the role "assigned_to" reassigned?	Logical	5th
For each profile, was at least one bug acted upon in the role "qa_contact" reassigned?	Logical	5th
For each profile in the subset of profiles where at least one bug acted upon in the role "reporter" was reassigned, what was the percentage of bugs that were reassigned that were acted upon in the "reporter" role?	Non-zero percentage	6th
For each profile in the subset of profiles where at least one bug acted upon in the role "assigned_to" was reassigned, what was the percentage of bugs that were reassigned that were acted upon in the "assigned_to" role?	Non-zero percentage	6th
For each profile in the subset of profiles where at least one bug acted upon in the role "qa_contact" was reassigned, what was the percentage of bugs that were reassigned that were acted upon in the "qa_contact" role?	Non-zero percentage	6th

Table 10: Variables constituting reassigning tendencies, the fifth and sixth measures of solution knowledge emergence at individual level

The seventh measure of solution knowledge emergence is the mean time to resolution for bugs acted upon in each role by each profile. This measure complements the problem-level time to resolution measure by examining the average resolution times at the individual level as the literature suggests that different individuals may have implicit factors that affect time to resolution independent from the problems themselves. The measure was calculated by taking the subset of all bugs acted upon in each role by each profile and taking the average time to resolution for each one, resulting in three variables: one for each role. Figure 19 summarises the operationalizations of the measures of the dependent variable of interest, solution knowledge emergence, at the individual level of analysis.

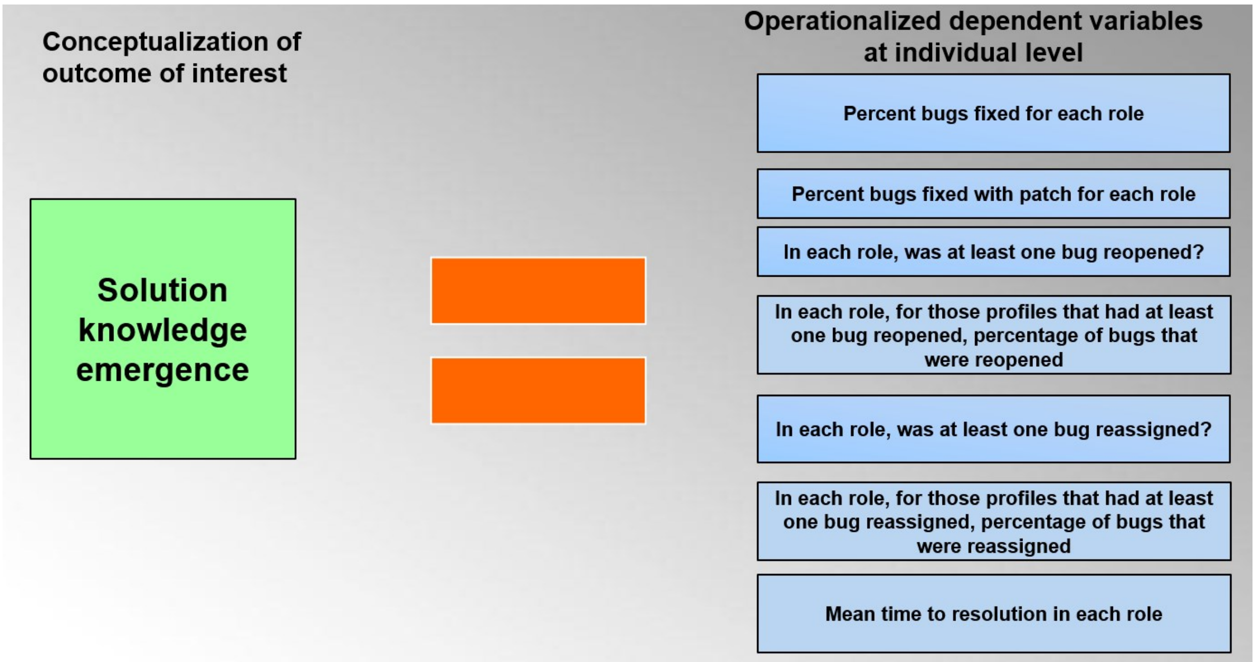

Figure 19: Operationalizations of measures of dependent variable of interest at individual level

Individual level operationalization: Independent variables

At the individual level, the independent variables were operationalized in line with each of the six hypotheses that were formulated for the conceptualizations of the antecedents of interest, in a manner similar to the operationalizations at the problem level. Each operationalization can be conceptualised as a distinct individual level measure used to triangulate the overall conceptualisations derived from the KBV and open source literatures as well as a testable sub-hypothesis with each of the above-discussed measures as antecedent independent variables.

Absorptive capacity

The first antecedent is absorptive capacity. At the individual level of analysis, it was triangulated with five measures derived from the open source and KBV literatures. The first measure was number of activities performed by each profile. Activities, tracked in the "activities" table, reflect actions taken by individuals during the knowledge creation process. These actions may include triaging, providing additional problem knowledge, moving problem knowledge through the knowledge life cycle, solving problems, and changing the signaling artefacts associated with sets of problem knowledge. The present measure, represented by a count variable of all the activities performed by each individual profile, captures the notion in the KBV literature that those individuals who are highly active may have lower absorptive capacity to engage new problem knowledge. It is therefore hypothesized that number of activities performed is negatively correlated with solution knowledge emergence. Preliminary analysis revealed that the frequency distributions of activity counts amongst individuals were non-linear but were readily transformed into a form suitable for analysis with assumptions of linearity with a standard log transformation which is common practice when working with count variables (Chambers, 1998). As such, the present measure was analysed using a log-transformed count of activities variable using the form log10 (1+x) given that there are a large number of profiles with zero counts which would result in infinite logs confounding analysis (R Foundation, 2017).

The second measure of absorptive capacity at the individual level complements the first measure by triangulating the notion in the KBV literature that absorptive capacity of individuals may be compartmentalised and have differing levels according to categories of knowledge (Cohen & Levinthal, 1990; Lane & Lubatkin, 1998; Zahra & George, 2002; Schmidt, 2010; Spithoven, Clarysse, & Knockaert, 2011). In the present case, the measure is captured with variables that count the number of activities of each individual for each platform, operating system, and product classification, which are the major categories of knowledge as distinguished in the database. Preliminary analysis of the frequency distributions of activities separated into platform, operating system and classification revealed a non-linear distribution. Experimentation with transformations and consolidation revealed that there were several meta-categories that grouped the activity counts for platform, operating system, and product classification more evenly than their full categorical range permitted.

To even the distributions for analysis, the initial 12 categories of platforms were consolidated into 6 meta categories that collect together the platforms based on frequency distribution and similarity. The initial categories of "all", "PowerPC", "x86", and "x86 64-bit" were maintained and a new category, "all others" was created to consolidate the less popular platforms. While an "other" category already existed, several other identified but infrequently used categories were also present in the database. These categories were consolidated to even the distribution for comparative analysis. This process was repeated to organize the 47 identified operating systems into 6 conceptual and more evenly distributed categories, namely "Apple PC", "Windows PC", "Windows Mobile", "Apple Mobile", "Other PC", and "Other Mobile". Incidentally, these categories represent the major conceptual meta-categories of operating systems, suggesting the consolidation not only evens frequency distribution for analysis but also matches the conceptual distinctions between the types of knowledge created in the meta-organisation. For the classifications, the initial 5 categories were reduced to four, maintaining the initial "client software", "server software" and "component" categories, while consolidating the "other" and "graveyard" categories into a single category. Finally, as with the first measure, each of these variables were log-transformed to permit analysis with assumptions of linearity as the count distributions were found to be log-linear. Table 11 summarises the 15 variables that encompass this measure for each profile.

Variables for each profile
(log) Number of activities on problem knowledge related to platform "All"
(log) Number of activities on problem knowledge related to platform "PowerPC"
(log) Number of activities on problem knowledge related to platform "x86"
(log) Number of activities on problem knowledge related to platform "x86 64-bit"
(log) Number of activities on problem knowledge related to platform "Other"
(log) Number of activities on problem knowledge related to operating system "Apple PC"
(log) Number of activities on problem knowledge related to operating system "Win PC"
(log) Number of activities on problem knowledge related to operating system "Win Mobile"
(log) Number of activities on problem knowledge related to operating system "Apple Mobile"
(log) Number of activities on problem knowledge related to operating system "Other PC"
(log) Number of activities on problem knowledge related to operating system "Other Mobile"
(log) Number of activities on problem knowledge related to classification "Client software"
(log) Number of activities on problem knowledge related to classification "Server software"
(log) Number of activities on problem knowledge related to classification "Components"
(log) Number of activities on problem knowledge related to classification "Other"

Table 11: Variables capturing activities of individuals according to knowledge categories

The third measure of absorptive capacity at the individual level is the number of activities performed by each individual separated according to severity level. This measure complements the previous two measures by capturing the notion in the literature that absorptive capacity may be a function of prioritization of the production of certain types of knowledge, reflected by the "severity" measure in the present database. As with the previous measures, given that the frequency distribution of problems organized according to severity was non-linear, the initial 7 categories of severity were consolidated into 3 meta-categories that enabled more even comparative analysis: "low", "average", and "high" severity. Further, there is no reason to believe that absorptive capacity impairment varies at a highly-refined level of categories. Given that the goal of the present study is to capture large scale effects, the consolidation improves the power of the analysis to more effectively detect such effects if they exist. Future research may wish to examine the degree to which absorptive capacity impairment is granular across problem knowledge severity (or knowledge categories, as per the previous measure). As with the previous measures, the distributions of the consolidated categories were found to be log-linear, so the log transform of each variable was taken. Table 12 summarises the three variables that were created for each individual consolidating their activities according to severity.

<table>
<tr><th>Consolidated severity category</th><th>Initial severity category</th></tr>
<tr><td rowspan="2">Low severity</td><td>Enhancement</td></tr>
<tr><td>Trivial</td></tr>
<tr><td rowspan="3">Average severity</td><td>Minor</td></tr>
<tr><td>Normal</td></tr>
<tr><td>Major</td></tr>
<tr><td rowspan="2">High severity</td><td>Critical</td></tr>
<tr><td>Blocker</td></tr>
</table>

Table 12: Consolidation of variables according to knowledge severity

The fourth measure of absorptive capacity at the individual level is the number of activities performed by each individual organized by activity type. This measure complements the previous measures by recognizing that the type of activity taken may have a separate level of absorptive capacity than factors related to the problems upon which the activity was done. There are 15 types of activities that each individual can undertake defined in the database as captured in the “activity” table. Preliminary analysis revealed a log-linear distribution for the activities amongst individuals. As a result, each of these count variables was log transformed in a manner similar to the previous measures. Table 13 summarises the types of activities that each individual can undertake, reflecting the 15 log-transformed count variables that constitute the present measure.

Activity type
Change a problem’s followers
Change a problem’s keyword
Change a problem’s product
Change a problem’s component
Change a problem’s status
Change a problem’s resolution
Change a problem’s flags
Change a problem’s whiteboard
Change a problem’s target milestone
Change a problem’s description
Change a problem’s priority
Change a problem’s severity
Assign a problem to a solution knowledge producer
Reassign a problem to a different solution knowledge producer
Reopen a problem that was previously closed

Table 13: Types of activities each individual can understake

The fifth measure of absorptive capacity at the individual level is the number of times each individual acted in each of the three roles, “problem knowledge producer”, “solution knowledge producer” and “solution knowledge verifier”. As discussed in the previous section, individuals can act upon problems in the knowledge production process in three different roles. Some individuals will engage in more than one role at a time, whereas other individuals will only engage in one (or zero roles, as is the case for the influence peripheral community members discussed at the problem level). The present measure captures the notion that absorptive capacity for individuals may vary according to role involvement. Preliminary analysis revealed the frequency distribution of actions in each role amongst individuals to be log-linear, so the three variables were log transformed in a manner similar to the previous measures. Figure 20 summarises the operationalizations of the measures of absorptive capacity at the individual level as well as their hypothesised direction of influence on the dependent outcome of interest, solution knowledge emergence.

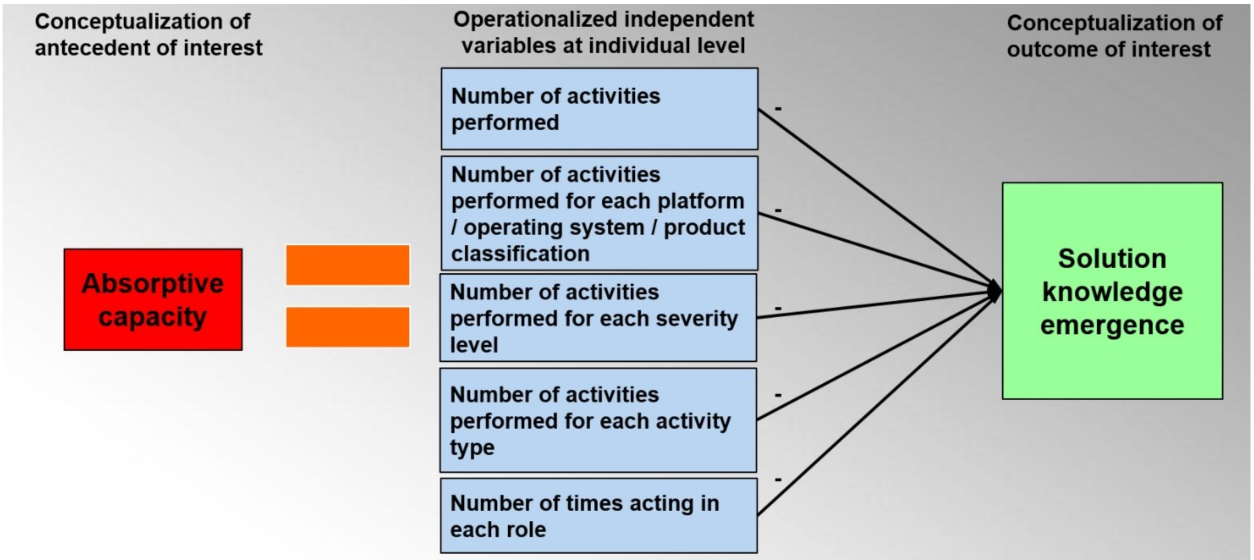

Figure 20: Operationalizations of measures of absorptive capacity at individual level

Codifiability

The second antecedent is codifiability. At the individual level of analysis, it was triangulated using five measures derived from the literature. The first measure was the mean description length of the bugs acted upon in each role. Whereas at the problem level each problem had a description whose length could be measured, at the individual level, each individual acts upon different problems in different roles. For each individual profile, acting in each of those roles, the mean description length of the problems acted upon was calculated, resulting in three variables per profile. These variables triangulate the notion that codifiability may be related to individual level abilities and actions in addition to problem-level factors. At the problem level title length was also considered. Preliminary analysis at the individual level for average title length revealed insufficient variability for appropriate analysis. As such the average title length was not included in this codifiability measure at the individual level. As was the case with previous measures, the frequency distribution was observed to be log-linear, so the log transform was done on each variable to permit analysis with assumptions of linearity. Table 14 summarises the variables created for each profile to calculate the mean description length of problems acted upon in each role in which individuals engage.

Mean description length variables
(log) Mean length of descriptions of problems acted upon in role of problem knowledge producer (**reporter**)
(log) Mean length of descriptions of problems acted upon in role of solution knowledge producer (**assigned_to**)
(log) Mean length of descriptions of problems acted upon in role of solution knowledge verifier **(qa_contact)**

Table 14: Variables capturing mean description length for problems acted upon in each role

The second measure of codifiability at the individual level was mean of the readability measures of the descriptions of the problems acted upon by individuals in each of the three roles. At the problem level, the readability measure used to assess each problem's description was the Flesch reading ease readability formula. The formula produces a numerical value that can be meaningfully averaged to calculate a mean individual level value that represents the readability associated with the full range of descriptions of problems upon which each profile has acted. Preliminary analysis of the resulting means suggested a distribution that is sufficiently linear for analysis without transformations. Table 15 summarises the variables created for each profile to calculate the mean description readability of problems acted upon in each role in which individuals engage.

Mean description readability variables
Mean Flesch reading ease of descriptions of problems acted upon in role of problem knowledge producer (**reporter**)
Mean Flesch reading ease of descriptions of problems acted upon in role of solution knowledge producer (**assigned_to**)
Mean Flesch reading ease of descriptions of problems acted upon in role of solution knowledge verifier **(qa_contact)**

Table 15: Variables capturing mean description readability of problems acted upon in each role

The third measure of codifiability at the individual level is the mean number of attachments to problems acted upon in each role. Much like the attachment measures at the problem level, the mean number of attachments to problems captures the notion that information enabling codifiability may reside at the individual level as well as at the problem level with regards to attachments to initial problem knowledge. Preliminary examination of the types of attachments at the individual level revealed insufficient variability for analysis. As such, a single variable capturing the mean number of attachments of any type for each role played by individuals in acting upon problems was created. Table 16 summarises the variables created for each profile to calculate the mean number of attachments to problems acted upon in each role in which individuals engage.

Mean attachment number variables
Mean number of attachments to problems acted upon in role of problem knowledge producer (**reporter**)
Mean number of attachments to problems acted upon in role of solution knowledge producer (**assigned_to**)
Mean number of attachments to problems acted upon in role of solution knowledge verifier (**qa_contact)**

Table 16: Variables capturing mean number of attachments to problems acted upon in each role

The fourth measure of codifiability at the individual level is the redundancy tendencies of the problem knowledge submitted by each individual. At the problem level, a give piece of problem knowledge can either be a duplicate of other problem knowledge or can be duplicated by other problem knowledge. At the individual level, the present measure captures the notion that redundancy in the knowledge available for codification can take place at the individual level and individual level effects may lead to problem knowledge that is a duplicate of or duplicated by other problem knowledge. This measure only makes sense from the perspective of the role of problem knowledge producer as the other two roles, solution knowledge producer and solution knowledge verifier do not synthesize the initial problem knowledge that could potentially be a duplicate or duplicated by other problem knowledge. As a result, two variables encapsulate this measure: The percentage of problem knowledge reports relative to all problem knowledge reports each individual submits that were identified as duplicates to other problem knowledge reports; and, the percentage of problem knowledge reports relative to all problem knowledge reports each individual submits that were duplicated by other problem reports.

The fifth measure of codifiability at the individual level is the mean number and length of comments attached to problems acted upon in each role. Much like their counterparts at the problem level, the individual levels of mean number of comments and mean comment length aim to capture the notion that the codification enabling additional information submitted via comments to supplant the initially submitted problem knowledge may have individual level effects. Therefore, three variables were created to capture the mean comment length amongst problem and three variables to capture mean comment count amongst problems, one for each role in which individuals engage. Preliminary examination of the frequency distribution of mean comment length and mean comment count revealed a log-linear relationship. As such, a log transformation was done on each variable to enable analysis with assumptions of linearity in a manner similar to previous measures. Table 17 summarises the six variables triangulating the tendencies of comments attached to problems acted upon in each of the roles undertaken by individuals.

Comment tendency variables
(log) Mean length of comments on problems acted upon in problem knowledge producer role **(reporter)**
(log) Mean length of comments on problems acted upon in solution knowledge producer role **(assigned_to)**
(log) Mean length of comments on problems acted upon in solution knowledge verifier role **(qa_contact)**
(log) Mean number of comments on problems acted upon in problem knowledge producer role **(reporter)**
(log) Mean number of comments on problems acted upon in solution knowledge producer role **(assigned_to)**
(log) Mean number of comments on problems acted upon in solution knowledge verifier role **(qa_contact)**

Table 17: Variables capturing tendencies of comments on problems acted upon in each role at individual level

Collectively, these measures triangulate the concept of codifiability in terms of individual level effects. Figure 21 summarises the operationalizations of the measures of codifiability at the individual level as well as their hypothesised direction of influence on the dependent outcome of interest, solution knowledge emergence.

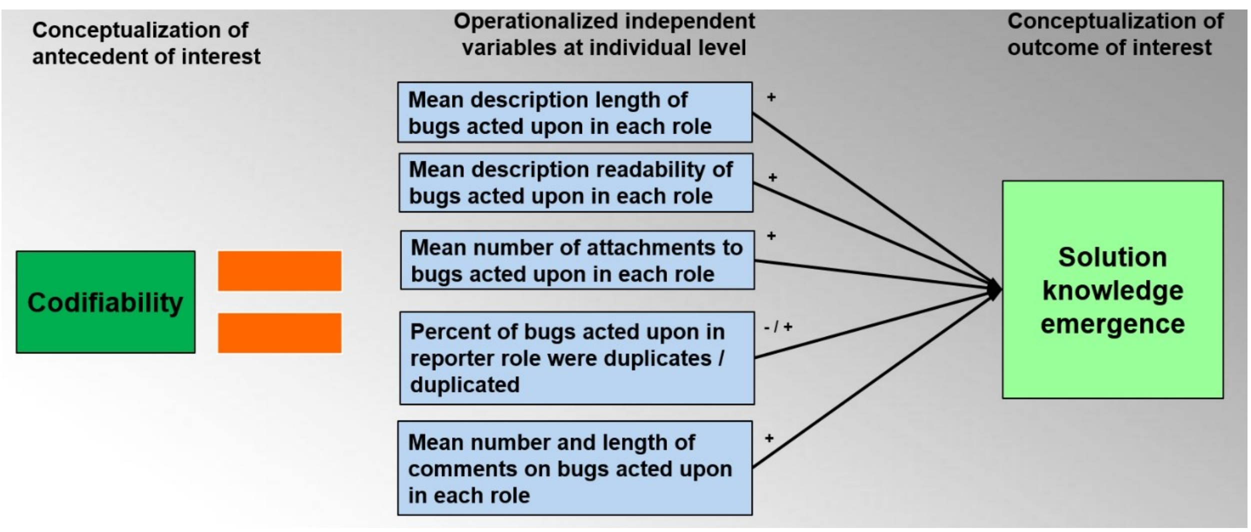

Figure 21: Operationalizations of measures of codifiability at individual level

Dominant knowledge paradigm

The third antecedent of interest is dominant knowledge paradigm. It was triangulated at the individual level using three measures derived from the literature. Whereas five measures were used at the problem level, given the large number of products and components in the database, it was impractical to create all variable permutations as the frequencies would have been too low for comparative analysis.

The first measure was percent of actions in each role upon bugs of each type of platform. Whereas the problem-level dominant knowledge paradigm measure simply seeks to classify problems according to their platform, the individual level measure seeks to examine the action tendencies of each individual acting in each of the standard roles of "reporter", "assigned_to", and "QA_contact". The platforms were consolidated into five categories, "PowerPC", "x86", "x86 64-bit", "all", and "other", to more evenly distribute them for comparative analysis. The "other" category was maintained as the reference category during analysis as the sum of all the categories for each individual-role is always 100%, by definition. The result was 12 variables per individual (3 roles times 4 platforms) making up this platform tendency measure. Table 18 summarises the variables for the platform dominant knowledge paradigm measure for each role at the individual level.

The second measure was percent of actions in each role upon bugs of each type of operating system. Much like the previous measure, this measure captures the individual level knowledge paradigm tendencies in each role for each operating system. The operating systems were consolidated into eight categories to improve frequency distributions for comparative analysis: "Android", "Linux", "Apple PC", "Windows PC", "Apple Mobile", "Windows

Mobile", "Other PC", and "Other Mobile". The "Other PC" category was held as the reference category during analysis. The result was 24 variables per individual (3 roles times 8 operating systems) making up this operating systems tendency measure. Table 19 summarises the variables capturing the operating system dominant knowledge paradigm measure for each role at the individual level.

Role	Platform	Variable
Problem knowledge producer (reporter)	All	% bugs as reporter for platform All
	PowerPC	% bugs as reporter for platform PowerPC
	x86	% bugs as reporter for platform x86
	x86_64	% bugs as reporter for platform x86 64-bit
Solution knowledge producer (assigned_to)	All	% bugs as assigned_to for platform All
	PowerPC	% bugs as assigned_to for platform PowerPC
	x86	% bugs as assigned_to for platform x86
	x86_64	% bugs as assigned_to for platform x86 64-bit
Solution knowledge verifier (qa_contact)	All	% bugs as qa_contact for platform All
	PowerPC	% bugs as qa_contact for platform PowerPC
	x86	% bugs as qa_contact for platform x86
	x86_64	% bugs as qa_contact for platform x86 64-bit

Table 18: Variables capturing platform dominant knowledge paradigm measure for each role at individual level

The third measure was percent of actions in each role upon bugs of each type of classification. As with the previous measures, this measure captures the individual level knowledge paradigm tendencies in each role for each classification. The classifications were consolidated into "Client Software", "Server Software", "Components", and "Other", with the "Other" category held as the reference category during analysis. The result was 9 variables per individual (3 roles times 3 classifications) making up this classification tendency measure. Table 20 summarises the variables capturing the classification dominant knowledge paradigm measure for each role at the individual level.

Collectively, these measures triangulate the concept of dominant knowledge paradigm in terms of individual level effects. Figure 22 summarises the operationalizations of the measures of dominant knowledge paradigm at the individual level as well as their hypothesised direction of influence on the dependent outcome of interest, solution knowledge emergence.

Role	Operating system	Variable
Problem knowledge producer (reporter)	All	% bugs as reporter for operating system All
	Android	% bugs as reporter for operating system Android
	Linux	% bugs as reporter for operating system Linux
	Apple PC	% bugs as reporter for operating system Apple PC
	Windows PC	% bugs as reporter for operating system Windows PC
	Apple Mobile	% bugs as reporter for operating system Apple Mobile
	Windows Mobile	% bugs as reporter for operating system Windows Mobile
	Other Mobile	% bugs as reporter for operating system Other Mobile
Solution knowledge producer (assigned_to)	All	% bugs as assigned_to for operating system All
	Android	% bugs as assigned_to for operating system Android
	Linux	% bugs as assigned_to for operating system Linux
	Apple PC	% bugs as assigned_to for operating system Apple PC
	Windows PC	% bugs as assigned_to for operating system Windows PC
	Apple Mobile	% bugs as assigned_to for operating system Apple Mobile
	Windows Mobile	% bugs as assigned_to for operating system Windows Mobile
	Other Mobile	% bugs as assigned_to for operating system Other Mobile
Solution knowledge verifier (qa_contact)	All	% bugs as qa_contact for operating system All
	Android	% bugs as qa_contact for operating system Android
	Linux	% bugs as qa_contact for operating system Linux
	Apple PC	% bugs as qa_contact for operating system Apple PC
	Windows PC	% bugs as qa_contact for operating system Windows PC
	Apple Mobile	% bugs as qa_contact for operating system Apple Mobile
	Windows Mobile	% bugs as qa_contact for operating system Windows Mobile
	Other Mobile	% bugs as qa_contact for operating system Other Mobile

Table 19: Variables capturing operating system dominant knowledge paradigm measure for each role at individual level

Role	Classification	Variable
Problem knowledge producer (reporter)	Client Software	% bugs as reporter for classification Client Software
	Server Software	% bugs as reporter for classification Server Software
	Component	% bugs as reporter for classification Component
Solution knowledge producer (assigned_to)	Client Software	% bugs as assigned_to for classification Client Software
	Server Software	% bugs as assigned_to for classification Server Software
	Component	% bugs as assigned_to for classification Component
Solution knowledge verifier (qa_contact)	Client Software	% bugs as qa_contact for classification Client Software
	Server Software	% bugs as qa_contact for classification Server Software
	Component	% bugs as qa_contact for classification Component

Table 20: Variables capturing classification dominant knowledge paradigm measure for each role at individual level

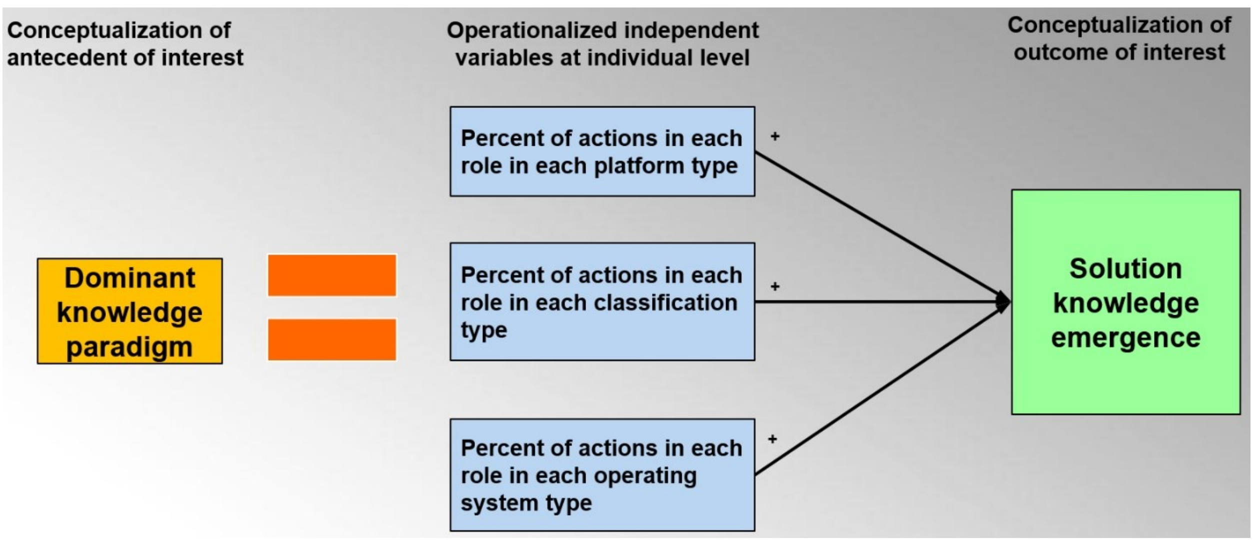

Figure 22: Operationalizations of measures of dominant knowledge paradigm at individual level

Knowledge flow impediments

The fourth antecedent of interest is knowledge flow impediments. It was triangulated at the individual level using six measures derived from the literature. The first measure was percent of bugs acted upon in each role that violated the bug life cycle. Whereas at the problem level each bug was examined to determine whether or not the bug life cycle was followed as per the knowledge flow depicted in Figure 10, the goal of the present measure is to capture the individual level bug life cycle violation tendencies in each role in which participants engage. The result was 3 percentages, one for each role, for each profile, representing violation of bug life cycle. Necessarily, the percentage of bugs acted upon in each role that did not violate the bug life cycle became the reference category for analysis.

The second measure was percent of bugs acted upon in each role whose target milestone was changed at least once. This measure captures the individual level tendencies of bugs acted upon in each of the three roles. Examination of tendencies for target milestone changes for bugs acted upon in each role revealed that the comparative frequency distribution at the individual level was best represented by the percentage of bugs acted upon in each role where "target milestone changed at least once" and "target milestone never changed", the latter being the reference category for analysis. As with the previous measure, the result was 3 percentages, one for each role, for each profile.

The third measure was percent of bugs acted upon in each role whose severity was changed at least once. Like the previous measure, this measure captures individual level severity change tendencies in each of the three roles in which individuals act. Similarly, the comparative frequency distribution at the individual level was best represented as percentage of bugs acted upon in each role where "severity changed at least once" and "severity never changed", the latter being the reference category for analysis. Three percentages variables were created for each profile for this measure, one for each role.

The fourth and fifth measures were percent of bugs acted upon in each role that were reopened or reassigned at least once. As with the previous measures, the comparative frequency distributions at the individual level were best represented as percentage of bugs acted upon in each role where “bug was reopened/reassigned at least once” and “bug was never reopened/reassigned”, the latter being the reference categories for analysis. Three percentage variables were created for each measure, one for each role.

The sixth measure was number of activities taking place on each bug acted upon in each role within certain time frames. This measure seeks to capture the individual level counterpart of the problem level examination of activities on problems. At the individual level, the goal was to establish the activity tendencies for all problems acted upon in each role for each individual. Preliminary frequency distribution analysis of the mean number of activities occurring in various ranges of time on each problem in each role revealed a log-linear relationship at thresholds comparable to those used at the problem level. The result was the creation of 21 variables for each individual, 7 for each of the three roles in which individuals engage, that capture the log of the mean number of activities taking place in each time range depicted in Table 21. For activities occurring in the first 24 hours or more than 365 days after bug creation, preliminary analysis revealed a non-linear tendency for each individual-role that was not readily transformable for comparative analysis. Given that the literature suggests that very-quickly acted upon problems and very slowly acted upon problems may have unique problem-specific features that cause them to be outliers, the decision was made to exclude these activities as outliers from analysis at the individual level as they are likely to skew results in a manner that obscures effects occurring in the retained time ranges.

Collectively, these measures triangulate the concept of knowledge flow impediments in terms of individual level effects. Figure 23 summarises the operationalizations of the measures of knowledge flow impediments at the individual level as well as their hypothesised direction of influence on the dependent outcome of interest, solution knowledge emergence.

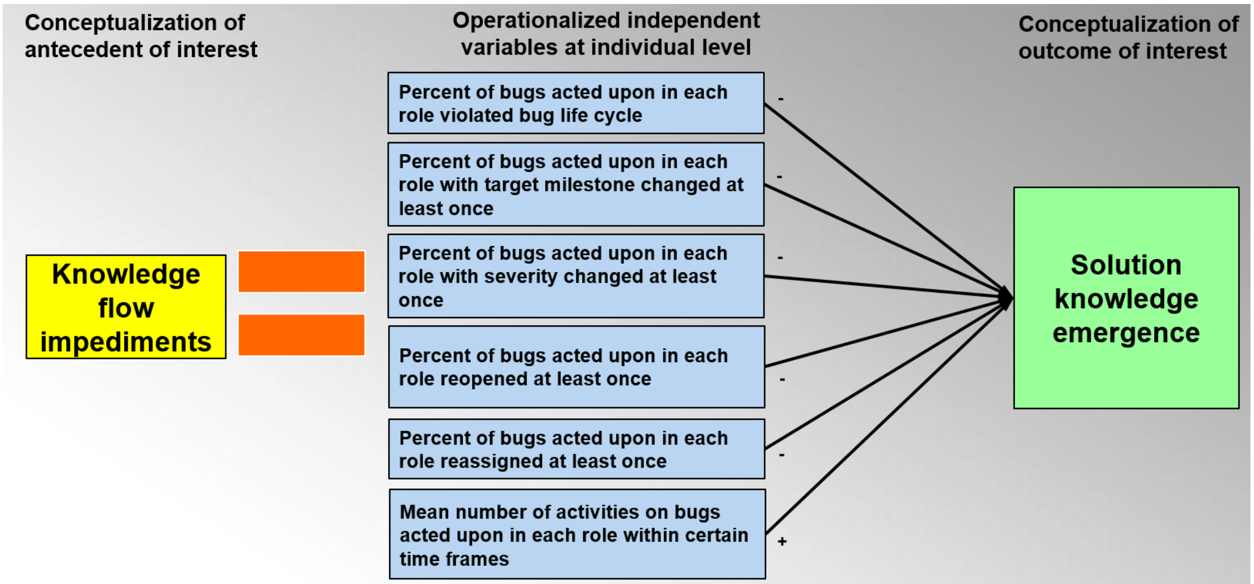

Figure 23: Operationalizations of measures of knowledge flow impediments at individual level

Role	Time range	Variable
Problem knowledge producer (reporter)	1 < t <= 3 days	(log) mean number of activities from 1 to 3 days after creation of bugs acted on as reporter by profile
	3 < t <= 7 days	(log) mean number of activities from 3 to 7 days after creation of bugs acted on as reporter by profile
	7 < t <= 15 days	(log) mean number of activities from 7 to 15 days after creation of bugs acted on as reporter by profile
	15 < t <= 45 days	(log) mean number of activities from 15 to 45 days after creation of bugs acted on as reporter by profile
	45 < t <= 90 days	(log) mean number of activities from 45 to 90 days after creation of bugs acted on as reporter by profile
	90 < t <= 180 days	(log) mean number of activities from 90 to 180 days after creation of bugs acted on as reporter by profile
	180 < t <= 365 days	(log) mean number of activities from 180 to 365 days after creation of bugs acted on as reporter by profile
Solution knowledge producer (assigned_to)	1 < t <= 3 days	(log) mean number of activities from 1 to 3 days after creation of bugs acted on as assigned_to by profile
	3 < t <= 7 days	(log) mean number of activities from 3 to 7 days after creation of bugs acted on as assigned_to by profile
	7 < t <= 15 days	(log) mean number of activities from 7 to 15 days after creation of bugs acted on as assigned_to by profile
	15 < t <= 45 days	(log) mean number of activities from 15 to 45 days after creation of bugs acted on as assigned_to by profile
	45 < t <= 90 days	(log) mean number of activities from 45 to 90 days after creation of bugs acted on as assigned_to by profile
	90 < t <= 180 days	(log) mean number of activities from 90 to 180 days after creation of bugs acted on as assigned_to by profile
	180 < t <= 365 days	(log) mean number of activities from 180 to 365 days after creation of bugs acted on as assigned_to by profile
Solution knowledge verifier (qa_contact)	1 < t <= 3 days	(log) mean number of activities from 1 to 3 days after creation of bugs acted on as qa_contact by profile
	3 < t <= 7 days	(log) mean number of activities from 3 to 7 days after creation of bugs acted on as qa_contact by profile
	7 < t <= 15 days	(log) mean number of activities from 7 to 15 days after creation of bugs acted on as qa_contact by profile
	15 < t <= 45 days	(log) mean number of activities from 15 to 45 days after creation of bugs acted on as qa_contact by profile
	45 < t <= 90 days	(log) mean number of activities from 45 to 90 days after creation of bugs acted on as qa_contact by profile
	90 < t <= 180 days	(log) mean number of activities from 90 to 180 days after creation of bugs acted on as qa_contact by profile
	180 < t <= 365 days	(log) mean number of activities from 180 to 365 days after creation of bugs acted on as qa_contact by profile

Table 21: Variables capturing time-based activity tendency measures for problems acted upon in each role by each profile at individual level

Knowledge stakeholder influence

The fifth antecedent of interest is knowledge stakeholder influence. It was triangulated at the individual level using three measures derived from the literature. The first measure of knowledge stakeholder influence is whether or not each profile is a core knowledge actor. This measure is captured by a simple logical variable that flags each profile. The definition of "core knowledge actor" is the same as described at the problem level. Whereas the problem level measure examined whether or not the profile that reported each focal problem was a core knowledge actor, the present measure examines whether or not each focal profile is a core knowledge actor at the individual level. Together these measures ensure proper localisation of the level of any effects related to knowledge actor centrality in the meta-organisation.

The second measure of knowledge stakeholder influence is the tendencies related to following, voting, and comments by each of the three classes of knowledge actors on bugs acted upon in each of the three roles in which each profile engages. The definitions of the three classes of knowledge actors, core knowledge actor, knowledge flow participant actor, and peripheral knowledge actor, are the same as at the problem level of analysis. The present measure complements the problem level measures in order to localise the level of any effects related to knowledge stakeholder influence in the following, voting, and commenting tendencies. The measure was operationalized with 30 variables, 10 per role in which each focal profile engages. Three measures capture the voting tendencies for core, participant, and peripheral knowledge actors on bugs acted upon by individuals in each role; three measures capture the following tendencies of core, participant, and peripheral knowledge actors on bugs acted upon by individuals in each role; and, three measures capture the commenting tendencies of core, participant, and peripheral knowledge actors on bugs acted upon by individuals in each role. A fourth measure for commenting tendencies captures the mean distinct number of commenters for bugs acted upon by individuals in each role as actors, of any involvement level, since individuals may comment multiple times on a given problem, whereas they can only follow or vote once for each problem. Therefore, the comment count variables do not fully represent the degree of stakeholder influence. The distinctiveness measure complements the count measures to separate degree of interest from range of interest amongst participants in the meta-organisation in the analysis. Table 22 summarises the variables that capture the knowledge stakeholder influence related activity tendencies at the individual level.

The third measure of knowledge stakeholder influence is the count and type of individuals and organisations each profile is observing and observed by. Much like how, at the problem level, each problem report can be followed through its life cycle by individuals and organisations, at the individual level, each profile can watch and be watched by other profiles.

Role	Tendency	Actor	Variable
Problem knowledge producer (reporter)	Votes	Core	(log) Mean number of votes by core actors on bugs acted on as reporter by profile
	Votes	Participant	(log) Mean number of votes by participant actors on bugs acted on as reporter by profile
	Votes	Peripheral	(log) Mean number of votes by peripheral actors on bugs acted on as reporter by profile
	Following	Core	(log) Mean number of follows by core actors on bugs acted on as reporter by profile
	Following	Participant	(log) Mean number of follows by participant actors on bugs acted on as reporter by profile
	Following	Peripheral	(log) Mean number of follows by peripheral actors on bugs acted on as reporter by profile
	Commenting	Core	(log) Mean number of comments by core actors on bugs acted on as reporter by profile
	Commenting	Participant	(log) Mean number of comments by participant actors on bugs acted on as reporter by profile
	Commenting	Peripheral	(log) Mean number of comments by peripheral actors on bugs acted on as reporter by profile
	Commenting	Distinct	(log) Mean number of distinct actors commenting on bugs acted on as reporter by profile
Solution knowledge producer (assigned_to)	Votes	Core	(log) Mean number of votes by core actors on bugs acted on as assigned_to by profile
	Votes	Participant	(log) Mean number of votes by participant actors on bugs acted on as assigned_to by profile
	Votes	Peripheral	(log) Mean number of votes by peripheral actors on bugs acted on as assigned_to by profile
	Following	Core	(log) Mean number of follows by core actors on bugs acted on as assigned_to by profile
	Following	Participant	(log) Mean number of follows by participant actors on bugs acted on as assigned_to by profile
	Following	Peripheral	(log) Mean number of follows by peripheral actors on bugs acted on as assigned_to by profile
	Commenting	Core	(log) Mean number of comments by core actors on bugs acted on as assigned_to by profile
	Commenting	Participant	(log) Mean number of comments by participant actors on bugs acted on as assigned_to by profile
	Commenting	Peripheral	(log) Mean number of comments by peripheral actors on bugs acted on as assigned_to by profile
	Commenting	Distinct	(log) Mean number of distinct actors commenting on bugs acted on as assigned_to by profile
Solution knowledge verifier (qa_contact)	Votes	Core	(log) Mean number of votes by core actors on bugs acted on as qa_contact by profile
	Votes	Participant	(log) Mean number of votes by participant actors on bugs acted on as qa_contact by profile
	Votes	Peripheral	(log) Mean number of votes by peripheral actors on

			bugs acted on as qa_contact by profile
	Following	Core	(log) Mean number of follows by core actors on bugs acted on as qa_contact by profile
	Following	Participant	(log) Mean number of follows by participant actors on bugs acted on as qa_contact by profile
	Following	Peripheral	(log) Mean number of follows by peripheral actors on bugs acted on as qa_contact by profile
	Commenting	Core	(log) Mean number of comments by core actors on bugs acted on as qa_contact by profile
	Commenting	Participant	(log) Mean number of comments by participant actors on bugs acted on as qa_contact by profile
	Commenting	Peripheral	(log) Mean number of comments by peripheral actors on bugs acted on as qa_contact by profile
	Commenting	Distinct	(log) Mean number of distinct actors commenting on bugs acted on as qa_contact by profile

Table 22: Variables capturing knowledge stakeholder influence related activity tendency measures for problems acted upon in each role by each profile at individual level

By extension, the organisations that are associated with each profile can watch and be watched by others as well. Comparison of profile and organisation watching and watched by measures allows localisation of any effects. To distinguish between the following of problems, the terms "watching" and "watched by" are used in the database to denote the observing of profiles and their associated organisations.

Preliminary analysis of the frequency distributions of the counts of the different types of profiles and organisations watched by and watching each profile revealed a log-linear relationship suitable for analysis with a standard log transformation. Ten variables triangulate the measure with five variables capturing the (log) count of distinct actors, distinct organisations, only core knowledge actors, only participant knowledge actors, and only peripheral knowledge actors watched by each profile. Five variables capture the (log) count of distinct actors, distinct organisations, only core knowledge actors, only participant knowledge actors, and only peripheral knowledge actors who are watching each profile. Table 23 summarises the variables capturing knowledge stakeholder observational influence measures at the individual level.

Action	Actor	Variable
Watching	All actors	(log) Count of actors watching profile
Watching	Organisations	(log) Count of organisations watching profile
Watching	Core actors	(log) Count of core actors watching profile
Watching	Participant actors	(log) Count of participant actors watching profile
Watching	Peripheral actors	(log) Count of peripheral actors watching profile
Watched by	All actors	(log) Count of actors watched by profile
Watched by	Organisations	(log) Count of organisations watched by profile
Watched by	Core actors	(log) Count of core actors watched by profile
Watched by	Participant actors	(log) Count of participant actors watched by profile
Watched by	Peripheral actors	(log) Count of peripheral actors watched by profile

Table 23: Variables capturing knowledge stakeholder observational influence measures at individual level

Collectively, these measures triangulate the concept of knowledge stakeholder influence in terms of individual level effects. Figure 24 summarises the operationalizations of the measures of knowledge stakeholder influence at the individual level as well as their hypothesised

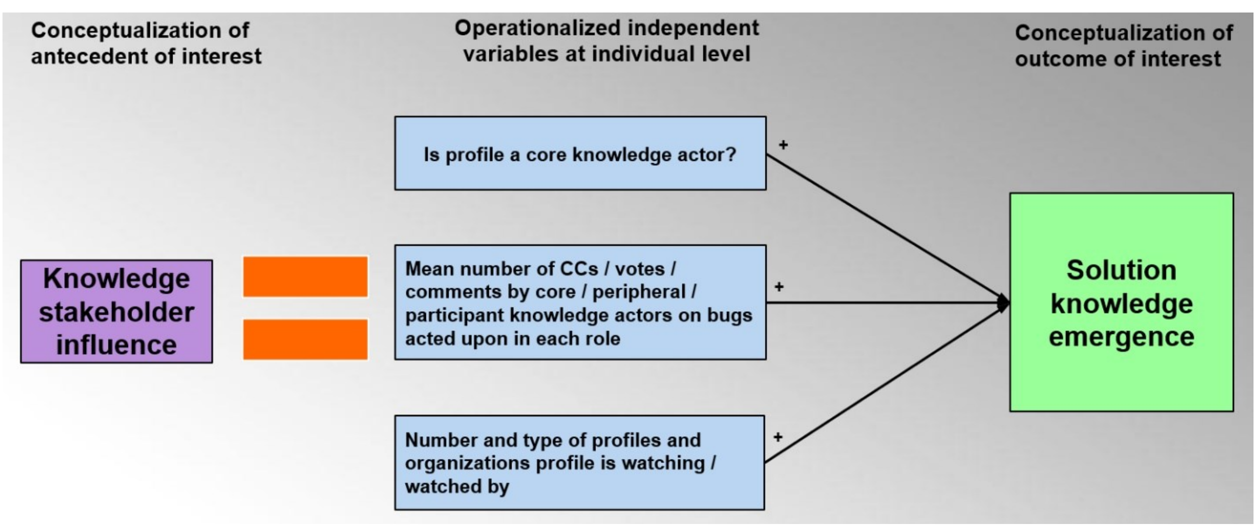

Figure 24: Operationalizations of measures of knowledge stakeholder influence at individual level

direction of influence on the dependent outcome of interest, solution knowledge emergence.

Solution knowledge value

The sixth antecedent of interest is solution knowledge value. It was triangulated at the individual level with four measures derived from the literature. The first measure was the tendencies of profiles to act upon bugs of differing severity and priority in each of the 3 roles in which individuals engage. This measure captures the notion in the literature that solution knowledge value may be reflected at the individual level in the tendencies to engage with problem knowledge that is classified at higher or lower severity or priority levels. Tendencies to engage higher severity or priority problem knowledge are hypothesized to promote solution knowledge emergence as it is theorized that solutions to those problems have greater value. Eighteen variables were created to capture each of the tendency to act upon bugs in each of the 6 severity levels from trivial to blocker, with the severity "enhancement" held as the reference category for the purpose of analysis, in each of the 3 roles, problem knowledge producer, solution knowledge producer, and solution knowledge verifier. Likewise, 15 percentage variables were created for priority, 5 for each role, from P1 to P5, with "priority not set" held as the reference category for analysis. Table 24 summarizes variables that make up the solution knowledge value severity and priority measure at individual level.

The second measure of solution knowledge value was the tendencies of profiles to act upon bugs in each of the 3 roles in which individuals engage whose severity or priority level had changed at least once since the initial reveal of the problem knowledge to the meta-organisation. Preliminary analysis of the frequency distribution of percentages of bugs acted upon in each role with varying numbers of severity and priority changes revealed that there was only sufficient variability for analysis between “changed at least once” and “never changed”. Instances of priority and severity changes occurring more than once, as previously observed at the problem level, were sufficiently infrequent as to be statistical outliers. As a result, the variables making up this measure at the individual level were selected to match their counterparts at the problem level and focus on the tendencies to act upon bugs with or without the occurrence of severity or priority change rather than continuous counts of such changes.

It should be noted that the severity change variable in the hypothesis six models is the same variable as the one used in the hypothesis four: knowledge flow impediments models. It is included again in the present models to assess its impact in combination with the priority changes in the same model to more clearly separate their effects. Given that severity changes are theorized to have both an impact on knowledge flow and on solution knowledge value signalling, and severity level is considered to be one of the most important predictors of solution knowledge emergence in the literature (c.f. Ahmed & Gokhale, 2009; Bougie et al., 2010; Giger, Pinzger, & Gall, 2010; Guo, et al., 2011; Zhang, et al., 2012), its inclusion in multiple models allows for triangulation of the concept in independent model fits to reduce the likelihood of errors due to particular combinations of variables creating spurious model fits. The observed characteristics of the severity change variable’s effects are discussed relative to the relevant hypotheses in the results and discussion section.

Role	Severity / Priority	Variable
Problem knowledge producer (reporter)	Trivial	% bugs as reporter had severity trivial
	Minor	% bugs as reporter had severity minor
	Normal	% bugs as reporter had severity normal
	Major	% bugs as reporter had severity major
	Critical	% bugs as reporter had severity critical
	Blocker	% bugs as reporter had severity blocker
	P1	% bugs as reporter had priority P1
	P2	% bugs as reporter had priority P2
	P3	% bugs as reporter had priority P3
	P4	% bugs as reporter had priority P4
	P5	% bugs as reporter had priority P5
Solution knowledge producer (assigned_to)	Trivial	% bugs as assigned_to had severity trivial
	Minor	% bugs as assigned_to had severity minor
	Normal	% bugs as assigned_to had severity normal
	Major	% bugs as assigned_to had severity major
	Critical	% bugs as assigned_to had severity critical
	Blocker	% bugs as assigned_to had severity blocker
	P1	% bugs as assigned_to had priority P1
	P2	% bugs as assigned_to had priority P2
	P3	% bugs as assigned_to had priority P3
	P4	% bugs as assigned_to had priority P4
	P5	% bugs as assigned_to had priority P5
Solution knowledge verifier (qa_contact)	Trivial	% bugs as qa_contact had severity trivial
	Minor	% bugs as qa_contact had severity minor
	Normal	% bugs as qa_contact had severity normal
	Major	% bugs as qa_contact had severity major
	Critical	% bugs as qa_contact had severity critical
	Blocker	% bugs as qa_contact had severity blocker
	P1	% bugs as qa_contact had priority P1
	P2	% bugs as qa_contact had priority P2
	P3	% bugs as qa_contact had priority P3
	P4	% bugs as qa_contact had priority P4
	P5	% bugs as qa_contact had priority P5

Table 24: Variables capturing solution knowledge value severity and priority measure at individual level

As with the previous measure, this measure captures the individual level counterpart of the severity and priority change measures at the problem level, reflecting the notion in the literature that tendencies to act on problems whose value is debated in the meta-organisation, as reflected by the changes in the severity and priority variables, is hypothesized to positively correlate with solution knowledge mergence. Six variables, 2 for each of the 3 roles in which individuals engage, were created, capturing the percentage of bugs acted upon in each role whose severity/priority changed at least once, with the percentage of bugs whose severity/priority never changed acting as the reference category for analysis. Table 25 summarises the variables that constitute the solution knowledge value severity and priority change tendency measure at the individual level.

Role	Value changed	Variable
Problem knowledge producer (reporter)	Severity	% bugs as reporter w/ severity changed at least once
	Priority	% bugs as reporter w/ priority changed at least once
Solution knowledge producer (assigned_to)	Severity	% bugs as assigned_to w/ severity changed at least once
	Priority	% bugs as assigned_to w/ priority changed at least once
Solution knowledge verifier (qa_contact)	Severity	% bugs as qa_contact w/ severity changed at least once
	Priority	% bugs as qa_contact w/ priority changed at least once

Table 25: Variables capturing solution knowledge value severity and priority change tendency measure at individual level

The third measure of solution knowledge value was the tendency to act upon bugs, in each of the three roles in which individuals engage, which had one or more top keywords. As with the previous measure, this measure captures the individual level counterpart to the presence of popular keywords attached to bugs at the problem level. This measure reflects the notion that the tendency to act upon bugs with top keywords at the individual level reflects higher solution knowledge value that is hypothesized to promote solution knowledge mergence. Twelve percent variables were created, 3 for each role, which were selected to match their problem level counterparts by examining the percentage of bugs acted upon in each role that have one or more top 3, top 10, top 25, and/or top 50 keywords. In each role, the reference category for analysis was all other bugs acted upon, such as those with no keywords or keywords not in the top 50 or higher. Table 26 summarises the variables that constitute the solution knowledge value keyword popularity tendency measure at the individual level.

Role	Keyword popularity	Variable
Problem knowledge producer (reporter)	Top 3	% bugs as reporter with top 3 keyword
	Top 10	% bugs as reporter with top 10 keyword
	Top 25	% bugs as reporter with top 25 keyword
	Top 50	% bugs as reporter with top 50 keyword
Solution knowledge producer (assigned_to)	Top 3	% bugs as assigned_to with top 3 keyword
	Top 10	% bugs as assigned_to with top 10 keyword
	Top 25	% bugs as assigned_to with top 25 keyword
	Top 50	% bugs as assigned_to with top 50 keyword
Solution knowledge verifier (qa_contact)	Top 3	% bugs as qa_contact with top 3 keyword
	Top 10	% bugs as qa_contact with top 10 keyword
	Top 25	% bugs as qa_contact with top 25 keyword
	Top 50	% bugs as qa_contact with top 50 keyword

Table 26: Variables capturing solution knowledge value keyword popularity tendency measure at individual level

The fourth measure of solution knowledge value is average overall number of follows, votes, comments, and flags attached to bugs acted upon in each of the three roles in which individuals engage. This measure complements the problem level measure by examining the average tendencies in each role at the individual level. It also complements the knowledge stakeholder influence measure by providing variables that examine the overall count of follows, votes and comments rather than those variables previously described that only consider such variables according to the stakeholder's power and influence in the meta-organisation. The measure for mean number of comments is the same measure as the one used in the hypothesis two individual level of analysis models. As discussed in the organisation level of analysis operationalisation section, alternate operationalisations of the comment count variables were created to facilitate comparing and contrasting of variables within and across levels of analysis. The mean comment variable is duplicated in hypothesis six as a robustness check that the significance of effects of the independent variables are not merely a function of the other variables with which they are paired in the models, subject to the initial classification into the six hypotheses. While non-cascading regression model fitting was used, this variable duplication in alternate models provides a validation that the model fitting is considering the independent variables independently, as intended. Together, with the alternate representations discussed in the organisation level section, these measures allow better localisation of any effect on solution knowledge mergence, improving the validity and distinctiveness of the measures.

Twelve variables make up this measure, four for each role in which individuals engage. They reflect the notion in the literature that, at the individual level, average levels of following, votes, commenting and flags amongst the roles in which individuals engage are hypothesized to positively correlate with solution knowledge emergence as such variables constitute solution knowledge value. Preliminary analysis revealed a non-linear distribution of the averages amongst profile-roles. Sufficient linearity for analytical assumptions was readily induced by taking the log transformation of the variables in a manner similar to previous variables. Table 27 summarises the variables that make up the solution knowledge value measure reflected in following, voting, commenting, and flag averages at the individual level.

Role	Tendency	Variable
Problem knowledge producer (reporter)	Following	(log) Mean number of votes on bugs acted on as reporter
	Votes	(log) Mean number of votes on bugs acted on as reporter
	Comments	(log) Mean number of comments on bugs acted on as reporter
	Flags	(log) Mean number of flags on bugs acted on as reporter
Solution knowledge producer (assigned_to)	Following	(log) Mean number of votes on bugs acted on as assigned_to
	Votes	(log) Mean number of votes on bugs acted on as assigned_to
	Comments	(log) Mean number of comments on bugs acted on as assigned_to
	Flags	(log) Mean number of flags on bugs acted on as assigned_to
Solution knowledge verifier (qa_contact)	Following	(log) Mean number of votes on bugs acted on as qa_contact
	Votes	(log) Mean number of votes on bugs acted on as qa_contact
	Comments	(log) Mean number of comments on bugs acted on as qa_contact
	Flags	(log) Mean number of flags on bugs acted on as qa_contact

Table 27: Variables capturing solution knowledge value as captured by following, voting, commenting, and flag averages measure at individual level

Collectively, these measures triangulate the concept of solution knowledge value in terms of individual level effects. Figure 25 summarises the operationalizations of the measures of solution knowledge value at the individual level as well as their hypothesised direction of influence on the dependent outcome of interest, solution knowledge emergence.

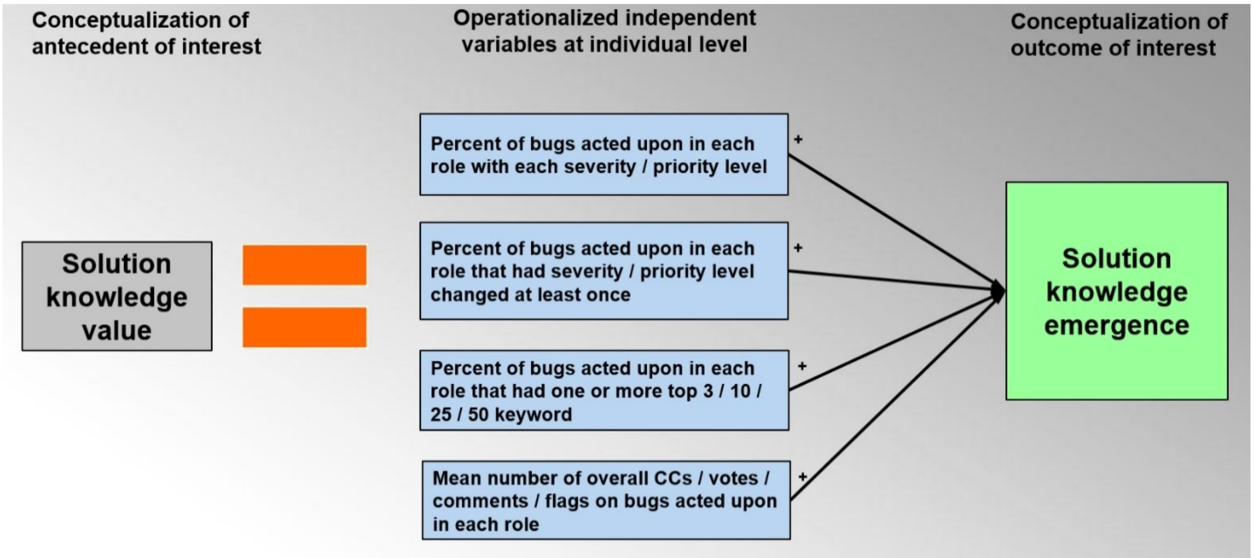

Figure 25: Operationalizations of measures of solution knowledge value at individual level

Organisation level operationalization: Dependent variables

At the organisation level, the dependent outcome of interest, solution knowledge emergence, was operationalized using seven types of measurement derived from the literature that were measured or calculated in the database. Each of these measures is described in the literature as a desired outcome of knowledge revealing strategies. As per the theoretical framework of this study, factors that improve these outcomes are of strategic relevance to organisations participating in open source meta-organisations. These measures are distinct from the dependent variable measures at the problem and individual levels in that they relate to each of the aggregate knowledge actor roles at the organisation level of analysis. As a result, the measures are related to the organisation level unit of analysis, the “organisation”, rather than the problem level unit of analysis, the “bug”, or the individual level of analysis, the “profile”. Each of the seven measures is separately assessed on the subset of the organisations in the database that have a sufficient number of actors to create aggregate values for the roles described in the previous section on the individual level of analysis, creating, effectively, twenty-one measures for solution knowledge emergence for each hypothesis at the organisation level.

The first measure of solution knowledge emergence at the organisation level is the percent of bugs acted upon in each aggregate role that achieved a final status of “fixed”. At the problem level, this measure’s counterpart is the outcome of the problem, i.e., did it ever reach status “fixed” with resolution “resolved”. At the individual level, this measure second counterpart reflects the notion in the literature that individual actors have different success rates that are attributable to the individual rather than the problems alone. Likewise, at the organisation level, this measure reflects the notion in the literature that aggregate actors in an organisation have different success rates that are attributable to the organisation rather than any given individual alone. Therefore, a percentage of “fixed” vs. “not_fixed” problems can be calculated for each organisation in terms of each aggregate role in which actors in the organisation act upon problems.

For example, suppose a focal organisation has, amongst the aggregation of all its individual actors, acted as problem knowledge producer 100 times and therefore has its members listed as “reporter” on 100 different bug reports. And, further assume that at the time of analysis 37 of those bugs have a status “pending”. Of the remaining 63 bugs, 21 had an outcome of “fixed” and 42 had an outcome of “not fixed”. Therefore, the percentage of bugs fixed for the focal organisation in the aggregate role of problem knowledge producer is 21/63 = ~33%. This process is repeated for the same organisation considering only the bugs upon which the organisation’s members, in aggregate, acted as solution knowledge producer (assigned_to) and repeated again for the aggregate role of solution knowledge verifier (QA_contact). The result is

3 percentage variables for each organisation that collectively constitute this measure at the organisation level. The separation of this measure into the three aggregate roles reflects the notion in the literature that organisations may have differing abilities, different strengths and weaknesses, and, as a result, may have different solution knowledge outcome success rates depending on the aggregate roles they play in the knowledge production process. This measure complements its counterpart individual and problem level measures to allow better localisation of any outcomes at the correct level of analysis in the data.

The second measure of solution knowledge emergence at the organisation level is the counterpart to the second measure at the problem and individual levels namely percent of bugs acted upon in each aggregate role that achieved a final status of "fixed" with a patch attached. This measure is calculated similarly to the previous measure except the percentages result are calculated using number of bugs "fixed with patch" versus all other results, including "fixed without a patch" for each aggregate role. As a result, this second measure will necessarily always result in a percentage that is lower than the first measure. This measure reflects the notion in the literature that organisations may have abilities related to outcomes that involve patches independently from problem resolution alone and that this measure is a sought-after outcome for participant organisations, representing a distinct from of solution knowledge emergence. Much like at the individual level, as there is clear correlation between the two measures, they are never analyzed together in a single regression model as it would violate assumptions of orthogonality. Instead they are always analyzed in a complementary and comparative manner using separate regression models.

The third and fourth measures of solution knowledge emergence at the organisation level capture the reopening tendencies of bugs upon which each organisation acts in each aggregate role. Similar to the case at the individual level, preliminary analysis of the frequency distribution of bug reopenings per organisation revealed a heavily skewed non-linear distribution that was not readily transformable to linearity with standard functions. Instead, the reopening tendencies for each organisation were split into two measures, with the first being three logical variables that capture whether or not at least one bug acted upon in each aggregate role was reopened for each organisation, and the second being three non-zero percentage variables that measure the percentage of bugs that were reopened for each aggregate role, where each retained organisation acted upon at least one bug that was reopened. The first measure separates the skewed frequency distribution of reopening tendencies into organisation-aggregate-roles that have any reopenings vs. those organisation-aggregate-roles that do not have any reopenings. The second measure focusses in on the subset of organisation-aggregate-roles with reopenings and examines the comparative percentage distribution amongst them. Taken together, these measures triangulate organisation-level reopening tendencies in a manner that is best suited to the observed frequency distributions in the data. Table 28 summarises the definitions of the variables that constitute the third and fourth measures of solution knowledge emergence at the organisation level, collectively referred to as “reopening tendencies”.

Reopening tendencies captured	Variable type	Measure
For each organisation, was at least one bug acted upon in the aggregate role “reporter” reopened?	Logical	3rd
For each organisation, was at least one bug acted upon in the aggregate role “assigned_to” reopened?	Logical	3rd
For each organisation, was at least one bug acted upon in the aggregate role “qa_contact” reopened?	Logical	3rd
For each organisation in the subset of organisations where at least one bug acted upon in the aggregate role “reporter” was reopened, what was the percentage of bugs that were reopened that were acted upon in the “reporter” aggregate role?	Non-zero percentage	4th
For each organisation in the subset of organisations where at least one bug acted upon in the aggregate role “assigned_to” was reopened, what was the percentage of bugs that were reopened that were acted upon in the “assigned_to” aggregate role?	Non-zero percentage	4th
For each organisation in the subset of organisations where at least one bug acted upon in the aggregate role “qa_contact” was reopened, what was the percentage of bugs that were reopened that were acted upon in the “qa_contact” aggregate role?	Non-zero percentage	4th

Table 28: Variables constituting reopening tendencies, the third and fourth measures of solution knowledge emergence at organisation level

The fifth and sixth measures of solution knowledge emergence at the organisation level capture the reassigning tendencies of bugs upon which each organisation acts in each aggregate role. Similar to at the individual level, during preliminary analysis of the frequency distributions, which proved to be non-linear, it was found that the best representation of these measures was six variables that triangulate the measures in a manner similar to measures three and four. The result was three logical variables separating the rare cases of reassignments occurring at all and three non-zero percentage variables distinguishing amongst those organisation-aggregate-roles that had reassignments. Table 29 summarises the definitions of the variables that constitute the fifth and sixth measures of solution knowledge emergence at the organisation level, collectively referred to as “reassigning tendencies”.

Reassigning tendencies captured	Variable type	Measure
For each organisation, was at least one bug acted upon in the aggregate role "reporter" reassigned?	Logical	5th
For each organisation, was at least one bug acted upon in the aggregate role "assigned_to" reassigned?	Logical	5th
For each organisation, was at least one bug acted upon in the aggregate role "qa_contact" reassigned?	Logical	5th
For each organisation in the subset of organisations where at least one bug acted upon in the aggregate role "reporter" was reassigned, what was the percentage of bugs that were reassigned that were acted upon in the "reporter" aggregate role?	Non-zero percentage	6th
For each organisation in the subset of organisations where at least one bug acted upon in the aggregate role "assigned_to" was reassigned, what was the percentage of bugs that were reassigned that were acted upon in the "assigned_to" aggregate role?	Non-zero percentage	6th
For each organisation in the subset of organisations where at least one bug acted upon in the aggregate role "qa_contact" was reassigned, what was the percentage of bugs that were reassigned that were acted upon in the "qa_contact" aggregate role?	Non-zero percentage	6th

Table 29: Variables constituting reassigning tendencies, the fifth and sixth measures of solution knowledge emergence at organisation level

The seventh measure of solution knowledge emergence at the organisation level is the mean time to resolution for bugs acted upon in each aggregate role by each organisation. This measure complements the problem and individual level time to resolution measures by examining the average resolution times at the organisation level as the literature suggests that different organisations may have implicit factors that affect time to resolution independent from the problems and individuals themselves. The measure was calculated by taking the subset of all bugs acted upon in each aggregate role by each organisation and taking the average time to resolution for each one, resulting in three variables: one for each aggregate role. Figure 26 summarises the operationalizations of the measures of the dependent variable of interest, solution knowledge emergence, at the organisation level of analysis.

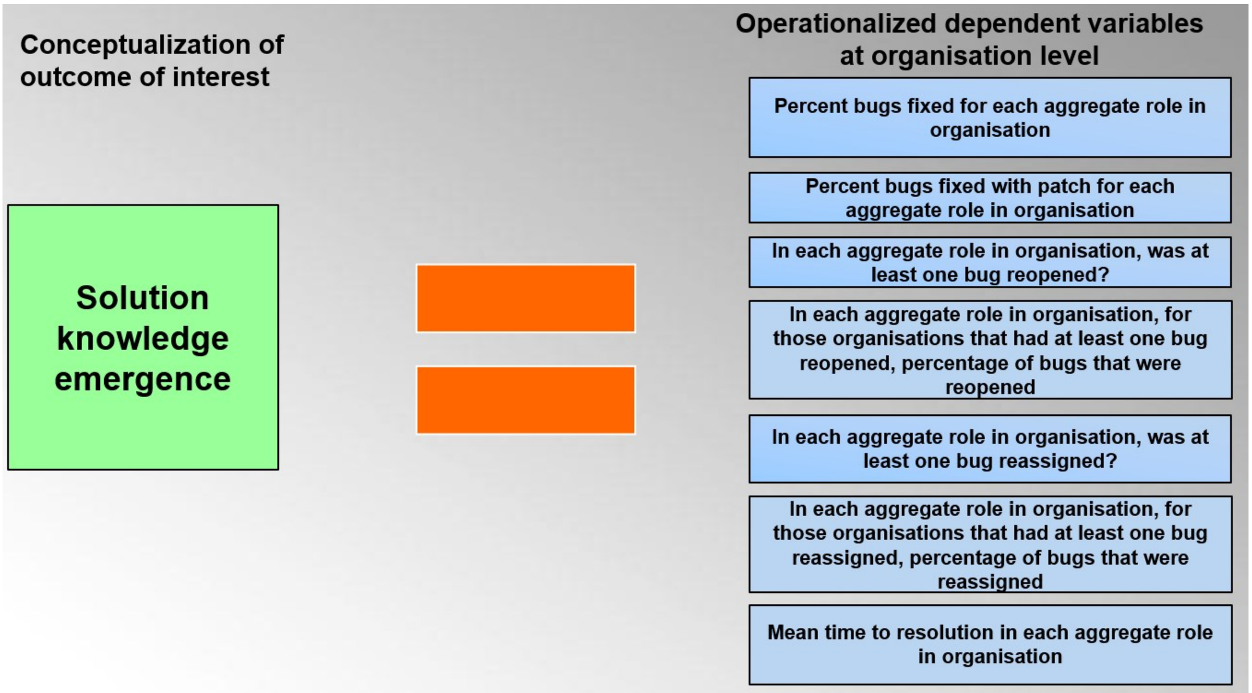

Figure 26: Operationalizations of measures of dependent variable of interest at organisation level

Organisation level operationalization: Independent variables

At the organisation level, in a manner similar to problem and individual levels, the independent variables were operationalized in line with each of the six hypotheses that were formulated for the conceptualizations of the antecedents of interest. Each operationalization can be conceptualised as a distinct organisation level measure used to triangulate the overall conceptualisations derived from the KBV and open source literatures as well as a testable sub-hypothesis with each of the above-discussed measures as antecedent independent variables.

Absorptive capacity

The first antecedent is absorptive capacity. At the organisation level of analysis, similar to the individual level, it was triangulated with five measures derived from the open source and KBV literatures. The first measure was number of activities performed by each organisation.

Activities, tracked in the “activities” table, reflect actions taken by organisations during the knowledge creation process. These actions are the same actions taken by individuals, but, at the organisation level of analysis, may have been taken by different individuals in the same organisation, creating a distinct numerical aggregation for each organisation in the data. The present measure, represented by a count variable of all the activities performed by each organisation, captures the notion in the KBV literature that those organisations that are very active in the meta-organisation may have lower absorptive capacity to engage new problem knowledge (Cohen & Levinthal, 1990; Zahra & George, 2002; Jansen, Van Den Bosch, & Volberda, 2005; Todorova & Durisin, 2007). It is therefore hypothesized that number of activities performed by an organisation is negatively correlated with solution knowledge emergence. Similar to the individual level, preliminary analysis revealed that the frequency distributions of activity counts amongst organisations were non-linear but were readily transformed into a form suitable for analysis with assumptions of linearity with a standard log transformation. As such, the present measure was analysed using a log-transformed count of activities variable using the form log10 (1+x) given that there are a large number of organisations with zero counts which would result in infinite logs confounding analysis (R Foundation, 2017).

The second measure of absorptive capacity at the organisation level complements the first measure by triangulating the notion in the KBV literature that absorptive capacity of organisations may be compartmentalised, such as, for example, by organisation department or function, and have differing levels according to categories of knowledge that the different organisation structures engage (Cohen & Levinthal, 1990; Lane & Lubatkin, 1998; Zahra & George, 2002; Jansen, Van Den Bosch, & Volberda, 2005; Schmidt, 2010; Spithoven, Clarysse,

& Knockaert, 2011). In the present case, the measure is captured with variables that count the number of activities of each organisation for each platform, operating system, and product classification, which are the major categories of knowledge as distinguished in the database. Similar to the individual level, preliminary analysis of the frequency distributions of activities separated into platform, operating system and classification revealed a non-linear distribution amongst organisations. Experimentation with transformations and consolidation revealed that there were several meta-categories that grouped the activity counts for platform, operating system, and product classification more evenly than their full categorical range permitted. The identified consolidated categories were the same as at the individual level, as summarised in Table 30.

Variables for each organisation
(log) Number of activities by organisation on bugs with platform "**All**"
(log) Number of activities by organisation on bugs with platform "**PowerPC**"
(log) Number of activities by organisation on bugs with platform "**x86**"
(log) Number of activities by organisation on bugs with platform "**x86 64-bit**"
(log) Number of activities by organisation on bugs with platform "**Other**"
(log) Number of activities by organisation on bugs with operating system "**Apple PC**"
(log) Number of activities by organisation on bugs with operating system "**Win PC**"
(log) Number of activities by organisation on bugs with operating system "**Win Mobile**"
(log) Number of activities by organisation on bugs with operating system "**Apple Mobile**"
(log) Number of activities by organisation on bugs with operating system "**Other PC**"
(log) Number of activities by organisation on bugs with operating system "**Other Mobile**"
(log) Number of activities by organisation on bugs with classification "**Client software**"
(log) Number of activities by organisation on bugs with classification "**Server software**"
(log) Number of activities by organisation on bugs with classification "**Components**"
(log) Number of activities by organisation on bugs with classification "**Other**"

Table 30: Variables capturing activities of organisations according to knowledge categories

The third measure of absorptive capacity at the organisation level is the number of activities performed by each organisation separated according to severity level. This measure complements the previous two measures by capturing the notion in the literature that absorptive capacity may be a function of prioritization of the production of certain types of knowledge, reflected by the "severity" measure in the present database, by organisations. As with the previous measures, given that the frequency distribution of problems organized according to severity was non-linear, the initial 7 categories of severity were consolidated into 3 meta-categories that enabled more even comparative analysis: "low", "average", and "high" severity. As with the previous measures, the distributions of the consolidated categories were found to be log-linear, so the log transform of each variable was taken. The organisation level consolidated severity variables were the same as at the individual level, as previously summarised in Table 12.

The fourth measure of absorptive capacity at the organisation level is the number of activities performed by each organisation separated according to activity type. This measure complements the previous measures by recognizing that the type of activity taken may have a separate level of absorptive capacity than factors related to the problems upon which the activity was done, particularly in the case of organisations with multiple departments or functions participating in the meta-organisation. Organisations can engage in the same 15 types of activities that individual can undertake, previously summarised in Table 13. As was the case at the individual level, preliminary analysis revealed a log-linear distribution for the activities amongst organisations. As a result, each of these count variables was log transformed in a manner similar to the previous measures.

The fifth measure of absorptive capacity at the organisation level is the number of times each organisation acted in each of the three aggregate roles, "problem knowledge producer", "solution knowledge producer" and "solution knowledge verifier". Organisations can act upon problems in the knowledge production process in three different aggregate roles. Much like individuals, some organisations will engage in more than one aggregate role at a time, whereas other organisation will only engage in one aggregate role. The present measure captures the notion that absorptive capacity for organisation may vary according to aggregate role involvement, which is often more nuanced and distinct at the organisation level than at the

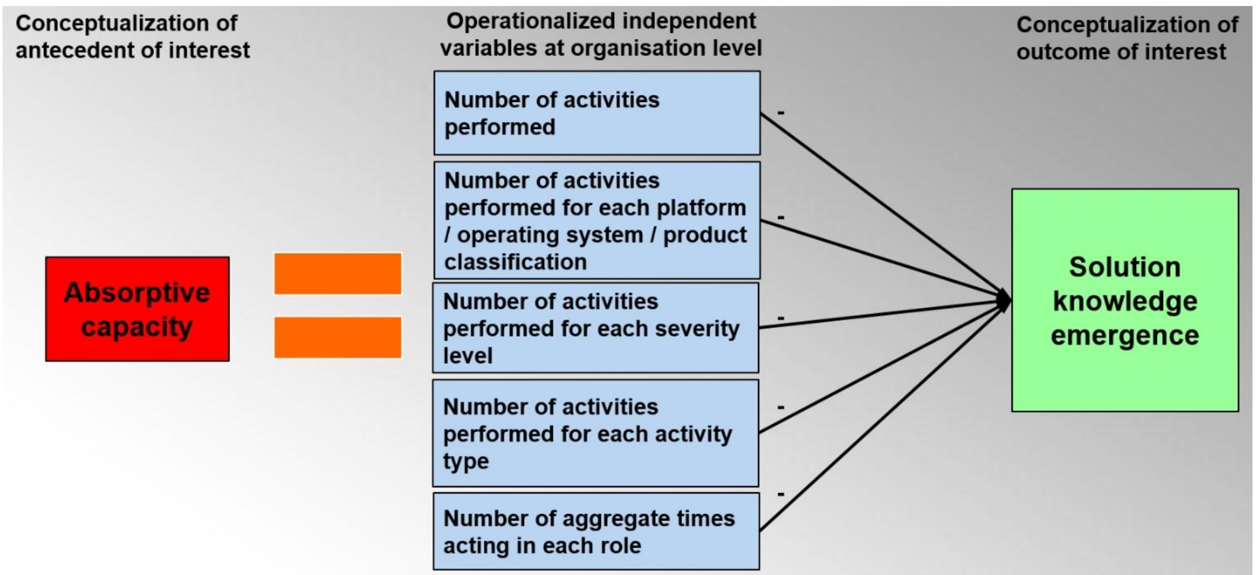

Figure 27: Operationalizations of measures of absorptive capacity at organisation level

individual level by virtue of the aggregation of roles. Preliminary analysis revealed the frequency distribution of actions in each aggregate role amongst organisations to be log-linear, so the three variables were log transformed in a manner similar to the previous measures. Figure 27 summarises the operationalizations of the measures of absorptive capacity at the organisation level as well as their hypothesised direction of influence on the dependent outcome of interest, solution knowledge emergence.

Codifiability

The second antecedent is codifiability. At the organisation level of analysis, it was triangulated using five measures derived from the literature. The first measure was the mean description length of the bugs acted upon in each aggregate role. Much like at the individual level, for each organisation, the mean description length of the problems acted upon in each aggregate role was calculated, resulting in three variables per organisation. These variables complemented their counterparts at the other letters by helping localise the level of codifiability factors. Much like at the individual level, title length had insufficient variability for appropriate analysis. As such the average title length was not included in this codifiability measure at the organisation level. As was the case with previous measures, the frequency distribution was observed to be log-linear, so the log transform was done on each variable to permit analysis with assumptions of linearity. Table 31 summarises the variables created for each organisation to calculate the mean description length of problems acted upon in each aggregate role by each organisation.

Mean description length variables
(log) Mean length of descriptions of problems acted upon in aggregate role of problem knowledge producer (**reporter**) by organisation
(log) Mean length of descriptions of problems acted upon in aggregate role of solution knowledge producer (**assigned_to**) by organisation
(log) Mean length of descriptions of problems acted upon in aggregate role of solution knowledge verifier **(qa_contact)** by organisation

Table 31: Variables capturing mean description length for problems acted upon in each aggregate role by organisations

The second measure of codifiability at the organisation level was the mean of the readability measures of the descriptions of the problems acted upon by organisations in each of the three aggregate roles. Much like at the individual level, the Flesch reading ease readability formula was selected as its value can be meaningfully averaged to calculate a mean organisation level value that represents the readability associated with the full range of descriptions of problems upon which each organisation has acted in each aggregate role. Preliminary analysis of the resulting means suggested a distribution that is sufficiently linear for analysis without transformations. Table 32 summarises the variables created for each organisation to calculate the mean description readability of problems acted upon in each aggregate role by each organisation.

Mean description readability variables
Mean Flesch reading ease of descriptions of problems acted upon in aggregate role of problem knowledge producer (**reporter**) by organisation
Mean Flesch reading ease of descriptions of problems acted upon in aggregate role of solution knowledge producer (**assigned_to**) by organisation
Mean Flesch reading ease of descriptions of problems acted upon in aggregate role of solution knowledge verifier **(qa_contact)** by organisation

Table 32: Variables capturing mean description readability of problems acted upon in each aggregate role by organisations

The third measure of codifiability at the organisation level is the number of attachments to problems acted upon in each aggregate role by organisations' members. Much like at the problem and individual levels, the number of attachments to problems captures the notion that information enabling codifiability may reside at the organisation level as well as at other level. Like at the individual level, preliminary examination of the types of attachments at the organisation level revealed insufficient variability for analysis. As such, a single variable capturing the number of attachments of any type for each aggregate role played by organisations in acting upon problems was created. The choice was made at the organisation level to take the "count" of attachments related to each aggregate role rather than the mean, which would otherwise be the mean of the mean variable from the individual level of analysis that would not introduce sufficient variability at the organisation level relative to the individual level of analysis. A log transformation was taken of the “count” variable to induce sufficient normality for analysis. This choice was made to present an alternate view of the variable at the organisation level of analysis to compare and contrast the logarithmic nature of the attachment counts to the measures of central tendencies of the variable. Table 33 summarises the variables created for each organisation to calculate the (log) number of attachments to problems acted upon in each aggregate role by organisations’ members.

Attachment count variables
(log) Number of attachments to problems acted upon in aggregate role of problem knowledge producer (**reporter**) by organisations’ members
(log) Number of attachments to problems acted upon in aggregate role of solution knowledge producer (**assigned_to**) by organisations’ members
(log) Number of attachments to problems acted upon in aggregate role of solution knowledge verifier (**qa_contact)** by organisations’ members

Table 33: Variables capturing (log) number of attachments to problems acted upon in each aggregate role by organisations’ members

The fourth measure of codifiability at the organisation level is the redundancy tendencies of the problem knowledge submitted by each organisation. Much like at the individual level, the present measure captures the notion that redundancy in the knowledge available for codification can take place at the organisation level and organisation level effects may lead to problem knowledge that is a duplicate of or duplicated by other problem knowledge. This measure only makes sense from the perspective of the aggregate role of problem knowledge producer as the other two aggregate roles, solution knowledge producer and solution knowledge verifier do not synthesize the initial problem knowledge that could potentially be a duplicate or duplicated by other problem knowledge. As a result, two variables encapsulate this measure: The percentage of problem knowledge reports relative to all problem knowledge reports each organisation submits that were identified as duplicates to other problem knowledge reports; and, the percentage of problem knowledge reports relative to all problem knowledge reports each organisation submits that were duplicated by other problem reports.

The fifth measure of codifiability at the organisation level is the number and length of comments attached to problems acted upon in each aggregate role by organisations. Much as their counterparts at the problem and individual levels, the organisation level number of comments and mean comment length aim to capture the notion that the codification enabling additional information submitted via comments to supplant the initially submitted problem knowledge may have organisation level effects. Therefore, three variables were created to capture the comment length amongst problems and three variables to capture comment count amongst problems, one for each aggregate role in which organisations' members engage. In a manner similar to the attachment variables, the choice was made at the organisation level to take the "count" of manually submitted comments rather than the mean, which would otherwise be the mean of the mean variable from the individual level of analysis that would not introduce sufficient variability at the organisation level relative to the individual level of analysis. Manually submitted comments represent a portion of the total comments that are appended to problem knowledge. Given that automated comments are typically generated from the information that already is contained in the problem knowledge, the manual comments are more likely to add new information that affects codifiability, as per the hypotheses. The choice was made to present an alternate view of the variable at the organisation level of analysis to compare and contrast these variable representations. A log transformation was done on each variable to enable analysis with assumptions of linearity in a manner similar to previous measures. Table 34 summarises the six variables triangulating the tendencies of comments attached to problems acted upon in each of the aggregate roles undertaken by organisations.

Comment tendency variables
(log) Mean length of comments on problems acted upon in aggregate problem knowledge producer role (**reporter**) by organisations' members
(log) Mean length of comments on problems acted upon in aggregate solution knowledge producer role (**assigned_to**) by organisations' members
(log) Mean length of comments on problems acted upon in aggregate solution knowledge verifier role (**qa_contact**) by organisations' members
(log) Number of manual comments on problems acted upon in aggregate problem knowledge producer role (**reporter**) by organisations' members
(log) Number of manual comments on problems acted upon in aggregate solution knowledge producer role (**assigned_to**) by organisations' members
(log) Number of manual comments on problems acted upon in aggregate solution knowledge verifier role (**qa_contact**) by organisations' members

Table 34: Variables capturing tendencies of comments on problems acted upon in each aggregate role by organisations' members

Collectively, these measures triangulate the concept of codifiability in terms of organisation level effects. Figure 28 summarises the operationalizations of the measures of codifiability at the organisation level as well as their hypothesised direction of influence on the dependent outcome of interest, solution knowledge emergence.

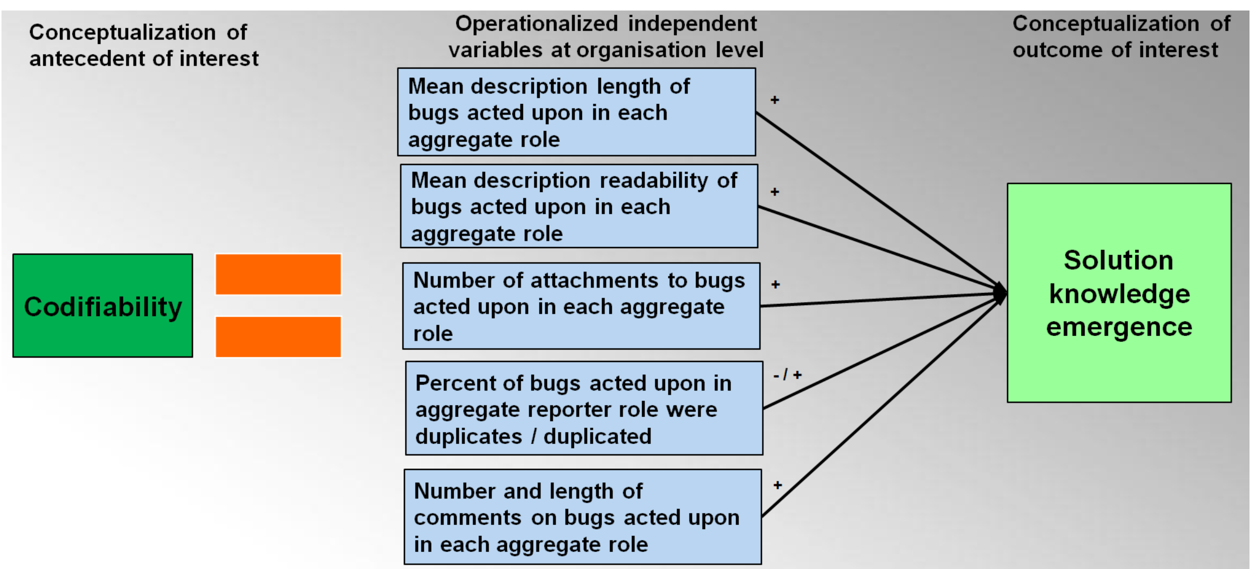

Figure 28: Operationalizations of measures of codifiability at organisation level

Dominant knowledge paradigm

The third antecedent of interest is dominant knowledge paradigm. It was triangulated at the organisation level using three measures derived from the literature. Much like at the individual level, whereas five measures were used at the problem level, given the large number of products and components in the database, it was impractical to create all variable permutations at the organisation level as the frequencies were too low for comparative analysis amongst organisations.

The first measure was percent of actions by organisations in each aggregate role upon bugs of each type of platform. Similar to the individual level, the platforms were consolidated into five categories, “PowerPC”, “x86”, “x86 64-bit”, “all”, and “other”, to more evenly distribute them for comparative analysis. The “other” category was maintained as the reference category during analysis. The result was 12 variables per organisation (3 aggregate roles times 4 platforms) making up this platform tendency measure. Table 35 summarises the variables for the platform dominant knowledge paradigm measure for each aggregate role engaged in by organisations.

The second measure was percent of actions in each aggregate role done on bugs of each type of operating system. Much like the previous measure, this measure captures the organisation level knowledge paradigm tendencies in each aggregate role for each operating system. Like at the individual level, the operating systems were consolidated into eight categories to improve frequency distributions for comparative analysis: “Android”, “Linux”,

Aggregate role	Platform	Variable
Problem knowledge producer (reporter)	All	% bugs acted upon by organisation in aggregate role of reporter with platform All
	PowerPC	% bugs acted upon by organisation in aggregate role of reporter with platform PowerPC
	x86	% bugs acted upon by organisation in aggregate role of reporter with platform x86
	x86_64	% bugs acted upon by organisation in aggregate role of reporter with platform x86 64-bit
Solution knowledge producer (assigned_to)	All	% bugs acted upon by organisation in aggregate role of assigned_to with platform All
	PowerPC	% bugs acted upon by organisation in aggregate role of assigned_to with platform PowerPC
	x86	% bugs acted upon by organisation in aggregate role of assigned_to with platform x86
	x86_64	% bugs acted upon by organisation in aggregate role of assigned_to with platform x86 64-bit
Solution knowledge verifier (qa_contact)	All	% bugs acted upon by organisation in aggregate role of qa_contact with platform All
	PowerPC	% bugs acted upon by organisation in aggregate role of qa_contact with platform PowerPC
	x86	% bugs acted upon by organisation in aggregate role of qa_contact with platform x86
	x86_64	% bugs acted upon by organisation in aggregate role of qa_contact with platform x86 64-bit

Table 35: Variables capturing platform dominant knowledge paradigm measure for each aggregate role at organisation level

"Apple PC", "Windows PC", "Apple Mobile", "Windows Mobile", "Other PC", and "Other Mobile". The "Other PC" category was held as the reference category during analysis. The result was 24 variables per organisation (3 aggregate roles times 8 operating systems) making up this operating systems tendency measure. Table 36 summarises the variables capturing the operating system dominant knowledge paradigm measure for each aggregate role at the organisation level.

The third measure was percent of actions in each aggregate role upon bugs of each type of classification. As with the previous measures, this measure captures the organisation level

Aggregate role	Operating system	Variable
Problem knowledge producer (reporter)	All	% bugs acted upon by organisation in aggregate role of reporter for operating system All
	Android	% bugs acted upon by organisation in aggregate role of reporter for operating system Android
	Linux	% bugs acted upon by organisation in aggregate role of reporter for operating system Linux
	Apple PC	% bugs acted upon by organisation in aggregate role of reporter for operating system Apple PC
	Windows PC	% bugs acted upon by organisation in aggregate role of reporter for operating system Windows PC
	Apple Mobile	% bugs acted upon by organisation in aggregate role of reporter for operating system Apple Mobile
	Windows Mobile	% bugs acted upon by organisation in aggregate role of reporter for operating system Windows Mobile
	Other Mobile	% bugs acted upon by organisation in aggregate role of reporter for operating system Other Mobile
Solution knowledge producer (assigned_to)	All	% bugs acted upon by organisation in aggregate role of assigned_to for operating system All
	Android	% bugs acted upon by organisation in aggregate role of assigned_to for operating system Android
	Linux	% bugs acted upon by organisation in aggregate role of assigned_to for operating system Linux
	Apple PC	% bugs acted upon by organisation in aggregate role of assigned_to for operating system Apple PC
	Windows PC	% bugs acted upon by organisation in aggregate role of assigned_to for operating system Windows PC
	Apple Mobile	% bugs acted upon by organisation in aggregate role of assigned_to for operating system Apple Mobile
	Windows Mobile	% bugs acted upon by organisation in aggregate role of assigned_to for operating system Windows Mobile
	Other Mobile	% bugs acted upon by organisation in aggregate role of assigned_to for operating system Other Mobile
Solution knowledge verifier (qa_contact)	All	% bugs acted upon by organisation in aggregate role of qa_contact for operating system All
	Android	% bugs acted upon by organisation in aggregate role of qa_contact for operating system Android
	Linux	% bugs acted upon by organisation in aggregate role of qa_contact for operating system Linux
	Apple PC	% bugs acted upon by organisation in aggregate role of qa_contact for operating system Apple PC
	Windows PC	% bugs acted upon by organisation in aggregate role of qa_contact for operating system Windows PC
	Apple Mobile	% bugs acted upon by organisation in aggregate role of qa_contact for operating system Apple Mobile

	Windows Mobile	% bugs acted upon by organisation in aggregate role of qa_contact for operating system Windows Mobile
	Other Mobile	% bugs acted upon by organisation in aggregate role of qa_contact for operating system Other Mobile

Table 36: Variables capturing operating system dominant knowledge paradigm measure for each aggregate role at organisation level

knowledge paradigm tendencies in each aggregate role for each classification. Like at the individual level, the classifications were consolidated into "Client Software", "Server Software", "Components", and "Other", with the "Other" category held as the reference category during analysis. The result was 9 variables per organisation (3 aggregate roles times 3 classifications) making up this classification tendency measure. Table 37 summarises the variables capturing the classification dominant knowledge paradigm measure for each aggregate role at the organisation level.

Aggregate role	Classification	Variable
Problem knowledge producer (reporter)	Client Software	% bugs acted upon by organisation in aggregate role of reporter for classification Client Software
	Server Software	% bugs acted upon by organisation in aggregate role of reporter for classification Server Software
	Component	% bugs acted upon by organisation in aggregate role of reporter for classification Component
Solution knowledge producer (assigned_to)	Client Software	% bugs acted upon by organisation in aggregate role of assigned_to for classification Client Software
	Server Software	% bugs acted upon by organisation in aggregate role of assigned_to for classification Server Software
	Component	% bugs acted upon by organisation in aggregate role of assigned_to for classification Component
Solution knowledge verifier (qa_contact)	Client Software	% bugs acted upon by organisation in aggregate role of qa_contact for classification Client Software
	Server Software	% bugs acted upon by organisation in aggregate role of qa_contact for classification Server Software
	Component	% bugs acted upon by organisation in aggregate role of qa_contact for classification Component

Table 37: Variables capturing classification dominant knowledge paradigm measure for each aggregate role at organisation level

Collectively, these measures triangulate the concept of dominant knowledge paradigm in terms of organisation level effects. Figure 29 summarises the operationalizations of the measures of dominant knowledge paradigm at the organisation level as well as their hypothesised direction of influence on the dependent outcome of interest, solution knowledge emergence.

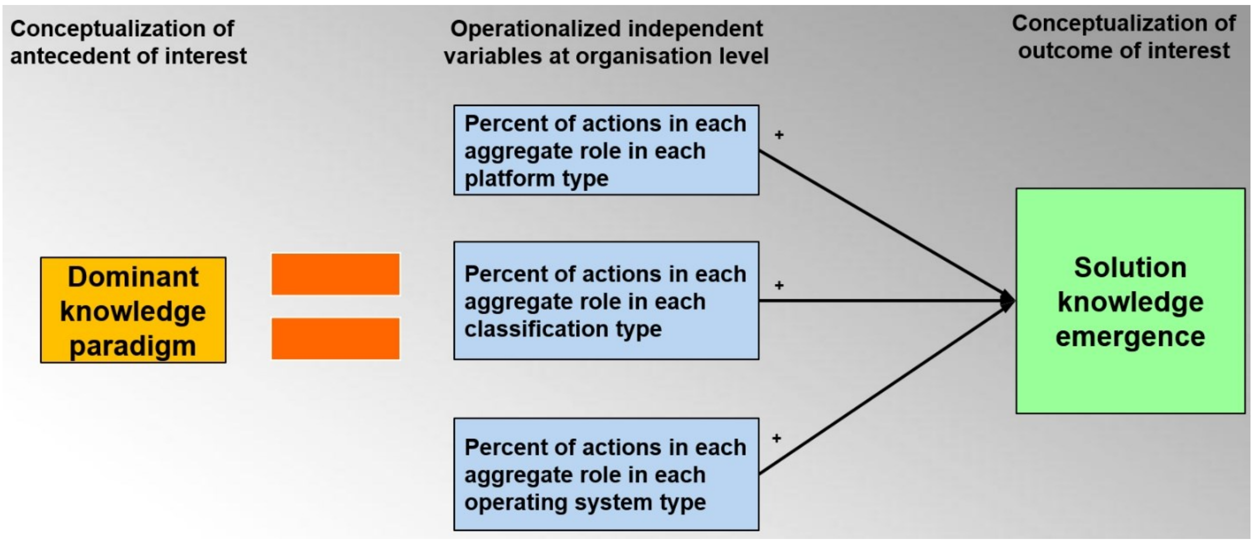

Figure 29: Operationalizations of measures of dominant knowledge paradigm at organisation level

Knowledge flow impediments

The fourth antecedent of interest is knowledge flow impediments. It was triangulated at the organisation level using six measures derived from the literature. The first measure was percent of bugs acted upon by organisations in each aggregate role that violated the bug life cycle. Similar to at the individual level, 3 percentages variables were created, one for each aggregate role, for each organisation, representing violation of bug life cycle by the bugs acted upon. Necessarily, the percentage of bugs acted upon in each aggregate role by organisations which did not violate the bug life cycle was the reference category for analysis.

The second measure was percent of bugs acted upon in each aggregate role whose target milestone was changed at least once. This measure captures the organisation level tendencies of bugs acted upon by organisations in each of the three aggregate roles. Examination of tendencies for target milestone changes for bugs acted upon in each aggregate role revealed that the comparative frequency distribution at the organisation level was best represented by the percentage of bugs acted upon in each aggregate role where "target milestone changed at least once" and "target milestone never changed", the latter being the reference category for analysis. As with the previous measure, the result was 3 percentages, one for each aggregate role, for each organisation.

The third measure was percent of bugs acted upon in each aggregate role whose severity was changed at least once. Like the previous measure, this measure captures organisation level severity change tendencies in each of the three aggregate roles in which organisations act. Similarly, the comparative frequency distribution at the organisation level was best represented as percentage of bugs acted upon in each aggregate ole where "severity changed at least once" and "severity never changed", the latter being the reference category for analysis. Three percentages variables were created for each organisation for this measure, one for each aggregate role.

The fourth and fifth measures were percent of bugs acted upon in each aggregate role by organisations which were reopened or reassigned at least once. As with the previous measures, the comparative frequency distributions at the organisation level were best represented as percentage of bugs acted upon by organisations in each aggregate role where "bug was reopened/reassigned at least once" and "bug was never reopened/reassigned", the latter being the reference categories for analysis. Three percentage variables were created for each measure, one for each aggregate role.

The sixth measure was number of activities taking place on each bug acted upon by organisations in each aggregate role within certain time frames. This measure seeks to capture the organisation level counterpart of the problem and individual level examination of activities on problems. Similar to at the individual level, the goal was to establish the activity tendencies for all problems acted upon in each aggregate role by each organisation. Preliminary frequency distribution analysis of the mean number of activities occurring in various ranges of time on each problem in each aggregate role revealed a log-linear relationship at thresholds comparable to those used at the problem and individual levels. The result was the creation of 21 variables for each organisation, 7 for each of the three aggregate roles in which organisations engage, that capture the log of the mean number of activities taking place in each time range depicted in Table 38.

Collectively, these measures triangulate the concept of knowledge flow impediments in terms of organisation level effects. Figure 30 summarises the operationalizations of the measures of knowledge flow impediments at the organisation level as well as their hypothesised direction of influence on the dependent outcome of interest, solution knowledge emergence.

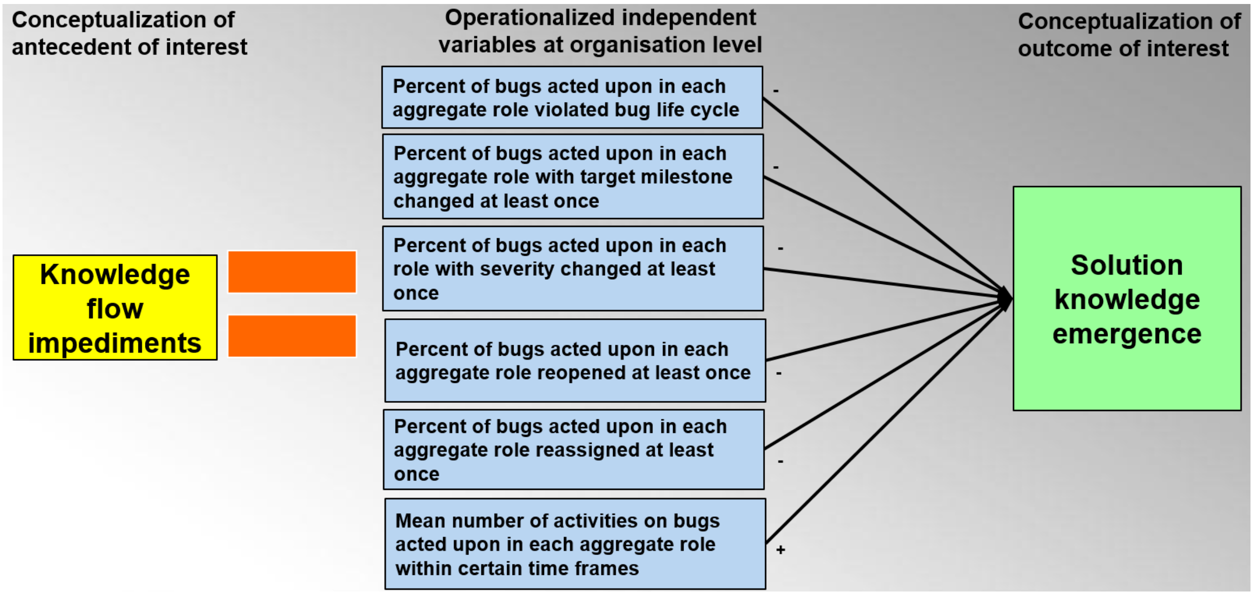

Figure 30: Operationalizations of measures of knowledge flow impediments at organisation level

Aggregate role	**Time range** (days)	**Variable**
Problem knowledge producer (reporter)	1 < t <= 3	(log) mean number of activities from 1 to 3 days after creation of bugs acted on in aggregate reporter role by organisation
	3 < t <= 7	(log) mean number of activities from 3 to 7 days after creation of bugs acted on in aggregate reporter role by organisation
	7 < t <= 15	(log) mean number of activities from 7 to 15 days after creation of bugs acted on in aggregate reporter role by organisation
	15 < t <= 45	(log) mean number of activities from 15 to 45 days after creation of bugs acted on in aggregate reporter role by organisation
	45 < t <= 90	(log) mean number of activities from 45 to 90 days after creation of bugs acted on in aggregate reporter role by organisation
	90 < t <= 180	(log) mean number of activities from 90 to 180 days after creation of bugs acted on in aggregate reporter role by organisation
	180 < t <= 365	(log) mean number of activities from 180 to 365 days after creation of bugs acted on in aggregate reporter role by organisation
Solution knowledge producer (assigned_to)	1 < t <= 3	(log) mean number of activities from 1 to 3 days after creation of bugs acted on in aggregate assigned_to role by organisation
	3 < t <= 7	(log) mean number of activities from 3 to 7 days after creation of bugs acted on in aggregate assigned_to role by organisation
	7 < t <= 15	(log) mean number of activities from 7 to 15 days after creation of bugs acted on in aggregate assigned_to role by organisation
	15 < t <= 45	(log) mean number of activities from 15 to 45 days after creation of bugs acted on in aggregate assigned_to role by organisation
	45 < t <= 90	(log) mean number of activities from 45 to 90 days after creation of bugs acted on in aggregate assigned_to role by organisation
	90 < t <= 180	(log) mean number of activities from 90 to 180 days after creation of bugs acted on in aggregate assigned_to role by organisation
	180 < t <= 365	(log) mean number of activities from 180 to 365 days after creation of bugs acted on in aggregate assigned_to role by organisation
Solution knowledge	1 < t <= 3	(log) mean number of activities from 1 to 3 days after creation of bugs acted on in aggregate qa_contact role by

verifier (qa_contact)		organisation
	3 < t <= 7	(log) mean number of activities from 3 to 7 days after creation of bugs acted on in aggregate qa_contact role by organisation
	7 < t <= 15	(log) mean number of activities from 7 to 15 days after creation of bugs acted on in aggregate qa_contact role by organisation
	15 < t <= 45	(log) mean number of activities from 15 to 45 days after creation of bugs acted on in aggregate qa_contact role by organisation
	45 < t <= 90	(log) mean number of activities from 45 to 90 days after creation of bugs acted on in aggregate qa_contact role by organisation
	90 < t <= 180	(log) mean number of activities from 90 to 180 days after creation of bugs acted on in aggregate qa_contact role by organisation
	180 < t <= 365	(log) mean number of activities from 180 to 365 days after creation of bugs acted on in aggregate qa_contact role by organisation

Table 38: Variables capturing time-based activity tendency measures for problems acted upon in each aggregate role by each organisation at organisation level

Knowledge stakeholder influence

The fifth antecedent of interest is knowledge stakeholder influence. It was triangulated at the organisation level using four measures derived from the literature. The first measure of knowledge stakeholder influence is the degree of involvement in the meta-organisation of each organisation's members. Whereas at the individual level, each profile was flagged as "core", "participant", or "peripheral" knowledge actors, at the organisation level, given that each organisation is made up of multiple individual members, the complementary measures are the count and percentage of members of each degree of involvement in each organisation. This measure complements the individual and problem level measures to allow better localisation of any effects at the correct level of analysis.

Five variables were created for each profile making up this measure. Three count variables, log transformed to fit assumptions of linearity during analysis, were created for each organisation, reflecting the number of members who were core, participant, and peripheral knowledge actors respectively. Two percentage variables, the percentage of organisation members who were core knowledge actors and the percentage of organisation members who were participant knowledge actors were also created. The percentage of organisation members that were peripheral knowledge actors was held as the reference category for analysis as the percentages necessarily always add to 100%, precluding linear regression analysis.

The second measure of knowledge stakeholder influence is the tendencies related to following, voting, and acting by each of the three classes of knowledge actors on bugs acted upon in each of the three aggregate roles in which each organisation engages. The definitions of the three classes of knowledge actors, core knowledge actor, knowledge flow participant actor, and peripheral knowledge actor, are the same as at the problem and individual levels of analysis. The present measure complements the problem and individual level measures in order to localise the level of any effects related to knowledge stakeholder influence in the following, voting, and acting tendencies. The measure was operationalized with 27 variables, 9 per aggregate role in which each organisation engages. Three measures capture the voting tendencies for core, participant, and peripheral actors on bugs acted upon by organisation in each aggregate role. Three measures capture the following tendencies of core, participant, and peripheral knowledge actors on bugs acted upon by organisations in each aggregate role. And, three measures capture the acting tendencies of core, participant, and peripheral knowledge actors on bugs acted upon by organisations in each aggregate role. The log transform of each of these count variables was taken to meet assumptions of linearity for analysis. The choice was made to take log-transformed count variables at the organisation level rather than means of the means presented at the individual level that would have had insufficient distinctiveness at the organisation level in a manner similar to that discussed for previous variables. Table 39 summarises the variables that capture the knowledge stakeholder influence related activity tendencies at the individual level.

The third measure of knowledge stakeholder influence is the mean number of distinct actors commenting and acting upon bugs acted upon in each of the three aggregate roles in which organisations engage. Given that actors in the meta-organisation can comment and act upon problems more than once, the present measure complements the mean measures of the individual level of analysis by considering the averages of distinct actors commenting and acting upon bugs that organisations have acted upon in each of the three aggregate roles. Comparison of the present and previous measures allows a separation of the effects due to a small number of actors engaging problems multiple times from the effects of a large number of actors engaging problems a few times. It also allows separation of individual involvement effects (core vs. peripheral) from organisation level effects. Six variables were created, two for each aggregate role for each organisation. The log transform of the variables was taken to meet assumptions of linearity for analysis.

The fourth measure of knowledge stakeholder influence is the count and degree of involvement of individuals and organisations each organisation is observing and observed by. At the organisation level, each organisation can watch and be watched by other profiles and organisations. Eight variables triangulate the measure, with four variables capturing the (log) count of distinct actors, distinct organisations, only core knowledge actors, and only participant knowledge actors, watched by each organisation. Four variables capture the (log) count of distinct actors, distinct organisations, only core knowledge actors, and only participant knowledge actors who are watching each organisation. The count of peripheral knowledge actors watching and watched by each organisation is held as the reference category because the definition of "peripheral knowledge actor" necessarily implies a lack of involvement in the knowledge production process, precluding visibility to the meta-organisation's members that is necessary to populate the watching/watched_by count variables. Table 40 summarises the variables capturing knowledge stakeholder observational influence measures at the organisation level.

Aggregate role	Tendency	Actor	Variable
Problem knowledge producer (reporter)	Votes	Core	(log) Count of votes by core actors on bugs acted on in aggregate reporter role by organisation
	Votes	Participant	(log) Count of votes by participant actors on bugs acted on in aggregate reporter role by organisation
	Votes	Peripheral	(log) Count of votes by peripheral actors on bugs acted on in aggregate reporter role by organisation
	Following	Core	(log) Count of follows by core actors on bugs acted on in aggregate reporter role by organisation
	Following	Participant	(log) Count of follows by participant actors on bugs acted on in aggregate reporter role by organisation
	Following	Peripheral	(log) Count of follows by peripheral actors on bugs acted on in aggregate reporter role by organisation
	Acting	Core	(log) Count of actions by core actors on bugs acted on in aggregate reporter role by organisation
	Acting	Participant	(log) Count of actions by participant actors on bugs acted on in aggregate reporter role by organisation
	Acting	Peripheral	(log) Count of actions by peripheral actors on bugs acted on in aggregate reporter role by organisation
Solution knowledge producer (assigned_to)	Votes	Core	(log) Count of votes by core actors on bugs acted on in aggregate assigned_to role by organisation
	Votes	Participant	(log) Count of votes by participant actors on bugs acted on in aggregate assigned_to role by organisation
	Votes	Peripheral	(log) Count of votes by peripheral actors on bugs acted on in aggregate assigned_to role by organisation
	Following	Core	(log) Count of follows by core actors on bugs acted on in aggregate assigned_to role by organisation
	Following	Participant	(log) Count of follows by participant actors on bugs acted on in aggregate assigned_to role by organisation
	Following	Peripheral	(log) Count of follows by peripheral actors on bugs acted on in aggregate assigned_to role by organisation
	Acting	Core	(log) Count of actions by core actors on bugs acted on in aggregate assigned_to role by organisation
	Acting	Participant	(log) Count of actions by participant actors on bugs acted on in aggregate assigned_to role by organisation
	Acting	Peripheral	(log) Count of actions by peripheral actors on bugs acted on in aggregate assigned_to role by organisation
Solution knowledge verifier (qa_contact)	Votes	Core	(log) Count of votes by core actors on bugs acted on in aggregate qa_contact role by organisation
	Votes	Participant	(log) Count of votes by participant actors on bugs acted on in aggregate qa_contact role by organisation
	Votes	Peripheral	(log) Count of votes by peripheral actors on bugs acted on in aggregate qa_contact role by organisation
	Following	Core	(log) Count of follows by core actors on bugs acted on in aggregate qa_contact role by organisation

	Following	Participant	(log) Count of follows by participant actors on bugs acted on in aggregate qa_contact role by organisation
	Following	Peripheral	(log) Count of follows by peripheral actors on bugs acted on in aggregate qa_contact role by organisation
	Acting	Core	(log) Count of actions by core actors on bugs acted on in aggregate qa_contact role by organisation
	Acting	Participant	(log) Count of actions by participant actors on bugs acted on in aggregate qa_contact role by organisation
	Acting	Peripheral	(log) Count of actions by peripheral actors on bugs acted on in aggregate qa_contact role by organisation

Table 39: Variables capturing knowledge stakeholder influence activity tendency measures for problems acted upon in each aggregate role by each organisation at organisation level

Action	Actor	Variable
Watching	All actors	(log) Count of actors watching organisation
Watching	Organisations	(log) Count of organisations watching organisation
Watching	Core actors	(log) Count of core actors watching organisation
Watching	Participant actors	(log) Count of participant actors watching organisation
Watched by	All actors	(log) Count of actors watched by organisation
Watched by	Organisations	(log) Count of organisations watched by organisation
Watched by	Core actors	(log) Count of core actors watched by organisation
Watched by	Participant actors	(log) Count of participant actors watched by organisation

Table 40: Variables capturing knowledge stakeholder observational influence measures at organisation level

Collectively, these measures triangulate the concept of knowledge stakeholder influence in terms of organisation level effects. Figure 31 summarises the operationalizations of the measures of knowledge stakeholder influence at the organisation level as well as their hypothesised direction of influence on the dependent outcome of interest, solution knowledge emergence.

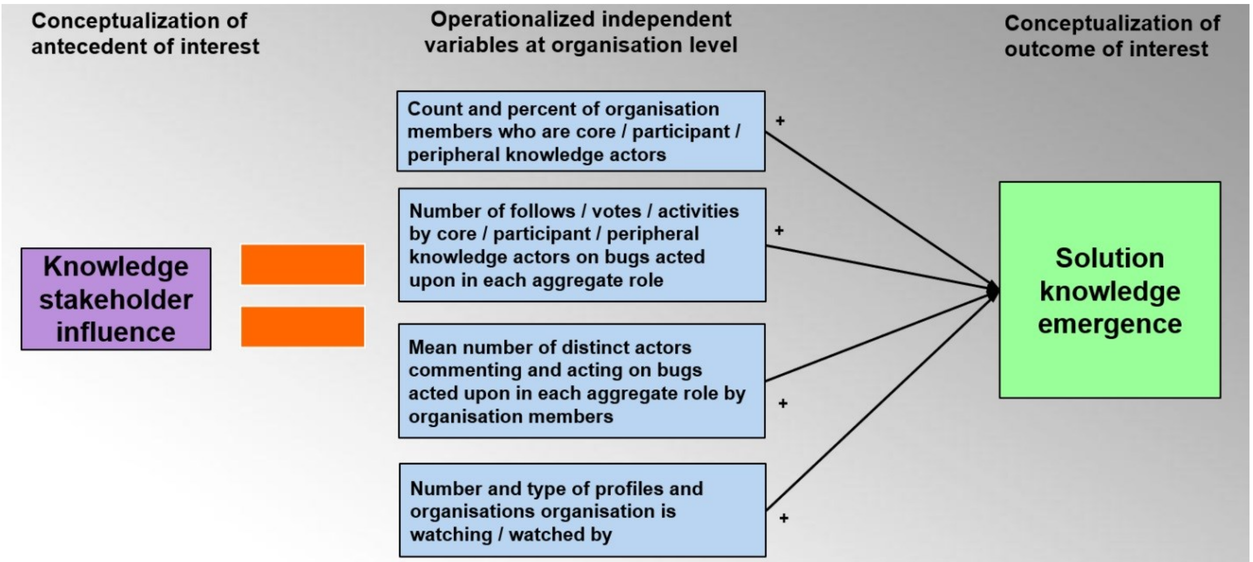

Figure 31: Operationalizations of measures of knowledge stakeholder influence at organisation level

Solution knowledge value

The sixth antecedent of interest is solution knowledge value. It was triangulated at the organisation level with five measures derived from the literature. The first measure was the tendencies of profiles to act upon bugs of differing severity levels in each of the 3 aggregate roles in which organisations engage. This measure complements its individual level counterpart by capturing the notion in the literature that solution knowledge value may be reflected at the organisation level in the tendencies to engage with problem knowledge that is classified at higher or lower severity levels. Tendencies to engage higher severity problem knowledge are hypothesized to promote solution knowledge emergence as it is theorized that solutions to those problems have greater value. Whereas at the individual level priority levels were also considered, given the inclusion constraints for data at the organisation level, there was insufficient variability in the priority levels for analysis. Therefore, the priority levels were excluded from analysis. Eighteen percentage variables were created to capture each of the tendencies to act upon bugs in each of the 6 severity levels from enhancement to blocker, with the severity "normal" held as the reference category for the purpose of analysis, in each of the 3 aggregate roles. The choice was made to change the reference category to "normal" at the organisation level of analysis instead of "enhancement" as used at the individual level of analysis because the exclusion of the "priority" categories at the organisation level of analysis warrants the direct examination of the enhancement category as a substitute as priorities are only defined for the enhancement category of severity. Further, it allows a contrasting of severity effects from level of analysis effects to better localize any observed outcome effects. Table 41 summarises variables that make up the solution knowledge value severity measure at organisation level.

The second measure of solution knowledge value was the tendencies of organisation to act upon bugs in each of the 3 aggregate roles in which organisation engage whose severity or priority level had changed at least once since the initial reveal of the problem knowledge to the meta-organisation. Preliminary analysis of the frequency distribution of percentages of bugs acted upon in each aggregate role with varying numbers of severity and priority changes revealed that there was only sufficient variability for analysis between "changed at least once" and "never changed", similar to the individual level. Instances of priority and severity changes occurring more than once, as previously observed at the problem and individual levels, were sufficiently infrequent as to be statistical outliers. As a result, the variables making up this measure at the organisation level were selected to match their counterparts at the problem and individual levels and focus on the tendencies to act upon bugs with or without the occurrence of severity or priority change rather than continuous counts of such changes.

Aggregate role	Severity	Variable
Problem knowledge producer (reporter)	Enhancement	% bugs acted on by organisation in aggregate role of reporter had severity enhancement
	Trivial	% bugs acted on by organisation in aggregate role of reporter had severity trivial
	Minor	% bugs acted on by organisation in aggregate role of reporter had severity minor
	Major	% bugs acted on by organisation in aggregate role of reporter had severity major
	Critical	% bugs acted on by organisation in aggregate role of reporter had severity critical
	Blocker	% bugs acted on by organisation in aggregate role of reporter had severity blocker
Solution knowledge producer (assigned_to)	Enhancement	% bugs acted on by organisation in aggregate role of assigned_to had severity enhancement
	Trivial	% bugs acted on by organisation in aggregate role of assigned_to had severity trivial
	Minor	% bugs acted on by organisation in aggregate role of assigned_to had severity minor
	Major	% bugs acted on by organisation in aggregate role of assigned_to had severity major
	Critical	% bugs acted on by organisation in aggregate role of assigned_to had severity critical
	Blocker	% bugs acted on by organisation in aggregate role of assigned_to had severity blocker
Solution knowledge verifier (qa_contact)	Enhancement	% bugs acted on by organisation in aggregate role of qa_contact had severity enhancement
	Trivial	% bugs acted on by organisation in aggregate role of qa_contact had severity trivial
	Minor	% bugs acted on by organisation in aggregate role of qa_contact had severity minor
	Major	% bugs acted on by organisation in aggregate role of qa_contact had severity major
	Critical	% bugs acted on by organisation in aggregate role of qa_contact had severity critical
	Blocker	% bugs acted on by organisation in aggregate role of qa_contact had severity blocker

Table 41: Variables capturing solution knowledge value severity measure at organisation level

As with the previous measure, this measure captures the organisation level counterpart of the severity and priority change measures at the problem and individual level, reflecting the notion in the literature that tendencies to act on problems whose value is debated in the meta-organisation, as reflected by the changes in the severity and priority variables, is hypothesized to positively correlate with solution knowledge mergence. This measure also complements its counterparts at other levels to promote localisation of the correct level of any effects on the dependent outcome of interest. Six variables, 2 for each of the 3 aggregate roles in which organisations engage, were created, capturing the percentage of bugs acted upon in each aggregate role whose severity/priority changed at least once, with the percentage of bugs whose severity/priority never changed acting as the reference category for analysis. Table 42 summarises the variables that constitute the solution knowledge value severity and priority change tendency measure at the organisation level.

Aggregate role	Value changed	Variable
Problem knowledge producer (reporter)	Severity	% bugs acted on by organisation in aggregate role of reporter with severity changed at least once
	Priority	% bugs acted on by organisation in aggregate role of reporter with priority changed at least once
Solution knowledge producer (assigned_to)	Severity	% bugs acted on by organisation in aggregate role of assigned_to with severity changed at least once
	Priority	% bugs acted on by organisation in aggregate role of assigned_to with priority changed at least once
Solution knowledge verifier (qa_contact)	Severity	% bugs acted on by organisation in aggregate role of qa_contact with severity changed at least once
	Priority	% bugs acted on by organisation in aggregate role of qa_contact with priority changed at least once

Table 42: Variables capturing solution knowledge value severity and priority change tendency measure at organisation level

The third measure of solution knowledge value was the tendencies to act upon bugs, in each of the 3 aggregate roles in which organisations engage, which had one or more top keywords. As with the previous measure, this measure captures the organisation level counterpart to the presence of popular keywords attached to bug. This measure reflects the notion that the tendency to act upon bugs with top keywords at the organisation level reflects higher solution knowledge value that is hypothesized to promote solution knowledge mergence. Twelve percentage variables were created, 3 for each aggregate role, which were selected to match their problem and individual level counterparts by examining the percentage of bugs acted upon in each aggregate role that have one or more top 3, top 10, top 25, and/or top 50 keywords. For each aggregate role, the reference category for analysis was all other bugs acted upon, such as those with no keywords or keywords not in the top 50 or higher. Table 43 summarises the variables that constitute the solution knowledge value keyword popularity tendency measure at the organisation level.

Aggregate role	Keyword popularity	Variable
Problem knowledge producer (reporter)	Top 3	% bugs acted on by organisation in aggregate role of reporter with top 3 keyword
	Top 10	% bugs acted on by organisation in aggregate role of reporter with top 10 keyword
	Top 25	% bugs acted on by organisation in aggregate role of reporter with top 25 keyword
	Top 50	% bugs acted on by organisation in aggregate role of reporter with top 50 keyword
Solution knowledge producer (assigned_to)	Top 3	% bugs acted on by organisation in aggregate role of assigned_to with top 3 keyword
	Top 10	% bugs acted on by organisation in aggregate role of assigned_to with top 10 keyword
	Top 25	% bugs acted on by organisation in aggregate role of assigned_to with top 25 keyword
	Top 50	% bugs acted on by organisation in aggregate role of assigned_to with top 50 keyword
Solution knowledge verifier (qa_contact)	Top 3	% bugs acted on by organisation in aggregate role of qa_contact with top 3 keyword
	Top 10	% bugs acted on by organisation in aggregate role of qa_contact with top 10 keyword
	Top 25	% bugs acted on by organisation in aggregate role of qa_contact with top 25 keyword
	Top 50	% bugs acted on by organisation in aggregate role of qa_contact with top 50 keyword

Table 43: Variables capturing solution knowledge value keyword popularity tendency measure at organisation level

The fourth and fifth measures of solution knowledge value were the count of follows, votes, and comments, and the mean number of flags attached to bugs acted upon in each of the 3 aggregate roles in which organisations engage. These measures complement the problem and individual level measures by examining the counts and averages related to in each aggregate role at the organisation level. They also complement the knowledge stakeholder influence measure by providing variables that examine the overall count of follows, votes and comments rather than those variables previously describe that only consider such variables according to the stakeholder’s power and influence in the meta-organisation. Further, the comment variable is an alternate representation of commenting effects that is distinct from the manual comments that were considered as part of the codifiability hypothesis, capturing instead a broader solution knowledge value representation of commenting tendencies. Together, these measures allow better localisation of any effect on solution knowledge mergence, improving the validity and distinctiveness of the measures.

Twelve variables make up these two measures, 4 for each aggregate role in which organisations engage. They reflect the notion in the literature that, at the organisation level, the count of following, votes, commenting, and average flags on problems amongst the aggregate roles in which organisations engage are hypothesized to positively correlate with solution knowledge emergence as such variables are signals of solution knowledge value. At the organisation level, the choice was made to focus on overall organisation-aggregate-role counts rather than overall organisation means or means of individual means, as this value is the most conducive to analysis that separates individual tendencies from organisation tendencies, as desired in this separate level of analysis. The exception was the case of flags, where the count of flags on a given problem may be sufficiently large as to obscure level distinctiveness effects when aggregated at the organisation level. As such, for flags, an average was taken for each organisation-aggregate-role. Sufficient linearity for analytical assumptions was readily induced by taking the log transformation of the variables in a manner similar to previous variables. Table 44 summarises the variables that make up the solution knowledge value measures reflected in following, voting, commenting, and flag averages at the organisation level.

Aggregate role	Tendency	Variable
Problem knowledge producer (reporter)	Following	(log) Count of votes on bugs acted on by organisation in aggregate role of reporter
	Votes	(log) Count of votes on bugs acted on by organisation in aggregate role of reporter
	Comments	(log) Count of comments on bugs acted on by organisation in aggregate role of reporter
	Flags	(log) Mean number of flags on bugs acted on by organisation in aggregate role of reporter
Solution knowledge producer (assigned_to)	Following	(log) Count of votes on bugs acted on by organisation in aggregate role of assigned_to
	Votes	(log) Count of votes on bugs acted on by organisation in aggregate role of assigned_to
	Comments	(log) Count of comments on bugs acted on by organisation in aggregate role of assigned_to
	Flags	(log) Mean number of flags on bugs acted on by organisation in aggregate role of assigned_to
Solution knowledge verifier (qa_contact)	Following	(log) Count of votes on bugs acted on by organisation in aggregate role of qa_contact
	Votes	(log) Count of votes on bugs acted on by organisation in aggregate role of qa_contact
	Comments	(log) Count of comments on bugs acted on by organisation in aggregate role of qa_contact
	Flags	(log) Mean number of flags on bugs acted on by organisation in aggregate role of qa_contact

Table 44: Variables capturing solution knowledge value as captured by following, voting, commenting, and flag averages measures at organisation level

Collectively, these measures triangulate the concept of solution knowledge value in terms of organisation level effects. Figure 32 summarises the operationalizations of the measures of solution knowledge value at the organisation level as well as their hypothesised direction of influence on the dependent outcome of interest, solution knowledge emergence.

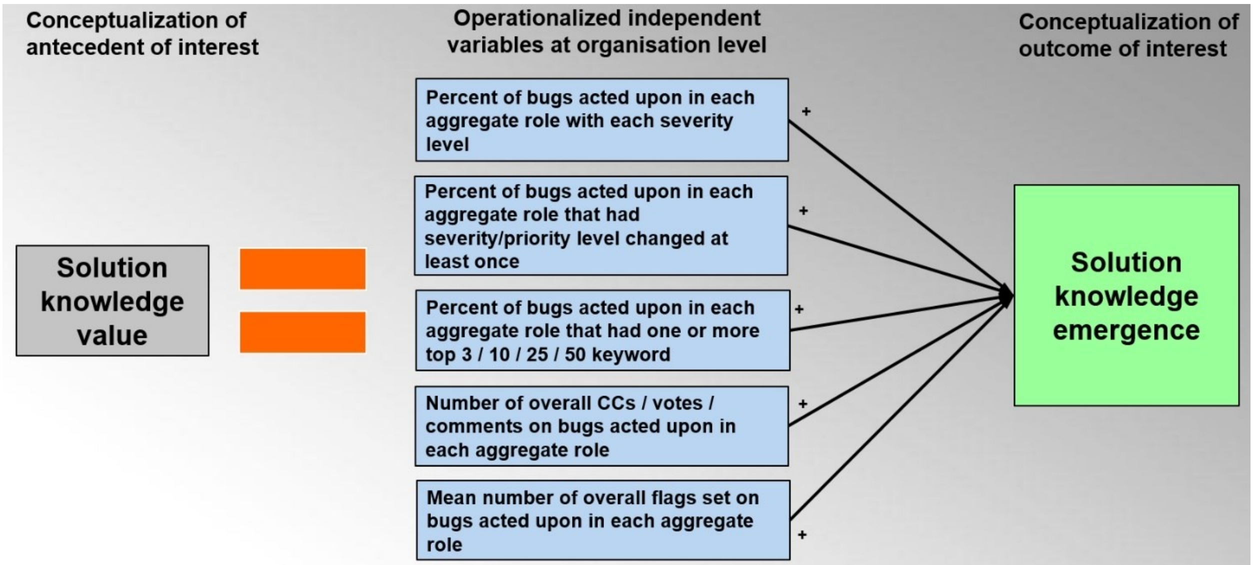

Figure 32: Operationalizations of measures of solution knowledge value at organisation level

In summary, this chapter described the operationalizations of the six antecedent factors of the theoretical framework, namely absorptive capacity, codifiability, dominant knowledge paradigm, knowledge flow impediments, knowledge stakeholder influence, and solution knowledge value, which are hypothesized to affect solution knowledge emergence at the problem level (bug), individual level (profile), and organisation level. The R code (R Foundation, 2017) that implements the operationalizations discussed in this chapter is reproduced in Appendix A: Operationalization code. The next chapter describes the process used to analyze the data.

CHAPTER FIVE: ANALYSIS

Given the complexity of the dataset and the nature of the operationalization of the variables discussed in the previous chapter, a systematic approach was used to determine the appropriate analytical tools to test this study's hypotheses. The analysis was separated into three parts based on level of analysis.

Problem level of analysis

As discussed in the previous chapter, problem level dependent variables were operationalized to conform to the data constraints and to maximize validity and reliability of the analytical process. The operationalization created three continuous variables, namely days to resolution, days to first assignment, and days from first assignment to resolution. Twenty-six logical variables were also created, namely logical outcome (fixed/not fixed), fixed with patch, was reopened, was reassigned, ever confirmed, as well as the twenty-one threshold-based timing variables, with seven thresholds for resolution time, seven thresholds for time to first assignment, and seven thresholds for time from first assignment to resolution.

The first step of the analysis, following conventional statistical analysis practices (c.f. Tabachnick & Fidell, 2007) was to examine the summary statistics of each dependent variable, which are each described in turn in the subsequent sections. Quantiles were calculated at 10% intervals following the formula of Hyndman & Fan (1996). Skewness and kurtosis were calculated following the formula of Joanes & Gill (1998). Both formulae are commonly used in major statistical analysis packages including SPSS and SAS.

Dependent variables: Reopening and reassigning tendencies

Examination of the dependent variables for reopening and reassigning tendencies produced the summary statistics described in Table 45 and the quantiles described in Table 46. Normal Q-Q plots (Tabachnick & Fidell, 2007), which show significant non-normal properties to the variables, are depicted in Appendix C: Additional analysis details.

Variable	N	Mean	Median	Stdev	Min	Max	Skewness	Kurtosis
reopened_count	774,765	0.091	0	0.349	0	24	6.488447	116.0092
reassigned_count	774,765	0.047	0	0.251	0	9	6.802974	65.13482

Table 45: Summary statistics of reopening and reassigning tendencies at problem level

Variable	Quantiles										
	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
reopened_count	0	0	0	0	0	0	0	0	0	0	24
reassigned_count	0	0	0	0	0	0	0	0	0	0	9

Table 46: Quantiles of reopening and reassigning tendencies at problem level

Upon inspection of the summary quantiles, skewness, kurtosis, and normal Q-Q plots it became clear that the reopening and reassigning count variables, while collected as continuous variables, are better understood as bimodal variables given that reopening and reassigning activities occur infrequently, as summarized in Table 47. As a result, binomial versions of each variable were created, and logistic regression (GLM with logit link) (Tabachnick & Fidell, 2007) and analysis of co-variance (ANCOVA type II) (Fox & Weisberg, 2011) were selected as the appropriate analytical approaches for analyze these dependent variables.

Variable	Count = 0	Count >= 1	Count >=2
reopening	714,086 (92.23%)	60,679 (7.83%)	7,718 (1.00%)
reassigning	743,360 (95.95%)	31,405 (4.05%)	4,270 (0.55%)

Table 47: Frequency of reopening and reassigning occurrences at problem level

Dependent variables: Confirmation, fixing, and patching tendencies

Given that the variables representing confirmation, fixing, and patching tendencies were already logical in nature, the major analytical concern was sufficient variability in responses for meaningful analysis. The variability described in Table 48 suggests sufficient variability is present in the data for analysis with logistic regression (GLM with logit link) and ANCOVA (type II) in a manner similar to the reopening and reassigning tendency variables.

Variable	N	True	False
everconfirmed	774,765	520,444	254,321
fixed	664,993	280,477	384,516
fixed_with_patch	664,993	148,843	516,150

Table 48: Variability of confirmation, fixing, and patching tendencies at problem level

Dependent variables: Resolution, assignment, and development timing

Examination of the dependent variables for resolution, assignment, and development timing produced the summary statistics described in Table 49 and quantiles described in Table 50.

Variable	N	Mean	Median	Stdev	Min	Max	Skewness	Kurtosis
resolution (days)	664,993	220.80	23.05	487.85	0.00003	4,861.79	3.826	17.569
assignment (days)	125,549	75.26	3.8	254.38	0.0001	4196.73	6.753	59.232
development (days)	116,621	213.40	22.3	555.362	0.0001	4783.17	4.201	19.191

Table 49: Summary statistics resolution, assignment, and development timing at problem level

Variable	Quantiles										
	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
resolution (days)	8.00	12.05	17.12	24.20	33.92	46.82	63.94	87.10	118.70	160.67	216.00
assignment (days)	<0.001	0.004	0.053	0.366	1.192	3.792	8.740	20.770	51.196	160.760	4196.73
development (days)	<0.001	0.70	2.19	5.71	11.56	22.34	44.10	87.56	188.86	524.94	4783.17

Table 50: Quantiles of resolution, assignment, and development timing at problem level

Normal Q-Q plots which show significant non-linear properties are depicted in Appendix C: Additional analysis details.

Given the S-shape curve features in the normal Q-Q plots of all three timing variables, standard transformations including square root, cube root, inverse, and log transformations were attempted to try to induce linearity, as depicted in Appendix C: Additional analysis details.

Examination of the boxplots of each transformation reveals that in all three cases the log transformation induces symmetry better than any other transformation. The log transformation is not ideal but is sufficient for ordinary least-squares (OLS) regression and ANCOVA analysis using standard assumptions of normality. In order to correct for the unbalanced nature of the data and the resulting non-linearity, it was decided to run heteroskedasticity correction on all regression models subsequent to fitting to reduce the likelihood of spurious results due to violations of assumptions of nonlinearity (Fox & Weisberg, 2011).

Given that the log transformation doesn't account for the S-shape observed in the normal Q-Q plots, additional manipulations were attempted (see Appendix C: Additional analysis details) to try to capture the full extent of the non-linearity in the data. Inspections of normal Q-Q plots for numerous isolated ranges of the timing data variables revealed inflection points around certain fast and slow timing tendencies for all three variables. These inflections were captured by creating logical threshold variables at each change in tendency reflecting seven categories of timing for each variable: extremely fast, very fast, fast, average, slow, very slow, and extremely slow. The specific timing, in days, for each inflection point that were calculated with this process were described in Table 4 in the previous chapter.

Each problem level dependent variable and the regression types selected as the result of these preliminary analyses are described in Table 51.

Dependent variable	Variable type	**Regression type** (Heteroskedasticity correction in all)
Fixed (logical_outcome)	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
Fixed and patched (fixed_with_patch)	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
Reopening tendencies (was_reopened)	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
Reassigning tendencies (was_reassigned)	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
Confirmation tendencies (everconfirmed)	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
(log) Resolution time (days_to_resolution)	Continuous	Ordinary least squares (OLS with dummy variables) + ANCOVA (Type II)
(log) Assignment time (days_to_first_assignment)	Continuous	Ordinary least squares (OLS with dummy variables) + ANCOVA (Type II)
(log) Development time (days_from_first_assignment_to_resolution)	Continuous	Ordinary least squares (OLS with dummy variables) + ANCOVA (Type II)
extremely_fast_resolution	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
very_fast_resolution	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
fast_resolution	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
average_resolution	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
slow_resolution	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
very_slow_resolution	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
extremely_slow_resolution	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
extremely_fast_assignment	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
very_fast_assignment	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
fast_assignment	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
average_assignment	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
slow_assignment	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)

very_slow_assignment	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
extremely_slow_assignment	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
extremely_fast_development	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
very_fast_development	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
fast_development	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
average_development	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
slow_development	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
very_slow_development	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
extremely_slow_development	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)

Table 51: Dependent variables and chosen regression types at problem level

Modelling: Problem level control variables

Modelling was conducted in two stages. The first stage consisted of the identification of suitable control variables for each model. Thirteen variables that are commonly reported in the literature as having direct influence on the problem level dependent variables described in the previous section were selected as control variable candidates (Huntley, 2003; Fershtman & Gandal, 2004; Sandusky, Gasser, & Ripoche, 2004; Anvik, Hiew, & Murphy, 2006; Koponen, 2006; Hooimeijer & Weimer, 2007; Panjer, 2007; Antoniol, et al., 2008; Dalle, et al., 2008; Francalanci & Merlo, 2008; Herraiz, 2008; Ahmed & Gokhale, 2009; Au et al., 2009; Bougie et al., 2010; Giger, Pinzger, & Gall, 2010; Shihab, et al, 2010; Zimmermann, et al., 2010; Guo, et al., 2011; Zhang, et al., 2012; Baysal, et al., 2013). For each of the 29 problem level dependent variables, a test regression model was fitted with the thirteen control variable candidates as independent variables. Examination of the analysis of variance and type II ANCOVA models with heteroskedasticity correction revealed which of the candidate control variables were significant for each dependent variable at a $p < 0.05$ degree of certainty. In each case, those variables that were found to be significant were retained as controls and those that were found to not be significant were dropped from the model, as per standard statistical modeling practice (Tabachnick & Fidell, 2007). As discussed in the previous chapter, in cases where a control variable was the same as or directly correlated to a given dependent variable, it was omitted from the regression model. The control variables and their data type are summarized in Table 52.

Control variable	Variable type
logical_outcome	Logical
everconfirmed	Logical
is_duplicate	Logical
bug_severity	Categorical
has_vote	Logical
violated_bug_lifecycle	Logical
rep_platform	Categorical
classification_name	Categorical
op_sys_id	Categorical (Converted to continuous)
product_id	Categorical (Converted to continuous)
component_id	Categorical (Converted to continuous)
days_to_resolution	Continuous
creation_year	Categorical

Table 52: Control variables at problem level

Modelling: Problem level independent variables

During second stage of modelling, the significant controls for each dependent variable were combined with the independent variables described in the problem level of the conceptual framework to create regression models that test each hypothesis. Each of the 29 models was created in the standard regression model form:

$$DV \sim Control_1 + Control_2 + \ldots + Control_n + IV_1 + IV_2 + \ldots + IV_n$$

The name and type of each independent variable associated to each hypothesis at the problem level of analysis are summarized as follows: Absorptive capacity in Table 53; codifiability in Table 54; dominant knowledge paradigm in Table 55; knowledge flow impediments in Table 56; knowledge stakeholder influence in Table 57; and, solution knowledge value in Table 58.

Independent variable	Variable type
open_bugs_at_creation_all_count	Count
open_bugs_at_creation_same_rep_platform_count	Count
open_bugs_at_creation_same_op_sys_count	Count
open_bugs_at_creation_same_classification_id_count	Count
open_bugs_at_creation_same_product_id_count	Count
open_bugs_at_creation_same_component_id_count	Count
bugs_created_past_1_day_count	Count
bugs_created_past_3_days_count	Count
bugs_created_past_7_days_count	Count
bugs_created_past_30_days_count	Count
bugs_created_past_90_days_count	Count
bugs_created_past_180_days_count	Count
bugs_created_past_1_year_count	Count
bugs_created_past_2_years_count	Count
bugs_censored_past_1_day_count	Count
bugs_censored_past_3_days_count	Count
bugs_censored_past_7_days_count	Count
bugs_censored_past_30_days_count	Count
bugs_censored_past_90_days_count	Count
bugs_censored_past_180_days_count	Count
bugs_censored_past_1_year_count	Count
bugs_censored_past_2_years_count	Count
days_to_resolution	Continuous
creation_year	Categorical
creation_month	Categorical
creation_weekday	Categorical
creation_monthday	Categorical

Table 53: Absorptive capacity independent variables at problem level

Independent variable	Variable type
title_length	Continuous
has_long_description	Logical
has_attachment	Logical
has_image_attachment	Logical
title_description_merged_ngram_distance_fixed_KLJ	Continuous
title_description_merged_ngram_outcome_prediction_ranks	Logical
description_readability_Flesch_reading_ease	Continuous
is_duplicate	Logical
has_duplicate	Logical
has_more_than_fifty_comments	Logical
comments_mean_length	Continuous

Table 54: Codifiability independent variables at problem level

Independent variable	Variable type
rep_platform	Categorical
classification_name	Categorical
op_sys_id	Categorical (Converted to continuous)
product_id	Categorical (Converted to continuous)
component_id,	Categorical (Converted to continuous)

Table 55: Dominant knowledge paradigm independent variables at problem level

Independent variable	Variable type
was_reopened	Logical
was_reassigned	Logical
had_keyword_change	Logical
had_flag_change	Logical
had_whiteboard_change	Logical
had_target_milestone_change	Logical
is_blocking_bug	Logical
is_blocked_by_bug	Logical
is_pre_fast_release	Logical
violated_bug_lifecycle	Logical
has_more_than_twenty_activities_total	Logical
had_activity_0_to_3_hours_after_creation	Logical
had_activity_3_to_6_hours_after_creation	Logical
had_activity_6_to_12_hours_after_creation	Logical
had_activity_12_to_24_hours_after_creation	Logical
had_activity_1_to_3_days_after_creation	Logical
had_activity_3_to_7_days_after_creation	Logical
had_activity_7_to_15_days_after_creation	Logical
had_activity_15_to_45_days_after_creation	Logical
had_activity_45_to_90_days_after_creation	Logical
had_activity_90_to_180_days_after_creation	Logical
had_activity_180_to_365_days_after_creation	Logical
had_activity_1_to_2_years_after_creation	Logical
had_activity_2_plus_years_after_creation	Logical
had_more_than_twenty_activities_later_than_2_years_after_creation	Logical

Table 56: Knowledge flow impediments independent variables at problem level

Independent variable	Variable type
reporter_id	Categorical (Converted to continuous)
reporter_domain_id	Categorical (Converted to continuous)
is_org_reporter_domain	Logical
is_reporter_core_actor	Logical
cc_core_actors_count	Count
has_core_actor_vote	Logical
has_core_actor_comment	Logical
has_peripheral_actor_cc	Logical
has_peripheral_actor_vote	Logical
has_peripheral_actor_comment	Logical

Table 57: Knowledge stakeholder influence independent variables at problem level

Independent variable	Variable type
bug_severity	Categorical
severity_change_count	Count
priority	Categorical
priority_change_count	Count
has_top_3_keyword	Logical
has_top_10_keyword	Logical
has_top_25_keyword	Logical
has_top_50_keyword	Logical
cc_all_actors_count	Count
votes_all_actors_count	Count

Table 58: Solution knowledge value independent variables at problem level

At the outset, a pair of models, consisting of one control-only model and one control plus independent variable model, was created for each of the 29 dependent variables for each of the six hypotheses, resulting in 348 models at the problem level. For each model, the goodness of fit and significance of calculated coefficients for each variable were evaluated using a range of standard statistical measures discussed in the following sections.

Evaluating models: OLS regression

In order to evaluate the 3 pairs of OLS regression model for each of the 6 hypotheses (18 models in total), a range of standard statistical analysis procedures were conducted. First, heteroskedasticity-corrected coefficients and standard errors were calculated for all control and independent variables. P values, indicating degree of confidence in rejecting the null hypothesis that there are no effects, were calculated for each variable and annotated using the standard stars annotation format with three stars (***) indicating $p < 0.001$ (double-tailed), two stars (**) indicating $p < 0.01$ (double-tailed), and one star (*) indicating $p < 0.05$ (double-tailed). P values greater than 0.05 were interpreted as the model fits having an insufficient degree of certainty to reject the null hypothesis. Given the magnitude of the data set, higher than typical p value cut-off thresholds were targeted as the goal of the present study is to identify powerful effects. Subsequent studies may wish to examine less prominent effects.

The model F statistic, chi-squared statistic, and corresponding p values and degrees of freedom were calculated for each model using the standard Wald test approach (Chambers, 1992; Hothorn, Zeileis, Farebrother, Cummins, Millo, & Mitchell, 2017) with heteroscedasticity correction of the variance-covariance matrix. Comparative Wald test F statistics and associated p values were calculated between the control and full model pairs to provide a measurement of the effect of the independent variables relative to the control variables alone. In order to estimate effect size, three measures were selected as the best choices from the statistical methods literature and calculated in turn. The R^2 statistic representing the fraction of variance explained by the model was calculated using the standard formula (Chambers & Hastie, 1992):

$$R^2 = 1 - Sum(R[i]^2) / Sum((y[i] - y^*)^2)$$

where y is the mean of y[i] if there is an intercept and zero otherwise*

Cohen's f^2 effect size and f^2 additive effect size were also calculated using the formulae widely accepted as the most appropriate for regression models of the nature used in this study, respectively (Cohen, 1988, 1992; Sawilowsky, 2009):

$$f^2 = R^2 / 1 - R^2$$

and

$$f^2 = R^2_{AB} - R^2_{A} / 1 - R^2_{AB}$$

where A denotes the control-only model and AB denotes the full model

For ease of interpretation of Cohen's f^2 effect size statistic, effect size descriptors were included in brackets below each value. When interpreting the results, these descriptors were considered in combination with other measures and used primarily in a between-model comparative manner rather than taken purely at face value in isolation, to ensure that they do not mislead given the whole context of each regression model, as cautioned by Sawilowsky (2009).

Lastly, for each variable in the models, an analysis of deviance test (type II) was conducted to evaluate the contribution of each variable to the overall model goodness of fit, represented by separate F statistic values and associated p values for degree of certainty, along with degrees of freedom and residuals. This complementary test allows a comparison of model with dummy variables representing each category of categorical values evaluated separately and collectively as a single variable.

Evaluating models: Logistic regression

For the logistic regression models, alternate measures that are suitable for generalized linear models (GLM) were calculated. Residual and null deviance for each model were calculated using standard approaches for GLM fits (Hastie & Pregibon, 1992). Akaike's An Information Criterion (AIC) and control vs. full model comparative delta-AIC (Sakamoto, Ishiguro, & Kitagawa, 1986) were calculated using the formula:

$$AIC = -2*\text{log-likelihood} + 2*npar$$

where npar is the number of parameters in the fitted model

Schwarz's Bayesian Information Criterion (BIC) (Schwarz, 1978) was calculated using the formula:

$$BIC = -2*log\text{-}likelihood + log(n)*npar$$

where npar is the number of parameters in the fitted model and n is the number of observations

AIC and BIC were used during analysis to facilitate model selection based on weights (Burnham & Anderson, 2002; Wagenmakers & Farrell, 2004).

Given that there is no single agreed-upon R^2 formula for generalized linear models (Jackman, 2015), a single "pseudo-R^2" value was calculated for each model by taking the statistical mean of the three values calculated using the three most common pseudo- R^2 formulae in the statistical literature: Cragg & Uhler's pseudo-R^2 (Cragg & Uhler, 1970), McFadden's pseudo-R^2 (McFadden, 1973), and Long's maximum likelihood pseudo-R^2 (Long, 1997). The resultant single value pseudo-R^2 was also used to calculate Cohen's f^2 effect size and f^2 additive effect size in a manner similar to the approach used for the linear models, and with similar contextual interpretation caution.

Lastly, in a manner similar to the approach used for the linear models, an analysis of deviance test (type II) for each variable in the models was conducted to evaluate the contribution of each variable to the overall model goodness of fit, represented by separate Chi-squared statistic values and associated p values for degree of certainty, along with degrees of freedom. This complementary test allows a comparison of model with dummy variables representing each category of categorical values evaluated separately and collectively as a single variable. It further offers a complementary p value interpretation of the likelihood ratio of the Chi-squared test of the coefficients in each model in order to address the concerns regarding inappropriate p value estimations for generalized linear models (Hastie & Pregibon, 1992; Fox, 2008;

Wasserstein & Lazar, 2016) when carefully interpreted in context and in combination with other values.

Individual level of analysis

As discussed in the previous chapter, individual level dependent variables were operationalized to conform to the data constraints and to maximize validity and reliability of the analytical process. The operationalization created 2 logical variables, 4 percentage variables, and 1 continuous variable for each of the roles in which individuals engage, resulting in 21 dependent variables in total. The role separation of dependent variables is necessary to reflect the nature of the distinctiveness between problem and individual level, which manifests in how individuals participate in the knowledge creation process orthogonal to the properties of the knowledge itself.

The three roles in which individuals engage are problem knowledge producer (reporter), solution knowledge producer (assigned_to), and solution knowledge verifier (QA_contact). These roles are inherent to the structure of the data used in this study. Every set of problem knowledge must have a “reporter” actor and associated organisation. All emergent solution knowledge must have an “assigned_to” actor and associated organisation. Only a select number of problems go through a formal verification and therefore have an associated “QA_contact” and organisation because only a small number of solutions are of sufficient complexity that they require a formal process to verify their match to the problem knowledge. In many cases, emergent solution knowledge is accepted at face value without the formal process. The association of the three roles with the problems in the data are is described in Table 59.

Role	Problems acted upon by role
Problem knowledge producer (reporter)	774,744 (100%)
Solution knowledge producer (assigned_to)	504,990 (65.2%)
Solution knowledge verifier (qa_contact)	265,588 (34.3%)

Table 59: Distribution of individual level roles relative to number of problems acted upon

As a result of the role distinction at the individual level, in line with the research question, it is necessary to aggregate from the problem level to the individual level according to role because individuals influence problem level outcomes acting in some combination of the reporter, assigned_to, and QA_contact roles. These are the primary participants in the knowledge production process, as operationalized. While those individual actors who never take on one of these three roles may influence the knowledge production process, such effects are captured as independent variable "community" effects because their action is not directly linked to the knowledge production process, as discussed in the operationalizations chapter. This approach provides a conservative estimate of the individual level effects on the outcome variables which fits this study's goal of focusing on large effects and leaving smaller effects to future research. It is also consistent with the methodological guidelines for operational aggregation of variables across levels (c.f. Rousseau & House, 1994; Chan, 1998; Bliese, 2000).

At the individual level, the unit of analysis is the profile, with a subset of all profiles engaging in the three roles that are theorized to influence the knowledge production process. The breakdown of number of profiles that engage in each of the roles is described in Table 60.

Profiles	Count
All	459,214 (100%)
Acted in reporter role	158,618 (34.5%)
Acted in assigned_to role	5,165 (1.1%)
Acted in qa_contact role	930 (0.2%)

Table 60: Distribution of roles in which individuals engage relative to all profiles in database

Given that only a subset of all profiles engages in active roles in the knowledge creation process, the first step in the analysis was to exclude those profiles not involved in one or more role because they do not represent meaningful degrees of freedom in the analysis and would spuriously inflate n values and resulting power effects. Next, the operationalized variables and associated data were organized into subsets according to each individual level role and analyzed each in turn.

Dependent variables: Reopening tendencies of problem knowledge producer role

Examination of the variables for the reopening tendencies of problems that each individual who engaged in the problem knowledge producer role acted upon produced the summary statistics described in Table 61 and the quantiles described in Table 62.

Variable	N	Mean	Median	Stdev	Min	Max
reopened_count	158618	0.45	0	5.31	0	589
reopened_mean	158618	0.07	0	0.26	0	10
reopened_at_least_once	158618	0.38	0	4.41	0	498
reopened_at_least_twice	158618	0.05	0	0.71	0	90
reopened_thrice_or_more	158618	0.01	0	0.188	0	19

Table 61: Summary statistics of reopening tendencies of problems acted upon by each individual engaged in problem knowledge producer role at individual level:

Variable	Quantiles										
	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
reopened_count	0	0	0	0	0	0	0	0	0	1	589
reopened_mean	0	0	0	0	0	0	0	0	0	0.133	10
reopened_at_least_once	0	0	0	0	0	0	0	0	0	1	498
reopened_at_least_twice	0	0	0	0	0	0	0	0	0	0	90
reopened_thrice_or_more	0	0	0	0	0	0	0	0	0	0	19

Table 62: Quantiles of reopening tendencies of problems acted upon by each individual engaged in problem knowledge producer role at individual level

Examination of the summary statistics and quantile distributions of the reopening tendencies reveal a very heavy skew, with the clear majority of those acting in the reporter role never reporting any bugs that are ever reopened. By comparing the mean number of bug-reopenings-per-profile to the mean number of bugs-reopened-at-least-once-per-profile, it becomes clear that the upper 10% heavily skew the overall results. In order to identify the source of the skew, the next step was to look at the percentage of bugs with some form of reopening relative to the number of bugs reported by each individual acting in the reporter role. This approach controls for the fact that some individuals who act in the reporter role report a large number of bugs whereas others only report one or two. The summary statistics of bug reporting tendencies are described in Table 63; the quantiles are described in Table 64; and, the Normal QQ-plot is depicted in Appendix C: Additional analysis details.

Variable	N	Mean	Median	Stdev	Min	Max	Skewness	Kurtosis
bugs_reported_count	158618	4.88	1	52.9	1	6883	51.61	4302

Table 63: Summary statistics of problem knowledge creation (bug reporting) at individual level

Variable	Quantiles										
	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
bugs_reported_count	0	0	0	0	0	0	0	0	0	1	589

Table 64: Quantiles of problem knowledge creation (bug reporting) at individual level

Inspection of the summaries and plots of bug reporting tendency reveals that some individuals are responsible for many bug reports while a large number of individuals only file a single piece of problem knowledge. As a result, for the analysis at the individual level, the influence of the number of bugs reported by each individual must be controlled for in the measure of the reopening variable to avoid spuriousness. A percentage-type variable was calculated that creates a uniform measure that normalizes the effects of bug reporting and focuses on the reopening effect associated with each individual's reported problems. The summary statistics of the percentage reopening tendencies are described in Table 65; the quantiles are described in Table 66; and, the Normal Q-Q plots are depicted in Appendix C: Additional analysis details.

Variable	N	Mean	Median	Stdev	Min	Max
percent_bugs_reopened_at_least_once_	158618	0.063	0	0.21	0%	100%
percent_bugs_reopened_at_least_twice	158618	0.006	0	0.07	0%	100%
percent_bugs_reopened_thrice_or_more	158618	0.001	0	0.03	0%	100%

Table 65: Summary statistics of percentages of reported problems that are reopened at individual level

Variable	Quantiles										
	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
percent_bugs_reopened_at_least_once_	0	0	0	0	0	0	0	0	0	0.13	1
percent_bugs_reopened_at_least_twice	0	0	0	0	0	0	0	0	0	0	1
percent_bugs_reopened_thrice_or_more	0	0	0	0	0	0	0	0	0	0	1

Table 66: Quantiles of percentages of reported problems that are reopened at individual level

Examination of the summaries and plots of the percentage variables reveals a different problem. While the skew due to some profiles reporting many bugs is addressed, there is insufficient variability in the data to get meaningful outcome values at the individual level because many profiles submit just one or a couple of problems and have undue weight on the percentage variables as a result. For example, a profile that has only reported one bug that happens to have been reopened will have a value of percent_bugs_reported_reopened_at_least_once of 100%, whereas a profile that has reported 50 bugs of which 10 have been reopened will only have a value of 20%. There are insufficient degrees of freedom in the former value for the 100% value to be meaningfully compared to the 20% value.

This effect demonstrates the need for a minimum threshold of actions in a given role for an individual level outcome variable to have sufficient degrees of freedom for non-spurious analysis and interpretation, as discussed in the operationalizations chapter. Based on inspections of the summary statistics and graph, a cut-off level of at least 4 actions in a given role was chosen in order for sufficient number of actions on problems to aggregate to have sufficient variability for profile-level analysis. Introducing this cut-off to the analysis results in the summary statistics described in Table 67, the quantiles described in Table 68, and the Normal Q-Q plots depicted Appendix C: Additional analysis details.

Variable	N	Mean	Median	Stdev	Min	Max
bugs_reported_count	17591	34.09	7	155.9	4	6883
percent_bugs_reopened_at_least_once_	17591	0.083	0.04	0.110	0	0.86
percent_bugs_reopened_at_least_twice	17591	0.010	0	0.038	0	0.75
percent_bugs_reopened_thrice_or_more	17591	0.002	0	0.018	0	0.75

Table 67: Summary statistics of reporting and reopening tendencies constrained to individuals who engaged in reporter role at least 4 times at individual level

Variable	Quantiles										
	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
bugs_reported_count	4	4	4	5	6	7	9	13	20	46	6883
percent_bugs_reopened_at_least_once_	0	0	0	0	0	0.04	0.08	0.12	0.17	0.25	0.86
percent_bugs_reopened_at_least_twice	0	0	0	0	0	0	0	0	0	0.02	0.75
percent_bugs_reopened_thrice_or_more	0	0	0	0	0	0	0	0	0	0	0.75

Table 68: Quantiles of reporting and reopening tendencies constrained to individuals who engaged in reporter role at least 4 times at individual level

The application of the constraint for bugs reported led to a trade-off of a lower n (n = 17, 591 vs. n = 15861) for better variability in the constrained data, exposing a clear exponential relationship, which is consistent with the theoretically expected distribution of these variables given the nature of the way individuals engage in the knowledge creation process. Given that the constrained n value is still very large, the trade-off was deepened worthwhile to improve the validity of the operationalizations of the outcome variables.

The constrained percentage variables show much better variability for analysis. Yet, significant overrepresentation of the 0% result for all reopening tendencies still exists. The large amount of 0% results was interpreted as reflecting the relatively rare nature of reopening events for problems, which is consistent with the theory of the knowledge flow through the bug life cycle in the literature (Koponen, 2006). As such, rather than attempting additional constrains or transformations to eliminate the 0% results, they were maintained because they have theoretical significance and are therefore valid representations of the operationalized variables in the data.

In order to address the split nature of the data the data was analyzed in two complementary ways. Given that the data shows that bug reopening is an unusual event and, as discussed in the previous section, there are theoretical reasons to believe that this is an accurate representation of this outcome variable, a logical variable was created that delineates profiles that reported at least one bug that was reopened from profiles that have never reported a bug that was reopened. This variable is the individual level counterpart to the problem level variable capturing reopening occurrence in the data. The distribution of this logical variable, subject to the same constraints discussed above, is described in Table 69. The distribution of the logical variable is very good for analytical purposes.

N	FALSE	TRUE
17,591	8436 (48.0%)	9155 (52.0%)

Table 69: Distribution of whether or not each profile reported at least one bug that was reopened at individual level

A second variable was created to capture the continuous variability in the percentage of bugs reported that were reopened for each profile if at least one reopening takes place. This variable complements the first logical variable by excluding the 8436 profiles that did not report at least one bug that was reopened at least once and focusing on the variability in reopening tendencies in the remaining profiles. This variable acts as the individual level counterpart to the problem level reopening tendency variable. The inclusion of both variables allows for better localisation of effects at the appropriate level. The summary statistics of the constrained percent variable are described in Table 70; the percentiles are described in Table 71; and, the normal Q-Q plot is depicted in Appendix C: Additional analysis details.

Variable	N	Mean	Median	Stdev	Min	Max
percent_bugs_reported_reopened_at_least_once	9155	0.16	0.14	0.10	0.008	0.857

Table 70: Summary statistics of constrained percentage of bugs reported that were reopened once or more at individual level

Quantiles of variable percent_bugs_reported_reopened_at_least_once										
0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
0.008	0.053	0.074	0.092	0.112	0.143	0.167	0.200	0.250	0.250	0.857

Table 71: Quantiles of constrained percentage of bugs reported that were reopened once or more at individual level

Examination of the summary statistics, quantiles, and normal Q-Q plot suggest that the distribution of the variable is now suitable for analysis with the selection of an appropriate regression model.

The logical reopening variable was analyzed using logistic regression (GLM with binomial logit link) and ANCOVA (type II), both with heteroskedasticity correction, in a manner similar to its counterpart at the problem level ove analysis. The percentage variable was analyzed using beta regression model fitting (Smithson & Verkuilen, 2006; Cribari-Neto & Zeileis, 2010; Simas, Barreto-Souza, & Rocha, 2010; Grün, Kosmidis, and Zeileis, 2012), which builds upon classic binomial probability statistical analysis principles (c.f. Williams, 1986; Smithson & Verkuilen, 2006) to address the problems that arise when analyzing constrained variables such as percentages with traditional regression models that assume Gaussian symmetry. This approach has been found to be superior to log or power based transformations because it maintains interpretability of the variables while correcting for heteroskedasticity, kurtosis, and different probability density functions inherent to constrained unit intervals. Further, beta regression promotes data-driven statistical analysis in a manner analogous to logistic regression rather than forcing non-ideal data to fit common regression models that are poorly suited to the data (Smithson & Verkuilen, 2006; Cribari-Neto & Zeileis, 2010; Simas, Barreto-Souza, & Rocha, 2010; Grün, Kosmidis, and Zeileis, 2012). In addition, analysis of covariance (ANCOVA type II) was conducted on the beta regression model fits (Fox & Weisberg, 2011).

Dependent variables: Reassigning tendencies of problem knowledge producer role

Given the analytical constraints for reopening tendencies discussed in the previous section and the fact that reassigning occurs with similar infrequence in the data, the same constraints were applied: Minimum number of 4 actions as problem knowledge producer for inclusion at individual level of analysis; and, reassigning tendencies split into two variables. The first variable was logical, capturing the occurrence of at least one reassignment of a problem acted upon by an individual in a problem knowledge producer (reporter) role. The second variable was a non-zero percent variable that captured the variability in reassigning tendencies amongst individuals for whom at least one problem that was acted upon in reporter role was reassigned. The distribution of the logical at_least_one_reassigned variable is described in Table 72. While the distribution is less even than the reopening occurrence variable, it is still more than sufficient for analysis using logistic regression (generalized linear modeling with logit link).

N	FALSE	TRUE
17,591	12760 (72.5%)	4831 (27.5%)

Table 72: Distribution of whether or not each profile reported at least one bug that was reassigned at individual level

The summary statistics of the constrained percent variable for reassigning tendencies are described in Table 73; the percentiles are described in Table 74; and, the normal Q-Q plot is depicted in Appendix C: Additional analysis details.

As with the reopening tendency percentage variable, the reassigning tendency percentage variable was analyzed using beta regression. In addition, analysis of covariance (ANCOVA type II) was conducted on the beta regression model fits (Fox & Weisberg, 2011).

Variable	N	Mean	Median	Stdev	Min	Max
percent_bugs_reported_reassigned_at_least_once	4831	0.126	0.10	0.10	0.001	0.818

Table 73: Summary statistics of constrained percentage of bugs reported that were reassigned once or more at individual level

Quantiles of variable percent_bugs_reported_reassigned_at_least_once										
0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
0.001	0.023	0.040	0.058	0.077	0.100	0.125	1.167	0.200	0.250	0.818

Table 74: Quantiles of constrained percentage of bugs reported that were reassigned once or more at individual level

Dependent variables: Outcome tendencies of problem knowledge producer role

The individual level counterparts to the problem level outcome tendency variables necessarily separate the variables based on the roles in which individuals engage when acting upon the problems that manifest the variable outcomes, similar to the previously discussed variables. Therefore, the same constraint of a minimum of four actions being required in the problem knowledge producer role is applied as the cut-off for inclusion in the analysis at the individual level. Whereas at the problem level, the outcome status is a simple logical fixed or not fixed outcome, at the individual level, the goal is to capture tendency effects that relate to the individual engaging in each role. Because, as discussed in the section on reopening tendencies, the number of actions in a given role will bias any count measures, the decision was made to use a percent variable to represent reopening tendencies. Two percent variables were created: percent of bugs acted upon in problem knowledge producer role that were fixed and percent of bugs acted upon in problem knowledge producer role that were fixed with a patch. Given that percent bugs not fixed is simply 1 – percent bugs fixed, it is redundant and not included as a separate variable. Bugs with pending status are not considered here for the same reasons discussed in the problem level outcome tendencies analysis section.

The summary statistics of the two percent outcome tendency variables are described in Table 75; the quantiles are described in Table 76; and, the normal Q-Q plots are depicted in Appendix C: Additional analysis details.

Variable	N	Mean	Median	Stdev	Min	Max
percent_bugs_reported_fixed	17560	0.243	0.1667	0.273	0	1
percent_bugs_reported_fixed_at_least_one_patch	17560	0.122	0.000	0.198	0	1

Table 75: Summary statistics of constrained percentage of bugs reported that were fixed / fixed with at least one patch at individual level

Variable	Quantiles										
	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
percent_bugs_reported_fixed	0.00	0.00	0.00	0.00	0.09	0.17	0.25	0.33	0.50	0.68	1.00
percent_bugs_reported_fixed_at_least_one_patch	0.00	0.00	0.00	0.00	0.00	0.00	0.06	0.14	0.25	0.40	1.00

Table 76: Quantiles of constrained percentage of bugs reported that were fixed / fixed with at least one patch at individual level

While the central portion of the normal Q-Q plot for both variables has linear properties, the large number of 0 and 1 values result in S-shapes. Additional splits and transformations were attempted on the variables (not shown—see Appendix C: Additional analysis details) and it was concluded that the best approach was to use beta regression model fitting (Smithson & Verkuilen, 2006; Cribari-Neto & Zeileis, 2010; Simas, Barreto-Souza, & Rocha, 2010; Grün, Kosmidis, and Zeileis, 2012) because it accounts for the floor and ceiling biases inherent in constrained range variables such as percentages. In order to ensure that the floor and ceiling values do not undyly bias the maximum-likelihood estimates in the regression models, following the advice of Smithson and Verkuilen (2006), a standard transformation was applied to each percent variable such that:

$$y^{1} = (y * (n - 1) + 0.5) / n$$

where n is the sample size

Given that the floor and ceiling values have theoretical meaning in the context of outcome tendencies, their inclusion contributes to maintaining validity during the analysis process. The skew and heteroskedasticities observed in the variable are accounted for in the beta regression modelling (Cribari-Neto & Zeileis, 2010). In addition, analysis of covariance (ANCOVA type II) was conducted on the beta regression model fits (Fox & Weisberg, 2011).

Dependent variables: Resolution timing tendencies of problem knowledge producer role

Thereas there are three timing tendency variables at the problem level: resolution time, assignment time, and development time, given that at the individual level the variables necessarily aggregate around the role in which individuals engage on problems, only the resolution time tendency variable was considered at the individual level of analysis. Both the assignment time and development time variables are problematic to consider at the individual level because they necessarily involve the interaction of multiple individuals in determining the timing tendencies, making it impossible to attribute the individual level effects to a particular individual. Further, aside from the confounding nature of the source of individual level contributions, given that there is a theoretical relationship between the resolution time, assignment time, and development time variables, when taking their mean at the invidual level, their variability is reduced to central tendency deviance such that there is too much co-linearity for analysis with assumptions of orthogonality. As a result, examination of assignment time and development time tendencies at the individual level is suggested for future research when additional data are available that allow for the disambiguation of the individual contributions to these timing effects and that enable the separation of the co-linearity of the variables at the individual level.

At the individual level, the resolution time tendencies for each individual are aggregated as a simple mean of the resolution times of all the bugs acted upon in the problem knowledge producer role. As with previous variables at the individual level, a constraint of 4 actions as problem knowledge producer was applied as a threshold for inclusion at the individual level of analysis. By taking the average, the total number of actions taken in the role do not unduly skew the resolution timing effects, allowing comparison across individuals with different degrees of involvement, as desired.

The summary statistics of the constrained variable for resolution timing tendencies are described in Table 77; the percentiles are described in Table 78, and, the normal Q-Q plot is depicted in Appendix C: Additional analysis details.

Variable	N	Mean	Median	Stdev	Min	Max	Skewness	Kurtosis
bugs_reported_mean_ days_to_resolution	17591	195.6	122.8	210.3	<0.01	1826.4	1.870	4.474

Table 77: Summary statistics of constrained resolution timing tendency variable for reporter role at individual level

Quantiles of variable bugs_reported_mean_days_to_resolution										
0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
<0.01	13.54	33.96	57.65	86.97	122.80	168.16	235.06	327.39	486.85	1826.36

Table 78: Quantiles of constrained resolution timing tendency variable for reporter role at individual level

Examination of the summary statistics and normal Q-Q plot reveals significant non-linearity in the distribution. Several standard transformations were applied to attempt to induce linearity for analysis (not shown—see Appendix C: Additional analysis details). Boxplots of the standard transformations are depicted in Appendix C: Additional analysis details.

Whereas both the log and square-root transformations result in better normality, the log transformation better captures the values that are very small, close to zero. Since these values are theoretically relevant, representing very quick average resolution times, the log transformation was chosen as a better mapping of the underlying data as the square-root transformation would diminish their contribution to the overal distribution of the data. Further, the log transformation is more readily interpretable, facilitating hypothesis testing, as is the intent of this analysis. Lastly, use of the log transformation facilitates direct cross-level comparison given the log transformation was also used for timing tendences at the problem level of analysis, allowing for localisation of the level of any effect, as per the theoretical framework of the study. Both log (base 10) and log + 1 transformations were attempted, with little difference noted between the two. The resulting log-transformed resolution timing variable was analyzed using ordinary least-squares (OLS) regression and ANCOVA using standards assumptions with heteroskedasticity correction for those non-linearities not addressed by the log transformation (Fox & Weisberg, 2011).

The reporter role dependent variables at the individual level and the regression types selected as the result of these preliminary analyses are described in Table 79.

Dependent variable	**Variable type**	**Regression type** (Heteroskedasticity correction in all)
At least one reported bug was reopened	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
(Non-zero) Percent of reported bugs were reopened at least once	Percent	Beta regression + ANCOVA (Type II)
At least one reported bug was reassigned	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
(Non-zero) Percent of reported bugs were reassigned at least once	Percent	Beta regression + ANCOVA (Type II)
Percent of reported bugs were fixed	Percent	Beta regression + ANCOVA (Type II)
Percent of reported bugs were fixed with at least one patch	Percent	Beta regression + ANCOVA (Type II)
(log) Mean resolution time of reported bugs	Continuous	Ordinary least squares (OLS) + ANCOVA (Type II)

Table 79: Problem knowledge producer role dependent variables and chosen regression types at individual level

Dependent variables: Solution knowledge producer and solution knowledge verifier roles

The dependent variables for the solution knowledge producer and solution knowledge verifier roles were analyzed in a manner similar to the problem knowledge producer role except constrained according to their respective roles. The same analytical refinement process was applied as the one described in the previous section (not shown—see Appendix C: Additional analysis details). The result of the process was the selection of the same analytical methods as for the problem knowledge producer role, which is desirable to simplify cross-role comparison at the individual level. The most notable difference relative to the problem knowledge producer role was the significantly lower n value for the solution knowledge producer role and the very low n value for the solution knowledge verifier role. These lower n values reflect the relative infrequency of individual level engagement in these roles as expected in the data. The lower n values present some challenges that are carefully considered in the interpretation of the results, as discussed in the next chapter.

The summary statistics of the logical and (non-zero) percent variables for reopening and reassigning tendencies of solution knowledge producer (assigned_to) and solution knowledge verifier (QA_contact) roles are described in Table 80 and Table 81; the quantiles of the percent variables are described in Table 82; and, the normal Q-Q plots of the percent variables are depicted in Appendix C: Additional analysis details.

Variable	N	FALSE	TRUE
assigned_to_reopened_at_least_once	2274	372	1902
assigned_to_reassigned_at_least_once	2274	737	1537
qa_contact_reopened_at_least_once	439	42	397
qa_contact_reassigned_at_least_once	439	121	318

Table 80: Distribution of whether or not each profile acted in role of assigned_to / qa_contact upon at least one bug that was reopened / reassigned at least once at individual level

Variable	N	Mean	Median	Stdev	Min	Max
percent_bugs_assigned_to_reopened_at_least_once	1902	0.126	0.101	0.092	0.008	1.000
percent_bugs_assigned_to_reassigned_at_least_once	1537	0.117	0.071	0.132	0.001	1.000
percent_bugs_qa_contact_reopened_at_least_once	397	0.126	0.105	0.092	0.015	0.650
percent_bugs_qa_contact_reassigned_at_least_once	318	0.111	0.075	0.109	0.002	0.750

Table 81: Summary statistics of constrained percentage of bugs acted upon as assigned_to / qa_contact that were reopened / reassigned once or more at individual level

Variable	Quantiles										
	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
percent_bugs_assigned_to_reopened_at_least_once	0.008	0.045	0.063	0.077	0.089	0.101	0.118	0.143	0.167	0.250	1.000
percent_bugs_assigned_to_reassigned_at_least_once	0.001	0.013	0.024	0.036	0.052	0.071	0.100	0.132	0.182	0.259	1.000
percent_bugs_qa_contact_reopened_at_least_once	0.015	0.051	0.067	0.081	0.092	0.105	0.118	0.133	0.158	0.215	0.650
percent_bugs_qa_contact_reassigned_at_least_once	0.002	0.010	0.031	0.045	0.059	0.075	0.100	0.136	0.175	0.250	0.750

Table 82: Quantiles of constrained percentage of bugs acted upon as assigned_to / qa_contact that were reopened / reassigned once or more at individual level

As was the case for the analysis of the reporter role dependent variable counterparts, for the assigned_to and QA_contact roles, the logical reopening and reassigning tendency variables were analyzed using logistic regression, and the percent reopening and reassigning tendency variables were analyzed using beta regression. Once again, beta regression was selected to analyze the percentage variables because it addresses the unit interval constraint while also accounting for the non-linearities observed in the normal Q-Q plots (Smithson & Verkuilen, 2006; Cribari-Neto & Zeileis, 2010; Simas, Barreto-Souza, & Rocha, 2010; Grün, Kosmidis, and Zeileis, 2012). In addition, analysis of covariance (ANCOVA type II) was conducted on the respective logistic and beta regression model fits (Fox & Weisberg, 2011).

The summary statistics of the two percent outcome tendency variables, percent fixed and percent fixed with patch, for assigned_to and QA_contact roles are described in Table 83; the quantiles are described in Table 84; and, the normal Q-Q plots are depicted in Appendix C: Additional analysis details. All these percent variables were analyzed using beta regression and analysis of covariance.

Variable	N	Mean	Median	Stdev	Min	Max
percent_bugs_assigned_to_fixed	2453	0.753	0.857	0.271	0.000	1.000
percent_bugs_assigned_to_fixed_at_least_one_patch	2453	0.467	0.483	0.386	0.000	1.000
percent_bugs_qa_contact_fixed	461	0.610	0.625	0.277	0.000	1.000
percent_bugs_qa_contact_fixed_at_least_one_patch	461	0.254	0.154	0.257	0.000	1.000

Table 83: Summary statistics of constrained percentage of bugs acted upon in assigned_to / qa_contact role that were fixed / fixed with at least one patch at individual level

	Quantiles										
Variable	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
percent_bugs_assigned_to_fixed	0.00	0.30	0.54	0.70	0.80	0.86	0.90	0.94	0.98	1.00	1.00
percent_bugs_assigned_to_fixed_at_least_one_patch	0.00	0.00	0.01	0.08	0.25	0.48	0.68	0.80	0.89	1.00	1.00
percent_bugs_qa_contact_fixed	0.00	0.21	0.33	0.45	0.54	0.63	0.73	0.82	0.88	0.98	1.00
percent_bugs_qa_contact_fixed_at_least_one_patch	0.00	0.00	0.01	0.07	0.11	0.15	0.25	0.36	0.47	0.67	1.00

Table 84: Quantiles of constrained percentage of bugs acted upon in assigned_to / qa_contact role that were fixed / fixed with at least one patch at individual level

The summary statistics of the constrained variable for resolution timing tendencies for assigned_to and QA_contact roles are described in Table 85; the quantiles are described in Table 86; and, the normal Q-Q plots and the boxplots of the standard transformations for both variables are depicted in Appendix C: Additional analysis details. The log transformation of the resolution timing variables was selected as most appropriate both in terms of analytical validity and comparability to the reporter role equivalent variable. Analysis was conducted using OLS regression and ANCOVA (type II), both with heteroskedasticity correction for remaining non-linearities not addressed by the log transformation (Fox & Weisberg, 2011).

Variable	N	Mean	Median	Stdev	Min	Max	Skewness	Kurtosis
bugs_assigned_to_mean_ days_to_resolution	2470	192.17	115.00	228.72	0.308	2182.9	2.481	9.461
bugs_qa_contact_mean_ days_to_resolution	463	167370	110.30	204.69	0.231	2111.8	3.478	21.278

Table 85: Summary statistics of constrained resolution timing tendency variables for assigned_to / qa_contact roles at individual level

	Quantiles										
Variable	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
bugs_assigned_to_mean_ days_to_resolution	0.3	10.9	26.3	47.6	77.8	115.0	162.4	219.1	305.4	490.6	2182.9
bugs_qa_contact_mean_ days_to_resolution	0.2	8.7	23.8	46.7	69.4	110.3	154.4	204.7	258.4	374.5	2111.8

Table 86: Quantiles of constrained resolution timing tendency variables for assigned_to / qa_contact roles at individual level

The assigned_to role and QA_contact role dependent variables at the individual level and the regression types selected as the result of these preliminary analyses are described in Table 87.

Dependent variable	Variable type	Regression type (Heteroskedasticity correction in all)
At least one bug acted upon in assigned_to role was reopened	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
(Non-zero) Percent of bugs acted upon in assigned_to role that were reopened at least once	Percent	Beta regression + ANCOVA (Type II)
At least one bug acted upon in assigned_to role was reassigned	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
(Non-zero) Percent of bugs acted upon in assigned_to role that were reassigned at least once	Percent	Beta regression + ANCOVA (Type II)
Percent of bugs acted upon in assigned_to role that were fixed	Percent	Beta regression + ANCOVA (Type II)
Percent of bugs acted upon in assigned_to role that were fixed with at least one patch	Percent	Beta regression + ANCOVA (Type II)
(log) Mean resolution time of bugs acted upon in assigned_to role	Continuous	Ordinary least squares (OLS) + ANCOVA (Type II)
At least one bug acted upon in qa_contact role was reopened	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
(Non-zero) Percent of bugs acted upon in qa_contact role that were reopened at least once	Percent	Beta regression + ANCOVA (Type II)
At least one bug acted upon in qa_contact role was reassigned	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
(Non-zero) Percent of bugs acted upon in qa_contact role that were reassigned at least once	Percent	Beta regression + ANCOVA (Type II)
Percent of bugs acted upon in qa_contact role that were fixed	Percent	Beta regression + ANCOVA (Type II)
Percent of bugs acted upon in qa_contact role that were fixed with at least one patch	Percent	Beta regression + ANCOVA (Type II)
(log) Mean resolution time of bugs acted upon in qa_contact role	Continuous	Ordinary least squares (OLS) + ANCOVA (Type II)

Table 87: Solution knowledge producer role and solution knowledge verifier role dependent variables and chosen regression types at individual level

Modelling: Individual level control variables

Modelling was conducted in two stages in a manner similar to the approach used at the problem level of analysis. In the first stage, suitable control variables were selected from the literature. (Huntley, 2003; Fershtman & Gandal, 2004; Sandusky, Gasser, & Ripoche, 2004; Anvik, Hiew, & Murphy, 2006; Koponen, 2006; Hooimeijer & Weimer, 2007; Panjer, 2007; Antoniol, et al., 2008; Dalle, et al., 2008; Francalanci & Merlo, 2008; Herraiz, 2008; Ahmed & Gokhale, 2009; Au et al., 2009; Bougie et al., 2010; Giger, Pinzger, & Gall, 2010; Shihab, et al, 2010; Zimmermann, et al., 2010; Guo, et al., 2011; Zhang, et al., 2012; Baysal, et al., 2013). At the individual level of analysis, similar to the dependent variables discussed in the previous section, many of the control variables are necessarily defined relative to the role in which each individual engages upon problems. In cases where the variables had significant non-linear distribution properties amongst individuals (typically "count" type variables), the log transformation was taken to normalize the variable for analysis.

For the problem knowledge producer role, seven control variables were identified that were expected to affect individual level outcome measures: 1) whether the individual is a "core actor"; 2) the (log) activity count of the individual; 3) the (log) number of times the individual engaged in the problem knowledge producer (reporter) role; 4) the (log) mean resolution time of problems acted upon in reporter role; 5) the percent of reported bugs that were duplicates; 6) the percent of reported bugs that were fixed; and, 7) the percent of reported bugs that violated the bug life cycle.

For the solution knowledge producer (assigned_to) and solution knowledge verifier (QA_contact) roles, six counterparts of each of these control variables were used, relative to their respective roles; e.g., for the regression models created to analyze the solution knowledge producer (assigned_to) role, the counterpart to control variable 6) was percent of bugs acted upon in assigned_to role that were fixed. Control variable 5), the percent of reported bugs that were duplicates, is only defined for the problem knowledge producer role, because duplicate bugs, by definition, never proceed along the bug life cycle to solution knowledge creation or solution knowledge verification. As a result, this control variable was only used for models that analyzed the reporter role at the individual level.

As discussed in the operationalization chapter and in the section on problem level analysis, in cases where a control variable was the same as or directly correlated to a given dependent variable, it was omitted from the regression model. For each of the 21 individual level dependent variables (seven for each of the three roles in which individuals engage), a test regression model was fitted with the seven (or six, for assigned_to and QA_contact roles) control variable candidates as independent variables. Examination of the analysis of variance and type II ANCOVA models with heteroskedasticity correction revealed which of the candidate control variables were significant for each dependent variable at a $p < 0.05$ degree of certainty. In each case, those variables that were found to be significant were retained as controls and those that were found to not be significant were dropped from the model, as per standard statistical modeling practice (Tabachnick & Fidell, 2007). The control variables, the roles to which they pertain in the analysis, and their variable type are summarized in Table 88.

Role	Control variable	Variable type
All	is_core_actor	Logical
All	(log) activity_count	Count
Problem knowledge producer (reporter)	(log) bugs_reported_count	Count
Problem knowledge producer (reporter)	(log) bugs_reported_ mean_days_to_resolution	Continuous
Problem knowledge producer (reporter)	percent_bugs_reported_ is_duplicate	Percent
Problem knowledge producer (reporter)	percent_bugs_reported_fixed	Percent
Problem knowledge producer (reporter)	percent_bugs_reported_ violated_bug_lifecycle	Percent
Solution knowledge producer (assigned_to)	(log) bugs_assigned_to_count	Count
Solution knowledge producer (assigned_to)	(log) bugs_assigned_to_ mean_days_to_resolution	Continuous
Solution knowledge producer (assigned_to)	percent_bugs_assigned_to_fixed	Percent
Solution knowledge producer (assigned_to)	percent_bugs_assigned_to_ violated_bug_lifecycle	Percent
Solution knowledge verifier (qa_contact)	(log) bugs_qa_contact_count	Count
Solution knowledge verifier (qa_contact)	(log) bugs_qa_contact_ mean_days_to_resolution	Continuous
Solution knowledge verifier (qa_contact)	percent_bugs_qa_contact_fixed	Percent
Solution knowledge verifier (qa_contact)	percent_bugs_qa_contact_ violated_bug_lifecycle	Percent

Table 88: Control variables at individual level

Modelling: Individual level independent variables

During the second stage of modelling, the significant controls for each dependent variable were combined with the independent variables described in the individual level of the conceptual framework to create regression models that test each hypothesis. Each of the 21 models (7 dependent variables for each of the 3 roles in which individuals engage) was created in the standard regression model form in a manner similar to that discussed in the problem level analysis section.

The name, type, and related role of each independent variable associated to each hypothesis at the individual level of analysis are summarized as follows: Absorptive capacity in Table 89; codifiability in Table 90; dominant knowledge paradigm in Table 91; knowledge flow impediments in Table 92; knowledge stakeholder influence in Table 93; and, solution knowledge value in Table 94.

Role	Independent variable	Variable type
reporter	(log) bugs_reported_count	Count
assigned_to	(log) bugs_assigned_to_count	Count
qa_contact	(log) bugs_qa_contact_count	Count
All	(log) activity_count	Count
All	(log) activity_cc_change_count	Count
All	(log) activity_keywords_change_count	Count
All	(log) activity_product_change_count	Count
All	(log) activity_component_change_count	Count
All	(log) activity_status_change_count	Count
All	(log) activity_resolution_change_count	Count
All	(log) activity_flags_change_count	Count
All	(log) activity_whiteboard_change_count	Count
All	(log) activity_target_milestone_change_count	Count
All	(log) activity_description_change_count	Count
All	(log) activity_priority_change_count	Count
All	(log) activity_severity_change_count	Count
All	(log) activity_assigning_count	Count
All	(log) activity_reassigning_count	Count
All	(log) activity_reopening_count	Count
All	(log) activity_rep_platform_All_count	Count
All	(log) activity_rep_platform_PowerPC_count	Count
All	(log) activity_rep_platform_x86_64_count	Count
All	(log) activity_rep_platform_x86_count	Count
All	(log) activity_rep_platform_Combined_Other_count	Count
All	(log) activity_op_sys_Mac_pc_count	Count
All	(log) activity_op_sys_Windows_pc_count	Count
All	(log) activity_op_sys_Windows_mobile_count	Count
All	(log) activity_op_sys_iOS_mobile_count	Count
All	(log) activity_op_sys_other_mobile_count	Count
All	(log) activity_op_sys_other_pc_count	Count
All	(log) activity_product_classification_client_software_count	Count
All	(log) activity_product_classification_components_count	Count
All	(log) activity_product_classification_server_software_count	Count
All	(log) activity_product_classification_Combined_Other_count	Count
All	(log) activity_bugs_low_severity_count	Count
All	(log) activity_bugs_average_severity_count	Count
All	(log) activity_bugs_high_severity_count	Count

Table 89: Absorptive capacity independent variables at individual level

Role	Independent variable	Variable type
reporter	(log) bugs_reported_all_types_description_mean_length	Continuous
reporter	bugs_reported_description_readability_Flesch_reading_ease_mean	Continuous
reporter	(log) bugs_reported_attachments_all_types_mean	Continuous
reporter	percent_bugs_reported_is_duplicate	Percent
reporter	percent_bugs_reported_was_duplicated	Percent
reporter	(log) bugs_reported_all_types_comments_mean_length	Continuous
reporter	(log) bugs_reported_comments_all_actors_mean	Continuous
assigned_to	(log) bugs_assigned_to_all_types_description_mean_length	Continuous
assigned_to	bugs_assigned_to_description_readability_Flesch_reading_ease_mean	Continuous
assigned_to	(log) bugs_assigned_to_attachments_all_types_mean	Continuous
assigned_to	(log) bugs_assigned_to_all_types_comments_mean_length	Continuous
assigned_to	(log) bugs_assigned_to_comments_all_actors_mean	Continuous
qa_contact	(log) bugs_qa_contact_all_types_description_mean_length	Continuous
qa_contact	bugs_qa_contact_description_readability_Flesch_reading_ease_mean	Continuous
qa_contact	(log) bugs_qa_contact_attachments_all_types_mean	Continuous
qa_contact	(log) bugs_qa_contact_all_types_comments_mean_length	Continuous
qa_contact	(log) bugs_qa_contact_comments_all_actors_mean	Continuous

Table 90: Codifiability independent variables at individual level

Role	Independent variable	Variable type
reporter	percent_bugs_reported_rep_platform_All	Percent
reporter	percent_bugs_reported_rep_platform_PowerPC	Percent
reporter	percent_bugs_reported_rep_platform_x86_64	Percent
reporter	percent_bugs_reported_rep_platform_x86	Percent
reporter	percent_bugs_reported_op_sys_All	Percent
reporter	percent_bugs_reported_op_sys_Android	Percent
reporter	percent_bugs_reported_op_sys_Linux	Percent
reporter	percent_bugs_reported_op_sys_Mac_pc	Percent
reporter	percent_bugs_reported_op_sys_Windows_pc	Percent
reporter	percent_bugs_reported_op_sys_Windows_mobile	Percent
reporter	percent_bugs_reported_op_sys_iOS_mobile	Percent
reporter	percent_bugs_reported_op_sys_other_mobile	Percent
reporter	percent_bugs_reported_product_classification_client_software	Percent
reporter	percent_bugs_reported_product_classification_components	Percent
reporter	percent_bugs_reported_product_classification_server_software	Percent
assigned_to	percent_bugs_assigned_to_rep_platform_All	Percent
assigned_to	percent_bugs_assigned_to_rep_platform_PowerPC	Percent
assigned_to	percent_bugs_assigned_to_rep_platform_x86_64	Percent
assigned_to	percent_bugs_assigned_to_rep_platform_x86	Percent
assigned_to	percent_bugs_assigned_to_op_sys_All	Percent
assigned_to	percent_bugs_assigned_to_op_sys_Android	Percent
assigned_to	percent_bugs_assigned_to_op_sys_Linux	Percent
assigned_to	percent_bugs_assigned_to_op_sys_Mac_pc	Percent
assigned_to	percent_bugs_assigned_to_op_sys_Windows_pc	Percent
assigned_to	percent_bugs_assigned_to_op_sys_Windows_mobile	Percent
assigned_to	percent_bugs_assigned_to_op_sys_iOS_mobile	Percent
assigned_to	percent_bugs_assigned_to_op_sys_other_mobile	Percent
assigned_to	percent_bugs_assigned_to_product_classification_client_software	Percent
assigned_to	percent_bugs_assigned_to_product_classification_components	Percent
assigned_to	percent_bugs_assigned_to_product_classification_server_software	Percent
qa_contact	percent_bugs_qa_contact_rep_platform_All	Percent
qa_contact	percent_bugs_qa_contact_rep_platform_PowerPC	Percent
qa_contact	percent_bugs_qa_contact_rep_platform_x86_64	Percent
qa_contact	percent_bugs_qa_contact_rep_platform_x86	Percent
qa_contact	percent_bugs_qa_contact_op_sys_All	Percent
qa_contact	percent_bugs_qa_contact_op_sys_Android	Percent
qa_contact	percent_bugs_qa_contact_op_sys_Linux	Percent
qa_contact	percent_bugs_qa_contact_op_sys_Mac_pc	Percent
qa_contact	percent_bugs_qa_contact_op_sys_Windows_pc	Percent
qa_contact	percent_bugs_qa_contact_op_sys_Windows_mobile	Percent
qa_contact	percent_bugs_qa_contact_op_sys_iOS_mobile	Percent
qa_contact	percent_bugs_qa_contact_op_sys_other_mobile	Percent
qa_contact	percent_bugs_qa_contact_product_classification_client_software	Percent

qa_contact	percent_bugs_qa_contact_product_classification_components	Percent
qa_contact	percent_bugs_qa_contact_product_classification_server_software	Percent

Table 91: Dominant knowledge paradigm independent variables at individual level

Role	Independent variable	Variable type
reporter	percent_bugs_reported_violated_bug_lifecycle	Percent
reporter	percent_bugs_reported_reopened_at_least_once	Percent
reporter	percent_bugs_reported_reassigned_at_least_once	Percent
reporter	percent_bugs_reported_ target_milestone_changed_at_least_once_count	Percent
reporter	percent_bugs_reported_severity_changed_at_least_once_count	Percent
reporter	(log) bugs_reported_activity_1_3days_mean	Continuous
reporter	(log) bugs_reported_activity_3_7days_mean	Continuous
reporter	(log) bugs_reported_activity_7_15days_mean	Continuous
reporter	(log) bugs_reported_activity_15_45days_mean	Continuous
reporter	(log) bugs_reported_activity_45_90days_mean	Continuous
reporter	(log) bugs_reported_activity_90_180days_mean	Continuous
reporter	(log) bugs_reported_activity_180days_1year_mean	Continuous
assigned_to	percent_bugs_assigned_to_violated_bug_lifecycle	Percent
assigned_to	percent_bugs_assigned_to_reopened_at_least_once	Percent
assigned_to	percent_bugs_assigned_to_reassigned_at_least_once	Percent
assigned_to	percent_bugs_assigned_to_ target_milestone_changed_at_least_once_count	Percent
assigned_to	percent_bugs_assigned_to_severity_changed_at_least_once_count	Percent
assigned_to	(log) bugs_assigned_to_activity_1_3days_mean	Continuous
assigned_to	(log) bugs_assigned_to_activity_3_7days_mean	Continuous
assigned_to	(log) bugs_assigned_to_activity_7_15days_mean	Continuous
assigned_to	(log) bugs_assigned_to_activity_15_45days_mean	Continuous
assigned_to	(log) bugs_assigned_to_activity_45_90days_mean	Continuous
assigned_to	(log) bugs_assigned_to_activity_90_180days_mean	Continuous
assigned_to	(log) bugs_assigned_to_activity_180days_1year_mean	Continuous
qa_contact	percent_bugs_qa_contact_violated_bug_lifecycle	Percent
qa_contact	percent_bugs_qa_contact_reopened_at_least_once	Percent
qa_contact	percent_bugs_qa_contact_reassigned_at_least_once	Percent
qa_contact	percent_bugs_qa_contact_ target_milestone_changed_at_least_once_count	Percent
qa_contact	percent_bugs_qa_contact_severity_changed_at_least_once_count	Percent
qa_contact	(log) bugs_qa_contact_activity_1_3days_mean	Continuous
qa_contact	(log) bugs_qa_contact_activity_3_7days_mean	Continuous
qa_contact	(log) bugs_qa_contact_activity_7_15days_mean	Continuous
qa_contact	(log) bugs_qa_contact_activity_15_45days_mean	Continuous
qa_contact	(log) bugs_qa_contact_activity_45_90days_mean	Continuous
qa_contact	(log) bugs_qa_contact_activity_90_180days_mean	Continuous
qa_contact	(log) bugs_qa_contact_activity_180days_1year_mean	Continuous

Table 92: Knowledge flow impediments independent variables at individual level

Role	Independent variable	Variable type
All	is_core_actor	Logical
All	(log) watching_all_actors_count	Count
All	(log) watching_all_orgs_count	Count
All	(log) watching_knowledge_actors_count	Count
All	(log) watching_core_actors_count	Count
All	(log) watching_peripheral_actors_count	Count
All	(log) watched_by_all_actors_count	Count
All	(log) watched_by_all_orgs_count	Count
All	(log) watched_by_knowledge_actors_count	Count
All	(log) watched_by_core_actors_count	Count
All	(log) watched_by_peripheral_actors_count	Count
reporter	(log) bugs_reported_votes_core_actors_mean	Continuous
reporter	(log) bugs_reported_votes_knowledge_actors_mean	Continuous
reporter	(log) bugs_reported_votes_peripheral_actors_mean	Continuous
reporter	(log) bugs_reported_cc_core_actors_mean	Continuous
reporter	(log) bugs_reported_cc_knowledge_actors_mean	Continuous
reporter	(log) bugs_reported_cc_peripheral_actors_mean	Continuous
reporter	(log) bugs_reported_comments_distinct_actor_mean	Continuous
reporter	(log) bugs_reported_comments_core_actors_mean	Continuous
reporter	(log) bugs_reported_comments_peripheral_actors_mean	Continuous
reporter	(log) bugs_reported_comments_knowledge_actors_mean	Continuous
assigned_to	(log) bugs_assigned_to_votes_core_actors_mean	Continuous
assigned_to	(log) bugs_assigned_to_votes_knowledge_actors_mean	Continuous
assigned_to	(log) bugs_assigned_to_votes_peripheral_actors_mean	Continuous
assigned_to	(log) bugs_assigned_to_cc_core_actors_mean	Continuous
assigned_to	(log) bugs_assigned_to_cc_knowledge_actors_mean	Continuous
assigned_to	(log) bugs_assigned_to_cc_peripheral_actors_mean	Continuous
assigned_to	(log) bugs_assigned_to_comments_distinct_actor_mean	Continuous
assigned_to	(log) bugs_assigned_to_comments_core_actors_mean	Continuous
assigned_to	(log) bugs_assigned_to_comments_peripheral_actors_mean	Continuous
assigned_to	(log) bugs_assigned_to_comments_knowledge_actors_mean	Continuous
qa_contact	(log) bugs_qa_contact_votes_core_actors_mean	Continuous
qa_contact	(log) bugs_qa_contact_votes_knowledge_actors_mean	Continuous
qa_contact	(log) bugs_qa_contact_votes_peripheral_actors_mean	Continuous
qa_contact	(log) bugs_qa_contact_cc_core_actors_mean	Continuous
qa_contact	(log) bugs_qa_contact_cc_knowledge_actors_mean	Continuous
qa_contact	(log) bugs_qa_contact_cc_peripheral_actors_mean	Continuous
qa_contact	(log) bugs_qa_contact_comments_distinct_actor_mean	Continuous
qa_contact	(log) bugs_qa_contact_comments_core_actors_mean	Continuous
qa_contact	(log) bugs_qa_contact_comments_peripheral_actors_mean	Continuous
qa_contact	(log) bugs_qa_contact_comments_knowledge_actors_mean	Continuous

Table 93: Knowledge stakeholder influence independent variables at individual level

Role	Independent variable	Variable type
reporter	percent_bugs_reported_trivial	Percent
reporter	percent_bugs_reported_minor	Percent
reporter	percent_bugs_reported_normal	Percent
reporter	percent_bugs_reported_major	Percent
reporter	percent_bugs_reported_critical	Percent
reporter	percent_bugs_reported_blocker	Percent
reporter	percent_bugs_reported_priority_P1	Percent
reporter	percent_bugs_reported_priority_P2	Percent
reporter	percent_bugs_reported_priority_P3	Percent
reporter	percent_bugs_reported_priority_P4	Percent
reporter	percent_bugs_reported_priority_P5	Percent
reporter	percent_bugs_reported_priority_changed_at_least_once_count	Percent
reporter	percent_bugs_reported_severity_changed_at_least_once_count	Percent
reporter	(log) bugs_reported_votes_all_actors_mean	Continuous
reporter	(log) bugs_reported_cc_all_actors_mean	Continuous
reporter	(log) bugs_reported_flags_mean	Continuous
reporter	(log) bugs_reported_comments_all_actors_mean	Continuous
reporter	percent_bugs_reported_has_top_3_keyword	Percent
reporter	percent_bugs_reported_has_top_10_keyword	Percent
reporter	percent_bugs_reported_has_top_25_keyword	Percent
reporter	percent_bugs_reported_has_top_50_keyword	Percent
assigned_to	percent_bugs_assigned_to_trivial	Percent
assigned_to	percent_bugs_assigned_to_minor	Percent
assigned_to	percent_bugs_assigned_to_normal	Percent
assigned_to	percent_bugs_assigned_to_major	Percent
assigned_to	percent_bugs_assigned_to_critical	Percent
assigned_to	percent_bugs_assigned_to_blocker	Percent
assigned_to	percent_bugs_assigned_to_priority_P1	Percent
assigned_to	percent_bugs_assigned_to_priority_P2	Percent
assigned_to	percent_bugs_assigned_to_priority_P3	Percent
assigned_to	percent_bugs_assigned_to_priority_P4	Percent
assigned_to	percent_bugs_assigned_to_priority_P5	Percent
assigned_to	percent_bugs_assigned_to_priority_changed_at_least_once_count	Percent
assigned_to	percent_bugs_assigned_to_severity_changed_at_least_once_count	Percent
assigned_to	(log) bugs_assigned_to_votes_all_actors_mean	Continuous
assigned_to	(log) bugs_assigned_to_cc_all_actors_mean	Continuous
assigned_to	(log) bugs_assigned_to_flags_mean	Continuous
assigned_to	(log) bugs_assigned_to_comments_all_actors_mean	Continuous
assigned_to	percent_bugs_assigned_to_has_top_3_keyword	Percent
assigned_to	percent_bugs_assigned_to_has_top_10_keyword	Percent
assigned_to	percent_bugs_assigned_to_has_top_25_keyword	Percent
assigned_to	percent_bugs_assigned_to_has_top_50_keyword	Percent
qa_contact	percent_bugs_qa_contact_trivial	Percent

qa_contact	percent_bugs_qa_contact_minor	Percent
qa_contact	percent_bugs_qa_contact_normal	Percent
qa_contact	percent_bugs_qa_contact_major	Percent
qa_contact	percent_bugs_qa_contact_critical	Percent
qa_contact	percent_bugs_qa_contact_blocker	Percent
qa_contact	percent_bugs_qa_contact_priority_P1	Percent
qa_contact	percent_bugs_qa_contact_priority_P2	Percent
qa_contact	percent_bugs_qa_contact_priority_P3	Percent
qa_contact	percent_bugs_qa_contact_priority_P4	Percent
qa_contact	percent_bugs_qa_contact_priority_P5	Percent
qa_contact	percent_bugs_qa_contact_priority_changed_at_least_once_count	Percent
qa_contact	percent_bugs_qa_contact_severity_changed_at_least_once_count	Percent
qa_contact	(log) bugs_qa_contact_votes_all_actors_mean	Continuous
qa_contact	(log) bugs_qa_contact_cc_all_actors_mean	Continuous
qa_contact	(log) bugs_qa_contact_flags_mean	Continuous
qa_contact	(log) bugs_qa_contact_comments_all_actors_mean	Continuous
qa_contact	percent_bugs_qa_contact_has_top_3_keyword	Percent
qa_contact	percent_bugs_qa_contact_has_top_10_keyword	Percent
qa_contact	percent_bugs_qa_contact_has_top_25_keyword	Percent
qa_contact	percent_bugs_qa_contact_has_top_50_keyword	Percent

Table 94: Solution knowledge value independent variables at individual level

At the outset, a pair of models, consisting of one control-only model and one control plus independent variable model, was created for each of the 21 dependent variables for each of the six hypotheses, resulting in 252 models at the individual level. For each model, the goodness of fit and significance of calculated coefficients for each variable were evaluated using OLS regression, logistic regression, beta regression, and post-fit ANCOVA with heteroskedasticity correction, as appropriate, depending on the nature of the dependent variable of each model, as discussed in previous sections. The measures used to assess the model fits and test the hypotheses were the same as those used at the problem level of analysis, with the exception of those related to beta regression, which was not used at the problem level of analysis. In the case of beta regression, the same measures as those used for logistic regression were calculated, because the analysis process for beta regression is similar to that used for generalized linear models with logit link (Smithson & Verkuilen, 2006; Cribari-Neto & Zeileis, 2010; Simas, Barreto-Souza, & Rocha, 2010; Grün, Kosmidis, and Zeileis, 2012).

Organisation level of analysis

As discussed in the previous chapter, organisation level dependent variables were operationalized to conform to the data constraints and to maximize validity and reliability of the analytical process. The operationalization created 2 logical variables, 4 percentage variables, and 1 continuous variable for each of the aggregate roles in which members of an organisation engage, resulting in 21 dependent variables in total. The role separation of the variables is also necessary at the organisational level because organisations are defined as aggregations of individuals, who, in turn, participate in the knowledge creation process orthogonal to the properties of the knowledge itself. The aggregation of actors engaging in each role that are also in the same organisation creates a distinct representation of the influence of the organisation on the knowledge production process by virtue of its members. Comparison across organisations allows the separation of organisation level effects from those that are better attributed to the individual.

In order to ensure that there is sufficient variability and distinctiveness between the organisation level and the individual level, a threshold is necessary to restrict the data sample frame to organisations that have a sufficient number of distinct actors. Organisations with too few actors confound the distinction between an individual and organisation level influence on the knowledge production process. Since the goal of this study is to capture large scale effects, excluding organisations with too few actors results in a conservative estimate of any organisation level effects, as desired.

The distribution of number of actors in each organisation, as identified by the “domain” in their profile is described in Table 95. The distribution shows that the “domain” identification token often simply represents an individual level actor, with more than 85% of “organisations” thus identified consisting of only a single actor. The proportion of “organisations” with more than one actor steadily decreases for two, three, and four actors. At five or more actors, an inflection appears. This inflection is in part due to the inclusion of all numbers of actors beyond five in the single count variable but also reflects the fact that the “domain” identification token is imprecise and will often match what are known as “webmail” or other non-organisational domains used by individuals for their online profiles (Baysal, et al., 2013). To address this spuriousness in the representation of organisation affiliation, a database of known non-organisational domain names (Tarr, 2012) was combined with a manually created list of domain names, resulting in 5490 domain names that were known to not be organisational in nature. All profiles that were registered with those domain names were excluded from the sample frame, resulting in the distribution described in Table 96.

Number of actors	Count
Organisations with any number of actors	111,047 (100.000%)
Organisations with only one actor	94,569 (85.161%)
Organisations with only two actors	8,809 (7.933%)
Organisations with only three actors	2,639 (2.376%)
Organisations with only four actors	1,177 (1.060%)
Organisations with five or more actors	3,3853 (3.470%)

Table 95: Distribution of distinct individual actors in organisations

Number of actors	Count
Organisations with any number of actors	109,366 (98.486%)
Organisations with only one actor	94,188 (84.818%)
Organisations with only two actors	8,631 (7.772%)
Organisations with only three actors	2,532 (2.280%)
Organisations with only four actors	1,101 (0.991%)
Organisations with five or more actors	2,914 (2.624%)

Table 96: Distribution of distinct individual actors in organisations with non-organisational domains excluded

While the exclusion of known non-organisational domains did not significantly reduce the overall numbers, it resulted in the elimination of nearly 25% of the "organisations" with five or more actors, which stands to reason as non-organisation domains, such as "hotmail.com" or "gmail.com", have many registered users.

In order to determine the appropriate threshold for minimum number of actors, the profiles associated with organisations with only two actors were manually inspected in the data. The examination revealed that in many cases, the "two" profiles associated with the same organisation were likely managed by a single individual actor. For example, in some cases, the "two" profiles had similar names with slightly different spellings, suggesting one spelling was an error. In other cases, generic names were used such as "support@organisation.com" in addition to a named actor, i.e., "john.smith@organisation.com". In such cases, it was not clear that there were distinct actors in the organisation because both profiles were likely to be operated by a single individual actor. Therefore, a threshold of a minimum of 3 actors was selected for inclusion of an organisation in the sample frame, resulting in n = 6,547 organisations. Manual inspection of the retained organisations further validated that the organisations in the constrained sample frame were representative of organisational actors and distinct from individual actors, as desired.

As discussed in the profile level of analysis section, there are three roles in which actors participate in the knowledge creation process. At the organisation level of analysis, these roles are aggregated according to organisation rather than profile. Similar to at the individual level of analysis, a minimum threshold of 4 actions in each aggregate role was set to ensure that there is sufficient distinctiveness between organisation level factors and problem level factors. However, at the organisation level, the "actions" need not be taken by the same individual. Different actors in the organisation may collectively engage in an aggregate role in the knowledge creation process in order to reach the 4-action threshold for inclusion. The application of this further constraint resulted in the distribution described in Table 97.

Constraints	**Count**
Organisations with three or more actors	6,547 (100.000%)
Organisations with three or more actors that have acted in reporter role at least four times	2,338 (35.711%)
Organisations with three or more actors that have acted in assigned_to role at least four times	206 (3.146%)
Organisations with three or more actors that have acted in qa_contact role at least four times	42 (0.642%)

Table 97: Distribution of role engagement of organisations with three or more actors

The distribution shows that, as theoretically expected, fewer organisations participate in the solution knowledge producer and solution knowledge verifier aggregate roles. In particular, manual inspection of the 42 retained organisations that participate in the solution knowledge verifier aggregate role revealed that, as expected, they are organisations with specialized knowledge and strategies that pertain to technical areas developed by the Mozilla meta-organisation, including IBM, Adobe, Dreamhost, MIT, Nokia, Oracle, Redhat, Sun, Qualcomm, and Pixar. While the constraints traded sample size for validity, given that the resulting sample included these prominent organisations that were theorized to have motivations to participate in the knowledge production process of open source meta-organisations (MacAulay, 2013), the choice was made to proceed with the analysis with the smaller sample sizes and to carefully consider the implications of the sample size limitations when interpreting results in the next chapter.

Dependent variables: Reopening and reassigning tendencies of each aggregate role

The dependent variables at the organisation level of analysis were analyzed in a manner similar to at the individual level of analysis. The major distinction was the aggregation of roles for participation in the knowledge creation process according to organisation, adjusting the nature of the constraints on the data to fit the operational definition of organisation, as discussed in the previous section. The same analytical refinement process was applied as the one described in the section on the individual level of analysis (not shown –see Appendix C: Additional analysis details) because the variables were similar except with different constraints and aggregation. The result of the process was the selection of the same analytical methods as for the individual level of analysis, which facilitates cross-level comparisons of the results to better localize the level of any observed effects.

The summary statistics of the logical and (non-zero) percent variables for reopening and reassigning tendencies of the aggregate problem knowledge producer (reporter) role, the aggregate solution knowledge producer (assigned_to) role, and the aggregate solution knowledge verifier (QA_contact) role are described in Table 98 and Table 99; the quantiles of the percent variables are described in Table 100; and, the normal QQ-plots of the percent variables are depicted in Appendix C: Additional analysis details.

Aggregate role variable	N	FALSE	TRUE
reported_reopened_at_least_once	2338	1021	1317
reported_reassigned_at_leaset_once	2338	1603	735
assigned_to_reopened_at_least_once	206	51	155
assigned_to_reassigned_at_least_once	206	75	131
qa_contact_reopened_at_least_once	42	4	38
qa_contact_reassigned_at_least_once	42	10	32

Table 98: Distribution of whether or not each organisation acted in each aggregate role upon at least one bug that was reopened / reassigned at least once at organisation level

Aggregate role variable	N	Mean	Median	Stdev	Min	Max
percent_bugs_reported_reopened_at_least_once	1317	0.144	0.125	0.089	0.015	0.600
percent_bugs_reported_reassigned_at_least_once	735	0.110	0.083	0.092	0.006	0.667
percent_bugs_assigned_to_reopened_at_least_once	155	0.133	0.105	0.085	0.016	0.429
percent_bugs_assigned_to_reassigned_at_least_once	131	0.118	0.077	0.121	0.005	0.667
percent_bugs_qa_contact_reopened_at_least_once	38	0.123	0.101	0.088	0.018	0.500
percent_bugs_qa_contact_reassigned_at_least_once	32	0.117	0.079	0.099	0.009	0.351

Table 99: Summary statistics of constrained percentage of bugs acted upon by organisation members in each aggregate role that were reopened / reassigned once or more at organisation level

Aggregate role variables	Quantiles										
	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
percent_bugs_reported_reopened_at_least_once	0.015	0.055	0.071	0.091	0.105	0.125	0.143	0.167	0.200	0.250	0.600
percent_bugs_reported_reassigned_at_least_once	0.006	0.026	0.040	0.053	0.067	0.083	0.105	0.129	0.167	0.250	0.667
percent_bugs_assigned_to_reopened_at_least_once	0.016	0.052	0.067	0.080	0.092	0.105	0.127	0.156	0.188	0.253	0.429
percent_bugs_assigned_to_reassigned_at_least_once	0.005	0.017	0.032	0.042	0.059	0.077	0.103	0.125	0.197	0.263	0.667
percent_bugs_qa_contact_reopened_at_least_once	0.018	0.041	0.062	0.084	0.093	0.101	0.111	0.138	0.171	0.219	0.500
percent_bugs_qa_contact_reassigned_at_least_once	0.009	0.014	0.026	0.044	0.056	0.079	0.121	0.175	0.241	0.250	0.351

Table 100: Quantiles of constrained percentage of bugs acted upon by organisation members in aggregate roles that were reopened / reassigned once or more at organisation level

As was the case for the analysis of the individual level dependent variable counterparts, the logical reopening and reassigning tendency variables were analyzed using logistic regression and the percent reopening and reassigning tendency variables were analyzed using beta regression. In addition, analysis of covariance (ANCOVA type II) was conducted on the respective logistic and beta regression model fits.

Dependent variables: Outcome tendencies of each aggregate role

The summary statistics of the two percent outcome tendency variables, percent fixed and percent fixed with patch, for each aggregate role, are described in Table 101; the quantiles are described in Table 102; and, the normal Q-Q plots are depicted in Appendix C: Additional analysis details. All of these percent variables were analyzed using beta regression and analysis of covariance.

Aggregate role variables	N	Mean	Median	Stdev	Min	Max
percent_bugs_reported_fixed	2337	0.181	0.141	0.206	0.000	1.000
percent_bugs_reported_fixed_at_least_one_patch	2337	0.088	0.000	0.147	0.000	1.000
percent_bugs_assigned_to_fixed	205	0.769	0.836	0.232	0.000	1.000
percent_bugs_assigned_to_fixed_at_least_one_patch	205	0.533	0.600	0.358	0.000	1.000
percent_bugs_qa_contact_fixed	42	0.676	0.695	0.223	0.143	1.000
percent_bugs_qa_contact_fixed_at_least_one_patch	42	0.327	0.306	0.244	0.000	1.000

Table 101: Summary statistics of constrained percentage of bugs acted upon by organisation members in each aggregate role that were fixed / fixed with at least one patch at organisation level

	Quantiles										
Aggregate role variables	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
percent_bugs_reported_fixed	0.00	0.00	0.00	0.00	0.08	0.14	0.18	0.23	0.32	0.50	1.00
percent_bugs_reported_fixed_at_least_one_patch	0.00	0.00	0.00	0.00	0.00	0.00	0.05	0.10	0.17	0.26	1.00
percent_bugs_assigned_to_fixed	0.00	0.43	0.60	0.71	0.80	0.84	0.88	0.91	0.96	1.00	1.00
percent_bugs_assigned_to_fixed_at_least_one_patch	0.00	0.01	0.11	0.25	0.40	0.60	0.74	0.84	0.91	1.00	1.00
percent_bugs_qa_contact_fixed	0.14	0.32	0.50	0.60	0.63	0.69	0.72	0.84	0.90	0.97	1.00
percent_bugs_qa_contact_fixed_at_least_one_patch	0.00	0.00	0.10	0.18	0.25	0.61	0.38	0.44	0.50	0.69	1.00

Table 102: Quantiles of constrained percentage of bugs acted upon by organisation members in each aggregate role that were fixed / fixed with at least one patch at organisation level

The summary statistics of the constrained variables for resolution timing tendencies for each aggregate role are described in Table 103; the quantiles are described in Table 104; and, the normal Q-Q plots are depicted in Appendix C: Additional analysis details. Once again, the log transformation of the resolution timing variables was selected as most appropriate both in terms of analytical validity and comparability amongst aggregates roles and across levels of analysis. OLS regression and ANCOVA (type II), both with heteroskedasticity correction, were used for the analysis.

Aggregate role variables	**N**	**Mean**	**Median**	**Stdev**	**Min**	**Max**	**Skewness**	**Kurtosis**
bugs_reported_mean_days_to_resolution	2338	101.35	75.43	103.21	0.025	1076.6	2.999	14.427
bugs_assigned_to_mean_days_to_resolution	206	175.95	111.80	196.94	1.01	1034.55	2.105	4.547
bugs_qa_contact_mean_days_to_resolution	42	156.97	130.80	111.24	4.30	420.1	0.656	-0.417

Table 103: Summary statistics of constrained resolution timing tendency variables for each aggregate role at organisation level

	Quantiles										
Aggregate role variables	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
bugs_reported_mean_days_to_resolution	0.02	15.92	30.91	44.57	58.22	74.53	92.28	114.04	148.57	208.71	1076.6
bugs_assigned_to_mean_days_to_resolution	1.01	20.65	40.87	58.71	79.15	111.78	148.71	188.93	238.77	451.43	1034.6
bugs_qa_contact_mean_days_to_resolution	4.30	19.90	56.61	93.32	110.13	130.82	158.08	228.04	254.33	322.49	420.07

Table 104: Quantiles of constrained resolution timing tendency variables for each aggregate role at organisation level

The organisation level dependent variables for each of the aggregate roles in which organisation members engage and the regression types selected to perform the analyses are described in Table 105.

Aggregate role dependent variables	**Variable type**	**Regression type** (Heteroskedasticity correction in all)
At least one bug acted upon by organisation members in aggregate reporter role was reopened	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
(Non-zero) Percent of bugs acted upon by organisation members in aggregate reporter role that were reopened at least once	Percent	Beta regression + ANCOVA (Type II)
At least one bug acted upon by organisation members in aggregate reporter role was reassigned	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
(Non-zero) Percent of bugs acted upon by organisation members in aggregate reporter role that were reassigned at least once	Percent	Beta regression + ANCOVA (Type II)
Percent of bugs acted upon by organisation members in aggregate reporter role that were fixed	Percent	Beta regression + ANCOVA (Type II)
Percent of bugs acted upon by organisation members in aggregate reporter role that were fixed with at least one patch	Percent	Beta regression + ANCOVA (Type II)
(log) Mean resolution time of bugs acted upon by organisation members in aggregate reporter role	Continuous	Ordinary least squares (OLS) + ANCOVA (Type II)
At least one bug acted upon by organisation members in aggregate assigned_to role was reopened	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
(Non-zero) Percent of bugs acted upon by organisation members in aggregate assigned_to role that were reopened at least once	Percent	Beta regression + ANCOVA (Type II)
At least one bug acted upon by organisation members in aggregate assigned_to role was reassigned	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
(Non-zero) Percent of bugs acted upon by organisation members in aggregate assigned_to role that were reassigned at least once	Percent	Beta regression + ANCOVA (Type II)
Percent of bugs acted upon by organisation members in aggregate assigned_to role that were fixed	Percent	Beta regression + ANCOVA (Type II)
Percent of bugs acted upon by organisation members in aggregate assigned_to role that were fixed with at least one patch	Percent	Beta regression + ANCOVA (Type II)
(log) Mean resolution time of bugs acted upon by organisation members in aggregate assigned_to role	Continuous	Ordinary least squares (OLS) + ANCOVA (Type II)
At least one bug acted upon by organisation members in aggregate qa_contact role was reopened	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)

(Non-zero) Percent of bugs acted upon by organisation members in aggregate qa_contact role that were reopened at least once	Percent	Beta regression + ANCOVA (Type II)
At least one bug acted upon by organisation members in aggregate qa_contact role was reassigned	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)
(Non-zero) Percent of bugs acted upon by organisation members in aggregate qa_contact role that were reassigned at least once	Percent	Beta regression + ANCOVA (Type II)
Percent of bugs acted upon by organisation members in aggregate qa_contact role that were fixed	Percent	Beta regression + ANCOVA (Type II)
Percent of bugs acted upon by organisation members in aggregate qa_contact role that were fixed with at least one patch	Percent	Beta regression + ANCOVA (Type II)
(log) Mean resolution time of bugs acted upon by organisation members in aggregate qa_contact role	Continuous	Ordinary least squares (OLS) + ANCOVA (Type II)

Table 105: Aggregate role dependent variables and chosen regression types at organisation level

Modelling: Organisation level control variables

Modelling was conducted in two stages in a manner similar to the approach used at the individual level of analysis. In the first stage, control variables suitable to the organisation level of analysis were selected from the literature with the added intention of maintaining as close a match to the individual level control variables as possible to promote comparison during the interpretation of the results. As was the case at the individual level, many of the control variables were necessarily defined relative to the aggregate role in which each organisation's members engaged upon problems. In cases where the variables had significantly non-linear distribution properties amongst organisations, the log transformation was taken to normalize the variables for analysis.

For the aggregate problem knowledge producer role, eight control variables were identified that were expected to affect organisation level outcome measures: 1) the (log) number of actors in the organisation; 2) the percent of actors in the organisation that were "core actors";

3) the (log) activity count of the organisation; 4) the (log) number of times the individuals in the organisation engaged in the aggregate problem knowledge producer (reporter) role; 5) the (log) mean resolution time of problems acted upon by organisation members in the aggregate reporter role; 6) the percent of bugs reported by organisation members that were duplicates; 7) the percent of bugs reported by organisation members that were fixed; and, 8) the percent of bugs reported by organisation members that violated the bug life cycle.

For the solution knowledge producer (assigned_to) and solution knowledge verifier (QA_contact) aggregate roles, seven counterparts of each of these control variables were used, relative to their respective aggregate roles; e.g., for the regression models created to analyze the solution knowledge producer (assigned_to) aggregate role, the counterpart to control variable 7) was percent of bugs acted upon by organisation members in aggregate assigned_to role that were fixed. Control variable 6), the percent of bugs reported by organisation members that were duplicates, is only defined for the problem knowledge producer aggregate role, because duplicate bugs, by definition, never proceed along the bug life cycle to solution knowledge creation or solution knowledge verification. As a result, this control variable was only used for models that analyzed the aggregate reporter role at the organisation level.

As discussed in the operationalization chapter and in the section on individual level analysis, in cases where a control variable was the same as or directly correlated to a given dependent variable, it was omitted from the regression model. For each of the 21 organisation level dependent variables (seven for each of the three aggregate roles in which members of organisations engage), a test regression model was fixed with the eight (or seven, for assigned_to and QA_contact aggregate roles) control variable candidates as independent variables.

Examination of the analysis of variance and type II ANCOVA models with heteroskedasticity correction revealed which of the candidate control variables were significant for each dependent variable a $p < 0.05$ degree of certainty. In each case, those variables that were found to be significant were retained as controls and those that were found to not be significant were dropped from the model, as per standard statistical modeling practice (Tabachnick & Fidell, 2007). The control variables, the aggregate roles to which they pertain in the analysis, and their variable types are summarized in Table 106.

Aggregate role	Control variable	Variable type
All	(log) all_actors_count	Count
All	percent_core_actors	Percent
All	(log) activity_count	Count
Problem knowledge producer (reporter)	(log) bugs_reported_count	Count
Problem knowledge producer (reporter)	(log) bugs_reported_ mean_days_to_resolution	Continuous
Problem knowledge producer (reporter)	percent_bugs_reported_ is_duplicate	Percent
Problem knowledge producer (reporter)	percent_bugs_reported_fixed	Percent
Problem knowledge producer (reporter)	percent_bugs_reported_ violated_bug_lifecycle	Percent
Solution knowledge producer (assigned_to)	(log) bugs_assigned_to_count	Count
Solution knowledge producer (assigned_to)	(log) bugs_assigned_to_ mean_days_to_resolution	Continuous
Solution knowledge producer (assigned_to)	percent_bugs_assigned_to_fixed	Percent
Solution knowledge producer (assigned_to)	percent_bugs_assigned_to_ violated_bug_lifecycle	Percent
Solution knowledge verifier (qa_contact)	(log) bugs_qa_contact_count	Count
Solution knowledge verifier (qa_contact)	(log) bugs_qa_contact_ mean_days_to_resolution	Continuous
Solution knowledge verifier (qa_contact)	percent_bugs_qa_contact_fixed	Percent
Solution knowledge verifier (qa_contact)	percent_bugs_qa_contact_ violated_bug_lifecycle	Percent

Table 106: Control variables at organisation level

Modelling: Organisation level independent variables

During the second stage of modelling, the significant controls for each dependent variable were combined with the independent variables described in the organisation level of the conceptual framework to create regression models that test each hypothesis. Each of the 21 models (7 dependent variables for each of the 3 aggregate roles in which members of organisations engage) was created in the standard regression model form in a manner similar to that discussed in the problem level analysis section.

The name, type, and related aggregate role of each independent variable associated to each hypothesis at the organisation level of analysis are summarized as follows: Absorptive capacity in Table 107; codifiability in Table 108; dominant knowledge paradigm in Table 109; knowledge flow impediments in Table 110; knowledge stakeholder influence in Table 111; and, solution knowledge value in Table 112.

Aggregate role	Independent variable	Variable type
reporter	(log) bugs_reported_count	Count
assigned_to	(log) bugs_assigned_to_count	Count
qa_contact	(log) bugs_qa_contact_count	Count
All	(log) activity_count	Count
All	(log) activity_cc_change_count	Count
All	(log) activity_keywords_change_count	Count
All	(log) activity_product_change_count	Count
All	(log) activity_component_change_count	Count
All	(log) activity_status_change_count	Count
All	(log) activity_resolution_change_count	Count
All	(log) activity_flags_change_count	Count
All	(log) activity_whiteboard_change_count	Count
All	(log) activity_target_milestone_change_count	Count
All	(log) activity_description_change_count	Count
All	(log) activity_priority_change_count	Count
All	(log) activity_severity_change_count	Count
All	(log) activity_assigning_count	Count
All	(log) activity_reassigning_count	Count
All	(log) activity_reopening_count	Count
All	(log) activity_rep_platform_All_count	Count
All	(log) activity_rep_platform_PowerPC_count	Count
All	(log) activity_rep_platform_x86_64_count	Count
All	(log) activity_rep_platform_x86_count	Count
All	(log) activity_rep_platform_Combined_Other_count	Count
All	(log) activity_op_sys_Mac_pc_count	Count
All	(log) activity_op_sys_Windows_pc_count	Count
All	(log) activity_op_sys_Windows_mobile_count	Count
All	(log) activity_op_sys_iOS_mobile_count	Count
All	(log) activity_op_sys_other_mobile_count	Count
All	(log) activity_op_sys_other_pc_count	Count
All	(log) activity_product_classification_client_software_count	Count
All	(log) activity_product_classification_components_count	Count
All	(log) activity_product_classification_server_software_count	Count
All	(log) activity_product_classification_Combined_Other_count	Count
All	(log) activity_bugs_low_severity_count	Count
All	(log) activity_bugs_average_severity_count	Count
All	(log) activity_bugs_high_severity_count	Count

Table 107: Absorptive capacity independent variables at organisation level

Aggregate role	Independent variable	Variable type
reporter	(log) bugs_reported_all_types_description_mean_length	Continuous
reporter	bugs_reported_description_readability_Flesch_reading_ease_mean	Continuous
reporter	(log) bugs_reported_attachments_all_types_count	Count
reporter	percent_bugs_reported_is_duplicate	Percent
reporter	percent_bugs_reported_was_duplicated	Percent
reporter	(log) bugs_reported_all_types_comments_mean_length	Continuous
reporter	(log) bugs_reported_comments_manual_count	Count
assigned_to	(log) bugs_assigned_to_all_types_description_mean_length	Continuous
assigned_to	bugs_assigned_to_description_readability_Flesch_reading_ease_mean	Continuous
assigned_to	(log) bugs_assigned_to_attachments_all_types_count	Count
assigned_to	(log) bugs_assigned_to_all_types_comments_mean_length	Continuous
assigned_to	(log) bugs_assigned_to_comments_manual_count	Count
qa_contact	(log) bugs_qa_contact_all_types_description_mean_length	Continuous
qa_contact	bugs_qa_contact_description_readability_Flesch_reading_ease_mean	Continuous
qa_contact	(log) bugs_qa_contact_attachments_all_types_count	Count
qa_contact	(log) bugs_qa_contact_all_types_comments_mean_length	Continuous
qa_contact	(log) bugs_qa_contact_comments_manual_count	Count

Table 108: Codifiability independent variables at organisation level

Aggregate role	Independent variable	Variable type
reporter	percent_bugs_reported_rep_platform_All	Percent
reporter	percent_bugs_reported_rep_platform_PowerPC	Percent
reporter	percent_bugs_reported_rep_platform_x86_64	Percent
reporter	percent_bugs_reported_rep_platform_x86	Percent
reporter	percent_bugs_reported_op_sys_All	Percent
reporter	percent_bugs_reported_op_sys_Android	Percent
reporter	percent_bugs_reported_op_sys_Linux	Percent
reporter	percent_bugs_reported_op_sys_Mac_pc	Percent
reporter	percent_bugs_reported_op_sys_Windows_pc	Percent
reporter	percent_bugs_reported_op_sys_Windows_mobile	Percent
reporter	percent_bugs_reported_op_sys_iOS_mobile	Percent
reporter	percent_bugs_reported_op_sys_other_mobile	Percent
reporter	percent_bugs_reported_product_classification_client_software	Percent
reporter	percent_bugs_reported_product_classification_components	Percent
reporter	percent_bugs_reported_product_classification_server_software	Percent
assigned_to	percent_bugs_assigned_to_rep_platform_All	Percent
assigned_to	percent_bugs_assigned_to_rep_platform_PowerPC	Percent
assigned_to	percent_bugs_assigned_to_rep_platform_x86_64	Percent
assigned_to	percent_bugs_assigned_to_rep_platform_x86	Percent
assigned_to	percent_bugs_assigned_to_op_sys_All	Percent
assigned_to	percent_bugs_assigned_to_op_sys_Android	Percent
assigned_to	percent_bugs_assigned_to_op_sys_Linux	Percent
assigned_to	percent_bugs_assigned_to_op_sys_Mac_pc	Percent
assigned_to	percent_bugs_assigned_to_op_sys_Windows_pc	Percent
assigned_to	percent_bugs_assigned_to_op_sys_Windows_mobile	Percent
assigned_to	percent_bugs_assigned_to_op_sys_iOS_mobile	Percent
assigned_to	percent_bugs_assigned_to_op_sys_other_mobile	Percent
assigned_to	percent_bugs_assigned_to_product_classification_client_software	Percent
assigned_to	percent_bugs_assigned_to_product_classification_components	Percent
assigned_to	percent_bugs_assigned_to_product_classification_server_software	Percent
qa_contact	percent_bugs_qa_contact_rep_platform_All	Percent
qa_contact	percent_bugs_qa_contact_rep_platform_PowerPC	Percent
qa_contact	percent_bugs_qa_contact_rep_platform_x86_64	Percent
qa_contact	percent_bugs_qa_contact_rep_platform_x86	Percent
qa_contact	percent_bugs_qa_contact_op_sys_All	Percent
qa_contact	percent_bugs_qa_contact_op_sys_Android	Percent
qa_contact	percent_bugs_qa_contact_op_sys_Linux	Percent
qa_contact	percent_bugs_qa_contact_op_sys_Mac_pc	Percent
qa_contact	percent_bugs_qa_contact_op_sys_Windows_pc	Percent
qa_contact	percent_bugs_qa_contact_op_sys_Windows_mobile	Percent
qa_contact	percent_bugs_qa_contact_op_sys_iOS_mobile	Percent
qa_contact	percent_bugs_qa_contact_op_sys_other_mobile	Percent
qa_contact	percent_bugs_qa_contact_product_classification_client_software	Percent

qa_contact	percent_bugs_qa_contact_product_classification_components	Percent
qa_contact	percent_bugs_qa_contact_product_classification_server_software	Percent

Table 109: Dominant knowledge paradigm independent variables at organisation level

Aggregate role	Independent variable	Variable type
reporter	percent_bugs_reported_violated_bug_lifecycle	Percent
reporter	percent_bugs_reported_reopened_at_least_once	Percent
reporter	percent_bugs_reported_reassigned_at_least_once	Percent
reporter	percent_bugs_reported_ target_milestone_changed_at_least_once_count	Percent
reporter	percent_bugs_reported_severity_changed_at_least_once_count	Percent
reporter	(log) bugs_reported_activity_1_3days_mean	Continuous
reporter	(log) bugs_reported_activity_3_7days_mean	Continuous
reporter	(log) bugs_reported_activity_7_15days_mean	Continuous
reporter	(log) bugs_reported_activity_15_45days_mean	Continuous
reporter	(log) bugs_reported_activity_45_90days_mean	Continuous
reporter	(log) bugs_reported_activity_90_180days_mean	Continuous
reporter	(log) bugs_reported_activity_180days_1year_mean	Continuous
assigned_to	percent_bugs_assigned_to_violated_bug_lifecycle	Percent
assigned_to	percent_bugs_assigned_to_reopened_at_least_once	Percent
assigned_to	percent_bugs_assigned_to_reassigned_at_least_once	Percent
assigned_to	percent_bugs_assigned_to_ target_milestone_changed_at_least_once_count	Percent
assigned_to	percent_bugs_assigned_to_severity_changed_at_least_once_count	Percent
assigned_to	(log) bugs_assigned_to_activity_1_3days_mean	Continuous
assigned_to	(log) bugs_assigned_to_activity_3_7days_mean	Continuous
assigned_to	(log) bugs_assigned_to_activity_7_15days_mean	Continuous
assigned_to	(log) bugs_assigned_to_activity_15_45days_mean	Continuous
assigned_to	(log) bugs_assigned_to_activity_45_90days_mean	Continuous
assigned_to	(log) bugs_assigned_to_activity_90_180days_mean	Continuous
assigned_to	(log) bugs_assigned_to_activity_180days_1year_mean	Continuous
qa_contact	percent_bugs_qa_contact_violated_bug_lifecycle	Percent
qa_contact	percent_bugs_qa_contact_reopened_at_least_once	Percent
qa_contact	percent_bugs_qa_contact_reassigned_at_least_once	Percent
qa_contact	percent_bugs_qa_contact_ target_milestone_changed_at_least_once_count	Percent
qa_contact	percent_bugs_qa_contact_severity_changed_at_least_once_count	Percent
qa_contact	(log) bugs_qa_contact_activity_1_3days_mean	Continuous
qa_contact	(log) bugs_qa_contact_activity_3_7days_mean	Continuous
qa_contact	(log) bugs_qa_contact_activity_7_15days_mean	Continuous
qa_contact	(log) bugs_qa_contact_activity_15_45days_mean	Continuous
qa_contact	(log) bugs_qa_contact_activity_45_90days_mean	Continuous
qa_contact	(log) bugs_qa_contact_activity_90_180days_mean	Continuous
qa_contact	(log) bugs_qa_contact_activity_180days_1year_mean	Continuous

Table 110: Knowledge flow impediments independent variables at organisation level

Aggregate role	Independent variable	Variable type
All	percent_core_actors	Percent
All	(log) core_actors_count	Count
All	percent_knowledge_actors	Percent
All	(log) knowledge_actors_count	Count
All	(log) peripheral_actors_count	Count
All	(log) watching_all_actors_count	Count
All	(log) watching_all_orgs_count	Count
All	(log) watching_knowledge_actors_count	Count
All	(log) watching_core_actors_count	Count
All	(log) watched_by_all_actors_count	Count
All	(log) watched_by_all_orgs_count	Count
All	(log) watched_by_knowledge_actors_count	Count
All	(log) watched_by_core_actors_count	Count
reporter	(log) bugs_reported_votes_core_actors_count	Count
reporter	(log) bugs_reported_votes_knowledge_actors_count	Count
reporter	(log) bugs_reported_votes_peripheral_actors_count	Count
reporter	(log) bugs_reported_cc_core_actors_count	Count
reporter	(log) bugs_reported_cc_knowledge_actors_count	Count
reporter	(log) bugs_reported_cc_peripheral_actors_count	Count
reporter	(log) bugs_reported_activity_core_actors_count	Count
reporter	(log) bugs_reported_activity_knowledge_actors_count	Count
reporter	(log) bugs_reported_activity_peripheral_actors_count	Count
reporter	(log) bugs_reported_comments_distinct_actor_mean	Continuous
reporter	(log) bugs_reported_activity_distinct_actor_mean	Continuous
assigned_to	(log) bugs_assigned_to_votes_core_actors_count	Count
assigned_to	(log) bugs_assigned_to_votes_knowledge_actors_count	Count
assigned_to	(log) bugs_assigned_to_votes_peripheral_actors_count	Count
assigned_to	(log) bugs_assigned_to_cc_core_actors_count	Count
assigned_to	(log) bugs_assigned_to_cc_knowledge_actors_count	Count
assigned_to	(log) bugs_assigned_to_cc_peripheral_actors_count	Count
assigned_to	(log) bugs_assigned_to_activity_core_actors_count	Count
assigned_to	(log) bugs_assigned_to_activity_knowledge_actors_count	Count
assigned_to	(log) bugs_assigned_to_activity_peripheral_actors_count	Count
assigned_to	(log) bugs_assigned_to_comments_distinct_actor_mean	Continuous
assigned_to	(log) bugs_assigned_to_activity_distinct_actor_mean	Continuous
qa_contact	(log) bugs_qa_contact_votes_core_actors_count	Count
qa_contact	(log) bugs_qa_contact_votes_knowledge_actors_count	Count
qa_contact	(log) bugs_qa_contact_votes_peripheral_actors_count	Count
qa_contact	(log) bugs_qa_contact_cc_core_actors_count	Count
qa_contact	(log) bugs_qa_contact_cc_knowledge_actors_count	Count
qa_contact	(log) bugs_qa_contact_cc_peripheral_actors_count	Count
qa_contact	(log) bugs_qa_contact_activity_core_actors_count	Count
qa_contact	(log) bugs_qa_contact_activity_knowledge_actors_count	Count
qa_contact	(log) bugs_qa_contact_activity_peripheral_actors_count	Count

qa_contact	(log) bugs_qa_contact_comments_distinct_actor_mean	Continuous
qa_contact	(log) bugs_qa_contact_activity_distinct_actor_mean	Continuous

Table 111: Knowledge stakeholder influence independent variables at organisation level

Aggregate role	Independent variable	Variable type
reporter	percent_bugs_reported_enhancement	Percent
reporter	percent_bugs_reported_trivial	Percent
reporter	percent_bugs_reported_minor	Percent
reporter	percent_bugs_reported_major	Percent
reporter	percent_bugs_reported_critical	Percent
reporter	percent_bugs_reported_blocker	Percent
reporter	percent_bugs_reported_priority_changed_at_least_once_count	Percent
reporter	percent_bugs_reported_severity_changed_at_least_once_count	Percent
reporter	(log) bugs_reported_votes_all_actors_count	Count
reporter	(log) bugs_reported_cc_all_actors_count	Count
reporter	(log) bugs_reported_flags_mean	Continuous
reporter	(log) bugs_reported_comments_all_actors_count	Count
reporter	percent_bugs_reported_has_top_3_keyword	Percent
reporter	percent_bugs_reported_has_top_10_keyword	Percent
reporter	percent_bugs_reported_has_top_25_keyword	Percent
reporter	percent_bugs_reported_has_top_50_keyword	Percent
assigned_to	percent_bugs_assigned_to_enhancement	Percent
assigned_to	percent_bugs_assigned_to_trivial	Percent
assigned_to	percent_bugs_assigned_to_minor	Percent
assigned_to	percent_bugs_assigned_to_major	Percent
assigned_to	percent_bugs_assigned_to_critical	Percent
assigned_to	percent_bugs_assigned_to_blocker	Percent
assigned_to	percent_bugs_assigned_to_priority_changed_at_least_once_count	Percent
assigned_to	percent_bugs_assigned_to_severity_changed_at_least_once_count	Percent
assigned_to	(log) bugs_assigned_to_votes_all_actors_count	Count
assigned_to	(log) bugs_assigned_to_cc_all_actors_count	Count
assigned_to	(log) bugs_assigned_to_flags_mean	Continuous
assigned_to	(log) bugs_assigned_to_comments_all_actors_count	Count
assigned_to	percent_bugs_assigned_to_has_top_3_keyword	Percent
assigned_to	percent_bugs_assigned_to_has_top_10_keyword	Percent
assigned_to	percent_bugs_assigned_to_has_top_25_keyword	Percent
assigned_to	percent_bugs_assigned_to_has_top_50_keyword	Percent
qa_contact	percent_bugs_qa_contact_enhancement	Percent
qa_contact	percent_bugs_qa_contact_trivial	Percent
qa_contact	percent_bugs_qa_contact_minor	Percent
qa_contact	percent_bugs_qa_contact_major	Percent
qa_contact	percent_bugs_qa_contact_critical	Percent
qa_contact	percent_bugs_qa_contact_blocker	Percent
qa_contact	percent_bugs_qa_contact_priority_changed_at_least_once_count	Percent
qa_contact	percent_bugs_qa_contact_severity_changed_at_least_once_count	Percent
qa_contact	(log) bugs_qa_contact_votes_all_actors_count	Count
qa_contact	(log) bugs_qa_contact_cc_all_actors_count	Count
qa_contact	(log) bugs_qa_contact_flags_mean	Continuous

qa_contact	(log) bugs_qa_contact_comments_all_actors_count	Count
qa_contact	percent_bugs_qa_contact_has_top_3_keyword	Percent
qa_contact	percent_bugs_qa_contact_has_top_10_keyword	Percent
qa_contact	percent_bugs_qa_contact_has_top_25_keyword	Percent
qa_contact	percent_bugs_qa_contact_has_top_50_keyword	Percent

Table 112: Solution knowledge value independent variables at organisation level

At the outset, a pair of models, consisting of one control-only model and one control plus independent variable model, was created for each of the 21 dependent variables for each of the sis hypotheses, resulting in 252 models at the organisation level. For each model, the goodness of fit and significance of calculated coefficients for each variable were evaluated using OLS regression, logistic regression, beta regression, and post-fit ANCOVA with heteroscedasticity correction, as appropriate, depending on the nature of the dependent variable of each model, as discussed in the previous section. The measures used to assess the model fits and test the hypotheses were the same as those used at the individual levels of analysis.

Individual-organisation nested cross-level analysis

As discussed in the previous chapter, operationalization of the dependent and independent variables was done at separate levels of analysis: problem, individual, and organisation level. While there were significant data source and definitional distinctions between the problem level and individual level of analysis, the organisation level was largely an aggregation of individual level data according to organisation. In many cases, the aggregation from individual to organisation level resulted in isomorphic variables (c.f. Rousseau & House, 1994; Chan, 1998; Bliese, 2000). In other cases, as discussed in the previous section, the choice was made to introduce slight variances in the representation of variables at the organisation level to better localise any effects to the appropriate level while also accounting for multiple representations of similar variables in the data. A further analytical concern is the effect of the nested nature of individuals within organisations as it is possible that endogenous organisational effects may confound effects noted at the individual level. The present section discusses the additional analysis that was conducted to identify the nature of and, if applicable, degree of any cross-level influence of organisation on profile level effects.

The first step of the profile-organisation nested analysis was to create a subset of profiles that are constrained according to both the individual level constraints and the organisation level constraints. At the individual level, as discussed in the individual level analysis section, analysis was conducted according to the role in which individuals engage when participating in the knowledge production process: problem knowledge producer (reporter), solution knowledge producer (assigned_to), and solution knowledge verifier (QA_contact) roles. In order to be retained as meaningful at the individual level, profiles must have engaged in each role at least 4 times. In order to be retained as meaningful at the organisation level, organisations must have at least 3 members, not designated as a non-organisational domain name, and the organisation must have engaged in each aggregate role at least 4 times. Combining these constraints results in a sample frame of individuals and organisations as described in Table 113.

Nested role	Retained profile count	Retained organisation count
Problem knowledge producer (reporter)	3769 (17,591 without nesting)	1409 (2338 without nesting)
Solution knowledge producer (assigned_to)	1185 (2470 without nesting)	195 (206 without nesting_
Solution knowledge verifier (qa_contact)	288 (462 without nesting)	41 (42 without nesting)

Table 113: Sample frame of profiles and organisations subject to nesting constraints

While the reduction in the sample size in the sample frame is significant, it remains sizeable and sufficient for nested analysis. Given that the purpose of the nested analysis is only to determine the degree of effect of the nesting itself, not to separately assess the independent variable effects, which was already done in the profile and organisation level analyses, the lower sample size was deemed acceptable.

Modelling: Base and nested models

Nested analysis is conducted by comparing a base regression model to a nested (random effects) model based on goodness of fit (c.f. Crainiceanu & Ruppert, 2004; Scheipl, Greven, & Kuechenhoff, 2008; Lefcheck, 2015; Bates, Mächler, Bolken, & Walker, 2015). Therefore, while the goal of the nested analysis is to investigate the nested effects that may be present in the individual level of analysis, the dependent variables at the individual level of analysis must be adapted to be conducive to nested (mixed-effects / random effects) modelling in the present analysis. While at the individual level, the choice was made to use beta regression modelling for the percent variables, mixed-effects beta regression modeling has not yet been implemented in any statistical software at the time of writing. As such, an alternative representation of the percent dependent variables was constructed for the nested analysis by using the logit transformation (Davison & Hinkley, 1997), which is conducive to both linear and mixed-effects (random effects) analysis. While some of the drawbacks of the unit interval constraint of percent variables remain in the logit transformed dependent variable representation (Smithson & Verkuilen, 2006; Cribari-Neto & Zeileis, 2010; Simas, Barreto-Souza, & Rocha, 2010; Grün, Kosmidis, and Zeileis, 2012), their effect is minimized by the comparative nature of the analysis, where both base and nested models suffer the same drawbacks. Given that beta regression was used in the individual level analysis and the purpose of the nested analysis is solely to identify type I errors in the individual level analysis, the tradeoff of switching types of analysis to enable mixed-effects regression was considered acceptable. It also provides an alternate representation of the individual level analysis that allows for a more comprehensive interpretation of results, as discussed in the next chapter.

Modelling was conducted in two stages. In the first stage, a base model was created for each dependent variable (21 models, 7 per role) in a manner similar to the full models that were created at the individual level of analysis. Three major differences were introduced. First, the percent dependent variables were logit transformed, as discussed above. Second, control variables were dropped from the models because their effect was already evaluated at the individual level of analysis. Third, only the independent variables that were found to have a high probability of significance ($p < 0.001$) in the full independent level models (which included significant controls) were retained for the nested analysis models. Given that the purpose of the nested analysis is to identify type I errors in observed effects at the individual level that are the result of the nested nature of individuals within organisations, the inclusion of independent variables that were not likely to be significant would serve only to lower the power of the analysis and reduce the likelihood of identifying the type I errors. Those variables that were found to have a high probability of significance, rejecting the null hypothesis in each case, represent the only possible source of type I errors at the individual level of analysis. While an argument could be made for the inclusion of those independent variables that showed an effect with a moderate, yet notable probability of significance ($0.001 < p < 0.05$), as the goal of this study is to focus on large effects, the inclusion of only those variables with high probability of significance represents the most conservative estimate of probable type I error due to endogeneity in the data. Future research may wish to examine a more nuanced view of the cross-level nested relationship of the data, which is beyond the scope of the present study.

In the second stage, matching mixed-effects models were created by taking the base models that were created in the first stage, as described above, and adding a random effect term that captures the nested effects of the organisation of which each individual is a member, independent of the effects of the other independent variables (which represent the fixed effects), in the standard mixed-effect model form:

$$DV \sim IV_1 + IV_2 + \ldots + IV_n \mid Organisation$$

Or, more formally:

$$Y_{ij} = x_{ij}\beta + u_{ij}\gamma_i + \varepsilon_{ij}$$

$$\text{where } i = 1, \ldots, m; j = 1, \ldots, n_i$$

and:

Y_{ij} = *response of the j-th individual member of organisation i*

m = *number of organisations*

n_i = *number of individual members in organisation i*

x_{ij} = *covariate vector of j-th individual member of organisation i for fixed effects*

β = *fixed effects parameter*

u_{ij} = *covariate vector of j-th individual member of organisation i for random effects*

γ_i = *random effects parameter*

ε_{ij} = *residual standard error*

The inclusion of the random effect "organisation" term results in the localisation of organisation-specific effects in this term before the measurement of the fixed effects in the independent variables. Comparison of the mixed-effects model to the base model allows for the separation of those effects on the dependent variables that were due to the inherent nature of the independent variables in the hypotheses from those effects on the dependent variables that were endogenous to the embeddedness of individuals in organisations.

At the outset, a pair of models, consisting of one base model and one mixed-effects model, was created for each of the 21 dependent variables for each of the six hypotheses, resulting in 252 models at the individual-organisation nested level of analysis. For each model, the goodness of fit and significance of calculated coefficients for each variable were evaluated using a range of standard statistical measures discussed in the following section.

Evaluating models: Base and nested models

For the base models, OLS regression was used to evaluate the continuous and logit transformed percent dependent variable models and logistic regression was used to evaluate the logical dependent variable models. A similar range of standard statistical analysis procedures were conducted on the fitted models in a manner similar to the OLS and logistic regression modelling procedures conducted at other levels.

For the nested models, linear mixed-effects regression was used to evaluate the continuous and logit transformed percent dependent variables and generalised linear mixed-effects regression was used to evaluate the logical dependent variable models (Bates, Mächler, Bolken, & Walker, 2015). AIC and BIC were calculated and used to compare the nested models to the base models to assist with model selection (Burnham & Anderson, 2002; Wagenmakers & Farrell, 2004). Pseudo marginal (fixed-effects) R^2 and pseudo conditional (random-effects) R^2 (Lefcheck, 2015) were calculated and used to estimate Cohen's f^2 effect size of the random effects isolated from the fixed effects.

Lastly, in a manner similar to the evaluation of the individual level models, an analysis of deviance test (type II) for each variable in both base and mixed-effects models was conducted to evaluate the contribution of each variable to the overall model goodness of fit, represented by separate Chi-squared statistic values and associated p values for degree of certainty, along with degrees of freedom. The comparison of the significance of each base-mixed-effect variable pair across the two models enables interpretation of the separate contribution of the organisation embeddedness on the observed effects on the dependent variables.

The adapted dependent variables, their variable type, and the type of regression used to analyse both the associated base and mixed-effect models are described in Table 114.

Nested role dependent variables	**Variable type**	**Base regression type** (Heteroskedasticity correction in all)	**Mixed-effects regression type** (Heteroskedasticity correction in all)
At least one bug acted upon in reporter role was reopened	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)	Generalised linear mixed-effects + ANCOVA (Type II)
(Non-zero) Percent of bugs acted upon in reporter role that were reopened at least once	Logit transformed percent	Ordinary least squares (OLS) + ANCOVA (Type II)	Linear mixed-effects + ANCOVA (Type II)
At least one bug acted upon in reporter role was reassigned	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)	Generalised linear mixed-effects + ANCOVA (Type II)
(Non-zero) Percent of bugs acted upon in reporter role that were reassigned at least once	Logit transformed percent	Ordinary least squares (OLS) + ANCOVA (Type II)	Linear mixed-effects + ANCOVA (Type II)
Percent of bugs acted upon in reporter role that were fixed	Logit transformed percent	Ordinary least squares (OLS) + ANCOVA (Type II)	Linear mixed-effects + ANCOVA (Type II)
Percent of bugs acted upon in reporter role that were fixed with at least one patch	Logit transformed percent	Ordinary least squares (OLS) + ANCOVA (Type II)	Linear mixed-effects + ANCOVA (Type II)
(log) Mean resolution time of bugs acted upon in reporter role	Continuous	Ordinary least squares (OLS) + ANCOVA (Type II)	Linear mixed-effects + ANCOVA (Type II)
At least one bug acted upon in assigned_to role was reopened	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)	Generalised linear mixed-effects + ANCOVA (Type II)
(Non-zero) Percent of bugs acted upon in assigned_to role that were reopened at least once	Logit transformed percent	Ordinary least squares (OLS) + ANCOVA (Type II)	Linear mixed-effects + ANCOVA (Type II)
At least one bug acted upon in assigned_to role was reassigned	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)	Generalised linear mixed-effects + ANCOVA (Type II)
(Non-zero) Percent of bugs acted upon in assigned_to role that were reassigned at least once	Logit transformed percent	Ordinary least squares (OLS) + ANCOVA (Type II)	Linear mixed-effects + ANCOVA (Type II)
Percent of bugs acted upon in assigned_to role that were fixed	Logit transformed percent	Ordinary least squares (OLS) + ANCOVA (Type II)	Linear mixed-effects + ANCOVA (Type II)
Percent of bugs acted upon in assigned_to role that were	Logit transformed	Ordinary least squares (OLS)	Linear mixed-effects + ANCOVA (Type II)

fixed with at least one patch	percent	+ ANCOVA (Type II)	
(log) Mean resolution time of bugs acted upon in assigned_to role	Continuous	Ordinary least squares (OLS) + ANCOVA (Type II)	Linear mixed-effects + ANCOVA (Type II)
At least one bug acted upon in qa_contact role was reopened	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)	Generalised linear mixed-effects + ANCOVA (Type II)
(Non-zero) Percent of bugs acted upon in qa_contact role that were reopened at least once	Logit transformed percent	Ordinary least squares (OLS) + ANCOVA (Type II)	Linear mixed-effects + ANCOVA (Type II)
At least one bug acted upon in qa_contact role was reassigned	Logical	Logistic (GLM with logit link) + ANCOVA (Type II)	Generalised linear mixed-effects + ANCOVA (Type II)
(Non-zero) Percent of bugs acted upon in qa_contact role that were reassigned at least once	Logit transformed percent	Ordinary least squares (OLS) + ANCOVA (Type II)	Linear mixed-effects + ANCOVA (Type II)
Percent of bugs acted upon in qa_contact role that were fixed	Logit transformed percent	Ordinary least squares (OLS) + ANCOVA (Type II)	Linear mixed-effects + ANCOVA (Type II)
Percent of bugs acted upon in qa_contact role that were fixed with at least one patch	Logit transformed percent	Ordinary least squares (OLS) + ANCOVA (Type II)	Linear mixed-effects + ANCOVA (Type II)
(log) Mean resolution time of bugs acted upon in qa_contact role	Continuous	Ordinary least squares (OLS) + ANCOVA (Type II)	Linear mixed-effects + ANCOVA (Type II)

Table 114: Nested role dependent variables and base and mixed-effect chosen regression types

In summary, this chapter discussed the analytical procedures used to examine the data at the problem, individual, and organisation levels of analysis, as well as the procedures used to analyse the individual-organisation nested cross-level effects. The R code (R Foundation, 2017) that was written to implement these analyses is reproduced in Appendix B: Analysis code. The next chapter discusses the results of the analyses and hypothesis testing.

CHAPTER SIX: RESULTS AND DISCUSSION

This chapter discusses the results of the analyses to attempt to answer the research question, "*What are the factors driving successful solution knowledge emergence?*". The conceptual framework of the study formulates six hypotheses (as described in Table 2) that are tested using the methodology and analyses discussed in the previous chapters. Accordingly, this chapter is organized by hypothesis.

Hypothesis one: Absorptive capacity

The first hypothesis postulates that, "*The absorptive capacity of the meta-organisation is positively correlated with solution knowledge emergence*." This hypothesis is tested by analyzing data at the problem, individual, and organisation levels of analysis; cross-level nesting effects between individuals and the organisations of which they are a member are also assessed. The results for each level are discussed in turn in the following sections.

Problem level of analysis results

At the problem level of analysis there are seven measures that represent the conceptualization of the dependent outcome of interest, solution knowledge emergence, as depicted in Figure 11, and as operationalized into the variables described in Table 51. Six measures represent the conceptualization of the antecedent of interest in hypothesis one, absorptive capacity, as depicted in Figure 12, and as operationalized into the variables described in Table 53. The results of the analyses are depicted in the regression model output summaries in Appendix D: Regression models.

Measure one: Number of unresolved bugs at time of new problem knowledge reveal

The first measure of absorptive capacity at the problem level of analysis considers the number of "open" bugs at the time new problem knowledge is revealed to the meta-organisation. It is theorised that the more open bugs at the time a new problem is submitted, the worse the outcomes associated with the newly revealed problem.

Examination of the results described in the summary of the regression models reveals that the number of open bugs at creation time is significantly negatively correlated ($p < 0.001$) with resolution time, one of the key dependent outcome measures. This effect is the inverse of that expected according to the extant theory on absorptive capacity, which suggests that the more unresolved problems currently drawing the attention of members of the meta-organisation, the longer it would take for new problems to be resolved. A similar result is observed for the outcome measure of development time ($p < 0.001$), also contrary to expectation. This inverse result may suggest that there is something particular about how absorptive capacity limits are handled in the meta-organisation that is specific to the individuals whose absorptive capacity is affected by the number of open problems. The effect of the other outcome variables is considered in tandem with this result below in order to provide a broader context of interpretation.

The assignment time of new problems does not seem to be affected by number of open bugs. This result is as expected given the definition of "open" bugs often includes those problems that have already been assigned, so it is not surprising that it is not clearly associated with changes in absorptive capacity of those members of the meta-organisation involved in the triaging of problem knowledge. However, the impact on the knowledge creation process can be seen by the highly significant ($p < 0.001$) negative correlation with the “everconfirmed” outcome. When there are too many open bugs, bugs may slip by the confirmation process and are simply ignored until they are picked up by another actor elsewhere in the process. This result is particularly important for practice because the confirmation process is key to assessing the validity of bug reports and the proper allocation of meta-organisation resources (c.f. Matter, Kuhn, & Nierstrasz, 2009).

The number of bugs open at submission of new problem knowledge is also correlated ($p < 0.001$) with increased fix emergence tendencies, yet, negatively correlated ($p < 0.001$) with patch emergence tendencies (which is measured independently of fix emergence tendencies, as discussed in Chapter Four: Research Method). This result provides a more nuanced understanding of the different impact of open bugs on different types of solution knowledge emergence. The positive effect on fix emergence and negative effect on patch emergence suggest that the absorptive capacity of solution knowledge producers in the meta-organisation—developers who create the patch type solution knowledge that emerges to address submitted problems—differs from the absorptive capacity of other actors in the organisation. The heterogeneous absorptive capacity limits of actors result in problems being handled differently. Those problems that can be resolved quickly, which often don’t involve patches in the first place, are resolved first, and those problems that involve patches suffer from the absorptive capacity limits of developers. It may be that the meta-organisation has a process that triggers when the number of open bugs hits a certain threshold, at which point all the “easy” problems are quickly and rapidly resolved to reset the number of outstanding bugs to a more acceptable level, leading to the increase of “fixes” in periods of heavy load. The strong negative correlation ($p < 0.001$) with reassigning tendencies supports this explanation, because easy to fix bugs will not end up bounced around between multiple developers over time. Examination of the threshold timing outcome effects sheds more light on this possible explanation.

The number of open bugs at time of new problem knowledge submission is strongly positively correlated ($p < 0.001$) with extremely fast, very fast, and fast resolution times, and strongly negatively correlated ($p < 0.001$) with extremely slow resolution, slow resolution, and with average resolution times. It is also strongly positively correlated ($p < 0.001$) with extremely fast development and strongly negatively correlated ($p < 0.001$) with extremely slow, very slow, and average development. These results, in tandem with the inverse relationship for overall time to resolution, support the plausible explanation for the unexpected increased probability of bug fix described above by painting a picture of the way that too many open bugs prompt action. When a large number of unresolved bugs accumulate, they prompt action by a knowledge production process triager who focuses specifically on the task of addressing this accumulation. The triager, in a short period of time, quickly reviews a large number of open bugs that had previously gone unnoticed at the time they were initially submitted, possibly because absorptive capacity was maxed out at the time they were submitted due to work on other problems. The triager then quickly "resolves" a large number of bugs at once that can be rapidly addressed, requiring only minimal solution knowledge creation. Once the triager has quickly lowered the number of open bugs back down to an "acceptable" level, the process returns to a normal state, with the normal impact of absorptive capacity limits hindering development, until a "high" threshold of open bugs is once again hit, triggering a new round of rapid triage. When development is necessary on "quick to fix" bugs, the development is of a type that doesn't require patches, which are typically more complex in nature, supported by both the lower likelihood of patches emerging and the development timing threshold effects observed.

This explanation for the results is supported by qualitative accounts in the literature of bug processing practices by triagers in meta-organisations (c.f. Anvik, Hiew, & Murphy, 2005; D'Ambros, Lanza, & Pinzger, 2007; Breu, et al., 2010; Guo, et al., 2011; Marks, Zou, & Hassan 2011; Baysal et al., 2012ab, Khomh, Dhaliwal, Zou, & Adams, 2012; Baysal, Holmes, & Godfrey, 2013). This punctuated "cleanup" process represents an active process or routine by members of the meta-organisation to "reset" their absorptive capacity such that they can tackle new problems. Future research that compares this meta-organisational absorptive capacity management process to similar processes in traditional organisations that have been reported in the literature (c.f. Harrington & Guimaraes, 2004; Lane, Koka, & Pathak, 2006; Roberts, Galluch, Dinger, & Grover, 2012) could be fruitful in further expanding our theoretical understanding of absorptive capacity management processes.

Measure two: Number of unresolved bugs at problem knowledge reveal with same platform, operating system, classification, product, or component as new problem knowledge reveal

The second measure of absorptive capacity is the number of unresolved bugs at the time new problem knowledge is revealed to the organisation that are associated with the same platform, operating system, classification, product, or component as the new problem knowledge reveal. This second measure is distinct from the first measure in that it seeks to narrow down the degree to which absorptive capacity is specific to the major ways in which knowledge is classified in the meta-organisation: platform, operating system, classification, product, and component (c.f. Lane & Lubatkin, 1998; Zahra & George, 2002; Schmidt, 2010; Spithoven, Clarysse, & Knockaert, 2011). Whereas the first measure examines the impact of overall number of open bugs at time of new problem knowledge creation, the second measure matches the type of the new problem knowledge to open bugs of the same type to examine the specific knowledge type-based absorptive capacity effect on the outcomes of interest.

As is the case for all open bugs, the number of open bugs in the same product as the new problem knowledge reveal at the time it is submitted is strongly negatively correlated with resolution time ($p < 0.001$). However, the number of open bugs in the same platform, operating system, classification, and component at the time of a new problem knowledge reveal are strongly positively correlated with resolution time ($p < 0.001$, except platform which is $p < 0.05$). This result suggests a split in how meta-organisation members address the absorptive capacity limits, described in the results of the first measure, based on type. Whereas platforms, operating systems, and classifications describe broad knowledge types that are managed by many different individuals in the meta-organisation, products are typically managed by a single or a few individuals, and therefore more abundantly tax the absorptive capacity of one or a couple individuals associated with that product without affecting the absorptive capacity of others in the meta-organisation. It is the individuals responsible for each product that are prompted into action to reduce the excessive number of "open" bugs in their product category when the open bug threshold is reached.

This explanation is supported by the split in correlations observed between number of open bugs in each type and everconfirmed outcome. When there are a lot of open bugs in the same platform, operating system, classification, or product, the likelihood of confirmation of new problem knowledge is higher ($p < 0.001$). By contrast, there is a negative correlation ($p < 0.001$) in the case of same component. This outcome is likely the result of individuals often managing multiple components, which are rapidly added rather than changing the number of open bugs in existing components. There are over 1000 component types in the meta-organisation at the time of analysis. As such, as individuals move their focus away from older components towards newer components, more and more bugs in older components are "forgotten" and hence never confirmed.

In terms of fix emergence tendencies, platform and operating system are not correlated with outcome or patch emergence, whereas number of open bugs with the same classification as new problem knowledge is correlated with increased fix and patch emergence tendencies ($p < 0.001$) and number of open bugs in same product or component as new problem knowledge reveal are correlated ($p < 0.001$) with reduced of fix and patch emergence tendencies.

Once again, this result appears to delineate between the effects of multiple absorptive capacity thresholds in the meta-organisation. The absorptive capacity limits of actors in the meta-organisation who engage with slow changing types like platform and operating system are higher, and therefore do not affect outcome, whereas the absorptive capacity limits of actors engaged in types that tend to be managed by individuals, such as products, do negatively affect outcomes. In particular, this result supports the notion in the literature that individuals tend to exert strong control over particular features (components) and products that are of interest to them and their organisations, often rejecting the input of others (Dahlander & O'Mahony, 2011; Dahlander & Frederiksen, 2012). That behaviour may explain why new bugs are more likely to be rejected when there is a lot of activity around a product or component given that that activity may be done by a focal user who is not interested in pulling in the same direction as the new problem knowledge. The strong negative correlation ($p < 0.001$) between number of open bugs in same product as new problem knowledge and reassignment appears to support this explanation because bugs that are triaged to a well-known product developer who has a history of developing many solutions for that same product are less likely to be reassigned to a different solution knowledge developer. These results have implications for the degree to which individuals and organisations can affect the emergence of types of solution knowledge that may be controlled by others in the meta-organisation.

The difference in outcome effects across types suggests support for the theory that absorptive capacity limits are specific to knowledge typologies inherent to organisations and may not have the same effect across all knowledge types. In particular, the “product” type appears to be highly relevant in the Mozilla meta-organisation, as supported by the threshold timing effects matching the observed effects for the “all” open bugs first measure, whereas the other types do not appear to be have similar absorptive capacity limits that translate into negative outcome effects. These other type results may be attributable to the popularity of certain types of knowledge more than absorptive capacity limits. Hypothesis three: dominant knowledge paradigm, examines the popularity of knowledge types more directly to consider this possibility.

Measure three: Number of bugs at that were created within quantile-based time thresholds of new problem knowledge reveal

The third measure is the number of bugs that were created within quantile-based time thresholds of a new problem knowledge submission to the meta-organisation. This third measure is distinct from the first two measures in that it narrows down the degree to which absorptive capacity limits are time frame specific. Whereas the first measure examines overall absorptive capacity effects and the second measure examines meta-organisation specific knowledge type-based absorptive capacity effects, the present measure examines the effects of the number of bugs that were created during 1, 3, 7, 30, 90, 180, 365, and 730 day windows prior to a new problem knowledge submission to the meta-organisation.

Examination of the outcomes of the analysis reveals inconsistent results, suggesting that number of open bugs in time-based thresholds prior to a new knowledge reveal may not have a strong outcome effect at the problem level of analysis. Several results stand out: The number of bugs created in the past 7 days and in the past 365 days are strongly negatively correlated ($p < 0.001$) with resolution time, suggesting that there may be a punctuated "cleanup" process of rapidly closing easy to resolve open bugs every week and every year in the meta-organisation. This interpretation is supported by the positive correlation with "extremely fast resolution" ($p < 0.001$) threshold for both variables. Further, the 7 day window is negatively correlated with the "everconfirmed" outcome, which follows because the confirmation process is often skipped for easy to resolve bugs that don't go through the normal bug life cycle. The fifth measure examines the cyclic timing of bug reveals to consider this factor independently.

However, beyond the possible representation of "cleanup" process timing in the meta-organisation, interpreted in aggregate, the inconsistency of the results does not lend much support to the hypothesis of different time-based absorptive capacity limit effects on outcomes of interest.

Measure four: Number of bugs that were resolved within quantile-based time thresholds of new problem knowledge reveal

The fourth measure is the number of bugs that were resolved within quantile-based time thresholds of new a problem knowledge reveal to the meta-organisation. This measure considers absorptive capacity limits from the perspective of number of bugs recently resolved, rather than recently created, as is the case for the third measure. Whereas the third measure of number of open bugs represents “knowledge creation work that needs to be done” as a definition of load that taxes absorptive capacity, the fourth measure represents “knowledge creation work that was recently done” as an alternate definition to ensure that outcome effects are triangulated using multiple operationalisation approaches, as discussed in Chapter Four: Research Method.

The results suggest a momentum effect, where, contrary to the hypothesis, the more bugs that were resolved in the recent past, the lower the resolution time and development time of a newly submitted bug ($p < 0.001$). The momentum effect appears to sit between 7 and 180 days, with the strongest effect around the 90 day mark, which roughly coincides with the development life cycle used in the Mozilla meta-organisation (Mozilla, 2017a) that aims for approximately 150 day cycles, with delays often pushing cycles closer to 180 days. Within a release cycle, the number of open bugs creates momentum to resolve them faster before the next release cycle, rather than hindering the development. At the 90 day mark, the developers begin to anticipate the end of the cycle and increased prioritization of those piece of problem knowledge that require solutions that are related to the priorities of the present release cycle. The timing threshold effects support this interpretation with a bi-model result showing that large number of bugs resolved in the past 90 days is positively correlated with both extremely fast ($p < 0.05$) and extremely slow ($p < 0.001$) resolution, and negatively correlated with average resolution ($p < 0.001$), because bugs that are related to the priorities of the present release cycle are immediately dealt with in order to ensure the release cycle’s deadlines are met while bugs that are not related to the release cycle’s priorities are ignored, and forgotten, often for long periods of time.

This result highlights the importance of aligning new problem knowledge reveals with the release cycle priorities of the meta-organisation, lest they be ignored and forgotten. It also refines extant absorptive capacity theory by showing that organisations can use release cycle processes to focus effort and mitigate the impact of individual absorptive capacity limits by creating knowledge production momentum guided by shared priorities.

Measure five: Calendar timing of new problem knowledge reveal

The fifth measure of absorptive capacity is the calendar timing of new problem knowledge reveals to the meta-organisation. Whereas the previous measures consider the impact of overall number of open bugs, types of number open bugs, time-based windows of number of open bugs, and time-based windows of resolved bugs, this fifth measure considers the calendar timing of a new problem submissions. The results of measure four suggest that cycles within the meta-organisation may be related to the absorptive capacity of its members and affect the outcomes of interest. The fifth measure considers the impact of the weekday, the day of the month, the month, and the year of new problem knowledge reveals on the dependent outcomes of interest. These calendar timings represent social cycles that often impact organisations such as weekends or holiday periods. The absorptive capacity of meta-organisation participants may be lower around a holiday period or higher during a focused period of the year.

Examination of the output of the ANCOVA summaries of the regression models reveals that weekday, month, and year are all positively correlated with resolution time ($p < 0.001$), assignment time ($p < 0.001$ except month which was $p < 0.05$), confirmation ($p < 0.001$), and most time-based resolution, assignment, and development thresholds. These results suggest that social cycles have a strong impact on the knowledge production process, as theorized. Whereas year, month, and weekday show broad effects, month day is only correlated with resolution time, assignment time, and development time, and associated timing thresholds. These results suggest that there are differences in yearly, monthly, and weekly cycles.

Yearly cycles tend to relate to overall long-term goal prioritization of the meta-organisation, as laid out by the Mozilla Manifesto (Mozilla, 2017b). These cycles shift based on the emerging social and technical factors on the Internet and the stated priorities of the meta-organisation (Gulati, Puranam, & Tushman, 2012).

Monthly cycles relate to the development process (Mozilla, 2017a) discussed earlier, but also relate to social cycles such as holidays. Examination of the dummy regression model summaries reveals that October, November, December, January, and February have the strongest correlation with resolution timing, centered around the lead up to the winter holiday periods observed in Western Europe and America, the location of most Mozilla meta-organisation participants. However, the month is not correlated with fix or patch emergence tendencies, suggesting that the primary effect on the knowledge production process is time based.

Weekly processes align with the concept of the work week. Many of the participants in the Mozilla meta-organisation engage in the knowledge development process during the standard Monday-to-Friday work week. Therefore, it is unsurprising that the dummy regression model summaries show that new problem knowledge submissions on any weekday other than Friday (the reference category) are correlated with lower overall resolution time. An easy takeaway for organisations is that submitting new problem knowledge on a Friday delays solution knowledge emergence. Problems submitted on a Monday are much more likely to be confirmed ($p < 0.01$) and enter into bug life cycle process. However, as with months, weekdays are not correlated with fix or patch emergence tendencies, suggesting that the primary impact of weekday on the knowledge production process is time based.

Measure six: Amount of time between problem knowledge reveal and resolution

The sixth measure considers the effect of the amount of time that each bug remains open on the dependent outcomes of interest. In this context, the "resolution time" variable is treated as an independent variable rather than a dependent variable, as is the case with the previous measures. As a result, certain dependent variables that are correlated to resolution time by definition are not considered for this measure, as discussed in Chapter Four: Research Method.

Examination of the summaries of the regression model fit assessments suggests that resolution time is strongly negatively correlated with fix and patch emergence tendencies ($p < 0.001$). This result is as theorized. The longer a bug has been open, the more likely that it becomes irrelevant to the priorities of the meta-organisation. When it is finally resolved, therefore, it is more likely to be resolved as "not fixed". This result is consistent with the timing effect results observed with the previous measures and extends those results to clearly show that if the timing of new problem knowledge does not fit with the social and organisational cycles of the meta-organisation, solution knowledge emergence diminishes. This cycle prioritization explanation is also supported by the significant negative correlation with the bug being reopened ($p < 0.001$). If the bug were resolvable within the priorities of the meta organisation and some other factor were responsible for the observed effect on fix emergence tendencies, reopening would be expected to take place at the same rate regardless of timing. However, bugs that are deemed irrelevant, "out of date", don't end up reopened because the verification of a given solution doesn't take place—the solution is deemed irrelevant before it is even created, during the triage process. Hypothesis six: Solution knowledge value examines this factor independently from timing effects to provide a more robust picture of the overall effect.

This result is important for theory because it suggests that much more than bug characteristics alone is responsible for the eventual outcome resolution of the bug (Panjer, 2007; Marks, Zou, & Hassan, 2011). It is also important for practice as it suggests that organisations that seek to improve solution knowledge emergence must be aware of the timing cycles within the meta-organisation as an antecedent for success that is independent of the properties of the submitted problem knowledge itself.

Summary of dependent variable effect results at problem level of analysis

Comparison of the control and full regression models for the effect of absorptive capacity measures on the dependent outcomes of interest reveals an overall picture of effect size at the problem level of analysis. Whereas all the models are highly significant as assessed by the F statistic, Chi-squared statistic and/or AIC, as appropriate, given the very large number of observations in the database, the additive effect size provides a hypothesis-specific picture of the influence of the independent variables on the dependent outcomes of interest (Cohen, 1988, 1992; Sawilowsky, 2009).

The additive effect size of the independent variables above and beyond the control variables on the outcome of resolution time is “small” (Cohen, 1988, 1992; Sawilowsky, 2009) despite an overall R^2 of 0.221. The majority of that additive effect can be seen to be attributable to the development time effect size, which is “small-to-medium”, with an incremental R^2 of 0.280 over the control-only R^2 of 0.215. Likewise, examination of the additive effect sizes of the full models for the development time threshold effects reveals sizeable effects of the absorptive capacity independent variables on development time thresholds, particularly the "extremely slow development" threshold which has a "very large" effect size with an incremental pseudo-R^2 of 0.240 to 0.562. Numerous other development timing thresholds have "small-to-medium" effect levels with typical incremental pseudo-R^2 values around 0.06. These results lend strength to the notion that the absorptive capacity effects at the problem level primarily influence development time, as hypothesized, and consistent with the extant theory in the open source literature (Panjer, 2007; Marks, Zou, & Hassan, 2011).

Likewise, the comparative effect size of the fix and patch emergence tendencies are "small-to-medium" and "medium" with pseudo- R^2 increases of 0.550 to 0.584 and 0.406 to 0.470 respectively. Given the very high degrees of freedom, this result suggests very strong overall effects of absorptive capacity independent variables on fix and patch emergence tendencies.

By contrast, reopening tendencies are affected very little by the absorptive capacity independent variables, with an incremental R^2 of only 0.008, resulting in a Cohen's incremental effect size of "very small". Reopening tendencies do not appear to be affected strongly by absorptive capacity factors, which, in itself, is an interesting contribution to the theory of the bug life cycle as the causes of bug reopening are still not well understood (Zimmermann, Nagappan, & Guo, 2012; Shihab, et al., 2010, 2013).

In summary, while a range of problem level absorptive capacity effects are observed in the results, the largest effect appears to be related to development time. In particular, alignment of problem knowledge reveals to meta-organisational priorities and organisational and social cycle timings appears to be the absorptive capacity factor that most heavily affects development time and hence overall resolution time. Organisations wishing to reduce solution knowledge emergence time should take these factors into account.

Individual level of analysis results

At the individual level of analysis there are seven measures that represent the conceptualization of the dependent outcome of interest, solution knowledge emergence, as depicted in Figure 19, and as operationalized into the variables described in Table 79 and Table 87. Five measures represent the conceptualization of the antecedent of interest in hypothesis one, absorptive capacity, as depicted in Figure 20, and as operationalized into the variables described in Table 89. The results of the analyses are depicted in the regression model output summaries in Appendix D: Regression models.

As discussed in Chapter Four: Research Method, regression models are separated according to the roles in which individuals engage when participating in the knowledge production process: problem knowledge producer (reporter), solution knowledge producer (assigned_to), and solution knowledge verifier (QA_contact).

Measure one: Number of activities performed

The first measure of absorptive capacity at the individual level is the number of activities each individual has performed in each of the roles in which individuals engage when participating in the knowledge production process of the meta-organisation. It is hypothesized that the more activities in which individuals engage the worse the solution knowledge emergence tendencies for problems on which they act due to their lower absorptive capacity.

Examination of the regression model summaries reveals a strong positive correlation ($p < 0.001$) between number of activities performed by individuals engaging in the problem knowledge producer role and fix and patch emergence tendencies. This result is the opposite of hypothesized. Rather than large number of activities hindering solution knowledge emergence, instead the results suggest that individuals who engage in more activities are rewarded with better solution knowledge emergence. This result is useful for absorptive capacity theory refinement as it illustrates an experience effect in response to higher load rather than an impairment effect, challenging conventional perspectives (c.f. Cohen & Levinthal, 1990; Lane & Lubatkin, 1998), and lending support to emerging arguments that absorptive capacity effects on outcomes in certain organisational contexts, particularly involving innovation (c.f. Nieto & Quevedo, 2005; Chen, Lin, & Chang, 2009) may be different. Individuals who engage in more activities learn through an experience effect how to engage in the activities better such that outcomes improve so long as their activities don't become too taxing in number or in nature to result in negative outcomes.

The strong positive correlation ($p < 0.001$) with fix and patch emergence tendencies also holds for individuals engaging in the solution knowledge producer role, suggesting that similar experience effects apply to the developer role. The more activities developers perform, the better the fix and patch emergence tendencies for the problems with which they engage in the knowledge development process. However, in the case of the solution knowledge producer role, there is also a significant ($p < 0.05$) negative correlation with resolution time, which fits the notion that absorptive capacity thresholds are different for developers as this timing effect is not seen in the results for the problem knowledge producer role. These mixed results suggest that the absorptive capacity limits primarily affect development time, not overall resolution time, which matches the results that are observed at the problem level of analysis that are discussed in the previous section.

Measure two: Number of activities performed for each platform, operating system, and classification

The second measure of absorptive capacity at the individual level is the number of activities each individual has performed in each of the roles in which individuals engage when participating in the knowledge production process of the meta-organisation, separated into the three major types of knowledge used to categorized problems: platform, operating system, and classification. Like its counterpart at the problem level of analysis, the purpose of this second measure is to localize absorptive capacity limit effects to the knowledge categories used in the meta-organisation.

Examination of the regression model output tables reveals significant differences in the effects of activities of individuals engaging in the problem knowledge producer role acting on problems with different platforms and operating system types on resolution time. By breaking overall activities down by knowledge category, the results show that the net effect of all activities by individuals on resolution time is cancelled out by activities on different categories of problem knowledge having opposite effects. Activities on older platforms, such as x86 and PowerPC, are strongly correlated ($p < 0.001$) with increased resolution time. By contrast, activities on newer platforms, such as x86_64 are strongly correlated ($p < 0.001$) with reduced resolution time. Similarly, activities on popular operating systems such as Windows, iOS, and Android are strongly correlated ($p < 0.001$) with reduced resolution time, and activities on less popular operating systems show the inverse effect. This result matches the result observed at the problem level of analysis, suggesting that the priorities of the meta-organisation change over time and actors wishing to increase the speed of resolution of problems must align the type of problem knowledge submitted to the current focus of the meta-organisation, lest it be ignored.

The platform type-based separation of measure two also provides results that refine those observed with the first measure, demonstrating that while activities overall may increase fix and patch emergence tendencies, this effect is not consistent across types. Instead, there is a strong negative correlation ($p < 0.001$) between activities in the most popular type, x86_64 and fix and patch emergence tendencies. This result suggests that while experience effects and alignment with organisational cycles may help bugs getting resolved, there may also be a "flooding" effect where the absorptive capacity limits of developers cannot address all the needs at the same time.

Alignment with popularity cycles may, in fact, be a quadratic effect, where there is an optimal amount of popularity and priority alignment and too much or too little leads to either tapping of absorptive capacity limits, or the meta-organisation ignoring the new problem knowledge reveal. The quadratic effect is also apparent in the results for operating system type and classification, with activities on the most popular types correlated with a lower fix and patch emergence tendencies ($p < 0.001$) despite lack of overall effect. This quadratic effect explanation is also supported by the results for the solution knowledge producer role, which reveals a split based on type for positive or negative correlation ($p < 0.001$) with fix and patch emergence tendencies for platform, operating system, and classification knowledge type categorizations.

Measure three: Number of activities performed for each severity level

The third measure is the number of activities performed on problems of each severity level by individuals engaging in each of the roles. This third measure complements the previous measures by separating the effects of activities in which individuals engage in the knowledge production process by severity of the problem acted upon, which is commonly reported in the literature as a relevant factor (c.f. Ahmed & Gokhale, 2009; Bougie et al., 2010; Giger, Pinzger, & Gall, 2010; Guo, et al., 2011; Zhang, et al., 2012). Severity represents another means of knowledge development process prioritization in the meta-organisation, with problems of different severity levels potentially having different individuals with different absorptive capacity limits acting upon them as well as different levels of draw on the absorptive capacity limits of the individuals acting upon them, and hence different effects on the outcomes of interest. The definition of each severity level used appears in Table 7.

Examination of the regression model summary tables reveals further support for the notion that absorptive capacity effects on outcomes at the individual level, as is the case at the problem level, are stratified based on meta-organisation priorities. Activities on low severity problems are strongly correlated ($p < 0.001$) with increased resolution time and decreased fix emergence tendencies. By contrast, activities on problems of average severity are correlated with a reduction in resolution time ($p < 0.001$). The takeaway for actors wishing to improve the resolution time of the problems they submit to the meta-organisation by acting more frequently is that the activities must be on problems that the meta-organisation deems important enough.

The case of high severity level problems is a special case as it often represents a level of complexity that results in delays in resolution by virtue of the complexity alone, independent of the individual activity action. The setting of problems to the high severity level is usually reserved for such complex cases in the meta-organisation (Hooimeijer & Weimer, 2007). Therefore, it is unsurprising that activity on high severity level problems is correlated ($p < 0.05$)

with higher resolution time. This higher resolution time can be seen to be related to the development portion of the knowledge creation process as the degree of certainty of association is much higher ($p < 0.001$) for the solution knowledge producer role, suggesting a larger relative effect for the measure for the developer role.

Measure four: Number of activities performed of each activity type

The fourth measure is the number of activities of activity performed of each activity type by each individual acting in each role. This measure complements the previous measures by separating the effects of engaging in activities according to the type of activity. There are 15 activity types in which individuals can engage, as described in Table 13.

Examination of the regression model summary tables reveals that activities by problem knowledge producers that result in additional knowledge being appended to the problem knowledge they submit to the meta-organisation are correlated with reduced resolution times for those problems. More specifically, keyword and flag setting activities are negatively correlated ($p < 0.001$) with resolution time. The keyword setting activity is also correlated ($p < 0.01$) with increased patch emergence tendencies and the flag setting activity is correlated ($p < 0.001$) with both increased fix and patch emergence tendencies. By contrast, the number of product change activities is correlated with ($p < 0.001$) with longer resolution times as a result of the confusion that product changes introduce after the initial problem knowledge submission.

The clear takeaway for individual actors engaging in the problem knowledge producer role is that activities must be useful to the knowledge development process, promoting new knowledge creation. Activities that promote new knowledge creation are correlated with positive outcomes whereas activities that hinder knowledge creation are correlated with negative outcomes. This effect resides at the individual level of analysis. As such, organisations should ensure that organisation level processes take into account the individual level roles in which the organisation's members engage in order to promote outcomes that are aligned with the organisation's priorities. This split may shed light on the contradiction in the literature that certain activities do not result in taxing of absorptive capacity. It may be the nature of the activities themselves rather than the number of activities that leads to positive solution knowledge emergence outcomes.

For individuals engaging in the solution knowledge producer role, the results paint a similar picture based on the different knowledge needs of developers. Developers align their effort to the priorities of the meta-organisation, as seen in the problem level results. The present results highlight this effect with the number of activities related to whiteboard and priority changes negatively correlated ($p < 0.001$) with fix emergence tendencies. When whiteboards and priorities change, existing bugs are often relegated to irrelevance. Often this happens when a cycle deadline approaches and goals must be re-evaluated and some items dropped for the cycle. Developers who engage in these activities of changing the whiteboard and priorities of the bugs unsurprisingly then immediately go and "close" the bugs upon which they are acting as developers that are now out of scope, resolving them as "WONTFIX". Activities for the solution knowledge producer role represent reached limits that prompt changes in the meta-organisation, as observed in the problem level results. Those developers who wish to increase fix emergence tendencies must do so within the cycles of the meta-organisation to be successful.

Measure five: Number of times acting in each role in which individuals engage

The fifth measure is the number of times acting in each role in which individuals engage. This measure complements the previous measures by examining the effect of role engagement directly. This measure represents an alternative view of the concept of “activity” as engagement in role is represented differently in the data, by number of problems submitted, number of problems for which solutions are built, and number of problems for which solutions are verified, rather than by the activity types discussed in the previous measures. The roles in which individuals act upon problems are associated with the problems themselves, independent of the activities in which individuals engage in the meta-organisation overall, which were considered in the previous four measures, and as discussed in Chapter Four: Research Method. The role engagement measure’s effects are so significant that the measure is included it as a control variable in the regression models that assess most of the other measures.

Examination of the regression model summary tables reveals that the number of sets of problem knowledge individuals reveal is strongly correlated ($p < 0.001$) with increased resolution time, as hypothesized. The greater the number of new sets of problem knowledge an individual submits, the longer the resolution time. This correlation is observed in the results for all three roles, suggesting that role engagement, contrary to the other activities of individuals, hits absorptive capacity limits and results in negative outcomes. The more solutions an individual is creating and the more solutions an individual is verifying, the longer the resolution time for each problem set. These results support the traditional absorptive capacity limits reported in the literature.

Yet, the results also paint a broader picture of the effects of role engagement by describing some positive effects as well. Whereas resolution time is increased the more individuals submit new sets of problem knowledge, there is a correlation ($p < 0.001$) with reduced bug reopening and reassigning tendencies and increased fix and patch emergence tendencies, suggesting that individuals can improve their proficiency in desired outcomes by engaging more actively in the problem knowledge producer role. This result supports the perspective that individuals may have multiple types of absorptive capacity, with some types being taxed more easily than others. Learning can still take place surrounding some types of knowledge even as other types of learning are hindered for other types of knowledge, resulting in both positive and negative outcomes.

The picture is different when it comes to the solution knowledge producer role. As with the problem knowledge producer role, increased involvement in the solution knowledge producer role is correlated with ($p < 0.001$) with reduced reopening and reassigning tendencies. Yet the absorptive capacity limits do not only negatively affect resolution time; they also are strongly correlated ($p < 0.001$) with reduced fix and patch emergence. This different absorptive capacity for individuals based on role matches the results observed with the previous measures and makes sense given that the time investment involved in problem knowledge creation is often much lower than the time investment involved in solution knowledge creation. Individuals who are assigned to too many problems may therefore choose to "not fix" more of them and focus on those that they can manage within their available time. This result suggests that individuals who submit problem knowledge to a meta-organisation need to take into account the absorptive capacity of individuals engaging in other roles in the meta-organisation if they wish to improve solution knowledge emergence.

Effects of individual nestedness in organisations

Examination of the mixed-effects regression model summaries reveals that there are significant organisational influences on the profile level outcome effects observed in some cases. The AIC and BIC delta statistics reveal superior models with organisational effects on the time to resolution ("large"), percentage of fix ("medium-to-large"), and percentage of patch emergence ("medium") outcome variables. However, despite these cross-level effects, nearly all the observed significant relationships at the individual level remain significant when isolated from organisational effects.

The lone exception is related specifically to the platform activity type. The results at the individual level suggested that the prioritization of activities on certain platforms by individuals affects time to resolution. The mixed-effects model results suggest that platform-related activity prioritization may, instead, lie at the organisation level, rather than the individual level. Organisational priorities in how they engage with meta-organisations often revolve around the platforms that the organisations themselves use, which are typically not decided by specific individuals. As such, in context, individuals may be forced to align their activities with their organisation and, therefore, cannot influence change in this factor directly to improve outcomes of interest.

Summary of dependent variable effect results at individual level

Comparison of the control and full regression models for the effect of absorptive capacity independent variables on the dependent outcomes of interest reveals an overall picture of the effect size at the individual level of analysis. For the problem knowledge producer role, in addition to the model Chi-squared statistics and comparative AIC delta statistic models all showing significant superiority of the full regression models over the control-only models, the additive effect sizes are also sizeable for all models. For resolution time, the additive effects of the absorptive capacity independent variables above and beyond the control variables is "medium-to-large", with an increase of the R^2 from 0.074 to 0.24. The additive effect size of reopening and reassigning percentage models is "very large". The additive effect size of fix and patch emergence tendencies is "medium" with incremental pseudo-R^2 values of 0.243 to 0.331 and 0.214 to 0.339 respectively.

The additive effect sizes are similarly strong for the solution knowledge producer role: "very large" for resolution time and percentage of patch emergence, with incremental pseudo-R^2 values of 0.148 to 0.432 and 0.002 to 0.541 respectively; and, "large" for percentage of fix emergence, with incremental pseudo-R^2 value of 0.09 to 0.349.

The additive effect sizes in the solution knowledge verifier role are primarily attributable to the strong correlation observed between the role engagement variable and the outcomes of interest. The limited AIC delta statistic values suggest that while the models are overall significant, the other variables do not contribute as strongly to the overall effect size of the absorptive capacity independent variables. This result is expected given the lower number of observations inherent to this infrequent role.

In summary, the results at the individual level reveal role-specific absorptive capacity limits that have different effects on the different outcomes of interest. Individuals engaging in the problem knowledge producer role delay resolutions by submitting too many sets of problem knowledge but learn through this process to improve fix and patch emergence tendencies. Alignment of activities with the priorities of the meta-organisation and the organisation in which the individual is nested affect outcomes as well, but platform type effects may be organisation level factors, not individual level factors. Individuals engaging in the solution knowledge producer role can become too taxed by taking on too many problems and, as result may see both increased resolution length and decreased fix and patch emergence tendencies, likely due to the need to discard those problems upon which the individual cannot focus that are beyond the individual's absorptive capacity. Activity type alignment in the meta-organisation has a similar effect as in the reporter role.

Overall, the individual level absorptive capacity effects can be said to be a balance between the negative effects for types of activities that hit absorptive capacity limits and the positive effects of types of activities that do hit absorptive capacity limits and permit learning through knowledge absorption by individuals, resulting in different effects on outcomes of interest, in a manner similar to, but distinct from, the problem level of analysis. Hypothesis two: Codifiability more directly examines those observed effects that were likely attributable to the additional knowledge created by individuals through certain activities but not others. Hypothesis three: Dominant knowledge paradigm more directly examines those observed effects that were likely attributable to the popularity of certain platforms, operating systems, and classifications in the meta-organisation. Hypothesis six: Solution knowledge value more directly examines those observed effects that were likely attributable to activities associated with severity of problems acted upon by individuals and their alignment to meta-organisational priorities.

Organisation level of analysis results

At the organisation level of analysis there are seven measures that represent the conceptualization of the dependent outcome of interest, solution knowledge emergence, as depicted in Figure 26 and as operationalized into the variables described in Table 105. Five measures represent the conceptualization of the antecedent of interest in hypothesis one, absorptive capacity, as depicted in Figure 27, and as operationalized into the variables described in Table 107. The results of the analyses are depicted in the regression model output summaries in the section titled Appendix D: Regression models.

As discussed in Chapter Four: Research Method, regression models are separated according to the aggregate roles in which members of organisations engage when participating in the knowledge production process: aggregate problem knowledge producer (reporter) role, aggregate solution knowledge producer (assigned_to) role, and aggregate solution knowledge verifier (QA_contact) role.

Given that the organisation level variables are deliberately operationalised to be similar to the individual level variables, except aggregated to the organisation level, examination of the results focuses primarily on localising any effects that may take place either at both individual and organisation levels or only at organisation level.

Measure one: Number of activities performed

The first measure of absorptive capacity at the organisation level is the number of activities each organisation has performed in each of the aggregate roles in which organisational members engage when participating in the knowledge production process of the meta-organisation. Similar to at the individual level of analysis, it is hypothesized that the more activities in which organisations engage the lower the likelihood of the emergence of solution knowledge addressing the problems knowledge with which they are engaging due to the lower available absorptive capacity.

Examination of the regression model summaries reveals a positive correlation ($p < 0.05$) between number of activities performed by organisational members engaging in the aggregate problem knowledge producer role and fix emergence tendencies. This result matches the result observed at the individual level and is also in the opposite direction as hypothesized. As discussed at the individual level, these results either mean that absorptive capacity limits are not reached when organisation members engage in activities, or that the nature of the activities does not hinder knowledge flow. Further, the results suggest that there is an experience effect present, with greater activity engagement being associated with better fix outcomes for the aggregate problem knowledge producer role. However, unlike at the individual level, there is no correlation in the results between activity and patch emergence tendencies. This difference suggests a split between effects of activities at individual and organisation levels, with the former having a greater contribution to the effect on fix and patch emergence tendencies than the latter.

Contrary to the results observed at the individual level, for the aggregate solution knowledge producer (developer) role, at the organisation level, there is a negative correlation between activities and fix ($p < 0.01$) and patch emergence tendencies ($p < 0.001$). Examination of these results in combination with the individual level results and the mixed-effects results suggests that there is a small organisation level effect pulling in the opposite direction of the individual level effect—an effect that is small enough to not override the individual level effect—as seen in the only slight reduction of coefficients in the mixed-effects models as compared to the OLS models. While the organisation level results suggest a very strong effect, in the context of the highly constrained sample at the organisation level ($n = 205$ vs. 2453 at individual level), the results are better interpreted as some organisations having different absorptive capacity limits than other organisations when it comes to abilities, which fits with extant absorptive capacity theory. This result validates the importance of examining absorptive capacity at multiple levels of analysis to properly gauge its effect.

A lack of observed effect of activities on resolution time for any role at the organisation level suggests that time to resolution is primarily affected by individual level absorptive capacity limits rather than organisational limits. Given that these effects are strongest for the solution knowledge producer role and development is often done individually, this localization at the individual level fits with the literature's present understanding of the knowledge production process of meta-organisations.

Measure two: Number of activities performed for each platform, operating system, and classification

The second measure of absorptive capacity at the organisation level is the number of activities each organisation has performed in each of the aggregate roles in which organisation members engage when participating in the knowledge production process of the meta-organisation, separated into the three major knowledge types into which problems are categorized: platform, operating system, and classification. Like its counterparts at the problem and individual levels of analysis, the purpose of this second measure is to localize absorptive capacity limit effects to types of activities.

The results suggest that platform type of activity has very little effect on outcomes in the problem knowledge producer role, with significant correlation with fix ($p < 0.001$) and patch ($p < 0.05$) emergence tendencies observed only for the platform "all", which is, in fact, a category that is defined as "not platform specific". Given that platform must be specified for all problems, the platform "all" is the catch-all for problems that are not platform-specific. By contrast, whereas no effect is observed on resolution time for activities overall in measure one, similar to at the individual level of analysis, the platform effects observed in the results of measure two balance each other out amongst types, with activities on the platform “PowerPC” corelated with increased resolution time ($p < 0.01$) and activities on platforms “x86_64” ($p < 0.01$) and “other” ($p < 0.001$) correlated with decreased resolution time. As was discussed at the individual level of analysis, the popularity of the platforms likely accounts for this result, which is considered separately in hypothesis three.

The results are similar for the operating system type and classification type as well as for solution knowledge producer and solution knowledge verifier roles: The effect of activity types on outcomes of interest is more likely attributable to the popularity of the different types in the categories than to the different absorptive capacity limits of the types themselves. There is only moderate support for the notion that there are separate organisation level type-based activity effects on outcomes of interest in any aggregate role.

Measure three: Number of activities performed for each severity level

The third measure is the number of activities performed by each organisation on problems of different severity levels. It complements the previous measures by separating the effects of activities in which organisation members engage in the knowledge production process by the severity level of the problems with which they are engaging.

Examination of the results suggests that the effect of activities on problems of differing severity levels is correlated ($p < 0.001$) with increased resolution time in the case of low severity for the problem knowledge producer role, as was observed in the results at the individual level of analysis. This result further supports the notion that alignment of activities, including those activities guided at the organisation level, with priorities of the meta-organisation, affects the resolution time of submitted problems. The correlation is also apparent in the aggregate solution knowledge producer role ($p < 0.01$), as was the case in the individual level equivalent. In other words, too much organisational activity on problems classified as "low severity" in the meta-organisation slows down the resolution times of problems submitted by organisations.

Measure four: Number of activities performed according to type of activities

The fourth measure is the number of activities performed separated according to the type of activities. It complements the previous measures by separating the effects of engaging in activities according to the type of the activities in which organisation members engage in the knowledge production process. There are 15 types of activities in which organisation members can engage, as described in Table 13.

Examination of the results suggests that, as seen in the results at the individual level of analysis, certain activities are correlated with positive outcomes and other activities are correlated with negative outcomes. The common thread between activities that are correlated with positive outcomes is that they tend to increase the amount of knowledge that is associated with a given set of problem knowledge. Activities changing flags are correlated ($p < 0.001$) with reduction in resolution time and increased patch emergence tendencies ($p < 0.001$) for both the aggregate problem knowledge producer and the aggregate solution knowledge producer roles. Activities changing keywords are correlated ($p < 0.001$) with increased patch emergence tendencies for the aggregate problem knowledge producer role.

By contrast, activities that confuse the knowledge surrounding the problem with which actors are engaging are correlated with negative outcomes. Changes in product after initial problem knowledge submission are correlated ($p < 0.01$) with increased resolution time. Reopening activities, which take place when a given solution was found to not address a submitted problem, often because there was insufficient problem knowledge submitted, are correlated ($p < 0.01$) with reduced fix emergence tendencies for the aggregate problem knowledge producer role. In the aggregate solution knowledge verifier role, priority change activities are strongly correlated ($p < 0.001$) with reduced fix and patch emergence tendencies, further reflecting the notion that alignment with meta-organisational priorities is important for solution knowledge emergence.

Measure five: Number of times acting in each aggregate role in which organisation members engage

The fifth measure is the number of times acting in each aggregate role in which organisation members engage. As discussed in the individual level results section, the number of times engaging in each aggregate role has such a strong effect that it is included as a control variable for most of the regression models that evaluate the other measures.

Examination of the results suggests that number of times acting in each aggregate role at the organisation level has similar outcome effects as the counterpart roles at the individual level. For the aggregate problem knowledge producer role, role engagement is positively correlated ($p < 0.001$) with increased resolution time, delaying outcomes, negatively correlated ($p < 0.001$) with reopening and reassigning tendencies, smoothing the transition through the knowledge production process, and positively correlated ($p < 0.001$) with fix and patch emergence tendencies. For the aggregate solution knowledge producer role, role engagement is correlated with reduced fix ($p < 0.01$) and patch emergence ($p < 0.001$) tendencies.

These results, along with the individual level and mixed-effects results, suggest that there are independent organisation and individual absorptive capacity limits for the different roles that affect outcomes. Engagement in the aggregate problem knowledge producer role results in increased outcome resolution time, suggesting that absorptive capacity limits of role engagement hinder time-based outcome factors. Yet, simultaneously, the increased engagement is correlated with positive knowledge production cycle, fix, and patch outcomes, suggesting that those outcomes do not suffer as much from the absorptive capacity impairments that result from increased role engagement. Alternatively, it may be that there are multiple absorptive capacity limits with differing effects on time and fix and patch outcomes.

For the aggregate solution knowledge producer role, role engagement is negatively correlated with fix and patch emergence tendencies, consistent with the individual level results. These results suggest that organisations have development role absorptive capacity limits above and beyond individual development role absorptive capacity limits in the case of role engagement. This result further highlights the differences between role engagement and the activity counts that are considered in the previous measure, suggesting that role engagement is far more taxing on absorptive capacity than activities performed. Given that activities are individual self-contained tasks that are measured directly in the database and role engagement involves an entire knowledge production process from submission to resolution, it stands to reason that the latter would result in greater absorptive capacity impairment. This result also lends support to the notion that the observed positive effects associated with more activities in the previous measure are in fact due to absorptive capacity of organisation members not being taxed sufficiently to lead to knowledge absorption impairment. Rather, such impairment is clear in the aggregate development role engagement absorptive capacity limits and associated negative outcomes for the present measure.

These results illustrate a concept I refer to as "knowledge absorption impairment", which refers to a reduction in the ability of individuals and organisations to further absorb and apply useful knowledge to knowledge production activities. Role engagement and activities are knowledge absorption impairing factors with different degrees of impairment. They can be thought of as moderators of knowledge absorption and application. Role engagement more strongly moderates (impairs) knowledge absorption than activities. These factors would be ideal candidates for future research that isolated them and modeled their moderation effects as regression variable interactions to determine the degree to which they moderate absorptive capacity's effects on outcomes of interest.

Summary of dependent variable effect results at organisation level

Comparison of the control and full regression models for the effect of absorptive capacity independent variables on the dependent outcomes of interest reveals an overall picture of effect size at the organisation level of analysis. For the aggregate problem knowledge producer role, the Chi-squared statistic and comparative AIC delta statistic reveal that the full models are superior to the control-only models for all dependent variables. For resolution time, the additive effect size of the full model above and beyond the control model is "medium", with R^2 value increasing from 0.085 to 0.202. For fix and patch emergence tendencies, the additive effect sizes are "medium" and "medium-to-large" with R^2 values increasing from 0.170 to 0.301 and 0.169 to 0.318 respectively.

For reopening and reassigning tendencies, the additive effect size is “very large” in both cases, with incremental R^2 values of 0.239 to 0.479 and 0.280 to 0.612 respectively. These effect results are to be interpreted in the context of the constrained sample size in the case of reopening and reassigning tendencies (n = 1317 & n = 735 vs. n = 2337 for other models). Further, as discussed in the measures results sections, a select number of types have a disproportionate influence on the outcomes observed and some of these types are best understood as “catch all” types, representing significant “lack of effect on types”. These “catch all” types affect reopening and reassigning tendencies more than the other outcome measures. Therefore, it follows that the effect size results would be disproportionately inflated and should be interpreted with that in mind. While there is undoubtedly an additive effect of the absorptive capacity independent variables on the reopening and reassigning dependent variables, it is not nearly as large as the effect size statistics suggest when examined in isolation of these broader contextual factors.

For the aggregate solution knowledge producer role, the results show an additive effect size of “very large” for the full regression models except for reopening tendencies, which has an inferior full model above and beyond the control model (AIC delta statistic <= 0). Once again, given the lower sample size (n = 205), these effect size statistics are to be interpreted within the broader context of certain types affecting the outcomes disproportionately, artificially inflating the effect size statistic. Whereas there is undoubtedly an absorptive capacity independent variable effect observed in the regression models for aggregate solution knowledge producer role, it is most prominent in fix emergence tendencies where the incremental R^2 values of 0.117 to 0.483 is plausible given the number of significant coefficients observed in the regression model. The other “very large” effect sizes should be interpreted as inflated, resulting from the low sample size, given the much lower number of significant coefficients observed in the regression models and the insufficient AIC delta values to account for the very large incremental R^2 values.

For the aggregate solution knowledge verifier role, the results suggest that the low sample size (n <= 42) produces models that are untenable in many cases. Despite the Chi-squared statistic suggesting significance, the lack of significant coefficients in the resolution time model suggests spuriousness. The AIC delta statistics for most models suggest that the full models are not superior to the control models. The only plausibly significant effects are observed with the fix and patch emergence tendency models. Yet these results are blurred by huge influence swings across different types of independent variables, suggesting that the models converge more as a result of sample configuration than clear effects. Therefore, the conservative choice is made to interpret the results of the aggregate solution knowledge verifier role models, as a whole, as not significant.

In summary, the results at the organisation level reveal some similarities and some differences in effects compared to the individual level of analysis. Like at the individual level, the activities by organisation members engaging in the aggregate problem knowledge producer role are correlated with increased fix percentage, suggesting that, contrary to the hypothesis, activities do not tax absorptive capacity sufficiently to hinder knowledge transfer associated with fix emergence, and, rather, result in greater learning improving outcomes, with both individual and organisation level contributions to this positive outcome effect. Also, similar to the individual level of analysis results, different types and severity levels of activities appear to affect organisation level outcomes in a manner that is most attributable to popularity of types and priorities of the meta-organisation, which are examined separately in hypothesis three and hypothesis six.

Contrary to the results observed at the individual level, for the aggregate solution knowledge producer role, there is negative correlation between activities and fix and patch emergence tendencies. Contextual examination of the results suggests a small organisation level negative effect pulling in the opposite direction of a stronger individual level positive effect such that the net effect tends to be positive for activity engagement effects on fix and patch emergence tendencies for the developer role. Also, contrary to the individual level results, no effects were observed in any aggregate role on resolution time, suggesting that the time-based effects of activities reside primarily at the individual level.

Perhaps the most significant results observed are that aggregate problem knowledge producer role engagement has a strong effect at the organisation level, above and beyond the same effect observed at the individual level. Aggregate problem knowledge producer role engagement taxes absorptive capacity far more than other activities in the meta-organisation, resulting in increased time to resolution but also simultaneous experience effects that increase fix and patch emergence tendencies and facilitate the transition through knowledge production cycle. Aggregate solution knowledge producer role engagement at the organisation level taxes absorptive capacity for aggregate developers even more than for aggregate reporters, consistent with the results observed at the individual level. Organisation level developer absorptive capacity limits hinder resolution time, as well as fix and patch emergence tendencies, above and beyond individual level negative effects on those outcomes of interest.

A clear takeaway from the results is that there are separate organisation level effects that are often, but not always, pulling in the same direction as individual level effects. Actors in the meta-organisation must consider both levels of effects in order to improve the emergence of solution knowledge that is of benefit to them.

Hypothesis two: Codifiability

The second hypothesis postulates that, "*The codifiability of the problem knowledge revealed to the meta-organisation is positively correlated with solution knowledge emergence.*" This hypothesis is tested by analyzing data at the problem, individual, and organisation levels of analysis; cross-level nesting effects between individuals and the organisations of which they are a member are also assessed. The results for each level are discussed in turn in the following sections.

Problem level of analysis results

At the problem level of analysis there are seven measures that represent the conceptualization of the dependent outcome of interest, solution knowledge emergence, as depicted in Figure 11, and as operationalized into the variables described in Table 51. Seven measures represent the conceptualization of the antecedent of interest in hypothesis two, codifiability, as depicted in Figure 14, and as operationalized into the variables described in Table 54. The results of the analyses are depicted in the regression model output summaries in Appendix D: Regression models.

Measure one: Problem knowledge title and description lengths

The first measure of codifiability is the length of titles and the descriptions of problem knowledge sets. It is hypothesized that title and description lengths are negatively correlated with solution knowledge emergence due to the reduced codifiability of longer content.

Examination of the results suggests that title length is correlated ($p < 0.001$) with increased overall resolution time and increase time to assignment. The time-based resolution and assignment threshold regression models support these results, with significant positive correlations with the longer resolution and assignment time thresholds, and significant negative correlations with the shorter resolution and assignment time thresholds. These results are as hypothesized, suggesting that the increased codifiability complexity of longer titles has a negative effect on outcomes of interest.

Interestingly, the results also suggest that increased title length is positively correlated ($p < 0.001$) with patch emergence tendencies, which is contrary to the hypothesis and the increased time measures. Further, there is no significant correlation between development time and title length. However, title length is strongly correlated ($p < 0.001$) with increased confirmation tendencies. Considered together, these results suggest that the codifiability of the title has a different effect on the developer knowledge flow activities than on the knowledge flow activities by others in the meta-organisation. The tacit knowledge embedded in individuals or organisations engaging in the developer role is likely to be different than the tacit knowledge of other actors (c.f. Kogut & Zander, 1993; Miller, Zhao, & Calantone, 2006). It stands to reason that complexity will have different impacts based on the actor assessing the complexity. A title that is too complex for a non-developer to understand may offer valuable information for developers who decode it using their tacit knowledge. This result suggests that the outcomes of interest are affected both by the codifiable knowledge in the title and the individual engaged in the decodification process. Comparison of these results with those at individual and organisation levels of analysis sheds more light on the overall context of the effects.

Examination of the results for description length suggests that long descriptions are strongly correlated ($p < 0.001$) with increased resolution time, increased assignment time, decreased confirmation, and decreased fix and patch emergence tendencies. These negative outcomes lend support to the hypothesis that the reduced codifiability of long descriptions significantly hinders the knowledge production process resulting in reduced solution knowledge emergence. The results observed for the time based resolution and assignment time thresholds paint a clearer picture of the effects, with a positive correlation ($p < 0.01$) with "extremely slow resolution" and negative correlation ($p < 0.05$) with "extremely fast resolution", but no other significant results. It appears that very long descriptions result in problems being ignored for long periods of time until they become irrelevant, without their usefulness ever being assessed. This ignoring effect matches the absorptive capacity limit results observed with hypothesis one, where developers hit absorptive capacity limits more readily than other roles. It stands to reason that a developer whose absorptive capacity is taxed may pass over problems with very long descriptions in favour of those problems with moderate descriptions lest they end up biting off more than they can chew. The lack of impact on development time and the negative correlation with confirmation lend support to the "developer ignoring" explanation for long description results.

As discussed in Chapter Four: Research Method, the data are not conducive to quadratic analysis for description length. While the negative impact of long descriptions is evident in the results, future research may wish to consider the questions that remain concerning low and moderate description lengths as well as interactions between title and description lengths with a data set conducive to such analysis.

Measure two: Description readability

The second measure of codifiability is description readability. It is hypothesized that the more readable a description, the better the solution knowledge emergence due to the reduced complexity of the easier to read description. Readability is measured using the Flesch Reading Ease Readability Formula (Flesch, 1948; Kincaid, Aagard, O'Hara, & Cottrell, 1981).

Examination of the results reveals that description readability is negatively correlated with time to assignment of problems ($p < 0.05$) and extremely fast assignment ($p < 0.01$). This result suggests that the primary impact of readability is on the triage activity in the knowledge creation process. Easier to read descriptions are more readily understood by the triager who can quickly assign the associated problems to the right developers to address them.

It is interesting to note that the readability effect appears to be limited to the triage activity at the problem level of analysis. While the summary output and ANCOVA results suggest some plausible associations with other outcome factors such as patch emergence tendencies, confirmation tendencies, and overall resolution time, the significance of these associations drops sharply with the heteroskedasticity corrected coefficients reported in the summary tables. This drop in significance in the corrected-coefficient model results suggests that there are a number of statistical outliers that confound the base regression models and spuriously attribute effects to description reading ease that are unrelated. This finding is particularly interesting for practice as much effort has been put into coaching problem knowledge producers into creating more readable descriptions with the goal of improving solution knowledge emergence outcomes (c.f. Hooimeijer & Weimer, 2007; Dit & Marcus, 2008). Yet, this effort may be better allocated elsewhere given that it appears to only assist in triaging at the problem level of analysis, a role that is not typically viewed as needing that assistance in the meta-organisations engaging in description readability improvements.

Measure three: Presence and type of attachment

The third measure of codifiability is the presence and, if applicable, the type of attachment appended to a problem knowledge set submitted to the meta-organisation. It is hypothesized that attachments are positively correlated with solution knowledge emergence due to the alternative forms of problem knowledge representation that they provide. As discussed in Chapter Four: Research Method, the nature of the data used in the present study does not allow for analysis of types of attachment other than images. As such, this measure is operationalised as two logical variables, with the first representing presence or absence of attachments, and the second representing whether there is an attachment that was specifically of type "image". While the "image" type is particularly relevant given that it may be a separate dimension in the knowledge codification process (Chilton & Bloodgood, 2007; Johansson, et al., 2012), future research may wish to examine the effects of other types of attachments in studies with data conducive to more detailed type refinement analysis.

Examination of the regression model summary outputs reveals that presence of attachment is correlated with increased resolution time ($p < 0.001$), increased assignment time (p

< 0.05), increased confirmation ($p < 0.001$), increased reopening and reassigning tendencies ($p < 0.001$), and, increased fix emergence tendencies ($p < 0.001$). Image attachments are correlated ($p < 0.001$) with increased resolution, assignment, and development times, increased reopening tendencies, and decreased fix and patch emergence tendencies. These results paint a mixed picture of support for the hypothesis.

The improved fix emergence tendencies fit with the hypothesis that the alternate knowledge types in attachments, which facilitate codifiability have positive solution knowledge emergence outcomes. Yet, only the fix and confirmation outcomes are positive for general attachment presence. The time based and knowledge cycle based outcomes are negative, contrary to the hypothesis. One possible explanation is that attachments, which are frequently pieces of software code, require tacit subject matter knowledge in the person examining the attachment in order to improve knowledge transfer and affect outcomes positively. When an individual examines an attachment without the requisite tacit subject matter knowledge, the resulting codifiability complexity hinders the process. This explanation is supported by the lack of negative effect on development time, despite negative overall effect on resolution time and negative effect on assignment time. Developers have the tacit subject matter knowledge required to understand software code attachments and can use this knowledge to generate solutions, which explains the increased fix emergence tendencies. Triagers, by contrast, may not have the same tacit knowledge, resulting in a delayed assignment process, as observed.

The negative correlation between image attachments and fix and patch emergence tendencies observed in the results is puzzling. One possible explanation is that the image type of attachment is not particularly useful to the development process, as reflected by the correlation with increased development time ($p < 0.001$) that is not observed in the results for overall attachment presence. In fact, images may significantly hinder solution knowledge emergence as compared to other types of attachments such as software code. A clear takeaway is that not all types of attachments provide useful knowledge that improves the outcomes in the knowledge production process. Meta-organisations may be wise to restrict attachments only to types that improve solution knowledge emergence. This result is notably contrary to the survey-based results reported by Bettenburg et al. (2008), where developers involved in the Mozilla meta-organisation reported that they use screenshots (image type attachments) in the development process to improve outcomes. In their study, developers of other meta-organisations such as Eclipse reported using screenshots less than Mozilla developers and only for certain subsets of bugs (310). As such, this finding contributes an additional perspective to the literature that suggests that the self-reports of developers may differ from the actual circumstances of usefulness of image type attachments for solution knowledge production. Future research on which types of attachments have which effects on outcomes of interest using a data set conducive to such examination may be fruitful.

Measure four: N-gram profile distance of title and description from profile of fixed bugs

The fourth measure of codifiability is the n-gram profile distance of title and description of bugs from the n-gram profile of all fixed bugs. As discussed in Chapter Four: Research Method, n-gram profiles are a template-like representation of components of words that can be used as a pattern to classify pieces of textual knowledge, such as problem knowledge reports. All the bugs resolved as “fixed” are used to generate the n-gram profile of “problem knowledge reports that were fixed”. The distance of each other bug is then compared to this profile using the KLJ “distance” measure (Hornik, et al., 2013), with lower distance meaning “closer to the n-gram profile of fixed problems”.

Examination of the regression model outputs reveals that the title and description n-gram profile distance from the “fixed” n-gram profile is strongly negatively correlated ($p < 0.001$) with resolution time, assignment time, reopening tendencies, and development time, and positively correlated with fix and patch emergence tendencies and confirmation tendencies. This result is the opposite of hypothesized. The results seem to suggest that radically different sets of problem knowledge as compared to those previously submitted to the meta-organisation that were fixed are positively correlated with subsequent fix emergence. While there are circumstances where radically different knowledge is useful by virtue of its difference from existing knowledge in an organisation (c.f. Leonard-Barton, 1992; Szulanski, 1996), it is unclear how these circumstances would play out in the context of a meta-organisation.

A more likely explanation lies in the nature of the KLJ n-gram distance measure. It calculates the absolute distance between each problem’s title and description and those of the set of problems who were resolved as fixed. The magnitude of distance to the category of “fixed”, is not bounded by the magnitude of variations of distances of n-gram profiles from the “not-fixed” reference category (as considered in measure five in the next section). As such, the inverted results can be explained by the magnitude of difference between problem n-gram profiles and the reference profile so large on average that it overlaps with the magnitude of difference from the “not-fixed” category to such an extent that it flips the results. This explanation is supported by the results of measure five discussed in the following section. A clear takeaway is that unbounded single category distance measures are not suitable for classification purposes. As a result of this unbounded measure effect, the observed results are interpreted as spurious and not significant.

Measure five: Automatic classification based on n-gram profile

The fifth measure of codifiability is automatic classification of problem knowledge based on n-gram profile comparison to the collective profiles of “fixed” and “not_fixed” bugs. As discussed in Chapter Four: Research Method, this measure is a logical “guess” of how each bug should be classified based on the n-gram profiles of all other bugs (excluding itself to avoid endogeneity). This measure complements the previous measure by providing a “net distance” from both “fixed” and “not_fixed” n-gram profiles, rather than an absolute distance that is unbounded.

Examination of the regression model summaries reveals that whether a problem’s title and description are closer to the n-gram profile of “fixed” problems than the n-gram profile of “not_fixed” problems is negatively correlated with resolution time ($p < 0.001$), assignment time ($p < 0.001$), and reopening tendencies, and positively correlated with fix and patch emergence tendencies ($p < 0.001$), and confirmation tendencies ($p < 0.001$).

This result is as hypothesized and suggests that n-gram profile based categorization of problems based on title and description is a useful method for predicting solution knowledge emergence outcomes. It validates n-gram profiles as operational codifiability measures in the case of problem knowledge in meta-organisations, a context not previously considered in the literature. It also validates that unbounded absolute distance measures are insufficient for categorization purposes.

Given the theoretical and practical benefits associated with improving automatic bug classification (c.f. Čubranić & Murphy, 2004; Antoniol, et al., 2008; Herzig, Just, & Zeller, 2013) future research that refines the classification abilities of this measure are encouraged. In particular, the ability of this knowledge codification measure of predicting future outcomes that are not represented in the same database should be first priority to ensure that the present results do not suffer from endogeneity in the n-gram profile algorithm's calculations.

Measure six: Redundancy of new problem knowledge

The sixth measure of codifiability is the redundancy of new problem knowledge reveals to the meta-organisation. The effect of two variables, whether a bug is a duplicate, and whether a bug has a duplicate, are assessed on the outcomes of interest.

Examination of the regression model summaries reveals that new problem knowledge that is a duplicate is correlated with ($p < 0.001$) faster overall resolution and faster development, slower assignment ($p < 0.01$), reduced reopening and reassigning tendencies ($p < 0.001$), and increased ($p < 0.001$) confirmation tendencies. These results present a mixed picture of the impact of the duplicate problem knowledge that is as expected. When a duplicate problem enters the knowledge production process, the triager spends time attempting to locate its counterpart. In cases when the counterpart is found, no assignment becomes necessary, resulting in the exclusion of such cases, as noted in the observation count difference between the overall resolution time and assignment time models (651,244 vs 113,156). Those cases that are duplicates that do become assigned are necessarily the result of misidentification by the triager who doesn't immediately recognize them as duplicates despite having searched for a duplicate, which prolongs the time to assignment, as observed. By definition, these duplicate bugs do not go on to development, so when they are mistakenly assigned, developers very quickly resolve them, which explains the fast "development" time, which is actually the "reclassification as duplicate" by the developer instead of the triager, correcting the latter's initial mistake.

In other words, the negative impact on the participants in the meta-organisation of the submission of duplicate problem knowledge is evident in the regression model results. While the lower time to resolution would be considered a positive outcome in the case of an associated fix, for duplicate problem knowledge, it is simply the time spent on separating and discarding the duplicate knowledge; it is wasted time, even if the resolution is faster than producing solutions.

In the case of a set of problem knowledge having a duplicate, the regression model summaries reveal a strong ($p < 0.001$) positive correlation with resolution time, assignment time, development time, reopening and reassigning tendencies, confirmation tendencies, as well as with fix and patch emergence tendencies. Once again, these results paint a complex picture that fits the extant understanding of the knowledge production process. Problems that end up being duplicated are most often those problems that remain pending, or unresolved, for a long time, leaving open the possibility of a separate actor coming along and requesting a solution to the same problem, not realising that the problem knowledge had already been submitted previously. Were the problem resolved promptly, the subsequent actors would already have the solution knowledge and would have no motivation to submit the duplicate problem knowledge. Further, bugs that remain open for a long time are bugs that are often viewed as valid and important, as the correlation with confirmation reveals; they tend to be complex in nature, requiring complex solutions that are not always straightforward, as the correlation with reopening and reassigning tendencies reveals. Yet, given the importance of the solutions to these problems, fixes and patches do eventually emerge, as seen in the correlation with increased fix and patch emergence tendencies.

The clear takeaway is that duplicate problem knowledge, of its own accord, is a drain on the meta-organisation, but it plays an important role by signalling the value of the knowledge that it is duplicating in the meta-organisation. The problem itself will get resolved, though the resolution will be associated with the first problem knowledge reveal, not the subsequent duplicate reveal. This result further highlights the important distinction between knowledge outcome associations, where different outcomes may occur for the same set of problem knowledge based on the order of the problem knowledge reveals, independent of the properties of the problem knowledge itself. Future research that examines longitudinal data using survival hazard models may be able to better pinpoint the time based effects of redundant problem knowledge.

Measure seven: Number and length of comments

The seventh measure of codifiability is the number and length of comments. This measure complements the first measure by examining the effects of the knowledge contained in the comments attached to problem knowledge submissions on outcomes of interest. Greater number and length of comments are hypothesized to be positively corelated with solution knowledge emergence due to the additional problem knowledge contained therein. As discussed in Chapter Four: Research Method, the nature of the distribution of the comment count in the data is problematic for regression modelling such that it is necessary to operationalise it as a logical variable of “more than fifty comments” for the purpose of regression modeling and hypothesis testing.

Examination of the regression model summaries reveals that mean length of comments is positively correlated with resolution time ($p < 0.001$), assignment time (($p < 0.01$), development time ($p < 0.05$), reopening tendencies ($p < 0.05$), and reassigning tendencies (0.01). This result is the opposite of hypothesized, suggesting that longer comments introduce complexity that hinders emergence of solution knowledge. This finding is particularly interesting given the purpose of the comment system as a meta-organisational function is to facilitate the knowledge production process. Yet, it appears to have the opposite effect when it comes to the mean length of the comments. It may be that there is an optimal length of comment, not too long and not too short (U-shape relationship) that has a positive influence, whereas very long comments introduce disproportionate complexity. Future research with data conducive to quadratic analysis may wish to examine this possibility.

In the case of problems with a large number of comments, the regression model summaries reveal a strong positive correlation ($p < 0.001$) with all the outcomes of interest, suggesting that problems with a very large number of comments take longer to resolve, be assigned, and for solutions to emerge. The knowledge production process is also less direct, with higher confirmation probability but also higher reopening and reassigning tendencies. At the outset, fix and patch emergence tendencies are better. This result is partially as expected as the additional knowledge from the comments appears to improve fix and patch emergence tendencies, although it does so at the cost of increased time to resolution and a less direct knowledge production process. The additional knowledge requires additional time to be codified, transmitted, internalized, and applied by actors in the meta-organisation before it can improve fix and patch emergence tendencies. This result also suggests a difference between the knowledge that emerges from long comments and the knowledge that emerges from a large number of comments. It may be that it is the convergence of knowledge from multiple sources in multiple comments that is more useful for solution knowledge emergence, whereas a large amount of knowledge from fewer sources in a single large comment or a few large comments is less useful for solution knowledge emergence. Future research using data more conducive to refined analysis on the nature of comment effects based on size, count, and source may prove fruitful in further contextualizing the observed results.

Summary of dependent variable effect results at problem level

Comparison of the control and full regression models for the effect of codifiability measures on the dependent outcomes of interest reveals an overall picture of effect size at the problem level of analysis.

The model F, Chi-squared, and comparative AIC delta statistics suggest superiority of all full regression models above and beyond the control only models. The stronger incremental effects are observed in resolution timing, "medium", with R^2 increase from 0.117 to 0.232; fix emergence tendencies, "very large" with R^2 increase from 0.430 to 0.628; patch emergence tendencies, "medium-to-large", with R^2 increase from 0.371 to 0.480; and, confirmation tendencies, "medium-to-large", with R^2 increase from 0.525 to 0.603. The weaker incremental effects are observed for assignment and development timing, "small-to-medium" and "small" respectively, along with their timing threshold outcome effects, with most incremental effects being "very small". Reassigning and reopening tendency incremental effects are also relatively weak, with "small" effect size. Comparison of these effect sizes at the problem level of analysis with those at the individual and organisation levels of analysis provides a clearer picture of the level at which codifiability effects are predominant for each outcome.

In summary, the effects of the codifiability independent variables at the problem level analysis are a careful balance between knowledge that is so complex that it requires subject matter expertise tacit knowledge to untangle and knowledge that is more widely understandable. Long titles convey useful information for triaging, but long descriptions get ignored. Some types of attachments improve outcomes, whereas image attachments hinder. Duplicate knowledge wastes energy in its processing but provides useful signaling about the knowledge it is duplicating, which, in turn, is associated with positive outcomes as a result. And, comments slow outcomes by requiring time to be digested, but may improve fix and patch emergence tendencies as a result. Examination of the results at the individual and organisation levels of analysis provides a broader contextualisation of the results observed at the problem level of analysis.

Individual level of analysis results

At the individual level of analysis there are seven measures that represent the conceptualization of the dependent outcome of interest, solution knowledge emergence, as depicted in Figure 19, and as operationalized into the variables described in Table 79 and Table 87. Five measures represent the conceptualization of the antecedent of interest in hypothesis two, codifiability, as depicted in Figure 21, and as operationalized into the variables described in Table 90. The results of the analyses are depicted in the regression model output summaries in Appendix D: Regression models.

As discussed in Chapter Four: Research Method, regression models are separated according to the roles in which individuals engage when participating in the knowledge production process: problem knowledge producer (reporter), solution knowledge producer (assigned_to), and solution knowledge verifier (QA_contact).

Measure one: Mean description length of problems acted upon in each role

The first measure of codifiability at the individual level of analysis is the mean description length of problems acted upon in each role. A tendency for longer descriptions on problems on which individuals act is theorized to be correlated with positive solution knowledge emergence outcomes due to the additional knowledge in the longer descriptions.

Examination of the regression model summaries reveals that mean length of descriptions of problems acted upon is strongly positively correlated ($p < 0.001$) with resolution time and strongly negatively ($p < 0.001$) correlated with fix emergence tendencies. These results are the opposite of theorized but match the results observed at the problem level of analysis. They suggest that increased description length results in complexity that hinders codifiability, reducing fix emergence tendencies and increasing the time to resolution. The results are the same for all three roles in which individuals engage, suggesting that the effect is a function of the description itself, not of the role in which individuals engage when acting upon the description's associated problem. This result is particularly important as it is contrary to the conventional wisdom that longer descriptions are better because they provide more detail to assist in problem resolution. Rather, shorter descriptions that are concise and to the point may be more useful for promoting solution knowledge emergence.

Measure two: Mean readability of descriptions of problems acted upon in each role

The second measure of codifiability is the mean readability of the descriptions of problems acted upon in each role. A tendency to have descriptions on bugs on which individuals act that are more readable is theorized to be correlated with positive solution knowledge emergence outcomes due to the increased ease of interpreting more readable knowledge.

Examination of the regression model summaries reveals that the mean readability of descriptions is strongly positively correlated ($p < 0.001$) with resolution time and strongly negatively correlated ($p < 0.001$) with fix and patch emergence tendencies across roles. This finding is contrary to hypothesized, suggesting that easier to read descriptions result in negative solution knowledge emergence outcomes. Whereas at the problem level of analysis outcome effects on fix and patch emergence tendencies fell below an acceptable threshold of significance after heteroskedasticity correction, at the individual level of analysis, aggregating the description readabilities according to the individuals who create them, reveals that the skew observed at the problem level was likely the result of individual level effects creating problem level outliers. Given that the readability of a description is a function of the writing ability of the individual who creates the description, it stands to reason that individual level effects would present a clearer picture of the overall effect on the outcomes of interest.

The results suggest that general readability measures, even those measures designed for assessing technical material readability such as the Flesch Reading Ease Readability Formula used in the present study (Flesch, 1948; Kincaid, Aagard, O'Hara, & Cottrell, 1981), do not take into account the tacit knowledge of specialized subject matter experts and the codifiability advantages associated with shared tacit knowledge between problem knowledge producer and solution knowledge producer. By using overly simplified language, whereas the readability increases from the perspective of the general public who lacks tacit subject matter expertise, there is a lack of domain specific knowledge in the descriptions, possibly at the cost of very long descriptions attempting to describe something that could be described succinctly using an artefact of knowledge that is specific to experts in the domain (which explains the negative description length effects observed in the results of the previous measure). By contrast, descriptions that are less “readable” to the general public may be scored as such because they contain more subject matter specific artefacts that knowledgeable individuals can more readily decode with the tacit knowledge they accumulate during their expertise development. These descriptions are more “readable” by subject matter experts, which is not captured by the general readability measure.

In summary, these results suggest that readability effects of descriptions are strongly correlated with outcomes of interest. However, the nature of what constitutes readability is in question, and conventional measures may not be suitable for assessing the usefulness of the information contained when paired with tacit subject matter expertise in the consumer of the description knowledge. The contributions to research are that specialized measures of readability need to take into account the subject matter expertise of consumers of the knowledge being transmitted and readability measures designed for general public consumption may show inverse effects as a result. The contributions to practice are that subject matter specific artefacts improve codifiability of knowledge even if they hinder general public readability. As such, standardized terms in meta-organisations are important to ensure codified knowledge is passed efficiently between problem knowledge producer and solution knowledge producer, promoting solution knowledge emergence.

Measure three: Mean number of attachments to problems acted upon in each role

The third measure of codifiability is the mean number of attachments to problems acted upon in each role. A tendency to have more attachments to problems on which individuals act is theorized to be correlated with positive solution knowledge emergence outcomes due to the increased ease of codifying knowledge that is represented in different ways given that attachments complement the problem description knowledge with alternative forms of knowledge.

Examination of the regression model summaries reveals that the mean number of attachments to problems acted upon is strongly negatively correlated ($p < 0.001$) with reopening and reassigning tendencies and strongly positively correlated ($p < 0.001$) with fix and patch emergence tendencies across roles. This result is as hypothesized, supporting the notion that the alternative representations of knowledge contained in attachments improve codifiability of the problem knowledge, resulting in improved solution knowledge emergence.

At the individual level of analysis, the mean number of attachments is considered, whereas at the problem level of analysis only the presence or absence of an attachment is considered. This alternate representation of the attachments measure offers results that suggest that multiple attachments may be necessary for improving solution knowledge emergence. Whereas the presence of an attachment at the problem level results in improved fix emergence tendencies, it also results in increased reopening and reassigning tendencies, which are not desired. By contrast, at the individual level of analysis, greater mean number of attachments to problems acted upon in each role is correlated with reduced reopening and reassigning tendencies (in addition to improved fix emergence tendencies). This difference in results across levels suggests that the effect of attachments may not be completely incremental. A single attachment may have a different effect than multiple attachments, with multiple attachments being clearly superior. This effect may be attributable to the type of attachment given that the image type attachment results in uniformly negative effects at the problem level of analysis. It may be that when there is a single attachment type, the weight of the image type attachment draws the outcome effect in a negative direction. When there are multiple attachments, it is more likely that one ore more of them are not of the image type, resulting in the weight of attachment effects drawing the outcome effects in a positive direction.

In summary, across roles, attachments to problems improve outcomes, although careful thought should still be given to the type of attachment, as the image type likely is negative regardless of the level of analysis. For practice, the takeaway is that individuals who have a tendency to submit more attachments with the problems they submit to the meta-organisation tend to experience better solution knowledge emergence outcomes.

Measure four: Redundancy tendencies of problems acted upon in reporter role

The fourth measure of codifiability is the redundancy tendencies of problems acted upon in the problem knowledge producer role. A tendency to submit duplicate problems is theorized to be correlated with negative outcomes, whereas a tendency to act on problems knowledge that is subsequently duplicated by later problem knowledge submissions is theorized to be correlated with positive outcomes. Given that problem knowledge is only created by the problem knowledge (reporter) role, this measure is only defined for that role.

Examination of the regression model summaries reveals that tendencies to submit duplicate problem knowledge by individuals acting in the problem knowledge producer role is strongly negatively correlated ($p < 0.001$) with resolution time, reopening tendencies, and fix and patch emergence tendencies. These results match those observed at the problem level of analysis, suggesting that a tendency to submit duplicate problem knowledge has negative outcomes. As discussed at the problem level of analysis, whereas faster resolution would be considered a positive outcome in other contexts, for the case of duplicate bugs, the faster resolution time is the result of quick triaging into a "not fixed" outcome and represents faster discarding, rather than faster solution creation.

For the tendency to report problems that are later duplicated by subsequently submitted problem knowledge, the regression model summaries reveal a strong positive correlation ($p < 0.001$) with resolution time, reassigning tendencies, and fix and patch emergence tendencies. These results match those observed at the problem level of analysis, suggesting that there is significant value in duplicate bug reports that manifests in the outcomes of the original problem reports, as reported in the extant literature (c.f. Zimmermann, et al., 2010). Further, there is an individual level effect such that individuals who have a tendency to report bugs that are later duplicated benefit from consistently improved outcomes in terms of fix and patch emergence tendencies. As discussed at the problem level of analysis, increased resolution time in the present context suggests that bugs that are open long enough to have duplicate problem knowledge subsequently submitted are likely of sufficient complexity that they take longer to resolve than simpler bugs, making it not a clearly negative outcome when paired with positive fix and patch emergence tendencies.

Measure five: Mean number and length of comments on problems acted upon in each role

The fifth measure of codifiability is the mean number and length of comments on problems acted upon in each role. It is hypothesized that more and longer comments improve solution knowledge emergence because they provide additional problem knowledge that improves codifiability.

Examination of the regression model summaries reveals that mean comment length has a split correlation amongst the roles in which individuals engage on problems. For the problem knowledge producer role, it is positively correlated with resolution time ($p < 0.001$), and negatively correlated with reopening ($p < 0.01$) and reassigning ($p < 0.001$) tendencies and fix ($p < 0.001$) emergence tendencies. By contrast, for the solution knowledge producer role, the correlation with fix emergence tendencies is positive ($p < 0.01$) and patch emergence tendencies are also positive ($p < 0.01$). For the solution knowledge verifier role, there is also positive correlation ($p < 0.05$) with patch emergence tendencies.

The split in results amongst roles suggests an individual level effect independent from the predominantly negative effects observed at the problem level of analysis. The individual level results suggest that individuals who engage in the problem knowledge producer role suffer negative outcomes from a tendency for comments that are overly long, consistent with the findings at the problem level of analysis. Given that the problem knowledge producer role is the role in which individuals engage in most frequently, it stands to reason that this effect would be predominantly observed when considering all problems at the problem level of analysis. Yet, the solution knowledge producer role reveals a different picture. A tendency for longer comments results in positive fix and patch emergence tendencies, contrary to the problem knowledge producer role and problem level of analysis results. This result suggests that the nature of the comments on problems that solution knowledge producers choose to work on may be different from the comments on a broader class of problems—problems that solution knowledge producers may choose to not work on. Longer comments with additional knowledge that is useful to the solution knowledge producer may draw more developer attention, resulting in improved fix and patch emergence tendencies. In other words, these results suggest that the mean comment length alone is not the primary factor that affects solution knowledge emergence. Rather, the content of the comments and the roles that engage with the problems to which the comments are attached appear to also affect outcomes.

In the case of the mean number of comments, the role-specific effect is mostly the opposite as for mean comment length. For the problem knowledge role, the mean number of comments is strongly positively correlated with resolution time ($p < 0.001$), reopening and reassigning tendencies ($p < 0.001$), and fix and patch emergence tendencies ($p < 0.001$). A longer resolution time for more comments stands to reason simply because problems that are open longer will tend to, on average, collect more comments as they have more time for the comments to be submitted. The other effects appear inverted relative to the comment mean length results, with greater reopening and reassigning tendencies for more comments, whereas there are fewer for longer comments. And, surprisingly, there are positive fix and patch emergence tendencies for more comments, whereas fix emergence tendencies are negative for longer comments (and patch emergence is not significant).

For the solution knowledge producer role, the mean number of comments is strongly negatively correlated ($p < 0.001$) with fix and patch emergence tendencies, the opposite direction of association as for mean comment length. The solution knowledge verifier role also shows a correlation between the mean number of comments ($p < 0.001$) and reduced patch emergence tendencies. This result also suggests that number and length of comments have fundamentally different properties when it comes to outcomes, and, that these properties are a function of the role that engages on the problems with which they are associated. The result is a 2 X 2 matrix for each comment property, as described in Table 115 and Table 116.

		Role	
		Problem knowledge producer	**Solution knowledge producer**
Mean comment length	**Shorter**	+	-
	Longer	-	+

Table 115: Role based correlation between mean comment length and fix emergence tendencies

		Role	
		Problem knowledge producer	Solution knowledge producer
Mean number of comments	Fewer	-	+
	More	+	-

Table 116: Role based correlation between mean number of comments and fix emergence tendencies

The implication for theory is that "length" of additional knowledge may be a poor measure of codifiability as it says nothing about the nature of the knowledge. Lots of "bad" knowledge may lead to worse outcomes. Less "good" knowledge may lead to better outcomes. The implication for practice is that problem knowledge producers must be careful to produce and solicit additional knowledge in the comments such that the knowledge is useful not for the general meta-organisation populace, but rather specifically useful for the solution knowledge producer role. As is the case for the readability measure, it may be necessary to submit knowledge that is more difficult for the general user base to consume if it is more useful for the specific subset of individuals who have the necessary subject matter expertise tacit knowledge to produce associated solution knowledge. "More" knowledge, as measured by number of comments or length of comments may not uniformly result in better outcomes.

Effects of individual nestedness in organisations

Examination of the mixed-effects regression model summaries reveals that there are significant organisational effects that affect the individual level results. The AIC and BIC delta statistics reveal superior models with organisational random effects included, with the effect size of organisations isolated from the fixed individual level effects ranging from "medium" to "medium-to-large" on the dependent outcomes of interest for the problem knowledge producer role, from "small" to "very large" for the solution knowledge producer role, and "small-to-medium" to "large" for the solution knowledge verifier role.

Despite the significant organisational effects, the significance of the effects of the individual level independent variables remains largely unchanged, with some notable exceptions. For comment mean length effects on fix emergence tendencies for the problem knowledge producer role, the logit-transformed OLS model reports positive correlation ($p < 0.05$), whereas the beta regression percent model reports negative correlation ($p < 0.001$). The inclusion of the organisation random effects in the mixed-effects model results in loss of significance for the comment mean length independent variable. There are a few reasons that support the individual level regression model results as superior. First, beta regression is specifically designed for the purpose of assessing regressions of independent variables on percentage dependent variables, whereas logistic transformation of percent variables, while resulting in better linearity, doesn't account for the other properties inherent to percent dependent variable distributions (Smithson & Verkuilen, 2006; Cribari-Neto & Zeileis, 2010; Simas, Barreto-Souza, & Rocha, 2010; Grün, Kosmidis, and Zeileis, 2012). Second, the organisation random effect portion of the mixed-effect model, while resulting in an effect that is not significant, has a negative coefficient, the opposite of the coefficient in the logit-transformed OLS model and the same as the independent level of analysis beta regression model. This result suggests that there may be both individual and organisation level effects at play, which may not be well-accounted for in the logit transformation. Third, the sample size is much larger in the individual level of analysis models than the mixed-effects models (n = 17,554 vs. 3751) because the organisation level constraints are applied to the sample in the mixed-effects models. Therefore, the individual level model is more likely to contain a broader range of individual level specific effects whereas the individual level fixed effects in the mixed-effects model, by virtue of the selection contraints alone, may be skewed towards organisation level effects. And, fourth, as discussed in Chapter Five: Analysis, the individual level models include control variables that account more specifically for other effects, whereas the mixed-effects models do not, by design. Nevertheless, it would be prudent to interpret the comment mean length effects on fix emergence tendencies conservatively. Future research with a data set conducive to more detailed comment analysis may be fruitful in shedding further light on the mechamism behind the observed results.

A similar split of results is observed for the solution knowledge producer role and mean number of comments. The logit-transformed OLS models with the smaller sample size (n = 2453 vs. 1171) and associated mixed-effects models fail to achieve significance for fix and patch emergence tendencies, suggesting a similar confounding as with mean comment length. For similar reasons as those discussed above, the individual level models are likely to be superior, but triangulation with the organisation level models is the best method of determining the degree of certainty of individual level comment effects on the outcomes of interest, which is discussed in the next section.

Summary of dependent variable effect results at individual level

Comparison of the control and full regression models for the effect of codifiability independent variables on the dependent outcomes of interest reveals an overall picture of effect size at the individual level of analysis. For the problem knowledge producer role, the comparative Chi-squared statistic and comparative AIC delta statistic reveal that the models that include the codifiability independent variables are all superior to the control-only models. The additive effect size of the independent variables above and beyond the control variables is "large" for days to resolution dependent variable, "small-to-medium" and "medium" for reopening and reassigning tendencies, respectively, and "large" and "very large" for fix and patch emergence tendencies, respectively.

For the solution knowledge producer role, the comparative Chi-squared and comparative AIC delta statistics also suggest that the full models are superior to the control models in all cases. The additive effect size of the independent variables above and beyond the control variables is "large" for days to resolution, "medium" and "small-to-medium" for reopening and reassigning tendencies respectively, and "medium" and "very large" for fix and patch tendencies respectively.

The solution knowledge verifier role shows uncharacteristically strong independent variable effects despite the limited sample size, attributable primarily to the description length and comment mean length variables. While most effect size results are sizeable for good reason, of note, the effect size of "very large" for the patch emergence tendency model is likely spurious as its strongest coefficient is related to attachments, and the presence of attachments may be an endogenous effect that is more a function of verifying existing patches rather than a function of the role-specific effects in the case of the solution knowledge verifier role. Therefore, this particular effect size should be interpreted with caution and more moderate overall effects should be interpreted from the other roles.

In summary, the individual level codifiability effects are best characterised as a balance between the type of knowledge interacting with the role that is consuming it. Longer descriptions and comments tend to be worse for problem knowledge producer role outcomes but better for solution knowledge producer role outcomes. A greater number of comments tends to be better for problem knowledge producer role outcomes but worse for solution knowledge producer role outcomes.

Interestingly, whereas description readability had no significant effect at the problem level, its effect is prominent at the individual level, with more readable descriptions resulting in significantly worse outcomes. Solution knowledge producers interpret "readability" differently than the general public, where subject matter expertise specific jargon and tacit knowledge contained in the individuals flip the effects of hard to read descriptions such that they improve outcomes for developers.

Similar to at the problem level of analysis, attachments have positive outcomes for all roles, and duplication of problem knowledge improves outcomes for the original problem knowledge set across roles, suggesting that the effort spent on triaging the duplicate knowledge may payoff overall.

Collectively the results suggest that the nature of problem knowledge and emergent knowledge plays a large role in outcome effects. There is a careful balance between the usefulness of simplicity and subject matter expertise tacit knowledge in the codifiability. As observed in the results of Hypothesis one: Absorptive capacity, the role that consumes the knowledge significantly changes the outcome effects independent of the properties of the knowledge itself. Likewise, examination of the mixed-effects nested models suggests organisation level effects that are more directly examined in the next section.

Organisation level of analysis results

At the organisation level of analysis there are seven measures that represent the conceptualization of the dependent outcome of interest, solution knowledge emergence, as depicted in Figure 26 and as operationalized into the variables described in Table 105. Five measures represent the conceptualization of the antecedent of interest in hypothesis two, codifiability, as depicted in Figure 28, and as operationalized into the variables described in Table 108. The results of the analyses are depicted in the regression model output summaries in Appendix D: Regression models.

As discussed in Chapter Four: Research Method, regression models are separated according to the aggregate roles in which members of organisations engage when participating in the knowledge production process: aggregate problem knowledge producer (reporter) role, aggregate solution knowledge producer (assigned_to) role, and aggregate solution knowledge verifier (QA_contact) role.

Measure one: Mean description length of problems acted upon in each aggregate role

The first measure of codifiability at the organisation level of analysis is the mean description length of problems acted upon in each aggregate role. A tendency to act on problems with longer descriptions is theorized to be correlated with positive solution knowledge emergence outcomes due to the additional knowledge in those longer descriptions.

Examination of the regression model summaries reveals significant positive correlation ($p < 0.001$) with resolution time and significant negative correlation ($p < 0.001$) with reopening tendencies for the aggregate problem knowledge producer role; significant positive correlation ($p < 0.001$) with resolution time and significant negative correlation ($p < 0.001$) with patch emergence tendencies for the aggregate solution knowledge producer role; and, no significant effects for the aggregate solution knowledge verifier role.

These results are similar to those observed in the individual level of analysis results. They suggest that there is an additive organisation level negative effect of description length on resolution time for the aggregate roles. They further echo the negative effect on the aggregate developer role, with longer descriptions correlated with negative patch emergence tendencies.

The takeaway from these results is that longer descriptions may be ignored by organisational actors, resulting in negative outcomes. Some organisations, such as those more actively involved in open source, may have internal standards that dictate description formats that are deemed acceptable and those that are deemed irrelevant (Anvik & Murphy, 2011; Shokripour et al., 2013) and these standards may affect the aggregate role actors above and beyond individual level factors.

Measure two: Mean readability of descriptions of problems acted upon in each aggregate role

The second measure of codifiability is the mean readability of the descriptions of problems acted upon in each aggregate role. A tendency to act on problems with more readable descriptions is theorized to be correlated with positive solution knowledge emergence outcomes due to the increased ease of interpreting more readable knowledge.

Examination of the regression model summaries reveals that the mean readability of descriptions is correlated with negative outcomes for all aggregate roles, similar to at the individual level of analysis and contrary to the hypothesis. For the aggregate problem knowledge producer role, more readable descriptions are correlated with increased resolution time ($p < 0.05$) and worse fix emergence tendencies ($p < 0.001$). For the aggregate solution knowledge producer role, more readable descriptions are correlated with worse fix ($p < 0.05$) and patch emergence tendencies ($p < 0.001$). For the aggregate solution knowledge verifier role, more readable descriptions are correlated with worse fix emergence tendencies ($p < 0.001$).

The most likely explanation for these results, as discussed at the individual level of analysis, is that there are subject matter expertise tacit knowledge artefacts that reside within organisations, in addition to within individuals, that make problem descriptions that are less generally readable, as per the Flesch measure, less codifiable than those descriptions that use shared knowledge artefacts. The implication of this result is that actors wishing to improve solution knowledge emergence must take into account both individual and organisation-specific subject matter expertise when determining what problem knowledge to include in problem descriptions they reveal to the meta-organisation. The implication for theory is novel empirical evidence to support the notion that tacit knowledge may reside at the organisation level in addition to within individuals, as theorized in the extant literature (c.f. Nonaka, 1994; Grant, 1996a, 1996b; Cook & Brown, 1999; Ambrosini & Bowman, 2001; Nonaka & von Krogh, 2009).

Measure three: Mean number of attachments to problems acted upon in each aggregate role

The third measure of codifiability is the mean number of attachments to problems acted upon in each aggregate role. A tendency to act on problems with a greater number of attachments is theorized to be correlated with positive solution knowledge emergence outcomes due to the alternative forms of knowledge representation in the attachments.

Examination of the regression model summaries reveals that the mean number of attachments to problems is generally correlated with positive outcomes for all aggregate roles in which organisation members engage, with the mean number of attachments strongly positively correlated with increased fix emergence tendencies ($p < 0.001$) for all three aggregate roles.

Considering these organisation level results in combination with the mixed-effects nested regression model results suggests that there is a separate incremental organisation level specific effect of mean number of attachments above and beyond the observed individual level effects. One possible explanation is that certain organisations use a standard type or format of attachment that represents problem knowledge that is compatible with the tacit knowledge that resides in that organisation. The implication of this result for practice is that meta-organisation participants should be aware of the attachment practices of both individuals and organisations in order to improve solution knowledge emergence outcomes. The contributions to theory are novel empirical evidence that supports the notion that different organisations favour different types of attachments in the solution knowledge production process, as reported in the survey-based study by Bettenburg et al. (2008).

Measure four: Redundancy tendencies of problems acted upon in aggregate reporter role

The fourth measure of codifiability is the redundancy tendencies of problems acted upon in the aggregate problem knowledge producer role. A tendency to submit duplicate problems is theorized to be correlated with negative outcomes, whereas a tendency to act on problems knowledge that is subsequently duplicated by later problem knowledge submissions is theorized to be correlated with positive outcomes. Given that problem knowledge is only created by the aggregate problem knowledge (reporter) role, this measure is only defined for that aggregate role.

Examination of the regression model summaries reveals that tendencies to submit duplicate problem knowledge by organisation members acting in the aggregate problem knowledge producer role are strongly correlated with worse fix and patch emergence tendencies ($p < 0.001$) and faster resolution ($p < 0.001$), albeit meaning faster "rejection" in this context. There is also a negative correlation ($p < 0.01$) with reopening tendencies. These results match those observed at the individual level of analysis. They suggest that duplicate problem knowledge, itself, results in no positive solution knowledge emergence.

By contrast, a tendency to submit problem knowledge that is later duplicated by subsequent problem knowledge submissions is strongly correlated with positive fix emergence tendencies ($p < 0.01$), albeit with the problems being resolved more slowly ($p < 0.05$). Interestingly, at the organisation level, there is no significant correlation between knowledge duplication tendencies and increased patch emergence tendencies, which is observed in the individual level results. This difference suggests that while duplicate knowledge appears to improve outcomes for the duplicated problem knowledge and the roles that act upon it, at the organisation level the duplicate knowledge does not affect patch emergence tendencies. Patch emergence tendency improvement resides only at the problem and individual levels of analysis. A possible explanation is that individuals track duplicate bugs that are of interest to them, whereas organisations do not track duplicates as systematically. If that is the case, there may be value for organisations that participate in meta-organisations in engaging systematically with duplicate problem knowledge in a manner similar to individuals. If nothing else, this difference signals the possibility that duplicate knowledge is used differently by individuals and organisations. It may be that organisations are tapping too many sources of knowledge in the meta-organisation, resulting in them ignoring some in favour of others (Anand, Glick, & Manz, 2002); or, it may be that organisations have more difficulty with the format in which the duplicate knowledge emerges (Bock, et al., 2010), the electronic Bugzilla repository, and hence are not as easily able to access and make use of the duplicate knowledge as are individuals.

These results provide partial empirical support for the usefulness of duplicate problem knowledge in meta-organisations, building upon the work of Zimmermann, et al. (2010). Further research that specifically narrows down the nature of the effect of duplicate problem knowledge on patch emergence tendencies using a suitable data set may be fruitful for further theoretical development in this research direction.

Measure five: Mean number and length of comments on problems acted upon in each aggregate role

The fifth measure of codifiability is the mean number and length of comments on problems acted upon in each aggregate role. It is hypothesized that more and longer comments are correlated with improved solution knowledge emergence because of the additional problem knowledge they provide.

Examination of the regression model summaries reveals that the organisation level effects are very similar to the effects observed in the individual level results. Mean comment length, for the aggregate problem knowledge producer role, is positively correlated with longer resolution time ($p < 0.001$), and worse fix emergence tendencies ($p < 0.01$), albeit with fewer reassignments ($p < 0.001$). For the aggregate solution knowledge producer role, longer comments are correlated with improved patch emergence tendencies ($p < 0.001$).

Comment count, for the aggregate problem knowledge producer role, is positively correlated with better fix and patch emergence tendencies ($p < 0.001$), albeit at the cost of longer resolution time ($p < 0.001$) and higher reopening and reassigning tendencies ($p < 0.001$).

A notable difference at the organisation level is in the case of comment count for the aggregate solution knowledge producer role. Whereas at the individual level, a greater number of comments is correlated with worse outcomes for the solution knowledge producer role, at the organisation level, for the aggregate solution knowledge producer role, a greater number of comments is correlated with increased fix and patch emergence tendencies ($p < 0.001$). One possible explanation is that the organisation level effects pull in the opposite direction of the individual level effects in the case of the aggregate solution knowledge producer role. This explanation is unlikely because the logit-transformed mixed-effects models suggest that, while the overall logit-transformed models did not achieve significance for OLS or mixed-effects regression fitting, the random effects contribution of organisations pulled the coefficient in the negative direction, contrary to what would be expected if the organisation level effects were more positive for the aggregate solution knowledge producer role than the individual level solution knowledge producer role effects.

An alternative explanation relates to the operationalisation choice, discussed in Chapter Four: Research Method, to represent comment count at the organisation level as "manual" comments. A more plausible explanation is that manual comments are far more useful for the aggregate solution knowledge producer role than automated comments, resulting in the observed positive patch emergence tendencies in the results. Whereas automatic commenting exists with the intent of facilitating knowledge codification with standardized parameters, this result suggests that they may not be as useful as intended. Rather, a focus on manual comments may be the most useful. Future research that examines the nature of types of comments more specifically may be fruitful in further elaborating on the observed effect.

Summary of dependent variable effect results at organisation level

Comparison of the control and full regression models for the effect of codifiability independent variables on the dependent outcomes of interest reveals an overall picture of effect size at the organisation level of analysis. For the aggregate problem knowledge producer role, the Chi-squared statistics and comparative AIC delta statistics reveal that the full model is superior to the control-only model for all dependent variables. For resolution time, the additive effect size of the full model above and beyond the control model is "medium-to-large", with R^2 value increasing from 0.118 to 0.282. For bug fix and patch emergence tendencies, the additive effect sizes are "large" and "medium-to-large" with R^2 values increasing from 0.205 to 0.381 and 0.247 to 0.403 respectively.

For reopening and reassigning tendencies, the comparative AIC delta statistic values are relatively weak as compared to the other outcome effects, which stands to reason given the infrequently observed significant effects of the codifiability independent variables on the outcomes at the organisation level. For reopening tendencies, the incremental effect is “small”, and for reassigning tendencies it is “medium”. In context, with the low delta AIC statistics values, the overall effect sizes are best interpreted as “small”, as the “medium” effect size is likely spuriously inflated due to the fewer degrees of freedom compared to the other measures and the negligible full model superiority.

For the aggregate solution knowledge producer role, the Chi-squared statistics and comparative AIC delta statistics reveal all full models as superior, albeit negligibly so in the case of reopening and reassignment tendency outcome measures. For resolution time, the additive effect size is “large”, with R^2 values increasing from 0.203 to 0.380. The effect sizes for reopening and reassigning are likely artificially inflated given the negligible AIC delta statistic, suggesting that contextually they should be interpreted as “small”. For fix and patch emergence tendencies, the additive effect size is “medium to large” and “very large” respectively, with R^2 values increasing from 0.157 to 0.336 and 0.092 to 0.401 respectively. Whereas a “very large” effect size may elicit spuriousness concerns in some cases, in the present case, there is evidence to suggest that it is accurate. The delta AIC statistic for the patch emergence tendencies full model above and beyond the control model is very large relative to the degrees of freedom, suggesting a strong, independent variable effect on the model. In addition, all the independent variables are highly significant ($p < 0.001$), suggesting strong contributions to an overall model. Lastly, the theoretical mechanism by which the deliberately operationalized manual comment count variable is believed to contribute to increased patch emergence tendencies specifically is well described, supporting that the observed results are not likely spurious.

As is the case for the hypothesis one aggregate solution knowledge verifier role models, the very limited sample size ($n <= 42$) for this role precluded significant results. Whereas the models report significance in some cases, examination of the comparative Chi-squared and comparative AIC delta statistics suggests spuriousness, with the conservative interpretation being insufficient statistical power for meaningful analysis of this role in the codifiability regression models. Therefore, the choice is made to interpret the results of the aggregate solution knowledge verifier role models, as a whole, as not significant.

In summary, the organisation level codifiability effects include a separate additive effect of the negative influence of long descriptions on desired outcomes above and beyond the individual level effect; organisation specific tacit knowledge that makes less readable descriptions more useful for positive outcomes; and, organisation level utilization of greater number of attachments to problem knowledge further improving outcomes. Yet, organisations appear to use duplicate problem knowledge less effectively than individuals, not achieving outcome benefits as a result.

Like at the individual level, aggregate roles significantly influence the outcome effects. In many cases, the aggregate problem knowledge producer effects differ from the aggregate solution knowledge producer effects such that positive effects for the former negatively affect the latter and vice-versa. The takeaway for the codifiability independent variable effects remains that careful consideration of both the levels and roles involved is necessary to improve desired outcomes.

Hypothesis three: Dominant knowledge paradigm

The third hypothesis postulates that, "*The similarity of the problem knowledge revealed to the meta-organisation to the dominant knowledge paradigm in the meta-organisation is positively correlated with solution knowledge emergence*." This hypothesis was tested by analyzing data at the problem, individual, and organisation levels of analysis; cross-level nesting effects between individuals and the organisations of which they are a member were also assessed. The results for each level are discussed in turn in the following sections.

Problem level of analysis results

At the problem level of analysis there are seven measures that represent the conceptualization of the dependent outcome of interest, solution knowledge emergence, as depicted in Figure 11, and as operationalized into the variables described in Table 51. Five measures represent the conceptualization of the antecedent of interest in hypothesis three, dominant knowledge paradigm, as depicted in Figure 15, and as operationalized into the variables described in Table 55. The results of the analyses are depicted in the regression model output summaries in Appendix D: Regression models.

Measure one: Platform type

The first measure of dominant knowledge paradigm at the problem level of analysis is the platform type of the problem knowledge revealed to the meta-organisation. It is hypothesized that problem knowledge related to more popular platform types is correlated with better solution knowledge emergence outcomes.

Examination of the ANCOVA regression model summaries suggests that platform type is significantly correlated with resolution time ($p < 0.001$), assignment time ($p < 0.001$), development time ($p < 0.001$), fix and patch emergence tendencies ($p < 0.001$), reopening and reassigning tendencies ($p < 0.001$), and confirmation tendencies ($p < 0.001$). In short, platform type very strongly influences all dependent outcomes of interest at the problem level of analysis.

Examination of the dummy regression model summaries reveals that, with the exception of platforms “ARM” and “SGI”, all platforms are correlated with overall reductions in resolution time relative to the “all” category for problems that are not specific to a given platform. In terms of development time, problems specific to platforms “DEC” ($p < 0.01$), “HP”, ($p < 0.05$), “x86” ($p < 0.001$), and “x86_64” ($p < 0.001$) perform better than general problems, whereas problems specific to platforms “ARM” ($p < 0.001$) and “Xscale” ($p < 0.001$) perform worse than general problems. Yet, contextually, these effects are weak as the timing regression models only achieve marginal effect size increases above and beyond the control models.

In terms of fix and patch emergence tendencies, platforms “ARM” ($p < 0.001$), “other” ($p < 0.001$), “PowerPC” ($p < 0.001$), “Sun” ($p < 0.001$), “x86” ($p < 0.001$), and “x86_64” ($p < 0.001$) all perform markedly worse than general problems that are not platform specific. The “ARM” platform is correlated ($p < 0.01$) with increased reopening tendencies, and the “x86” platform is correlated ($p < 0.001$) with decreased reassignment tendencies. With the exception of platform “ARM”, which is correlated ($p < 0.001$) with increased confirmation tendencies, and platform “Xscale” which does not achieve significance, all other platforms are correlated ($p < 0.01$ to $p < 0.001$) with worse confirmation tendencies than platform non-specific problems. The timing threshold regression models suggest that platforms “x86”, “ARM”, and “PowerPC” represent the “average” central-tendency of resolution times, with other platforms distributed around them based on their popularity. The distribution of platform “ARM” resolution timings suggests that its correlation with longer resolution time overall may be due to skew in the data around reduced propensity of "extremely fast" resolution ($p < 0.01$) whereas there is positive correlation with "fast" resolution ($p < 0.01$), "average" resolution ($p < 0.001$), and negative correlation with "very slow" ($p < 0.05$) and "extremely slow" ($p < 0.001$) resolution timings. The overall picture suggests that platform "ARM" is a popular category (related to mobile development on Android cellular phone devices) that has a broad distribution of resolution times.

In summary, platform non-specific problem knowledge has better patch emergence tendencies and confirmation tendencies. This result is as hypothesized, as problems that are broadly relevant across platform types (type "all") represent a knowledge paradigm that is relevant to a broader range of meta-organisation interests, by definition. Whereas problems associated with some platforms perform better than other platforms, the primary takeaway is that problem knowledge that is as broadly relevant as possible to the meta-organisation improves solution knowledge emergence tendencies.

Measure two: Classification type

The second measure of dominant knowledge paradigm is the classification type of the problem knowledge revealed to the meta-organisation. It is hypothesized that problem knowledge related to more popular classification types is correlated with better solution knowledge emergence outcomes.

Examination of the ANCOVA regression model summaries suggests that classification type is significantly correlated with resolution time ($p < 0.001$), assignment time ($p < 0.001$), development time ($p < 0.001$), fix and patch emergence tendencies ($p < 0.001$), reopening and reassigning tendencies ($p < 0.001$), and confirmation tendencies ($p < 0.001$). As was the case with platform, classification of problem knowledge has a strong effect on the dependent outcomes of interest.

Examination of the dummy variable regression model summaries reveals that relative to the reference classification category “client software”, classification “components” is correlated ($p < 0.001$) with slower resolution time, classification “sever software” is correlated ($p < 0.001$) with faster resolution times, and classifications “graveyard” ($p < 0.01$) and “other” ($p < 0.001$) are correlated with faster resolution times. In terms of development times, classifications “components” ($p < 0.05$) and “graveyard” ($p < 0.01$) are correlated with slower development time, and classifications “server software” and “other” are correlated ($p < 0.001$) with faster development time relative to the “client software” classification. However, once again, the weak incremental effects observed for the full regression models for timing cast doubt on the meaningfulness of the size of the contribution of these variables.

In terms of fix emergence tendencies, all classifications are correlated ($p < 0.01$ to $p < 0.001$) with better fix emergence tendencies than “client software”, yet the patch emergence tendencies are mixed. Classification “component” is correlated ($p < 0.001$) with better patch emergence tendencies, whereas classifications “graveyard”, “other”, and “server software” are correlated ($p < 0.001$) with worse patch emergence tendencies. Whereas this result would be particularly exciting as it appears to suggest an unusual split amongst outcome types, where fix and patch emergence tendencies differ for the same classification, examination of the regression model comparative AIC delta statistics, as well as the additive effect sizes suggests caution in the interpretation that meta-organisation members can influence the type of solution knowledge that emerges using classifications. A more probable explanation for the results lies in the popularity of the reference "client software" category. Given that more problem knowledge is submitted for the "client software" category than the other categories, the hypothesis suggests that it should have the best relative outcome. The results for fix emergence appear to suggest the opposite. Yet, examination of the additive effect size suggests that the independent variables' contribution to the model is "small" above and beyond the control variable effects. The observed differences, even those that are individually significant in dummy variable coefficients, are unlikely to have significant overall effects. The reference category, in this case "client software", is representative of the average model and divergences are not significant. Contextually, that lack of significant deviance of other categories fits with the dominant knowledge paradigm hypothesis, albeit providing only weak support. By contrast, the additive effect size for the patch emergence regression model is "medium", suggesting that the differences from the reference categories are more meaningful in this model. The interpretation, therefore, is that classifications "components" and "client software" have better patch emergence tendencies than classifications "graveyard", "other", and "server software". Given the nature of the worse performing classifications, this result also fits with the dominant knowledge paradigm hypothesis, providing marginally stronger support than for fix emergence tendencies.

In terms of confirmation tendencies, the results suggest that all classifications perform better ($p < 0.001$) than the "client software" classification type. A possible explanation is that given that "client software" is the most popular type, there are too many sets of problem knowledge for triagers to confirm. By contrast, triagers engaging with less popular types have fewer problems to go through, increasing the likelihood of confirmation. This result raises questions about the usefulness of the "confirmation" parameter in meta-organisations in terms of its ability to improve the knowledge production process. "Confirmation" may simply be a measure of popularity, at least in aggregation, rather than a measure of progress through the knowledge production process.

In summary, whereas classification undoubtedly has an influence on outcomes of interest, the popularity of the reference category “client software” dominates the observed effects, resulting in negligible significance for other classification categories. These results suggest a dominant knowledge paradigm effect surrounding the popular “client software” category overall, at least in the case of patching and confirmation tendencies.

Measure three: Operating system type

The third measure of dominant knowledge paradigm is the operating system type of the problem knowledge revealed to the meta-organisation. It is hypothesized that problem knowledge related to more popular operating system types is correlated with better solution knowledge emergence outcomes.

Examination of the ANCOVA regression model summaries suggests that operating system type is significantly associated with assignment time ($p < 0.001$), development time ($p < 0.05$), fix and patch emergence tendencies ($p < 0.001$), and confirmation tendencies ($p < 0.001$). Of particular note, operating system type does not appear to affect resolution time, despite affecting assignment and development times. This lack of significant effect, at first glance, stands out as it appears to suggest a difference between the knowledge types of “platform”, “classification”, “operating system”, “product” and “component” used in the meta-organisation. Were the types meaningless, similar effects would be expected. Yet, this result is likely insignificant overall given the marginal incremental effect size observed for the independent variables on the timing outcomes variables.

Given the large (>50) number of operating system category types, as discussed in Chapter Four: Research Method, this variable is operationalized as numeric because dummy variable models are not possible. Therefore, the interpretation of this result is that operating systems have significantly different effects on some outcomes of interest, namely confirmation and patch emergence tendencies (the overall incremental effects for the models of the other outcome variables being of marginal significance), as hypothesized, but it would be overreach to interpret the results more specifically because the modelling precludes identification of which operating system types are correlated with positive outcomes and which are correlated with negative outcomes. A reasonable takeaway is that operating system types are relevant when producing problem knowledge and actors should carefully consider their selection. In addition, further research that builds upon the present study that aims to capture large scale effects, which focuses more specifically on the relative strengths and weaknesses of operating system categories with a suitable dataset, may be fruitful in illuminating more specific effects.

Measure four: Product type

The fourth measure of dominant knowledge paradigm is the product type of the problem knowledge revealed to the meta-organisation. It is hypothesized that problem knowledge related to more popular product types is correlated with better solution knowledge emergence outcomes.

Examination of the ANCOVA regression model summaries suggests that product type is significantly correlated with resolution time ($p < 0.001$), assignment time ($p < 0.001$), development time ($p < 0.001$), reopening and reassigning tendencies, ($p < 0.001$), fix and patch emergence tendencies ($p < 0.001$), and confirmation tendencies ($p < 0.001$). As is the case with operating system category types, given the large (>85) number of product category types, as discussed in Chapter Four: Research Method, this variable is operationalized as numeric because dummy variable models are not possible. Therefore, once again, the conservative interpretation of the observed results is that there is a significant effect of product type on patch emergence and confirmation tendencies (the overall incremental effects for the models of the other outcome variables being of marginal significance), but which specific product types should be sought for optimal solution knowledge emergence is left for future research beyond the scope of the present large-scale effect focused study.

Measure five: Component type

The fifth measure of dominant knowledge paradigm is the component type of the problem knowledge revealed to the meta-organisation. It is hypothesized that problem knowledge related to more popular component types is correlated with better solution knowledge emergence outcomes.

Examination of the ANCOVA regression model summaries reveals that component type, like product type, is significantly correlated with resolution time ($p < 0.001$), assignment time ($p < 0.001$), development time ($p < 0.001$), fix ($p < 0.01$) and patch ($p < 0.001$) emergence tendencies, and confirmation tendencies ($p < 0.001$). Whereas the ANCOVA regression models also suggest significant reopening and reassigning tendencies, examination of the heteroskedasticity corrected model summaries reveals that the significance disappears in the corrected models, suggesting that the effects are spurious due to non-linearities in the data.

As is the case with operating system and product category types, given the large (>1250) number of component category types, as discussed in Chapter Four: Research Method, this variable is operationalized as numeric because dummy variable models are not possible.

Therefore, once again, the conservative interpretation of the observed results is that there is a significant effect of component type on the patching and confirmation tendencies (the overall incremental effects for models of the other outcome variables being of marginal significance), but which specific component types should be sought for optimal solution knowledge emergence is left for future research beyond the scope of the present large-scale effect focused study.

Summary of dependent variable effect results at problem level

Comparison of the control and full regression models for the effect of dominant knowledge paradigm measures on the dependent outcomes of interest reveals an overall picture of effect size at the problem level of analysis.

For resolution time, assignment time, and development time, while the model F statistic and model Chi-squared statistic suggest superiority of the full models over the control models, the additive effect size "very small" for resolution time and assignment time, and "small" for development time, with R^2 values increasing from 0.209 to 0.214, 0.088 to 0.091, and 0.373 to 0.382 respectively. Given the very large number of observations and degrees of freedom, the contextual interpretation is that the dominant knowledge paradigm independent variables have insignificant to marginal effects on time-based outcomes above and beyond the control values. The threshold timing models also all suffer from weak incremental support, further supporting the notion that timing effects appear insignificant overall at the problem level of analysis.

For fix and patch emergence tendencies, the comparative AIC delta statistics suggest superiority of full models over control models, with fix tendency additive effect size of "small" and patch tendency additive effect size of "medium", and R^2 values increasing from 0.566 to

0.582 and 0.364 to 0.465 respectively. The contextual interpretation is that the dominant knowledge paradigm independent variables primarily affect patch emergence tendencies.

Whereas a strict conservative interpretation would be that the model design was not ideal for testing dominant knowledge paradigm effects at the problem level of analysis, only slight interpretive liberties need be taken to frame these results as weakly supporting the notion that the reference categories are dominantly representative of the overall outcome effects. This interpretation is supported by the incremental effect size of "medium" observed in the results of the confirmation tendencies outcome. "Confirmation" is a function of the triage activity that gets overwhelmed based on popular categories, as was observed in the hypothesis one: absorptive capacity results. A reasonable takeaway is that popularity of categories matters for some solution knowledge emergence outcome measures and not others, with patch emergence tendencies and confirmation tendencies being the outcomes most strongly affected by dominant knowledge paradigm factors. In addition to examining the types with many categories more closely, future research that considers the temporal element of knowledge type popularity may meaningfully refine the observed results. Given that "creation year" is highly significant ($p < 0.001$) as a control variable in all models, it stands to reason that type popularity would wax and wane over time. A research program that sought to disambiguate the temporal component of popularity from other problem knowledge factors could prove fruitful in this respect.

In summary, problem level dominant knowledge paradigm effects primarily lie in patch emergence tendencies and confirmation tendencies measures of solution knowledge emergence outcomes, with more popular knowledge paradigms correlated with improved patch emergence tendencies and reduced confirmation tendencies. The individual level of analysis results shed light on the role specific effects.

Individual level of analysis results

At the individual level of analysis there are seven measures that represent the conceptualization of the dependent outcome of interest, solution knowledge emergence, as depicted in Figure 19, and as operationalized into the variables described in Table 79 and Table 87. Three measures represent the conceptualization of the antecedent of interest in hypothesis three, dominant knowledge paradigm, as depicted in Figure 22, and as operationalized into the variables described in Table 91. The results of the analyses are depicted in the regression model output summaries in Appendix D: Regression models.

As discussed in Chapter Four: Research Method, regression models are separated according to the roles in which individuals engage when participating in the knowledge production process: problem knowledge producer (reporter), solution knowledge producer (assigned_to), and solution knowledge verifier (QA_contact).

Measure one: Percent of actions in each role in each platform type

The first measure of dominant knowledge paradigm is the percent of actions in each role on problems with each platform type. A tendency to act more frequently on problems with more popular platforms is theorized to be correlated with positive solution knowledge emergence outcomes due to the dominance of the knowledge paradigm of the popular platform type. As discussed in Chapter Four: Research Method, in order to allow for more specific localization of platform-type specific effects, the platforms are consolidated into five types: “All”, “PowerPC”, “x86_64”, “x86”, and “other”, with “other” held as the reference category by virtue of holding the balance of the total percentage of role specific activities for each individual.

Examination of the regression model summaries reveals that higher percentage of actions in the problem knowledge producer role on problems with platforms “all” and “x86_64” is correlated with reduced resolution time ($p < 0.001$ and $p < 0.001$ respectively), and higher percentage of actions on problems with platform “PowerPC” is correlated with increased resolution time ($p < 0.001$). The percent of actions on problems with platform “x86_64” is correlated with reduced fix and patch emergence tendencies ($p < 0.05$) and reduced reassignment tendencies ($p < 0.001$). The percent of actions on problems with platform “PowerPC” is correlated with increased reassignment tendencies ($p < 0.001$) and reduced fix emergence tendencies ($p < 0.001$).

These results are largely the opposite of hypothesized. Whereas reporter role actions on problems with platform “x86_64” are correlated with faster resolution time, the correlation with reduced fix and patch emergence tendencies suggest that it is faster “discarding” rather than “fixing”, which is undesirable. Actions on problems with platform “PowerPC” are correlated with increased resolution time as well as increased reassignment tendencies and reduced fix emergence tendencies, all undesirable. Whereas at the individual level, platform “all” appears to enjoy relative outcome benefits, aside from reduced resolution time, it is not clear that there is increased solution knowledge emergence. Taken collectively, the problem knowledge producer role platform specific effects appear to be marginally the opposite of expected, with the more popular platforms suffering relatively worse outcomes. A possible explanation is that with popularity comes increased frivolousness of problem knowledge, resulting in an overall reduction of problem knowledge quality in the case of the problem knowledge producer role.

For the solution knowledge producer role, the outcome effects are largely the inverse of those observed for the problem knowledge producer role. Higher percentage of developer role actions on problems with more popular platforms such as "x86_64" are correlated with reduced resolution time ($p < 0.001$), and increased fix and patch emergence tendencies ($p < 0.001$). Likewise, developer actions on problems classified as relevant to "all" platforms are correlated with reduced resolution time ($p < 0.05$), and increased fix ($p < 0.001$) and patch ($p < 0.01$) emergence tendencies. Moderately popular platform "x86" shows moderate effects, with percentage of developer actions on such problems correlated with reduced resolution time ($p < 0.01$) and increased fix emergence tendencies ($p < 0.05$).

The solution knowledge producer role results are as hypothesized, lending support to the notion that dominant knowledge paradigms, as represented by the platform type, in the meta-organisation, influence solution knowledge emergence. Notably, however, it is the developer role actions that are correlated with the outcome effects, not those of the reporter role. This split in role-based results matches the results for previous measures. It lends support to the notion in the literature that different types of participants in the meta-organisation have different perspectives on what type of knowledge is valuable, as well as different degrees of power in the meta-organisation to promote their knowledge value perspectives (c.f. von Krogh, Spaeth, & Lakhani, 2003; Fitzgerald, 2006; Dahlander & Frederiksen, 2012). Further, this difference is measurable in the dominant knowledge paradigm independent variables that track the developer role actions.

Measure two: Percent of actions in each role in each classification type

The second measure of dominant knowledge paradigm is the percent of actions in each role on problems with each classification type. A tendency to act more frequently on problems with more popular classification types is theorized to be correlated with positive solution knowledge emergence outcomes due to the dominance of the knowledge paradigm of the popular classification types. As discussed in Chapter Four: Research Method, in order to allow for more specific localization of classification-type specific effects, the classifications are consolidated into four types: "client software", "components", "server software", and "other", with "other" held as the reference category by virtue of holding the balance of the total percentage of role specific activities for each individual.

Examination of the regression model summaries reveals that higher percentage of actions in the problem knowledge producer role on problems with classifications "client software" and "components" is correlated with increased resolution time ($p < 0.001$), decreased fix emergence ($p < 0.001$), and increased patch emergence ($p < 0.001$). Percentage of actions in reporter role on problems with classification "server software" is correlated with decreased fix emergence ($p < 0.001$), and increased patch emergence ($p < 0.001$). These results largely correspond with those observed at the problem level of analysis suggesting a split in the type of problem knowledge emergence that is associated with certain classification types. The relatively stronger effect sizes at the individual level of analysis than those observed at the problem level of analysis suggest that the effect is localized in the problem knowledge producer role at the individual level. A potential explanation for the split is that problem knowledge producers who are too narrowly focused in terms of classification tend to submit a laundry list of problems, which, on average, reduces their relative fix percentages but increases their relative patch emergence because they have deeper problem knowledge in that classification relative to those who span classifications with their problem submissions. Examination of the mixed-effects nested models suggest that the organisation level effect is additive, both reducing the fix rates and increasing the patch rates observed in the individual level results. Therefore, there may also be an organisational breadth vs. depth effect that manifests in classification-related effects.

For the solution knowledge producer role, percent of actions on problems with classifications “client software” and “components” are correlated with increased fix and patch emergence ($p < 0.001$), while percent actions on problems with classification “server software” is correlated with increased patch emergence ($p < 0.001$). Examination of the mixed-effects models suggests that in the case of the “components” classification, the increased fix tendency effect may be due to organisational embeddedness rather than individual level action. Taken collectively, there is undoubtedly solution knowledge producer role specific effects at the individual level based on classification and these effects are different from the reporter role in that they are correlated with better fix emergence tendencies whereas reporter role effects are correlated with worse fix emergence tendencies. These results suggest further validation of the developer role having greater power than the reporter role in deciding the dominant knowledge paradigm effects in the meta-organisation.

Measure three: Percent of actions in each role in each operating system type

The third measure of dominant knowledge paradigm is the percent of actions in each role on problems with each operating system type. A tendency to act more frequently on problems with more popular operating system types is theorized to be correlated with positive solution knowledge emergence outcomes due to the dominance of the knowledge paradigm of the popular operating system types. As discussed in Chapter Four: Research Method, in order to allow for more specific localization of operating system-type specific effects, the operating systems are consolidated into nine types: “all”, “Android”, “Linux”, Mac_pc”, “Windows_pc”, “Windows_mobile”, “iOS_mobile”, “other_mobile”, and “other”, with “other” held as the reference category by virtue of holding the balance of the total percentage of role specific activities for each individual.

Examination of the regression model summaries reveals that higher percentage of actions in the problem knowledge producer role on problems with operating system “all”, is correlated with increased fix and patch emergence ($p < 0.001$), albeit with slower resolution time ($p < 0.001$); the percentage of reporter role actions on problems with operating system “Android” is correlated with increased patch emergence ($p < 0.001$) and reduced resolution time ($p < 0.001$); operating system “Linux”: increased resolution time ($p < 0.01$) and decreased fix emergence tendencies ($p < 0.001$); operating system “Mac_pc”: decreased resolution time ($p < 0.001$), decreased reopening and reassigning tendencies ($p < 0.001$), but no significant fix or patch emergence effects; “Windows_pc”: decreased fix ($p < 0.001$) and patch ($p < 0.05$) effects; “iOS_mobile”, markedly reduced fix emergence tendencies ($p < 0.01$), with “faster rejection” resolution time ($p < 0.05$); and, “other_mobile” correlated with faster resolution ($p < 0.001$) and increased fix and patch emergence tendencies ($p < 0.001$). The trend for the reporter role percentages appears to be largely split along the mobile vs. personal computer operating system line, with actions on mobile operating systems correlated generally with better solution knowledge emergence than actions on personal computer operating systems.

For the solution knowledge producer role, the result split is largely the same, with "Windows_pc" correlated with worse fix emergence tendencies ($p < 0.001$), and "all" ($p < 0.01$), "Android" ($p < 0.01$), and "other_mobile" ($p < 0.05$) all correlated with better patch emergence tendencies. Yet, there are distinctions from the reporter role as well. Contrary to the reporter role results, "Mac_pc" is correlated with improved fix outcome tendencies ($p < 0.01$), and "Linux" is correlated with improved patch outcome tendencies ($p < 0.01$). This distinction may be related to the computer choice of open source developers. Many developers prefer "Mac" laptops on which they can run "Linux" operating systems in parallel with the Mac OS (Asay, 2007). As a result, it stands to reason that they would prioritize problems that are associated with the operating systems they use as the problems would be more relevant to them. This result further supports the power of the developer in determining the effects of dominant knowledge paradigms in the meta-organisation.

Effects of individual nestedness in organisations

Examination of the mixed-effects regression model summaries in reveals that there are significant organisational effects that affect the individual level results. For the problem knowledge producer and solution knowledge producer roles, the AIC and BIC delta statistics reveal that the organisation random effect models are superior in all cases. The effect sizes of organisation nestedness isolated from the fixed individual level effects are "medium" to "medium-to-large" for the reporter role and "medium" to "very large" for the developer role.

Despite the significant organisational effects, the significance of the effects of the individual level independent variables remains largely unchanged, with the exception of the fix emergence impact of classification "component" for the developer role being more an organisation level effect than individual level. In all other cases, the marginal organisation embeddedness effects are insufficient to change the significance of the individual level effects, even in cases where organisations pull in different directions. This result suggests that individuals may have more control over the dominant knowledge paradigm than organisations on the whole.

It is interesting to note that the negative fix emergence tendency effect of developer actions on operating system "Windows_pc" remains virtually unchanged when considering organisational embeddedness. This result lends support to the notion that individual developers are very "anti-Windows", independent of organisation-level policies, an issue that has long been discussed in the open source literature (c.f. Raymond, 1999a; Lerner & Tirole, 2001, 2002; von Hippel, 2001; Bonaccorsi & Rossi, 2003; Lakhani & von Hippel, 2003; von Hippel & von Krogh, 2003; West, 2003; Söderberg, 2015). It further suggests that an "anti" dominant knowledge paradigm effect is pervasive in the Mozilla meta-organisation, centered around developer actions rather than participant organisations. A takeaway for organisations seeking to get involved in meta-organisations is that they should be aware of their individual members' biases if they do not align with those of the organisation as the desired outcome effects my not otherwise align.

Summary of dependent variable effect results at individual level

Comparison of the control and full regression models for the effect of dominant knowledge paradigm independent variables on the dependent outcomes of interest reveals an overall picture of effect size at the individual level of analysis. For the problem knowledge producer role, the comparative Chi-squared statistic and comparative AIC delta statistic reveal that the models that include the dominant knowledge independent variables are all superior to the control-only models. The additive effect size of the independent variables above and beyond the control variables is “small-to-medium” for days to resolution dependent variable, “small” and “medium” for reopening and reassigning tendencies, respectively, and “medium” for both fix and patch emergence tendencies.

For the solution knowledge producer role, the comparative Chi-squared and comparative AIC delta statistics also suggest that the full models are superior to the control models in all cases except reopening tendencies. The additive effect size of the independent variables above and beyond the control variables is “medium” for days to resolution, “medium-to-large” for reopening tendencies, and “medium” and “very large” for fix and patch emergence tendencies respectively.

Whereas the solution knowledge verifier role results appear to show sizeable effects, examination of the standard error for many of the coefficients in the models suggests that the observed effect sizes are spurious and the result of suboptimal model fitting. Overall, the solution knowledge verifier dominant knowledge paradigm effects appear to be of negligible significance on outcomes of interest.

In summary, the individual level dominant paradigm effects are best characterised as a balance between popularity of platforms, classifications, and operating systems drawing more problems from the problem knowledge producers and the power of solution knowledge producers in deciding what types of knowledge to focus on. Individuals engaging in the reporter role that focus primarily on popular platforms and classifications will tend to have worse solution knowledge emergence. By contrast, those individuals engaging in the developer role that focus primarily on popular platforms and classifications will tend to have better solution knowledge emergence. For operating systems, the dominant knowledge paradigm appears to be mobile operating systems, which generally are correlated with better solution knowledge emergence across roles. Yet, developers still exert disproportionate effects on the outcomes, with developer actions on Linux and MacOS personal computer operating systems resulting in better solution knowledge emergence. This latter effect is likely due to the personal investment of developers in these specific operating systems based on their own usage preferences, independent of the rest of the meta-organisation. The mixed-effects models suggest strong individual level effects independent of organisation level effects on most outcomes of interest, though organisation level effects undoubtedly exist in many cases, as examined in the following section.

Organisation level of analysis results

At the organisation level of analysis there are seven measures that represent the conceptualization of the dependent outcome of interest, solution knowledge emergence, as depicted in Figure 26 and as operationalized into the variables described in Table 105. Three measures represent the conceptualization of the antecedent of interest in hypothesis three, dominant knowledge paradigm, as depicted in Figure 29 and as operationalized into the variables described in Table 109. The results of the analyses are depicted in the regression model output summaries in Appendix D: Regression models.

As discussed in Chapter Four: Research Method, regression models are separated according to the aggregate roles in which members of organisations engage when participating in the knowledge production process: aggregate problem knowledge producer (reporter) role, aggregate solution knowledge producer (assigned_to) role, and aggregate solution knowledge verifier (QA_contact) role.

Measure one: Percent of actions in each aggregate role in each platform type

The first measure of dominant knowledge paradigm is the percent of actions in each aggregate role on problems with each platform type. A tendency of organisation members to act more frequently on problems with more popular platforms is theorized to be correlated with positive solution knowledge emergence outcomes due to the dominance of the knowledge paradigm of the popular platform types. As discussed in Chapter Four: Research Method, in order to allow for more specific localization of platform-type specific effects, the platforms are consolidated into five types: “All”, “PowerPC”, “x86_64”, “x86”, and “other”, with “other” held as the reference category by virtue of holding the balance of the total percentage of aggregate role specific activities for each organisation.

Examination of the regression model summaries reveals that percentage of action in the aggregate problem knowledge producer role on problems with platform “x86_64” is correlated with reduced resolution time ($p < 0.05$) and actions on problems with platform “all” are correlated with increased fix and patch emergence ($p < 0.05$). No other platform specific effects achieve significance for the aggregate problem knowledge producer role at the organisation level of analysis

For the aggregate solution knowledge producer role, percentage of actions on problems with platform “x86_64” are also correlated with faster resolution times ($p < 0.05$). The other models for the aggregate developer role and all the models for the aggregate solution knowledge verifier role do not achieve significance.

Collectively these results suggest that the organisation level platform effects are weak, with the small effects being primarily correlated with platform “x86_64” which is one of the most popular platforms. Whereas the individual level effects suggest that platform “x86_64” are associated with reduced resolution times, given that it is also associated with reduced fix and patch tendencies, popularity is negative in that case, with faster “discarding” rather than “resolving”. The present organisation level models along with the mixed-effects models suggest that the faster resolving effect does have an organisational component, with aggregate reporter role actions also increasing speed of “discarding” of problems, although the reduced fix and patch emergence effects appear to be limited to the individual level reporter role.

Interestingly, whereas for the individual level developer role, more work on problems with popular platforms such as “x86_64” was correlated with better fix and patch emergence outcomes, no such effect is significant at the organisation level of analysis. The implication is that it is individual developers who exert the power associated with dominant knowledge paradigm effects, not the organisations of which they are a member. This result makes significant contributions to theory as it localizes the level of the power that affects solution knowledge emergence. It also makes significant contributions to practice as it further suggests that organisations must be careful that their members’ knowledge paradigms align with those that are of benefit to the organisation, lest the misalignment hamper emergence of solution knowledge that is desirable for the organisation.

Measure two: Percent of actions in each aggregate role in each classification type

The second measure of dominant knowledge paradigm is the percent of actions in each aggregate role on problems with each classification type. A tendency of organisation members to act more frequently on problems with more popular classification types is theorized to be correlated with positive solution knowledge emergence outcomes due to the dominance of the knowledge paradigm of the popular classification types. As discussed in Chapter Four: Research Method, in order to allow for more specific localization of classification-type specific effects, the classifications are consolidated into four types: “client software”, “components”, “server software”, and “other”, with “other” held as the reference category by virtue of holding the balance of the total percentage of aggregate role specific activities for each organisation.

Examination of the regression model summaries reveals that higher percentage of actions in the aggregate problem knowledge producer role on problems with all three classifications, “client software”, “components”, and “server software” is correlated with reduced fix emergence tendencies ($p < 0.001$, $p < 0.01$, and $p < 0.001$ respectively) and increased patch emergence tendencies ($p < 0.001$ for all three). This result matches the unexpected results observed at the problem and individual levels of analysis, suggesting that organisation members engaging in the aggregate reporter role may suffer from breadth vs. depth trade-offs when acting on problems with different classifications in the meta-organisation. Whereas breadth improves fix emergence tendencies, patch emergence is less likely because the problem and solution knowledge are less specialized. By contrast, when depth increases patch emergence tendencies for specific problems, the narrow focus of the problems reduces overall fix emergence tendencies.

As was observed at the individual level of analysis, for the aggregate solution knowledge producer role, focus on depth does pay off. Classifications “components” and “server software” are correlated with both better fix ($p < 0.05$) and better patch ($p < 0.001$) emergence tendencies and better patch ($p < 0.001$) outcome tendencies are observed for classification “client software”.

Examination of the mixed-effect models suggests that the organisation level effects are additive to the individual level effects in most cases. The "client software" effects appear to be more individual level than organisation level, which stands to reason as client software is often used by individuals and therefore more likely to draw their personal focus whereas server software is often the focus of organisations who use it on their internal infrastructure. This effect complements the individual level developer preference effect by highlighting an organisation level "server software" preference effect. The takeaway for organisations is that if its developers focus in depth on classification "server software", solution knowledge emergence can align with the organisation's interests.

Measure three: Percent of actions in each aggregate role in each operating system type

The third measure of dominant knowledge paradigm is the percent of actions in each aggregate role on problems with each operating system type. A tendency to act more frequently on problems with more popular operating system types is theorized to be correlated with positive solution knowledge emergence outcomes due to the dominance of the knowledge paradigm of the popular operating system types. As discussed in Chapter Four: Research Method, in order to allow for more specific localization of operating system-type specific effects, the operating systems are consolidated into nine types: "all", "Android", "Linux", Mac_pc", "Windows_pc", "Windows_mobile", "iOS_mobile", "other_mobile", and "other", with "other" held as the reference category by virtue of holding the balance of the total percentage of aggregate role specific activities for each organisation.

Examination of the regression model summaries reveals that higher percentage of actions in the aggregate problem knowledge producer role on problems with personal computer operating systems is correlated with worse outcomes, with “Windows_pc” correlated with reduced fix ($p < 0.001$) and patch ($p < 0.01$) emergence tendencies and “Linux” associated with reduced fix emergence tendencies ($p < 0.05$). By contrast, mobile operating systems “Android” and “other mobile” are correlated with increased resolution time ($p < 0.001$). Whereas this result is similar to that observed for the individual level reporter role, the effects are not nearly as strong at the organisation level. Examination of the mixed-effect models suggests further support that the operating system dominant knowledge paradigm effects are predominantly individual level for the reporter role.

For the aggregate solution knowledge producer role, the results are consistent with those observed at the individual level of analysis suggesting a strong negative correlation between focus on problems with operating system “Windows_pc” and fix ($p < 0.05$) and patch ($p < 0.001$) emergence. Whereas there is a significant organisational effect, the mixed-effects model suggest that the effect is primarily an individual level effect, with the nature of organisation level measures consisting of individual organisation member aggregation most likely being responsible for the organisation level observed effects in this respect. Th practical takeaway is that organisations seeking solution knowledge related to the Windows desktop operating system must be exceptionally careful to ensure that their members’ biases do not hinder solution knowledge emergence.

Summary of dependent variable effect results at organisation level

Comparison of the control and full regression models for the effect of dominant knowledge paradigm independent variables on the dependent outcomes of interest reveals an overall picture of effect size at the organisation level of analysis. For the aggregate problem knowledge producer role, the comparative Chi-squared statistic and comparative AIC delta statistic reveal that the models that include the dominant knowledge independent variables are all superior to the control-only models, albeit only marginally so for the reopening and reassigning tendencies outcomes. The additive effect size of the independent variables above and beyond the control variables is “small-to-medium” for days to resolution dependent variable, “small” and “small-to-medium” for reopening and reassigning tendencies, respectively, and “medium” for both fix and patch emergence tendencies.

For the aggregate solution knowledge producer role, the comparative Chi-squared and comparative AIC delta statistics also suggest that the full models are superior to the control models in the case of resolution time, fix, and patch emergence tendency outcome variables. The reassigning and reopening tendencies full models fail to achieve superiority over the control only models. The additive effect size of dominant knowledge paradigm independent variables is “large” for resolution time and fix emergence tendencies, and “very large” for patch emergence tendencies. Whereas the “very large” effect size for patch emergence tendencies would be cause for concern of spuriousness in some circumstances, examination of the mixed-effects regression models suggests a sizeable organisation level effect on patch emergence tendencies, specifically attributable to the “server_software” classification and “x86_64” platform, both of which have strong theoretical reasons for having organisation level specific effects above and beyond individual level effects. As such, there is evidence to believe that this effect size is not spurious and is representative of the actual organisation level dominant knowledge paradigm factor effects.

As is the case at the individual level of analysis, the further reduced number of observations (n <= 42) for the aggregate solution knowledge verifier role precludes any significant independent variable effects being observed at the organisation level of analysis.

In summary, the organisation level dominant knowledge paradigm effects of platform are relatively weak compared to the individual level of analysis, with only small effects for the popular "x86_64" platform. The classification effects are a trade-off between breadth and depth where classification breadth increases fix tendencies and classification depth increases patch tendencies for the aggregate problem knowledge producer role. For the aggregate solution knowledge producer role, depth engagement in classification "server software" can pay off both in terms of positive fix and patch emergence tendencies. The operating system effects mirror those of the individual level suggesting the mobile operating systems are the dominant knowledge paradigm and promote better solution knowledge emergence outcomes than personal computer operating systems, albeit with less strength at the organisation level of analysis.

A takeaway for organisations is that, particularly in the case of types of knowledge that are associated with strong opinions in the meta-organisation, such as Microsoft's Windows desktop operating system in an open source meta-organisation, but also for less contentious types such as mobile or desktop, its members' actions must be aligned with the type of solution knowledge that the organisation is seeking, independent of the properties of the specific problem knowledge revealed to the meta-organisation. In this respect, individuals, and particularly developers, hold disproportionate power on the outcome effects, a factor which is considered directly in the testing of hypothesis three: knowledge stakeholder influence.

The results provide moderate empirical support for the accounts in the extant literature that individuals have significant power when it comes to prioritizing certain types of knowledge in meta-organisations (c.f. von Krogh, Spaeth, & Lakhani, 2003; Fitzgerald, 2006; Dahlander & Frederiksen, 2012). The results build on this theory by suggesting that organisations have relatively low power compared to individuals in designating the dominant knowledge paradigms. This additional result is important because meta-organisation members are often concerned that organisational participants exert disproportionate power in the meta-organisations relative to individuals (c.f. Dahlander & Magnusson, 2005; Bagozzi & Dholakia, 2006; Lerner, Pathak, & Tirole, 2006; West & Gallagher, 2006; Gawer & Cusumano, 2008; Boudreau & Lakhani, 2009). While that may be the case in other meta-organisations, the present results suggest that the concern may be unfounded at least in the case of the Mozilla meta-organisation. Future research may wish to replicate these models using data from another meta-organisation, such as the Eclipse Foundation, and compare and contrast the results across the meta-organisations in order to determine the generalizability of the present results.

Hypothesis four: Knowledge flow impediments

The fourth hypothesis postulates that, "*Knowledge flow impediments are negatively correlated with solution knowledge emergence*." This hypothesis is tested by analyzing data at the problem, individual, and organisation levels of analysis; cross-level nesting effects between individuals and the organisations of which they are a member are also assessed. The results for each level are discussed in turn in the following sections.

Problem level of analysis results

At the problem level of analysis there are seven measures that represent the conceptualization of the dependent outcome of interest, solution knowledge emergence, as depicted in Figure 11, and as operationalized into the variables described in Table 51. Six measures represented the conceptualization of the antecedent of interest in hypothesis four, knowledge flow impediments, as depicted in Figure 16, and as operationalized into the variables described in Table 56. The results of the analyses are depicted in the regression model output summaries in Appendix D: Regression models.

Measure one: Rapid-release strategy timing

The first measure of knowledge flow impediments at the problem level of analysis is the timing of the revelation of the problem knowledge to the meta-organisation relative to its switch to the rapid-release strategy for its knowledge production process. It is hypothesized that problem knowledge submitted before the rapid-release strategy switch is negatively correlated with solution knowledge emergence due to the knowledge flow improvements resulting from the change in knowledge production process.

Examination of the regression model summaries suggests that problems submitted before the transition to rapid-release strategy are assigned more slowly ($p < 0.01$) but are not correlated with any other significant outcome effects, providing only partial support for the hypothesis. This result suggests that the strategy of shifting to rapid-release is effective at the problem level in more rapidly triaging and assigning problems. Yet, it is not clear that solution knowledge is created any faster, nor that it is any more likely to emerge. This result is useful for practice as it suggests that changes in release strategy may only affect the assignment facet of the knowledge production process.

Measure two: Number of activities on problems within time frame quantiles

The second measure of knowledge flow impediments is the number of activities on new problems within quantile-based time frames after their submission to the meta-organisation. It is hypothesized that more activities on problem knowledge shortly after it is submitted is correlated with improved solution knowledge emergence outcomes and more activities a long time after the problem is submitted is correlated with worse solution knowledge emergence outcomes. Further, a problem having more than twenty activities is hypothesized to have positive solution knowledge outcomes irrespective of activity timing. As discussed in Chapter Four: Research Method, based on the distribution of the activities on problems in the data, quantile based thresholds are created at the intervals described in Table 5 and Table 6.

Examination of the regression model summaries suggests strong support for the hypothesis, with problems with activities 1 to 3 days, 3 to 7 days, and 7 to 15 days after creation being strongly correlated ($p < 0.001$) with increased fix emergence. As hypothesized, albeit slightly later in the timing than theorized, an inflection point appears at 15 to 45 days where activities in each interval are correlated ($p < 0.001$) with worse patch emergence outcomes up to interval 1 to 2 years after creation and intervals 45 to 90 days through 2 plus years after creation are correlated ($p < 0.001$) with worse fix emergence outcomes as well.

Of note, while it is theorized that more than twenty activities has a positive effect, the effect appears to be split, with both more than twenty activities in total and more than twenty activities later than 2 years after creation being correlated with worse fix emergence tendencies

($p < 0.001$) while also being correlated with better patch emergence tendencies ($p < 0.001$). This split in type of solution knowledge emergence is notable as it suggests that activities are associated with the type of solution knowledge that emerges independent of the specifics of the problem knowledge submitted to the meta-organisation. From a theory standpoint, it lends support to the notion that activities in the knowledge production process affect knowledge production, independent of the properties of the knowledge produced. From a practice standpoint, it suggests that meta-organisation participants must be aware of knowledge flow activities and may wish to curate that process if they wish to improve the emergence of solution knowledge of a type that is of benefit to them.

A note of caution is warranted in the interpretation of the other dependent variable results for activity count measures. There is endogeneity in timing outcome effects with time-based measures that is unavoidable. Activity on a problem implies that it is not yet resolved in most cases. As such, it stands to reason that there would be significant association between resolution, timing, and development outcomes by virtue of problems still being open alone. For this reason, those significant results should be interpreted as endogenous. Likewise, given that reopening, reassigning, and confirming are activities themselves, there is endogeneity in this outcome measures for these activity measures, making the associations in the model summaries spurious by variable definition. Future research using a database that is able to disambiguate specific timing and activity effects may be fruitful in order to examine those facets of solution knowledge emergence, with the present design only able to focus on fix and patch emergence variables, which are the primary solution knowledge emergence outcomes of interest.

Measure three: Reopening and reassigning of problems

The third measure of knowledge flow impediments is the reopening and reassigning tendencies of problems. It is hypothesized that problems that are reopened or reassigned are correlated with worse solution knowledge emergence outcomes due to the interruption of the flow of the knowledge production process.

Examination of the regression model summaries suggests that, contrary to as hypothesized, reopening and reassigning are correlated with increased fix ($p < 0.05$) and patch ($p < 0.01$ & $p < 0.001$ respectively) tendencies. Further, reopening is correlated with increased resolution time $p < 0.001$), increased development time ($p < 0.001$), and reduced confirmation ($p < 0.001$); and, reassigning is correlated with reduced resolution time ($p < 0.001$), reduced first assignment time ($p < 0.001$), increased development time ($p < 0.001$), and increased confirmation ($p < 0.001$). Of particular note, reassigning and reopening do not appear to be correlated with each other, supporting the orthogonal nature of these measures.

The threshold timing variables suggest that the distribution of reopening tendencies is evenly spaced around development timings such that it is generally negatively correlated with faster development and positively correlated with slower development. These development timing effects likely contribute to the observed overall slower resolution effect of reopened problems, as hypothesized. By contrast, for reassignment, there is significant correlation with “extremely fast” assignment ($p < 0.001$), “very fast” assignment ($p < 0.001$), and “fast” assignment ($p < 0.001$) suggesting that reassignment results from an overly quick assignment process selecting a suboptimal solution knowledge producer and increasing the time until an appropriate developer is found and development of solution knowledge is completed.

In summary, whereas the outcome effects associated with timing are negatively affected by reopening and reassigning, as hypothesized, the fix and patch outcome effects are contrary to as hypothesized. This mixed result suggests that while reopening and reassigning do impede the knowledge flow, the impediment is temporary, only delaying, not precluding, the emergence of solution knowledge. The contributions to practice are that reopening and reassigning activities generally succeed in their goal of re-entering the knowledge production process after an initial attempt does not result in solution knowledge creation. In addition, meta-organisations may wish to be careful with hasty assignment of problem knowledge as that haste ends up simply being shifted to longer development time. The contributions to theory are that knowledge production processes with iterative cycles may be useful mechanisms for creating solution knowledge more efficiently than inflexible, yet more direct processes.

Measure four: Keyword, flag, whiteboard, and target milestone changes

The fourth measure of knowledge flow impediments is the changing of keywords, flags, whiteboard, or target milestone associated with problems. It is hypothesized that changes in the keywords, flags, whiteboard, or target milestone of problems after they are submitted to the meta-organisation are correlated with worse solution knowledge emergence outcomes due to the disruption to the flow of the knowledge production process resulting from these changes.

Examination of the regression model summaries suggests that keyword changes are correlated with faster overall resolution ($p < 0.001$), slower assignment ($p < 0.001$), faster development ($p < 0.001$), increased patch emergence tendencies ($p < 0.001$), decreased reopening ($p < 0.001$) and increased confirmation tendencies ($p < 0.001$); flag changes are correlated with faster resolution ($p < 0.001$), faster assignment ($p < 0.001$), faster development (p

< 0.001), increased fix and patch emergence tendencies ($p < 0.001$), decreased reopening ($p < 0.001$), and increased confirmation tendencies ($p < 0.001$); whiteboard changes are correlated with slower resolution ($p < 0.001$), slower assignment ($p < 0.001$), slower development ($p < 0.001$), increased fix and patch emergence tendencies ($p < 0.001$), decreased reopening ($p < 0.001$), increased reassigning ($p < 0.05$), and decreased confirmation tendencies ($p < 0.001$); and, target milestone changes are correlated with increased resolution time ($p < 0.001$), increased assignment time ($p < 0.05$), increased development time ($p < 0.01$), increased fix and patch emergence tendencies ($p < 0.001$), decreased reopening ($p < 0.001$), increased reassigning ($p < 0.001$), and increased confirmation tendencies ($p < 0.001$).

The results suggest a mixed picture based on the type of change to problem knowledge during the knowledge production process. Contrary to as hypothesized, all types of knowledge change appear to be correlated with improved fix and/or patch emergence tendency outcomes. In the case of keyword and flag changes, the changes are correlated with faster development and reduced reopening tendencies, suggesting that the additional knowledge contained in the new keywords is useful for speeding up the development process and reducing the likelihood of mismatch between problems and solutions. In the case of whiteboard and target milestone changes, the changes are correlated with increased development time and increased reassigning tendencies, suggesting that the additional knowledge, while still having positive overall solution knowledge emergence effects in terms of positive fix and patch emergence tendencies, results in significant delays in the knowledge production process by requiring the involvement of new solution knowledge producers who take the new knowledge into account when creating the subsequent solution.

In all cases the additional knowledge is useful, but some knowledge speeds up the process and other knowledge slows it down. Examination of the threshold outcomes suggests that keyboard and flag changes are generally positively correlated with faster development thresholds and generally negatively correlated with slower development thresholds. The whiteboard and milestone change effects primarily appear to be around precluding extremely fast development and promoting more average development speeds, resulting in the net slower development observed in the continuous outcome measure results.

A possible explanation for the split in the effects based on the type of change relates to the amount of problem information that is changed and the resulting required effort to absorb that new information. Keywords and flags are small amounts of knowledge that rely on tacit subject matter expertise for interpretation rapidly and readily. By contrast, whiteboards and milestones are often much more elaborate and detailed, relying less on tacit knowledge to explain knowledge changes more succinctly. The consumption of the greater quantity of new problem knowledge does not benefit from the codifiability observed in the results of the tests of the previous hypotheses and taxes the absorptive capability of actors involved in the knowledge production process more than does the compact keyword and flag knowledge, delaying yet still improving solution knowledge emergence. Comparison of the problem level results with the individual level role-specific results sheds more light on the nature of the problem knowledge consumption of actors in the observed outcome effects.

Measure five: Bug life cycle violation

The fifth measure of knowledge flow impediments is the violation of the bug life cycle. It is hypothesized that bug life cycle violation is correlated with worse solution knowledge emergence outcomes due to the resulting disruption of the flow of the knowledge production process. As discussed in Chapter Four: Research Method, bug life cycle violation was determined based on the knowledge flow depicted in Figure 10, which is the formal representation of the knowledge production process used in the Mozilla meta-organisation.

Examination of the regression model summaries suggests that violation of the bug life cycle is correlated with increased resolution time ($p < 0.001$), increased assignment time ($p < 0.001$), decreased development time ($p < 0.001$), decreased fix emergence tendencies ($p < 0.05$), increased reopening and reassigning tendencies ($p < 0.001$), and decreased confirmation tendencies. Further, the ANCOVA and base summary regression models suggest a strong correlation with reduced patch emergence tendencies, although the significance of the association drops markedly after heteroskedasticity correction, suggesting that the association may exist with some class of bugs but not others. Examination of the timing threshold outcomes reveals a strong correlation ($p < 0.001$) with “extremely slow resolution” and “extremely slow assignment”, suggesting that problems that violate the bug life cycle are more likely to be ignored and forgotten for long periods of time. Life cycle violation is also correlated ($p < 0.01$) with “extremely fast” and “very fast” development, which, when paired with reduced fix tendencies, suggests that the “extremely fast” “development” is actually a hasty discarding by the solution knowledge producer who quickly declines to solve the problem that violated the life cycle.

This result is as hypothesized, supporting a strongly negative effects of bug life cycle violations on outcomes of interest. For theory, this result provides empirical evidence supporting the notion that agreed-upon knowledge production processes in organisations must be adhered to in order to promote knowledge creation. Attempting to sidestep certain states in the process results in negative outcomes. Rather, it is preferable to follow all of the steps in the knowledge production process, even if additional time is spent on the apparently unnecessary steps. For practice, for actors in meta-organisations seeking solution knowledge for their problems, the implications are clearly that they must adhere to the knowledge production processes of the meta-organisation to improve outcomes. The meta-organisations themselves should also periodically review their practices and adjust them as necessary to minimize violations and streamline knowledge production efforts.

Measure six: Bug blocking and blocked by tendencies

The sixth measure of knowledge flow impediments is whether a bug is blocking or blocked by another bug. It is hypothesized that bugs that are blocking other bugs are correlated with improved solution knowledge emergence whereas bugs that are blocked by other bugs are correlated with reduced solution knowledge emergence due to the resulting changes in the flow of the knowledge production process.

Examination of the regression model summaries suggests that bugs that are blocking are correlated with reduced resolution time ($p < 0.001$), reduced assignment time ($p < 0.001$), reduced development time ($p < 0.001$), increased fix and patch emergence tendencies ($p < 0.001$), decreased reopening ($p < 0.001$), and increased confirmation ($p < 0.001$). Examination of the timing threshold outcome effects reveals a positive correlation with “extremely fast” assignment ($p < 0.01$) and negative correlation $p < 0.01$) with “slow” development. Taken collectively, these results lend strong support to the hypothesis that problems that are blocking other problems are correlated with better solution knowledge emergence tendencies.

Specifically, it appears that the problem blocking results in much faster assignment effects and, while it does not seem to speed up development, ensures that the blocking problems are not slowing down development.

For problems that are blocked by other problems, there is correlation with increased development time ($p < 0.001$), increased fix emergence tendencies ($p < 0.001$), reduced patch emergence tendencies ($p < 0.05$), and increased confirmation tendencies ($p < 0.001$). Examination of the timing threshold outcome effects reveals negative correlation with "extremely fast" ($p < 0.001$) and "fast" ($p < 0.01$) resolution and positive correlation with "extremely slow" ($p < 0.001$) resolution. There is also positive correlation with "average" development time ($p < 0.05$), which suggests that the blocked by delay in resolution is not the result of slow development, but rather is the result of the blocking itself creating a lag effect. The blocking may go on for very long periods of time in some cases, resulting in the "extremely slow" resolution. Taken collectively, these results lend partial support to the hypothesis that problems that are blocked by other problems are correlated with worse solution knowledge emergence. A notable exception is the positive fix emergence tendencies, which is counterintuitive. Despite the delays and worse patch emergence tendencies, blocked problems are still correlated with generally more favourable fix emergence tendencies. By definition, a blocked bug is one that cannot be resolved until another bug is resolved first. One possible explanation is that the "blocking" signaling artefact is not being used correctly in the meta-organisation. It may be that "blocked" problems were "soft-blocked", as in, could actually be resolved without the blocking bug being resolved first, despite being marked as such. While the knowledge production process in the meta-organisation describes that blocking bugs need to be resolved before the bugs they are blocking, it is possible that participants sometimes ignore this knowledge flow and use "blocking" as a signal of a different purpose than the intended purpose as a signal of hard knowledge production order path dependency. A related possible explanation is that problems that are blocked by other problems are blocked because the association between problems speaks to their importance to the meta-organisation. The whole point of the blocking and blocked by tracking is to ensure that important problems that depend on other problems are not forgotten (Valdivia Garcia & Shihab, 2014). In the case of fix emergence tendencies, this strategy appears to be successful although it does not appear to extend to patch emergence tendency benefits. It may be that patch solution knowledge suffers more from the observed resolution delay than other types of non-patch solution knowledge. A third possible explanation is that the "blocked by" field in the database is not being correctly updated after a blocking bug is resolved, leaving the appearance of a block still existing on a focal bug even when it has already been cleared. Given that there is no automatic mechanism to clear the blocking field, it is plausible that human actors forget to revisit bugs and leave that database field unchanged.

The contributions to theory are that dependencies between different sets of problem knowledge have significant effects on both the knowledge production process and solution knowledge emergence. Further, identification of such dependences between knowledge is useful for improving outcomes. For practice, the blocking system appears to be an effective way of ensuring that problems are not independently forgotten, but may misrepresent the strictness of the dependency between problems, since blocked problems still have higher fix tendencies, which is counterintuitive. Yet, meta-organisation actors seeking solution knowledge in the form of patches may wish to ensure that if the problem knowledge they submit gets blocked that they shift their attention to solving that problem quickly in order to subsequently have patch solution knowledge emerge for their focal problem. In addition, open source meta-organisations may wish to create an automated process to clear the "blocked by" field of bugs that are blocked when the associated blocking bug is resolved to ensure that bugs aren't spuriously left marked as "blocked" even when such a block has been resolved.

Summary of dependent variable effect results at problem level

Comparison of the control and full regression models for the effect of knowledge flow impediment measures on the dependent outcomes of interest reveals an overall picture of effect size at the problem level of analysis. The model F & Chi-squared statistics as well as the comparative AIC delta statistics suggest that in all cases the knowledge flow impediment independent variable full models are significantly superior to the control variable only models.

For resolution time, assignment time, and development time, the Cohen's additive f^2 effect sizes are "very large", "medium-to-large" and "large" respectively, with incremental R^2 values of 0.210 to 0.494, 0.089 to 0.241, and 0.381 to 0.538 respectively. For fix and patch emergence tendencies, the additive effect sizes are "medium-to-large" and "very large" respectively, with incremental pseudo-R^2 values of 0.582 to 0.663 and 0.464 to 0.701 respectively. For reopening, reassigning, and confirmation tendencies, the additive effect sizes are "medium", "large", and "medium" respectively, with incremental pseudo-R^2 values of 0.004 to 0.171, 0.219 to 0.402, and 0.525 to 0.595 respectively. For the threshold based resolution time outcomes, the additive effect sizes are "very large" for all the thresholds. For the threshold based assignment time outcomes, the additive effect sizes are "large" for all thresholds except "extremely fast assignment" which is "small-to-medium" and "average assignment" which is "medium-to-large". For the threshold based development time outcomes, the additive effect sizes ranged from "medium" to "large".

In summary, significant effects were observed amongst most of the knowledge flow impediment independent variables, lending significant support to hypothesis four. The rapid release transition is correlated with faster assignment of problems; activities on problems in the first few days after their submission to the meta-organisation are correlated with positive outcomes, whereas activities in the 15 to 45 day range and beyond are correlated with negative outcomes; a large number of activities may compensate for timing by increasing patch type solution knowledge emergence despite a general reduction in fix emergence tendencies; reopening and reassigning of problems increases fix and patch emergence tendencies albeit at the cost of time delays; keyword and flag changes provide useful additional problem knowledge that increases fix and patch emergence tendencies as well as resolution speed, whereas whiteboard and target milestone changes also improve fix and patch emergence tendencies but slow resolution; problems that violate the knowledge flow life cycle are generally correlated with negative outcomes; and, problems that block other problems are correlated with increased fix and patch emergence tendencies and faster resolution time, whereas problems that are blocked by other problems have increased fix emergence tendencies but reduced patch emergence tendencies and slower resolution time.

Individual level of analysis results

At the individual level of analysis there are seven measures that represent the conceptualization of the dependent outcome of interest, solution knowledge emergence, as depicted in Figure 19, and as operationalized into the variables described in Table 79 and Table 87. Six measures represent the conceptualization of the antecedent of interest in hypothesis four, knowledge flow impediments, as depicted in Figure 23, and as operationalized into the variables described in Table 92. The results of the analyses are depicted in the regression model output summaries in Appendix D: Regression models.

As discussed in Chapter Four: Research Method, regression models are separated according to the roles in which individuals engage when participating in the knowledge production process: problem knowledge producer (reporter), solution knowledge producer (assigned_to), and solution knowledge verifier (QA_contact).

Measure one: Percent of bugs acted upon in each role that violated bug life cycle

The first measure of knowledge flow impediments at the individual level of analysis is the percent of bugs acted upon in each role that violated the bug life cycle. A tendency to act more frequently on bugs that violated the bug life cycle is theorized to be correlated with negative solution knowledge emergence outcomes due to the knowledge flow impediment of the life cycle violations.

Examination of the regression model summaries reveals that the percentage of bugs acted upon in the problem knowledge producer role that violated the bug life cycle is correlated with increased resolution time ($p < 0.001$), increased reopening and reassigning tendencies ($p < 0.001$), and decreased fix ($p < 0.001$) and patch ($p < 0.01$) emergence tendencies. For the solution knowledge producer role, increased percentage of bug life cycle violations is also correlated with decreased fix emergence tendencies ($p < 0.001$).

These results lend support to the hypothesis that bug life cycle violations are correlated with worse solution knowledge emergence outcomes and match the results observed at the problem level of analysis. It is interesting to note that the negative outcome effects appear to be more prominent for the problem knowledge producer role, reducing both fix and patch emergence tendencies whereas only fix emergence tendencies are hindered for the solution knowledge producer role (and at the problem level of analysis). A possible explanation is that the impact of life cycle violation on outcomes is different depending on the actor who causes the life cycle violation. If problem knowledge producers violate the knowledge flow life cycle, the result is uniformly negative because they lack the specific expertise to decide when bypassing certain stages of the knowledge flow is appropriate. By contrast, when solution knowledge producers, who have the expertise to more deeply understand the knowledge production process, decide to violate the life cycle with a given problem, the outcomes are less negative, with patch emergence tendencies not being as significantly hindered on average.

The takeaway for theory is that knowledge production processes may have different usefulness for different knowledge producer roles, where some roles may be able to ignore the process more than others. A takeaway for practice is that allowing some developers to bypass certain parts of the knowledge production process may not be as negative as life cycle violations by general actors. Separate knowledge flows entries, exits, and transitions for different classes of actors may be appropriate. Nevertheless, the simplest takeaway is that, on average, it is best to follow knowledge production processes and not attempt to take shortcuts.

Measure two: Percent of bugs acted upon in each role with at least one target milestone change

The second measure of knowledge flow impediments is the percent of bugs acted upon in each role that had at least one target milestone change. A tendency to act more frequently on bugs that had target milestone changes is theorized to be correlated with negative solution knowledge emergence outcomes due to the knowledge flow disruption of the target milestone change.

Examination of the regression model summaries reveals that percentage of bugs acted upon in the problem knowledge producer role that had at least one target milestone change is correlated with increased resolution time ($p < 0.05$), increased reassignment tendencies ($p < 0.001$), and increased fix and patch emergence tendencies ($p < 0.001$). The same results are seen for resolution time ($p < 0.01$), and fix and patch emergence tendencies ($p < 0.001$) for the solution knowledge producer role.

These results match those seen at the problem level of analysis and suggest that target milestone changes are correlated with better solution knowledge emergence, contrary to the hypothesis, albeit at the cost of resolution time delays and knowledge flow interruptions in the form of reassignments. The target milestone change, itself, only temporarily impedes, in a largely role-independent manner, solution knowledge emergence. In this respect, the target milestone changing activity appears to be working as intended by ensuring that problems can be dynamically re-prioritized as resource availability shifts in the meta-organisation without loss of solution knowledge emergence as a result.

A notable exception in the results is in the case of the solution knowledge verifier role, where higher percentage of bugs with target milestone change is correlated ($p < 0.001$) with reduced resolution time. Unlike previous models where the power of sample size of the solution knowledge verifier role precludes meaningful analysis, examination of the present model as well as the mixed-effects model of the QA_contact role suggests that this reduced resolution time is not spurious and attributable to the $n = 461$ observations. This role-specific effect is unexpected.

One possible explanation is that solution knowledge verifiers are sometimes assigned to specific products exclusively and some products have regularly changing milestones as a normal practice. It may be that, as a result, those solution knowledge verifiers who are associated with those regularly changing milestones are working on products that inherently have problems that are resolved faster. Hypothesis six: Solution knowledge value explores this factor independently.

Measure three: Percent of bugs acted upon in each role with at least one severity change

The third measure of knowledge flow impediments is the percent of bugs acted upon in each role that had at least one severity change. A tendency to act more frequently on bugs that had severity changes is theorized to be correlated with negative solution knowledge emergence outcomes due to the knowledge flow disruption resulting from the changes in severity after problem knowledge submission.

Examination of the regression model summaries reveals that percentage of bugs acted upon in the problem knowledge producer role that had at least one severity change is correlated with increased resolution time ($p < 0.001$) and decreased fix and patch emergence tendencies ($p < 0.001$). The increased resolution time ($p < 0.001$) and decreased patch emergence tendencies ($p < 0.05$) also hold for the solution knowledge producer role, although fix emergence tendencies don't achieve significance. The decreased fix ($p < 0.001$) and patch ($p < 0.01$) emergence tendencies are also noted for the solution knowledge verifier role.

These results provide support for the hypothesis that changes in severity subsequent to problem knowledge submission to the meta-organisation impede the knowledge flow, hindering solution knowledge emergence. While the negative outcome effects are fairly consistent, one possible explanation for the solution knowledge producer role fix emergence tendencies not being as negatively affected is that studies have shown that solution knowledge producers judge problem severity more accurately that problem knowledge producers on average (c.f. Zhou, Neamtiu, & Gupta, 2015). Therefore, it may be that solution knowledge producers are less negatively impacted by the change to the stated severity level because they already judge the severity independently of the stated level in the problem knowledge submission. In fact, in some cases, developers learn to ignore certain tokens in problem knowledge submission when they find them less useful than their own evaluations (Lewis, et al., 2013). While the present results cannot clearly be attributed to that effect, future research using data that can clearly separate the type of solution knowledge emergence and role specific knowledge flow impediments associated with severity changes may be fruitful in further exploring this result.

Measure four: Percent of bugs acted upon in each role with at least one reopening

The fourth measure of knowledge flow impediments is the percent of bugs acted upon in each role that had at least one reopening. A tendency to act more frequently on bugs that were reopened is theorized to be correlated with negative solution knowledge emergence outcomes due to the knowledge flow disruption of the reopening.

Examination of the regression model summaries reveals that the percentage of bugs acted upon in the problem knowledge producer role that were reopened is correlated with increased reassigning tendencies ($p < 0.001$), and decreased fix and patch emergence tendencies ($p < 0.001$). The increased reassigning ($p < 0.001$) and decreased patch emergence tendencies ($p < 0.001$) also hold for the solution knowledge producer role. This result lends support to the hypothesis that reopening tendencies reduce solution knowledge emergence.

Notably, the fix and patch emergence tendency results are the opposite of those observed at the problem level of analysis. This difference suggests that the negative effects may be more attributable to the individual level than the problem level. Whereas problems being reopened may not necessarily speak to the problem knowledge being problematic enough to hinder solution knowledge emergence, a tendency to engage with problems that are reopened may speak to an individual action that is causing the reopening independent of the problem knowledge properties themselves. The positive correlation with reassigning tendencies is also distinct from the problem level, where reopening and reassigning are not correlated. This result suggests that the correlation between reopening and reassigning tendencies resides particularly at the individual level, independent of the properties of the problem knowledge. These results build upon past research on bug reopening and reassigning tendencies that has focused on problem level factors (e.g. Guo, et al., 2010, 2011) by providing empirical evidence for not previously reported individual level specific effects.

As is the case with the severity change measure, the lack of negative effect on fix emergence tendencies for the solution knowledge producer role for reopening is interesting of its own accord. Given that reopening generally occurs when a solution knowledge producer produces a solution that is judged by the solution knowledge verifier to not address the problem it was intended to resolve, a possibility is that the negative effect is limited to solution knowledge that involves patches but not other types of solution knowledge. Solution knowledge producers who engage primarily with problems that do not require patches to resolve may be affected differently by reopening than those producing patches. Examination of the mixed-effects models supports this potential explanation by highlighting that the problem knowledge producer role fix hinderance effects may not, in fact, reside at the individual level. The reduced fix emergence tendency loses significance in the mixed-effects model that accounts for organisational nestedness. Therefore, the observed split in solution knowledge emergence hindrance may be more attributable to the different developers in different organisations. Comparison with the organisation level results provides a clearer picture of this result, as discussed in the next section.

Measure five: Percent of bugs acted upon in each role with at least one reassigning

The fifth measure of knowledge flow impediments is the percent of bugs acted upon in each role that had at least one reassigning. A tendency to act more frequently on bugs that were reassigned is theorized to be correlated with negative solution knowledge emergence outcomes due to the knowledge flow disruption of the reassigning.

Examination of the regression model summaries reveals that the percentage of bugs acted upon in the problem knowledge producer role that were reassigned is correlated with increased resolution time ($p < 0.001$), increased reopening tendencies ($p < 0.001$), and decreased patch emergence tendencies ($p < 0.001$). These results hold for the solution knowledge producer role, with the addition of reduced fix emergence tendencies ($p < 0.001$). Increased resolution time ($p < 0.01$) is apparent for the solution knowledge verifier role.

These results lend support to the hypothesis that tendencies to act on bugs that were reassigned, at the individual level, is correlated with worse solution knowledge emergence. As is the case with reopening tendencies, these results differ from the problem level of analysis suggesting that the negative effects on solution knowledge emergence are the result of individual level factors independent of the specifics of the problem knowledge. The consistent correlation across roles between reopening and reassigning tendencies suggests that these factors are specific to the individuals, not to the roles in which they engage.

A role specific effect appears in the difference in fix emergence tendencies, where there is only a significant correlation with reassigning tendencies for the solution knowledge producer role, despite reassigning for both roles being associated with reduced patch emergence tendencies. This effect is the opposite as that of reopening, where the negative fix correlation is only for the problem knowledge producer (albeit confounded by organisation level effects) and not the solution knowledge producer role. A possible explanation lies in the nature of the reassigning activity, which necessarily implies a change of solution knowledge producer, by definition. It follows that reassigning would more negatively affect the solution knowledge producer as it is the change in developer that is the impediment in the knowledge flow. The problem knowledge producer is not changed. Once again, these results build upon the extant literature on knowledge flow impediments associated with reopening and reassigning tendencies by providing empirical results that suggest an independent individual level effect that has not previously been reported given the problem level focus of past research (e.g. Guo, et al., 2010, 2011).

Measure six: Mean number of activities on bugs acted upon in each role within time frame quantiles

The sixth measure of knowledge flow impediments is the mean number of activities on bugs acted upon in each role within each of the time frame quantiles. A tendency to act more frequently on bugs that had fewer activities is theorized to be correlated with negative solution knowledge emergence outcomes due to the knowledge flow impediment of the lack of activities moving the knowledge production process along its life cycle.

Examination of the regression model summaries reveals that for the problem knowledge producer role, the mean number of activities, in all time frame quantiles, is correlated with increased fix and patch emergence tendencies ($p < 0.001$). For the solution knowledge producer role, the mean number of activities, in all time frame quantiles, is also correlated with increased patch emergence tendencies ($p < 0.05$ to $p < 0.001$ amongst quantiles), yet only correlated with increased fix tendencies for mean activities in the 3 to 7 days ($p < 0.05$) and 180 to 365 days ($p < 0.01$) quantiles.

For the solution knowledge verifier role, the mean number of activities, in the first three quantiles up to 15 days ($p < 0.05$, $p< 0.01$, and $p < 0.01$ respectively), is correlated with increased fix emergence tendencies; and, the mean number of activities in the first two quantiles, up to 7 days ($p < 0.01$ and $p < 0.01$), is correlated with increased patch emergence. Interestingly, higher mean number of activities in the 15 to 45 day range is correlated with reduced fix emergence tendencies ($p < 0.05$) exclusively for the solution knowledge verifier role.

These results lend support to the hypothesis that increased activities on problems is associated with better solution knowledge emergence. The uniformity of the results differs from the problem level of analysis where an inflection point appears around 45 days, where later activities result in worse solution knowledge emergence. This difference suggests the existence of an individual level effect where the tendencies of individuals acting in the knowledge production process moderates the degree to which activity timings facilitate or impede the knowledge flow. At the problem level, later activities represent a time based impediment in the knowledge production process, hindering solution knowledge emergence. By contrast, at the individual level, later activities, curated by actors who have a tendency to be involved with problems that have later activities, facilitate rather than impede the process by virtue of the actors directing the knowledge flow individually. Further, the problem knowledge producer and solution knowledge producer roles appear to moderate the usefulness of later activities better than the solution knowledge verifier role because the inflection point of positive to negative fix emergence tendencies is only observed in the latter role's case, just like at the problem level of analysis.

The implication for theory is that problem knowledge that is curated by an individual who follows it through the knowledge production process may have better solution knowledge emergence, independent of the problem knowledge properties alone. The implication for practice is that individuals in meta-organisations have agency over the knowledge production process and can curate problem knowledge that is important to them to improve outcomes after the problem knowledge is submitted to the meta-organisation and entered into the knowledge production process. In particular, individuals who continue engage with problems that have later activities can, on average, turn around the negative effects that those activities have independent of their involvement. Further, examination of the mixed-effects regression models suggests that while there are also organisation level effects, which will be considered in the next section, there are undoubtedly individual level effects independent of organisation embeddedness.

Effects of individual nestedness in organisations

Examination of the mixed-effects regression model summaries reveals that there are significant organisational effects that affect the individual level results. For the problem knowledge producer and solution knowledge producer roles, the AIC and BIC delta statistics reveal that the mixed-effects models are superior in all cases, with effect sizes ranging from "medium" to "very large". For the solution knowledge verifier role, the mixed-effects models for reassigning tendencies and fix tendencies are superior, with "very large" and "large" random effects, respectively.

Despite the significant organisational nestedness effects, most of the individual level results remain significant after isolating the organisational random effects. There are three notable exceptions for the problem knowledge producer role. The correlation between reassignment and increased resolution time loses significance in the mixed-effects model, suggesting that the delay may be the result of more organisation level than individual level effects. The correlation between reopening and reduced fix emergence tendencies loses significance, suggesting reduced fix effects are attributable to organisation level factors. And, the association between severity change and reduced fix emergence tendencies also loses significance in the mixed-effects model, suggesting organisation level factors.

For the solution knowledge producer role, the correlation between tendencies of individuals to act on problems that were reassigned and increased resolution time does not achieve significance for either OLS or mixed-effects models. Examination of the ANCOVA and standard regression summaries reveals that the lack of significance is likely due to the model differences rather than the variable effects. In the reduced OLS model, the variable achieves a certainty of significance of $p \sim= 0.10$, which is below an acceptable threshold, but the reduced significance is plausibly attributable to the lower number of observations ($n = 2453$ vs. $n = 1171$) and the change in variable configuration in the least-squares estimate. Further, the coefficients in the mixed-effects model achieve a certainty of significance of just slightly above the $p < 0.05$ threshold, further increasing the support for the existence of the effect observed in the individual level results. Nevertheless, caution is undoubtedly warranted, and a conservative perspective should be applied to the individual level effects between reassignment and resolution time in the case of the solution knowledge producer role.

By and large, the mixed-effects models suggest a strong individual level knowledge flow impediments effect on the solution knowledge emergence outcomes of interest independent of the organisation level effects that are considered in the next section.

Summary of dependent variable effect results at individual level

Comparison of the control and full regression models for the effect of knowledge flow impediments independent variables on the dependent outcomes of interest reveals an overall picture of effect size at the individual level of analysis. For the problem knowledge producer role, the comparative Chi-squared statistic and comparative AIC delta statistic reveal that the models that include the knowledge flow impediments variables are all superior to the control-only models. The additive effect sizes of the independent variables above and beyond the control variables are "medium" for days to resolution dependent variable, "small-to-medium" and "large" for reopening and reassigning tendencies, respectively, and "medium-to-large" and "large" for fix and patch emergence tendencies respectively.

For the solution knowledge producer role, the comparative Chi-squared and comparative AIC delta statistics also suggest that the full models are superior to the control models in all cases. The additive effect sizes of the independent variables above and beyond the control variables are “very large” for days to resolution, “medium” for reopening and reassigning tendencies, and “medium” and “very large” for fix and patch emergence tendencies respectively.

For the solution knowledge verifier role, the comparative Chi-squared and comparative AIC delta statistics also suggest that the full models are superior to the control models in all cases. The additive effect sizes of the independent variables above and beyond the control variables are “very large” for days to resolution, reopening tendencies, and reassigning tendencies, and “medium” and “medium-to-large” for fix and patch emergence tendencies respectively. Given the reduced (n <= 451) number of observations for the solution knowledge verifier role, some caution is warranted when considering the effect sizes to properly contextualize “very large” effects to the limited representation of this role in the dataset. Nevertheless, the knowledge flow impediment variables undoubtedly have a solution knowledge verifier role specific effect, whereas the variables associated with the previous hypotheses show negligible effects for this role. Given the solution knowledge verifier role, by definition, involves the knowledge production process, it stands to reason that the role would display larger effects in the knowledge flow impediments hypothesis.

In summary, the individual level knowledge flow impediments effects are best characterised as a balance between knowledge production processes serving their intended purposes and the impediments associated with changes to that knowledge flow. Tendencies to act on problems that violate the bug life cycle, like at the problem level, are correlated with worse outcomes, but the outcomes are worse for the problem knowledge producer role than the solution knowledge producer role; tendencies to act on problems with target milestone changes improve fix and patch emergence tendencies at the cost of slower resolution times; tendencies to act on problems with severity changes are generally correlated with worse solution knowledge emergence, though the outcomes are not as bad for solution knowledge producers as for problem knowledge producers, likely because developers evaluate severity on their own independent of the severity level in the original problem knowledge submission; tendencies to act on problems that were reopened and reassigned are generally correlated with worse solution knowledge emergence, with particularly strong individual level effects of reassigning generally less negative for problem knowledge producers and reopening generally less negative for solution knowledge producers; and, the mean number of activities on problems acted upon in all roles is generally correlated with better solution knowledge emergence, unlike at the problem level where there is an inflection point around 15 to 45 days where activities become negatively correlated with solution knowledge emergence; at the individual level, this inflection effect only appears in the solution knowledge verifier role.

The mixed-effects models suggest strong individual level effects independent of organisation level effects on most outcomes of interest, although organisation level effects undoubtedly exist in many cases, as examined in the following section.

Organisation level of analysis results

At the organisation level of analysis there are seven measures that represent the conceptualization of the dependent outcome of interest, solution knowledge emergence, as depicted in Figure 26 and as operationalized into the variables described in Table 105. Six measures represent the conceptualization of the antecedent of interest in hypothesis four, knowledge flow impediments, as depicted in Figure 30, and as operationalized into the variables described in Table 110. The results of the analyses are depicted in the regression model output summaries in Appendix D: Regression models.

As discussed in Chapter Four: Research Method, regression models are separated according to the aggregate roles in which members of organisations engage when participating in the knowledge production process: aggregate problem knowledge producer (reporter) role, aggregate solution knowledge producer (assigned_to) role, and aggregate solution knowledge verifier (QA_contact) role.

Measure one: Percent of bugs acted upon in each aggregate role that violated bug life cycle

The first measure of knowledge flow impediments at the organisation level of analysis is the percent of bugs acted upon in each aggregate role that violated the bug life cycle. A tendency of organisation members to act more frequently on bugs that violated the bug life cycle is theorized to be correlated with worse solution knowledge emergence outcomes due to the knowledge flow disruption of the bug life cycle violations.

Examination of the regression model summaries reveals that for the aggregate problem knowledge producer role, a tendency to act more often on bugs that violate the bug life cycle is correlated with increased reassignment tendencies ($p < 0.001$). This is the only significant effect observed. This result differs from those at the problem and individual levels, which suggests that bug life cycle violation effects are primarily problem and individual level, with very little effect attributable to the organisation level. This interpretation is also supported by the mixed-effects model results that show marginal organisational contribution to the life cycle violation effects. The increased reassignment effect suggests that some organisations who consistently violate the bug life cycle process make the triage process more difficult, but these actions do not appear to result in other negative solution knowledge emergence outcomes for these organisations.

The contributions to theory and practice are that knowledge flows appear to primarily be affected by individual actions rather than organisational actions. These results put into question the effectiveness of organisational processes surrounding knowledge flows, which may be ineffective when they are targeted at organisation level effects. Instead, it may be necessary to create processes for individuals and monitor their effectiveness at the individual level to improve knowledge flows and reduce impediments.

Measure two: Percent of bugs acted upon in each aggregate role with at least one target milestone change

The second measure of knowledge flow impediments is the percent of bugs acted upon in each aggregate role that had at least one target milestone change. A tendency of organisation members to act more frequently on bugs that had at least one target milestone change is theorized to be correlated with worse solution knowledge emergence outcomes due to the knowledge flow disruption of the target milestone change.

Examination of the regression model summaries reveals that for the aggregate problem knowledge producer role, a tendency to act more often on bugs that had at least one target milestone change is correlated with increased reopening tendencies ($p < 0.001$), and increased fix and patch emergence tendencies ($p < 0.001$). The increased fix and patch emergence tendencies are also observed for the aggregate solution knowledge producer role. These results are the opposite of hypothesized and match the results observed at the individual level of analysis.

Interpreted in combination with the mixed-effects regression model results, these results suggest a marginal additional organisation level effect on fix emergence tendencies.

As discussed at the individual level, these results suggest that, contrary to as hypothesized, target milestone changes do not impede solution knowledge emergence. To the contrary, they appear to improve solution knowledge emergence outcomes by ensuring that problem knowledge does not leave the knowledge production process due to temporary shifts in priority such as release deadlines. The results suggest that the target milestone process functions as intended and ensures that solution knowledge eventually emerges for problems whose milestones are changed. The organisation level knowledge production processes appear to further improve the fix emergence tendencies above and beyond the individual level processes, suggesting some effectiveness in that regard.

Measure three: Percent of bugs acted upon in each aggregate role with at least one severity change

The third measure of knowledge flow impediments is the percent of bugs acted upon in each aggregate role that had at least one severity change. A tendency of organisation members to act more frequently on bugs that had at least one severity change is theorized to be correlated with worse solution knowledge emergence outcomes due to the knowledge flow disruption of the severity changes.

Examination of the regression model summaries reveals that for the aggregate problem knowledge producer role, a tendency to act more frequently on problems that had at least one severity change is correlated with increased resolution time ($p < 0.001$), and decreased patch emergence tendencies ($p < 0.05$). Decreased fix emergence tendencies only barely missed the cut-off for sufficient probability of significance at $p < 0.06$. For the aggregate solution knowledge producer role, it was associated only with increased resolution time ($p < 0.001$).

These results match those observed at the individual level of analysis and lend support to the hypothesis that severity changes after problem knowledge submission to the meta-organisation hinder solution knowledge emergence. Despite a probability of significance that marginally missed the typical cut-off, the mixed-effects regression model comparisons suggest the existence of a fix emergence reduction effect at the organisation level in addition to the observed increased resolution time and decreased patch emergence tendency effects, above and beyond the individual level effects. As is the case at the individual level, organisation members engaging in the aggregate developer role are not as strongly negatively affected by severity changes, suggesting once again that they evaluate severity in a manner different from those engaging in the aggregate problem knowledge producer role. Nevertheless, severity changes delay resolution at the organisation level as well, suggesting that they should still be avoided in organisation knowledge production processes whenever possible.

Measure four: Percent of bugs acted upon in each aggregate role with at least one reopening

The fourth measure of knowledge flow impediments is the percent of bugs acted upon in each aggregate role that had at least one reopening. A tendency of organisation members to act more frequently on bugs that had at least one reopening is theorized to be correlated with worse solution knowledge emergence outcomes due to the knowledge flow disruption of the reopening.

Examination of the regression model summaries reveals that for the aggregate problem knowledge producer role, a tendency to act more frequently on problems that had at least one reopening is correlated with decreased fix and patch emergence tendencies ($p < 0.001$). No significant effects were observed for the other outcomes or roles.

This result matches that observed at the individual level of analysis and lends support to the hypothesis that reopening impedes solution knowledge emergence. Examination of the mixed-effects models suggests that there is an organisation level effect above and beyond the individual level effect. The implications are that organisations that attempt to have their problems reopened may see worse outcomes. Instead, they may wish to submit new problem knowledge that addresses the deficiencies of the previous problem knowledge submission that led to it being incorrectly resolved. Given that, at the problem level of analysis, reopening is correlated with better solution knowledge emergence, the negative effect clearly lies in organisation level tendencies that can be improved upon by participating organisations.

It is interesting to note that the aggregate solution knowledge producer role at the organisation level who engages on more bugs that are reopened is not as negatively affected as the individual level solution knowledge producer role. This difference suggests that, for the solution knowledge producer, the individual level effects dominate when it comes to the negative effects of reopening on solution knowledge emergence. It is also interesting to note that, at the organisation level, there is no strong correlation between reopening and reassignment tendencies, as there is at the individual level. The observation of these splits makes contributions to both theory and practice. For theory, the results suggest that similar impediments to knowledge flows can have different effects both within and across levels of analysis based on individual actions. For practice, the results suggest that factors such as reopening of bugs are multidimensional and must be analyzed as such when attempting to improve particular outcomes for meta-organisations. A focus on problem level factors, which is common in studies of open source meta-organisation, is insufficient and may provide confounding results relative to attempts at individual or organisation level process improvements.

Measure five: Percent of bugs acted upon in each aggregate role with at least one reassigning

The fifth measure of knowledge flow impediments is the percent of bugs acted upon in each aggregate role that had at least one reassigning. A tendency of organisation members to act more frequently on bugs that were reassigned is theorized to be correlated with negative solution knowledge emergence outcomes due to knowledge flow disruption of the reassigning.

Examination of the regression model summaries reveals that percentage of bugs acted upon in the aggregate problem knowledge producer role that had at least one reassigning is correlated with increased resolution time ($p < 0.001$). For the aggregate solution knowledge producer role, it is correlated with decreased patch emergence tendencies ($p < 0.05$). These results match those observed at the individual level of analysis and lend support to the hypothesis that reassigning negatively impacts solution knowledge emergence.

Comparison with the mixed-effects regression model summaries suggests that the increased resolution time outcome effect lies primarily at the organisation level of analysis rather than at the individual level. One explanation for why reassignment more strongly impedes the resolution of problems submitted by organisation members acting in the reporter role is that organisations may have schedules and dependencies that are less flexible than those of individuals, resulting in reassignment affecting the knowledge flow of organisation members more than individuals.

This result lends empirical support to the theory in the literature that the size of organisations (Serenko, Bontis, & Hardie, 2007) and intra-organisation networks (Sorenson, Rivkin, & Fleming, 2006; Huggins, Johnston, & Thompson, 2012) is inherently linked to knowledge flows such that the effect of knowledge flow disruptions is significantly different based on the number and configuration of actors. Future research that considers the network configuration of the individual and organisation level actors in meta-organisations may be fruitful in further refining the effects of the knowledge flow impediments of reopening and reassigning tendencies on solution knowledge emergence.

Measure six: Mean number of activities on bugs acted upon in each aggregate role within time frame quantiles

The sixth measure of knowledge flow impediments is the mean number of activities on bugs acted upon by organisation members in each aggregate role within each of the time frame quantiles. A tendency to act more frequently on bugs with fewer activities is theorized to be correlated with negative solution knowledge emergence outcomes due to the knowledge flow impediment of the lack of activities moving the knowledge production process along its life cycle.

Examination of the regression model summaries reveals that for the aggregate problem knowledge producer role, the mean number of activities in all time frame quantiles is correlated with improved fix and patch emergence ($p < 0.001$ to $p < 0.05$ with exception of fix tendencies for mean activities up to 3 days which has $p = 0.051$, only marginally missing the typical certainty of significance threshold). The mean number of activities in the 3 to 7 day range is correlated with reduced resolution time ($p < 0.01$) and the mean number of activities in 90 to 365 day range is correlated with increased resolution time ($p < 0.001$). For the aggregate solution knowledge reporter role, the mean number of activities in the 7 to 15 day range is correlated with increased fix and patch emergence ($p < 0.05$) and the mean number of activities in the 90 to 180 day range is correlated with increased patch emergence ($p < 0.05$).

These results match those observed at the individual level of analysis and lend support to the hypothesis that activities on problems are crucial to solution knowledge emergence. In addition, earlier activities are associated with better outcomes. The organisation level contribution to this effect is relatively small compared to the individual level contribution, relating primarily to the first few days effect for the aggregate problem knowledge producer role. It may be that individuals tend to follow bugs through the knowledge production process longer than organisations because organisations may shift priorities, resulting in later activities being less useful for organisations than individual actors who continue to curate a problem throughout its entire life cycle, regardless of duration. The implication is that organisations may wish to expand the time frames that they allocate to the knowledge production process because doing so may improve outcomes.

Summary of dependent variable effect results at organisation level

Comparison of the control and full regression models for the effect of knowledge flow impediments independent variables on the dependent outcomes of interest reveals an overall picture of effect size at the organisation level of analysis. For the aggregate problem knowledge producer role, the comparative Chi-squared statistic and comparative AIC delta statistic reveal that the models that include the knowledge flow impediments variables are all superior to the control-only models. The additive effect sizes on the outcomes of interest of the independent variables above and beyond the control variables are “medium” for resolution time, “small-to-medium” for reopening tendencies, “large” for reassigning tendencies, and “medium-to-large” for fix and patch emergence tendencies.

For the aggregate solution knowledge producer role, the comparative Chi-squared statistic and comparative AIC delta statistic reveal that the models that include the knowledge flow impediments variables are all superior to the control-only models, though only marginally so in the case of reassigning and reopening tendencies. The additive effect sizes on the outcomes of interest of the independent variables above and beyond the control variables are “large” for resolution time, and “large” and “very large” for fix and patch emergence tendencies respectively. The comparative AIC delta statistic for reassigning and reopening models is sufficiently small that the effect size numbers are artificially inflated and should be contextually interpreted as “small”. As is the case in previous results, the small sample size for the solution knowledge verifier role results in insignificant models across all outcome variables.

In summary, the organisation level effects of the knowledge flow impediment independent variables largely match those observed at the individual level of analysis with a few notable exceptions. Violation of the bug life cycle have only very small effects on reassignment and reopening at the organisation level, whereas much more negative solution knowledge emergence effects are observed at the individual level; target milestone changes are correlated with increased fix and patch emergence tendencies above and beyond the individual level effects; severity changes are correlated with decreased fix and patch emergence for the aggregate reporter role above and beyond the individual level but with only a marginal increase in resolution time for the aggregate developer role; reopening and reassigning tendencies are both correlated with negative outcomes, with strong organisation level effects of reassigning increasing resolution time for the aggregate reporter role and decreasing patch emergence tendencies for the aggregate developer role above and beyond individual level effects; and, more activities sooner is generally positive for solution knowledge emergence, though with organisation level effects only notable in the earlier time ranges after problem knowledge submission.

Hypothesis five: Knowledge stakeholder influence

The fifth hypothesis postulates that, "*Knowledge stakeholder influence is positively correlated with solution knowledge emergence*." This hypothesis is tested by analyzing data at the problem, individual, and organisation levels of analysis; cross-level nesting effects between individuals and the organisations of which they are a member were also assessed. The results for each level are discussed in turn in the following sections.

Problem level of analysis results

At the problem level of analysis there are ere seven measures that represent the conceptualization of the dependent outcome of interest, solution knowledge emergence, as depicted in Figure 11, and as operationalized into the variables described in Table 51. Five measures represent the conceptualization of the antecedent of interest in hypothesis five, knowledge stakeholder influence, as depicted in Figure 17, and as operationalized into the variables described in Table 57. The results of the analyses are depicted in the regression model output summaries in Appendix D: Regression models.

Measure one: Stakeholder who produced the problem knowledge

The first measure of knowledge stakeholder influence at the problem level of analysis is the stakeholder who produced the problem knowledge that was revealed to the meta-organisation. It is hypothesized that stakeholders are heterogeneous in their influence on solution knowledge emergence.

Examination of the ANCOVA model summaries suggests that the stakeholder who produced the problem knowledge is correlated with overall resolution time ($p < 0.001$), development time ($p < 0.01$), fix emergence ($p < 0.001$), reassignment tendencies ($p < 0.01$), and confirmation tendencies ($p < 0.05$). Patch emergence only marginally misses the $p < 0.05$ cut-off for certainty of significance. For the timing threshold models, the reporting stakeholder is correlated with "extremely fast" resolution ($p < 0.001$), "very fast" resolution ($p < 0.01$), "fast" resolution ($p < 0.01$), "average" resolution ($p < 0.001$), "extremely fast" assignment ($p < 0.01$), "extremely fast" development ($p < 0.01$), "fast" development ($p < 0.01$), and "average" development ($p < 0.001$). Several other timing threshold outcome effects only marginally miss the $p < 0.05$ cut-off for certainty of significance.

The ANCOVA results are the most suitable for analyzing the effects of this variable given that its underlying data is categorical in nature, with the very large number of categories, precluding dummy variable analysis. When converted to numerical form, while the ANCOVA analysis handles the effects of the incremental nature of the modified variable, the heteroskedasticity-corrected models assume that the repeated actions of certain actors, which show up as repeated values of the same user id, are to be interpreted as non-linearities that need to be smoothed out in the coefficient estimates. As a result, in the case of the present variable, the heteroskedasticity-correction incorrectly ignores real effects, making the tabular reports of significance less useful than the ANCOVA summary outputs.

These results lend strong support to the hypothesis that problem knowledge producer stakeholders are heterogeneous in terms of their effects on solution knowledge emergence. These results offer empirical support to the theoretical notions in the literature that individual actors may develop reputation and/or skills (c.f. Hooimeijer & Weimer, 2007; Au et al., 2009; Guo, et al., 2010, Shihab, et al, 2010) that enable them to outperform other stakeholders in the meta-organisation. It is clear that individual problem knowledge producer stakeholders have strong influence on solution knowledge emergence in the meta-organisation.

Measure two: Stakeholder who produced the problem knowledge is a core knowledge actor

The second measure of knowledge stakeholder influence is whether the stakeholder who produced the problem knowledge that was revealed to the meta-organisation is a core knowledge actor. It complements the previous measure by determining which subsets of problem knowledge producer stakeholders exert the strongest influence on solution knowledge emergence. It is hypothesized that problems with reporters who are core knowledge actors have better solution knowledge emergence due to the higher influence of those core stakeholders.

Examination of the model output summaries suggests that problem knowledge whose problem knowledge producer stakeholder is a core knowledge actor is correlated with reduced resolution time ($p < 0.001$), reduced assignment time ($p < 0.001$), reduced development time ($p < 0.001$), increased fix and patch emergence tendencies ($p < 0.001$), decreased reopening tendencies ($p < 0.001$), and increased confirmation tendencies ($p < 0.001$). The timing threshold models suggest a general positive correlation with faster resolution and assignment and negative correlation with slower resolution and assignment. For development, there is negative correlation with slower development only.

These results strongly support the hypothesis that knowledge stakeholders who are core knowledge actors exert positive influence on virtually all dimensions of solution knowledge emergence. These results provide additional empirical evidence to support the accounts in the literature of the influence of core knowledge stakeholders in meta-organisations (c.f. Dalle, Besten, & Masmoudi, 2008; Dalle, et al., 2008; Dahlander & Frederiksen, 2012; Masmoudi, 2012). The key implication is that organisations should aim to get core knowledge stakeholders involved in the problems that they submit to the meta-organisation wherever possible.

Measure three: Follows, votes, and comments on problem by core & peripheral knowledge actors

The third measure of knowledge stakeholder influence is the follows, votes, and comments on problems by core and peripheral knowledge stakeholders. It is hypothesized that follows, votes, and comments by core knowledge stakeholders are correlated with positive solution knowledge emergence, with stronger effects than the same action by peripheral knowledge actors. As discussed in Chapter Four: Research Method, due to the nature of the distribution of the count variables in the data set, at the problem level of analysis, logical variables are created that represent the presence or absence of the variables in all cases except core knowledge actor follows count, where the raw count variable is in a suitable form for regression analysis.

Examination of the regression model output summaries suggests that core knowledge actor follows are correlated with increased resolution time ($p < 0.001$), increased assignment time ($p < 0.001$), increased development time ($p < 0.001$), increased fix and patch emergence tendencies ($p < 0.001$), and increased reopening and reassigning tendencies ($p < 0.001$). Examination of the timing threshold models reveals a negative correlation between “very fast” and “fast” resolution ($p < 0.001$), a positive correlation between each of the resolution thresholds from “average” to “extremely slow” resolution ($p < 0.001$); negative correlation between “extremely fast” ($p < 0.001$) and “very fast” ($p < 0.01$) assignment, positive correlation between each of the assignment thresholds from “average” through “extremely slow” assignment ($p < 0.001$); and, negative correlation between “extremely fast” ($p < 0.001$), “very fast” ($p < 0.001$), and “fast” ($p < 0.05$) development, and positive correlation between “average”, “slow” and “very slow” ($p < 0.001$) development. Taken collectively, these results partially support the hypothesis in that core actor follows result in better fix and patch emergence tendencies, but at the cost of significant delays in the form of slower resolution, assignment, and development times, and increased reopening and reassigning tendencies.

For core knowledge actor voting, the regression model output summaries suggest a negative correlation with resolution time, assignment time, and development time ($p < 0.001$), decreased reopening (($p < 0.01$) and reassigning ($p < 0.001$) tendencies, and increased confirmation tendencies ($p < 0.001$). The timing threshold model summaries reveal a negative correlation with “average” ($p < 0.001$) and “very slow” ($p < 0.001$) resolution times, and a positive correlation with “extremely slow” resolution ($p < 0.05$) time. The assignment and development time threshold models are not significant.

These results also provide partial support for the hypothesis that core knowledge actor votes have positive solution knowledge emergence, only along different dimensions than those observed in the follow variable results. Votes by core knowledge actors appear to speed resolution, although it is unclear exactly where in the timing this effect takes place given the contradictory and non-significant threshold model results. Votes also appear to reduce reopening and reassigning, which may be the sole contributory effect to the reduced resolution, which explains the observed results. The lack of effects on fix and patch emergence is surprising and contrary to as hypothesized. The purpose of the voting system is to highlight those problems that are most important to stakeholders. Core knowledge actor stakeholders, therefore, are supposedly able to use their votes to direct the meta-organisation's resources toward creating solution knowledge that is valuable to them. A clear takeaway of the results is that votes may have nowhere near the same effect on solution knowledge emergence as follows. Examination of the individual level and organisation level results helps triangulate the level of this effect.

For core knowledge actor commenting, the model output summaries suggest a positive correlation with resolution time, assignment time, and development time ($p < 0.001$), increased fix ($p < 0.05$) and patch ($p < 0.001$) emergence tendencies, increased reopening tendencies ($p < 0.001$) and increased confirmation tendencies ($p < 0.001$). The time threshold model output summaries reveal a distributed resolution time effect, with negative correlation with "extremely fast" resolution time ($p < 0.001$) and positive correlation with "very fast" ($p < 0.05$), "average" ($p < 0.001$), "very slow" ($p < 0.001$), and "extremely slow" ($p < 0.001$) resolution times. For assignment time, there is only negative correlation with "extremely fast" assignment ($p < 0.01$). No time thresholds are significant for development time.

Taken collectively, these results also lend partial support to the hypothesis of core knowledge actor comments being correlated with better solution knowledge emergence. As is the case for follows, the outcome effect is distributed, with increased fix and patch emergence tendencies coming at the cost of delayed resolution and increased reopening, with the latter plausibly being responsible for the delay in resolution giving the inconclusive mix of threshold timing correlations. Once again, comparison with the individual and organisation level results helps localize the level of the observed effect, as discussed in the next sections.

For peripheral knowledge actor follows, the regression model summaries suggest a correlation with increased resolution and development times ($p < 0.001$). The timing threshold variables suggest a negative correlation with “extremely fast” ($p < 0.001$) and “fast” ($p < 0.05$) resolution times and a positive correlation with “extremely slow” ($p < 0.001$) resolution time. The assignment and development time threshold models are not significant. This result is the opposite of expected and suggests that peripheral knowledge actor follows may actually have a negative effect on solution knowledge emergence by delaying resolution and development. However, these weak effect results support the component of hypothesis five that theorizes that peripheral knowledge actor stakeholder actions are less influential than core knowledge actor stakeholder actions. Yet, the nature of the effects and their differences across actor types are unclear at the problem level of analysis and may be more attributable to individual or organisation level effects, which are considered in the following sections.

For peripheral knowledge actor votes, the regression model summaries suggest negative correlation with resolution, assignment, and development times ($p < 0.001$), positive correlation with fix emergence (($p < 0.05$), negative correlation with reopening tendencies ($p < 0.05$), and positive correlation with confirmation tendencies ($p < 0.001$). The timing threshold model summaries suggest a negative correlation with “extremely fast” ($p < 0.05$) and “average” (p <

0.001) resolution times, and a negative correlation with “very slow” assignment ($p < 0.05$). These results partially support the hypothesis that peripheral knowledge actor votes improve solution knowledge emergence. Yet, the positive correlation with fix emergence tendencies in the absence of such effect for core knowledge actor votes contradicts the portion of the hypothesis that theorizes stronger core knowledge actor stakeholder influence on solution knowledge outcomes. The nature of influence with the voting action appears to differ from other types of actions.

For peripheral actor comments, the regression model summaries suggest a correlation with increased resolution and assignment times ($p < 0.001$), decreased fix emergence tendencies ($p < 0.01$), increased reopening tendencies ($p < 0.001$), and increased confirmation tendencies ($p < 0.001$). The timing threshold model summaries suggest a negative correlation with “extremely fast” ($p < 0.001$), “very fast” ($p < 0.001$), and “fast” ($p < 0.001$) resolution times, a positive correlation with “slow” ($p < 0.001$), “very slow” ($p < 0.001$), and “extremely slow” ($p < 0.001$) resolution times; a negative correlation with “extremely fast” assignment ($p < 0.01$) and a positive correlation with “very slow” ($p < 0.05$) and “extremely slow” ($p < 0.001$) assignment times; and, no significant effects for development time thresholds. These results are the opposite of hypothesized, suggesting that peripheral knowledge actor comments generally worsen solution knowledge emergence. The negative fix emergence and non-significant patch emergence tendencies are the opposite of what is observed for core knowledge actor comments, lending support to the notion of significant difference of influence amongst stakeholder types.

Comparison of the ANCOVA and base regression model summary results with the tabular results reveals that heteroskedasticity correction in the tabular results in numerous outcome effects that attained sufficient certainty of significance in the ANCOVA and base regression model summaries losing certainty of significance. This difference suggests significant non-linearities in the problem level effects of these knowledge stakeholder influence variables. A likely explanation for these non-linearities is that the primary knowledge stakeholder influence effects are not at the problem level of analysis but rather lie at the individual and/or organisation levels of analysis. Given that the independent variables are attempting to capture stakeholder-related factors, it stands to reason that individual level effects would dominate. The significant regression model non-linearities at the problem level are likely the projection of the individual level effects down to the problem level, resulting in the distributed and inconsistent results. Therefore, caution is suggested when interpreting the problem level results for these variables in isolation. They should only be considered in combination with the individual level results for the full picture of the effects of knowledge stakeholder influence on the outcomes of interest. Contextually, the only problem level specific takeaway that should be interpreted from the discussed results is a general difference in the effect of peripheral and core knowledge actor stakeholder follows, votes, and comments on outcomes of interest.

Measure four: Domain of the stakeholder who produced the problem knowledge

The fourth measure of knowledge stakeholder influence is the domain of the stakeholder who produced the problem knowledge. As discussed in Chapter Four: Research Method, the domain represents the organisation of which the individual stakeholder is a member and therefore represents the concept of organisation knowledge stakeholder. It is hypothesized that organisation knowledge stakeholders are heterogeneous in their influence on solution knowledge emergence.

Examination of the ANCOVA model summaries suggests that the domain of the stakeholder who produced the problem knowledge is correlated with resolution time ($p < 0.001$), assignment time ($p < 0.001$), development time ($p < 0.001$), fix ($p < 0.01$) and patch ($p < 0.001$) emergence tendencies, and confirmation tendencies ($p < 0.001$). It is also strongly correlated with most resolution, assignment, and development timing threshold outcome effects. As discussed in the section for measure one, the ANCOVA results are the most suitable for analyzing the effects of this variable given that its underlying data is made up of a large number of categories and the heteroskedasticity correction in the tabular model summaries incorrectly ignores significant effects as a result.

These results lend strong support to the hypothesis that organisation knowledge stakeholders are heterogeneous in terms of their influence on solution knowledge emergence. These results may be the first empirical evidence for an organisation level effect, which has not previously been reported in the literature that has typically focused only on individual level heterogeneity of influence (c.f. Hooimeijer & Weimer, 2007; Au et al., 2009; Guo, et al., 2010, Shihab, et al, 2010). These results lay the foundation for a novel research program centered on organisation level influence in meta-organisations.

Measure five: Whether domain of the stakeholder who produced the problem knowledge is a known webmail domain

The fifth measure of knowledge stakeholder influence is whether the domain of the stakeholder who produced the problem knowledge is a known webmail domain. As discussed in Chapter Four: Research Method, the domain represents the organisation of which the individual stakeholder is a member and therefore represents the concept of organisation knowledge stakeholder. The present variable separates those domains that are known to be webmail domains and therefore not likely representative of organisations. It complements the previous measure by aiding in determining what portion of the heterogeneity in the outcome effects is attributable to the actual organisation effects rather than potentially spurious effects that could emerge from the conversion of the former measure's categorical variable to numerical. At the problem level of analysis, the purpose of this variable is primarily verification of the validity of the underlying assumptions of the operationalizations. Based on these assumptions, it is hypothesized that domains that are not known webmail domains are correlated associated with better solution knowledge emergence because they more accurately represent organisation knowledge stakeholder influence than known webmail domains.

Examination of the regression model summaries suggests that problems with reporters whose domains are not known webmail domains are correlated with decreased resolution time ($p < 0.001$), decreased assignment time ($p < 0.001$), decreased development time ($p < 0.05$), increased fix and patch emergence tendencies ($p < 0.001$), and increased confirmation tendencies ($p < 0.001$). The timing threshold model summaries suggest a positive correlation with "extremely fast" resolution ($p < 0.01$) and a negative correlation with "extremely slow" resolution ($p < 0.001$); a positive correlation with "extremely fast" assignment ($p < 0.001$) and a negative correlation with "extremely slow" assignment ($p < 0.05$); and, no significant development timing threshold effects.   The ANCOVA model summaries suggest a broader range of significant timing threshold results that lose significance due to heteroskedasticity correction.

These results lend strong support to the hypothesis of better solution knowledge emergence for problems with reporters whose domains that are not known webmail domains. It also lends support to the validity of the underlying assumptions of the operationalization choices. Given that this variable is used as an inclusion criterion for the sample constraints at the organisation level of analysis, the present results suggest that it serves its intended purpose, improving the validity of the organisation level results. Whereas it is tempting to interpret the present results as evidence that organisation knowledge stakeholders exert stronger influence than individual knowledge stakeholders, the operationalization of the present variable precludes any such interpretation because known web mail domains do not necessarily correspond to individual actors. For this reason, this factor is only one of the sample inclusion factors for the organisation level analyses. Interpretations of relative individual and organisation level stakeholder effects are done in the following sections by comparing of the summaries of individual level, organisation level, and mixed-effects regression models.

Summary of dependent variable effect results at problem level

Comparison of the control and full regression models for the effect of knowledge stakeholder influence measures on the dependent outcomes of interest reveals an overall picture of effect size at the problem level of analysis. The model F & Chi-squared statistics as well as the comparative AIC delta statistics suggest that in all cases the knowledge stakeholder influence independent variable full models are significantly superior to the control variable only models, albeit only marginally so in the case of the several of the threshold assignment and development time models.

For resolution time, assignment time, and development time, the Cohen's additive f^2 effect sizes are "small-to-medium", "small-to-medium" and "small" respectively, with incremental R^2 values of 0.214 to 0.246, 0.091 to 0.132, and 0.382 to 0.393 respectively. For fix and patch emergence tendencies as well as reopening and reassigning tendencies, the additive effect sizes are "small" with incremental R^2 values of 0.582 to 0.591, 0.465 to 0.484, 0.095 to 0.112, and 0.352 to 0.364 respectively. And, for confirmation tendencies, the additive effect size is "large", with incremental R^2 values of 0.525 to 0.663. These effect size values, while significant, are all markedly lower than is the case for problem level effects observed in the models that tested previous hypotheses. This difference is not unexpected and follows given that the knowledge stakeholder influence effects are unlikely to be significant at the problem level and are more likely to lie at the individual and organisation levels of analysis.

In summary, the problem level knowledge stakeholder influence effects clearly demonstrate the heterogeneity of both individual and organisation knowledge stakeholders, as hypothesized. Further, core knowledge actors' and peripheral knowledge actors' follows, votes, and comments have different degrees of influence on solution knowledge emergence outcomes of interest. Yet, the overall problem level knowledge stakeholder influence effects are small and distributed in inconsistent patterns, suggesting that they are primarily the result of the projection of individual and/or organisation level effects, which are considered in the following sections.

Individual level of analysis results

At the individual level of analysis there are seven measures that represent the conceptualization of the dependent outcome of interest, solution knowledge emergence, as depicted in Figure 19, and as operationalized into the variables described in Table 79 and Table

87. Three measures represent the conceptualization of the antecedent of interest in hypothesis five, knowledge stakeholder influence, as depicted in Figure 24, and as operationalized into the variables described in Table 93. The results of the analyses are depicted in the regression model output summaries in Appendix D: Regression models.

As discussed in Chapter Four: Research Method, the regression models are separated according to the roles in which individuals engage when participating in the knowledge production process: problem knowledge producer (reporter), solution knowledge producer (assigned_to), and solution knowledge verifier (QA_contact).

Measure one: Whether profile is a core knowledge actor

The first measure of knowledge stakeholder influence at the individual level of analysis is whether the focal profile is a core knowledge actor. Core knowledge actor profiles are theorised to be correlated with better solution knowledge emergence outcomes due to the higher influence of core knowledge actor stakeholders in the meta-organisation's knowledge production process.

Examination of the regression model summaries reveals that problem knowledge producer role core knowledge actors are correlated with decreased reopening tendencies ($p < 0.001$) and decreased patch emergence tendencies ($p < 0.001$). Solution knowledge producer role and solution knowledge verifier role core knowledge actors are both correlated with decreased patch emergence tendencies ($p < 0.05$).

These results are the opposite of those observed at the problem level of analysis and largely the opposite of hypothesized. While core knowledge stakeholder actors are correlated with decreased reopening of problems in the case of the problem knowledge producer role, in all roles, core knowledge stakeholders are correlated with decreased patch emergence. One possible explanation is that core knowledge actors more frequently report problems whose solution does not entail patch type solution knowledge. The lack of significant effect on fix emergence tendencies supports this interpretation because were the patch emergence reduction a result of reduced overall solution knowledge emergence, a reduction in general fix emergence tendencies would be expected as well. Another possible explanation is that core knowledge actors are more overloaded than other actors and hence have lower absorptive capacity to curate new problem knowledge. This possibility is supported by both the absorptive capacity hypothesis results and the distracting nature of some of the non-core knowledge actor stakeholder effects discussed in the following sections.

This result provides novel empirical evidence at the individual level of analysis that contradicts the extant literature (c.f. Dalle, Besten, & Masmoudi, 2008; Dalle, et al., 2008; Guo, et al., 2010; Masmoudi, 2012), which suggests that core knowledge stakeholders have generally better solution knowledge emergence than peripheral knowledge stakeholders in meta-organisations. In fact, the opposite might be true, at least in the case of the reporter role. The responsibilities of core knowledge actors may detract for the emergence of solutions for the problems they submit.

Measure two: Mean number of follows, votes, and comments by core, peripheral, and participant knowledge actors on problems acted upon in each role

The second measure of knowledge stakeholder influence is the mean number of follows, votes, and comments by core, peripheral and participant knowledge actors on problems acted upon in each role. While all actions are theorized to be positive for solution knowledge emergence, actions by higher involvement knowledge actors, where core knowledge actors have higher involvement than participant knowledge actors and participant knowledge actors have higher involvement than peripheral knowledge actors, are theorized to have stronger effects on solution knowledge emergence than actions by lower status knowledge actors.

Examination of the regression model summaries reveals that, for the problem knowledge producer role, the mean number of follows by core knowledge actors is positively correlated with resolution time ($p < 0.001$) and negatively correlated with fix emergence ($p < 0.001$). The same positive resolution time ($p < 0.05$) and negative fix emergence tendencies ($p < 0.05$) correlations are observed in the results for the solution knowledge producer role as well.

The mean number of follows by participant knowledge actors for the problem knowledge producer role is correlated with reduced resolution time ($p < 0.001$), reduced reopening tendencies ($p < 0.001$), reduced reassigning tendencies ($p < 0.001$), and increased fix and patch emergence tendencies ($p < 0.001$). The reduced resolution time ($p < 0.001$), reduced reassignment tendencies ($p < 0.05$) and increased fix emergence tendencies ($p < 0.001$) hold for the solution knowledge producer role as well. The reduced resolution time ($p < 0.01$) is also observed for the solution knowledge verifier role.

The mean number of follows by peripheral knowledge actors for the problem knowledge producer role is only correlated with reduced reassignment ($p < 0.05$), though the correlation is not significant for the solution knowledge producer or solution knowledge verifier roles.

Taken collectively, these follow-related results lend partial support to the hypothesis. They also partially contradict the results observed at the problem level of analysis. On the one hand, follows by participant knowledge actors appear to improve solution knowledge emergence across a range of dimensions. On the other hand, follows by core knowledge actors, who are theorized to exert stronger influence on solution knowledge outcomes, appear to worsen solution knowledge emergence overall. As theorized, peripheral knowledge actor follows have a very small effect on outcomes. A possible explanation for these results is that core knowledge actor involvement results in more frequent rejection of problems due to the exclusionary influence of core knowledge participants that has been described in the literature (c.f. Crowston & Howison, 2005; Crowston, et al., 2006; Dahlander & O'Mahony, 2011; Dahlander & Frederiksen, 2012). Whereas participant knowledge actors may be more supportive of other participants' problems, core knowledge actors may be less supportive, preferring to focus meta-organisation knowledge creation energy on problems that they view as more important. The implications are that core knowledge actors have a strong influence and that they use this influence to hinder solution knowledge emergence. Further, the contrast with the problem level results suggests that core knowledge actors exert this negative influence at the individual level, hindering the solution knowledge emergence for certain individuals in particular, independent of the problem knowledge itself. Therefore, it may be that the core knowledge actors are prioritizing meta-organisation knowledge creation energy away from certain individuals that they view as not valuable contributors—not those problems that they view as valueless problems.

For the mean number of votes by core knowledge actors, in the case of the problem knowledge producer role, the results reveal a correlation with reduced resolution time ($p < 0.001$) and reduced patch emergence tendencies ($p < 0.05$). The reduced resolution time result is also observed for the solution knowledge producer role, but the reduced patch emergence tendency misses the certainty cut-off at $p \sim= 0.08$ for that role.

The mean number of votes by participant knowledge actors, for the problem knowledge producer role, there is correlation with increased resolution time ($p < 0.001$), increased reopening tendencies ($p < 0.01$), and decreased fix emergence tendencies ($p < 0.001$). For the solution knowledge producer role, the results reveal a correlation with increased resolution time ($p < 0.001$) and increased patch emergence tendencies ($p < 0.001$). The solution knowledge verifier role results also revealed a correlation with increased resolution time ($p < 0.001$).

For mean number of votes by peripheral knowledge actors, for the problem knowledge producer role, there is a correlation with reduced resolution time ($p < 0.001$) and increased fix emergence tendencies ($p < 0.001$). For the solution knowledge producer role, there is a correlation with decreased resolution time ($p < 0.001$) and decreased patch emergence tendencies ($p < 0.01$). Decreased fix emergence tendencies marginally miss the cut-off for sufficient certainty of significance with $p \sim= 0.07$. The solution knowledge verifier role results are not significant.

Taken collectively, these vote related results lend partial support to the hypothesis and partially match the results observed at the problem level of analysis. Yet, several results are unexpected. Whereas core knowledge actor votes uniformly speed up resolution time, as observed at the problem level of analysis, they are correlated with worse patch emergence tendencies for the problem knowledge producer role. As discussed for the core knowledge actor follow results, this result may be empirical evidence of an exclusionary influence effect whereby core knowledge actors primarily affect solution knowledge emergence by blocking the development of knowledge that benefits individuals that they do not view as valuable allocations of meta-organisational knowledge production effort. The correlation between votes by participant knowledge actors and reduced fix emergence for the problem knowledge producer role but increased patch emergence for the solution knowledge producer role supports this explanation. Solution knowledge producers are typically core knowledge actors, whereas problem knowledge producers, on average, are only participant knowledge actors. It follows that the observed effects would be different between core knowledge actor votes on problems submitted by participant knowledge actors and participant knowledge actor votes on problems acted upon (as solution knowledge producer) by core knowledge actors. Core knowledge actors influence by blocking whereas participant knowledge actors influence by promoting. The votes of peripheral knowledge actors who, by definition, have never been problem or solution knowledge producers, represent the broader meta-organisation community influence that is largely positive in terms of solution knowledge emergence when it comes to the problem knowledge producer role and irrelevant for the solution knowledge producer role, fitting the hypothesis of relative power of knowledge stakeholders. Core knowledge actors with strong influence have no need for votes from other actors to promote solution knowledge emergence—they simply drive the knowledge production process themselves. The correlation between participant knowledge actor votes and reduced fix emergence tendencies for the problem knowledge producer role then becomes clear: It is the clamoring of the "cosmopolitans" (Dahlander & Frederiksen, 2012: 988) that the core knowledge actors wish to stamp out. Too many votes by non-core yet involved participant actors represents a distraction in the focus of the meta-organisation and closing the associated problems as "WONTFIX" is often used as a way of redirecting attention to areas that the core knowledge actors would prefer (Ko & Chilana, 2010, 2011; Chilana, Ko, & Wobbrock, 2010; Guo, et al., 2011). Further, given that the effect is at the individual level, it may be the result of impressions that core knowledge actors form about the value of other actors in the meta-organisation (Marlow, Dabbish, & Herbsleb, 2013).

For the mean number of comments by core knowledge actors, in the case of the problem knowledge producer role, the results reveal a correlation with increased reopening tendencies ($p < 0.001$) and increased fix and patch emergence tendencies ($p < 0.001$). For the solution knowledge producer role, mean number of comments by core knowledge actors is correlated with increased resolution time ($p < 0.001$) and increased patch emergence tendencies ($p < 0.001$). For the solution knowledge verifier role, mean number of comments by core knowledge actors is correlated with increased fix and patch emergence tendencies ($p < 0.001$).

For the mean number of comments by participant knowledge actors, for the problem knowledge producer role, the results reveal a correlation with decreased reassigning tendencies ($p < 0.001$), and decreased fix ($p < 0.01$) and patch ($p < 0.001$) emergence tendencies. For the solution knowledge producer role, mean number of comments by participant knowledge actors is correlated with reduced resolution time and increased fix ($p < 0.01$) and patch ($p < 0.001$) emergence tendencies. Solution knowledge verifier role models do not achieve significance.

For the mean number of comments by peripheral knowledge actors, for the problem knowledge producer role, the results reveal a correlation with decreased reassignment tendencies ($p < 0.001$) and decreased fix emergence tendencies ($p < 0.001$). For the solution knowledge producer role, mean number of comments by peripheral knowledge actors is correlated with increased patch emergence ($p < 0.01$). Solution knowledge verifier role models do not achieve significance.

For the mean number of distinct commenters, for the problem knowledge producer role, the results suggest a correlation with increased resolution time ($p < 0.001$), increased reopening and reassigning tendencies ($p < 0.001$), and decreased fix and patch emergence tendencies ($p < 0.001$). For the solution knowledge producer role, these results all hold ($p < 0.001$). For the solution knowledge verifier role, a correlation with increased reassigning tendencies ($p < 0.001$) and decreased fix and patch emergence tendencies ($p < 0.001$) is observed.

Taken collectively, these results partially correspond with those observed at the problem level of analysis and lend partial support to the hypothesis. Comments by core knowledge actors are correlated with better solution knowledge emergence, albeit often at the cost of increased resolution time and/or increased reopening tendencies. In the case of comments, as opposed to votes, it appears that core knowledge actors exert their influence positively at the individual level when it comes to solution knowledge emergence. Whereas by not voting for the problems of individuals that they do not feel contribute to the meta-organisation reduces solution knowledge emergence with a negative core knowledge actor influence, proactively commenting on the problems of individuals that they feel do contribute to the meta-organisation increases solution knowledge emergence with a positive core knowledge actor influence.

As observed with votes, comments by participant knowledge actors have a split effect, reducing the emergence of solutions for problem knowledge producers, who are themselves typically participant knowledge actors, and increasing the emergence of solutions for solution knowledge producers, who are themselves typically core knowledge actors. This split provides strong evidence for the differential degree of influence exerted by core actors relative to other actors in the meta-organisation, as hypothesized. This interpretation is supported by the peripheral actor comments' correlations, which are similar but less powerful to those of participant knowledge actors, as theorized.

A significant divergence from the hypothesis and problem level results is observed primarily in the distinct commenter effects, which are correlated with worse solution knowledge emergence for all roles. One possible explanation is that problems with lots of different commenters are examples of "contentious" problem submissions (Ko & Chilana, 2010: 1666), which are hotly disputed in the meta-organisation. In particular, the involvement of non-core actors in the debate has been reported in the literature as a factor that can negatively influence solution knowledge emergence in highly-contentious cases (Ko & Chilana, 2010). In such cases, the lack of subject matter expertise of the peripheral knowledge actors results in misunderstandings of the technical requirements for creating the solution knowledge and detracts from the overall discussion by focusing on issues that core knowledge actors deem irrelevant (Ko & Chilana, 2010). Therefore, the individual level effect of comments on solution knowledge emergence is both a function of the number and the type of knowledge actor doing the commenting, with peripheral actor involvement being negative (rather than positive, but with weaker relative influence, as theorized) and core knowledge actor involvement being positive (with stronger relative influence, as theorized).

Measure three: Number and type of profiles and organisations watching and watched by each profile engaging in each role

The third measure of knowledge stakeholder influence is the number and type of profiles and organisations watching and watched by each profile engaging in each role. While both watching and being watched are theorized to be positive for solution knowledge emergence, watching and being watched by higher involvement knowledge actors, where core knowledge actors have higher involvement than participant knowledge actors and participant knowledge actors have higher involvement than peripheral knowledge actors, are theorized to have stronger effects on solution knowledge emergence than watching and being watched by lower involvement knowledge actors.

Examination of the regression model summaries reveals that, for the problem knowledge producer role, the effects of watching on outcomes of interest are limited to fix emergence tendencies and are split. Whereas the overall number of individual actors watched has no significant effect on solution knowledge emergence, watching a greater number of different core knowledge actors ($p < 0.01$), participant knowledge actors ($p < 0.05$), or peripheral knowledge actors ($p < 0.05$) is correlated with worse fix emergence tendencies. By contrast, watching a greater number of different organisations is correlated with better fix emergence tendencies ($p < 0.01$). Watching independent variables are broadly insignificant in the solution knowledge producer and solution knowledge verifier models. These results suggest that the benefits of watching other actors accrue not from a range of individual actors, of any type, but rather from a range of distinct organisations. Following many individuals from the same organisation may lead to negative solution knowledge emergence outcomes as a result of insufficient novel knowledge to better solve problems (Nickerson & Zenger, 2004). The present results offer novel empirical evidence of the effects of the range of knowledge seeking by individuals in meta-organisations, supporting the knowledge-based view of the firm theory developed by Nickerson and Zenger (2004).

The effects of a profile being watched by other actors are markedly different from the watching effects. For the problem knowledge producer role, the number of different actors the profile is watched by is correlated with reduced resolution time ($p < 0.001$) and increased fix ($p < 0.001$) and patch ($p < 0.05$) emergence tendencies. For the solution knowledge producer and solution knowledge verifier roles, the number of different actors the profile is watched by is correlated with increased resolution time ($p < 0.05$). Organisational watching does not achieve significance in any of the models.

Being watched by different types of actors has different effects, as theorized. For the problem knowledge producer role, being watched by core knowledge actors is correlated with better patch emergence tendencies ($p < 0.05$). Yet being watched by participant or peripheral knowledge actors is correlated with worse fix ($p < 0.001$) and patch ($p < 0.05$) emergence tendencies, and increased resolution time ($p < 0.001$) in the case of greater number of participant knowledge actors the profile is watched by.

The results are similar for the solution knowledge producer role, where being watched by participant or peripheral actors is correlated with reduced patch ($p < 0.05$) and reduced fix ($p < 0.01$) emergence tendencies respectively, with reduced fix emergence tendencies for being watched by participant actors only narrowly missing certainty of significance cut-off at $p < 0.06$. However, being watched by core knowledge actors is correlated with increased fix ($p < 0.05$) and patch ($p < 0.001$) emergence tendencies for the developer role. For the solution knowledge verifier role, being watched by peripheral knowledge actors is correlated with worse fix emergence tendencies ($p < 0.05$).

Taken collectively, the individual level effects of being watched by other knowledge actors are significantly stratified by type of actor doing the watching, as hypothesized. Yet, not all watching is positive, contrary to the hypothesis. Being watched by core knowledge actors is generally positive for solution knowledge emergence, suggesting a "vote of confidence" effect by those actors most knowledgeable in the meta-organisation. Whereas actual voting by core knowledge actors is used in an exclusionary fashion at the individual level, watching is used in an inclusionary fashion, highlighting those actors who tend to be associated with better solution knowledge emergence. By contrast, being followed by less knowledgeable actors is generally negative. For both the problem knowledge producer and solution knowledge producer roles, being watched by non-core knowledge actors is a distraction that hinders solution knowledge emergence. The influence of these types of stakeholders is largely negative in that it bogs down the knowledge production process and splits effort rather than unifying it due to the lack of knowledge and specialization of less involved users drawing the focus away from the necessary components of the solution knowledge (von Krogh, Spaeth, & Lakhani, 2003). These results provide broad-scale empirical evidence of the effects described by von Krogh, Spaeth, and Lakhani in their 2003 case study, increasing the generalizability of that case study evidence to a much broader longitudinal, multi-project context, albeit one still limited to a single meta-organisation. The contributions to practice are that policies requiring newer stakeholders to spend time learning before they can be actively involved in a meta-organisation's knowledge production process may be a beneficial barrier to entry that reduces the negative influence of less knowledgeable stakeholders on solution knowledge emergence.

Effects of individual nestedness in organisations

Examination of the mixed-effects regression model summaries reveals that there are significant organisational effects that affect the individual level results, albeit only marginally for most outcomes. For the problem knowledge producer role, the AIC and BIC delta statistics reveal that the mixed-effects models are marginally superior in all cases, with isolated organisation nestedness effect sizes ranging from "small" to "medium-to-large". For the solution knowledge producer role, the mixed-effects models revealed isolated organisation nestedness effect sizes ranging from "small" to "very large". For the solution knowledge verifier role, the isolated effect size for resolution time was "medium" and for reassignment tendencies was "large", with the remaining mixed-effects models not superior to the OLS models.

Despite the significant organisational nestedness effects, most of the individual-level level results remain significant after isolating the organisational random effects, with three notable exceptions. The first case is the case of the effect of comments by core and participant knowledge actors on reassigning tendencies, which lose significance in the mixed-effects models for problem knowledge producer; the solution knowledge producer role results also show a loss of significance for the effect of comments by core knowledge actors on patch emergence tendencies, suggesting that these effects might be primarily at the organisation level rather than the individual level.

The second case is related to the effects of problem knowledge producers being watched by different types of knowledge actors, which lose significance in the mixed-effects models for fix emergence tendencies. Once again, these watching effects may be more attributable to organisation level factors than individual level factors.

The third case questions the effect of participant knowledge actor votes on patch emergence for the solution knowledge producer role, once again suggesting that this effect may be at the organisation level rather than at the individual level of analysis. Examination of the organisation level results in the following section in combination with the present results provides a clearer picture of the overall knowledge stakeholder influence effects.

Summary of dependent variable effect results at individual level

Comparison of the control and full regression models for the effect of knowledge stakeholder influence measures on the dependent outcomes of interest reveals an overall picture of effect size at the individual level of analysis. The model F & Chi-squared statistics as well as the comparative AIC delta statistics suggest that in all cases the knowledge stakeholder influence independent variable full models are significantly superior to the control variable only models, albeit only marginally so in the case of the solution knowledge verifier role models.

For the problem knowledge producer role, for resolution time, the Cohen's additive f^2 effect size is "medium-to-large"; for reopening and reassigning tendencies they are "small-to-medium" and "large" respectively; and, for fix and patch emergence tendencies they are "medium-to-large" and "large" respectively. For the solution knowledge producer role, for resolution time, additive the effect size is "large"; for reopening and reassigning tendencies they are "medium" and "medium-to-large" respectively; and, for fix and patch emergence tendencies they are "medium" and "large" respectively. For the solution knowledge verifier role, for resolution time, the additive effect size is "large"; reopening is of questionable significance; reassigning has an effect size of "very large"; and, fix and patch emergence tendency effects are "medium" and "very large" respectively.

In summary, the effects of individual level knowledge stakeholder influence are best described as a balance between endorsement, rejection, and distraction, and core and non-core knowledge actors. Core knowledge actors reject non-core knowledge actors through votes and follows and endorse them through comments and watching; core knowledge actors endorse other core knowledge actors with watching and commenting and distract them with follows. Non-core knowledge actors reject other non-core knowledge actors with votes and comments and endorse them with follows; non-core knowledge actors endorse core knowledge actors with votes and follows and distract them with comments and watching. Core knowledge actors, by virtue of the responsibilities associated with their higher involvement, are more readily bogged down by extraneous, irrelevant knowledge, resulting in worse outcomes for the problems upon which they act.

These results paint a multidimensional picture of the individual level stakeholder influence effects and provide empirical evidence to support it. As reported by Murphy-Hill et al. (2013) in their individual-level interview and survey based study of knowledge actors in meta-organisations, there are a large number of individual stakeholder factors that influence the solution knowledge development process. The present results provide empirical evidence from the analysis of a larger database that lend support to their theories and expands the scope of observed effects.

Given that the mixed-effects model results show some organisation embeddedness effects, the results discussed above must be contextualized in the results of the organisation level of analysis discussed in the following section.

Organisation level of analysis results

At the organisation level of analysis there are seven measures that represent the conceptualization of the dependent outcome of interest, solution knowledge emergence, as depicted in Figure 26 and as operationalized into the variables described in Table 105. Four measures represent the conceptualization of the antecedent of interest in hypothesis five, knowledge stakeholder influence, as depicted in Figure 31, and as operationalized into the variables described in Table 111. The results of the analyses are depicted in the regression model output summaries in Appendix D: Regression models.

As discussed in Chapter Four: Research Method, regression models are separated according to the aggregate roles in which members of organisations engage when participating in the knowledge production process: aggregate problem knowledge producer (reporter) role, aggregate solution knowledge producer (assigned_to) role, and aggregate solution knowledge verifier (QA_contact) role.

Measure one: Count and percent of organisation members who are core, participant, and peripheral knowledge actors

The first measure of knowledge stakeholder influence at the organisation level of analysis is the count and percent of organisation members who are core, participant and peripheral knowledge actors. Organisations with higher number and percentage of higher involvement members, where core knowledge actors are more involved than participant knowledge actors and participant knowledge actors are more involved than peripheral knowledge actors, are theorized to have better solution knowledge emergence than organisations with fewer or lower percentages of high involvement members due to the higher influence of those higher involvement actors on the knowledge production process.

Examination of the regression model summaries reveals that the percentage and count of organisation members who are core, participant, and peripheral knowledge actors have no statistically significant effects on solution knowledge emergence in any aggregate role or measure. This result is surprising. Examination of the ANCOVA regression model summaries suggests plausible correlations between the number of core knowledge actors in an organisation and increased reopening and increased fix emergence tendencies, but these correlations fall well short of conventional standards of certainty of significance at $p \sim= 0.10$. If such effects do in fact exist, they are weak relative to other factors that affect solution knowledge emergence.

This result suggests that the effects of different types of involvement in the meta-organisation on solution knowledge emergence is most likely limited to the individual level of analysis. Further, taking into account the mixed-effects regression models, which suggest significant organisation-nestedness effects, it is likely that any organisation level effects are not representative of organisational heterogeneity as a whole, but rather representative of certain select organisations who have significant influence in the meta-organisations that are outliers relative to the majority of organisations included in the sample. This interpretation is supported by the preliminary analysis outcomes, discussed in Chapter Five: Analysis, which suggest that a small number of organisations do most of the knowledge production in the meta-organisation, with a long-tail effect of a large number of organisations each doing small amounts of knowledge production. Future research that uses data that span multiple meta-organisations may be fruitful in further refining organisation-level involvement effects. In particular, data from the organisation-driven Eclipse Foundation meta-organisation may be well suited to such a research agenda.

Measure two: Number of follows, votes, and activities by core, participant, and peripheral knowledge actors on problems acted upon by organisation members in each aggregate role

The second measure of knowledge stakeholder influence is the number of follows, votes, and activities by core, participant, and peripheral knowledge actors on problems acted upon by organisation members in each aggregate role. Organisations with tendencies to act upon problems with a greater number of follows, votes, and activities by higher involvement actors is theorized to be correlated with better solution knowledge emergence do to the higher influence on the knowledge production process of higher involvement actors.

Examination of the regression model summaries reveals that for the aggregate problem knowledge producer role, votes by core actors are correlated with reduced resolution time ($p < 0.001$) and decreased fix emergence tendencies ($p < 0.01$); votes by participant knowledge actors are correlated with increased resolution time ($p < 0.001$), with decreased fix emergence tendencies missing the standard levels of certainty of significance at $p \sim= 0.06$; and, votes by peripheral knowledge actors are correlated with decreased resolution time ($p < 0.001$).

For the aggregate solution knowledge producer role, votes by participant knowledge actors are correlated with increased resolution time ($p < 0.001$), with decreased fix emergence tendencies missing the standard levels of certainty of significance at $p \sim= 0.11$. Votes by peripheral knowledge actors were also correlated with decreased resolution time ($p < 0.05$). The results do not suggest a correlation with patch emergence tendencies, which, when interpreted in combination with the mixed-effects model results suggests that the effects of peripheral actor votes on patch emergence tendencies may be insignificant at both individual and organisation levels of analysis.

Taken collectively, these results largely match the results observed at the individual level of analysis and contradict the hypothesis, suggesting that votes by core knowledge actors are correlated with worse solution knowledge emergence. The results lend partial support to the notion that more involved knowledge actors have greater influence on solution knowledge emergence. In the case of votes, the results suggest that this influence is primarily negative, excluding solution knowledge emergence associated with problems knowledge submitted by organisations that are not deemed worthwhile by core knowledge stakeholders.

For follows, in the case of the aggregate problem knowledge producer role, number follows by core and peripheral knowledge actors are not significantly correlated with any outcomes. The number of follows by participant knowledge actors is correlated with increased resolution time ($p < 0.05$) and decreased reopening and reassigning tendencies ($p < 0.001$). For the aggregate solution knowledge producer role there is correlation between participant knowledge actor follows and reduced patch emergence tendencies ($p < 0.05$).

These results are markedly different from the results observed at the individual level of analysis, which are significant across a broader range of solution knowledge emergence outcomes. Interpreting these results in combination with the mixed-effects regression model results suggests that the effects of follows on solution knowledge emergence exist primarily at the individual level of analysis. A notable exception is the case of the effects of follows by participant knowledge actors on patch emergence tendencies for the aggregate solution knowledge producer role. Whereas increased fix emergence tendencies are observed at the individual level, decreased patch emergence tendencies are observed at the organisation level. This difference suggests that there are patch type solution knowledge specific factors that lie at the organisation level of analysis when it comes to the following tendencies of participant knowledge actors, who represent the influence of the broader meta-organisation community.

For activity, in the case of the aggregate problem knowledge producer role, the number of activities by core knowledge actors is correlated with increased resolution time ($p < 0.001$), decreased reopening tendencies ($p < 0.05$), and increased fix and patch emergence tendencies ($p < 0.001$). Activities by participant knowledge actors are correlated with reduced resolution time ($p < 0.01$), increased reopening tendencies ($p < 0.001$), and increased fix emergence tendencies ($p < 0.05$). Activities by peripheral knowledge actors do not achieve significance. For the solution knowledge producer role, activities by core and peripheral knowledge actors do not achieve significance; activities by participant knowledge actors are correlated with increased fix ($p < 0.01$) and patch ($p < 0.001$) emergence tendencies.

Taken collectively, these results largely support the hypothesis by suggesting that activities by more involved actors are positive for solution knowledge emergence, at least in the case of the aggregate problem knowledge producer role. A notable exception is in resolution time, where activities by core knowledge actors are correlated with increased resolution time, whereas activities by participant knowledge actors are correlated with reduced resolution time. Yet, the increased resolution time for core knowledge actor activity may be related to the higher complexity of problems with which core knowledge actors tend to engage given that the fix and patch emergence tendencies are overwhelmingly positive.

By contrast, the results for the aggregate solution knowledge producer role suggest that participant knowledge actor activities have a greater influence on fix and patch emergence, whereas the activities of core knowledge actors are not significant, contrary to as hypothesized. As seen at the individual level of analysis, it is clear that there are different effects between the types of actions: votes, follows, comments, and activities. The present results provide further evidence of the different effects amongst degrees of knowledge actor involvement. Whereas participant knowledge actor follows distract aggregate solution knowledge producers, worsening solution knowledge emergence, participant knowledge actor activities provide useful information that assists aggregate solution knowledge producers, improving solution knowledge emergence. In both of these cases, core knowledge actors have lower levels of influence on solution knowledge emergence despite increased influence in other cases.

The contributions to research are that it is important to stratify models of knowledge stakeholder influence factors using all three dimensions, i.e., type of action, knowledge actor involvement, and aggregate role, given that each permutation yields different outcome effects. The overuse of simplistic models in extant research may be responsible for the broad range of reported factors that are purported to affect solution knowledge emergence because these models insufficiently control for the interaction of certain factors. The contributions to practice are that simple measures underspecify the complexity of factors that affect solution knowledge emergence and meta-organisations should be careful to not enact policies based on single factor or single level results.

Measure three: Mean number of distinct actors commenting and acting on problems acted upon by organisation members in each aggregate role

The third measure of knowledge stakeholder influence is the mean number of distinct actors commenting and acting on problems acted upon by organisation members in each aggregate role. Organisations with tendencies to act upon problems with a greater number of distinct actors commenting and acting on them are theorized to be associated with better solution knowledge emergence due to the collective stakeholder influence of the greater number of distinct actors.

Examination of the regression model summaries reveals that, for the aggregate problem knowledge producer role, the mean number of distinct actors commenting is correlated with increased reopening tendencies ($p < 0.05$) and decreased patch emergence tendencies ($p < 0.01$). Reduced fix emergence tendencies reach a certainty of significance of p ~0= .10. For the aggregate solution knowledge producer role, the results are similar, with increased reopening ($p < 0.05$) and reassigning tendencies ($p < 0.01$), and decreased patch emergence tendencies only narrowly missing the typical cut-off for sufficient certainty of significance at p ~= 0.050.

For activities by distinct actors, the results suggest a plausible correlation with reduced reopening tendencies for both aggregate problem knowledge producer and aggregate solution knowledge producer roles, but it fails to achieve sufficient certainty of significance in both cases at $p \sim 0.10$. If such an effect exists, it is considerably weaker than other factors that affect solution knowledge emergence.

These results largely match those observed at the individual level of analysis and contradict the hypothesis by suggesting that the distinct number of commenters is correlated with worse solution knowledge emergence. Interpreting these results in combination with the mixed-effects regression model results suggests that the majority of the distinct actor comment effect on solution knowledge emergence lies at the individual level of analysis, with the organisation level effects being marginal at best.

Measure four: Number and type of profiles and organisations watching and watched by each organisation

The fourth measure of solution knowledge value is the number and type of profiles and organisations watching and watched by each organisation. While both the number of organisations watching and being watched by are theorized to be correlated with positive solution knowledge emergence at the organisation level of analysis, organisations watching and being watched by higher involvement knowledge actors, where core knowledge actors have higher involvement than participant knowledge actors and participant knowledge actors have higher involvement than peripheral knowledge actors, are theorized to have stronger effects on solution knowledge emergence than organisations watching and being watched by lower involvement knowledge actors due to the higher influence of the higher involvement knowledge actors.

Examination of the regression model summaries reveals that, for the aggregate problem knowledge producer role, the overall number of actors the organisation is watching is correlated with decreased resolution time ($p < 0.05$) and the overall number of other organisations the organisation is watching is correlated with increased fix emergence tendencies ($p < 0.01$). The number of core or participant knowledge actors that the organisation is watching are not significantly correlated with any outcomes of interest. None of the watching factors are significantly correlated with solution knowledge emergence outcomes for the solution knowledge producer or solution knowledge verifier roles.

These results largely match the results observed at the individual level of analysis, suggesting that watching other knowledge actors is only primarily useful for improving solution knowledge emergence when watching a broad range of knowledge actors who are members of multiple organisations. When watching other actors, breadth is preferable over depth. Comparison of the present results with the mixed-effects regression model results suggests that there is an organisation level effect for increased fix emergence tendencies correlated with the number of organisations watched, above and beyond the individual level effect. The effect is limited to the (aggregate) problem knowledge producer role.

These results provide novel empirical evidence for organisation level experience effects derived from observing other organisations (c.f. Carayannopoulos & Auster, 2010). Future research may wish to further refine the examination of the organisation experience effects by considering the change in outcome effects over time amongst and between organisations based on the nature of watching dyads and testing the induced isomorphism organisational knowledge adoption theory of Alexy, George, and Salter (2013).

For the effects of organisations being watched by other actors or organisations, examination of the regression model summaries reveals that, for the aggregate problem knowledge producer role, the number of overall actors watched by is correlated with increased resolution time ($p < 0.05$), yet the number of participant knowledge actors is correlated with reduced resolution time ($p < 0.01$). The number of organisations that the focal organisation is watched by and the number of core actors that the focal organisation is watched by have no significant effects. For the aggregate solution knowledge producer role, the overall number of actors that the focal organisation is watched by is correlated with worse fix emergence tendencies ($p < 0.05$). No other significant effects are observed.

These results are markedly different from those observed at the individual level of analysis, where being watched by knowledge actors has significant and sizeable effects on solution knowledge emergence. Examination of the mixed-effects regression models in combination with the organisation and individual level results suggest that the effects of being watched by actors on resolution time lie primarily at the individual level of analysis, with the organisation level effects being marginal at best. By contrast, the negative effects of being watched by knowledge actors on fix emergence tendencies for the aggregate solution knowledge producer role lie primarily at the organisation level of analysis. It is the developers in organisations as a whole who are distracted by being watched by too many knowledge stakeholders, whereas individual developers are able to avoid such distraction with their personal prioritization in the allocation of their knowledge production efforts.

Summary of dependent variable effect results at organisation level

Comparison of the control and full regression models for the effect of knowledge stakeholder influence independent variables on the dependent outcomes of interest reveals an overall picture of effect size at the organisation level of analysis. For the aggregate problem knowledge producer role, the comparative Chi-squared statistic and comparative AIC delta statistic reveal that the models that include the knowledge stakeholder influence variables are all superior to the control-only models. The additive effect sizes of the independent variables on the outcomes of interest above and beyond the control variables are “medium-to-large” for resolution time, “medium-to-large” for reopening tendencies, “large” for reassigning tendencies, and “large” for fix and patch emergence tendencies.

For the aggregate solution knowledge producer role, the comparative Chi-squared statistic and comparative AIC delta statistic reveal that the models that include the knowledge stakeholder influence variables are superior for the resolution time and fix and patch emergence tendencies outcomes, albeit only marginally so for fix emergence tendencies. The reopening and reassigning tendency models are not superior to the control models. The additive effect sizes on the outcomes of interest of the independent variables above and beyond the control variables are “very large” for resolution time, and “large” and “very large” for fix and patch emergence tendencies respectively. As is the case in previous results, the small sample size for the aggregate solution knowledge verifier role results in insignificant models across all outcome variables.

In summary, the organisation level effects of the knowledge stakeholder influence independent variables largely match those observed at the individual level of analysis with a few notable exceptions. The number and percent of organisation members with higher degrees of involvement in the meta organisation is not significantly associated with solution knowledge emergence; core knowledge actors use votes to exert negative influence on the emergence of solution knowledge for certain organisations; aggregate solution knowledge producers are negatively affected by being watched by or having their problems followed by large numbers of participant knowledge actors; activities are generally positively correlated with solution knowledge emergence at the organisation level of analysis above and beyond individual level effects, with activities by more involved actors being positive for aggregate problem knowledge producers and activities by moderately involved actors being positive for aggregate solution knowledge producers; the distinct number of commenters and actors has no, or marginal at best, effect at the organisation level of analysis; and, watching a larger number of other organisations is correlated with organisation level specific increased solution knowledge emergence tendencies, suggesting a form of organisational learning.

Hypothesis six: Solution knowledge value

The sixth hypothesis postulates that, "*Solution knowledge value is positively correlated with solution knowledge emergence.*" This hypothesis is tested by analyzing data at the problem, individual, and organisation levels of analysis; cross-level nesting effects between individuals and the organisations of which they are a member are also assessed. The results for each level are discussed in turn in the following sections.

Problem level of analysis results

At the problem level of analysis there are seven measures that represent the conceptualization of the dependent outcome of interest, solution knowledge emergence, as depicted in Figure 11, and as operationalized into the variables described in Table 51. Four measures represent the conceptualization of the antecedent of interest in hypothesis six, solution knowledge value, as depicted in Figure 18, and as operationalized into the variables described in Table 57. The results of the analyses are depicted in the regression model output summaries in Appendix D: Regression models.

Measure one: Severity and priority levels

The first measure of solution knowledge value at the problem level of analysis is the severity and priority levels. It is hypothesized that higher severity and priority levels are positively correlated with solution knowledge emergence due to the higher value of the solution knowledge designated by the higher severity and priority levels.

Examination of the ANCOVA output summaries suggests that severity and priority levels are strongly correlated with overall resolution time ($p < 0.001$), assignment time ($p < 0.001$), development time ($p < 0.001$), fix and patch emergence tendencies ($p < 0.001$), reopening tendencies ($p < 0.001$ for severity and $p < 0.05$ for priority), reassigning tendencies ($p < 0.001$), and confirmation tendencies ($p < 0.001$). Severity and priority are also strongly correlated ($p < 0.001$) with each of the time based threshold outcomes for resolution time, assignment time, and development time.

Examination of the dummy regression model output summaries for the relative effects of different severity levels reveals that relative to the highest severity category, “blocker”, all other severity levels, ranging from “trivial” through “critical” are strongly correlated with increased resolution time ($p < 0.001$), increased assignment time ($p < 0.001$), and increased development time ($p < 0.001$), validating that “blocker” class severity problems—those that are designated as preventing the solution knowledge emergence of other problems—are generally resolved much faster than other severity level problems. Problems with “normal” ($p < 0.01$), “minor” ($p < 0.001$), “major” ($p < 0.001$), “critical” ($p < 0.001$), and “enhancement” ($p < 0.001$) severity levels are all correlated with worse fix emergence tendencies relative to problems with the “blocker” severity level. A notable exception is problems with severity level “trivial”, which are correlated with better fix emergence tendencies (($p < 0.05$) relative to “blocker” severity level problems. Further, while it is only weak evidence, the relative weights of the coefficients as compared to the reference “blocker” severity level for fix emergence tendencies suggest a general ordering of the degree of effects that are consistent with the magnitude of the severity levels. In terms of patch emergence tendencies, relative to reference severity level “blocker”, “critical” level problems are correlated with worse patch emergence ($p < 0.05$), and “trivial”, “normal”, “minor”, and “enhancement” severity level problems are correlated with better patch emergence tendencies ($p < 0.001$). Severity level “major” is not significantly different from reference severity level “blocker” in terms of patch emergence tendencies.

With the exception of problems with severity level “normal”, which are correlated with increased reassigning tendencies ($p < 0.05$) relative to “blocker” severity level problems, there are no significant differences between severity levels’ correlations with reopening or reassigning tendencies. Relative to reference severity level “blocker”, problems of other severity levels are all correlated with worse confirmation tendencies ($p < 0.001$ except “normal” which was $p < 0.05$).

The timing threshold regression model summaries suggest a roughly even distribution of negative correlation with faster and positive correlation with slower resolution, assignment, and development times for all severity levels relative to severity level “blocker”. Once again, the relative levels of coefficient contributions generally suggest that the severity level ordering is consistent and increasing according to the labels, lending support to the validity of the underlying constructs.

Collectively, these severity level results largely support the hypothesis and suggest that higher severity level problems are correlated with better solution emergence tendencies. A couple of notable exceptions to this trend paint a clearer picture of the full nature of the effects. “Trivial” severity level problems are correlated with increased fix tendencies relative to "blocker" severity level problems. This result is contrary to the hypothesis and notably different from all the other severity levels. It suggests a u-shape relationship between severity level and fix emergence tendencies, where very low severity level problems, i.e., "trivial", are so easily solved that solution knowledge emerges for them more readily than for other, more severe problems. Once past this "trivial" threshold, however, the most severe problems have the highest fix emergence tendencies. The solution knowledge for "blocker" severity level problems is particularly valuable in that respect, as it enables solution knowledge to be developed for other problems that the "blocker" severity level problems are blocking. It follows that they have the highest fix emergence tendencies.

The case of patch emergence tendencies is similarly u-shaped, albeit with a higher inflection point. "Trivial", "enhancement", "minor", and "normal" severity level problems have increased patch emergence relative to "blocker" severity level problems. At the lower end of severity, patch emergence is higher, likely because the amount of effort to produce the patch type solution knowledge is lower. At the "major" severity level, there is an inflection point towards negative, with "critical" also being negative. The "blocker" reference category sits around the same patch emergence tendencies as the central severity levels for patch emergence tendencies, suggesting a trade-off between solution knowledge value and effort affecting patch emergence tendency outcomes. Higher severity level problems require significantly greater levels of effort to resolve such that patch emergence is worse for higher severity problems; yet, the urgency of "blocker" severity level problems preventing solution knowledge from being developed for other problems offsets this effect such that the net effect is comparable to mid-level severity problems on patch emergence tendencies. The contributions to theory are that the effects of severity levels of problems are markedly different depending on the type of solution knowledge required to address the problem.

Examination of the dummy regression model output summaries allows comparison of the relative effects of different priority levels on solution knowledge emergence outcomes of interest. The reference priority level is “no priority”, which is the default for all problems when no priority is specifically designated. Priorities are typically only designated on the small subset of problems whose severity level is set to “enhancement”. As such, priority levels should be interpreted as meta-organisational priorities for new things. The “no priority” reference category, therefore, necessarily has a selection bias of problems with higher severity levels than “enhancement”, though it includes a range of severities from “trivial” to “blocker” that may tend to average out. The reference category for priority dummy variables can be said to be the “average problem”. Relative priority level results are interpreted accordingly.

The regression model summaries reveal that, relative to the reference priority level “no priority”, “P1” (highest) priority problems are correlated with reduced resolution time ($p < 0.001$), and priority levels “P2” through “P5” (lowest) are correlated with successively greater increased resolution times ($p < 0.001$). Assignment time for priority “P1” is not significantly different from reference priority level “no priority”. Yet assignment time is significantly slower, in an increasing fashion, for priority levels “P2” through “P5”. All priority levels are correlated with increased development time relative to the reference “no priority” level, in an increasing fashion ($p < 0.001$) from “P1” to “P5”.

In terms of fix emergence tendencies, relative to “no priority” problems, “P1” priority level problems are correlated with the most increased fix emergence tendencies, and priority level “P5” problems are correlated with the most reduced fix emergence tendencies ($p < 0.001$). The inflection point shifts from increased to decreased fix emergence between priority levels “P3” and “P4”. The results are similar for patch emergence tendencies, with an earlier inflection point, such that priority level “P1” and “P2” problems are correlated with increased patch emergence tendencies, and priorities “P3” through “P5” are correlated with increasingly worse patch emergence tendencies ($p < 0.001$).

Reopening and reassigning tendencies are not markedly different between priority levels and the reference category. Confirmation tendencies are higher for all priority levels in a roughly uniform, rather than increasing fashion relative to the reference category ($p < 0.001$). Examination of the timing threshold models suggests a general tendency of positive and negative correlations that correspond with the overall resolution, assignment, and development times for each priority level. Further, the magnitude of effects appears to be ordered according to priority levels relative to the reference priority levels, as observed in other measures.

Collectively, the priority level results largely support the hypothesis in that problems with higher priority levels generally have better solution knowledge emergence across all outcomes of interest. Further, the results suggest there are differing inflection points for the value of different priority levels relative to the reference “average” problem in the meta-organisation. “P1” priority level problems are resolved faster than average, but “P2” priority level problems are resolved slower than average. All problems with set priority levels are developed slower than problems with no set priority level, which validates that “enhancement” severity level—necessary for priority levels being set—is generally considered the lowest severity level of the ordinal category.

Notably, problems with priority levels “P1” and “P2” have better fix and patch emergence tendencies than “average” problems despite being only of “enhancement” severity level. This result suggests that certain priority levels may have fix emergence tendencies that are closer to those of higher severity levels than “enhancement” as a result of the prioritization process. The contributions to theory are that severity and priority levels likely interact in their influence on solution knowledge emergence. Future research using a severity and priority level stratified sample of problems may be fruitful in further refining our understanding of the interactive nature of the effects of severity and priority levels on solution knowledge emergence outcomes.

Measure two: Number of severity and priority level changes after problem knowledge submission

The second measure of solution knowledge value is the number of severity and priority level changes that take place after the initial submission of the problem knowledge to the meta-organisation. It is hypothesized that the number of solution and priority changes is positively correlated with solution knowledge emergence due to the refined tuning of the severity and priority levels more accurately reflecting the value of the solution knowledge.

Examination of the regression model output summaries reveals that the number severity changes is positively correlated with resolution time ($p < 0.001$), assignment time ($p < 0.001$), and development time ($p < 0.05$). It is also positively correlated with reopening and reassigning tendencies ($p < 0.001$). It is not significantly correlated with fix or patch emergence tendencies.

These severity level change related results are largely contrary to the hypothesis, which suggests that severity level changes improve solution knowledge emergence. Instead, severity level changes appear to be related to delays in the knowledge production process without any notable effect on fix or patch emergence despite these delays.

For the number of priority changes, examination of the regression model output summaries reveals a positive correlation with resolution time ($p < 0.001$), development time ($p < 0.001$), patch emergence tendencies ($p < 0.001$), and reopening ($p < 0.05$) and reassigning ($p < 0.001$) tendencies. These results are markedly different from the severity change results and lend partial support to the hypothesis. In the case of priority changes, there is also correlation with delays in the knowledge production process, but these delays pay off in terms of increased patch emergence tendencies.

The implication of these results for theory is that changes in severity and priority levels have different impacts on solution knowledge emergence, with the former being uniformly negative and the latter being positive for patch type solution knowledge emergence, albeit at the cost of delays. The contributions to practice are that changing severity levels, which is commonly done by more involved knowledge actors, who are believed to be able to better judge the severity of problems than less involved knowledge actors (c.f. Lewis, et al., 2013; Zhou, Neamtiu, & Gupta, 2015), may have detrimental effects on solution knowledge emergence. Meta-organisational policies should carefully weigh the degree of incorrectness of severity levels deemed in need of changing against the negative impact of the change. In some cases, leaving only slightly “incorrect” severity levels unchanged may be the better policy for promoting solution knowledge emergence.

Measure three: Whether problem has one or more top 3, 10, 25, or 50 keywords

The third measure of solution knowledge value is whether each focal problem has one of the top 3, top 10, top 25, or top 50 keywords used in the meta-organisation. It is hypothesized that problems with more frequently used keywords are correlated with better solution knowledge emergence due to the increased solution knowledge value that these top keywords designate.

Examination of the regression model output summaries reveals that the effects of top keywords are inconsistent and sometimes contradictory on outcomes of interest. For resolution time, problems with top 3 keywords are correlated with reduced resolution time ($p < 0.001$). But problems with top 10 keywords and top 50 keywords are correlated with increased resolution time ($p < 0.001$) and problems with top 25 keywords do not have significantly different resolution times. Similarly, problems with top 3 keywords ($p < 0.001$) and top 25 keywords ($p < 0.001$) are correlated with increased assignment time, whereas problems with top 10 keywords and top 50 keywords have do not have statistically different assignment times. Problems with top 3 keywords and top 25 keywords are correlated with reduced development time ($p < 0.001$) and problems with top 50 keywords are correlated with increased development time. Problems with top 10 keywords do not have statistically different development times.

In terms of fix and patch emergence tendencies, problems with top 3 keywords do not have significantly different fix or patch emergence tendencies; problems with top 10 keywords are correlated with reduced fix and patch emergence tendencies ($p < 0.001$); problems with top 25 keywords are correlated with increased fix emergence ($p < 0.01$) and patch emergence ($p < 0.05$) tendencies; and, problems with top 50 keywords are correlated with increased fix ($p < 0.05$) and patch emergence ($p < 0.001$) tendencies.

Reopening and reassigning tendencies are not significantly different for most keyword tiers with the exception of problems with top 3 keywords, which are correlated with decreased reassignment tendencies ($p < 0.01$). All top keyword tiers are correlated with increased confirmation tendencies ($p < 0.001$).

Examination of the timing threshold regression model summaries reveals a general pattern for resolution time threshold correlations for each keyword tier that is consistent with the single variable resolution time for the corresponding tier. For assignment time thresholds, only the top 3 tier has significant correlations, with a strong negative correlation with "extremely fast" assignment ($p < 0.001$), moderately positive correlations with "very fast" and "fast" assignment ($p < 0.01$), and negative correlation with "extremely slow" assignment ($p < 0.05$) resulting in the net increased assignment time observed in the single continuous variable model. The top keyword tiers are all insignificant in the development time threshold variable models.

Taken collectively, these results do not lend consistent support to the hypothesis that problems with top keywords are correlated with better solution knowledge emergence. One possible explanation is that the effects of top keywords is cubic in nature, such that problems with top 3 through top 10 keywords are correlated with worse solution knowledge outcomes, problems with top 25 through top 50 keywords are correlated with better solution knowledge outcomes, and the remaining problems have relatively worse solution knowledge emergence outcomes. While this explanation fits the observed results, there are no known theoretical mechanisms that explain why such an effect would be cubic in nature, making that explanation tantamount to "data dredging", a contentious practice in management research (Goodman & Kruger, 1988; Bedeian, Taylor, & Miller, 2010; Woodside, 2016) that is carefully avoided in the design and methods of the present study. A more plausible explanation is that keyword popularity does not reliably signal solution knowledge value. Rather, it may be correlated with the signaling of some other property of knowledge that affects solution outcomes in a way that is not ordinal relative to keyword popularity. Future research that focuses specifically on keyword effects may wish to compare specific keyword clusters or clouds and keyword associations based on topics, synonyms and other linguistic factors in order to identify the type of knowledge that keywords provide in the knowledge production process. As a result of the observed inconsistencies, the conservative interpretive conclusion is a general lack of support for the top keyword sub-hypothesis.

Measure four: Overall number of follows and votes on problem

The fourth measure of solution knowledge value is the overall number of follows and votes on each focal problem. It is hypothesized that problems with more follows and votes are correlated with better solution knowledge emergence due to the increased value of the solution knowledge signalled by the following and voting activities.

Examination of the regression model output summaries reveals that the number of follows on problems is positively correlated with resolution time ($p < 0.001$), assignment time ($p < 0.001$), development time ($p < 0.001$), fix and patch emergence tendencies ($p < 0.001$), reopening, reassigning, and confirmation tendencies ($p < 0.001$). By contrast, the number of votes on problems is negatively correlated with resolution time ($p < 0.001$), assignment time ($p < 0.001$), development time ($p < 0.001$), and fix and patch emergence tendencies ($p < 0.05$). The timing threshold regression model results for both variables reveal a fairly consistent distribution of negative and positive correlations with fast and slow resolution, assignment, and development timing thresholds, the net of which corresponds with the continuous timing variable results.

The results for follows partially support the hypothesis, as greater number of follows is correlated with increased fix and patch emergence. Yet, this increased solution knowledge emergence comes at the cost of delays in assignment and development, and loops in the knowledge development life cycle, such that the overall resolution time is longer.

The results for votes is contrary to the hypothesis, as greater number of votes is correlated with worse fix and patch emergence tendencies. In this context, the faster resolution time correlation should be interpreted as "faster rejection". These results make significant contributions to both theory and practice, as they suggest that the voting mechanism, when assessed purely numerically, at the problem level of analysis, independent of individual and organisation level stakeholder influences, operates contrary to its intended purpose. The goal of voting on problems is problems with more votes having higher solution knowledge emergence tendencies. The results suggest an opposite effect. Comparison with individual and organisation level results provides clearer triangulation of the level the observed effects.

Summary of dependent variable effect results at problem level

Comparison of the control and full regression models for the effect of solution knowledge value measures on the dependent outcomes of interest reveals an overall picture of effect size at the problem level of analysis. The model F & Chi-squared statistics as well as the comparative AIC delta statistics suggest that in all cases the solution knowledge value independent variable full models are significantly superior to the control variable only models, albeit only marginally for most models.

For resolution time, assignment time, and development time, the Cohen's additive f^2 effect sizes are "small-to-medium", "small" and "small" respectively, with incremental R^2 values of 0.182 to 0.229, 0.07 to 0.102, and 0.375 to 0.393 respectively. For fix and patch emergence tendencies as well as reopening and reassigning tendencies, the additive effect sizes are "small" with incremental R^2 values of 0.580 to 0.592, 0.463 to 0.480, 0.090 to 0.107, and 0.350 to 0.369 respectively. And, for confirmation tendencies, the additive effect size is "medium", with incremental R^2 values of 0.520 to 0.577. These effect size values, while significant, are all markedly lower than is the case for problem level effects observed in the models that tested hypotheses one through four and are comparable to the effects observed for hypothesis five models. This difference is not unexpected and follows given that the solution knowledge value is relative to individual and organisation stakeholders, and, as such, solution knowledge value effects are unlikely to be significant at the problem level and are more likely to reside at the individual and organisation levels of analysis.

In summary, the problem level solution knowledge value independent variable effects show marked differences stratified according to the ordered severity and priority levels of problems, number of severity level changes, and number of follows and votes. By contrast, the popularity of keywords is not consistently correlated with solution knowledge emergence, suggesting that keyword popularity may impart knowledge that is not related to solution knowledge value signaling.

Individual level of analysis results

At the individual level of analysis there are seven measures that represent the conceptualization of the dependent outcome of interest, solution knowledge emergence, as depicted in Figure 19, and as operationalized into the variables described in Table 79 and Table 87. Four measures represent the conceptualization of the antecedent of interest in hypothesis six, solution knowledge value, as depicted in Figure 25, and as operationalized into the variables described in Table 94. The results of the analyses are depicted in the regression model output summaries in Appendix D: Regression models.

As discussed in Chapter Four: Research Method, the regression models are separated according to the roles in which individuals engage when participating in the knowledge production process: problem knowledge producer (reporter), solution knowledge producer (assigned_to), and solution knowledge verifier (QA_contact).

Measure one: Percent of problems of each severity and priority level acted upon in each role

The first measure of solution knowledge value at the individual level of analysis is the percent of problems of each severity and priority level acted upon in each role. A tendency to act upon problems with higher severity and priority levels is theorised to be correlated with better solution knowledge emergence outcomes due to the higher value of the solution knowledge of higher severity and priority level problems.

Examination of the regression model summaries reveals that, for the problem knowledge producer role, the percent of bugs reported with each severity level other than minor is negatively correlated with resolution time ($p < 0.001$ except "trivial" which is $p < 0.01$). No significant correlations are observed for reopening or reassigning tendencies. For fix emergence tendencies, a u-shaped response is observed based on severity levels. Acting on higher percentages of problems with "trivial" ($p < 0.001$) and "minor" ($p < 0.05$) severity levels is correlated with increased fix emergence tendencies; acting on "normal" severity level problems is not correlated associated with different fix emergence tendencies of other problems; acting on "major" and "critical" severity level problems is correlated with worse fix emergence tendencies ($p < 0.001$); and, acting on "blocker" severity level problems is correlated with better fix emergence tendencies ($p < 0.001$). The patch emergence tendencies are similar, but with a much flatter u-shape such that the positive ends drop below significance thresholds. Acting on "normal" ($p < 0.05$), "major" ($p < 0.001$), and "critical" ($p < 0.001$) problems is correlated with worse patch emergence tendencies. The coefficients for "trivial", "minor" and "blocker" match the u-shape observed for fix emergence tendencies but fail to achieve significance.

For the solution knowledge producer role, the results are similar in direction to the problem knowledge producer role albeit with weaker overall effects. For resolution time, acting on higher percentages of "normal" ($p < 0.001$), and "critical" ($p < 0.05$) severity level problems is correlated with reduced resolution time. For fix emergence tendencies, acting on "trivial" ($p < 0.001$), "minor" ($p < 0.01$), and "normal" ($p < 0.001$) severity problems is correlated with increased fix emergence and acting on "critical" ($p < 0.05$) severity problems is correlated with decreased fix emergence tendencies. As observed in the results for the problem knowledge producer role, acting on a greater percentage of "blocker" is correlated with a positive fix emergence tendency coefficient, similarly suggesting a u-shape relationship, although given the lower sample size inherent to the developer role, the coefficient fails to achieve standard levels of certainty of significance at $p \sim= 0.08$. For patch emergence tendencies, there is a positive correlation with greater percentage of "trivial" severity level problems acted upon. While the remaining coefficients largely match the direction and magnitude of those observed for the reporter role, they all fail to achieve sufficient certainty significance by large margins.

For the solution knowledge verifier role, a greater percentage of actions on problems with "normal" severity level is correlated with better fix emergence tendencies ($p < 0.01$) but worse patch emergence tendencies ($p < 0.05$); percentage of actions on problems with "blocker" severity level is correlated with both better fix ($p < 0.001$) and patch ($p < 0.05$) emergence tendencies; and, percentage of actions on problems with "major" severity level is correlated with reduced patch ($p < 0.05$) emergence tendencies.

Taken collectively, these results offer partial support for the hypothesis that involvement with higher severity level problems is correlated with better solution knowledge emergence. However, the relationship appears to be quadratic rather than linear, with both a higher percentage of actions on very low severity problems, i.e., "trivial", and a higher percentage of actions on very high severity problems, i.e., "blocker", correlated with increased solution knowledge emergence, and a higher percentage of actions on middle severity level problems correlated with worse solution knowledge emergence. A notable exception is the solution knowledge verifier role, where focus on any specific severity level other than "blocker" appears to pay off in better fix emergence tendencies but worse patch emergence tendencies. It may be that solution knowledge verifiers who focus on a narrow range of problem severities perform better when there is no patch type knowledge in the solution and perform worse when there is.

In terms of percentages of actions on problems of different priority levels, for the problem knowledge producer role, the regression model summaries reveal that a higher percentage of actions on any priority of bugs is generally correlated with faster resolution time ("P1" $p < 0.001$, "P3" $p < 0.001$, and "P5 $p < 0.05$; "P2" and "P4" not significant), increased reopening tendencies ($p < 0.001$ for all priority levels), and increased fix emergence tendencies

($p < 0.001$ except "P5" which was at $p \sim= 0.11$). Patch emergence tendencies are largely insignificant with the exception of priority level "P4" which is associated with worse patch emergence tendencies ($p < 0.01$).

For the solution knowledge producer role, greater percentage of acting on problems with priority levels "P1" ($p < 0.001$) and "P3" ($p < 0.01$) are correlated with decreased resolution time and greater percentage of acting on problems with priority level "P5" ($p < 0.05$) is correlated with increased resolution time. Priority level "P5" is correlated with increased reopening tendencies ($p < 0.05$) and priority level "P3" is correlated with increased reassigning tendencies ($p < 0.001$). Fix emergence tendencies are positive for "P2" ($p < 0.05$) and negative for "P3" ($p < 0.001$). Patch emergence tendencies are broadly negative for higher percentage of actions on problems with each priority levels ($p < 0.05$ to $p < 0.01$) except priority level "P3" which is not significant.

For the solution knowledge verifier role, greater percentage of acting on problems with priority level "P1" is correlated with reduced resolution time ($p < 0.05$), and increased reassigning tendencies ($p < 0.01$). Percentage of actions on problems with priority levels "P2" ($p < 0.001$) and "P5" ($p < 0.05$) are correlated with increased fix emergence tendencies.

Taken collectively, these results suggest partial support for the hypothesis that focusing on higher priority problems is correlated with better solution knowledge emergence. For the problem knowledge producer role, focus rather than breadth pays off in terms of fix emergence tendencies, with coefficients roughly ordered according to priority level. Yet, this effect is largely the opposite for the solution knowledge producer role, where focus on particular priority levels is correlated with worse patch type solution knowledge emergence tendencies roughly evenly across different priority levels. The solution knowledge verifier results fall between the results of the other two roles, reflecting both reduced resolution time and increased fix emergence tendencies, albeit with far weaker effects. The conservative takeaway from these results is that when it comes to priority levels, focus pays off for problem knowledge producers but is detrimental for solution knowledge producers, who may be better off with breadth of priority levels.

Measure two: Percent of problems whose severity and priority levels changed at least once acted upon in each role

The second measure of solution knowledge value is the percent of problems acted upon in each role whose severity and priority level changed at least once after submission of the problem knowledge to the meta-organisation. A tendency to act upon problems whose severity and priority levels were changed at least once is theorised to be correlated with better solution knowledge emergence outcomes due to the refined tuning of the severity and priority levels more accurately reflecting the value of the solution knowledge. As discussed in Chapter Four: Research Method, the severity change variable used in the present measure is the same as the one used to test hypothesis four. It is included to better separate the model fit characteristics attributable to severity vs. priority change tendencies at the individual level of analysis. Whereas the hypothesis four discussion section considers its effects relative to knowledge flow impediments, the present section considers its effects relative to solution knowledge value signalling in combination with the priority change effects.

Examination of the regression model summaries reveals that, for the problem knowledge producer role, the percent of bugs reported whose severity level changed at least once is only correlated with increased resolution time ($p < 0.05$). For the solution knowledge producer role, it is correlated with increased resolution time ($p < 0.05$) and increased fix emergence tendencies ($p < 0.01$). For the solution knowledge verifier role, it is associated with decreased patch emergence tendencies ($p < 0.01$).

A tendency to act on problems with at least one priority change, for the problem knowledge producer role, is correlated with increased resolution time ($p < 0.001$), and increased patch emergence tendencies. For both the solution knowledge producer and solution knowledge verifier roles, a tendency to act on problems with at least one priority change is correlated with reduced fix emergence tendencies ($p < 0.01$).

Taken collectively, these results provide a broader contextualization of the results observed at the problem level of analysis as well as in the hypothesis four models and suggest partial support for the hypothesis, with role-based contextualization. Priority changes delay solution knowledge emergence, but result in better patch emergence tendencies for problem knowledge producers, as observed at the problem level of analysis. However, priority changes are distracting for solution knowledge producers and solution knowledge verifiers, resulting in worse fix emergence tendencies. By contrast, while severity changes delay with no benefit for the problem knowledge producer and as observed at the problem level of analysis, for the solution knowledge producer role, the delay pays off in terms of increased fix emergence tendencies. Further, a portion of the severity change effect observed in the results of the hypothesis four models is likely more attributable to associated priority change effects that are not included in those models. The interaction between priority and severity changes results in a moderating of the effect severity changes on solution knowledge emergence outcomes, as can be seen by the slightly different fix and patch emergence tendencies.

The contributions to theory build upon the reports that developers are able to better measure the severity and priority levels than less involved users (c.f. Lewis, et al., 2013; Zhou, Neamtiu, & Gupta, 2015) by further suggesting that that changes in priority and severity benefit different participants in the meta-organisation. A novel contribution to theory is the interaction of severity and priority change effects, which has not previously been reported. Future research that isolates these factors and examines the interaction effects using interaction terms in the regression modelling, rather than isolation of variable in model fitting, may be fruitful in further refining our understanding of the impact of these two variables on solution knowledge emergence.

Building upon the problem level discussion, the individual level effects suggest that the contributions to practice are that the benefits of severity changes for developers must be weighed against the delay it introduces and the potential negative effect of that delay on other meta-organisation participants. Likewise, priority change decisions must carefully weigh the negative effect on developers against the benefits of shifting priorities for other meta-organisation participants. The interaction of both severity and priority changes are also likely to have an impact on practice, but the implications are as of yet unclear, subject to further research building upon the effects discovered and reported in the present study.

Measure three: Percent of problems with one or more top 3, 10, 25, or 50 keywords acted upon in each role

The third measure of solution knowledge value is the percent of problems with one or more top 3, top 10, top 25, or top 50 keywords acted upon in each role. It is hypothesized that a tendency to act more on problems with more frequently used keywords is correlated with better solution knowledge emergence due to the increased solution knowledge value that these top keywords designate.

Examination of the regression model summaries reveals that, for the problem knowledge producer role, the percent of bugs reported with a top 3 keyword is correlated with reduced resolution time ($p < 0.001$), and increased fix ($p < 0.001$) and patch ($p < 0.05$) emergence tendencies. Percent of bugs reported with a top 10 keyword is largely insignificant, with reduced resolution time missing sufficient certainty of significance at p ~= 0.12. Percent of bugs reported with a top 25 keyword is correlated with increased resolution time ($p < 0.001$) and increased reassigning tendencies ($p < 0.01$). Percent of bugs reported with a top 50 keyword has no significant correlation with outcomes of interest.

For the solution knowledge producer role, percent of problems acted upon with a top 3 keyword or a top 50 keyword are not significantly correlated with any outcomes of interest. Percentage of problems acted upon with a top 10 keyword is correlated with reduced resolution time ($p < 0.01$) and increased fix emergence tendencies ($p < 0.05$). Percentage of problems acted upon with a top 25 keyword is correlated with increased resolution time ($p < 0.001$) and reduced fix emergence tendencies misses the cut-off for sufficient certainty of significance of association at p ~= 0.09.

For the solution knowledge verifier role, percentage of problems acted upon with a top 3 keyword is correlated with worse patch emergence ($p < 0.05$) and percentage of problems acted upon with a top 10 keyword is correlated with better patch emergence ($p < 0.05$). However, given the much smaller sample size for these roles, these results should be interpreted with caution as the regression model evaluation statistics suggest that the apparent power may be spuriously inflated.

Taken collectively, these results match those observed at the problem level of analysis in that they are largely inconsistent and contradictory. They suggest that keyword popularity may not have inherent validity as a construct representing solution knowledge value, at least with respect to solution knowledge emergence outcomes of interest. Whereas a plausible u-shape relationship with solution knowledge emergence is apparent in the results for the problem knowledge producer role, examination of the mixed-effects models suggest that the observed correlations are more attributable to organisation level nestedness effects than individual level effects. Contextually, with the mixed-effects regression model results, the conservative interpretation is that there are limited, if any, individual level effects of tendencies to act on problems with more popular keywords, across roles, on solution knowledge emergence, lending no support to the hypothesis.

Measure four: Mean overall number of follows, votes, comments, and flags on problems acted upon in each role

The fourth measure of solution knowledge value is the mean overall number of follows, votes, comments, and flags on problems acted upon in each role. It is hypothesized that a tendency to act on problems with more follows, votes, comments, and flags is correlated with better solution knowledge emergence due to the increased value of the solution knowledge to the stakeholders doing the following, voting, commenting, and flagging.

Examination of the regression model output summaries reveals that, for all three roles in which individuals engage, the mean number of follows on problems acted upon is correlated with reduced resolution time ($p < 0.001$), reduced reopening and reassigning tendencies ($p < 0.001$), increased fix emergence ($p < 0.05$) and decreased patch emergence ($p < 0.01$) tendencies.

These follow-related results partially match those observed at the problem level of analysis but paint a slightly different picture of individual level effects. Whereas a greater number of follows on problems is correlated with increased resolution time at the problem level, a greater number of mean follows is correlated with reduced resolution time at the individual level of analysis, across all roles in which individuals engage. Likewise, reopening and reassigning tendencies are increased by follows at the problem level and are decreased by follows at the individual level. The result of increased fix emergence tendencies is consistent with the problem level, but the patch emergence tendencies are worse at the individual level, whereas they were better at the problem level of analysis.

For the mean number of votes on problems acted upon in the problem knowledge producer role, there is significant correlation with increased resolution time ($p < 0.001$) and decreased fix and patch emergence tendencies ($p < 0.001$). The increased resolution time results hold for the solution knowledge producer role ($p < 0.001$), although the decreased fix and patch emergence tendencies fail to achieve sufficient certainty of significance. The solution knowledge verifier role results also reflect the same pattern of increased resolution time ($p < 0.001$) and decreased fix ($p < 0.001$) and patch ($p < 0.01$) emergence tendencies.

These vote related results partially match the results observed at the problem level of analysis and largely contradict the hypothesis. They suggest an individual level negative effect on fix and patch emergence above and beyond the same effect observed at the problem level of analysis. Yet, at the individual level, resolution time appears increased rather than decreased as it is at the problem level. This difference suggests that. at the problem level. votes lead to problems being rejected faster, whereas at the individual level, votes cause problems to spend more time in the knowledge production cycle before being rejected. This latter effect suggests that individuals do pay attention to votes, but that votes might represent a significant disagreement in the meta-organisation such that they redirect effort fruitlessly.

For the mean number of comments on problems acted upon in the problem knowledge producer role, there is significant correlation with increased resolution time ($p < 0.001$), increased reopening and reassigning tendencies ($p < 0.001$), and increased fix and patch emergence tendencies ($p < 0.001$). These results largely hold for the solution knowledge producer and solution knowledge verifier roles, with only the increased fix emergence tendencies failing to reach sufficient certainty of significance for the latter role due to the lower sample size in the role.

These comment related results largely match those observed at the problem level of analysis (for hypotheses two and five), and in the comparative model at the individual level of analysis in hypothesis two, lending support to the validity of both the operationalization choices and regression model fit methods, as discussed in Chapter Four: Research Method. The results largely support the hypothesis by suggesting that a tendency to have more comments on problems acted upon is beneficial for individual level solution knowledge emergence across roles. Whereas comments delay the emergence of solution knowledge and loop problem knowledge through the knowledge production cycle with increased reopening and reassigning tendencies, these delays pay off in terms of increased fix and patch emergence tendencies for all roles. Whereas votes delay without payoff in terms of fix emergence tendencies, comments signal solution knowledge value by providing additional knowledge for use in the creation of the solution knowledge. It may be that votes are “too easy”, requiring a single click, whereas comments require thought and effort to provide additional knowledge. The former results in “whims” being counted, whereas the latter’s necessary effort better signals solution knowledge that is truly valued.

For the mean number of flags on problems acted upon in the problem knowledge producer role, there is significant correlation with reduced resolution time ($p < 0.001$), decreased reopening and reassigning tendencies ($p < 0.001$), and increased fix and patch emergence tendencies ($p < 0.001$). The solution knowledge producer role results also show decreased reopening and reassigning tendencies ($p < 0.001$) as well as increased fix and patch emergence tendencies ($p < 0.001$), although they also show a correlation with increased resolution time ($p < 0.05$). The solution knowledge verifier role results also show decreased reopening and reassigning tendencies ($p < 0.001$) as well as increased fix and patch emergence tendencies ($p < 0.001$), with no significant correlation observed with resolution time.

These flag related individual level results largely match those observed at the problem level of analysis (for hypothesis four) and largely support the hypothesis by suggesting that flags are positively correlated with solution knowledge emergence at the individual level across roles. Whereas comments provide additional knowledge to assist in the creation of the solution knowledge at the cost of delays, flags appear to provide additional knowledge in a concise form, reducing delays, yet still promoting solution knowledge emergence.

Flags appear to represent an optimal middle ground between votes and comments, where the former are so simple they detract, and the latter are sufficiently complex that they are useful but at the cost of delays. Flags are the "just right" amount of knowledge, requiring a selection of an appropriate term—one that might depend on tacit subject matter expertise in the meta-organisation for interpretation—with more effort than a simple click, but less effort than a comment. The correlations observed in the results for individual level comment size effects in hypothesis two support this interpretation, showing that, the larger comments get, the worse the effect on solution knowledge emergence overall. There is an optimum amount of knowledge that can be applied to the knowledge production process that individuals can absorb; too little or too much new knowledge detracts from solution knowledge emergence. Future research may wish to examine this quadratic relationship in more detail. Non-linear modeling using a data set suitable for such analyses may be particularly fruitful in further elucidating the optimal size of additions of knowledge during the knowledge production process for optimal levels of solution knowledge emergence.

Effects of individual nestedness in organisations

Examination of the mixed-effects regression model summaries reveals that there are significant organisational effects that affect the individual level results, albeit only marginally for most outcomes and almost not at all for the solution knowledge verifier role results. For the problem knowledge producer role, the AIC and BIC delta statistics reveal that the mixed-effects models are marginally superior in all cases, with isolated organisation nestedness effect sizes ranging from “small” to “medium-to-large”. For the solution knowledge producer role, the mixed-effects models reveal isolated organisation nestedness effect sizes ranging from “small” to “large”. For the solution knowledge verifier role, the isolated effect size for reassignment tendencies is “very large” and for fix emergence tendencies is “large”, with the remaining mixed-effects models not superior to the OLS models.

Despite the significant organisational nestedness effects, most of the individual level results remain significant after isolating the organisational random effects, with one notable exception. The individual level results suggest a u-shape relationship between tendencies to focus on problems with certain severity levels and fix emergence tendencies. The mixed-effects results suggest that, in particular for the low and high endpoint severity levels, “trivial” and “blocker” respectively, the observed effects may not be as strong as reported. Comparison with the organisation level of analysis in the next section facilitates contextual interpretation of the results.

Summary of dependent variable effect results at individual level

Comparison of the control and full regression models for the effect of solution knowledge value measures on the dependent outcomes of interest reveals an overall picture of effect size at the individual level of analysis. The model F & Chi-squared statistics as well as the comparative AIC delta statistics suggest that in all cases the solution knowledge value independent variable full models are significantly superior to the control variable only models.

For the problem knowledge producer role, for resolution time the Cohen’s additive f^2 effect size is “large”; for reopening and reassigning tendencies it is “small-to-medium” and “large” respectively; and, for fix and patch emergence tendencies it is “medium-to-large” and “very large” respectively. For the solution knowledge producer role, for resolution time, the additive effect size is “large”; for reopening and reassigning tendencies it is “medium” and “medium-to-large” respectively; and, for fix and patch emergence tendencies it is “medium-to-large” and “very large” respectively. For the solution knowledge verifier role, for resolution time, the effect size is “large”; reopening and reassigning have effect sizes of “large” and “very large” respectively; and, fix and patch emergence tendency effect sizes are “large” and “very large” respectively.

In summary, the effects of individual level solution knowledge value independent variables are best described as a balance between the disruptions associated with new knowledge being added to the knowledge production process and the value of that new knowledge in more accurately signaling solution knowledge value to prioritize solution knowledge emergence. Severity levels generally signal solution knowledge value, but the specific severity levels individuals focus on may have non-linear relationships with solution knowledge emergence. Severity changes introduce delays in resolution but improve fix emergence tendencies for developers. Priority levels generally signal solution knowledge value and focus on specific priority levels is good for problem knowledge producers, bad for solution knowledge producers, and in between for solution knowledge verifiers. Priority changes also introduce delays, with improved patch emergence for problem knowledge producer but worse fix emergence for solution knowledge producers and solution knowledge verifiers.

As seen at the problem level of analysis, the top keyword related results are inconsistent and contradictory, suggesting spuriousness in the validity of the underlying construct as a signaling of solution knowledge value. The conservative interpretation is a general lack of significance at the individual level of analysis.

The number of follows on problems upon which individuals act is correlated with reduced resolution time and increased fix emergence tendencies, but also reduced patch emergence tendencies, across roles, suggesting a solution knowledge type specific effect for the number of follows at the individual level of analysis; votes are generally bad for solution knowledge emergence across roles, contrary to the hypothesis; comments are generally good for fix and patch emergence, at the cost of increased resolution time; and, flags appear to be the perfect balance of new knowledge vs. required effort, with correlation with decreased resolution time and increased fix and patch emergence tendencies at the individual level.

Organisation level of analysis results

At the organisation level of analysis there are seven measures that represent the conceptualization of the dependent outcome of interest, solution knowledge emergence, as depicted in Figure 26 and as operationalized into the variables described in Table 105. Five measures represent the conceptualization of the antecedent of interest in hypothesis six, solution knowledge value, as depicted in Figure 32, and as operationalized into the variables described in Table 112. The results of the analyses are depicted in the regression model output summaries in Appendix D: Regression models.

As discussed in Chapter Four: Research Method, regression models are separated according to the aggregate roles in which members of organisations engage when participating in the knowledge production process: aggregate problem knowledge producer (reporter) role, aggregate solution knowledge producer (assigned_to) role, and aggregate solution knowledge verifier (QA_contact) role.

Measure one: Percent of problems of each severity level acted upon in each aggregate role

The first measure of solution knowledge value at the organisation level is the percent of problems of each severity level acted upon in each aggregate role. A tendency of organisation members to act upon problems with higher severity levels is theorized to be correlated with better solution knowledge emergence outcomes due to the higher value of the solution knowledge signaled by the higher severity level.

Examination of the regression model summaries reveals that, for the aggregate problem knowledge producer role, the percent of bugs reported of most severity levels are not significantly correlated with solution knowledge emergence. A notable exception is the tendency to act on problems with the “critical” severity level, which is correlated with worse fix emergence tendencies ($p < 0.05$). A tendency to act on problems with the “enhancement” and “minor” severity levels is also correlated with increased resolution time ($p < 0.001$).

The aggregate solution knowledge producer role effects are more prominent. A tendency to act on “enhancement” ($p < 0.05$) or “major” ($p < 0.01$) severity level problems is correlated with worse patch emergence tendencies ($p < 0.05$) and, for “critical” severity level problems, worse fix emergence tendencies ($p < 0.01$). By contrast, a tendency to act on “minor” severity problems is correlated with better patch emergence tendencies ($p < 0.01$). The “trivial” severity level is also correlated with increased resolution time ($p < 0.001$). The low sample size for the aggregate solution knowledge verifier role leads to largely spurious models, making the observed coefficients in the results unreliable.

Taken collectively, these results suggest a moderate incremental contribution of organisation level effects of focus on problems of specific severity levels for the aggregate solution knowledge producer role. The results suggest that a tendency to focus more on just low severity or just high severity problems is correlated with worse solution knowledge emergence, whereas a tendency to focus on a median severity level of problem is correlated with better solution knowledge emergence. Yet, the observed effects for the developer role are derived from a relatively small sample size (n = 205) at the organisation level and should therefore be interpreted with caution. The results for the aggregate reporter role, interpreted together with the mixed-effects model results, suggest that the observed effects at the individual level for fix emergence tendencies for very low ("trivial") and very high ("blocker") severity levels may be spurious as they disappear in the mixed-effects model but do not appear at the organisation level of analysis. Therefore, the individual level conclusions should also be conservative with respect to fix emergence tendency effects.

Measure two: Percent of problems whose severity and priority levels changed at least once acted upon by organisation members in each aggregate role

The second measure of solution knowledge value is the percent of problems acted upon by organisation members in each aggregate role whose severity and priority levels changed at least once after submission of the problem knowledge to the meta-organisation. A tendency to act on problems whose severity and priority levels were changed at least once is theorised to be correlated with better solution knowledge emergence outcomes due to the refined tuning of the severity and priority levels more accurately reflecting the value of the solution knowledge.

Examination of the regression model summaries reveals that, for the aggregate problem knowledge producer role, the percent of problems reported whose severity level changed at least once is correlated with reduced patch emergence tendencies ($p < 0.05$). For the aggregate solution knowledge producer role, it is correlated with increased resolution time ($p < 0.05$) and increased fix emergence tendencies ($p < 0.001$).

These results partially match those observed at the individual level of analysis and lend support to the role-specific differential effects of severity changes on solution knowledge emergence. A tendency to act on problems with more severity changes has negative solution knowledge implications for the aggregate problem knowledge producer but positive solution knowledge emergence implications for the aggregate solution knowledge producer. Further, there are organisation-level specific effects, suggesting that organisational actors are broadly affected by these changes above and beyond the individual level effects.

In terms of the organisation level effects of tendencies to act on problems with priority changes, for the aggregate problem knowledge producer role, is correlated with increased resolution time ($p < 0.001$), and increased fix ($p < 0.001$) and patch ($p < 0.05$) emergence tendencies. For the solution knowledge producer role, it is correlated with reduced fix emergence tendencies ($p < 0.05$).

These results also match those observed at the individual level of analysis, suggesting that priority level changes have the opposite effect of severity level changes, promoting solution knowledge emergence for aggregate problem knowledge producers and reducing solution knowledge emergence for aggregate solution knowledge producers. Further, there are clear organisation level effects above and beyond the individual level effects.

These results expand on the reports in the literature that solution knowledge producers are able to better determine the severity and priority levels of problems (c.f. Lewis, et al., 2013; Zhou, Neamtiu, & Gupta, 2015) by demonstrating differential benefits amongst organisational actors. The contributions to practice are that the benefits to some organisational actors of changes in priority and severity levels should be weighed against the hindrances to other organisational actors.

Measure three: Percent of problems with one or more top 3, 10, 25, or 50 keywords acted upon by organisation members in each aggregate role

The third measure of solution knowledge value is the percent of problems acted upon by organisation members in each aggregate role with one or more top 3, top 10, top 25, or top 50 keywords. It is hypothesized that a tendency to act more on problems with more frequently used keywords is correlated with better solution knowledge emergence due to the increased solution knowledge value that these top keywords signal.

Examination of the regression model summaries reveals that the percentages of organisation members engaging in the aggregate problem knowledge producer role acting on problems with "top 3" keywords is positively correlated with fix emergence tendencies ($p < 0.05$). With the exception of this effect, no other effects are significant for other rankings of keyword popularity or aggregate roles.

While this result matches the result observed for the problem knowledge producer role at the individual level of analysis, the general lack of other effects adds support to the notion that the popularity of keywords has weak construct validity as a measure of solution knowledge value, at least as compared to the other factors examined. The conservative interpretation is a general lack of support for the keyword popularity related sub-hypotheses in the results.

Measure four: Number of follows, votes, and comments on problems acted upon by organisation members in each aggregate role

The fourth measure of solution knowledge value is the number of follows, votes, and comments on problems acted upon by organisation members in each aggregate role. It is hypothesized that a tendency to act on problems with more follows, votes, and comments is correlated with better solution knowledge emergence due to the increased solution knowledge value signaled by those actions by stakeholders in the meta-organisation.

Examination of the regression model output summaries suggest that the number of follows on problems acted upon in the aggregate problem knowledge producer role is correlated with increased resolution time ($p < 0.01$), and decreased reopening and reassigning tendencies ($p < 0.001$). There are no significant effects for the aggregate solution knowledge producer or aggregate solution knowledge verifier roles. These results partially match those observed at the individual level of analysis but suggest a distinct organisation level effect. The decreased reopening and reassigning tendencies are similar, with incremental contributions at the organisation level. Yet, the resolution time effects are opposite. Examination of the mixed-effects model summaries suggests that both the individual and the organisation level effects on resolution time are significant, but pulling in opposing directions, validating the importance of separating the effects of follows both by role and level of analysis to understand the full effect.

For the number of votes on problems acted upon in the aggregate problem knowledge producer role, there is a significant correlation with increased resolution time ($p < 0.001$), increased reassigning tendencies ($p < 0.001$), and decreased fix emergence tendencies ($p < 0.001$). For the aggregate solution knowledge producer role, a correlation with increased resolution time ($p < 0.001$) is also observed and the correlation with decreased fix emergence tendencies narrowly misses sufficient certainty of significance at $p \sim= 0.052$. These results largely match those observed at the individual level of analysis, suggesting that votes are broadly correlated with worse solution knowledge emergence, contrary to the hypothesis and contrary to extant theory and practice assumptions. This result further cements the implication that conventional perspectives on the value of votes may be incorrect, to the detriment of solution knowledge emergence at all levels of analysis and across (aggregate) roles.

For the number of comments on problems acted upon in the aggregate problem knowledge producer role, there is a significant correlation with increased resolution time ($p < 0.001$), increased reopening and reassigning tendencies ($p < 0.001$), and increased fix and patch emergence tendencies ($p < 0.001$). The increased reopening ($p < 0.01$) and reassigning ($p < 0.001$) tendencies are also observed for the aggregate solution knowledge producer role. These results largely match those observed at the individual level of analysis and lend support to the hypothesis that comments are positively correlated with solution knowledge emergence. Interpreted in combination with the mixed-effects regression model summaries, these results suggest an incremental effect at the organisation level of analysis above and beyond the individual level effects.

Measure five: Mean number of flags on problems acted upon by organisation members in each aggregate role

The fifth measure of solution knowledge value is the mean number of flags on problems acted upon by organisation members in each aggregate role. It is hypothesized that a tendency to act on problems with more flags is correlated with better solution knowledge emergence due to the increased solution knowledge value signaled by the flags.

Examination of the regression model output summaries suggest that the mean number of flags on problems acted upon in the aggregate problem knowledge producer role is correlated with increased fix and patch emergence tendencies ($p < 0.001$). These increased fix and patch emergence tendencies hold for both the solution knowledge producer and solution knowledge verifier role (although the models in the latter role are of questionable significance). Examination of the mixed-effects regression model results suggests that the organisation level effects are above and beyond the individual level effects, albeit weakly in the case of patch emergence tendencies, which reside primarily at the individual level.

These results match those observed at the problem and individual levels of analysis suggesting that the signaling of solution knowledge value through the number of flags on problems is generally correlated with better solution knowledge emergence and reduced time for that solution knowledge to emerge. The contributions to practice are that flags should possibly be favoured over other signaling mechanisms, such as votes, to promote the emergence of solution knowledge in meta-organisations.

Summary of dependent variable effect results at organisation level

Comparison of the control and full regression models for the effect of solution knowledge value measures on the dependent outcomes of interest reveals an overall picture of effect size at the organisation level of analysis. For the aggregate problem knowledge producer role, the comparative Chi-squared statistic and comparative AIC delta statistic reveal that the models that include the solution knowledge variables are all superior to the control-only models, albeit only marginally so for the reopening and reassigning tendency outcomes. The additive effect sizes on the outcomes of interest of the independent variables above and beyond the control variables are "medium" for resolution time, "small-to-medium" for reopening tendencies, "medium" for reassigning tendencies, and "large" and "very large" for fix and patch emergence tendencies respectively.

For the aggregate solution knowledge producer role, the comparative Chi-squared statistic and comparative AIC delta statistic reveal that the models that include the solution knowledge value variables are superior for the resolution time and fix and patch emergence tendencies outcomes. The reopening and reassigning tendency models are not superior to the control models. The additive effect sizes on the outcomes of interest of the independent variables above and beyond the control variables are "very large" for resolution time and "very large" for fix and patch emergence tendencies respectively. As is the case in previous results, the small sample size for the aggregate solution knowledge verifier role results in insignificant models across all outcome variables.

In summary, the organisation level effects of the solution knowledge value independent variables largely match those observed at the individual level of analysis, with a few notable exceptions. For the effects of focusing on problems of particular severity levels, the results observed for the aggregate reporter role generally match those observed at the problem level of analysis, suggesting that focus is worse than breadth of effort, but the organisation level effects are generally weak, with the majority of the effects residing at the individual level of analysis. The aggregate developer role effects are similar, matching the individual level u-shaped effects, with focus on low or high severity level problems being negative and focus on median severity level problems being positive, with weak additive contributions to the overall effect at the organisation level.

The effects of tendencies to act on problems with severity and priority changes after problem knowledge submission to the meta organisation are similar to those observed at the individual level of analysis, with significant incremental organisation level specific effects; priority changes are generally positive for aggregate problem knowledge producers and negative for aggregate solution knowledge producers, while the opposite is true for severity changes.

A tendency to act more frequently on problems with keywords of differing levels of popularity is generally insignificant at the organisation level of analysis, similar to as observed at problem and individual levels of analysis. The keyword popularity construct appears to have insufficient validity as a measure of solution knowledge value. While there is a plausible "top 3 keyword" effect, given that it is very small relative to other observed effects and given the construct validity challenges observed at the problem and individual levels of analysis, if the effect is not spurious, it still falls below the large-effect target levels of the present study and is therefore treated as insignificant.

The number of follows on problems acted upon by organisation members are correlated with increased resolution time for the aggregate reporter role, the opposite direction of effect as that observed at the individual level of analysis. The correlation between follows and decreased reopening and reassigning tendencies is consistent with the individual level, as is the lack of effect for other (aggregate) roles. The effect of votes on solution knowledge emergence is generally negative across roles, as observed at the individual and problem levels of analysis, with an incremental effect specific to the organisation level of analysis. This result is of particular relevance given it is opposite to the intentions of this deliberate solution knowledge value signalling mechanism used in meta-organisations. The effect of comments is similar to that observed at the individual level of analysis: longer resolution times but increased solution knowledge emergence tendencies as a result. The organisation level effects are incremental above and beyond the individual level effects. And, flags are broadly correlated with better and faster solution knowledge emergence tendencies, across roles, with incremental organisation level effects above and beyond the individual level effects.

Summary of results of hypothesis testing

The testing of the hypotheses resulted in the identification of the antecedent factors affecting successful solution knowledge emergence, answering the research question. Further, the triangulation of variables and levels of analysis resulted in a clear picture of the nature and degrees of the effects of these factors with both depth and breadth not previously considered in studies of meta-organisations. Some of the results lend support to extant theories in the literature. Other results contradict our current understanding of factors affecting knowledge production processes. Yet other results contribute novel evidence for factors not previously empirically examined. As with most studies, the results pose new questions, setting the stage for a future research agenda to further improve our understanding.

A summary of the high level antecedent factors affecting solution knowledge emergence types is presented in Table 117, offering a concise and digestible answer to the research question. The results of the testing of each hypothesis, summarizing the detailed antecedent factors affecting the measures of successful solution knowledge emergence, are presented in Appendix E: Summary of results. In those tables, "(+)" in front of a listed factor denotes a significant positive correlation between the independent variable factor and the outcome of interest (column header) and "(-)" denotes a significant negative correlation between the variables. In the case of categorical independent variables, the chosen symbol represents the aggregate tendencies of the categories relative to the formulation of the hypothesis.

Measure of solution knowledge emergence	Contingency factors driving successful solution knowledge emergence across all levels					
	Absorptive capacity	**Codifiability**	**Dominant knowledge paradigm**	**Knowledge flow impediments**	**Knowledge stakeholder influence**	**Solution knowledge value**
Resolution time	Activities, flags, keywords, operating system, platform, product, role engagement, severity	Attachments, comments, description, n-gram profile, readability, redundancy, title	Classification, operating system, platform	Activities, blocking, flags, keywords, life cycle violation, milestone, reassigning, reopening, severity, whiteboard	Actor involvement, comments, follows, profile domain, votes, watching	Blocking, comments, flags, follows, priority, severity, votes
Assignment time	NIL	Attachments, comments, description, n-gram profile, redundancy, title	NIL	Blocking, flags, keywords, life cycle violation, milestone, reassigning, timing, whiteboard	Actor involvement, comments, follows, profile domain, votes	NIL
Development time	Closed bugs, open bugs, timing	NIL	NIL	Blocking, flags, keywords, life cycle violation, milestone, reassigning, reopening, whiteboard	NIL	NIL
Reopening tendencies	Role engagement	Attachments, comments, redundancy	NIL	Blocking, flags, keywords, life cycle violation, milestone, reassigning, whiteboard	Activities, actor involvement, comments, follows, votes	Comments, flags, follows, priority
Reassigning tendencies	Open bugs, product, role engagement	Attachments, redundancy, comments	Platform	Life cycle violation, milestone, reopening, whiteboard	Actor involvement, comments, follows	Comments, flags, follows, priority, votes
Confirmation tendencies	Classification, component, operating system, open bugs, platform, product, timing	Title, description, attachments, n-gram profile, redundancy, comments	Classification, components, operating system, product, platform	Blocking, flags, keywords, life cycle violation, milestone, reassigning, reopening, whiteboard	Actor involvement, comments, profile domain, votes	Blocking, follows, priority, severity
Fix emergence tendencies	Activities, classification, component, flags, open bugs, operating system, platform, product, reopening, resolution time, role engagement, severity, whiteboard	Description, attachments, n-gram profile, redundancy, comments, readability	Classification, operating system, platform	Activities, blocking, flags, life cycle violation, milestone, reassigning, reopening, severity, whiteboard	Activities, actor involvement, comments, follows, votes, watching	Comments, flags, follows, priority, severity, votes
Patch emergence tendencies	Activities, classification, component, flags, keywords, open bugs, operating system, platform, product, resolution time, role engagement, severity, timing	Attachments, comments, description, n-gram profile, redundancy, title	Classification, component, operating system, product, platform	Activities, blocking, flags, keywords, life cycle violation, milestone, reassigning, reopening, severity, whiteboard	Activities, actor involvement, comments, follows, votes, watching	Comments, flags, follows, priority, severity, votes

Table 117: Summary of antecedent factors driving successful solution knowledge emergence

CHAPTER SEVEN: CONTRIBUTIONS, LIMITATIONS, & CONCLUSION

This study provides the first comprehensive assessment of factors affecting solution knowledge emergence in open source meta-organisations. The results refine recent research, which suggests that firms may be able to create a competitive advantage by deliberately revealing specific problem knowledge beyond firm boundaries to open source meta-organisations such that solution knowledge is created that benefits the focal firm more than its competitors (Alexy, George, & Salter, 2013), by providing a novel explanation for the heterogeneity in the success of the use of these knowledge-revealing strategies. The identification and measurement of the 35 antecedent factors that affect solution knowledge emergence in different ways across three levels of analysis makes numerous contributions to research and practice, which are discussed in turn.

Contributions to research

This study makes numerous contributions to strategic management research, building on the extant conversations in the KBV, open source, and organisational forms of production literatures, discussed in turn.

Contributions to open source literature

This dissertation makes at least seven contributions to the open source literature. First, it expands beyond the usual technical lens through which open source meta-organisations have typically been considered by applying a strategic management theory based lens that evaluates organisation level participation in open source meta-organisations and KBV-based success factors. The result is an expanded understanding of the nuances of open source strategies other than just bundling, branding, and contractual guarantees packaged with free loss leaders.

Second, by measuring dozens of factors never before assessed and by measuring these and previously measured factors with multiple triangulated operationalizations, it contributes a broader understanding of the specific factors that affect open source strategies and a deeper understanding of the degree to which variations in representations of factors change outcomes of interest. This increased breadth and depth provides an explanation for some of the disparity in the open source literature and cements the necessity of considering a comprehensive range of factors as well as multiple levels of analysis in order to fully understand the phenomenon.

Third, it refines our understanding of the interaction between the nature of knowledge and the consumers of knowledge, highlighting the importance of considering both measures in tandem given that either measure alone is insufficient to predict outcome effects. As a result, the extant technical literature readability measures do not properly account for subject matter expertise of consumers of the written knowledge, revealing the need for modern, more targeted measures; the length and count of artefacts used in open source meta-organisations, such as descriptions and comments, are poor proxies for the amount of knowledge used in the knowledge production process without co-consideration of both the producer and the consumer of the knowledge; and, much of the disparity in the open source literature on the effects of problem level factors can potentially be explained by individual level and organisation level factors have rarely been considered in extant research.

Fourth, it contributes to our understanding of the impact of actors in open source meta-organisations. The results demonstrate heterogeneity of organisations participating in open source meta-organisations across a range of measures, including number of actors, degrees of involvement, activities, role engagement, rates of success for solution knowledge emergence, types of knowledge consumed and produced in the meta-organisation, degrees of influence, and potential experience effects. These differences between organisations provide a novel explanation for why some organisations are viewed as valuable contributors and other organisations are viewed as “leeches” by open source communities (Fitzgerald, 2006; Shah, 2006; MacAulay, 2010b; Söderberg, 2015).

Fifth, it refines our understanding of the actual actions of individuals in open source meta-organisations, highlighting cases where the actual actions differ from the stated actions. The results demonstrate that the individual self-reports of the type of attachments used by developers in the Mozilla meta-organisation, reported in the survey-based study of Bettenburg, et al. (2008), differ from the actual actions of developers. Further, the stated priorities of developers, as represented by priority flags, milestones, and votes often differ from the actual priorities of developers as measured by the nature and timing of solution knowledge production.

Sixth, it highlights the power and influence differences amongst actors in open source meta-organisations, demonstrating that influence on the knowledge production process can be both positive (promoting) and negative (hindering) depending on the factor that is being influenced and the actor doing the influencing. It provides novel evidence that power and influence may be separately exerted over both knowledge production factors and individuals and organisations engaging in the knowledge production process. In particular, it the results demonstrate powerful individual level knowledge type bias effects that are disproportionately large compared to the power and influence of organisational actors. The result is a refinement of the extant understanding of degree of involvement as a moderator of power and influence in open source meta-organisations (Dahlander & O'Mahony, 2011; Dahlander & Frederiksen, 2012).

Seventh, it contributes to our understanding of the factors affecting the efficiency and effectiveness of knowledge production processes used in open source meta-organisations. In particular, the results demonstrate that bug life cycle processes such as reopening and reassigning generally work as intended but certain individuals and organisations may subvert these processes resulting in suboptimal outcomes. Likewise, the results demonstrate that the severity and priority level setting and changing processes are generally effective but can be disrupted by some actors and improved by others. The result is a refined understanding of the nuances of factors that affect the bug life cycle process, expanding the work of Zimmermann, Nagappan, and Guo (2012) and Shihab, et al., (2010, 2013).

These contributions to the open source literature build on existing conversations to add new levels that enable an ongoing open source strategy research program.

Contributions to KBV literature

This dissertation makes at least seven contributions to the KBV literature. First, it links the KBV and open source literatures by matching and empirically testing similar factors with carefully constructed operationalisations that inform both literatures. The result is a bridging of both literatures' agendas that enables future joint research programs by developing methods for analysing databases that have previously only been considered in the computer science open source literature through a KBV management theory lens.

Second, it develops a novel method, novel measurements, and novel operationalisations of knowledge resources and knowledge production processes and empirically tests them in a novel context. These novel approaches both inform existing KBV theories with new empirical results and set the stage for novel theory development that extends extant theories. In particular, it considers “meta-organisations” as novel knowledge sharing contexts and examines the impact of the novel knowledge sharing production processes used in these contexts, broadening the scope of types of knowledge constructs empirically examined in the KBV research. It also assists in the efforts to standardize KBV constructs with clear operationalisations, which has been reported as challenging in extant research (Lane, Koka, & Pathak, 2006; Todorova & Durisin, 2007). One particularly useful novel construct is the artefacts of knowledge value signalling used in open source meta-organisations, given that measurements of knowledge value are of prime interest in the KBV empirical research agenda (c.f. Davis & Botkin, 1994; Kanevsky & Housel, 1998; Swan, Newell, Scarbrough, & Hislop, 1999; Das, 2003; Tomas & Hult, 2003; Martin, 2004; MacAulay, 2010b; Majchrzak, More, & Faraj, 2011; Xy & Bernard, 2011; Alexy, George, & Salter, 2013; Massingham, 2016).

Third, it identifies as measures the success factors that affect the solution knowledge emergence outcomes of deliberate knowledge spillover strategies, extending the research of Alexy, George, and Salter (2013) to explain why some organisations are more successful than others when using deliberate knowledge spillover strategies. The results provide a novel explanation for why some organisations are more successful than others at drawing knowledge from “spillover knowledge pools” that accumulate in open source meta-organisations from which they can draw solution knowledge that is of benefit to them, building on the research of Yang, Phelps, and Steensma (2010). The results also extend the “knowledge-based boundaries of the firm" and empirically measure both firm-specific and environment-specific factors within these expanded boundaries, answering the call for such research by Bogers, Afuah, and Bastian (2010).

Fourth, it improves our understanding of the locus of effects of factors that influence knowledge resources and knowledge production processes, building on an extensive stream of research that has attempted to pinpoint the specific order of knowledge resources effects in firms and reported significant difficulty in localisation efforts (Dierickx & Cool, 1989; Pisano, 1994; von Hippel, 1994; Priem & Butler, 2001; Arend, 2006; Volberda, Foss, & Lyles, 2010; Bingham & Eisenhardt, 2011). In so doing, it also identifies dependencies between types and sets of knowledge that affect knowledge production efforts and knowledge resource creation not previously reported in the KBV literature. In particular, it demonstrates the effects of paradigm shifts in knowledge representation on knowledge resources and knowledge production processes, effectively testing knowledge paradigm shift theoretical perspectives (Prahalad & Bettis, 1986; Nonaka, Umemoto, & Senoo, 1996).

Fifth, it refines our understanding of absorptive capacity localisation, lending support to theories of the existence of separate organisation and individual level absorptive capacities, lending support to theoretical arguments to that effect (Cohen & Levinthal, 1990; Lane & Lubatkin, 1998). It also provides a novel explanation for why the potential absorptive capacity of an individual or organisations may not be wholly fulfilled in terms of realized absorptive capacity in specific knowledge consumption and application circumstances (Zahra & George, 2002) by identifying "absorptive capacity impairment" factors that act as moderators that reduce the ability of individuals and organisations to consume and apply knowledge in the knowledge production process. The results suggest that absorptive capacity impairment factors affect individuals and organisations to different degrees. They also affect individual and organisational actors to different degrees based on their level of involvement and tacit subject matter expertise. The latter results, specifically, provide novel empirical evidence for the existence of tacit knowledge in both individuals and organisations orthogonally, lending support for theoretical arguments to that effect (c.f. Nonaka, 1994; Grant, 1996a, 1996b; Cook & Brown, 1999; Ambrosini & Bowman, 2001; Nonaka & von Krogh, 2009).

Sixth, it improves our understanding of the differential degrees of influence that individuals and organisations have on knowledge production processes, suggesting that knowledge flow is primarily affected by individuals. The results provide a novel explanation for the disparate results in the KBV literature with respect to knowledge production process influence factors by demonstrating that it is necessary to model such factors with at least three dimensions: 1) the type of action taken in the knowledge production process, 2) the formal role of the actor in the knowledge production process, and, 3) the degree of involvement or experience of the actor in the knowledge production process. The results also lend support to theories that iterative knowledge production processes may be superior to inflexible linear processes (c.f. Pisano, 1994; Basili & Caldiera, 1995; Pérez-Bustamante, 1999; Bhatt, 2000). The empirical evidence also reveals that inflexible knowledge production processes may be suboptimal for some classes of actors such that differential processes for different classes of actors may be useful, an issue not previously considered in the KBV literature.

Seventh, it informs the knowledge comparison research stream by demonstrating that similar knowledge can be handled very differently in meta-organisations based on a broad range of factors unrelated to the knowledge itself. In particular, duplicate problem knowledge is shown to be valuable in the knowledge production process when both duplicate and duplicated knowledge sets are assessed together, providing a novel explanation for the disparate evidence in the literature on the value of duplicate knowledge sets. The results also demonstrate that comparative classification of similar knowledge must be relative to two or more classification categories as single absolute distance measures have unbounded ceilings that obfuscates the comparative measurement scale beyond usefulness. These results provide a novel explanation for some of the challenges in comparing and classifying knowledge using n-gram profile distance measures reported in the literature and offer and empirically validate a solution to the issue.

These contributions to the KBV literature build on existing conversations to add new levels that enable an ongoing KBV research program.

Contributions to organisational forms of production literature

This dissertation makes at least four contributions to the organisational forms of production literature. First, it links the open source, KBV, and organisation forms of production literatures by matching and empirically testing similar theoretical factors with carefully constructed operationalisations that join the conversations in each literature stream. In so doing, it provides evidence of firm agency over factors of production in the open source meta-organisational form that further rebukes old assumptions in the classical industrial economics literature, which suggests that management is largely irrelevant because firms are at the mercy of the form of production itself. In bridging open source and KBV perspectives with the organisational forms of production literature, this dissertation breathes fresh life into the debates that establish management as field that is distinct from economics.

Second, it answers the call for research on non-traditional organisational forms by Foss, Husted, and Michailova (2010) and Gulati, Puranam, and Tushman (2012) by examining traditional and novel factors thought to affect success in classic forms of production in the novel open source meta-organisation form of production that has low stratification and low barriers to entry (Gulati, Puranam, & Tushman, 2012). It lays the groundwork for examination of other non-traditional organisational forms such as meta-organisations with different degrees of stratification, barriers to entry, creation conditions, intellectual property ownership and use conditions, governance structures, degrees of task decomposability, distribution of problem solving, locus of value creation, and appropriability of created value (Chesbrough & Appleyard, 2007; O'Mahony & Bechky, 2008; Lakhani, Lifshitz-Assaf, & Tushman, 2012). In particular, it contributes a novel explanation for the success factors for distant search efforts by organisations for solutions to complex problems by measuring the factors that determine success in terms of solution knowledge emergence, empirically evaluating the theoretical work of Afuah and Tucci (2012).

Third, it expands on the "loose alliances" perspective of meta-organisations, adding depth and breadth to the extant guidance in the alliances literature on managing knowledge sharing relationships outside of firm boundaries, particularly with competitors, developed by von Hippel and von Krogh (2003) with more recent empirical evidence that reflects the ongoing evolution of open source forms of production (West & Wood, 2014). The present study enables comparison of open source meta-organisation knowledge revealing strategies with the strategies for optimizing the forms of production that have been extensively considered in the alliances literature (c.f. Hamel, Doz, & Prahalad, 1989; Mowery, Oxley, & Silverman, 1996; Dyer & Singh, 1998; Inkpen, 1998; Ireland, Hitt, & Vaidyanath, 2002; Tsai, 2002; Grant & Baden-Fuller, 2003; Soekijad & Andriessen, 2003; Lavie, 2006; Gulati, Nohria, & Zaheer, 2000). In particular, it measures the degree to which organisations have control over the emergence of solution knowledge when that type of knowledge is controlled by another stakeholder in the meta-organisation.

Fourth, it provides a novel explanation for the failures of early attempts to set up open source forms of production as novel means of tapping into the knowledge that resides outside of traditional firm boundaries (DiBona, Ockman, & Stone, 1999; Lerner & Tirole, 2001, 2002; Anand, Glick, & Manz, 2002; Goldman & Gabriel, 2005; Wood & Guliani, 2005; Demil & Lecocq, 2006; DiBona, Stone, Cooper, 2008) such as that of Nokia as documented by West and Wood (2014) and RedHat, as described by Young (1999). It sets the stage for an open source theory of the firm that postulates that firms exist as a form of production because they are better able to manage the knowledge production efforts of actors within and outside the firm, as loosely defined by the knowledge boundaries of the firm that are distinct from the hierarchical and legal boundaries of the firm, by more efficiently and effectively manipulating the antecedent factors identified in this study to create appropriable value than markets, hierarchies, networks, alliances, or other forms of production. This open source theory of the firm is the starting point for a research agenda that seeks to answer the many questions that arise as the result of this study.

Contributions to practice

This study makes numerous contributions to practice, building on the extant conversations in the practitioner literature geared towards strategic managers in firms and the governance of open source meta-organisations, discussed in turn.

Contributions to the strategic management of firms

The outcomes of this dissertation make at least four practical contributions to the strategic managers of firms considering using deliberate knowledge revealing strategies by participating in open source meta organisations. First, this study identifies the antecedent factors for successful solution knowledge emergence following the use of deliberate knowledge spillover strategies in open source meta organisations. The identification and measurement of these factors enables more efficient allocation of resources and application of effort during the strategy's execution. Further, it provides a refined understanding of the levels of effects of these factors, allowing strategic managers to plan separate problem level, individual level, and organisation level strategies that collectively improve the effectiveness of desired outcomes. It also highlights the factors that affect different types of solution knowledge emergence, enabling organisations to promote the emergence of knowledge types that are the most useful for them depending on their recombinative capabilities and complementary assets. The identification and measurement of these factors sets the stage for firms achieving competitive advantage over competitors who are executing knowledge revealing strategies blindly without understanding the antecedent factors involved.

Second, it identifies the actors involved in open source meta-organisations and measures the characteristics of these actors that affect knowledge production such that strategic managers can better plan knowledge exchanges with actors and promote their involvement in resolving problems that are of relevance to the firm. In particular, the results suggest that firms must consider the subject matter expertise of potential individual and organisational solution knowledge creators in the meta-organisations when formulating the problem knowledge that they submit to ensure that the problem knowledge is codified in a manner that leverages that subject matter expertise. The results also suggest that firms must be aware of the knowledge paradigm biases of actors, both those of the actors within the firm who are participating in the collective knowledge production process and those of other actors in the meta-organisation, particularly when related to knowledge paradigms that are historically taboo in open source communities such as those related to Microsoft products and platforms (Raymond, 1999ab). The results further suggest that firms need to find ways to get high involvement "core" actors interested in their problem knowledge submissions to improve outcomes. The measurements of "core" actor factors that promote solution knowledge emergence provide better precision for improving the effectiveness sponsored open source development strategies, where firms remunerate certain actors in the open source meta-organisation in exchange for focus on their submitted problems, an emergent strategy that is discussed in the extant practitioner literature (c.f. Bergquist & Ljungberg, 2001; Dahlander & Magnusson, 2005; Chesbrough, 2007; West & O'Mahony, 2008; Enkel, Gassmann, & Chesbrough, 2009; Salter, Criscuolo, & Ter Wal, 2014; Spaeth, von Krogh, & He, 2014).

Third, it identifies the meta-organisation practices and cycles that affect knowledge production, enabling strategic managers to align their knowledge revealing strategies with the patterns within the meta-organisation to improve their effectiveness. In particular, the results reveal the importance of formulating problems in terms of the meta-organisation's stated priorities and dominant knowledge paradigms wherever possible. They also reveal the importance of adhering to meta-organisation prioritization cycles such as release cycles and social cycles such as weekdays and holidays. In cases where the solution knowledge needs of firms can be represented in multiple ways and the timing needs of the firm are flexible, strategic managers should choose to represent the problem knowledge in the platform, operating system, category, and attachment dominant knowledge paradigms of the meta-organisation at the time of submission, which should be timed to align with the meta-organisation's cycles. In cases where the solution knowledge paradigm or timing needs of the firm do not align well with the meta-organisation's priorities and/or dominant knowledge paradigms at the time of problem knowledge submission or with the meta-organisation's cycles, strategic managers may wish to consider alternate meta-organisations that are a better fit. This latter implication bridges the work of Alexy, George, and Salter (2013) by extending meta-organisation specific factors into the consideration of when to use knowledge revealing strategies by highlighting the need for a fit between solution knowledge needs and meta-organisation properties not previously considered in the literature.

Fourth, it provides advice to strategic managers on how to manage the process of knowledge creation when participating in meta-organisations. In particular, the results suggest that firms should curate problem knowledge throughout the knowledge production process, actively engaging with it throughout the period between submission and solution knowledge emergence. Simply submitting problem knowledge and waiting is not as effective as active engagement. The results also reveal which activities firms should take during the knowledge production process, highlighting the importance of focus on activities that create additional, emergent knowledge, and avoidance of activities that create contradictory or confounding knowledge. Further, the results provide guidance on the limits of additional emergent knowledge that can be consumed by the meta-organisation, cautioning strategic managers that excessive knowledge, such as very long descriptions or comments, may reduce rather than increase the effectiveness of solution knowledge creation.

These contributions to strategic management practice refine and extend existing conversations in the practice literature and set the stage for further refinement of open source strategies and business models.

Contributions to the governance of open source meta-organisations

The outcomes of this dissertation make at least five practical contributions to the members of open source meta-organisations who are in charge of their governance. First, this study reveals the importance of subject matter expertise in meta-organisations at both individual and organisation levels, for effective and efficient solution knowledge creation. The results suggest that open source meta-organisations may wish to carefully define a shared jargon and encourage actors to learn it and use it so that knowledge is more efficiently codified and applied. A shared subject matter expertise between actors improves outcomes for the whole meta-organisation. The results also suggest that the meta-organisation may wish to require new participants to spend time learning the meta-organisation's knowledge codification jargon and practices before allowing them to be more actively involved in the knowledge production process. This barrier to participation for less experienced actors may reduce their negative influence on the knowledge production process.

Second, it highlights the importance of clearly defined meta-organisation level knowledge production processes and the adherence to those processes by actors. A key governance activity in open source meta-organisations should be the state-by-state definition of the life cycle of the knowledge production process from problem knowledge submission to solution knowledge emergence. This process should be continuously reviewed with an eye for the frequency with which the knowledge life cycle is violated for the purpose of improving the knowledge production process. In fact, the Mozilla meta-organisation did this very thing shortly after the end of data collection for the present study (Rocha, de Oliveira, Valente, & Marques-Neto, 2016), updating the original knowledge life cycle process depicted in Figure 10, to the one depicted in Figure 33. Of particular note, additional entries and transitions were added to the knowledge life cycle process such that certain actions that are violations of the previous life cycle are no longer violations of the updated life cycle. The additional entries reflect another implication of the results of this study, which is that meta-organisations may wish to allow certain special actors, such as highly involved "core" developers, to bypass certain portions of the knowledge life cycle because the negative impact of life cycle violation is most prominent when the life cycle is violated by less involved actors. An alternative approach could be multiple "valid" life cycle transitions, where some transitions can only be validly used by specially designated actors and "regular" actors are forced to select from the typical life cycle transitions.

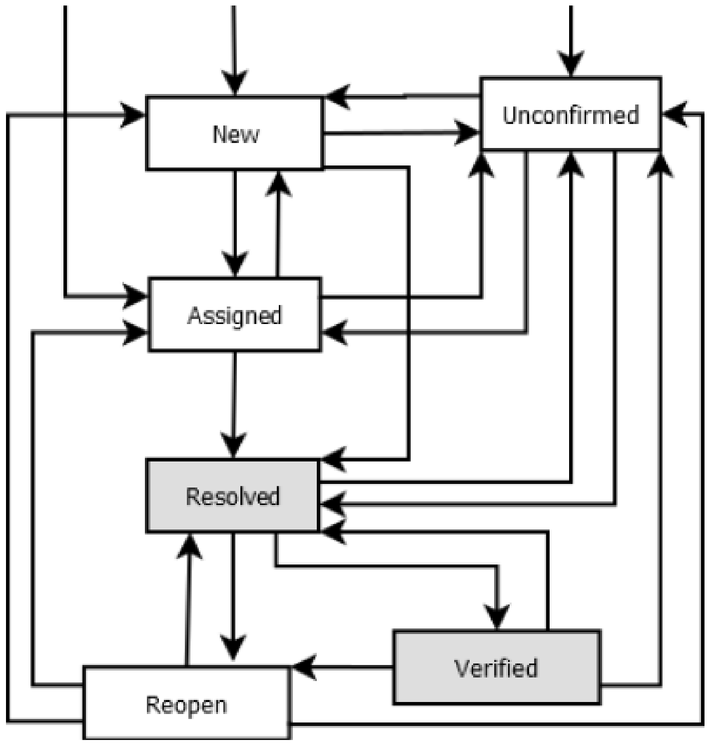

Figure 33: Updated knowledge flow life cycle in Mozilla open source meta-organisation

Third, it measures the effectiveness of specific meta-organisation processes and signalling artefacts, revealing those that work as intended and those that do not. In particular, the results suggest that reopening and reassigning loops in the knowledge production process generally work as intended when assessed at the problem level. However, the use of these redirection processes by certain individuals can result in consistently negative outcomes, suggesting that meta-organisations may wish to constrain reopening and reassigning to individuals who are associated with positive outcomes when using these processes. The results also suggest that votes work contrary to their intended purpose where more votes is associated with worse solution knowledge emergence at all levels of analysis. By contrast, the results suggest that flags are uniformly positive at all levels of analysis, with the number of flags implementing the intended purpose of votes. Given the visibility and constant use of these signalling artefacts by actors, meta-organisations may wish to carefully examine the reasons why they are functioning in a manner different from their intended purpose and address those effectiveness problems. For example, the results suggest that certain types of attachments represent valuable emergent knowledge in the knowledge production process, whereas other types of attachments hinder. Meta-organisations may wish to constrain the types of attachments that are allowed to mitigate the negative effects and promote useful emergent knowledge. The results further suggest that the bug duplication process generally functions as a useful signal of the value of the solution to a given problem. The number of duplications to a given problem may be a more effective signal than votes and may warrant its own tracking measure. This implication is of particular relevance to open source meta-organisation practice as it suggests that attempts to "reduce" duplicate bug submission may be misguided as they provide useful value signaling information. Rather, the duplicate bug submissions should be quantified and used as a value proxy when prioritizing meta-organisation knowledge creation effort.

Fourth, it reveals the connection between processes and artefacts and the actors who consume the knowledge they contain. In particular, the results suggest that the usefulness of votes, follows, and comments to the knowledge production effort varies based on the involvement of the actor creating these artefacts. Meta-organisations may wish to split the quantification of these measures based on the involvement of the actor that created them. One approach is to use a "badges" system, where degrees of involvement in the open source meta-organisation are designated by "gold", "silver", and "bronze" or similar typologies of actors. Votes, follows, comments, attachments, and other artefacts can then be weighed separately based on the badge of the actor who enacted them. This approach is already used in other knowledge creation communities such as Stack Exchange (Squire, 2014), where the differential levels affect both motivation of actors and perceptions of quality of the knowledge they produce (Rughiniş &

Matei, 2013; Hart & Sarma, 2014; Teo & Johri, 2014). Such approaches bridge the results of this study with the practitioner literature on leveraging "gamification", the use of game design elements including reward mechanisms for certain desired activities in non-game contexts such as in collective knowledge production endeavors, while improving the usefulness of measures, and enabling more effective and efficient open source meta-organisation governance (Lotufo, Passos, & Czarnecki, 2012; Dubois & Tamburrelli, 2013 Thongtanunam, et al., 2013; Vasilescu, Serebrenik, Devanbu, & Filkov, 2014; Wei, Chen, & Zhu, 2015). In deciding how to allocate actors to different involvement typologies, meta-organisations may wish to define involvement and the associated granted authority based on the association between individuals' actions and positive solution knowledge emergence outcomes, offering a novel representation of authority structures that builds upon those described in the open source governance literature (O'Mahony & Ferraro, 2007).

Fifth, it cautions that meta-organisations should not enact policies based on single factor or single level measures because these measures underspecify that underlying complexity of effects. The results shed light on some of the contradictory advice for open source meta-organisations found in the literature by revealing that even typically examined measures are far more complex than has historically been empirically examined. For example, severity levels are one of the most studied signalling artefacts in open source meta-organisations. The results reveal that there are at least four orthogonal dimensions of effects related to severity levels that are not captured when measuring the impact of severity level alone. The impact of severity level is better defined as a combination of the actor who sets the severity level, the actor or interprets and acts according to the severity level, the interacting priority level, and the number of changes of severity levels after knowledge submission. Each of these orthogonal dimensions alone depicts only a shadow projection of the full effect. Meta-organisations using empirical metrics to guide their governance activities are strongly cautioned to ensure that the metrics are not underspecified. Rather, policies should be derived from the empirical testing of models of the meta-organisation's processes and artefacts that are comprehensive and cross-level, lest the policies be enacted on the basis of spurious results.

These contributions to the governance of open source meta-organisations refine and extend existing conversations in the open source practice literature and set the stage for further refinement of open source governance structures and knowledge production processes.

Limitations

This dissertation focuses exclusively on the Mozilla Foundation as a representative open source meta-organisation. While this focus limits the generalizability of the findings, as different open source meta-organisations may have different antecedent factors for successful knowledge emergence, the choice is justified by the breadth of the organisation and the availability of the data. The Mozilla Foundation houses more than 20 distinct open source projects, each with its own participants, products, release cycles, and target users. This breadth allows for normalisation of disparate projects within the meta-organisation to address project-specific influences on the outcome. Further, the Mozilla Foundation is one of the oldest and most active open source meta-organisations in existence, with a high number of corporate participants, a broad geographic diversity, and a carefully documented history. Thus, the focus on the single meta-organisation ensured that the dissertation remained manageable and primes future research that compares the findings across different meta-organisations.

The dissertation extends the work of Alexy, George, and Salter. (2013), which considered the process that a firm goes through when considering whether to use a knowledge-revealing strategy, by starting at the firm's decision to use a knowledge-revealing strategy and bridging the gap between the use of this strategy and useful outcomes for the firm. Given the broad range of actors in the Mozilla meta-organisation, there is an inherent limitation in separating organisational actors from other actors who may not be using deliberate management strategies in their engagement in the knowledge production process. To address this limitation, careful operationalisation procedures were used to compare conservatively refined samples of the data that are more readily attributable to organisational action and intent. Nevertheless, absent direct insights from the participating organisations, there is a necessary limitation in the interpretation of the intent of the knowledge revealing that demands interpretive caution as a result. Other researchers have argued that the act of selective revealing to an open source meta-organisation itself is a deliberate strategic action or "profit-oriented behavior…which implies that the focal actor does not reveal out of principle but rather as a result of weighing the commercial pros and cons" (Henkel, Schöberl, & Alexy, 2014: 880). This dissertation uses that assumption given that its intent is to set the stage for future research that bridges strategic intent and the theorized resulting competitive advantage by identifying the contingent factors linking the two. With these factors identified and measured, it will be possible for future research to more directly assess strategic intent and test the assumption of Henkel, Schöberl, and Alexy (2014).

While not the purpose of this dissertation, another limitation of this dissertation is its inability to assess the net value increase of firm participation in open source meta-organisations. It focuses exclusively on the value creation side of knowledge revealing strategies, taking the position that these outcomes are valuable to the focal firm that chooses to participate and express a desire for these outcomes to the meta-organisation. There is also a cost side to participation in meta-organisations that is beyond the scope of this research as it is not reflected in the data set. As a result, it is possible that firms may suffer a net value loss even when all the antecedent factors are aligned and valuable knowledge assets emerge from the participation. This limitation is reasonable because value creation and costs are typically treated separately in the strategy literature and may depend on separate factors. Further, the factors affecting value creation establish a case for the possibility of net positive outcomes, making them an appropriate first research focus. Future research can weigh the costs to provide a more detailed picture of net value return for firms engaging in open source knowledge revealing strategies, building upon the emerging research stream empirically measuring the costs of open source strategies (c.f. Cassiman & Valentini, 2016).

A final limitation is that the Mozilla database may not contain all strategically relevant revealed knowledge for the purpose of assessing the factors that affect positive outcomes for the focal firm. In practice, open source meta-organisations use many different technologies to enable participation. Other sources of knowledge include mailing lists, code repositories, social media, and in-person meetings and interactions. The focus on the chosen database is justified because it is the most extensive and detailed data set for strategically-relevant interactions and because there are no known tractable ways to map the interactions that occur in this database to the other sources without significant distortion (Ayari, Meshkinfam, Antoniol, & Di Penta, 2007). Further, the focus on the single data source promotes manageability of the dissertation and the tractability of the operationalizations and statistical analyses. However, there may be other factors that affect the knowledge production process that are not captured in the database and examination of such external factors is left for future research.

Future research

This study builds the foundation for a research program on open source strategy that lies at the intersection of at the fields of strategic management, organisational management theory, technology and innovation management, engineering management and entrepreneurship. It opens up numerous avenues for theoretical refinement, empirical examination, and practitioner evaluation.

First, future research could usefully refine the analysis of the present data to better understand the large scale effects observed in the results. Such an examination should begin by collecting the significant factors described in Table 117 and modelling the pairwise interactions between each permutation of antecedent factor. In select cases, three-or-more-way interaction modelling may be warranted. This first step is necessary to account for the observed moderating effects of variables such as severity and priority and knowledge absorption impairment factors on one another. Next, a single model that includes the observed interactions should be created for each outcome variable at each level, with comparison based refinement of candidate models, including controls, based on AIC and effect size until a satisfactory single "comprehensive" model is defined. Finally, this comprehensive model should be evaluated for clustering of common factors to refine the distinctiveness of the factors and eliminate as much co-linearity as possible (c.f. Husson, Lê & Pagès, 2011). The result of this process should be a single, reduced, refined model that captures the largest orthogonal cluster antecedent factors affecting solution knowledge emergence dependent variables.

Second, with a reduced model in hand, future research could conduct longitudinal analysis given the underlying data can be represented as time series. The problem and solution submission and emergence timings in the data make them suitable for econometric panel data analysis (c.f. Frees, 2004; Baltagi, 2013) and survival hazard analysis methods (c.f. Cox & Oakes, 1984; Moore, 2016). Such longitudinal modelling allows for the examination of firm-level experience effects, extending the current model of firm solution knowledge emergence heterogeneity to account for variances over time within and amongst firms. More directly measuring experience effects allows for better refinement of the understanding of second and higher order effects on solution knowledge outcomes. It also allows for better control of time-related factors than the simple control variables used in the cross-sectional models in the present study. In particular, it allows for disambiguation of factors that are a function of time and time itself, such as activities that take time to perform. The outcomes of such modelling would also include the ability to compare the top 3, 10, or X firms over time as measured by a host of factors, to get a clearer picture of the competitive landscape amongst meta-organisation participants.

Third, future research could extend the levels of analysis considered in the data by defining and measuring levels not considered in the present study. Such new levels of analysis could include a "community" level that compares and contrasts the effects of different definitions of community debated in the extant open source literature (O'Mahony, 2007); an "activity" level that examines activities as their own unit of analysis independent from problems; and, the knowledge type representations used in the meta-organisation, i.e., platform, operating system, classification, product, and component, each as an independent level that aggregates the problems and solutions represented in that paradigm.

Fourth, future research could incorporate new open source meta-organisation data sources to test the limits to the generalizability of the results of the present study. Two data sources that are in a comparable format are the development databases of the Eclipse Foundation and the Apache Foundation. Like the Mozilla Foundation, these two organisations maintain some of the largest and most active open source projects and have detailed data on firm participation in their meta-organisations dating back decades. Incorporation of data sets from these or other open source meta-organisations would allow for comparison of factors across meta-organisations and facilitate control for within-meta-organisation heterogeneity. The analytical code for the present study was deliberately written in a modular fashion such that it could be readily adapted to similar data sources with minimal integration modifications, facilitating this next step in the research program.

Fifth, future research could incorporate other data sources to cross-reference the implications of participation in meta-organisation with other metrics that are of relevance to strategic management research and practice. In particular, COMPUSTAT data are amenable to matching to the firm level data in the present research. Such a combination could allow for the modelling of the antecedent and outcome factors of meta-organisation participation relative to traditional measures such as sales, revenues, expenses, and so on. The effects of firm use of open source strategies, over time, on profits relative to competitors in the same industry who do not use open source strategies would be an excellent empirical test to evaluate an open source theory of the firm. Preliminary examination of COMPUSTAT data revealed that data for more than 1800 firms can be definitively matched to the open source meta-organisation data used in the present research, suggesting that this avenue of future research could yield fruitful novel insights. When combined with the previous future research suggestion, it could be possible to track the same firm's actions across different meta-organisations over time, which could dramatically increase our understanding of the deliberate and emergent strategies they employ.

Sixth, future research may wish to craft novel methods and apply them to addressing the data distortion hurdles encountered by Ayari, et al. (2007) when attempting to match the problem tracking database used in this study to the actual code level changes submitted to code repositories. More recent code repositories, such as GitHub (Marlow, Dabbish, & Herbsleb, 2013) facilitate the matching of contributor actions to code submitted, which may enable novel methods to consider factors affecting the success of knowledge revealing strategies that were beyond the scope of the present study. In particular, the "size" of software code patches, often measured as "number of lines of code" is a plausible hidden moderating variable for many of the effects observed in the results of this study (Baysal, et al., 2012b; Zhang, et al., 2012) which could be more directly measured if such hurdles were addressed.

Seventh, future research may wish to use alternate methods to test some of the assumptions of the present study, such as the assumption of strategic intent in the use of knowledge revealing strategies as per Henkel, Schöberl, and Alexy (2014). A combination of qualitative methods and multiple case study methods may be fruitful in providing richer detail on the intended vs. realised strategies of firms that participate in open source meta-organisations, enriching the accounts of MacAulay (2017) and others by triangulating them with the empirical measurements in this study and their extensions in future work. In particular, such research enables a more robust measurement of the costs of knowledge revealing strategies which are necessary in order to determine if such strategies result in net value for firms that use them, building on the work of Cassiman and Valentini (2016).

Eighth, future research may wish to study other types of meta-organisations that have different degrees of stratification and boundary characteristics than open source meta-organisations to evaluate the degree of generalizability to other novel organisational forms of production. In particular, rich data have been collected during open innovation “contests” by major firms that seek to innovate with the help of their customers (Chesbrough, Vanhaverbeke, & West, 2006; Laursen & Salter, 2006; Chesbrough & Appleyard, 2007; Enkel, Gassmann, & Chesbrough, 2009; Dahlander & Gann, 2010) in a broad range of industries that extend beyond high tech (Chesbrough & Crowther, 2006). These data offer a prime opportunity for extending this research into the other quadrants of Gulati, Puranam, and Tushman’s (2012) taxonomy of meta-organisations.

Ninth, the present data set is sufficiently rich that it may be useful for testing other research questions that are of relevance to strategic management and related fields. Given that this study is the first management study to use this type of data and consequently has developed novel methods to tap into the richness therein, there is an opportunity to apply these methods to other strategy research questions. In particular, the second outcome of the use of knowledge revealing strategies by firms proposed by Alexy, George, and Salter (2013) is what they refer to as “induced isomorphism”, which they define as “deliberate strategic action to induce other [participants] to become more similar to the focal firm, particularly with respect to the production of knowledge” (272). The notion of inducing imitative behaviour by a competitor is contrary to many of the fundamental assumptions in extant strategic management research (c.f. Barney, 1986, 1991; Peteraf, 1993; Peteraf & Barney, 2003) making it a prime target for empirical evaluation in this context of novel forms of production. The present data are amenable to network analysis where dyads and clusters of firms could be modelled for the degree to which they influence each other to resemble each other as measured by appropriate similarity factors such as code reuse (Haefliger, von Krogh, & Spaeth, 2008).

Tenth, as with all good studies, the results of the present study raise many new questions. In addition to the nine major avenues of future research discussed above, numerous refinement opportunities exist to deepen the understanding of many of the effects of the factors measured in this study. These suggested extensions are discussed in the context of the related factors' results in Chapter Six: Results and Discussion and summarized in Table 118.

Related factors	Extension in future research
Descriptions, comments	Assessment of stratified range of lengths and contents; randomized samples of descriptions/comments analyzed and classified manually by researchers
Actor involvement	Assessment of stratified sample of degrees of actor involvement to determine impact on weights of other factors
All	Non-linear modelling of factors conducive to such analysis
Attachments	Assessment of stratified sample of types of attachments to better measure difference of effect amongst attachment types
Signalling	Assessment of stratified sample of number of changes to signalling variables to better measure nuanced effects beyond single change
Keywords	More precise measurement of nature of knowledge contained in keywords using clusters/tag clouds of related words and topics
Severity, actor involvement	Measurement of the degree to which developers or other actors with differing degrees of involvement judge reported severity level of problems better (or worse) than other actors using an independent objective severity level measure as reference
Change signalling	Separate cases when change signalling does not take place because change was not needed because information was accurate from cases where there was a lack of interest in problem so change was not signalled despite incorrect information
Absorptive capacity impairment	More precise measurement of absorptive capacity impairment factors using more nuanced ranges and interaction regression modelling
Assignment and development time	Measurement of assignment time and development time outcome effects at individual and organisation levels of analysis with data able to disambiguate individual and organisational contributions to timing effects without problem-level aggregation endogeneity
Individual nestedness in organisations	More nuanced and precise measurement of the effect of individual nestedness in organisations using statistical methods such as hierarchical linear modeling and structural equation modeling
Knowledge production processes	Comparison of effectiveness and efficiency of knowledge production processes used in Mozilla meta-organisation to those used in traditional organisations
Problem classification	Refinement of automatic problem classification routines using split training/prediction data sets and variations on the n-gram and compression algorithm approaches used in this study
Knowledge life cycle	Measure impact of change to bug life cycle process used in Mozilla meta-organisation and compare outcomes relative to pre-change data

Table 118: Opportunities for future refinement and extension of factors measured in this study

Conclusion

This dissertation sought to answer the research question, "*What are the factors driving successful solution knowledge emergence?*" The results of a comprehensive analysis of longitudinal data spanning from 1998 to end of 2012 obtained from the Mozilla meta-organisation identified 35 factors that drive successful solution knowledge emergence. These factors exert their influence in 180 different ways, across three distinct levels of analysis, on a broad range of successful solution knowledge emergence outcome measures. These factors extend the research of Alexy, George, and Salter (2013) by providing a novel explanation for the heterogeneity of the success of organisations that use deliberate knowledge revealing strategies. This dissertation provides one of the first comprehensive empirical assessments of strategic factors affecting firm participation in the open source meta-organisation non-traditional organisational form of production, resulting in numerous contributions to both research and practice. The outcomes of this research set the stage for an ongoing research program on open source strategy.

BIBLIOGRAPHY

Adam, F. 2014. **Measuring national innovation performance – The innovation union scoreboard revisited**. Springer: Boston, MA, USA.

Afuah, A., & Tucci, C.L. 2012. Crowdsourcing As a Solution to Distant Search. ***Academy of Management Review***. 37(3): 355-375.

Agarwal, R., Anand, J., Bercovitz, J., & Croson, R. 2012. Spillovers across organisational architectures: The role of prior resource allocation and communication in post-acquisition coordination outcomes. ***Strategic Management Journal***. 33: 710-733.

Ahmed, M.F., & Gokhale, S.S. 2009. Linux bugs: Lifecycle, resolution and architectural analysis. ***Information and Software Technology***. 51: 1618-1627.

Alavi, M., & Leidner, D.E. 2001. Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research Issues. ***MIS Quarterly***. 25(1): 107-136.

Alavi, M., Kayworth, T.R., & Leidner, D.E. 2014. An Empirical Examination of the Influence of Organizational Culture on Knowledge Management Practices. ***Journal of Management Information Systems***. 22(3): 191-224.

Allarakhia, M., & Walsh, S. 2011. Managing knowledge assets under conditions of radical change: The case of the pharmaceutical industry. ***Technovation***. 31(2-3): 105-117.

Allarakhia, M., & Walsh, S. 2012. Analyzing and organizing nanotechnology development: Application of the institutional analysis development framework to nanotechnology consortia. ***Technovation***. 32(3-4): 216-226.

Alexy, O., George, G., & Salter, A.J. 2013. Cui Bono? The selective revealing of knowledge and its implications for innovative activity. ***Academy of Management Review***. 38(2): 270-291.

Alexy, O., Henkel, J., & Wallin, M.W. 2013. From closed to open: Job role changes, individual predispositions, and the adoption of commercial open source software development. ***Research Policy***. 42(8): 1325-1340.

Alnuaimi, T., & George, G. 2016. Appropriability and the retrieval of knowledge after spillovers. ***Strategic Management Journal***. 37(7): 1263-1279.

Alvesson, M. 1993. Organizations as rhetoric: Knowledge-intensive firms and the struggle with ambiguity. ***Journal of Management Studies***. 30(6): 997-1015.

Ambrosini, V., & Bowman, C. 2001. Tacit knowledge: Some suggestions for operationalization. ***Journal of Management Studies***. 38(6): 811-829.

An, X., Deng, H., Chao, L., & Bai, W. 2014. Knowledge management in supporting collaborative innovation community capacity building. ***Journal of Knowledge Management***. 18(3): 574-590.

Anand, V., Glick, W.H, & Manz, C.C. 2002. Thriving on the knowledge of outsiders: Tapping organizational social capital. ***Academy of Management Perspectives***. 16(1): 87-101.

Antoniol, G., Ayari, K., Khomh, F., & Guéhéneuc, Y-G. 2008. Is it a bug or an enhancement? A text-based approach to classify change requests. ***Proceedings of the 2008 conference of the center for advanced studies on collaborative research: Meeting of minds***. ACM: New York, NY, USA.

Anvik, J., Hiew, L., & Murphy, G.C. 2005. Coping with an open bug repository. ***Proceedings of the 2005 OOPSLA workshop on Eclipse technology***. 35-39. ACM: New York, NY, USA.

Anvik, J., Hiew, L., & Murphy, G.C. 2006. Who should fix this bug? ***Proceedings of the 28th international conference on software engineering***: 361-370. ACM: New York, NY, USA.

Anvik, J., & Murphy, G.C. 2011. Reducing the effort of bug report triage: Recommendations for development-oriented decisions. ***ACM Transactions on Software Engineering and Methodology***. 20(3): Article No. 10.

Appleyard, M.M. 1996. How does knowledge flow? Interfirm patterns in the semiconductor industry. ***Strategic Management Journal***. 17(S2): 137-154.

Ardichvili, A., Page, V., & Wentling, T. 2003. Motivation and barriers to participation in virtual knowledge-sharing communities of practice. ***Journal of Knowledge Management***. 7(1): 64-77.

Arend, R.J. 2006. Tests of the resource-based view: Do the empirics have any clothes? ***Strategic Organisation***. 4(4): 409-422.

Argote, L., McEvily, B., & Reagans, R. 2003. Managing Knowledge in Organizations: An Integrative Framework and Review of Emerging Themes. ***Management Science***. 49(4): 571-582.

Armstrong-Warwick, S., Thompson, H.S., McKelvie, D., & Petitpierre, D. 1994. Data in your language: The ECI multilingual corpus 1. ***Proceedings of the international workshop on sharable natural language resources***: 97-106. Nara, Japan.

Asay, M. 2007. The ironic rise of the Mac among open source developers. ***CNET***. Available at: https://www.cnet.com/news/the-ironic-rise-of-the-mac-among-open-source-developers/

Au, Y.A., Carpenter, D., Chen, X., & Clark, J.G. 2009. Virtual organisational learning in open source software development projects. ***Information & Management***. 46: 9-15.

Autio, E., Dahlander, L., & Frederiksen, L. 2013. Information exposure, opportunity evaluation and entrepreneurial action: An empirical investigation of an online user community. ***Academy of Management Journal***. 56(5): 1348-1371.

Ayari, K., Meshkinfam, P., Antoniol, G., & Di Penta, M. 2007. Threats on building models from CVS and Bugzilla repositories: The Mozilla case study. ***Proceedings of the 2007 conference of the center for advanced studies on collaborative research***. 215-228.

Bagozzi, R.P., & Dholakia, U.M. 2006. Open source software user communities: A study of participation in Linux user groups. ***Management Science***. 52(7): 1099-1115.

Baldwin, C., & von Hippel, E. 2011. Modeling a Paradigm Shift: From Producer Innovation to User and Open Collaborative Innovation. ***Organization Science***. 22(6): 1399-1417.

Baltagi, B.H. 2013. **Econometric Analyais of Panel Data**. 5th Edition. Wiley: Hoboken, NJ, USA.

Barney, J.B. 1986. Strategic factor markets: Expectations, luck and business strategy. ***Management Science***. 42: 1231-1241.

Barney, J.B. 1991. Firm resources and sustained competitive advantage. ***Journal of Management***. 17: 99-120.

Baskerville, R., & Dulipovici, A. 2006. The theoretical foundations of knowledge management. ***Knowledge Management Research & Practice***. 4(2): 83-105.

Basili, V.R., & Caldiera, G. 1995. Improve Software Quality by Reusing Knowledge and Experience. ***MIT Sloan Management Review***. 37(1): 55-64.

Bates, D., Mächler, M., Bolken, B.M., & Walker, S.C. 2015. Fitting Linear Mixed-Effects Models Using lme4. ***Journal of Statistical Software***. 67(1). DOI: 10.18637/jss.v067.i01

Baysal, O., Holmes, R., & Godfrey, M.W. 2013. Developer Dashboards: The Need for Qualitative Analytics. ***IEEE Software***: 30(4): 46-52.

Baysal, O., Holmes, R., & Godfrey, M.W. 2012a. Revisiting bug triage and resolution practices. ***Proceedings of User Evaluation for Software Engineering Researchers (USER) 2012***: 29-30. Zurich, Switzerland.

Baysal, O., Kononenko, O., Holmes, R., & Godfrey, M.W. 2013. **The influence of non-technical factors on code review**. Working paper. University of Waterloo David R. Cheriton School of Computer Science: Waterloo, ON, Canada.

Baysal, O., Kononenko, O., Holmes, R., & Godfrey, M.W. 2012b. The secret life of patches: A Firefox case study. ***Proceedings of the 19th Working Conference on Reverse Engineering (WCRE 2012)***. Kingston, ON, Canada.

Bedeian, A.G., Taylor, S.G., & Miller, A.N. 2010. Management Science on the Credibility Bubble: Cardinal Sins and Various Misdemeanors. ***Academy of Management Learning & Education***. 9(4): 715-725.

Belenzon, S., & Schankerman, M. 2015. Motivation and sorting of human capital in open innovation. ***Strategic Management Journal***. 36: 795-820.

Bell, G.G., & Zaheer, A. 2007. Geography, Networks, and Knowledge Flow. ***Organization Science***. 18(6): 955-972.

Benkler, Y. 2002. Coase's penguin, or, Linux and the nature of the firm. ***Yale Law Journal***. 112(3): 369-446.

Benner, M.J., & Tushman, M.L. 2003. Exploitation, Exploration, and Process Management: The Productivity Dilemma Revisited. ***Academy of Management Review***. 28(2): 238-256.

Bergquist, M., & Ljungberg, J. 2001. The power of gifts: Organizing social relationships in open source communities. ***Information Systems Journal***. 11(4): 305-320.

Bettenburg, N., Just, S., Schröter, A., Weiss, C., Premraj, R., & Zimmermann, T. 2008. What makes a good bug report? ***Proceedings of SIGSOFT 2008***: 308-318. Atlanta, Georgia, USA.

Bhardwaj, M., & Monin, J. 2006. Tacit to explicit: An interplay shaping organization knowledge. ***Journal of Knowledge Management***. 10(3): 72-85.

Bhatt, G.D. 2000. Organizing knowledge in the knowledge development cycle. ***Journal of Knowledge Management***. 4(1): 15-26.

Bingham, C.B., & Eisenhardt, K.M. 2011. Rational heuristics: The 'simple rules' that strategists learn from process experience. ***Strategic Management Journal***. 32: 1437-1464.

Birkinshaw, J., Nobel, R., & Ridderstråle, J. 2002. Knowledge as a Contingency Variable: Do the Characteristics of Knowledge Predict Organization Structure? ***Organization Science***. 13(3): 274-289.

Birkinshaw, J., & Sheehan, T. 2002. Managing the knowledge life cycle. ***MIT Sloan Management Review***. 44(1): 75-83.

Bitzer, J., & Schröder, P.J.H. 2005. Bug-fixing and code-writing: The private provision of open source software. ***Information Economics and Policy***. 17: 389-406.

Bliese, P.D. 2000. Within-group agreement, non-independence, and reliability: Implications for data aggregation and analysis. In Klein, K.J., & Kozlowski, S.W.J. Eds. **Multilevel theory, research, and methods in organisations**. 349-381. Jossey-Bass: San Francisco, CA, USA.

Bock, G-W., Mahmood, M., Sharma, S., & Kang, Y. J. 2010. The Impact of Information Overload and Contribution Overload on Continued Usage of Electronic Knowledge Repositories. ***Journal of Organisational Computing and Electronic Commerce***. 20(3): 257-278.

Bogers, M., Afuah, A., & Bastian, B. 2010. Users as innovators: A review, critique, and future research directions. ***Journal of Management***. 36(4): 857-875.

Bonaccorsi, A., & Rossi, C. 2003. Why open source software can succeed. ***Research Policy***. 32(7): 1243-1258.

Bonaccorsi, A., & Rossi, C. 2006. Comparing motivations of individual programmers and firms to take part in the open source movement: from community to business. ***Knowledge, Technology and Policy***. 18(4): 40-64.

Boudreau, K.J., & Lakhani, K.R. 2009. How to Manage Outside Innovation. ***MIT Sloan Management Review***. 50(4): 69-76.

Bougie, G., Treude, C., German, D.M., & Storey, M-A. 2010. A comparative exploration of FreeBSD bug lifetimes. ***Proceedings of the 7th IEEE Working Conference on Mining Software Repositories***: 106-109. Cape Town, South Africa.

Bozeman, B., & Rogers, J.D. 2002. A churn model of scientific knowledge value: Internet researchers as a knowledge value collective. ***Research Policy***. 31(5): 769-794.

Bresnahan, T.F., & Greenstein, S. 1999. Technological competition and the structure of the computer industry. ***Journal of Industrial Economics***. 47(1): 1-40.

Breu, S., Premraj, R., Sillito, J., & Zimmermann, T. 2010. Information needs in bug reports: Improving cooperation between developers and users. ***Proceedings of CSCW 2010***: 301-310. Savannah, Georgia, USA.

Brown, J.S., & Duguid, P. 1991. Organizational Learning and Communities-of-Practice: Toward a Unified View of Working, Learning, and Innovation. ***Organization Science***. 2(1): 40-57.

Burnham, K.P., & Anderson, D.R. 2002. **Model Selection and Multimodel Inference: A practical Information-Theoretic Approach**. Springer-Verlag: New York, NY, USA.

Canfora, G., & Cerulo, L. 2006. Where is bug resolution knowledge stored? ***Proceedings of the 3rd IEEE Working Conference on Mining Software Repositories***: 183-184. Shanghai, China.

Carayannis, E.G., Alexander, J., & Ioannidis, A. 2000. Leveraging knowledge, learning, and innovation in forming strategic government-university-industry (GUI) R&D partnerships in the US, Germany, and France. ***Technovation***. 20(9): 477-488.

Carayannopoulos, S., & Auster, E.R. 2010. External knowledge sourcing in biotechnology through acquisition versus alliance: A KBV approach. ***Research Policy***. 39(2): 254-267.

Carlile, P.R. 2004. Transferring, Translating, and Transforming: An Integrative Framework for Managing Knowledge Across Boundaries. ***Organization Science***. 15(5): 555-568.

Carlucci, D., Marr, B., & Schiuma, G. 2004. The knowledge value chain: How intellectual capital impacts on business performance. ***International Journal of Technology Management***. 27(6-7): 575-590.

Casadesus-Masanell, R., & Llanes, G. 2011. Mixed source. ***Management Science***. 57(7): 1212-1230.

Cassiman, B., & Valentini, G. 2016. Open innovation: Are inbound and outbound knowledge flows really complementary? ***Strategic Management Journal***. 37(6): 1034-1046.

Cavnar, W.B. & Trenkle, J.M. 1994. N-gram-based text categorization. ***Proceedings of SDAIR-94, 3rd annual symposium on document analysis and information retrieval***: 161-175. Las Vegas, NV, USA.

Chambers, J.M. 1992. Chapter 4: Linear models. In Chambers, J.M., & Hastie, T.J. Eds. **Statistical Models in S**. Wadsworth & Brooks/Cole: Monterey, CA, USA.

Chambers, J.M. 1998. **Programming with data. A guide to the S language**. Springer: New York, NY, USA.

Chan, D. 1998. Functional relations among constructs in the same content domain at different levels of analysis: A typology of composition models. ***Journal of Applied Psychology***. 83(2): 234-246.

Chang, R.M., Kauffman, R.J., & Kwon, Y. 2013. Understanding the paradigm shift to computational social science in the presence of big data. ***Decision Support Systems***. 63: 67-80.

Chen, Y-S., Lin, M-J. J., & Chang, C-H. 2009. The positive effects of relationship learning and absorptive capacity on innovation performance and competitive advantage in industrial markets. ***Industrial Marketing Management***. 38(2): 152-158.

Chesbrough, H.W., Vanhaverbeke, W., & West, J. Eds. 2006. **Open Innovation**. Researching a New Paradigm. Oxford University Press: Oxford, UK.

Chesbrough, H.W. 2007. Why Companies Should Have Open Business Models. ***MIT Sloan Management Review***. 48(2): 22-28.

Chesbrough, H.W. & Appleyard, M.M. 2007. Open innovation and strategy. ***California Management Review***. 50(1): 57-77.

Chesbrough, H.W. & Crowther, A.K. 2006. Beyond high tech: Early adopters of open innovation in other industries. ***R&D Management***. 36(3): 229-236.

Cheung, C.F., Lee, W.B., & Wang, Y. 2005. A multi-facet taxonomy system with applications in unstructured knowledge management. ***Journal of Knowledge Management***. 9(6): 76-91.

Chilana, P.K., Ko, A.J., & Wobbrock, J.O. 2010. Understanding expressions of unwanted behavior in open bug reporting. ***Proceedings of the 2010 IEEE Symposium on Visual Languages and Human-Centric Computing***: 203-206. Leganés, Spain.

Chilton, M.A., & Bloodgood, J.M. 2007. The Dimensions of Tacit & Explicit Knowledge: A Description and Measure. ***Proceedings of the 40th Annual Hawaii International Conference on System Sciences (HICSS 2007)***. Waikoloa, HI, USA.

Coase, R.H. 1937. The nature of the firm. ***Economica***. 4: 386-405.

Cohen, D. 1998. Toward a Knowledge Context: Report on the First Annual U.C. Berkeley Forum on Knowledge and the Firm. ***California Management Review***. 40(3): 22-39.

Cohen, J. 1988. **Statistical power analysis for the behavioral sciences**. 2nd Edition. Lawrence Erlbaum Associates: New Jersey, NJ, USA.

Cohen, W.M., & Levinthal, D.A. 1990. Absorptive Capacity: A New Perspective on Learning and Innovation. ***Administrative Science Quarterly***. 35(1): 128-152.

Cohen, J. 1992. A power primer. ***Psychological Bulletin***. 112(1): 155-159.

Colazo, J., & Fang, Y. 2009. Impact of license choice on Open Source Software development activity. ***Journal of the Association for Information Science and Technology***. 60(5): 997-1011.

Colombo, M.G., Piva, E., & Rossi-Lamastra, C. 2014. Open innovation and within-industry diversification in small and medium enterprises: The case of open source software firms. ***Research Policy***. 43: 891-902.

Contreras-Reyes, J.E., & Arellano-Valle, R.B. 2012. Kullback-Leibler divergence measure for multivariate skew-normal distributions. ***Entropy***. 14: 1606-1626.

Cook, S.D.N., & Brown, J.S. 1999. Bridging Epistemologies: The Generative Dance Between Organizational Knowledge and Organizational Knowing. ***Organization Science***. 10(4): 381-400.

Cooke, P. 2005. Regionally asymmetric knowledge capabilities and open innovation: Exploring 'Globalisation 2'—A new model of industry organisation. ***Research Policy***. 34(8): 1128-1149.

Cowan, R. 2001. Expert systems: aspects of and limitations to the codifiability of knowledge. ***Research Policy***. 30(9): 1355-1372.

Cox, D.R., & Oakes, D. 1984. **Analysis of Survival Data**. Monographs on statistics and applied probability. Chapman & Hall: London, UK.

Cragg, J.G., & Uhler, R.S. 1970. The demand for automobiles. ***Canadian Journal of Economics***. 3: 386-406.

Cribari-Neto, F., & Zeileis, A. 2010. Beta Regression in R. ***Journal of Statistical Software***. 34(2): 1-24.

Crossan, M.M., & Apaydin, M. 2009. A Multi-Dimensional Framework of Organizational Innovation: A Systematic Review of the Literature. ***Journal of Management Studies***. 47(6): 1154-1191.

Crowston, K., Wei, K., Li, Q., & Howison, J. 2006. Core and Periphery in Free/Libre and Open Source Software Team Communications. ***Proceedings of the 39th Annual Hawaii International Conference on System Sciences***. Kauia, HI, USA.

Crowston, K., & Howison, J. 2005. The social structure of free and open source software development. ***First Monday***. 10(2). DOI: http://dx.doi.org/10.5210/fm.v10i2.1207

Crowston, K., & Scozzi, B. 2008. Bug fixing practices within free/libre open source software development teams. ***Journal of Database Management***. 19(2): 1-30.

Čubranić, D., & Murphy, G.C. 2004. Automatic bug triage using text categorization. ***Proceedings of the Sixteenth International Conference on Software Engineering & Knowledge Engineering***: 92-97. Banff, AB, Canada.

Cusumano, M.A., & Selby, R.W. 1995. **Microsoft Secrets**. Simon & Schuster: New York, NY, USA.

D'Ambros, M., Lanza, M., & Pinzger, M. 2007. "A Bug's Life" Visualizing a Bug Database. ***Proceedings of 4th IEEE International Workshop on Visualizing Software for Understanding and Analysis***. Banff, AB, Canada.

Dahlander, L., & Frederiksen, L. 2012. The core and cosmopolitans: A relational view of innovation in user communities. ***Organization Science***. 23(4): 988-1007.

Dahlander, L., & Gann, D.M. 2010. How open is innovation? ***Research Policy***. 39(6): 699-709.

Dahlander, L., & Magnusson, M.G. 2005. Relationships between open source software companies and communities: Observations from Nordic firms. ***Research Policy***. 34: 481-493.

Dahlander, L., & Wallin, M.W. 2006. A man on the inside: Unlocking communities as complementary assets. ***Research Policy***. 35(8): 1243-1259.

Dahlander, L., & O'Mahony, S. 2011. Progressing to the center: Coordinating project work. ***Organization Science***. 22(4): 961-979.

Dalle, J-M., den Besten, M., & Masmoudi, H. 2008. Channeling Firefox developers: Mom and dad aren't happy yet. In: Russo, B., Damiani, E., Hissam, C., Lundell, B., & Succi, G. (Eds.). 2008. **Open source Development, Communities and Quality**. IFIP International Federation for Information Processing, Volume 275. Springer: Boston, MA, USA.

Dalle, J-M., den Besten, M., Masmoudi, H., & David, P.A. 2008. Bug-patching for Mozilla's Firefox. Working Paper. http://ssrn.com/abstract=1112811

Damanpour, F. 1996. Organizational Complexity and Innovation: Developing and Testing Multiple Contingency Models. ***Management Science***. 42(5): 693-716.

Dantas, E., & Bell, M. 2009. Latecomer firms and the emergence and development of knowledge networks: The case of Petrobras in Brazil. ***Research Policy***. 38(5): 829-844.

Das, A. 2003. Knowledge and Productivity in Technical Support Work. ***Management Science***. 49(4): 416-431.

Davenport, T., Barth, P., & Bean, R. 2012. How Big Data Is Different. ***MIT Sloan Management Review***. 54(1): 43-46.

Davis, S., & Botkin, J. 1994. The Coming of Knowledge-Based Business. ***Harvard Business Review***. September-October: 165-170.

Davison, A.C., & Hinkley, D.V. 1997. **Bootstrap Methods and their Application**. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press: Cambridge, UK.

Decarolis, D.M., & Deeds, D.L. 1999. The impact of stocks and flows of organizational knowledge on firm performance: An empirical investigation of the biotechnology industry. ***Strategic Management Journal***. 20: 953-968.

Demil, B., & Lecocq, X. 2006. Neither market nor hierarchy nor network: the emergence of bazaar governance. ***Organisation Studies***. 27(10): 1447-1466.

DiBona, C., Stone, M., & Cooper, D. 2008. **Open Sources 2.0**. O'Reilly & Associates: Sebastopol, CA, USA.

DiBona, C., Ockman, S., & Stone, M. 1999. **Open Sources: Voices from the Open Source Revolution**. O'Reilly & Associates: Sebastopol, CA, USA.

Dierickx, I., & Cool, K. 1989. Asset Stock Accumulation and Sustainability of Competitive Advantage. ***Management Science***. 35(12): 1504-1511.

Dewar, R.D., & Dutton, J.E. 1986. The Adoption of Radical and Incremental Innovations: An Empirical Analysis. ***Management Science***. 32(11): 1422-1433.

Dierickx, I, & Cool, K. 1989. Asset stock accumulation and sustainability of competitive advantage. ***Management Science***. 35: 1504-1511.

Dit, B., & Marcus, A. 2008. Improving the readability of defect reports. ***Proceedings of the 2008 international workshop on recommendation systems for software engineering***: 47-49. Atlanta, GA, USA.

Dittrich, K., & Duysters, G. 2007. Networking as a Means to Strategy Change: The Case of Open Innovation in Mobile Telephony. ***The Journal of Product Innovation Management***. 24(6): 510-521.

Du, J., Leten, B., & Vanhaverbeke, W. 2014. Managing open innovation projects with science-based and market-based partners. ***Research Policy***. 43: 828-840.

Dubois, D.J., & Tamburrelli, G. 2013. Understanding gamification mechanisms for software development. ***Proceedings of the 2013 9th Joint Meeting on Foundations of Software Engineering***: 659-662. Saint Petersburg, Russia.

Dyer, J.H., & Nobeoka, K. 2000. Creating and managing a high-performance knowledge-sharing network: The Toyota case. ***Strategic Management Journal***. 21: 345-367.

Dyer, J.H., & Singh, H. 1998. The Relational View: Cooperative Strategy and Sources of Interorganizational Competitive Advantage. ***Academy of Management Review***. 23(4): 660-679.

Easterby-Smith, M., & Prieto, I.M. 2008. Dynamic Capabilities and Knowledge Management: An Integrative Role for Learning? ***British Journal of Management***. 19(3): 235-249.

Eisenhardt, K.M., & Martin, J.A. 2000. Dynamic capabilities: What are they? ***Strategic Management Journal***. 21(10-11): 1105-1121.

Eisenhardt, K.M., & Santos, F. 2002. Knowledge-based view: A new theory of strategy? In: Pettigrew, A.M., Thomas, H., & Whittington, R. Eds. 2006. **Handbook of Strategy and Management**. SAGE Publications: Thousand Oaks, CA, USA.

Enkel, E., Gassmann, O., & Chesbrough, H. 2009. Open R&D and open innovation: Exploring the phenomenon. ***R&D Management***. 39(4): 311-316.

Etzkowitz, H. 1997. From zero-sum to value-added strategies: The emergence of knowledge-based industrial policy in the states of the United States. ***Policy Studies Journal***. 25(3): 412-424.

Fahey, L., & Prusak, L. 1998. The Eleven Deadliest Sins of Knowledge Management. ***California Management Review***. 40(3): 265-276.

Felin, T., & Zenger, T.R. Closed or open innovation? Problem solving and the governance choice. ***Research Policy***. 43: 914-925.

Fershtman, C., & Gandal, N. 2004. The determinants of output per contributor in open source projects: An empirical examination. Working Paper. Tel Aviv University. http://ssrn.com/abstract=515282

Fey, C.F., & Birkinshaw, J. 2005. External sources of knowledge, governance mode, and R&D performance. ***Journal of Management***. 31(4): 597-621.

Fischer, M., Pinzger, M., & Gall, H. 2003. Populating a release history database from version control and bug tracking systems. ***Proceedings of the International Conference on Software Maintenance***: 23-32. Amsterdam, The Netherlands.

Fitzgerald, B. 2004. A critical look at open source. ***IEEE Computer***. 37(7): 92-94.

Fitzgerald, B. 2006. The transformation of open source software. ***MIS Quarterly***. 30(3): 587-598.

Flesch, R.A. 1948. A new readability yardstick. ***Journal of Applied Psychology***. 32(3): 221-233.

Foss, N.J., Husted, K., & Michailova, S. 2010. Governing knowledge sharing in organisations: Levels of analysis, governance mechanisms, and research directions. ***Journal of Management Studies***. 47(3): 455-482.

Foss, N.J., Lyngsie, J., & Zahra, S.A. 2013. The role of external knowledge sources and organizational design in the process of opportunity exploitation. ***Strategic Management Journal***. 34: 1453-1471.

Fox, J. 2008. **Applied Regression Analysis and Generalized Linear Models**. 2nd Edition. Sage: Thousand Oaks, CA, USA.

Fox, J., & Weisberg, S. 2011. **An R Companion to Applied Regression**. 2nd Edition. Sage: Thousand Oaks, CA, USA.

Francalanci, C., & Merlo, F. 2008. Empirical analysis of the bug fixing process in open source projects. In: Russo, B., Damiani, E., Hissam, C., Lundell, B., & Succi, G. (Eds.). 2008. **Open source Development, Communities and Quality**. IFIP International Federation for Information Processing, Volume 275. Springer: Boston, MA, USA.

Franke, N., & von Hippel, E. 2003. Satisfying heterogeneous user needs via innovation toolkits: The case of Apache security software. ***Research Policy***. 32(7): 1199-1215.

Frees, E.W. 2004. **Longitudinal and Panel Data**. Analysis and Application in the Social Sciences. Cambridge University Press: Cambridge, UK.

Gambardella, A., & Panico, C. 2014. Research Policy. On the management of open innovation. ***Research Policy***. 43: 903-913.

Garrett Jr., R.P., & Covin, J.G. 2015. Internal Corporate Venture Operations Independence and Performance: A Knowledge-Based Perspective. ***Entrepreneurship Theory and Practice***. 39(4): 763-790.

Garud, R., & Kumaraswamy, A. 2005. Vicious and Virtuous Circles in the Management of Knowledge: The Case of Infosys Technologies. ***MIS Quarterly***. 29(1): 9-33.

Gawer, A., & Cusumano, M.A. 2008. How companies become platform leaders. ***MIT Sloan Management Review***. 49(2): 28-35.

George, G., Haas, M.R., & Pentland, A. 2014. Big data and management. ***Academy of Management Journal***. 57(2): 321-326.

Gheorghe, G. 2012. Business Process Reengineering, a Crisis Solution or a Necessity. ***Annals of "Dunarea de Jos" University - Fascicle I. Economics and Applied Informatics***. 18(2): 47-52.

Giger, E., Pinzger, M., & Gall, H. 2010. Predicting the fix time of bugs. ***Proceedings of RSSE 2010***: 52-56. Cape Town, South Africa.

Girard, N. 2015. Knowledge at the boundary between science and society: A review of the use of farmers' knowledge in agricultural development. ***Journal of Knowledge Management***. 19(5): 949-967.

Goldman, R., & Gabriel, R.P. 2005. **Innovation happens elsewhere: Open source as a business strategy.** Morgan Kaufmann: Burlington, MA, USA.

Goodman, R.S., & Kruger, E.J. 1988. Data Dredging or Legitimate Research Method? Historiography and Its Potential for Management Research. ***Academy of Management Review***. 13(2): 315-325.

Grand, S., von Krogh, G., Leonard, D., & Swap, W. 2004. Resource allocation beyond firm boundaries: A multi-level model for open source innovation. ***Long Range Planning***. 37(6): 591-610.

Grant, R.M. 1996a. Toward a knowledge-based theory of the firm. ***Strategic Management Journal***. 17: 109-122.

Grant, R.M. 1996b. Prospering in Dynamically-Competitive Environments: Organizational Capability as Knowledge Integration. ***Organization Science***. 7(4): 375-387.

Grant, R.M., & Baden-Fuller, C. 2004. A Knowledge Accessing Theory of Strategic Alliances. ***Journal of Management Studies***. 41(1): 61-84.

Grün, B., Kosmidis, I., & Zeileis, A. 2012. Extended Beta Regression in R: Shaken, Stirred, Mixed, and Partitioned. ***Journal of Statistical Software***. 48(11): 1-25.

Gulati, R., Nohria, N., & Zaheer, A. 2000. Strategic Networks. ***Strategic Management Journal***. 21: 203-215.

Gulati, R., Puranam, P., & Tushman, M. 2012. Meta-organisation design: Rethinking design in interorganizational and community contexts. ***Strategic Management Journal***. 33: 571-586.

Guo, P.J., Zimmermann, T., Nagappan, N., & Murphy, B. 2010. Characterizing and predicting which bugs get fixed: An empirical study of Microsoft Windows. ***Proceedings of ICSE 2010***: 495-504. Cape Town, South Africa.

Guo, P.J., Zimmermann, T., Nagappan, N., & Murphy, B. 2011. "Not my bug!" and other reasons for software bug report reassignments. ***Proceedings of CSCW 2012***: 395-404. Hangzhou, China.

Gupta, A.K., & Govindarajan, V. 1991. Knowledge Flows and the Structure of Control Within Multinational Corporations. ***Academy of Management Review***. 16(4): 768-792.

Gupta, A.K., & Govindarajan, V. 2000. Knowledge flows within multinational corporations. ***Strategic Management Journal***. 21(4): 473-496.

Haas, M.R. 2006. Knowledge Gathering, Team Capabilities, and Project Performance in Challenging Work Environments. ***Management Science***. 52(8): 1170-1184.

Haefliger, S., von Krogh, G., & Spaeth, S. 2008. Code Reuse in Open Source Software. ***Management Science***. 54(1): 180-193.

Hamel, G., Doz, Y.L., & Prahalad, C.K. 1989. Collaborate with Your Competitors—and Win. ***Harvard Business Review***. January-February: 133-139.

Hart, K., & Sarma, A. 2014. Perceptions of answer quality in an online technical question and answer forum. ***Proceedings of the 7th International Workshop on Cooperative and Human Aspects of Software Engineering***: 103-106. Hyderabad, India.

Hassard, J., & Kelemen, M. 2002. Production and Consumption in Organizational Knowledge: The Case of the 'Paradigms Debate'. ***Organization***. 9(2): 331-355.

Hastie, T.J., & Pregibon, D. 1992. Chapter 6: Generalized linear models. In Chambers, J.M., & Hastie, T.J. Eds. **Statistical Models in S**. Wadsworth & Brooks/Cole: Monterey, CA, USA.

Henkel, J, Schöberl, S., & Alexy, O. 2014. The emergence of openness: How and why firms adopt selective revealing in open innovation. ***Research Policy***. 43: 879-890.

Henkel, J., & von Hippel, E. 2004. Welfare implications of user innovation. ***Journal of Technology Transfer***. 30(1-2): 73-87.

Herraiz, I., German, D.M., Gonzalez-Barahona, J.M., & Robles, G. 2008. Towards a simplification of the bug report form in Eclipse. ***Proceedings of the 5th IEEE Working Conference on Mining Software Repositories***: 145-148. Leipzig, Germany.

Hertel, G., Niedner, S., & Herrmann, S. 2003. Motivations of software developers in open source projects: an internet-based survey of contributors to the Linux kernel. ***Research Policy***. 32: 1159-1177.

Herzig, K., Just, S., & Zeller, A. 2013. It is not a bug, it is a feature: how misclassification impacts bug prediction. ***Proceedings of the 2013 International Conference on Software Engineering***: 392-401. San Francisco, CA, USA.

Hlavac, M. 2015. **Stargazer: Well-Formatted Regression and Summary Statistics Tables**. R package version 5.2. http://CRAN.R-project.org/package=stargazer

Ho, D., Lorenz, J., Dovgan, V., Fleming, K., Fowler, T., Cuvier, C., Legault, J., Brotherstone, D., Gruël, J., Boyer, F-R., Jonsson, A., Radić, I., Gérard, D., & contributors. 2017. **Notepad++**. https://notepad-plus-plus.org/

Hoang, H., & Rothaermel, F.T. 2010. Leveraging internal and external experience: exploration, exploitation, and R&D project performance. ***Strategic Management Journal***. 31(7): 734-758.

Hooimeijer, P., & Weimer, W. 2007. Modeling bug report quality. ***Proceedings of ASE 2007***: 34-43. Atlanta, Georgia, USA.

Hornik, K., Mair, P., Rauch, J., Geiger, W., Buchta, C., & Feinerer, I. 2013. The textcat package for n-gram based text categorization in R. ***Journal of Statistical Software***. 52(6): 1-17.

Hosseini, H., Nguyen, R., & Godfrey, M.W. 2012. A Market-Based Bug Allocation Mechanism Using Predictive Bug Lifetimes. ***Proceedings of the 16th European Conference on Software Maintenance and Reengineering***: 149-158. Szeged, Hungary.

Hothorn, T., Zeileis, A., Farebrother, R.W., Cummins, C., Millo, G., & Mitchell, D. 2017. **Package 'lmtest': Testing Linear Regression Models**. https://cran.r-project.org/package=lmtest

Huggins, R., Johnston, A., & Thompson, P. 2012. Network Capital, Social Capital and Knowledge Flow: How the Nature of inter-organizational Networks Impacts on Innovation. ***Industry and innovation***. 19(3): 203-232.

Huizingh, E.K.R.E. 2011. Open innovation: State of the art and future perspectives. ***Technovation***. 31(1): 2-9.

Huntley, C.L. 2003. Organisational learning in open-source software projects: An analysis of debugging data. ***IEEE Transactions on Engineering Management***. 50(4): 485-493.

Husson, F., Lê, S., & Pagès, J. 2010. **Exploratory Multivariate Analysis by Example Using R**. CRC Press: Boca Raton, FL, USA.

Hyndman, R.J., & Fan, Y. 1996. Sample quantiles in statistical packages. ***American Statistician***. 50: 361-365.

Inkpen, A.C. 1998. Learning and knowledge acquisition through international strategic alliances. ***Academy of Management Perspectives*** 12(4): 69-80.

Inkpen, A.C. 2000. Learning Through Joint Ventures: A Framework of Knowledge Acquisition. ***Journal of Management Studies***. 37(7): 1019-1044.

Inkpen, A.C., & Dinur, A. 1998. Knowledge Management Processes and International Joint Ventures. ***Organization Science***. 9(4): 454-468.

Ireland, R.D., Hitt, M.A., & Vaidyanath, D. 2002. Alliance Management as a Source of Competitive Advantage. ***Journal of Management***. 28(3): 413-446.

Jackman, S. 2015. **pscl: Classes and Methods for R Developed in the Political Science Computational Laboratory**. Stanford University, Department of Political Science. Sanford, CA, USA. http://pscl.stanford.edu/

Jansen, J.J.P., van den Bosch, F.A.J., & Volberda, H.W. 2005. Managing Potential and Realized Absorptive Capacity: How do Organizational Antecedents Matter? ***Academy of Management Journal***. 48(6): 999-1015.

Jansen, J.J.P., Van Den Bosch, F.A.J., & Volberda, H.W. 2006. Exploratory Innovation, Exploitative Innovation, and Performance: Effects of Organizational Antecedents and Environmental Moderators. ***Management Science***. 52(11): 1661-1674.

Jeppesen, L.B., & Frederiksen, L. 2006. Why Do Users Contribute to Firm-Hosted User Communities? The Case of Computer-Controlled Music Instruments. ***Organization Science***. 17(1): 45-63.

Jeppesen, L.B., & Laursen, K. 2009. The role of lead users in knowledge sharing. ***Research Policy***. 38(10): 1582-1589.

Joanes, D.N., & Gill, C.A. 1998. Comparing measures of sample skewness and kurtosis. ***The Statistician***. 47: 183-189.

Johansson, C., Frostevarg, J., Kaplan, A.F.H., Bertoni, M., & Chirumalla, K. 2012. Enhancing intra-cognitive communication between engineering designers and operators: A case study in the laser welding industry. ***Proceedings of the 2012 IEEE 3rd International Conference on Cognitive Infocommunications (CogInfoCom)***. Kosice, Slovakia.

Kanevsky, V., & Housel, T. 1998. The learning-knowledge-value cycle. In: von Krogh, G., Roos, J., & Kleine, D. Eds. **Knowing In Firms**. Sage: Thousand Oaks, CA, USA.

Kärreman, D. 2009. The Power of Knowledge: Learning from 'Learning by Knowledge-Intensive Firms'. ***Journal of Management Studies***. 47(7): 1405-1416.

Kessler, E.H., & Chakrabarti, A.K. 1996. Innovation Speed: A Conceptual Model of Context, Antecedents, and Outcomes. ***Academy of Management Review***. 21(4): 1143-1191.

Khomh, F., Dhaliwal, T., Zou, Y., & Adams, B. 2012. Do faster releases improve software quality?: An empirical case study of Mozilla Firefox. ***Proceedings of the 9th IEEE Working Conference on Mining Software Repositories***: 179-188. Zurich, Switzerland.

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

Kincaid, J.P., Aagard, J.A., O'Hara, J.W., & Cottrell, L.K. 1981. Computer readability editing system. ***IEEE Transactions on Professional Communication***. PC-24(1): 38-42.

Ko, A.J., & Chilana, P.K. 2010. How power users help and hinder open bug reporting. ***Proceedings of chi 2010***: 1665-1674. Atlanta, Georgia, USA.

Ko, A.J., & Chilana, P.K. 2011. Design, discussion, and dissent in open bug reports. ***Proceedings of the 2011 iConference***: 106-113. Seattle, Washington, USA.

Kogut, B. 2000. The network as knowledge: Generative rules and the emergence of structure. ***Strategic Management Journal***. 23: 405-425.

Kogut, B., & Zander, B. 1992. Knowledge of the firm, combinative capabilities, and the replication of technology. ***Organization Science***. 3(3): 383-397.

Kogut, B., & Zander, B. 1993. Knowledge of the Firm and the Evolutionary Theory of the Multinational Corporation. ***Journal of International Business Studies***. 24(4): 625-645.

Kogut, B., & Zander, B. 1996. What firms do? Coordination, identity, and learning. ***Organization Science***. 7(5): 502-518.

Koponen, T. 2006. Lifecycle of defects in open source software projects. In: Damiani, E., Fitzgerald, B., Scacchi, W., Scotto, M., & Succi, G. IFIP International Federation for Information Processing, Volume 203. **Open Source Systems**. 195-200. Springer: Boston, MA, USA.

Kuan, J. 2001. **Open source software as consumer integration into production**. Working paper. Stanford Institute for Economic Policy Research (SIEPR). http://ssrn.com/abstract=259648

Kuan, J. 2004. **Is open source software “better” than closed source software? Using bug-fix rates to compare software quality**. Working paper. Stanford Institute for Economic Policy Research (SIEPR). http://isapapers.pitt.edu/

Kuhn, T. 1970. **The structure of scientific revolutions**. 2nd Edition. University of Chicago Press: Chicago, IL, USA.

Kuk, G. 2006. Strategic interaction and knowledge sharing in the KDE developer mailing list. ***Management Science***. 52(7): 1031-1042.

Kulkarni, U.R., Ravindran, S., & Freeze, R. 2006. A Knowledge Management Success Model: Theoretical Development and Empirical Validation. ***Journal of Management of Information Systems***. 23(3): 309-347.

Kullback, S., & Leibler, R.A. 1951. On information and sufficiency. ***The annals of mathematical statistics***. 22: 79-86.

Lado, A.A., Boyd, N.G., & Hanlon, S.C. 1997. Competition, cooperation, and the search for economic rents: A syncretic model. ***Academy of Management Review***. 22(1): 110-141.

Lavie, D. 2006. The competitive advantage of interconnected firms: An extension of the resource-based view. ***Academy of Management Review***. 31(3): 638-658.

Lakhani, K., Lifshitz-Assaf, H., & Tushman, M. 2012. Open innovation and organisational boundaries: The impact of task decomposition and knowledge distribution on the locus of innovation. Working Paper No 12-57. Harvard Business School Technology & Operations Management Unit: Harvard Business School, MA, USA. Available at SSRN: http://ssrn.com/abstract=1980118

Lakhani, K.R., & von Hippel, E. 2003. How open source software works: "Free" user-to-user assistance. ***Research Policy***. 32: 923-943.

Lam, A. 1997. Embedded Firms, Embedded Knowledge: Problems of Collaboration and Knowledge Transfer in Global Cooperative Ventures. ***Organization Studies***. 18(6): 973-996.

Lam, A. 2000. Tacit Knowledge, Organizational Learning and Societal Institutions: An Integrated Framework. ***Organization Studies***. 21(3): 487-513.

Lamkanfi, A., & Demeyer, S. 2013. Predicting reassignments of bug reports: An exploratory investigation. ***Proceedings of the 17th European Conference on Software Maintenance and Reengineering***. 327-330. Genova, Italy.

Lane, P.J., & Lubatkin, M. 1998. Relative absorptive capacity and interorganizational learning. ***Strategic Management Journal***. 19(5): 461-477.

Lane, P.J., Koka, B.R., & Pathak, S. 2006. The Reification of Absorptive Capacity: A Critical Review and Rejuvenation of the Construct. ***Academy of Management Review*** 31(4): 833-863.

Lane, P.J., Salk, J.E., & Lyles, M.A. 2001. Absorptive capacity, learning, and performance in international joint ventures. ***Strategic Management Journal***. 22(12): 1139-1161.

Lanubile, F., Ebert, C., Prikladnicki, R., & Vizcaino, A. 2010. Collaboration Tools for Global Software Engineering. ***IEEE Software***. 27(2): 52-55.

Laursen, K., & Salter, A. 2006. Open for innovation: The role of openness in explaining innovation performance among U.K. manufacturing firms. ***Strategic Management Journal***. 27(2): 131-150.

LaValle, S., Lesser, E., Shockley, R., Hopkins, M.S., & Kruschwitz, N. 2011. Big Data, Analytics and the Path From Insights to Value. ***MIT Sloan Management Review***. 52(2): 21-32.

Lefcheck, J.S. 2015. piecewiseSEM: Piecewise structural equation modeling in R for ecology, evolution, and systematics. ***Methods in Ecology and Evolution***. 7(5): 573-579. DOI: 10.1111/2041-210X.12512.

Leonard-Barton, D. 1992. Core capabilities and core rigidities: A paradox in managing new product development. ***Strategic Management Journal***. 13(S1): 111-125.

Lerner., J., & Tirole, J. 2001. The open source movement: Key research questions. ***European Economic Review***. 45(4-6): 819-826.

Lerner, J., & Tirole, J. 2002. Some simple economics of open source. ***Journal of Industrial Economics***. 50(2): 197-234.

Lerner, J., Pathak, P.A., & Tirole, J. 2006. The Dynamics of Open-Source Contributors. ***American Economic Review***. 96(2): 114-118.

Levi, M., Kleindorfer, P.R., & Wu, D.J. 2003. Codifiability, Relationship-Specific Information Technology Investment, and Optimal Contracting. ***Journal of Management Information Systems***. 20(2): 77-98.

Levine, S.S. Prietula, M.J. 2012. Open source, open innovation, open communities: What drives the performance of "open"? In Toombs, L. (Ed.). ***Academy of Management Best Paper Proceedings***. Boston, MA, USA.

Lewis, T.G. 1995. Where is client/server software headed? ***IEEE Computer***. 38(4): 49-55.

Lewis, K. 2004. Knowledge and Performance in Knowledge-Worker Teams: A Longitudinal Study of Transactive Memory Systems. ***Management Science***. 50(11): 1519-1533.

Lewis, C., Lin, Z., Sadowski, C., Zhu, X., Ou, R., & Whitehead Jr., E.J. 2013. Does bug prediction support human developers? Findings from a Google case study. ***Proceedings of the 2013 International Conference on Software Engineering***: 372-381. San Francisco, CA, USA.

Li, Z., Tan, L., Wang, X., Lu, S., Zhou, Y., & Zhai, C. 2006. Have things changed now? An empirical study of bug characteristics in modern open source software. ***Proceedings of ASID 2006***: 25-33. San Jose, California, USA.

Lichtenthaler, U. 2009. Absorptive Capacity, Environmental Turbulence, and the Complementarity of Organizational Learning Processes. ***Academy of Management Journal***. 52(4): 822-846.

Long, J.S. 1997. **Regression Models for Categorical and Limited Dependent Variables**. 104-106. Sage: Thousand Oaks, CA, USA.

Lotufo, R., Passos, L., & Czarnecki, K. 2012. Towards improving bug tracking systems with game mechanics. ***Proceedings of the 9th IEEE Working Conference on Mining Software Repositories***: 2-11. Zurich, Switzerland.

MacAulay, M. 2010a. Understanding the user: The marginalized open source contributor. ***Proceedings of the 2010 Cambridge University Interdisciplinary Conference***. University of Cambridge. Cambridge, U.K.

MacAulay, M. 2010b. What's the Value of an Eyeball? Passive Participation in Open Source Ecosystems. ***Technology Innovation Management Review***. January: 19-24.

MacAulay, M. 2013. Why do firms use open source strategies?: An uncertainty reduction theory. ***Proceedings of the 2013 Academy of Management Annual Conference***. Orlando, FL, USA. doi: 10.5465/AMBPP.2013.12702abstract

MacAulay, M. 2017. The Open Source Strategies of Canadian Organizations. ***Proceedings of the 45th Administrative Sciences Association of Canada Annual Conference***. HEC Montreal. Montreal, QC, Canada.

Mahr, D., & Lievens, A. 2012. Virtual lead user communities: Drivers of knowledge creation for innovation. ***Research Policy***. 41(1): 167-177.

Maier, R., & Remus, U. 2003. Implementing process-oriented knowledge management strategies. ***Journal of Knowledge Management***. 7(4): 62-74.

Majchrzak, A., More, P.H.B., & Faraj, S. 2011. Transcending Knowledge Differences in Cross-Functional Teams. ***Organization Science***. 23(4): 951-970.

Makadok, R., & Coff, R. 2009. Both market and hierarchy: An incentive-system theory of hybrid governance forms. ***Academy of Management Review***. 34(2): 297-319.

Mani, S., Nagar, S., Mukherjee, D., Narayanam, R., Sinha, V.S., & Nanavati, A.A. 2013. Bug resolution catalysts: Identifying essential non-committers from bug repositories. ***Proceedings of the 10th IEEE Working Conference on Mining Software Repositories***: 193-202. San Francisco, CA, USA.

March, J.G. 1991. Exploration and Exploitation in Organizational Learning. ***Organization Science***. 2(1): 71-87.

Macpherson, A., & Holt, R. 2007. Knowledge, learning, and small firm growth: A systematic review of the evidence. ***Research Policy***. 36(2): 172-192.

Malhotra, A., Gosain, S., & El Sawy, O.A. 2005. Absorptive Capacity Configurations in Supply Chains: Gearing for Partner-Enabled Market Knowledge Creation. ***MIS Quarterly***. 29(1): 145-187.

Marks, L., Zou, Y., & Hassan, A.E. 2011. Studying the fix-time for bugs in large open source projects. ***Proceedings of the 7th International Conference on Predictive Models in Software Engineering***. ACM: New York, NY, USA.

Marlow, J., Dabbish, L., & Herbsleb, J. 2013. Impression formation in online peer production: activity traces and personal profiles in GitHub. ***Proceedings of the 2013 conference on computer supported cooperative work***: 117-128. San Antonio, TX, USA.

Martin, W.J. 2004. Demonstrating knowledge value: A broader perspective on metrics. ***Journal of Intellectual Capital***. 5(1): 77-91.

Masmoudi, H., Peigné, Q., de Loupy, C., den Besten, M., & Dalle, J-M. 2012. "Peeling the onion": Analyzing core and periphery in the Firefox community with text-mining methods. Working paper. Available at: http://www.academia.edu/3138074/

Massingham, P. 2016. Knowledge Accounts. ***Long Range Planning***. 49(3): 409-425.

Matter, D., Kuhn, A., & Nierstrasz, O. 2009. Assigning bug reports using a vocabulary-based expertise model of developers. ***Proceedings of the 6th IEEE International Working Conference on Mining Software Repositories***. Vancouver, BC, Canada.

Matusik, S.F. 2002. An empirical investigation of firm public and private knowledge. ***Strategic Management Journal***. 23(5): 457-467.

Matusik, S.F., & Heeley, M.B. 2005. Absorptive Capacity in the Software Industry: Identifying Dimensions That Affect Knowledge and Knowledge Creation Activities. ***Journal of Management***. 31(4): 549-572.

Matusik, S.F., & Hill, C.W.L. 1998. The Utilization of Contingent Work, Knowledge Creation, and Competitive Advantage. ***Academy of Management Review***. 23(4): 680-697.

McAfee, A., & Brynjolfsson, E. 2012. Big Data: The Management Revolution. ***Harvard Business Review***. October: 1-9.

McEvily, S.K., & Chakravarthy, B. 2002. The persistence of knowledge-based advantage: An empirical test for product performance and technological knowledge. ***Strategic Management Journal***. 23(4): 285-305.

McFadden, D. 1973. Conditional logit analysis of qualitative choice behavior. In Zarembka, P. Ed. **Frontiers in Econometrics**. 105-142. Academic Press: New York, NY, USA.

MERIT. 2006. **Study on the economic impact of open source software on innovation and the competitiveness of the information and communication technologies sector in the EU**. Commissioned Report for the European Commission. Available at: http://ec.europa.eu/enterprise/sectors/ict/files/2006-11-20-flossimpact_en.pdf

Miller, K.D., Zhao, M., & Calantone, R.J. 2006. Adding Interpersonal Learning and Tacit Knowledge to March's Exploration-Exploitation Model. ***Academy of Management Journal***. 49(4): 709-722.7uy

Mockus, A., Fielding, R., & Herbsleb, J.D. 2002. Two case studies of open source software development: Apache and Mozilla. ***ACM Transactions on Software Engineering and Methodology***. 11(3): 309-346.

Moore, D.F. 2016. **Applied Survival Analysis Using R**. Springer: Cham, Switzerland.

Mowery, D.C., Oxley, J.E., & Silverman, B.S. 1996. Strategic Alliances and Interfirm Knowledge Transfer. ***Strategic Management Journal***. 17(Winter Special Issue): 77-91.

Mozilla. 2017a. **Mozilla Firefox: Development Process**. Available at: https://mozilla.github.io/process-releases/draft/development_overview/

Mozilla. 2017b. **The Mozilla Manifesto**. Available at: https://www.mozilla.org/en-US/about/manifesto/

Nemati, H.R., Steiger, D.M., Iyer, L.S., & Herschel, R.T. 2002. Knowledge warehouse: An architectural integration of knowledge management, decision support, artificial intelligence and data warehousing. ***Decision Support Systems***. 33(2): 143-161.

Nguyen, T.H.D., Adams, B., & Hassan, A.E. 2010. A case study of bias in bug-fix datasets. ***Proceedings of the 17th Working Conference on Reverse Engineering (WCRE 2010):*** 259-268. Beverly, MA, USA.

Nickerson, J.A., & Zenger, T.R. 2004. A Knowledge-Based Theory of the Firm – The Problem-Solving Perspective. ***Organization Science***. 15(6): 617-632.

Nieto, M., & Quevedo, P. 2005. Absorptive capacity, technological opportunity, knowledge spillovers, and innovative effort. ***Technovation***. 25: 1141-1157.

Nonaka, I. 1994. Dynamic theory of organisational knowledge creation. ***Organization Science***. 5(1): 14-37.

Nonaka, I., & Konno, N. 1998. The concept of "ba": Building a foundation for knowledge creation. ***California Management Review***. 40(3): 40-55.

Nonaka, I., Toyama, R., & Nagata, A. 2000. A firm as a knowledge creating entity: A new perspective on the theory of the firm. ***Industrial and Corporate Change***. 9: 1-20.

Nonaka, I., Umemoto, K., & Senoo, D. 1996. From information processing to knowledge creation: A paradigm shift in business management. ***Technology in Society***. 18(2): 203-218.

Nonaka, I., & von Krogh, G. 2009. Tacit knowledge and knowledge conversion: Controversy and advancement in organizational knowledge creation theory. ***Organization Science***. 20 (3): 635-652.

North Bridge. 2016. **2016 Future of Open Source Study**. Technical report. Available at https://www.slideshare.net/North_Bridge/2016-future-of-open-source-study-61431845

O'Mahony, S. 2007. The governance of open source initiatives: What does it mean to be community managed? ***Journal of Management & Governance***. 11(2): 139-150.

O'Mahony, S., & Bechky, B.A. 2008. Boundary organisations: Enabling collaboration among unexpected allies. ***Administrative Science Quarterly***. 53: 422-459.

O'Mahony, S., & Ferraro, F. 2007. The emergence of governance in an open source community. ***Academy of Management Journal***. 50(5): 1079-1106.

Panjer, L.D. 2007. Predicting Eclipse bug lifetimes. ***Proceedings of the 4th IEEE Working Conference on Mining Software Repositories***. Minneapolis, MN, USA.

Pereira, A.M., Gonçalves, R.Q., Von Wangenheim, C.G., & Buglione, L. 2013. Comparison of open source tools for project management. ***International Journal of Software Engineering and Knowledge Engineering***. 23(2): 189-209.

Pérez-Bustamante, G. 1999. Knowledge management in agile innovative organisations. ***Journal of Knowledge Management***. 3(1): 6-17.

Peteraf, M.A. 1993. The cornerstones of competitive advantage: A resource-based view. ***Strategic Management Journal***. 14: 179-191.

Peteraf, M.A., & Barney, J.B. 2003. Unraveling the resource-based tangle. ***Managerial and Decision Economics***. 24(4): 309-323.

Pisano, G.P. 1994. Knowledge, Integration, and the Locus of Learning: An Empirical Analysis of Process Development. ***Strategic Management Journal***. 15(S1): 85-100.

Popper, C. 1957. **The poverty of historicism**. Routledge: New York, NY, USA.

Powell, W.W. 1990. Neither market nor hierarchy: Network forms of organisation. ***Research in Organisational Behaviour***. 12: 295-336.

Powell, A. 2012. Democratizing production through open source knowledge: from open software to open hardware. ***Media, Culture & Society***. 34(6): 691-708.

Prahalad, C.K., & Bettis, R.A. 1986. The dominant logic: A new linkage between diversity and performance. ***Strategic Management Journal***. 7(6): 485-501.

Priem, R. L. & Butler, J. E. 2001. Is the resource-based "view" a useful perspective for strategic management research? ***Academy of Management Review***. 26(1): 22-40.

Quintas, P., Lefrere, P., & Jones, G. 1997. Knowledge management: A strategic agenda. ***Long Range Planning***. 30(3): 322, 385-391.

R Foundation. 2017. **The R project for statistical computing**. Available at: https://www.r-project.org

Rahman, M.M., Ruhe, G., & Zimmermann, T. 2009. Optimized assignment of developers for fixing bugs – An initial evaluation for eclipse projects. ***Proceedings of the 3rd International Symposium on Empirical Software Engineering and Measurement***: 439-442. Lake Buena Vista, FL, USA.

Raymond, E.S. 1999a. **The cathedral and the bazaar**. O'Reilly Media Inc.: Sebastopol, CA, USA.

Raymond, E.S. 1999b. The cathedral and the bazaar. **Knowledge, Technology, & Policy**. 12(3): 23-49.

Reagans, R., & McEvily, B. 2003. Network Structure and Knowledge Transfer: The Effects of Cohesion and Range. ***Administrative Science Quarterly***. 48(2): 240-267.

Reagle Jr., J. 2007. Bug Tracking Systems as Public Spheres. ***Techné: Research in Philosophy and Technology***. 11(1): 32-41.

Rhemtulla, M., Brosseau-Liard, P.E., & Savalei, V. 2012. When can categorical variables be treated as continuous? A comparison of robust continuous and categorical SEM estimation methods under suboptimal conditions. ***Psychological Methods***. 17(3): 354-373.

Roberts, N., Galluch, P.S., Dinger, M., & Grover, V. 2012. Absorptive capacity and information systems research: Review, synthesis, and directions for future research. ***MIS Quarterly***. 36(2): 625-648.

Roberts, S., & Pashler, H. 2000. How persuasive is a good fit? A comment on theory testing. ***Psychological Review***. 107(2): 358-367.

Rocha, H, de Oliveira, G., Valente, M. T., & Marques-Neto, H. 2016. Characterizing Bug Workflows in Mozilla Firefox. ***Proceedings of the 30th Brazilian Symposium on Software Engineering***: 43-52. Maringá, Brazil.

Rodan, S., & Galunic, C. 2004. More than network structure: How knowledge heterogeneity influences managerial performance and innovativeness. ***Strategic Management Journal***. 25(6): 541-562.

Rousseau, D.M., & House, R.J. 1994. Meso Organizational Behavior: Avoiding Three Fundamental Biases. In Cooper, C.L., & Rousseau, D.M. Eds. **Trends in Organizational Behavior**. Volume 1. 13-30. John Wiley & Sons: Oxford, UK.

Ruffin, C., & Ebert, C. 2004. Using open source software in product development: a primer. ***IEEE Software***. 21(1): 82-86.

Rughiniş, R., & Matei, S. 2013. Digital Badges: Signposts and Claims of Achievement. ***Proceedings of the International Conference on Human-Computer Interaction***: 84-88. Las Vegas, NV, USA.

Saha, R.K., Lawall, J., Khurshid, S., & Perry, D.E. 2015. Are these bugs really "normal"? ***Proceedings of the 12th Working Conference on Mining Software Repositories***: 258-268. Florence, Italy.

Sakamoto, Y., Ishiguro, M., & Kitagawa, G. 1986. **Akaike Information Criterion Statistics**. D. Reidel Publishing Company: Dordrecht, Netherlands.

Salter, A., Criscuolo, P., & Ter Wal, A.L.J. 2014. Coping with Open Innovation: Responding to the Challenges of External Engagement in R&D. ***California Management Review***. 56(2): 77-94.

Sanchez, R., & Mahoney, J.T. 1996. Modularity, flexibility, and knowledge management in product and organization design. ***Strategic Management Journal***. 17(S2): 63-76.

Sandusky, R.J., Gasser, L., & Ripoche, G. 2004. Bug report networks: Varieties, strategies, and impacts in a F/OSS development community. ***Proceedings of the 26th International Conference on Software Engineering***: 80-84. Edinburgh, UK.

Sawilowsky, S. 2009. New effect size rules of thumb. ***Journal of Modern Applied Statistical Methods***. 8(2): 467-474.

Schmidt, T. 2010. Absorptive capacity—one size fits all? A firm-level analysis of absorptive capacity for different kinds of knowledge. ***Managerial and Decision Economics***. 31(1): 1-18.

Schulz, M. 2001. The Uncertain Relevance of Newness: Organizational Learning and Knowledge Flows. ***Academy of Management Journal***. 44(4): 661-681.

Schulz, M. 2003. Pathways of Relevance: Exploring Inflows of Knowledge into Subunits of Multinational Corporations. ***Organization Science***. 14(4): 440-459.

Schulz, M., & Jobe, L.A. 2001. Codification and tacitness as knowledge management strategies: an empirical exploration. ***The Journal of High Technology Management Research***. 12(1): 139-165.

Serenko, A., Bontis, N., & Hardie, T. 2007. Organisational size and knowledge flow: a proposed theoretical link. ***Journal of Intellectual Capital***. 8(4): 610-627.

Serrano, N., & Ciordia, I. 2005. Bugzilla, ITracker, and other bug trackers. ***IEEE Software***. 22(2): 11-13.

Shah, S.K. 2006. Motivation, governance, and the viability of hybrid forms in open source software development. ***Management Science***. 52(7): 1000-1014.

Shah, S.K., & Tripsas, M. 2007. The accidental entrepreneur: The emergent and collective process of user entrepreneurship. ***Strategic Entrepreneurship Journal***. 1(1-2): 123-140.

Sharma, S., Sugumaran, V., & Rajagopalan, B. 2002. A framework for creating hybrid-open source software communities. ***Information Systems Journal***. 12(1): 7-25.

Shihab, E., Ihara, A., Kamei, Y., Ibrahim, W.M., Ohira, M., Adams, B., Hassan, A.E., & Matsumoto, K-i. 2010. Predicting re-opened bugs: A case study on the Eclipse project. ***Proceedings of the 17th Working Conference on Reverse Engineering (WCRE 2010)***: 249-258. Beverly, MA, USA.

Shihab, E., Ihara, A., Kamei, Y., Ibrahim, W.M., Ohira, M., Adams, B., Hassan, A.E., & Matsumoto, K-i. 2013. Studying re-opened bugs in open source software. ***Empirical Software Engineering***. 18(5): 1005-1042.

Shokripour, R., Anvik, J., Kasirun, Z.M., & Zamani, S. 2013. Why so complicated? Simple term filtering and weighting for location-based bug report assignment recommendation. ***Proceedings of the 10th Working Conference on Mining Software Repositories***: 2-11. San Francisco, CA, USA.

Simas, A.B., Barreto-Souza, W., & Rocha, A.V. 2010. Improved Estimators for a General Class of Beta Regression Models. ***Computational Statistics & Data Analysis***. 54(2): 348-366.

Singh, J. 2005. Collaborative Networks as Determinants of Knowledge Diffusion Patterns. ***Management Science***. 51(5): 756-770.

Singleton Jr., R.A. & Straits, B.C. 2005. **Approaches to social research**. 4th Edition. Oxford University Press: New York, NY, USA.

Smith, E.A., & Kincaid, J.P. 1970. Derivation and validation of the automated readability index for use with technical materials. ***Human Factors***. 12(5): 457-564.

Smithson, M., & Verkuilen, J. 2006. A Better Lemon Squeezer? Maximum-Likelihood Regression with Beta-Distributed Dependent Variables. ***Psychological Methods***, 11(1): 54-71.

Söderberg, J. 2015. **Hacking Capitalism: The Free and Open Source Software Movement**. Volume 9 of Routledge Research in Information Technology and Society. Routledge: London, UK.

Soekijad, M., & Andriessen, E. 2003. Conditions for Knowledge Sharing in Competitive Alliances. ***European Management Journal***. 21(5): 578-587.

Sorenson, O., Rivkin, J.W., & Fleming, L. 2006. Complexity, networks and knowledge flow. ***Research Policy***. 35(7): 994-1017.

Sourceforge. 2013. **About Sourceforge.net**. Available at: http://sourceforge.net/about

Spaeth, S., von Krogh, G., & He, F. 2014. Perceived Firm Attributes and Intrinsic Motivation in Sponsored Open Source Software Projects. ***Information Systems Research***. 26(1): 224-237.

Spender, J-C. 1996. Making knowledge the basis of a dynamic theory of the firm. ***Strategic Management Journal***. 17(Winter Special Issue): 45-62.

Spithoven, A., Clarysse, B., & Knockaert, M. 2011. Building absorptive capacity to organise inbound open innovation in traditional industries. ***Technovation***. 31(1): 10-21.

Squire, M. 2014. Forge++: The Changing Landscape of FLOSS Development. ***Proceedings of the 2014 47th Hawaii International Conference on System Sciences***: 3266-3275. Waikoloa, Hawaii, USA.

Starbuck, W.H. 1992. Learning by knowledge-intensive firms. ***Journal of Management Studies***. 29(6): 713-740.

Subramaniam, M., & Youndt, M.A. 2005. The Influence of Intellectual Capital on the Types of Innovative Capabilities. ***Academy of Management Journal***. 48(3): 450-463.

Swan, J., Newell, S., Scarbrough, H., & Hislop, D. 1999. Knowledge management and innovation: Networks and networking. ***Journal of Knowledge Management***. 3(4): 262-275.

Szulanski, G. 1996. Exploring internal stickiness: Impediments to the transfer of best practice within the firm. ***Strategic Management Journal***, 17(Winter Special): 27-43.

Tabachnick, B.G., & Fidell, L.S. 2007. **Using Multivariate Statistics**. 5th Edition. Pearson: Toronto, ON, Canada.

Tanenbaum, A.S., & Bos, H. 2014. **Modern Operating Systems**. 4th Edition. Pearson: Upper Saddle River, NJ, USA.

Tarr, D. 2012. **Webmail Domains**. Available at: https://github.com/tarr11/Webmail-Domains

Teece, D.J., Pisano, G., & Shuen, A. 1997. Dynamic Capabilities and Strategic Management. ***Strategic Management Journal***. 18(7): 509-533.

Teo, H.J., & Johri, A. 2014. Fast, functional, and fitting: Expert response dynamics and response quality in an online newcomer help forum. ***Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing***: 332: 341. Baltimore, Maryland, USA.

Thongtanunam, P., Kula, R.G., Cruz, A.E.C., Yoshida, N., Ichikawa, K., & Iida, H. 2013. Mining History of Gamification Towards Finding Expertise in Question and Answering Communities: Experience and Practice with Stack Exchange. ***Review of Socionetwork Strategies***. 7(2): 115-130.

Todorova, G., & Durisin, B. 2007. Absorptive capacity: Valuing a reconceptualization. ***Academy of Management Review***. 32(3): 774-786.

Tomas, G., & Hult, M. 2003. An Integration of Thoughts on Knowledge Management. ***Decision Sciences***. 34(2): 189-195.

Tsai, W. 2002. Social Structure of "Coopetition" Within a Multiunit Organization: Coordination, Competition, and Intraorganizational Knowledge Sharing. ***Organization Science***. 13(2): 179-190.

Turner, K.L., & Makhija, M.V. 2006. The Role of Organizational Controls In Managing Knowledge. ***Academy of Management Review***. 31(1): 197-217.

Uotila, J., Maula, M., Keil, T., & Zahra, S.A. 2009. Exploration, exploitation, and financial performance: analysis of S&P 500 corporations. ***Strategic Management Journal***. 30(2): 221-231.

Valdivia Garcia, H, & Shihab, E. 2014. Characterizing and predicting blocking bugs in open source projects. ***Proceedings of the 11th Working Conference on Mining Software Repositories***: 72-81. Hyderabad, India.

van den Berg, H.A. 2013. Three shapes of organisational knowledge. ***Journal of Knowledge Management***. 17(2): 159-174.

van den Bosch, F.A.J., Volberda, H.W., & de Boer, M. 1999. Coevolution of Firm Absorptive Capacity and Knowledge Environment: Organizational Forms and Combinative Capabilities. ***Organization Science***. 10(5): 551-568.

van Wijk, R., van den Bosch, F.A.J, & Volberda, H.W. 2011. Absorptive Capacity: Taking Stock of its Progress and Prospects. In: Easterby-Smith, M., & Lyles, M.A. Eds. **Handbook of Organizational Learning and Knowledge Management**. 2nd Edition. Wiley: Hoboken, NJ, USA.

Vasilescu, B., Serebrenik, A., Devanbu, P., & Filkov, V. 2014. How social Q&A sites are changing knowledge sharing in open source software communities. ***Proceedings of the 17th ACM conference on computer supported cooperative work & social computing***: 342-354. Baltimore, Maryland, USA.

Viseur, R. 2013. Identifying success factors for the Mozilla project. In: Petrinja, E., Succi, G., El loini., N., & Sillitti, A. 2013. **Open Source Software: Quality Verification**. IFIP Advances in Information and Communication Technology, Volume 404. 45-60. Springer: Boston, MA, USA.

Volberda, H.W., Foss, N.J., & Lyles, M.A. 2010. PERSPECTIVE—Absorbing the Concept of Absorptive Capacity: How to Realize Its Potential in the Organization Field. ***Organization Science***. 21(4): 931-951.

von Hippel, E. 1994. "Sticky information" and the locus of problem solving: Implications for innovation. ***Management Science***. 40(4): 429-439.

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

von Hippel, E. 2005. **Democratizing Innovation**. MIT Press: Cambridge, MA, USA.

von Hippel, E., & von Krogh, G. 2003. Open source software and the "private-collective" innovation model: Issues for organisation science. ***Organization Science***. 14(2): 209-223.

von Krogh, G., Spaeth, S., & Haefliger, S. 2005. Knowledge reuse in open source software: An exploratory study of 15 open source projects. ***Proceedings of the 38th Hawaii International Conference on System Sciences***. Waikoloa, Hawai'i, USA

Wagenmakers, E.J., & Farrell, S. 2004. AIC model selection using Akaike weights. ***Psychonomic Bulletin & Review***. 11: 192-196.

Wang, H.C., He, J., & Mahoney, J.T. 2009. Firm-specific knowledge resources and competitive advantage: the roles of economic- and relationship-based employee governance mechanisms. ***Strategic Management Journal***. 30(12): 1265-1285.

Wang, J., & Zhang, H. 2012. Predicting defect numbers based on defect state transition models. ***Proceedings of the ACM-IEEE International Symposium on Empirical Software Engineering and Measurement***: 191-200. Lund, Sweden.

Weiß, C., Premraj, R., Zimmermann, T., & Zeller, A. 2007. How long will it take to fix this bug? ***Proceedings of the 4th IEEE Working Conference on Mining Software Repositories***. Minneapolis, MN, USA.

Wernerfelt, B. 1984. A resource-based view of the firm. ***Strategic Management Journal***. 5(2): 171-180.

Wasserstein, R.L., & Lazar, N.A. 2016. The ASA's Statement on p-Values: Context, Process, and Purpose. ***The American Statistician***. 70(2): 129-133.

Wei, X., Chen, W., & Zhu, K. 2015. Motivating User Contributions in Online Knowledge Communities: Virtual Rewards and Reputation. ***Proceedings of 2015 48th Hawaii International conference on System Sciences***: 1530-1605. Kauai, HI, USA.

West, J. 2003. How open is open enough?: Melding proprietary and open source platform strategies. ***Research Policy***. 32(7): 1259-1285.

West, J., & Dedrick, J. 2001. Open Source Standardization: The Rise of Linux in the Network Era. ***Knowledge, Technology, & Policy***. 14(2): 88-112.

West, J., & Gallagher, S. 2006. Challenges of open innovation: the paradox of firm investment in open-source software. ***R&D Management***. 36(3): 319-331.

West, J., & O'Mahony, S. 2008. The role of participation architecture in growing sponsored open source communities. ***Industry and Innovation***. 15(2): 145-168.

West, J., Salter, A., Vanhaverbeke, W., & Chesbrough, H. 2014. Open innovation: The next decade. ***Research Policy***. 43: 805-811.

West, J., & Wood, D. 2014. Evolving an Open Ecosystem: The Rise and Fall of the Symbian Platform. In: Adner, R., Oxley, J.E., & Silverman, B.S. Eds. Collaboration and Competition in Business Ecosystems. ***Advances in Strategic Management***. 30: 27-67.

Wiig, K.M. 1997. Integrating intellectual capital and knowledge management. ***Long Range Planning***. 30(3): 399-405.

Williams, R.B.G. 1986. **Intermediate statistics for geographers and earth scientists**. Macmillan: London, UK.

Williamson, O.E. 1975. **Markets and hierarchies: Analysis and antitrust implications**. Free Press: New York, NY, USA.

Williamson, O.E. 1985. **The economic institutions of capitalism: Firms, markets, relational contracting**. Free Press: New York, NY, USA.

Williamson, O.E. 1991. Comparative economic organisation: The analysis of discrete structural alternatives. ***Administrative Science Quarterly***. 36: 269-296.

Wikipedia. 2013. ***Software Bug***. http://en.wikipedia.org/wiki/Software_bug

Winter, S.G., & Szulanski, G. 2001. Replication as strategy. ***Organization Science***. 12(6): 730-743.

Woods, D., & Guliani, G. 2005. **Open Source for the Enterprise**. O'Reilly & Associates: Sebastopol, CA, USA.

Woodside, A.G. 2016. The good practices manifesto: Overcoming bad practices pervasive in current research in business. ***Journal of Business Research***. 69(2): 365-381.

Xu, Y., & Bernard, A. 2011. Quantifying the value of knowledge within the context of product development. ***Knowledge-Based Systems***. 24(1): 166-175.

Yang, H., Phelps, C., & Steensma, H.K. 2010. Learning from what others have learned from you: The effects of knowledge spillovers on originating firms. ***Academy of Management Journal***. 53(2): 371-389.

Young, R. 1999. Giving It Away—How Red Hat Software Stumbled Across a New Economic Model and Helped Improve an Industry. In: DiBona, C., Ockman, S., & Stone, M. Eds. **Open Sources: Voices from the Open Source Revolution**. O'Reilly & Associates: Sebastopol, CA, USA.

Zahra, S.A., & Filatotchev, I. 2004. Governance of the Entrepreneurial Threshold Firm: A Knowledge-based Perspective. ***Journal of Management Studies***. 41(5): 885-897.

Zahra, S.A., & George, G. 2002. Absorptive Capacity: A Review, Reconceptualization, and Extension. ***Academy of Management Review***. 27(2): 185-203.

Zhang, F., Khomh, F., Zou, Y., & Hassan, A.E. 2012. ***Proceedings of the 19th Working Conference on Reverse Engineering (WCRE 2012)***. Kingston, ON, Canada.

Zander, U, & Kogut, B. 1995. Knowledge and the speed of the transfer and imitation of organisational capabilities: An empirical test. ***Organization Science***. 6(1): 76-92.

Zhou, B., Neamtiu, I., & Gupta, R. 2015. Experience report: How do bug characteristics differ across severity classes: A multi-platform study. ***Proceedings of the IEEE 26th International Symposium on Software Reliability Engineering***. Gaithersbury, MD, USA.

Zimmermann, T., Premraj, R., Bettenburg, N., Just, S., Schrõter, A., & Weiss, C. 2010. What makes a good bug report? ***IEEE Transactions on Software Engineering***. 36(5): 618-643.

Zimmermann, T., Nagappan, N., Guo, P.J., & Murphy, B. 2012. Characterizing and predicting which bugs get reopened. ***Proceedings of ICSE 2012***: 1074-1083. Zurich, Switzerland.

Zucker, L.G., Darby, M.R., Furner, J., Liu, R.C., & Ma, H. 2007. Minerva unbound: Knowledge stocks, knowledge flows and new knowledge production. ***Research Policy***. 36(6): 850-863.

APPENDICES

Appendix A: Operationalization code

Please see the documents located at: "http://mekki.ca/dissertation/code" (HTML formatted and syntax-highlighted source code), "http://mekki.ca/dissertation/code/raw" (plain text executable R source code), and "http://mekki.ca/dissertation/Appendix%20A.pdf" (letter page size formatted version of source code).

Appendix B: Analysis code

Please see the documents located at: "http://mekki.ca/dissertation/code" (HTML formatted and syntax-highlighted source code), "http://mekki.ca/dissertation/code/raw" (plain text executable R source code), and "http://mekki.ca/dissertation/Appendix%20B.pdf" (letter page size formatted version of source code).

Appendix C: Additional analysis details

Please see the documents located at: "http://mekki.ca/dissertation/figures" (image files of additional analysis graphs) and "http://mekki.ca/dissertation/Appendix%20C.pdf" (letter page size formatted version of additional analysis graphs).

Appendix D: Regression models

Please see the documents located at: "http://mekki.ca/dissertation/models" (HTML heteroskedasticity-corrected model fit analysis), "http://mekki.ca/dissertation/models/raw" (plain text uncorrected and ANCOVA type II model fit analyses), and "http://mekki.ca/dissertation/Appendix%20D.pdf" (letter page size formatted version of all model fit analyses).

Appendix E: Summary of results

Please see the document located at: "http://mekki.ca/dissertation/Appendix%20E.pdf".