EXAMINING THE EFFECT OF DEVOPS ADOPTION CAPABILITY ON ORGANIZATIONAL AGILITY

A Dissertation
Submitted to
the Temple University Graduate Board

In Partial Fulfillment
of the Requirements for the Degree
EXECUTIVE DOCTOR OF BUSINESS ADMINISTRATION

---

by
Iris S. Burrell
Temple University, Fox School of Business
Diploma Date May 2018

Examining Committee Members:

Dr. David Schuff, Dissertation Advisor, Management Information Systems
Dr. Detmar Straub, Management Information Systems
Dr. Richard Flanagan, Management Information Systems
Dr. Susan Mudambi, External Reader, Marketing and Supply Chain Management

ABSTRACT

The purpose of this research is to investigate the effect of DevOps adoption on organizational agility. The first chapter establishes through literature review and a pilot study, a theoretical definition of DevOps and identifies through survey data, the key DevOps adoption capabilities. The definition and key capabilities identified during the pilot study establishes the foundation through which a subsequent study is performed. The second chapter details an examination of the effects of DevOps adoption capability on organizational agility. Through capability theory, we propose that DevOps adoption capability positively affects organizational agility. We also propose that the capabilities of collaboration/communication, continuous monitoring, measurement and automation positively affect DevOps adoption capability and likewise; the capabilities of responsiveness, competency, flexibility, and quickness affect organizational agility capability. We test our model with survey data collected from 333 respondents and find that our hypotheses are supported and that DevOps adoption capability has a positive effect on organizational agility capability.

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my advisor Dr. David Schuff for his continuous support and dedication of my Doctoral study, for his patience, motivation, and knowledge this would not have been possible. I would also like to thank my dissertation committee members, Dr. Detmar Straub, Dr. Richard Flanagan, and Dr. Susan Mudambi for their guidance, support and constructive feedback during this process. Finally, I would like to thank my husband Danny, who spent many long days and late weekend nights entertaining the kids while I continued researching. Without your support, this would not have been possible. I love you! To my children and extended family for their constant reminders of my deadlines and for supporting my efforts over the last three years. Finally, I would like to thank Temple University for providing access to the best DBA support staff one could imagine - without their assistance, I would still be researching. Thank you all for your support and encouragement over the last three years.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure Page

CHAPTER 1

INTRODUCTION

Many changes occurring in the business enterprise, result in the need for C-suite executives to respond quickly or risk market share, profits or both. In software firms, organizational agility is essential to being able to rapidly create usable software (Schwaber, 2007). C-suite executives have recognized the importance of agility and have charged their teams with adopting agile methods within the enterprise, with the hopes of fostering greater operational and strategic flexibility in order to maintain and/or gain competitive advantages in the marketplace. Collectively, various agile methods have become known as DevOps (Sanders, 2007). While we have seen a rapid rise in the adoption of DevOps in the enterprise, these agile methods that make up DevOps have been around for a long time (Buchanan, 2015). Which of these DevOps adoption capabilities are most important and what effect do they have on an organization's agility is yet to be determined.

This research aims to accomplish three main goals through two studies. First, through a synthesis of existing research and practitioner literature, we will examine various definitions of DevOps through compare and contrast and will establish an empirical definition of DevOps. This new definition will establish the foundation for understanding the capabilities and methods that make up DevOps adoption. Secondly, we will synthesize, and validate the key capabilities and agile methods through a pilot study and establish, through literature review, potential effects of DevOps adoption on organizational agility.

The third and main goal is to validate the effect of DevOps adoption on organizational agility. To accomplish this, a second research study will be performed that will utilize the output from the pilot study to validate the DevOps adoption and the

organizational agility capabilities. We will also validate and establish whether DevOps adoption capability has an effect on organizational agility.

CHAPTER 2

DEFINING DEVOPS AND IDENTIFYING KEY DEVOPS ADOPTION CAPABILITIES

Introduction

Advances in technologies and the ongoing consumerization of Information Technology (IT) have changed the way firms compete, ultimately dictating the winners and losers in multiple industry segments. These technologies have driven new markets, created new opportunities, and given birth to new firms in old industry segments. Modern methods for software development, such as agile, represent a significant opportunity to improve a software company's bottom line. Market leaders such as Google, Apple, Yahoo, and Amazon have broken down the rigid development concepts of the past and have moved to adopt these more agile methods and practices (Debois et al., 2011). Agile methods and capabilities represent a significant opportunity to reduce a firm's development expense and improve the bottom line by expanding the application of agile methods and processes beyond software development into other parts of the firm in order to further improve organizational agility. When it comes to responding to changing business environments, literature suggests that high performing organizations exhibit key characteristics such as increased organizational agility, an agile organizational structure, highly mature cultures and processes, and advanced delivery methods (PuppetLabs & DORA, 2016).

Further, Schwaber (2007) argued that agile methods and practices yield several benefits including reduced time-to-market, increased quality, reduced waste, better predictability, and better employee morale. Among other benefits are an increased ability to respond to dynamic market changes (Lycett, Macredie, Patel, & Paul, 2003). Firms have

recognized that organizational agility is a core differentiator in today's changing business environment. Emerging technologies have led the way to advanced approaches that have resulted in end-to-end, continuous delivery solutions that are being successfully applied across multiple organizational and business units. Collectively, these advanced approaches and agile methods have come to be known as DevOps. DevOps is an approach to the more traditional application lifecycle management (ALM) process that emphasizes transforming each unit of an organization into efficient operations that can quickly develop, deploy and respond to changes. These approaches involve enhanced team collaboration/communication, tool and process automation, and team integration. Collectively, the adoption of DevOps capabilities and the resulting potential impact on organizational agility stand to revolutionize how highly effective firms are being run. Firms must be agile and be able to handle extreme changes, survive unprecedented threats, and capitalize on emerging business opportunities (Prahalad, 2009). DevOps adoption capabilities enable firms to quickly align their organizational culture and technology infrastructure with rapidly changing business needs (Lesser & Ban, 2016), resulting in increased quality, reduced time to market, and improved customer satisfaction.

While the DevOps phenomenon continues to emerge rapidly within the firm and across many types of organizations, the concept lacks a clear definition (Smeds, Nybom, & Porres, 2015). Based on an extensive literature review, there is no single, comprehensive definition of DevOps. The lack of a uniform definition has resulted in many criticisms and skeptics and in some cases, has led to the development of entirely new roles within software development organizations (Kerzazi & Adams, 2016). In addition, many practitioners, frustrated by the lack of clarity within the practitioner and academic community, have

developed their own definitions. This has increased the tension around the topic of what DevOps is really about – whether it is a technical solution, a new culture to be created or just a fancy new term to describe existing agile methods, practices and processes. All of this confusion results in adoption variations and impedes adoption success. While it is acknowledged that there have been several attempts to clarify the term within the academic and practitioner community (de França, Jeronimo, & Travassos, 2016; Lwakatare, Kuvaja, & Oivo, 2016; Dyck, Penners, & Lichter, 2015; Smeds et al., 2015; Mohamed, 2016; Allspaw & Hammond, 2009; Kim, Humble, Debois, & Willis, 2016; Willis, 2012), however; a comprehensive definition in the literature that clearly summarizes all aspects of DevOps relative to both the academic perspective and the practitioner perspective is not clear. In addition, given the similarity of the characteristics of organizational agility and DevOps, there is reason to believe that the two have an effect on each other in some way.

To that end, the goal of this study is to theoretically define a single, uniform definition of DevOps by examining and identifying the key adoption capabilities and agile methods that contribute to its adoption. This will be accomplished through a comprehensive literature review of both academic and practitioner research. Once identified and thoroughly examined, the DevOps adoption capabilities will be examined by conducting a pilot study. The resulting data, including the pilot study instrument; will be consumed and analyzed for further research contributions including exploring the effects of key DevOps adoption capabilities on organization agility. Accordingly, this research study will explore two primary research questions:

RQ1: What is DevOps?

RQ2: What are the key capabilities of DevOps adoption and what is the effect on organizational agility?

This research study is arranged as follows: first, through literature review and a pilot study, a theoretical definition of DevOps is established. Next, we present a detailed discussion of the key capabilities of DevOps adoption and organizational agility. Our research model is presented in the next section along with a detailed discussion of our data collection process and analysis of the pilot results. Finally, we present the conclusion and limitations to our research.

Literature Review

Given that DevOps is a practitioner-driven movement, it is appropriate to utilize an exploratory practice technique to review practitioner and academic literature as part of a comprehensive review of prior work (Allwright, 2005). To support the literature review, a structured search was performed using the web. Since DevOps is a relatively new phenomenon, discussed and developed mainly by practitioners, a dynamic list of sources was considered which included industry reports and white papers, blogs and social media work group publications, conference proceedings and white papers, industry journals, books, and academic journals. Key words used in the structured search contained text commonly associated with DevOps concepts, such as "DevOps", "dev", "ops", "development", "agile methods", "agile capabilities", DevOps capabilities", "DevOps methods", "operations", "dev-ops", "development operations", "continuous delivery", "devops adoption", "infrastructure as code." The search for organizational agility contained key words such as "organizational agility", "flexibility", "agility", "agile", "agile methods", and "agility measurement." Review of search results involved analyzing links and categorizing sources based on an initial review of content for relevance towards defining DevOps and organizational agility. The search was performed between April 3, 2016 and May 31, 2017 and resulted in 58 results.

A subjective review of the literature results was conducted by reading all abstracts, if available, and reading all relevant sections containing relevant key words associated with DevOps and organizational agility. During the DevOps literature review, we tagged each article by type and recorded the definitions and key concepts for DevOps and agility. From the set of all results, we selected 165 articles that met the specific search criteria related to

DevOps key words used in the search. Of the 165 articles, 58 were associated with DevOps concepts, 104 were associated with agility concepts, and 3 were associated with both DevOps and agility concepts. This in itself is interesting, as it may indicate that while DevOps and agility receive a good deal of attention, they rarely are considered together. Further review was conducted to extract more detailed information related to the definition of DevOps and organizational agility and their relationships. Of the 58 DevOps sources, 45 sources were identified that provide relevant information towards defining DevOps. Sources that were excluded may have contained "buzz words" that were picked up in the search but did not necessarily provide information that contributed to a definition of DevOps. More than 80% (37) of the DevOps sources were sourced from websites, books, conference literature, and practitioner white papers. This further reinforces the idea that DevOps is primarily being discussed and developed in the practitioner community. Table 1 presents the DevOps literature review sources included by resource type category.

Table 1

DevOps Literature Review Sources

Types	# of Sources	References
White Papers	15	Dyck et al., 2015; Enterprise & Arevalo, 2015; Enterprises, 2014-2015, 2015, May 2016; Freeform Dynamics, 2015; Hinchcliffe, 2014; Infosys, 2016; Pankaj, 2004; PuppetLabs, 2015; PuppetLabs & DORA, 2016; RebelLabs, 2013; Relic 2014;Technologies,2014; Waterhouse, 2015
Academic Journals	8	Cancialosi, 2016; Day, 1994; Debois et al., 2011; Eisenhardt & Martin, 2000; Humble & Molesky, 2011; Lesser & Ban, 2016; Lwakatare, Kuvaja, & Oivo, 2015; Lwakatare et al., 2016
Conference Proceedings	3	Horlach, Drews, & Schirmer, 2016; Lwakatare et al., 2016; Rana & Staron, 2016
Books	5	Huttermann, 2012; Kidd, 1994; Kim et al., 2013; Kim et al., 2016; Walls, 2013
Blogs, Websites	14	AgileAlliance, 1990; Allspaw & Hammond, 2009; Barker, 2013; Betteley, 2015; Bradley, 2015; Edwards, 2010; HPE, 2013; ISACA, 2016; Kim et al., 2016; Minick, 2015; PuppetLabs & DORA, 2016; Relic, 2016; Willis, 2010, 2012
Total	45	

DevOps

The term “DevOps” was originally coined by Patrick Debois and Andrew Shafer in 2008 (Willis, 2012). It entered common usage in 2009 during the Velocity Conference (Kim et al., 2013) community with the presentation “10+Deploys Per Day: Dev and Ops Corporation at Flickr,” given by John Allspaw and Paul Hammond (2009). As the DevOps movement has evolved and firms have adopted DevOps capabilities and methods into the enterprise, it has been perceived as the “golden ticket” to unlocking a firm’s ability to achieve success responding to change and enabling organizational agility. However, to date DevOps lacks a clear and precise definition (Lwakatare et al., 2016). This ambiguity around what DevOps really is, results in confusion within organizations, on teams and hinders adoption. Dyck et al. (2015) have proposed a scientific definition of DevOps derived from comparing and contrasting various descriptions of related methodologies. They define DevOps as “a mindset, encouraging cross-functional collaboration between teams – especially development and IT operations – within a software development organization, in order to operate resilient systems and accelerate delivery of changes.” In addition, Kim et al., (2016), defined DevOps as “the outcome of applying the most trusted principles from the domain of physical manufacturing and leadership to the IT value stream. It relies on bodies of knowledge from Lean, Theory of Constraints, the Toyota Production System, resilience engineering, learning organizations, safety culture, human factors, and many others”.

According to a study by Lwakatare et al., (2016), there is general agreement among practitioners that the term DevOps is a combination of development and operations, that encourages collaboration between software development and operations activities. In

addition, since some practitioners rely on Wikipedia as a source, the definition there states that DevOps is a term used to refer to a set of practices that emphasize the collaboration and communication of both software developers and information technology professionals while automating the process of software delivery and infrastructure changes. Furthermore, Huttermann, (2012) offers a practitioner's perspective regarding the definition of DevOps. He suggest that DevOps describes practices that streamline the software delivery process, emphasizing the learning by streaming feedback from production to development and improving cycle time. Kim et al., (2016) offered an additional summary of DevOps as "the emerging professional movement that advocates a collaborative working relationship between development and IT operations, resulting in the fast flow of planned work, while simultaneously increasing the reliability, stability, resilience and security of the production environment".

The literature review suggests that the definition of DevOps encompasses aspects of "The Three Ways", the underlying principles of capabilities that center on systems thinking, experimentation and learning, and feedback (Kim et al., 2013). Systems thinking is supported by the principles of flow, which accelerate the delivery of work from development to operations and ultimately to the customer. Experimentation and learning foster a high-trust organizational culture that supports a dynamic, disciplined, and scientific approach to experimentation and risk taking, facilitating the creation of organizational learning, both from the firm's successes and failures (Kim et al., 2016). Feedback supports the first way of systems thinking by enabling the fast and constant flow of feedback from right to left at all stages of the value stream to proactively prevent challenges.

Through literature review of academic and non-academic sources, 17 capabilities of DevOps adoptions were identified. According to Hair (2017), in the event that there is a lack of an established measurement approach, it is acceptable to develop a new set of measures or substantially modify an existing set of measures. As DevOps is a relatively new concept in the research community, the lack of acceptable measurement approaches is evident in the literature. As a result, the 17 capabilities will be utilized as the measures in the pilot study instrument to be narrow down utilizing the results. The capabilities and their mappings to the pilot study instrument are summarized in Table 2.

Table 2

Capability Characteristic Mapping to Pilot Study Instrument

Capabilities	**Pilot Study Question**
Common metrics for development and operations	Q11.1 Common Metrics
Continuous Software Delivery & Deployment	Q11.2 Continuous Deployment
Continuous Integration and Testing Build Out	Q11.3 Continuous Integration/Test
Continuous Planning and Release Management	Q11.4 Continuous Planning & Release Management
Continuous Monitoring, Optimization and Feedback	Q11.16 Continuous Monitoring, Opt, & Feedback
Common Goals for development & operations	Q11.6 Common Goals
Common Processes & Tools (Standardization)	Q11.7 Standardization
Open channels for constant, effortless communication	Q11.8 Open channels for communication
Promote continuous experimentation and learning	Q11.9 Continuous experimentation
Shared code responsibility, collective ownership	Q11.10 Shared code responsibility

Table 2 continue

Capability Characteristic Mapping to Pilot Study Instrument

Capabilities	Pilot Study Question
Shared goals, definition of success, common incentives across teams	Q11.11 Shared definition of success
Shared values, respect, trust across teams	Q11.12 Shared values, respect, trust
Enterprise Automation (Build, Test, Deployment, Recovery, Monitoring, Infrastructure)	Q11.5 Infrastructure Monitoring
Automated Infrastructure Provisioning	Q11.13 Automated Infrastructure Provisioning
Automated Security Testing	Q11.14. Automated Security Testing
Configuration Management (Software & Infrastructure)	Q11.15 Configuration management
Environment Virtualization	Q11.17 Environment Virtualization

As DevOps adoptions grow, organizations are learning to balance their newly developed capabilities with control, compliance, and security requirements with the need for speed and innovation. Audit, control, and security approaches are being integrated slowly with the DevOps capabilities being implemented across the enterprise such that risks are proactively planned and mitigated and issues are being addressed utilizing an iterative approach. It is also important to recognize that, as DevOps capabilities continue to evolve, so do the requirements for application security, audit and controls (Dyck et al., 2015) such as those imposed by Sarbanes-Oxley, HIPAA, and other auditing compliance bodies. Further exploratory review of the literature characterizes the 17 capabilities into five broad categories critical to DevOps capability:

1. Collaboration/Communication Capability

Overwhelmingly, literature suggests that improving collaboration/communication among teams is a critical capability of adopting DevOps in the enterprise.

Collaboration/communication is a process in which autonomous or semi-autonomous actors interact through formal and informal negotiation, jointly creating rules and structures governing their relationships and ways to act or decide on the issues that brought them together; it is a process involving shared norms and mutually beneficial interactions (Thomson, Perry, & Miller, 2007). Active and open collaboration/communication are the foundation through which a DevOps culture is created. Typically technical teams have been siloed and interact through complex ticketing systems and/or archaic request procedures which require elevated levels of management approvals before the people who can solve the problem get involved. This laborious process is at odds with the type of collaboration/communication required by the adoption of DevOps.

2. Continuous Monitoring Capability

Continuous monitoring is a firm's ability to track the processes performed within a computing system and provide reporting services to the system or network administrator (PuppetLabs, 2015; Relic, 2016). This also involves proactive detection of events that occur in the environments in order to expose issues before they become catastrophic failures. To be effective, continuous monitoring must be able to quickly determine when a service is unavailable, teams must understand the underlying causes and most importantly, apply these learnings to anticipate problems before they occur (Pivotto, 2013). Literature suggests that continuous monitoring is integral to service delivery and therefore suggest that it is part of service definition (Relic, 2014). Continuous monitoring is most effective utilizing tools that can monitor server performance and application performance, such as AppDynamics and NewRelic.

3. Measurement Capability

Measurement capability is a firm's ability to collect efficient metrics to support the decision-making in the software development and operations lifecycle (Shamow, 2011) (Debois et al., 2011). DevOps requires continuous refinement as the firm continuously adopt new capabilities and introduce them into the business and development environment. Measuring the results of integrated capabilities is key to understanding how to evolve DevOps throughout the firm.

4. Automation Capability

Automation is the ability to utilize technology and tools to reduce effort and improve software delivery by automating manual and repetitive tasks (PuppetLabs & DORA, 2016). DevOps relies heavily on the ability to automate large parts of the end-to end product development process. Two common practices that are typically automated as part of a DevOps adoption include, but is not limited to, the deployment process and infrastructure as code. The deployment process is invoked when newly-written code is committed to version control, built, tested and deployed into the production environment. This process is repetitive and often error-prone; an automated process would manage all of the details and dependencies of the deployment, ensuring that there is consistency in the execution. Infrastructure as code involves treating infrastructure the same as developers treat code. This means that the management of the infrastructure environment is done utilizing a similar automated process as used by developers to manage source code. Automation of the repetitive infrastructure provisioning steps enable quality and speed of delivery to customers and reduce the amount of disruption to the environment. This new way of managing the infrastructure in a firm enables teams to deliver stable environments rapidly and at scale. Teams avoid manual configuration of environments and enforce consistency

by representing the desired state of their environments by code. Deployments are repeatable and prevent runtime issues caused by configuration drift or missing dependencies.

5. Organizational Culture Capability

Organizational culture capability is a complex set of values, beliefs, assumptions, and symbols that define the way in which a firm conducts its business (Barney, 1986). Groups of people create culture through shared values and behaviors (Walls, 2013). A good working environment that welcomes innovation, experimenting and stops finger pointing when mistakes are made helps to facilitate and embrace all that DevOps has to offer. The culture required to support a DevOps adoption involves respect, trust, open communication, incentive and responsibility alignment (Walls, 2013). Organizational culture has been shown to be significant in prior literature (Homburg & Pflesser, 2000) (Berson, Oreg, & Dvir, 2008) (Gregory, Harris, Armenakis, & Shook, 2009) and is highlighted in capabilities theory as an important factor in a firm (Day, 1994). In the context of DevOps, organizational culture is a distinct capability that provides clear guidance on how to execute in an environment that highlights the principles underpinning DevOps.

Just as the literature suggests various definitions of DevOps, the literature provides insights into what DevOps is not. According to the literature, DevOps is not a tool, a process or a methodology, yet is enabled by tools and technologies, processes and methodologies defined in the enterprise. As outlined, there have been several attempts within the academic and practitioner community to standardize on a definition. Each of these definitions support the original, more conceptualized definition of DevOps as

presented by Dyck et al., (2015). Lwakatare et al., (2016) further proposed that DevOps is mind-set substantiated with a set of practices to encourage cross-functional collaboration between teams – especially development and IT operations – within a software development organization, in order to operate resilient systems and accelerate delivery of change. Likewise, Dyck et al., (2015) proposed a scientific definition of DevOps derived from comparing and contrasting various descriptions of the term and release engineering because the two terms seemed to overlap and share similar goals of providing high quality software outputs as expeditiously as possible. Dyck et al., (2015) defined DevOps as a mindset, encouraging cross-functional collaboration between teams, especially development and IT operations, within a software development organization, in order to operate resilient systems and accelerate delivery of changes. While these definitions offer the most comprehensive and scientific that could be found in academic literature to date, they both focused strictly on the software development environment and the roles of IT operations and development teams. The definitions also lack inclusivity of key capabilities that characterize successful DevOps adoptions such as collaboration/communication, automation, culture, and measurement capabilities. DevOps practices and capabilities do not have to be confined to just development and IT operations; these capabilities can be employed in many other parts of the organization to improve organizational capabilities and to achieve maximum effect on a firm's ability to respond to changes in the market place. To that end, we put forth an answer to research question (RQ1): what is the definition of DevOps? We posit a definition that can be applied broadly and generally as: *"DevOps is a convergence of ideas, a set of capabilities, practices and methods that aim to eliminate the gaps between teams by enabling a culture of cross-functional*

collaboration/communication, continuous monitoring, measurement and automation capabilities in order to develop and operate resilient systems and accelerate delivery of change."

Bringing clarity to the definition of DevOps and understanding the key challenges, will enable successful adoptions of these capabilities. While the literature review presents a summary of available information, it is not all inclusive, due to the lack of availability of research regarding DevOps implementations in the business environment. We do know, however; that accelerating delivery of change involves adopting a variety of agile methods, practices and capabilities to enable the ability to respond to anticipated or unexpected change in the business environment in the proper. This results in the ability to exploit those changes to take advantage of them as opportunities (Sharifi and Zhang, 1999). In this new era of business competition, a firm's success is driven largely by its ability to cope with this change. This ability is what drives organizational agility of the firm and thus creates a competitive advantage (Horney & O'Shea, 2015).

Organizational Agility

Agility is not a concept unique to IT, although the formation of the Agile Alliance and the publication of the Agile Manifesto formally introduced agility to the IT sector, more specifically to the software development field (AgileAlliance, 1990). The concept of agility goes back as early as 1992, first introduced in the 21st Century Manufacturing Enterprise Strategy report from Lehigh University's Iacocca Institute (Nagel, 1992). Agility from a software perspective arose from the literature on flexible and lean manufacturing (Börjesson, Martinsson, & Timmerås, 2006) and has been rapidly adopted by organizations producing software using agile programming methodologies (Aoyama, 1998). Agile programming methodologies refer to a set of software development methodologies that are based on the Agile Manifesto and utilize an iterative development approach where requirements are elaborated and developed through collaboration between self-organized cross functional teams (AgileAlliance, 1990). The most popular agile methodologies are Extreme Programming (XP), Scrum, Crystal, Dynamic Systems Development Method (DSDM), Lean Development, and Feature Driven Development (FDD). These methodologies provide an alternative approach to help teams respond to unpredictability in the business environment through incremental, iterative cadences and feedback versus the traditional waterfall, sequential development approach which allows the team one opportunity for feedback before the product is built, tested and validated.

The iterative cadences and feedback loops of agile programming methods promote a more optimized approach compared to the waterfall methodologies of the past which traditionally allow one iteration and limited feedback loops. Agile programming methodologies help firms build the right product. Instead of committing to market a

product that hasn't even been built yet, agile empowers teams to continuously re-plan their release to optimize its value throughout development, allowing them to react quickly to change and allows for the firm to be as competitive as possible in the marketplace. The results are a more agile process that is able to support an organizational and competitive need to be quick and nimble and to be able to adjust as the market and end customers' demands. It is through these benefits that we believe that effective deployment and use of agile methods leads to enhanced organizational agility.

The role that IT has played in enhancing agility has been studied extensively in literature (Lu & Ramamurthy, 2011; Pankaj, 2004; van Oosterhout, Waarts, & van Hillegersberg, 2006). In the last decade agility has gained more support as a measure in both academic and practitioner literature as a means to address the need to understand the impact of customer demands resulting from the unpredictable changes and technological advances in the marketplace. As agile has grown as a capability, there is increased focus in other industry segments to extend the adoption of agile practices and methods beyond development and operations to include the convergence of movements that support the product lifecycle, such as DevOps. One might also argue that Enterprise architecture is one such movement where the extension of agile practices is applicable and heavily impacts an organizations ability to be agile. An exhaustive literature review suggest that successful digital transformations require a more solid agile approach to enterprise architecture in order to effectively impact an organizations agility. If we build on the strength of both enterprise architecture and DevOps, the concepts are complimentary and thus, lead to the creation of enterprises that grow and develop as one unit without having

central, command and control oversight that tends to stifle innovation and limit organizational agility.

The study of organizational agility and DevOps face the challenge of diverse and often contradicting definitions due to the multidimensional and vagueness of the concepts leading to difficulty in defining and measuring the two constructs. These issues lead to problems in operationalization and consistency in measurement (Bahrami, 1992) (Golden and Powell, 2000). The importance of organizational agility as a dynamic capability allows a company to make timely, effective, and sustained changes repeatedly (Worley, Williams, and III, 2014). It is the ability to quickly identify (strategic responsiveness) and execute initiatives (organizational flexibility) to respond to opportunities and risks that align with a firms overall strategy. Kidd (1994) posits that an agile organization is a quick, compatible and aware enterprise which could adapt rapidly to unexpected and unpredicted changes, market opportunities and customer's demands.

One common theme in the literature is that organizational capabilities are composite bundles of coherent competences, skills, and technologies, rather than single discrete skills (Mohrman, Docherty, Shani, Teigland, & Schenkel, August 2006). Through dynamic capability view, the concept of agility explains how organizations can leverage the internal resources to acquire necessary competencies in dealing with hypercompetitive marketplaces (Madhavaram and Hunt, 2007; Teece, 2007). Dynamic capabilities view posits agility as the capability of sensing and responding, which is an organizations ability to sense and respond quickly to opportunities for innovation and competitive action. Furthermore, literature review agrees that strong sensing and responding capabilities are critical to an organizations success in turbulent environments and having the ability to

sense new opportunities include the capabilities to scan, learn, and interpret activity (Roberts and Grover, 2012; Zaheer and Zaheer, 1997) . Once an opportunity for competitive action is sensed, it is then addressed by mobilizing an organizations existing processes. Processes exist throughout an organization and through various frameworks, however; DevOps, as a process and a framework, enables the capabilities to support organizational agility through the various techniques introduced into the work environment and into the organizations culture, such as embracing of and the ability to respond to change, encouragement of open communication across the enterprise, and promotion of a culture that values and promotes innovation and change.

Yu and Ramamurthy (2011) posited that IT capability enables market capitalizing agility and operational adjustment agility. This same study also revealed that IT capability is essential to achieve organizational agility (Lu and Ramamurthy, 2011). Firms need to continuously nurture and develop superior firm-wide IT capability to successfully manage and leverage their IT resources, tools, processes and behaviors in order to build an agile organization. Data from Hewlett Packard Enterprises suggests that agile organizations exhibit IT capabilities that enable growth, improve profitability, increase agility, boost productivity, improve the customer experience, increase innovation and reduce risks (Enterprise and Arevalo, 2015).

While agility establishes a for launching competitive actions with speed, surprise, and disruption (Sambamurthy, Bharadwaj, & Grover, 2003), it is not enough for an organization to just possess agility. For an organization to maintain these competitive attributes, agility must be continually activated and exercised. This spawns innovation and entrepreneurship which in turn activate the necessary capabilities to facilitate competitive

actions. Organizations that are able to sense and respond to obstacles and opportunities better and faster than the competition will have a clear advantage. A true agility advantage will be very hard for others to replicate and hence overcome (Horney and O'Shea, 2015).

To enable agility, there are many models that have been researched and found in academic literature. A vast number of measurement approaches have been offered over the years to measure organizational agility. Many of these approaches have focused on the operational features of the firm to measure agility, while others have focused on the business environment or the process of agility itself. Additionally, a quantitative index of agility was first proposed by Yauch and Navaresse (2006) to be used to measure agility of manufacturing organizations.

Several approaches have been explored for measuring and assessing agility. One such example found in the literature review, a process based approach, performed by Sieger, Badiru, and Milatovic (2000) concentrated on a single type of process, such as software development or product management. This approach measured responsiveness of companies relative to the product development cycle time. Another approach, put forth by Caswell and Nigam (2005) posits that business agility requires both intrinsically agile IT systems support and a low impedance path from the specification of business change to a specification of IT change.

Sharifi and Zhang (1999) will be utilized to understand and measure organizational agility, given that is the only approach found in the review of literature that can be applied across a diverse enterprise regardless of process or type of organization. It includes an extensive analysis of environmental turbulence and provides a thorough understanding of the major factors and triggers for change that an enterprise would face (Yauch, 2011).

Table 3 presents the four key organizational agility capabilities as defined by Sharifi and Zhang (1999).

Table 3

Four Key Capabilities of Organizational Agility

Agility Capability	Definition	Reference
Responsiveness	The ability to identify changes and respond fast to them, reactively or proactively, and recover from them.	Sharifi and Zhang, 1999; Yusef, Sarhadi & Gunasekaran
Competency	The extensive set of abilities that provide productivity, efficiency, and effectiveness of activities towards the aims and goals of the firm.	Sharifi and Zhang, 1999; Yusef, Sarhadi & Gunasekaran
Flexibility	The ability to process different products and achieve different objectives with the same facilities.	Sharifi and Zhang, 1999; Yusef, Sarhadi & Gunasekaran
Quickness	The ability to carry out tasks and operations in the shortest possible time.	Sharifi and Zhang, 1999; Yusef, Sarhadi & Gunasekaran

Sharifi and Zhang (1999) proposed that an agile organization should demonstrate each of the four capabilities in order to enable appropriate response to changes taking place in its business environment. These capabilities represent various capabilities that are demonstrated at different levels of the organization by way of implementation of practices methods, and through the use of certain tools. Coincidentally, many of these methods, processes and tools have made their way into being categorized as DevOps practices and have found to be necessary to gaining the required capabilities of agility.

Discussion of Key Capabilities

Capability theory provides an appropriate framework through which to explore and investigate DevOps adoption capabilities and their impacts (Day, 1994). Capabilities as defined by Eisenhardt and Martin (2000), are the firm's processes that use resources, specifically the processes to integrate, reconfigure, gain and release resources, to match and even create market change. Capabilities are the organizational and strategic routines by which firms achieve new resource configurations as markets emerge, collide, split, evolve, and die (Eisenhardt & Martin, 2000).

DevOps represents one such collection of organizational and strategic routines that has emerged as markets have evolved over time. Moreover, DevOps adoption, in its ability to facilitate quick respond to change, allows the firm to leverage its information technology to create new resource configurations based on dynamic firm needs. It is through leveraging DevOps adoption capabilities that firms are able to effect organizational agility which is needed in order to quickly respond to the rapid, continuous change that we see in the marketplace and the business environment today. DevOps adoption capability is also comprised of a combination of a set of distinct resources: development and IT operations labor, organizational culture, collaboration/communication, automation and continuous monitoring. Capability theory describes categories of these capabilities. They are broken into three distinct categories: outside-in processes, spanning processes and inside-out processes (Day, 1994). DevOps encompasses components that spawn each of the three classes of capabilities, but mainly focuses on effecting the inside-out processes as a technology development approach with a goal of increasing efficiency and effectiveness of a firm.

Through the pilot study, several DevOps adoption capabilities emerged from the literature review and are examined. To address the research question: what are the key methods, processes, and tools that enable DevOps adoption capability and how do they relate to organizational agility, we begin a discussion of the capabilities by prioritizing them based on the pilot study results. The relationships were examined based on the evidence seen in industry and in literature and it suggest that there is a relationship between DevOps adoption capability and organizational agility (Hand, 2015; James Sena, Coget, & Shani, 2009; Lu & Ramamurthy, 2011). Next we present a review of the prioritized DevOps adoption capabilities resulting from the pilot research.

Collaboration/communication capability

How an organization communicates determines its overall agility (Harraf et al., 2015). Agile organizations must be able to quickly and effectively respond to change, often while by themselves changing. While multiple communication channels usually exists within an organization, change in itself facilitates the need for collaboration and communication, yet, while in parallel; also makes collaboration and communication difficult. The need for organizational collaboration and communication among project team members has enabled the paradigm of how data and personal relationships will flow in an organization. This flow is aided by technology, through helping virtual teams collaborate and communicate and enabling smooth infrastructure transport. A collaborative work environment that promotes open communication engages employees at all levels resulting in an agile organization that is able to effectively innovate and quickly solve problems. Collaboration is more than just knowledge sharing, but rather it is about combining resources to create something new. This results in multi-directional, open

communication throughout the organization which is crucial to establishing organizational agility through the ability to actively engage (collaborate and communicate with) all levels of the organization such that resistance to change is minimized, differing opinions are offered, and more effective solutions are developed (Harraf et al., 2015).

Continuous monitoring, optimization and feedback and measurement capability

Organizations that conduct frequent internal assessments to monitor and measure the overall processes that are performed on a computing system are less likely to cite unaligned business processes as an obstacle to achieving organizational agility. Given the various moving parts involved with developing products and services, continuous monitoring enables business leaders the ability to recognize and interpret trends and patterns better and faster, thus; allowing quicker response to change and enabling organizational agility. While technology enables continuous monitoring, determining what to measure has an equal impact on the firm's ability to react to change in the business enterprise. What a firm measures and monitors, effectively define the priorities and create the level of focus required to enable maximum organizational agility.

Fact based measurement is a process for insuring that judgments about whether outcome and performance expectations are met are grounded in reliable, valid, and relevant information linked to the firms definition of success (Horney & O'Shea, 2015). The ability to anticipate change is built through real-time alignment of predictive, sensing capabilities that feed measuring and monitoring capability focused on tracking leading indicators to identify key patterns and trends (Horney & O'Shea, 2015). Successful agile organizations have established continuous measurement and monitoring capabilities that enable strategies and operating dexterity from the insight gained from the monitoring output.

Automation capability

While firms are focusing on improving communication/collaboration and measuring and monitoring their processes and operations to improve agility, many IT organizations continue to rely on manual processes and custom scripts to accomplish repetitive tasks. Manual processes are inherently slow. In the past, this was less of a concern because the pace of the competitive IT environment itself was slower. Today, the dynamic nature of the competitive environment has resulted in very complex IT systems and data centers to support them. These systems are made up of heterogeneous infrastructure that is highly dispersed; both inside and outside of corporate firewalls to enable IT to respond instantly to business needs. Without automation, delays are inevitable and these complex IT systems become very difficult to scale, track and maintain and require complex configuration management solutions. This results in increased bottlenecks that lead to elevated downtimes and decreased organizational agility. Inside the agile organization, from engineering to operations to customer service and marketing, processes are adapted to ever-increasing automation for greater speed and responsiveness in delivering the optimal customer experience (Schorsch, 2015). That automation reduces manual work, minimizes errors and defects, and combined with a modular, reusable IT architecture, lets you quickly innovate, launch new services and rapidly iterate to improve them (Schorsch, 2015).

Organizational Culture

Organizational culture has proven to be an important characteristic influencing the firm (Homburg & Pflesser, 2000; Berson et al., 2008; Gregory et al., 2009) and in DevOps transformations (Cancialosi, 2016; Walls, 2013). It is also highlighted in capabilities theory as an important factor in building high performing organizations (Day, 1994). The underlying behaviors that make up organizational culture have an influence on the behavior of organizational team members, as people rely on these values to guide their decisions and behaviors (Gregory et al., 2009). In fact, to build a high performance organization via DevOps one often needs to change the organizational culture. This means that people need to radically change their behaviors. Additionally, it also requires an organizational change in systems and processes to facilitate and sustain these behavioral changes. An organizational culture that is supportive of the ideals of the DevOps movement is crucial (Walls, 2013).

Firms that have been able to make this cultural shift, have been able to deliver value to their customers in ways that their competitors have not been able to do. Facebook, PayPal and eBay have embraced bringing together the disparate functional roles of the traditional environment in order to share accountability, to mitigate risks (Lwakatare et al., 2015; Cancialosi, 2016), and to increase the overall agility of the organization. Through multiple-case study, Kompella (2014), found that agility displayed could be attributed to the inherent nature of agile methods, but that changes in organizational culture were required in order to maximized agility. He further demonstrated insights into the complex interaction of agility and organizational culture and posited methods and processes to achieve agility. In this research, a measure of the maturity of organizational culture is

included that is based on trust, flow of information, response to change, and shared responsibility, all factors that are key to DevOps transformation (Hand, 2015; Walls, 2013).

Data Collection and Analysis

The DevOps phenomenon provides the basic structure and ideas for examining theories and methods regarding the impact of agile methods and capabilities on organizational agility. Based on the research model, this pilot study aimed to validate the theoretical definition of DevOps by examining the key factors that enable the core competencies of DevOps adoption capability. The results of the pilot study established the foundation for subsequent research. Table 4 presents a summary of the operational definitions, the measurable constructs associated with the capability, supporting literature and pilot study instrument reference numbers. Further, the resulting pilot research model is presented.

Table 4

Capability Definitions and Pilot Study Instrument Relationships

Measurable Capability	**Operational Definition**	**Supporting Literature**	**Pilot Study Question Number**
1.Collaboration/ Communication Capability	A process in which autonomous or semi-autonomous actors interact through formal and informal negotiation, jointly creating rules and structures governing their relationships and ways to act or decide on the issues that brought them together; it is a process involving shared norms and mutually beneficial interactions.	Thomson et al., 2007	Q11.7 Standardization Q11.8 Open channels for communication Q11.16 Continuous Monitoring, Opt, & Feedback
2. Continuous Monitoring Capability	A control activity that represents the process that a firm puts into place to ensure that the organization's policies, procedures, and business processes are operating effectively.	Coderre & Verschoor, 2006	Q11.4 Continuous Planning & Release Mgmt Q11.5 Infra Monitoring Q11.7 Standardization

Table 4

Capability Definitions and Pilot Study Instrument Relationships

Measurable Capability	Operational Definition	Supporting Literature	Pilot Study Question Number
3. Measurement Capability	A firm's ability to collect efficient metrics to support the decision-making in the software development and operations lifecycle	Humble & Molesky, 2011; Shamow, 2011	Q11.1 Common Metrics Q11.7 Standardization
4. Automation Capability	The ability of firm to utilize technology in considering that manual, and repetitive tasks can be automated to reduce unnecessary effort and improve software delivery	Hedemark 2014 Puppet Labs 2014	Q11.2 Continuous Deployment Q11.3 Continuous Integration/Test Q11.5 Infra Monitoring Q11.13Automated Infra Provisioning Q11.14. Automated Security Testing
5. Organizational Culture Capability	A complex set of values, beliefs, assumptions, and symbols that define the way in which a firm conducts its business.	(Barney, 1986) Homburg & Pflesser, 2000 Berson et al., 2008; Gregory et al., 2009	Q11.6 Common Goals Q11.7 Standardization Q11.8 Open channels for communication Q11.9 Continuous experimentation Q11.10 Shared code responsibility Q11.11 Shared definition of success Q11.12 Shared values, respect, trust

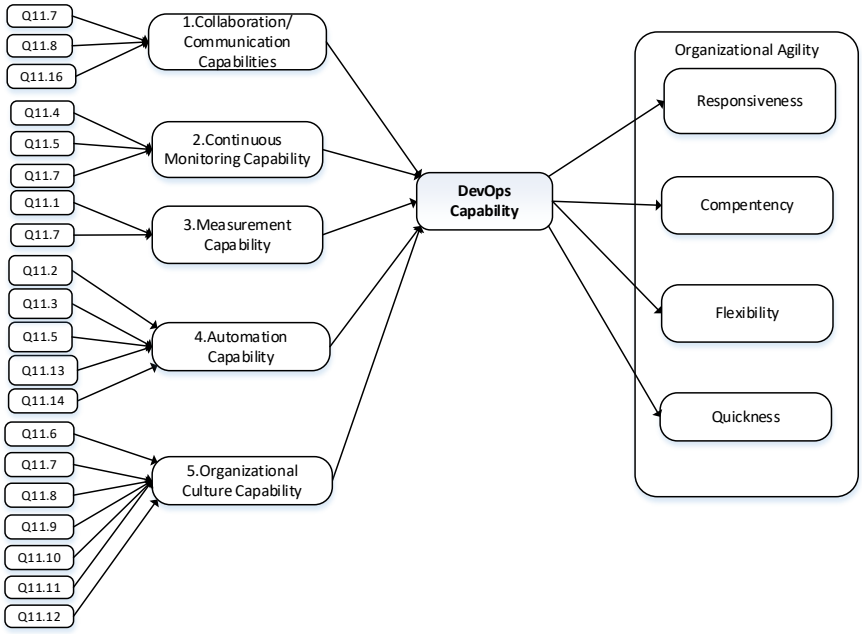


Figure 1**.** Pilot Study Research Model.

Respondents for the pilot study were gathered from three primary sources: 1) six LinkedIn DevOps interest groups, 2) Amazon's Mechanical Turk (MTurk) crowdsourcing internet marketplace, and 3) a private Fortune 100 DevOps Yammer DevOps interest group. An invitation to participate in the pilot study was also posted on a private Fortune 100 DevOps user group. Additionally, invitations were posted on LinkedIn user groups focused on DevOps and Agile practices. Snowball sampling methods were used to target greater sampling sizes. Participants were given a 14 days to complete the pilot study instrument survey and were asked to refer other DevOps colleagues to participate in the study.

Data was collected from a web-based survey utilizing the Qualtrics survey tool. The pilot study targeted global subject matter experts who are familiar with DevOps

practices and methods. Amazon's Mechanical Turk (MTurk) was used to identify subject matter experts who had exposure to DevOps adoption capabilities, were employed full time and had a US graduate degree. Respondents who completed a survey were provided with a completion code and were paid one-dollar upon validation.

After removing incomplete responses, there were 45 responses total. Of the 45 respondents, 47% (21) were the result of MTurk solicitations and 53% (24) resulted from DevOps user groups and snowballing techniques. Survey data was reviewed for validity and interesting associations that might drive execution of the full study. Descriptive statistics and frequencies were utilized to analyze key data provided by the survey and to analyze the importance of key capabilities of DevOps. The pilot study instrument measured each capability on a 5-point Likert scale developed based on prior academic and practitioner literature.

An initial descriptive analysis of the survey results across factors such as demographics, job role, and industry was performed. A wide variety of roles and industries were represented in the responses and they are described as shown in Tables 5 and 6. Industries represented included, 31% reported working in IT services, 22% reported working in education, 11% reported working in government, 7% reported working in industrial and manufacturing and the remainder reported working in other industries.

Table 5

Respondents by Industry

Industry	Number of Respondents
IT Services, Technology	32.56%
Education	23.26%
Government (Federal, State, or Local)	11.63%
Industrial & Manufacturing	6.98%
Pharma, Healthcare & Biotechnology	4.65%
Professional Services	4.65%
Other	4.65%
Aerospace, Defense	2.33%
Hospitality, Travel, Tourism	2.33%
Logistics, Distribution	2.33%
Media, Entertainment & the Arts	2.33%
Retail	2.33%

Of the respondents, 29% reported working in a role that was not identified in the survey, 18% reported working as a project/program manager, 16% reported working as a technical leader, 13% reported working as a technical manager, 11% reported working as a product owner/manager, and the remainder reported working as a designer or operations engineer.

Table 6

Respondents by Role

Industry	Number of Respondents
Other	30.95%
Project/Program Manager	19.05%
Technical Leader	16.67%
Technology Manager	14.29%
Product Owner/Manager	11.90%
Designer	4.76%
Operations Engineer	2.38%

Table 7 below presents the mean rankings and standard deviations for the seventeen (17) capabilities in ascending order of importance (1=Extremely Important, 2=Very

Important, 3=Moderately Important, 4=Slightly Important and 5=Not Important). Continuous Integration and Testing Build was viewed as most important by respondents. Likewise, enterprise automation, continuous monitoring, optimization and feedback, shared goals, common goals for development & operations, common processes & tools (standardization) round out the top five important capabilities identified by respondents.

Table 7

Mean rankings of capabilities by degree of importance

DevOps adoption capability	Mean	Std. Dev
Continuous Integration and Testing Build Out	1.78	0.698
Enterprise Automation Capability	1.85	0.662
Continuous Monitoring, Optimization and Feedback	1.85	0.818
Shared goals, definition of success, common incentives across teams	1.89	0.801
Common Goals for development & operations	2.00	0.832
Common Processes & Tools (Standardization)	2.00	0.784
Configuration Management (Software & Infrastructure)	2.11	0.934
Common metrics for development and operations	2.15	0.907
Continuous Planning and Release Management	2.15	0.818
Shared values, respect, trust across teams	2.15	0.989
Open channels for constant, effortless communication	2.15	0.989
Continuous Deployment	2.22	0.801
Automatic Security Testing	2.44	1.050
Environment Virtualization	2.48	0.935
Promote continuous experimentation and learning	2.48	0.935
Automated Infrastructure Provisioning	2.52	0.893
Shared code responsibility, collective ownership	2.59	0.971

The four key DevOps adoption capabilities that emerged from the analysis of the pilot study results are continuous integration and testing, continuous monitoring, optimization and feedback, automation capability, shared goals and common tools and

processes. In addition, due to the sample size of the pilot study, further analysis to validate the statistical models should be performed.

Limitations of Research

While many respondents (33%) represented that they worked in the IT Services and Technology industry, the expectation of this research is that the results will be general enough to be industry agnostic and guidance can be applied across multiple industry segments. In addition, this study targeted respondents who were familiar with DevOps methods, tools, and processes and therefore purposefully excluded those who may be practicing DevOps methods and processes and using DevOps tools but who are not familiar with the term DevOps. This decision during the pilot was meant to reduce variability in survey responses, however it creates limitations to gathering impacts from other contexts.

In addition, this pilot study has several limitations, since adoption of these complex methods and practices inherently face challenges as standalone concepts. While it is acknowledged that DevOps is a departure from traditional practices and as a term describing the merging of these concepts into a new phenomenon is relatively new; the underlying methods and practices that are merged is not new. For that reason, challenges related to these underlying methods, practices, and concepts continue to exist as organizations adopt DevOps practices and need to be handled accordingly. Additional research in these areas is warranted in order to understand the impacts of these challenges and how they might alter the key capabilities and relationships identified through the pilot research study.

Examples of concepts identified for future research that were not explored in detail in this research study but should be considered and explored as part of a DevOps adoption implementation includes, but is not limited to Dev and Ops toolset clashes, security and compliance related challenges, and migration of legacy enterprise architecture and

infrastructure. It would be beneficial for future research to focus on further empirical evidence of DevOps practices and methods as it relates to the challenges associated with adoption within an enterprise and for which kinds of system, organization and domains where DevOps adoption capabilities are being considered.

Conclusions

The goal of this pilot study was to answer two key research questions: 1) develop a definition of DevOps and to identify the key DevOps adoption capabilities and relatively examine their relationship to organizational agility. As a foundation of establishing an understanding of the key capabilities, the academic and practitioner literature was synthesized. Seventeen DevOps adoption capabilities were identified, with four key capabilities emerging from the study, indicated by the mean statistic where the top four have a mean of less than 2.0. Those four categories of capabilities critical to DevOps are as follows:

1. Continuous Integration and Testing
2. Automation Capability
3. Continuous Monitoring, Optimization and Feedback
4. Shared goals and common tools and processes

Although the statistical models indicate that the pilot study requires a larger sample size to ensure consistency of the results, the study results validate that certain DevOps adoption capabilities are more important than others. Furthermore, the relationship between the key DevOps adoptions capabilities and organizational agility need to be statistically assessed, correlated, and validated. Understanding the effects and the relationship could have an impact on a firm's strategic business decisions and performance outcomes.

CHAPTER 3

EXAMINING THE EFFECTS OF DEVOPS ADOPTION CAPABILITIES ON ORGANIZATIONAL AGILITY

Introduction

Of utmost importance for an organization is to manage uncertainties in the business environment. In fact, turbulent times and uncertainty in the business environment have been recognized as the cause of most business failures (Small & Downey, 1996). These uncertainties have been studied around the world within the research environment (Sharifi & Zhang, 1999). Academic groups and funded research institutes world-wide have carried out research programs in order to understand and diagnose the roots, causes and effects of changes in the business environment. Summarizing the literature on this topic, it can be concluded that the biggest threat to the new way of managing an organization is the ability to cope with unexpected changes, to survive unprecedented threats to the business environment, and to take advantage of changes as opportunities. In this second research study, the effects of the key DevOps adoption capabilities identified in the pilot research study are examined against the agility capabilities put forth by the model developed by Sharifi and Zhang. Additional analysis of the literature was performed to ensure a deeper understanding of the key DevOps adoption capabilities relative to their effects on organizational agility. Consistent with the research approach performed by (PuppetLabs & DORA, 2016), we developed a set of measures that were identified through pilot data analysis, literature review, and that were consistent with industry best practices.

Conceptual Model & Hypotheses

Consistent with the pilot study, this research study builds on capability theory which provides the theoretical framework to perform research into the capabilities of organizations and the impact of those capabilities on the firm (Day, 1994). Capability theory defines the capabilities of organizations into three very broad categories. One of those categories, inside-out, is a category of dynamic capabilities that are deployed from the inside-out and activated by market requirements, competitive challenges, and external opportunities (Day, 1994). Relative to this research study, organizational agility and DevOps can be considered inside-out capabilities because a firm has to develop these capabilities from within by improving the efficiency and effectiveness inside of the organization. Harraf et al., (2015) proposed that agility is the ability of an organization to respond swiftly and effectively to change. In addition, Sharifi and Zhang posits that agility is a capability that comprises two main factors: responding to change (anticipated or unexpected) in proper ways and in due time and exploiting changes and taking advantage of them as opportunities. DevOps, by way of definition, is a convergence of ideas, a set of capabilities, practices and methods that aim to eliminate the gaps between teams by enabling a culture of cross-functional collaboration/communication, continuous monitoring, and measurement and automation capabilities in order to develop and operate resilient systems and accelerate delivery of change. As inside-out capabilities, DevOps and organizational agility can be seen as dynamic, cultivated capabilities because they ultimately enable a firm to sense opportunities and threats, solve problems quickly, and effectively change the organization in a timely, effective and sustainable way in order to either maintain or improve the firm's position in the market. To illustrate the high level

relationships described, Figure 2 below previews the overall conceptual model and indicates how the constructs of DevOps adoption capability and organizational agility capability are measured.

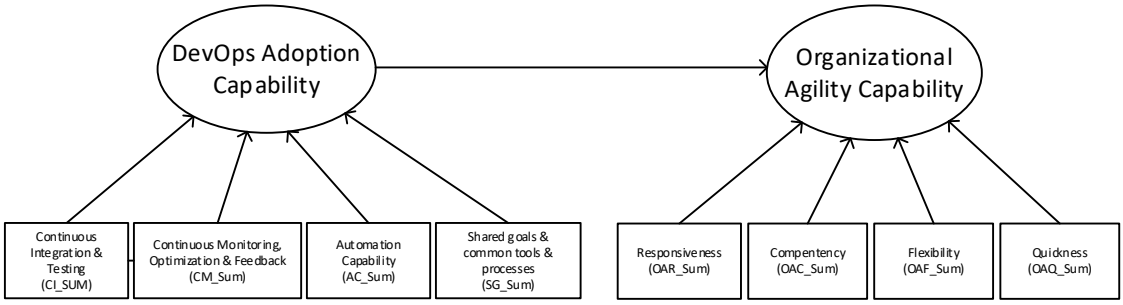


Figure 2. Conceptual Model.

In the context of this research study, the effects of DevOps adoption capability on organizational agility are examined. Therefore, we hypothesize:

H1. DevOps adoption capability has an effect on organizational agility capability.

As an inside-out capability, internal processes that contribute to DevOps adoption and organizational agility are identified in the model. These capabilities define processes that an organization should be able to carry-out. Results from the pilot study identified four key DevOps adoption capabilities: continuous integration and testing, continuous monitoring, optimization and feedback, automation capability, and shared goals, common tools and processes. Continuous integration and testing represents a set of capabilities performed within an organization that should be carried out incrementally and continuously to merge and test source updates from all contributors on a team thereby facilitating rapid defect resolution, improved software quality and reduced time to validate and release

products to end users. Performing these activities efficiently without failures allows an organization to respond quickly to changes in the business environment thereby contributing to the ability of the organization to be agile. At the core of DevOps are the principals facilitated by continuous integration and testing because they provide a means to enable full agility by making development, operations and testing happen in parallel. More importantly, agile and DevOps are necessary foundations for continuous integration and testing.

As processes and tasks are being executed in parallel, continuous monitoring and feedback becomes critical for all teams involved. According to DevOps, continuous monitoring and feedback is understood as using data collected from operating the service as input into the planning and development (Smeds et al., 2015). This data collected represents a continuous pattern of feedback that permeates throughout the organization from the production environment to the start of the development cycle (Lwakatare et al., 2016). These feedback loops enable the DevOps adoption capability of automation by reducing the amount of errors injected in the process and the system. As automation is enabled it facilitates other technological DevOps adoption capabilities such as test, deployment and recovery automation. Automation capability enables the series of steps that must be executed to effectively meet the objectives of DevOps. It is the ability of a firm to utilize technology to automate manual and repetitive tasks across the business enterprise in order to reduce unnecessary effort and improve delivery (Hedemark, 2014). By leveraging the benefits of automation, DevOps capabilities can be realized more quickly and can be accomplished consistently. As a core capability and success enabler, automation is critical to the success of DevOps adoption (Enterprises, 2014).

As more and more DevOps capabilities are introduced into the organization, people and teams are effected by the change inherent from the adoption of the technologies. By integrating the tools being used by teams and creating new processes resulting from the introduction of automation, organizations now have shared goals, common tools and processes that facilitate the need for collaboration/communication and support. A new climate for learning and enhancement results from the convergence of ideas, essentially eliminating the gaps between teams.

Ultimately the importance of DevOps is related to enabling capabilities that facilitate the ability to quickly identify and execute initiatives that allow organizations to respond quickly to opportunities and threats in the business environment. Agility enables adaptation and response and is increasingly linked to organizational success in today's competitive environment (Harraf, Wanasika, Tate, & Talbot, 2015). Through literature review and the research of Sharifi and Zhang (1999), an agile organization must have four capabilities in order to make appropriate response to change in the business environment. Sharifi and Zhang (1999) and Nikpour et al., (2015) define these capabilities as follows: Responsiveness is sensing, perceiving and anticipating changes. Competency is the knowledge capabilities that enable productivity, efficiency and effectiveness. Flexibility is the ability to process different products and achieve different objectives with the same facilities. Quickness is the ability to perform a tasks in the shortest possible time.

Figure 3 represents a revised research model from data analysis of the pilot study performed in our initial research pilot. The revised model results from pilot participant feedback, statistical results from the pilot and additional literature review. The overall rationale for the model is that the implementation of DevOps adoption capability effects

the organizational agility of a firm. While DevOps is a relatively new phenomenon emerging in the business environment, the analysis of the effects of its implementation has not been fully examined or validated in previous research against the business capabilities of the firm. These business capabilities represent the routines or practices that complement each other in order to deliver value to a firm (Aral & Weill, 2007). The research model from the pilot study was revised and is presented in Figure 3 below. It presents the business capabilities and the detailed indicators that will be utilized to measure each aspect of the business capabilities during the research study. Appendix B, Table B1 summarizes the definitions and references for the constructs and subconstructs and the indicators associated with each subconstruct.

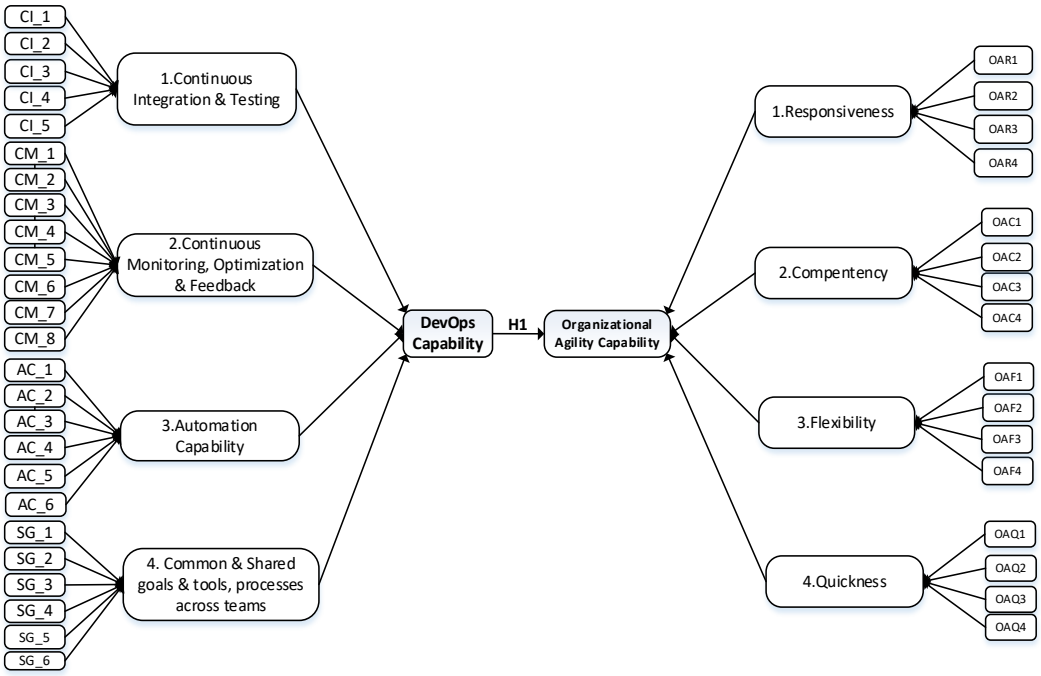


Figure 3. Revised Research Model.

Data Collection

We refined the web-based survey instrument that was developed during the pilot study by incorporating feedback from pilot participants, subject matter experts and from the organizational agility profiler survey (Worley, Williams, & Lawler, 2014). Survey measures were revised based on literature review and pilot study results. Table A1 in Appendix A summarizes the definitions and references for the constructs and subconstructs along with the measures of each subconstruct. The complete questions used for measuring each construct are included in Appendix C. The DevOps portion of the survey, focused on the four key DevOps adoption capabilities that emerged from the pilot survey, feedback from survey participants and through additional literature review. In addition, the organizational agility section of the survey incorporated the dimensions of organizational

agility. Survey questions were measured based on a 5-point Likert scale, anchored on 1=Strongly Disagree to 5=Strongly Agree.

An invitation to participate in the study was sourced from one primary source, excluding the professional message board groups that were utilized during the pilot study. The survey was delivered to respondents who attended a Federal DevOps Summit held in 2018 in Washington, D.C. For those in attendance, a paper survey was provided. Surveys were collected at the end of each workshop sessions and data was analyzed for missing data, suspicious or inconsistent response patterns, and outliers based on the processes outlined by Hair et al., (2017). While PLS is a nonparametric statistical method and does not require data to be normally distributed according to Hari et al., (2017), the data distribution was analyzed to validate skewness and kurtosis to ensure that variable distributions were as close to normal as possible. The results found that the data represented adequate levels of skewness and kurtosis with over 80% of the data being within the recommended +1/-1 ranges (Hari et al., 2010). Next, the surveys were manually coded for data analysis in SmartPLS.

After removing incomplete survey responses, there were 333 completed surveys. All of the 333 completed surveys were from participants who attended a DevOps Summit in Washington, DC in 2018. Descriptive statistics and frequencies were analyzed to understand the population of respondents, to understand data trends and to identify any additional data anomalies.

A descriptive analysis was performed on multiple demographics such as respondent's years of experience in their current role, in technology, and in working with agile technologies. In addition, descriptive analysis was performed on the industry and

organization and organization size. A wide variety of organizations and industries were represented in the responses along with a varied breath of experience in technology. The descriptive statistics can be found in Appendix A in Tables A1, A2, A3 and A4.

A diverse set of industries and roles were reported in the survey results with 68% of the respondents coming from the tow three industries of government (24%), information technology (22.5%) and consulting services (21.9%). The remainder, roughly evenly distributed among financial services, telecommunications, healthcare, manufacturing, utilities, hospitality and education. The top roles represented included development engineering (31.2%), consultant (30.3%) and operations/infrastructure (11.1%). The remainder cover the testing, networking, operations, security, product management, sales and marketing. Nearly half of the respondents (47.4%) reported that they were from organizations of ten thousand or more people.

The mean rankings and standard deviations for the lower order indicators of each construct can be found in Appendix A Table A4. Shared goals and common tools and process was viewed as the most agreed by respondents of the DevOps adoption capabilities. Likewise, competency was the most agreed by respondents of the organizational agility capabilities.

Data Validation and Analysis

Measurement Validation

The research model was validated using partial least squares (PLS) using SmartPLS. This method was selected because our model includes eight formative first-order and two formative second-order constructs which adds a level of complexity that is appropriate for PLS analysis. PLS is well suited for this particular type of study as it has been used in prior literature to assess organizational capability as it relates to technology impacts (Chin, Wang, Henseler, & Vinzi, 2010). PLS was also utilized to explore the formative variables to determine if there were any interpretations that could be concluded based on the weights of relative importance of each capability. As shown in Figure 3, the revised research model represents a hierarchical component model (HCM) with measures that are all multi-dimensional, second order formative-first order formative (Hair et al., 2017), described in Appendix B Table B1. The constructs are operationalized at a higher level of abstraction as shown in the conceptual model of Figure 2. To evaluate our formative measurement model, we followed the formative measurement procedure recommended by Hair et al., (2017). First, we assessed the model for convergent validity, next we assessed for collinearity, and finally we assessed the significance and relevance of the formative indicators and validated our original hypotheses.

With large numbers of formative indicators used to measure a single construct, it becomes more likely that one or more indicators will result in having low or nonsignificant outer weights (Hair et al., 2017). To deal with the potential impact, Cenfetelli and Bassellier (2009) propose grouping indicators into two or more distinct constructs. To accomplish this, we first developed a formative-formative hierarchical component model

(Becker et al., 2012; Kuppelwieser & Sarstedt, 2014; Ringle et al., 2012). The higher-order component, DevOps capability and organizational agility capability was then formed by the formatively measured lower order components (for DevOps capability) continuous integration & testing, continuous monitoring optimization & feedback, automation capability, and shared goals & common tools & processes and lower order components (for organizational agility capability) quickness, flexibility, competency and responsiveness. The resulting weights from the consistent PLS algorithm for both DevOps adoption capability and organizational agility capability are shown in Appendices E and F.

We then followed an approach recommended by Bagozzi and Fornell (1982) and Wu, Straub, & Liang (2015), where we multiplied the survey item values by their individual PLS weights from the PLSc algorithm and summed them for each of the first-order indicators for both organizational agility and DevOps capability. The second-order latent variables (DevOps adoption capability and organizational agility capability) were then measured by creating composite indices based on a weighted sum of the first-order indicators (Diamantopoulos and Winklhofer, 2001; Wu, Straub, & Liang, 2015). The generated composite index values were used as the measures for DevOps adoption capability and organizational agility capability.

Initially, theoretical reasoning was employed to determine that the constructs were all formative. According to Hair et al., 2017, an important characteristic of formative indicators is that they are not interchangeable, therefore; there should be very little overlap, and there should be a distribution that maximizes the amount of variance in the latent factor. To assess statistically whether the constructs should be measured formatively, the Variance Inflation Factor (VIF) which is a measure of the level of collinearity, was examined to

determine whether the formative measures are correlated too highly. The VIF values of our conceptual model resulted in outer VIF values for the organizational agility construct, specifically the measures of flexibility (OAF_Sum VIF=6.146) and quickness (OAQ_Sum VIF=5.268), to exceed the recommended threshold of 3.3. According to Petter, Straub, & Rai, 2007, general statistics theory suggests that multicollinearity is a concern if the VIF is higher than 10; however with formative measures, multicollinearity poses more of a problem and therefore; the VIF for formative measures should not be greater than 3.3.

In this instance where multicollinearity exists, Diamantopoulos & Siguaw (2006) suggest that construct validity and/or reliability be assessed utilizing redundancy analysis to test for convergent validity. This involves modeling the organizational agility construct as having both formative and reflective measurement items in order to test whether the formatively measured construct is highly correlated with a reflective measure of the same construct. As recommended by Hair (2017), the strength of the path coefficient between the formative and reflect measures should ideally be a magnitude of 0.80 with a minimum of 0.70 and above and the R^2 should be at least 0.50. In evaluating the measures of the organizational agility construct, one formative and one reflective, the path coefficient linking the two constructs was 0.697 and the R^2 was 0.486 this indicates that the formative indicators (with high multicollinearity) have the minimum convergent validity and contribute a sufficient degree to the measure of the organizational agility capability construct. Therefore, these indicators were kept in the study for further analysis.

Next, to ensure that any nonsignificant effects would be interpretable in the research model, the statistical power of the sample for the dependent variable organizational agility capability was calculated. Based on the number of predictors, observed R^2 of 0.486, at a

significance of 0.05, and sample size of n=333, the observed power was 0.99, which exceeded the recommended guideline of 0.80 (Cohen, 1988). This indicates that our sample has adequate statistical power in order to further test our research model and validate our hypotheses.

The results in Figure 3 indicate that our weights and paths are significant at the 0.5 significance level and that the model explains 69.7 percent of the variance in the organizational agility capability. The significance of the paths and relevance of the formative indicators were also examined and given that the values of the outer weights are standardized they can therefore be compared to each other (Hair, 2017). They express each indicators relative importance to forming the DevOps adoption and organizational agility constructs. The standardized path coefficient from DevOps adoption capability to organizational agility capability is 0.697 and the t-value is 25.238, at the 0.05 significance level, indicating that the effects of DevOps adoption capability on organizational agility capability are significant. Figure 4 shows graphically the results of the tests of the structural model.

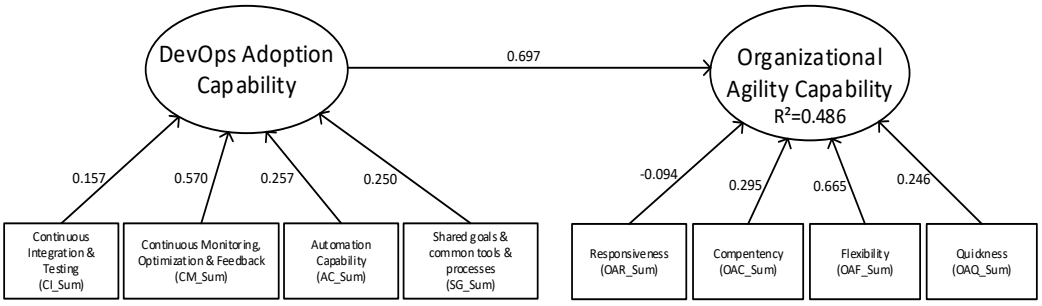


Figure 4. Results of Path Analysis

To assess how well the data supported the hypothesized model, a confirmatory tetrad analysis (CTA) was performed to statistically evaluate the specification of indicators in the measurement model. The CTA facilitates empirical evaluation of cause-effect relationships for latent variables (whether they are correctly modeled as reflective or formative) and their specification in the measurement model (Gudergan et al., 2008). The CTA provides results for each construct that has a minimum of at least four indicators. Per Gudergan et al., (2008), a significance test should be performed on each vanishing tetrad. A tetrad simply describes the relationship between pairs of covariances (Hair et al., 2017). A significance test with an alpha level of 0.05 was performed on the results of the CTA. Since each of our constructs contained the minimum number of four indicators each, the CTA test provided validation of our measurement specification. According to Gudergan et al., (2008), for organizational agility capability, we found that there was a significant relationship in both tetrads (p=0.025 and p=0.013), indicating that the measure should be modeled formatively. For DevOps adoption capability, we found that one tetrad had a significant relationship (p=0.000) and the other tetrad did not have a significant relationship (p=0.191), and therefore; confirmed our theoretical reasoning that the measurement is modeled formatively, however; the nonsignificant relationship indicates that the DevOps adoption capability measures could also be modeled relatively. For the purposes of this research study, as recommended by Hair et al., (2017) and Gudergan et al., (2007), we relied on the primary means of theoretical reasoning to decide to treat each of the DevOps adoption measures as formative.

To further assess how well the data supported the hypothesized model, an evaluation of significance, path coefficients and weights were assessed. H1 tests the effect

that DevOps adoption capability has on organizational agility capability. We evaluated the effect that continuous integration and testing, continuous monitoring, optimization and feedback, automation capability, and shared goals had on DevOps adoption capability. Likewise, we evaluated the effect that responsiveness, competency, flexibility and quickness had on organizational agility capability. Indicators were significant ($p < 0.05$) with the exception of continuous integration and testing, responsiveness, and quickness.

In addition, the weights of these three indicators also supported the non-significant findings, respectively, with weights (0.157, -0.094, and 0.246) being low for each. This indicates a possible weak relationship with DevOps adoption capability and weak effect on organizational agility capability. Given the low weights and lack of significance, additional analysis of the outer loadings of these indicators was warranted in order to consider the absolute importance to its corresponding construct.

The absolute contribution is given by the outer loading (Hair et al., 2017). The outer loading for continuous integration is 0.779, for responsiveness, 0.822, and for quickness it is 0.925. According to Hair et al., (2017), when an indicator's outer weight is nonsignificant but its outer loading is high (i.e. above 0.50), the indicator should be interpreted as absolutely important but not as relatively important. Based on this finding, continuous integration is absolutely important in effecting DevOps adoption capability, but not likely to be more effective than the other indicators with higher weights and loadings. Similarly, based on the weights, quickness and responsiveness are absolutely important in causing organizational agility, but not likely to more of the cause than the other indicators in the research model.

To assess the significance of the effect of DevOps adoption capability (independent variable) on organizational agility capability (dependent variable), we examined the path coefficient and the standard errors of the direct path between the constructs. The results can be found in Appendix A, Table A5, however; a summary follows. DevOps adoption capability had the strongest path coefficient (0.697) indicating a positive direct relationship to organizational agility capability. While the relationship is significant, it is also important to assess the relevance of significant relationships (Hair et al., 2017). The total effect of DevOps adoption capability in explaining organizational agility capability was assessed at 0.697 and the effect size was assessed at f^2=0.945 indicating that DevOps adoption capability has a significant effect on organizational agility capability. H1 states that DevOps adoption capability has an effect on organizational agility; this hypothesis is supported, therefore; we reject the null hypothesis (H1: b=0.697, p=0.000).

In addition to testing the main hypothesis, we further evaluated the strength of the indicators relative contribution to causing DevOps adoption capability and the strength of the indicators effect on organizational agility capability. Based on the analysis of the weights of continuous monitoring, optimization and feedback, automation capability, and shared goals, common tools, and processes (weights of 0.570, 0.257, and 0.250), we conclude that these indicators contribute more to the cause of DevOps adoption capability than continuous integration and testing (weight of 0.157). This finding is consistent with DevOps practitioner literature that supports the importance of these findings in the business enterprise (Enterprises, 2014, Puppet Labs, 2016).

Next, responsiveness and quickness, respectively; with weights and loadings of -0.094 and 0.822 (p=0.390) and 0.246 and 0.925 (p=0.052) indicate that their effect on

organizational agility capability is less than that of competency and flexibility with weights and loadings of 0.295 and 0.722 (p=0.006) and 0.665 and 0.958 (p<0.001), respectively. Given that these indicators were nonsignificant (p>0.05), we also evaluated the loadings. Hair, et al., (2017) recommended that when an indicator's outer weight is nonsignificant but its outer loading is high (above 0.50), the indicator should be interpreted as absolutely important but not as relatively important. We therefore conclude that even though responsiveness and quickness have less of an effect on organizational agility, based on their high loadings, their effect on the construct is absolutely important, but not relatively important. Comparatively, flexibility had the highest weight (0.665) and competency had the second highest weight (0.295), indicating that they contribute more of an effect on organizational agility capability than quickness and responsiveness.

Finally, in order to assess the strength of the relationship between our model and organizational agility, the R^2 measures of the model's in-sample predictive power was analyzed. In scholarly research, R^2 values of 0.75, 0.50, or 0.25, as a rule of thumb, are respectively described as substantial, moderate, or weak (Hair et al., 2011; Henseler et al., 2009). The level of predictive accuracy of our research model was moderate at 0.486. This indicates that our model does a relatively good job of explaining the changes in organizational agility.

Discussion, Implications and Limitations of Research

Overall, we find strong evidence, that DevOps adoption capability enables organizational agility capability. We developed and tested a theoretical model, and discovered that the three most important factors associated with DevOps adoption capability were shared goals, automation, and continuous monitoring, and the three most important indicators of organizational agility were quickness, competency, and flexibility. This finding is consistent with findings in practitioner data that suggest DevOps adoption leads to increased organizational agility, but is to our knowledge the first study to provide a theoretical rationale and empirical support for this relationship.

In addition, this study provides deeper insight into the mechanism behind DevOps adoption's effect on organization agility. Specifically, our findings suggest a strong relationship between continuous monitoring and flexibility. This finding has substantive implications for organizations implementing continuous monitoring practices in that the model indicates that this leads to greater flexibility of the organization. This significant effect of continuous monitoring on flexibility further corroborates what is seen in the practitioner literature in that continuous monitoring enables flexibility by resulting in shorter development cycles, accelerated deployment frequencies and frequent product release resulting in greater competitive flexibility for the business environment (Puppet, 2016).

While data for this study built on the lessons learned in the pilot study, there are limitations that should be considered. First, the research study relied on a single source sample, which possibly represented a population of respondents that may not represent the broader audience of the DevOps community. In addition, this study relied on the

experiences of respondents who attended a DevOps focused conference and therefore; excluded the experiences of those who were not in attendance of the conference. While utilizing respondents from this DevOps focused conference provided a means to gather the perspectives of those familiar with the concepts of the study, limiting the variability in our respondents, it does raise the question of generalizability. Future research should survey other populations to verify our results hold across different audiences. Also, common method variance could be present in the data because our single survey instrument collected data for both the independent and dependent variables (DevOps capability and organizational agility). Subsequent research should consider this limitation and plan to utilize strategies suggested by Podsakoff et al., (2003), such as the inclusion of a marker variable, to overcome concerns related to biases which may arise.

A second area for future research is to expand the scope of the study beyond DevOps adoption. Future research could focus on studying the effects of other organizational capabilities and consider moderating effects, such as cultural maturity and gender. Also, while an in-depth literature review was conducted and theoretical reasoning applied, given the results of the CTA analysis future research should focus on examining the effects of reflective measures in order to understand the differences in effects on the capabilities.

Observations resulting from this research reveal implications of the research findings and theoretical contributions to practice. First, the theoretical significance and importance of establishing a definition of DevOps provids the foundation for studying the effects of DevOps adoption capability on organizational agility. This foundation can be extended to study other organizational capabilities. The results also present empirical

evidence of the significance of the relationship between Devops adoption and organizational agility not established in prior academic literature. This relationship is important to practice because it establishes the foundation for which future DevOps adoptions should follow in terms of prioritizing implementation of capabilities that impact organizations and their quest to improve agility and impact performance outcomes. These implications should continue to be investigated in future research. Specifically, as recommended based on the CTA analysis, observations also support that future research focus on analyzing the effect of reflective measures of organizational agility and how they might change the importance of the capabilities and their relative relationships between DevOps adoption and organizational agility capability. In addition, additional formative measures of DevOps adoption capability might also offer comparative insights into the causes of DevOps adoption capability leading to successful adoption results and increased performance outcomes. Additional analysis of the formative measures developed in this study should be verified in future research with multiple indicator multiple causes (MIMIC) models recommended by Jarvis et al., (2003) and Kim et al., (2010).

Finally, this study also focused on using only quantitative data to support the hypothesized relationship. Future research should focus on utilizing both quantitative and qualitative data and other study methods to provide deeper insights into the effects that DevOps might have on organizational agility.

CHAPTER 4

CONCLUSION

As DevOps has emerged as an important capability in the competitive business environment, the ability to respond quickly and effectively to changes that are occurring in the environment is a necessity that separates the most successful organizations from those that struggle to improve performance outcomes. The objective of this research was to explore the answers to two primary research questions. The first research question set out to develop an empirical definition of DevOps. Through consolidation of knowledge gathered from exhaustive synthesis of existing practitioner and academic literature and pilot research, an empirical definition of DevOps was established. Next, we set out to answer the second research question – what are the key capabilities of DevOps adoption and what is the effect on organizational agility? To answer the second research question, an exhaustive literature review was performed, resulting in the emergence of seventeen characteristics critical to DevOps adoption and four characteristics of organizational agilitty. The seventeen capabilities were then validated and consolidated through a pilot research study. The top four key capabilities critical to DevOps adoption were then leveraged along with the four key capabilities of organizational agility to measure the effects of DevOps adoption on organizational agility and to study the relative importance of each capability on their corresponding construct.

The results of our study established that DevOps adoption capability has a positive effect on organizational agility, meaning the greater DevOps adoption capabilities of an organization, the greater effect it has on organizational agility. Sharifi and Zhang (1999) established that agility is a vital ability in the turning of the business environment.

Organizations today cannot ignore the implications of agility and we now have empirical evidence of the impacts that DevOps capability has on organizational agility. Our results support why so many C-suite executives are driving their teams towards adopting DevOps methods.

The predictive relationships of the indicators that effect both DevOps adoption and organizational agility were also analyzed. Our study validated that of the nine hypothesis posited, six were supported and three were not supported. However, relative to the three hypotheses that were not supported, the three associated indicators were also determined to not be as relatively important as the other indicators, but were found to be absolutely important to defining its construct. Additionally, our study found that continuous monitoring and feedback leads to greater flexibility. This finding is consistent with evidence seen in practice relative to a firm's ability to produce high levels of quality products in a more flexible, agile manner (Puppet, 2016). This finding has significant impacts on practice given that it establishes the foundation for future DevOps adoptions in terms of prioritizing adoption capabilities that lead to greater agility within the firm. Similarly, in analyzing additional output of our study, the validity of the content of continuous integration, reliability and quickness were deemed to be relevant and important and strongly supported the theory driven conceptualization of the significance of the relationship between DevOps adoption and organizational agility. Overall, our research theoretically proposes and empirically validates the research model put forth that establishes the positive effect that DevOps adoption has on organizational agility. In addition, our research successfully validates the strength of the relationships between the

key capabilities of DevOps adoption and organizational agility, establishing a foundation that can be used to assist with prioritizing adoption capabilities.

BIBLIOGRAPHY

Aghina, W., DeSmet, A., & Weerda, K. (2015). Agility It rhymes with stability. *McKinsey Quarterly*.

AgileAlliance. (1990). Agile Manifesto For Software Development. Retrieved from https://www.agilealliance.org/

Albert W. Small, P. D., & Downey, E. A. (1996). Orchestrating Multiple Changes: a Framework for Mianaging Concurrent Changes of Variled Type and Scope. *IEEE*.

Allspaw, J., & Hammond, P. (2009, June 23, 2009). 10+Deploys Per Day: Dev and Ops Cooperation at Flickr. Retrieved from http://velocityconf.com/velocity2009/public/schedule/detail/7641

Allwright, D. (2005). Developing Principles for Practitioner Research: The Case of Exploratory Practice. *The Modern Language Journal, 89*(3).

Aoyama, M. (1998). Web-based agile software development. *IEEE Software, 15*(6), 55-65.

Aral, S., & Weill, P. (2007). IT Assets, Organizational Capabilities, and Firm Performance: How Resource Allocations and Organizational Differences Explain Performance Variation. *Organization Science, 18*(5), 763-780.

Bagozzi, R. P., & Fornell, C. (1982). *Theoretical Concepts, Measurements and Meaning* (C. Fornell Ed. Vol. 2). New York, NY.

Bahrami, H. (1992). The Emerging Flexible Organization: Perspectives from the Silicon Valley. *California Management Review, 34*(4), 33-52.

Barker, C. (2013). DevOps offers big programming productivity gains, says report. *Enterprise Software.* Retrieved from http://www.zdnet.com/article/devops-offers-big-programming-productivity-gains-says-report/

Barney, J. B. (1986). Organizational Culture Can It Be a Source of Sustained Competitive Advantage? *The Academy of Management Review, 11*(3).

Becker, J.-M., Klein, K., & Wetzels, M. (2012). *Hierarchical Latent Variable Models in PLS-SEM: Guidelines fro Using Reflective-Formative Type Models* (Vol. 45): Long Range Planning.

Berson, Y., Oreg, S., & Dvir, T. (2008). CEO values, organizational culture and firm outcomes. *Journal of Organizational Behavior, 29*(5), 615-633. doi:10.1002/job.499

Betteley, J. (2015, May 11, 2015). What does DevOps have to do with Agile. *Dzone/Agile Zone.* Retrieved from https://dzone.com/articles/what-does-devops-have-do-agile

Börjesson, A., Martinsson, F., & Timmerås, M. (2006). Agile improvement practices in software organizations. *European Journal of Information Systems, 15*(2), 169-182. doi:10.1057/palgrave.ejis.3000603

Bradley, T. (2015, March 4, 2015). DevOps is Agile for the Rest of the Company. Retrieved from http://devops.com/2015/03/04/devops-is-agile-for-the-rest-of-the-company/

Buchanan, I. (2015). Agile and DevOps: Friends or Foes? Retrieved from https://www.atlassian.com/agile/devops

Cancialosi, C. (2016, May 23, 2016). DevOps and Culture: The Evolution of DevOps in the Tech Industry. *Forbes Entrepreneurs*.

Caswell, N. S., & Nigam, A. (2005). Agility Equals Change plus Coordination. *IEEE International Conference of E-Commerce Technology Workshops*, 131-139.

Cenfetelli, R. T., & Bassellier, G. (2009). Interpretation of Formative Measurement In Information Systems Research. *MIS Quarterly, 33*(4), 689-707.

Chin, W. W., Wang, H., Henseler, J., & Vinzi, V. E. (2010). *How to Write Up and Report PLS Analyses*. London.

Coderre, D., & Verschoor, C. C. (2006). Continuous Auditing: Implications for assurance, monitoring, and risk assessment (Global Technology Audit Guide) No. 3. *The Institute of Internal Auditors*, *21*(2), 44.

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). Hillsdale, NJ: Lawrence Erlbaum Associates.

Day, G. S. (1994). The Capabilities of Market-Driven Organizations. *Journal of Marketing*, *58*(4), 37-52.

de França, B. B. N., Jeronimo, H., & Travassos, G. H. (2016). Characterizing DevOps by Hearing Multiple Voices. 53-62. doi:10.1145/2973839.2973845

Debois, P., Humble, J., Molesky, J., Shamow, E., Fitzpatrick, L., Dillon, M., . . . DeGrandis, D. (2011). The Business Case for DevOps. *The Journal of Information Technology Management*, *24*(8).

Diamantopoulos, A., & Siguaw, J. A. (2006). Formative Versus Reflective Indicators in Organizational Measure Development: A Comparison and Empirical Illustration. *British Journal of Management*, *17*(4), 263-282.

Diamantopoulos, A., & Winklhofer, H. (2001). Index construction with formative indicators: An alternative to scale development. *Journal Of Marketing Research*, *38*(2), 269.

Dyck, A., Penners, R., & Lichter, H. (2015). *Towards Definitions for release engineering and devops*. Unpublished Practitioner Article. RWTH Aachen University.

Edwards, D. (2010, February 23, 2010). What is DevOps? Retrieved from http://dev2ops.org/?s=wall+of+confusion

Eisenhardt, K. M., & Martin, J. A. (2000). Dynamic Capabilities What are they. *Strategic Management Journal: Special Issue The Evolution of Firm Capabilities, 21*(10/11), 1105-1121.

Enterprise, H., & Arevalo, H. (2015). Defining the New Style of IT. Retrieved from https://community.hpe.com/t5/Infrastructure-Insights/Defining-the-New-Style-of-IT/ba-p/6711400#.WPPxkfnyuM8

Enterprises, H. (2014-2015). *Four Keys to your DevOps Journey*. Retrieved from https://www.hpe.com/h20195/V2/GetPDF.aspx/4AA5-5748ENW.pdf

Enterprises, H. (2015). *Through the DevOps Looking Glass: Learnings from HPs Own Transformation Initiative*. Retrieved from

Enterprises, H. (May 2016). *Applicatin Security and DevOps*. Retrieved from https://saas.hpe.com/en-us/sites/default/files/resources/files/HPE%20Application%20Security%20and%20DevOps%20Report.pdf

Freeform Dynamics, C. T. (2015). *Assembling the DevOps Jigsaw*. Retrieved from

Gefen, D., Straub, D., & Boudreau, M.-C. (2000). Structural Equation Modeling and Regression: Guidelines for Research Practice. *Communications of the Association for Information Systems, 4*, 1-77.

Golden, W., & Powell, P. (2000). Towards a Definition of Flexibility: In Search of a Holy Grail. *Omega: The International Journal of Management Science, 28*(4), 373-384.

Gregory, B. T., Harris, S. G., Armenakis, A. A., & Shook, C. L. (2009). Organizational culture and effectiveness: A study of values, attitudes, and organizational outcomes. *Journal of Business Research, 62*(7), 673-679. doi:10.1016/j.jbusres.2008.05.021

Gudergan, S. P., Ringle, C. M., Wende, S., & Will, A. (2008). Confirmatory tetrad analysis in PLS path modeling. *Journal of Business Research, 61*(12), 1238-1249. doi:10.1016/j.jbusres.2008.01.012

Hair, J., Hult, T., Ringle, C., & Sarstedt, M. (2017). *A Primer on Partial Least Squares Structural Equation Modeling (PLS-SEM)*. Los Angeles, CA: Sage.

Hand, J. (2015). The IT Culture War: The Struggle to Adopt DevOps. Retrieved from https://www.wired.com/insights/2015/03/culture-war-struggle-adopt-devops/

Harraf, A., Wanasika, I., Tate, K., & Talbott, K. (2015). Organizational Agility. *The Journal of Applied Business Research, 31*(2).

Henseler, J., Ringle, C. M., & Sinkovics, R. R. (2009). *The use of partial least squares path modeling in international marketing* (Vol. 20): Emerald Group Publishing Limited.

Hinchcliffe, D. (2014). *How Social Business Improves DevOps*. Retrieved from Computer Associates: https://www.ca.com/us/rewrite/articles/devops/how-social-business-improves-devops.html

Homburg, C., & Pflesser, C. (2000). A Multiple-Layer Model of Market-Oriented Organizational Culture Measurement Issues and Performance Outcomes. *Journal Of Marketing Research*, *37*(4), 449-462.

Horlach, B., Drews, P., & Schirmer, I. (2016). *Bimodal IT: Business-IT alignment in the age of digital transformation*. Paper presented at the MKWI 2016.

Horney, N., & O'Shea, T. (2015). *Focused, Fast and Flexible: Creating Agility Advantage in a VUCA World*. Oceanside, CA: Indie Books International LLC.

HPE. (2013, April 26, 2013). HPE DevOps Maturity Model. Retrieved from https://community.hpe.com/t5/IT-Operations-Management-ITOM/DevOps-and-OpsDev-How-Maturity-Model-Works/ba-p/6042901

Humble, J., & Molesky, J. (2011). Why Enterprises Must Adopt Devops to Enable Continuous Delivery. *Cutter IT Journal*, *24*(8), 6.

Huttermann, M. (2012). *DevOps for Developers*: Apress.

Infosys. (2016). *DevOps*. Retrieved from

ISACA. (2016). DevOps Overview. Retrieved from http://www.isaca.org/knowledge-center/research/researchdeliverables/pages/devops-overview.aspx

James Sena, Coget, J.-F., & Shani, A. B. R. (2009). Designing for Agility as an Organizational Capability: Learning from a Software Development Firm. *The International Journal of Knowledge, Culture & Change Management*, *9*(5).

Jarvis, C. B., Mackenzie, S. B., and Podsakoff, P. M. 2003. "A Critical Review of Construct Indicators and Measurement Model Misspecification in Marketing and Consumer Research, " *Journal of Consumer Research*, (30:2), pp. 199-218.

Kerzazi, N., & Adams, B. (2016, May 14-15 2016). *Who needs release and devops engineers, and why?* Paper presented at the Continuous Software Evolution and Delivery, Austin, TX.

Kidd, P. T. (1994). *Agile Manufacturing--Forging New Frontiers*. 1 Jacob Way, Reading, MA 01867: Addison-Wesley Publishing Company.

Kim, G., Behr, K., & Spafford, G. (2013). *The Phoenix Project: A Novel About IT, DevOps and Helping Your Business Win*: IT Revolution Press.

Kim, G., Humble, J., Debois, P., & Willis, J. (2016). *The DevOps Handbook: How to create world class agility, reliability and secrity in technology organizations*. Portland, OR: IT Revolution Press.

Kim, G., Shin, B., and Grover, V. 2010. "Investigating Two Contradictory Views of Formative Measurement in Information Systems Research," *MIS Quarterly* (34:2), pp.345-365.

Kompella, L. (2014). Agile methods, organizational culture and agility: some insights. *Association of Computing Machinery*, 40-47. doi:10.1145/2593702.2593708

Kuppelwieser, V., & Sarstedt, M. (2014). Applying the future time perspective scale to advertising research. *International Journal of Advertising, 33*(1), 113-136.

Lesser, E., & Ban, L. (2016). How leading companies practice software development and delivery to achieve a competitive edge. *Strategy & Leadership, 44*(1), 41-47. doi:10.1108/sl-11-2015-0083

Lu, Y., & Ramamurthy, K. R. (2011). The Link Between IT Capability & Organizational Agility. *MIS Quarterly, 35*(4), 931-954.

Lwakatare, L. E., Kuvaja, P., & Oivo, M. (2015). Dimensions of DevOps Presentation.

Lwakatare, L. E., Kuvaja, P., & Oivo, M. (2016). *An Exploratory Study of DevOps Extending the Dimensions of DevOps with Practices.* Paper presented at the ICSEA 2016 The Eleventh International Conference on Software Engineering Advances.

Lycett, M., Macredie, R. D., Patel, C., & Paul, R. J. (2003). Migrating agile methods to Standardize Development Practice. *IEEE, 36*(6), 79-85. doi:10.1109/MC.2003.1204379

Madhavaram, S., & Hunt, S. D. (2007). The service-dominant logic and a hierarchy of operant resources: developing masterful operant resources and implications for marketing strategy. *Academy of Marketing Science, 36*, 67-82. doi:10.1002/smj.64()

Minick, E. (2015). Surprise! Broad Agreement of the Definition of DevOps. Retrieved from http://devops.com/2015/05/13/surprise-broad-agreement-on-the-definition-of-devops/

Mohamed, S. I. (2016). DevOps Maturity Calculator DOMC - Value oriented approach. *International Journal of Engineering Research & Science, 2*(2).

Mohrman, S., Docherty, P., Shani, A. B. R., Teigland, R., & Schenkel, A. J. (August 2006). *The Development of New Organizational capabilities: A design-based model for managerial action*. Paper presented at the Academy of Management Annual Conference, Atlanta, Georgia.

Nagel, R. N. (1992). 21st Century Manufacturing Enterprise Strategy Report.

Panda, S., & Rath, S. K. (2017). The effect of human IT capability on organizational agility: an empirical analysis. *Management Research Review, 40*(7), 800-820. doi:10.1108/mrr-07-2016-0172

Pankaj. (2004). *An Analysis and Exploration of the Construct Of Information Systems Agility*. Unpublished Practitioner Article.

Petter, S., Straub, D., & Rai, A. (2007). Specifying Formative Construts in Information Systems Research. *MIS Quaterly, 31*(4), 623-656.

Pivotto, J. (2013). *The DevOps Approach to Monitoring Open Source and IAC Style*. Paper presented at the Open World Forum, Antwerp Belgium. http://www.slideshare.net/roidelapluie/devops-andmonitoringowf13

Podsakoff, P., MacKenzie, S., Lee, J.-Y., & Podsakoff, N. (2003). Common Method Biases in Behavioral Research: A Critical Review of the Literature and Recommended Remedies. *Journal of Applied Psychology, 88*(5).

Prahalad, C. K. (2009, September 21, 2009). In Volatile Times, Agility Rules. *Business Week*, 80.

PuppetLabs. (2015, 2015). 2015 State of DevOps Report.

PuppetLabs, & DORA. (2016). 2016 State of DevOps Survey. 2016. Retrieved from https://puppet.com/blog/2016-state-of-devops-survey-here

Rana, R., & Staron, M. (2016). *First International Workshop on Emerging Trends in DevOps and Infrastructure*. Paper presented at the Association of Computing Machinery XP-2016.

RebelLabs. (2013). *IT Ops & DevOps Productivity Report 2013 - Tools, Methodologies and People*. Retrieved from

Relic, N. (2014). [Navigating DevOps: What it is and why it matters to you and your business].

Relic, N. (2016). DevOps Continuous Monitoring. Retrieved from https://newrelic.com/resource/navigating-devops

Ringle, C. M., Sarstedt, M., & Straub, D. (2012). A critical look at the use of PLS-SEM. *MIS Quarterly, 36*(1).

Roberts, N., & Grover, V. (2012). Investigating firm's customer agility and firm performance: The importance of aligning sense and respond capabilities. *Journal of Business Research, 65*(5), 579-585. doi:10.1016/j.jbusres.2011.02.009

Sambamurthy, V., Bharadwaj, A., & Grover, V. (2003). Shaping agility through digital options: Reconceptualizing the role of information technology in contemporary firms. *MIS Quarterly, 27*(2), 237-263.

Sanders, J. (2017, May 20, 2017). DevOps: The Smart Person's Guide. Retrieved from https://www.techrepublic.com/article/devops-the-smart-persons-guide/

Schorsch, P. (2015, May 18, 2015). Organizational Agility: It's More Than Technology. Retrieved from http://www.ericsson.com/spotlight/transformative-it/blog/organizational-agility-its-more-technology/

Shamow, E. (2011). Devops at Advance Internet: How we got in the Door. Retrieved from https://www.cutter.com/sites/default/files/itjournal/fulltext/2011/08/itj1108.pdf

Sharifi, H., & Zhang, Z. (1999). A methodology for achieving agility in manufacturing organisations: An introduction. *International Journal of Production Economics*(62), 7-22.

Sharifi, H., & Zhang, Z. (1999). A methodology for achieving agility in manufacturing organizations: An Introduction. *International Journal Production Economics, 62*.

Sieger, D. B., Badiru, A. B., & Milatovic, M. (2000). A metric for agility measurement in product development. *IIE Transactions, 32*(7), 637-645. doi:10.1080/07408170008967422

Smeds, J., Nybom, K., & Porres, I. (2015). *DevOps: A Definition and Perceived Adoption Impediments*. Switzerland: Springer International Publishing.

Srivastava, M., Franklin, A., & Martinette, L. (2012). Building a Sustainable Competitive Advantage. *Journal of Technology Management & Inoovation, 8*(2), 2013.

Technologies, C. (2014). *DevOps: The Worst Kept Secret to Winning in te Application Economy*. Retrieved from CA Technologies: http://www.ca.com/content/dam/rewrite/files/White-Papers/devops-winning-in-application-economy-2.pdf

Teece, D. J. (2007). Explicating dynamic capabilities: the nature and microfoundations of (sustainable) enterprise performance. *Strategic Management Journal, 28*(13), 1319-1350. doi:10.1002/smj.640

Thompson, L., & Shafer, A. (2009). Agile Infrastructure: A Story in Three Acts. *Velocity Conference 2009*.

Thomson, A. M., Perry, J. L., & Miller, T. K. (2007). Conceptualizing and Measuring Collaboration. *Journal of Public Administration Research and Theory, 19*(1), 23-56. doi:10.1093/jopart/mum036

van Oosterhout, M., Waarts, E., & van Hillegersberg, J. (2006). Change factors requiring agility and implications for IT. *European Journal of Information Systems, 15*(2), 132-145. doi:10.1057/palgrave.ejis.3000601

Walls, M. (2013). *Building a DevOps Culture*.

Waterhouse, P. (2015). *DevOps Practioner Series: Metrics That Matter Developing and Tracking Key Indicators of High Performance IT*. Retrieved from http://www3.ca.com/~/media/Files/whitepapers/devops-practitioner-series-metrics-that-matter-developing-and-tracking-key-indicators.PDF

Willis, J. (2010). DevOps Culture Part 1. Retrieved from http://itrevolution.com/devops-culture-part-1/

Willis, J. (2012). The Convergence of DevOps. Retrieved from Http://itrevolution.com/the-convergence-of-devops/

Worley, C. G., Williams, T. D., & III, E. E. L. (2014). *The Agility Factor: Building Adaptable Organizations for Superior Performance*: Jossey-Bass.

Worley, C. G., Williams, T. D., & Lawler, E. E., III. (2014). *Assessing Your Organization's Agility : Creating Diagnostic Profiles to Guide Transformation*. Somerset, UNITED STATES: John Wiley & Sons, Incorporated.

Wu, P.-J. S., Straub, D., & Liang, T.-P. (2015). How Information Technology Governance Mechanisms and Strategic Alignment Influence Organizational Performance: Insights from A Matched Survey of Business and IT Managers. *MIS Quarterly, 39*(2), 497-518.

Yauch, C. A. (2011). Measuring agility as a performance outcome. *Journal of Manufacturing Technology Management, 22*(3), 384-404. doi:10.1108/17410381111112738

Yauch, C. A., & Navaresse, D. O. (2006). Assessing agility at 70 manufacturing plants. *International Journal of Agile Manufacturing, 9*(2), 19-28.

Zaheer, A., & Zaheer, S. (1997). Catching the wave Alertness, responsiveness, and market influence in global electronic networks. *Management Science, 43*(11), 1493-1509.

APPENDIX A

DESCRIPTIVE STATISTICS

Table A1

Respondents by Industry

Industry	Number	%
Government	80	24.00%
Information Technology	75	22.50%
Consulting Services	73	21.90%
Financial Services	26	8.00%
Telecommunications	17	5.10%
Healthcare, Biotechnology	13	3.90%
Manufacturing	11	3.00%
Utilities	10	3.00%
Other	10	3.00%
Hospitality	8	2.40%
Education	6	2.00%
Entertainment, Arts, Media	2	0.60%
Retail	2	0.60%

Table A2

Respondents by Role

Role	Number	%
Development Engineering	104	31.2%
Consultant	101	30.3%
Operations or Infrastructure	37	11.1%
Quality Assurance or Test	23	6.9%
Networking	21	6.3%
DevOps	15	4.5%
Security Engineering	8	2.4%
Product Management	6	1.8%
Other	6	1.8%
Student	5	1.5%
Sales or Marketing	3	0.9%
C-Level Executive	2	0.6%
Release Management	2	0.6%

Table A3

Organization Size

Size	Number	%
1 – 49	26	7.8%
50 – 999	63	18.9%
1,000 - 4,999	46	13.8%
5,000 - 9,999	35	10.5%
10,000 - or more	158	47.4%
I don't know	5	1.5%

Table A4

Mean rankings of capabilities by degree of agreement

Capability	Mean	Deviation
DevOps Adoption Capability		
Shared goals & common tools & process	4.11	0.50
Continuous Monitoring Optimization & Feedback	3.94	0.61
Automation Capability	3.80	0.60
Continuous Integration & Testing	3.70	0.65
Organizational Agility Capability		
Competency	3.87	0.51
Responsiveness	3.60	0.62
Quickness	3.40	0.84
Flexibility	3.35	0.94

Table A5

Model Testing Results

		p-value	*t*-value	Outer Weights	Outer Loading	alpha	Path Coefficient
H1	DOC-OAC	0.000	25.238	------	------	0.05	0.697
	CM	0.000	5.998	0.570	0.909	0.05	
	AC	0.000	4.113	0.257	0.702	0.05	
	SG	0.012	2.518	0.250	0.720	0.05	
	CI	0.158	1.414	0.157	0.779	0.05	
	OAF	0.000	4.555	0.665	0.958	0.05	
	OAC	0.006	2.73	0.295	0.722	0.05	
	OAQ	0.052	1.948	0.246	0.925	0.05	
	OAR	0.390	0.859	-0.094	0.822	0.05	
	N=333 R^2=0.486 Observed power = 0.999						

APPENDIX B

MEASUREMENT VALIDATION

Table B1

Construct Definitions and Measures

Construct	Definition	Type	Items	Source or Basis
DevOps Adoption Capability: a convergence of ideas, a set of capabilities and practices, methods and principals that aim to eliminate the gaps between teams. DevOps results in the combination of the right people, with the right attitude, and the right skills, who execute processes, utilize tools and methodologies that enable a culture of cross-functional collaboration and communication in order to develop and operate resilient systems and accelerate delivery of changes.		Formative 2nd Order		Burrell 2018
Continuous integration & testing	A technique that continually merges source code updates from all developers on a team into a shared mainline. A macro process that yields executable releases that grow in functionality at every release.	Formative 1st Order	CI1, CI2, CI3, CI4, CI5	NewRelic 2014 Booch 1995
Continuous Monitoring & Feedback	Instrumenting application and aggregating monitored data into insights. A pattern of a continuous feedback loop that runs from the production environment to the start of the development cycle, including a complete timeline of development and operations events. It involves proactive detection and awareness of events in critical environments, such as test and production, in order to expose (know the state of) issues before they cause failures.	Formative-1st Order	CM1, CM2, CM3, CM4, CM5, CM6, CM7, CM8	Lwakatare, Kuvaja, & Oivo 2016
Automation Capability	The ability of firm to utilize technology to automate manually and repetitive tasks in order to reduce unnecessary effort and improve software delivery	Formative-1st Order	AC1, AC2, AC3, AC4, AC5, AC6	Hedemark 2014 Puppet Labs 2014

Shared goals & common tools and processes	The approach and strategy of utilizing common tools and processes that adhere to a common set of shared goals that facilitate seamless collaboration and integration between teams.	Formative- 1st Order	SG1, SG2, SG3, SG4, SG5, SG6	Linthicum 2015 Puppet Labs 2015
Organizational Agility: The successful exploitation of competitive bases (speed, flexibility, innovation proactivity, quality and profitability) through the integration of reconfigurable resources and best practices in a knowledge-rich environment to provide customer-driven products and services in a fast changing market environment.		Formative- 2nd Order		Yusef, Sarhadi, & Gunasekaran, 1999
Responsiveness	The ability to identify changes and respond fast to them, reactively or proactively, and recover from them.	Formative- 1st Order	OAR1, OAR2, OAR3, OAR4	(H. Sharifi & Z. Zhang, 1999)
Competency	The extensive set of abilities that provide productivity, efficiency, and effectiveness of activities towards the aims and goals of the firm.	Formative- 1st Order	OAC1, OAC2, OAC3, OAC4	(H. Sharifi & Z. Zhang, 1999)
Flexibility	The ability to process different products and achieve different objectives with the same facilities.	Formative- 1st Order	OAF1, OAF2, OAF3, OAF4	(H. Sharifi & Z. Zhang, 1999)
Quickness	The ability to carry out tasks and operations in the shortest possible time.	Formative- 1st Order	OAQ1, OAQ2, OAQ3, OAQ4	(H. Sharifi & Z. Zhang, 1999)

APPENDIX C

PILOT STUDY SURVEY INSTRUMENT

Which statement best describes your current employment status?

- ❍ Working (paid employee) (1)
- ❍ Working (self-employed) (2)
- ❍ Not working (temporary layoff from a job) (3)
- ❍ Not working (retired) (5)
- ❍ Not working (other) (7)
- ❍ Prefer not to answer (8)

Which of the following industries most closely matches the one in which you were last employed?

- ❍ Aerospace, Defense (42)
- ❍ Automotive (43)
- ❍ Construction (44)
- ❍ Education (45)
- ❍ Entertainment, Media (46)
- ❍ Financial Services (47)
- ❍ Forestry, Agriculture (48)
- ❍ Government (Federal, State, or Local) (49)
- ❍ Hospitality, Travel, Tourism (50)
- ❍ Industrial & Manufacturing (51)
- ❍ Insurance (52)
- ❍ IT Services, Technology (53)
- ❍ Logistics, Distribution (54)
- ❍ Media, Entertainment & the Arts (55)
- ❍ Pharma, Healthcare & Biotechnology (57)
- ❍ Professional Services (58)
- ❍ Retail (59)
- ❍ Telecommunications (60)
- ❍ Utilities & Energy, Natural Resources (62)

How many years of industry experience do you have?

_____ In Total? (1)
_____ In your current department/organization? (2)
_____ In your current role? (3)

How many employees work in your company?

- ❍ 1-4 (1)
- ❍ 5-9 (2)
- ❍ 10-19 (3)
- ❍ 20-99 (4)
- ❍ 100-499 (5)
- ❍ 500-1,999 (6)
- ❍ 2,000-4,999 (7)
- ❍ 5,000-9,999 (8)
- ❍ 10,000 or more (9)
- ❍ I don't know (10)

Where are you employed?

- ❍ PRIVATE-FOR-PROFIT company, business or individual, for wages, salary or commissions (1)
- ❍ PRIVATE-NOT-FOR-PROFIT, tax-exempt, or charitable organization (2)
- ❍ Local GOVERNMENT employee (city, county, etc.) (3)
- ❍ State GOVERNMENT employee; 5-Federal GOVERNMENT employee (4)
- ❍ Federal GOVERNMENT employee (5)
- ❍ SELF-EMPLOYED in own NOT INCORPORATED business, professional practice, or farm (6)
- ❍ SELF-EMPLOYED in own INCORPORATED business, professional practice, or farm (7)
- ❍ Working WITHOUT PAY in family business or farm (8)

Which of the following best describes your role?

- ❍ Designer (1)
- ❍ Developer/QATester (2)
- ❍ Operations Engineer (3)
- ❍ Project/Program Manager (4)
- ❍ Product Owner/Manager (5)
- ❍ Security Engineer (6)

- ❍ Technical Leader (7)
- ❍ Technology Manager (8)
- ❍ Other (9)

Which choice best describes the status of DevOps in your organization?

- ❍ DevOps is foreign to my organization (1)
- ❍ DevOps plans are in the roadmap, but have not yet started (2)
- ❍ DevOps is just beginning in my organization (3)
- ❍ DevOps is implemented in some functions of my organization (4)
- ❍ DevOps is implemented consistently in all functions of my organization (5)
- ❍ I do not know (6)

In your opinion, please select the drivers of DevOps initiatives in your organization? Check all that apply.

- ❑ Accelerate operations tasks (1)
- ❑ Deliver product more frequently (2)
- ❑ Develop and deliver products to market faster (3)
- ❑ Identify defects as early as possible (before they get to production) (4)
- ❑ Improve Overall Agility (5)
- ❑ Improve quality (6)
- ❑ Increase collaboration/communication (7)
- ❑ Increase cultural maturity (8)
- ❑ Promote self managing teams (9)
- ❑ Reduce Expense (Cost Savings) (10)
- ❑ Reduce feedback cycle (11)
- ❑ To reduce the number of defects in development (12)
- ❑ Under pressure to deliver (13)

In your opinion, please rate the importance of the following DevOps Capabilities.	Extremely important (1)	Very important (2)	Slightly important (4)	Not at all important (5)
Common metrics for development and operations (1)	❍	❍	❍	❍
Continuous Deployment (2)	❍	❍	❍	❍
Continuous Integration and Testing Build Out (3)	❍	❍	❍	❍
Continuous Planning and Release Management (4)	❍	❍	❍	❍
Enterprise Automation (Build, Test, Deployment, Recovery, Monitoring, Infrastructure) (5)	❍	❍	❍	❍
Common Goals for development & operations (6)	❍	❍	❍	❍
Common Processes & Tools (Standardization) (7)	❍	❍	❍	❍
Open channels for constant, effortless communication (8)	❍	❍	❍	❍
Promote continuous experimentation and learning (9)	❍	❍	❍	❍
Shared code responsibility, collective ownership (10)	❍	❍	❍	❍
Shared goals, definition of success, common incentives across teams (11)	❍	❍	❍	❍
Shared values, respect, trust across teams (12)	❍	❍	❍	❍
Automated Infrastructure Provisioning (13)	❍	❍	❍	❍
Automatic Security Testing (14)	❍	❍	❍	❍
Configuration Management (Software & Infrastructure) (15)	❍	❍	❍	❍
Continuous Monitoring, Optimization and Feedback (16)	❍	❍	❍	❍
Environment Virtualization (17)	❍	❍	❍	❍

APPENDIX D

RESEARCH SURVEY INSTRUMENT

Part I. Individual Demographics

Please enter your response based on your experience:	
YrsCurrRole	How many years have you been in your current role?
YRSExpTech	How many years of experience do you have in technology overall?
YRSAgile	How many years of experience do you have working with agile technologies?

Please select the response that best describes the industry that matches the one in which you were last employed:	
1	Automotive
2	Construction
3	Education
4	Entertainment, Arts, Media
5	Financial Services
6	Government (Federal, State or Local)
7	Hospitality
8	Manufacturing
9	Information Technology
10	Healthcare, Biotechnology
11	Retail
12	Telecommunications
13	Utilities
14	Consulting Services
15	Other

Please select the response that best describes the organization that you currently work in:	
1	Development or Engineering
2	C-Level Executive
3	Consultant
4	DevOps
5	Networking
6	Operations or Infrastructure
7	Product Management
8	Quality Assurance or Test
9	Release Management

10	Sales or Marketing
11	Security Engineering
12	Student
13	Other

Please select the response that best describes the size of the organization that you currently work in:	
1	1-49
2	50-999
3	1000-4999
4	5000-9999
5	10,000 or more
6	I don't know

Part II. DevOps Capability

For each of the questions below, please select the response that most closely reflects what you observe about the organization in which you work.

Scale:
1=SD (Strongly Disagree); 2=D (Disagree); 3=N (Neutral); 4=A (Agree); 5=SA (Strongly Agree)

Continuous Integration & Testing		**SD**	**D**	**N**	**A**	**SA**
CI1	Our code commits result in an automated build of the software.					
CI2	Builds and tests are performed successfully every day.					
CI3	Current builds are available to testers for troubleshooting and exploratory testing.					
CI4	Developers get feedback from the acceptance and performance tests every day.					
CI5	Security features are integrated and tested as part of the automated test suite.					
Continuous Monitoring Optimization & Feedback						
CM1	Our monitoring solutions predict potential issues before end users are affected.					
CM2	My organization collects process metrics and measures regularly.					
CM3	Process feedback is actively sought in my company.					
CM4	Information security provides feedback on the designs of the applications that I work on.					
CM5	We use data from application performance monitoring tools to make daily business decisions.					
CM6	We use data from infrastructure monitoring tools to make daily business decisions.					
CM7	We receive failure alerts from logging and monitoring systems.					
CM8	We monitor system health based on threshold warnings.					

Automation Capability						
AC1	Our code commits result in automated test execution.					
AC2	The majority of our application (software) deployments are automated.					
AC3	Our scripts for automating build and configuration are in a version control system.					
AC4	Automated tests are regularly run as part of the main pipeline and workflow					
AC5	My team can deploy and release our product or service on demand, independently of other services it depends on.					
AC6	When automated tests pass, I am confident the software is releasable.					
Shared goals & common tools & processes						
SG1	Our teams share tools and common processes.					
SG2	Our teams share knowledge and goals.					
SG3	Our teams welcome new ideas.					
SG4	Our teams collaborate cross functionally and communicates openly.					
SG5	Our teams share responsibilities; treating failures as learning opportunities.					
SG6	Our teams use visual boards or dashboards to share information.					

Part III. Organizational Capability

For each of the questions below, please select the response that most closely reflects what you observe about the organization in which you work.

Scale:

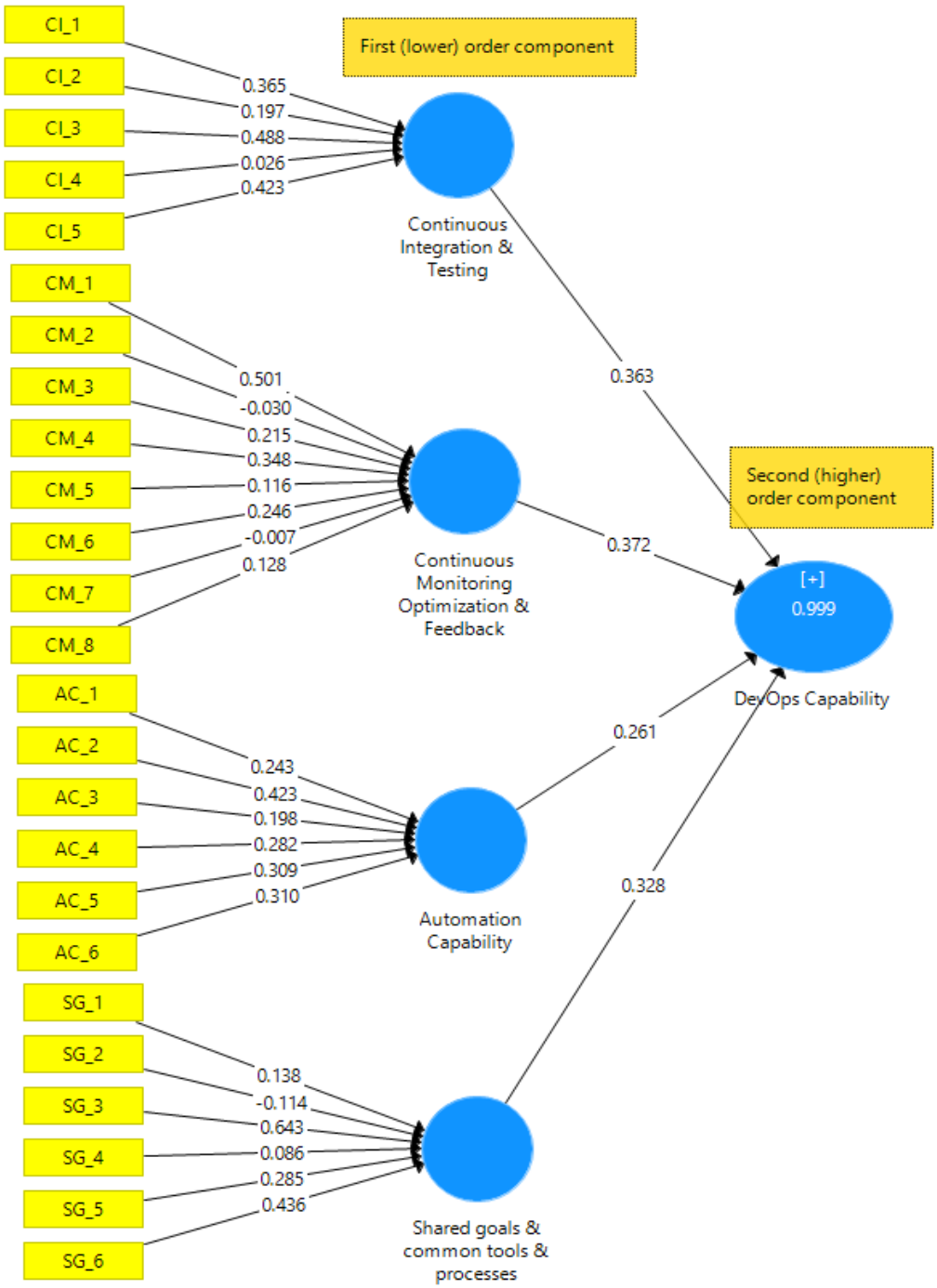
1=SD (Strongly Disagree); 2=D (Disagree); 3=N (Neutral); 4=A (Agree); 5=SA (Strongly Agree)

RESPONSIVENESS		**SD**	**D**	**N**	**A**	**SA**
OAR1	Our company demonstrates the ability to sense, perceive and anticipate changes occurring in the business environment.					
OAR2	Our company demonstrates the ability to immediately respond to change in the business environment by revising and/or adopting new processes and procedures into the organization.					
OAR3	Our company demonstrates the ability to shift quickly to take advantage of new opportunities.					
OAR4	Our company demonstrates the ability to recover from change in the business environment.					

COMPETENCY						
OAC1	Our company provides opportunities for the development of skills and knowledge in order to perform my role.					
OAC2	Our company has a unifying purpose and/or mission and strategic vision other than profitability and growth.					
OAC3	Our company encourages people to be innovative and develop new products or better ways of working together.					
OAC4	Our company allows information to flow freely from the outside to units and groups.					
FLEXIBILITY						
OAF1	Our company has the capability of shifting its structure quickly to address new opportunities in the marketplace.					
OAF2	Our company promotes a culture that embraces change as normal.					
OAF3	Our company develops strategies, goals and processes with flexibility in mind.					
OAF4	Our company has flexible organizational budgets and/or resources that enable quick response to marketplace changes.					
QUICKNESS						
OAQ1	Our company is able to quickly deliver new products and services to the market.					
OAQ2	Our company is able to quickly respond to changes in the business environment.					
OAQ3	Our company has a culture of speed/quickness across the organization that permeates every department and business process.					
OAQ4	Our company has processes that are built with just-in-time capability to handle sudden and fast arriving situations.					

APPENDIX E

DEVOPS CAPABILITY CONSISTENT PLS RESULTS



APPENDIX F

ORGANIZATIONAL AGILITY CAPABILITY CONSISTENT PLS RESULTS