The Impact of Control on Teams in Agile Information Systems Development

Inauguraldissertation

zur

Erlangung des Doktorgrades

der

Wirtschafts- und Sozialwissenschaftlichen Fakultät

der

Universität zu Köln

2022

vorgelegt

von

Dipl.-Wirt.-Inf. Tim Dreesen

aus

Mülheim a. d. Ruhr

Referent: Prof. Dr. Christoph Rosenkranz

Korreferent: Prof. Dr. Sean Hansen

Tag der Promotion: 21.04.2022

THE IMPACT OF CONTROL ON TEAMS IN AGILE INFORMATION SYSTEMS DEVELOPMENT

THEORETICAL FOUNDATION AND EMPIRICAL EVIDENCE

Tim Dreesen

University of Cologne

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

APU	Agile Practice Usage
ASD	Agile Software Development
BC	Behavior Control
CAA	Computer Aided Analysis
CB	Covariance-based
CC	Clan Control
CFI	Comparative Fit Index
CSA	Control Style: Authoritative
CSE	Control Style: Enabling
DV	Dependent Variable
EOC	Emergent Outcome Control
HICSS	Hawaii International Conference on System Sciences
ICIS	International Conference on Information Systems
IC	Input Control
IS	Information Systems
ISD	Information Systems Development
ISR	Information Systems Research
IT	Information Technology
IV	Independent Variable
LDA	Latent Dirichlet Allocation
OC	Outcome Control
PLR	Proportional Loss Reduction
PPR	Pair Programming
RET	Retrospective
RMSEA	Root Mean Square Error of Approximation
RQ	Research Question
SC	Self-Control
SD	Software Development
SEM	Structural Equation Model
SLR	Structured Literature Review
TA	Team Autonomy
TFI	Tucker-Lewis Index
XP	eXtreme Programming

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

The Ph.D. journey was one of the best experiences I have had in my life. This has not only to do with the fact that I was able to visit places that I would most likely never have reached otherwise, but also because I was always in good company. Like probably most students, I also went through phases in which enthusiasm and joy over partial successes alternated with total frustration and a feeling of hopelessness. Especially at times when things did not go my way, good company becomes more important than ever before. These “travel companions” therefore deserve my special thanks at this point.

First of all, I would like to thank Prof. Dr. Christoph Rosenkranz for the supervision of my Ph.D. project and the continued support in every project phase. Christoph, I'm still impressed that you always had a good reference to answer my (possibly not always so precise and unambiguous) questions. Thanks for guiding me through lows and bringing light into my darkness.

Furthermore, I would like to thank my key co-author Phil Hennel for his support and fruitful discussions about all kinds of topics (not only research related) during my Ph.D. journey. Phil, I must confess that I really appreciated working with you, not only because of your expertise in the field but also your unstinting commitment to teaching and research.

I would also like to thank Prof. Dr. Sean Hansen for his brilliant role as advisor and co-author, which he always filled with good ideas and honest opinions. Sean, I appreciated our conversations across time zones as much as our personal exchanges at conferences and seminars, among other things, especially the investigation of local drink and meal habits.

I would also like to thank my wonderful wife Elizabeth, who had a lot of patience with me and sometimes provided the necessary "pressure” to push me forward. Elli, I could not have done this without you, so you deserve just as much a part of this success as I do, I love you. I would also thank my daughter Marilena for reminding me what really matters and motivating me in difficult situations. I will never forget your first attempts of dancing efforts in the middle of the living room while I was busy with some boring statistics. This was a perfect distraction and made me laugh for the rest of the day. Then of course I would like to thank my parents, Vera and Helmut, for their continued support during the last years, especially with their support in childcare and sometimes in providing us with delicious food.

I also thank all students who supported me in data collection and data analysis for my cumulative dissertation. Finally, I would like to particularly express my gratitude to all study participants from the organizations that support the empirical studies included in my dissertation. Without their support and the permission to gain in-depth field access, this dissertation project would not have been possible.

ABSTRACT

Agile information systems development (ISD) methodologies can now look back on almost 30 years of history. However, it is precisely these methodologies that continue to attract the attention of the research community today. Agile ISD strongly relies on social interaction and teamwork. In consequence, team processes and agile practices adopted by team members take an integral part in the success of agile ISD projects. The ability to respond and react to changing or unforeseen user requirements becomes essential and is bolstered by granting higher levels of autonomy within an agile ISD team. However, existing studies on team autonomy in agile ISD imply that these teams not only benefit from team autonomy itself but also from different elements of control. Research suggests that control leads to better performance within a team, even though the exercise of control inevitably imposes certain boundaries on the concept of team autonomy. Yet, research faces an ongoing challenge in constituting a comprehensive understanding of how control should be used in agile ISD and how it affects certain levels along the whole ISD process.

The dissertation's objective is to improve our understanding of the influence of control on agile ISD teams in terms of team autonomy and team performance and how to enact control in agile ISD settings. This is achieved by conducting five independent but interrelated studies, which focus on the development and testing of a research model based on a solid theoretical foundation. As a theoretical framework, control theory is employed and extended with novel insights from the expanded theoretical framework of IS project control.

Collectively, these studies substantially extend our knowledge of the matter of control in agile ISD in general, and in particular, how control enactment can be linked to agile practices while considering different control styles, and how different types of control influence autonomy and performance in agile ISD teams.

1 Introductory Paper

1.1 Introduction

In today's practice for developing information systems (IS), approaches range from a variety of sequential and more plan-driven approaches (Royce 1970b, p. 174) to more cyclic, iterative approaches (Boehm 1988). During the last two decades, agile information systems development (ISD) methodologies such as eXtreme Programming (XP), rapid application development, or Lean complemented the iterative approach (Diegmann et al. 2018; Dreesen et al. 2019). Additionally, new management concepts associated with agile ISD, such as Scrum and Lean Software Management, have been proposed (Cohn 2010; Nurdiani et al. 2016; Poppendieck and Poppendieck 2003; Schwaber 1995; Schwaber and Sutherland 2013). These approaches promise to address the issue of dynamic environments and to be capable of reacting to unforeseen changes by trading plan-driven control for more flexibility and autonomy within the development teams themselves (Beck 1999; Beck et al. 2001a; Cockburn et al. 2001; Highsmith et al. 2001; Hummel et al. 2015).

While a few years ago these methodologies were rather used for smaller projects with relatively small development teams, nowadays agile methodologies are increasingly employed in distributed, or outsourced projects (e.g., Choudhury and Sabherwal 2003; Sarker and Sarker 2009), sometimes also combined with so-called 'large-scale frameworks', which are increasingly establishing themselves, making them the de facto standard for ISD (Conboy and Carroll 2019; Dingsoeyr et al. 2019; Jorgensen 2019; Leffingwell 2020; Moe et al. 2019). Consequently, an ongoing challenge for research constitutes a plethora of different aspects, for example, to increase our knowledge of how to introduce agile ISD methodologies beyond collocated settings or large-scale frameworks to teams and organizations (e.g., Cao et al. 2009b; Conboy and Carroll 2019; Dikert et al. 2016), usage and consequences of agile practices (e.g., Balijepally et al. 2009; Maruping et al. 2009a; Maruping et al. 2009b; Niederman et al. 2018; Recker et al. 2017), or even the adoption or tailoring of agile methodologies to a team's specific needs (Karlsson and Ågerfalk 2009; Lee and Xia 2010a; Wang et al. 2012). Another stream in literature puts emphasis on outcome variables in agile process models such as success (e.g., Chow and Cao 2008; Dikert et al. 2016; Lindsjørn et al. 2016; Wood et al. 2013) or job satisfaction (Melnik and Maurer 2006; Tripp and Armstrong 2014; Tripp and Riemenschneider 2014).

However, closely connected to success and mitigating risk in agile ISD is a well-known but for long neglected aspect within the development process, which emerged in research within the last years, that is the matter of *control* for agile ISD (e.g., Goh et al. 2013; Gregory et al. 2013b; Harris et al. 2009a; Sun and Schmidt 2018). This dissertation builds on this trend and draws on sound knowledge from organizational theory (e.g., Adler and Borys 1996; Orlikowski 1991; Ouchi 1979; Tannenbaum 1962), control in plan-driven ISD (e.g., Boehm 1991; Henderson and Lee 1992; Kirsch 1997) as well as on recently gained insights and extensions of control theory (e.g., Cram and Wiener 2018; Heumann et al. 2015; Remus et al. 2019; Wiener et al. 2016). This research contributes to the corresponding body of knowledge by investigating the general role of control in agile ISD, the linkage between agile practices and control and how control can be enacted through them, as well as the interplay and interdependencies of novel concepts such as control styles and control congruence.

1.1.1 Problem Statement and Research Objective

Although agile ISD methodologies continuously increase in usage and purportedly improve their outcomes compared to structured and plan-driven ISD approaches, agile ISD projects still fail, or do not meet the expected results (VersionOne 2020). Recent studies clearly described agile methodologies as "not a silver bullet" whose adoption also automatically results in success in the ISD process (Andrei et al. 2019; Brooks and Kugler 1987; Dreesen and Schmid 2018; Fitzgerald et al. 2006b; Fraser and Mancl 2008; Fraser et al. 2007). For example, a study of Jorgensen (2019) shows that, despite the growth of agile ISD, only 16% of software projects are successful and project failure rates have changed little since 2001, when the 'agile' designation was coined (Nelson 2005; Nelson 2007; Nguyen 2016). These findings suggest that agile project success still depends upon a wide variety of factors such as communication (Hummel et al. 2013), psychological safety (Hennel and Rosenkranz 2021), or self-organization (Hoda et al. 2013) affecting different facets of the agile ISD process (Siau et al. 2010).

Moreover, substantial research suggests that control, i.e., any attempt to ensure that individuals act according to organizational objectives (Jaworski 1988; Kirsch 1996), leads to better team performance within a team (Hackman 1987; Zellmer-Bruhn and Gibson 2006), even in agile ISD contexts (Harris et al. 2009a; Kirsch et al. 2002; Persson et al. 2011). Furthermore, the use of agile methodologies and the exercise of control have shown to be helpful in counteracting threats of success by having a positive effect on project quality (Maruping et al. 2009a). Nevertheless, many questions remain unanswered so far. For example, there is a lack of knowledge a) about the effect of control on team autonomy (e.g., Gerwin and Moffat 1997; Kirsch et al. 2002; Piccoli et al. 2004) b) which controls should be applied in agile projects (e.g., Cram et al. 2016b; Dikert et al. 2016; Simard and Lapalme 2019; Sun and Schmidt 2018), or c) how controls can best be put into practice (e.g., Heumann et al. 2015; Remus et al. 2019; Remus et al. 2016; Wiener et al. 2016). These are just a few of the issues where a fully, comprehensive understanding is not yet available. This is further complicated by the fact that some of the research results to date show partially contradictory results. To name just a few examples, it is largely unclear whether formal and informal controls are mutually exclusive or complementary (e.g., Persson et al. 2011; Srivastava and Teo 2012; Tiwana 2010), to what extent different control styles influence the effect of control modes (Chua et al. 2012; Remus et al. 2016; Wiener et al. 2016), or how control mechanisms can be associated with agile practices (e.g., Dreesen and Schmid 2018; Harris et al. 2009b; Mahadevan et al. 2015).

To tap these gaps in literature, this research zooms in on three essential aspects that must be considered for control in agile ISD. First, how can agile practices address and enact control in form of certain control modes? Second, how do styles of control (embodied by controllers such as managers, team leads or supervisor) influence autonomy and performance of agile teams? Third, what types of controls are needed, and how need both control and autonomy be balanced, to increase performance while still maintaining desired degrees of flexibility? Accordingly, this dissertation is based on the following overarching research question (RQ):

"How do control styles, agile practices, and modes of control affect each other and how do they influence an agile ISD team's autonomy and performance?"

1.1.2 Overall Research Design and Structure of the Dissertation

This dissertation is composed in a cumulative style. It consists of four building blocks comprising insights of five consecutive research articles that already have been published or are in the process of publication in academic journals and academic conferences (see Figure 1-1). It should be mentioned that the versions included do not differ in content but have been reformatted to the underlying style template for the purpose of consistency and better reading flow. One elementary core part of this dissertation is an 'introductory paper', which aims to summarize the whole research project, including motivation, theoretical underpinning, research design, results, and contributions. The summary narrative aligns with the four blocks and the underlying research articles, ultimately following a common thread, which I will discuss in more detail below.

The first part aims to justify the motivation of this work and to introduce the underlying theoretical background. This block is the only one to contain two different studies. Study I's objective is to introduce to the topic of control in agile ISD by presenting a preliminary research model, which comprises some novel concepts of control and combines same with solid theoretical knowledge from control theory. The research model is based on the results of a comprehensive structured literature review (SLR), which is based on the recommendations and guidelines of Webster and Watson (2002a) as well as Levy and Ellis (2006). In this context, research propositions are presented and opportunities for empirical testing of these are outlined.

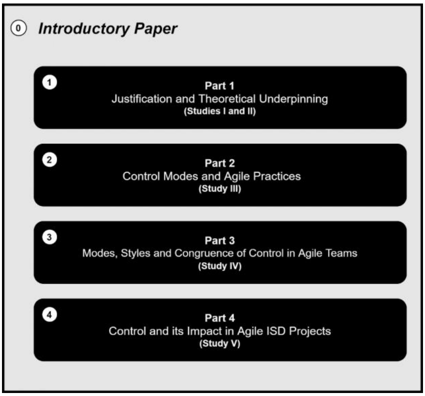


Figure 1-1: Overarching dissertation structure

In addition, a second study underpins the relevance of the topic of 'control' in agile ISD. Based on a combination of a structured literature review (Levy and Ellis 2006; Webster and Watson 2002a) with computer aided analysis (CAA) (Aggarwal and Zhai 2012; Debortoli et al. 2016a), we[1] approached a

[1] As all of the studies are outcomes of collaborations between different researchers and/or research teams, it may appear that sometimes I refer to "we" instead of "I" within this dissertation.

dataset of 1,376 research paper which dealt with the topic of agile ISD. The objective of this research was threefold: we wanted to 1) evaluate 'key articles' (proceedings and journals) of agile ISD, (2) analyze the development of agile ISD research, and (3) identify research foci of the past and gaps in our knowledge on agile ISD. One of the essential findings of this study identifies the topic group of 'Risk, Control & Success Factors in Agile' as an important area of research. In particular, the study implies the topic of 'Control in agile ISD' to be developed almost constantly over time, but to be generally less often discussed compared to other topics (Dreesen et al. 2019). Along with several other calls for further research (e.g., Wiener et al. 2016) this study's results indicate a gap in literature and consequently put emphasis to a need for further (research) action.

The second part of the dissertation focuses on an evaluation of the ways in which control can be enacted through agile practices. Specifically, there has been already some evidence in the literature that practices do embody various forms of control (e.g., Harris et al. 2009b; Mahadevan et al. 2015; Sun and Schmidt 2018). Nevertheless, these findings generally lack an empirical evaluation; exacerbating the issue, conflicting findings also exist in some cases. We took this as an opportunity to consolidate, interpret, and evaluate previous findings based on a SLR and then compare them using qualitative data (interview data). Our study suggests that a) practices are capable of exerting control, and b) only a few practices can be assigned to a single control mode.

Based on the results of Study III and recent findings of Wiener et al. (2016), we conducted a multiple case study and collected data from five different case organizations. Our objective was to confirm and improve our understanding of how to enact control (e.g., through agile practices) in agile teams and how these control mechanisms influence team autonomy and team performance. We found evidence for our previously theorized propositions that (1) controls enacted through agile practices positively influence team performance, and that (2) an enabling control style promotes team autonomy and congruence of control, which in turn (3) contributes to an increase of team performance (control congruence implies a mediating effect).

Finally, the fourth block of this dissertation aims to substantiate the previous findings with quantitative methods and to provide evidence for the claims of this research that have not been empirically demonstrated so far. We did this by conducting a field study and collected matched-pair survey data of 148 supervisor-team member dyads. We considered the role of different control styles, the interplay between formal and informal control mechanisms as well as their relationship to team autonomy. Our results show that the choice of control style significantly determines how agile practices are implemented and how agile practices and control mechanisms facilitate each other in an agile environment, while it is still possible to achieve the desired degree of autonomy within a team.

An overview of all studies and their respective papers, their underlying research design, and their corresponding submission status is given in Table 1-1.:

#	Title	Research Design	Outlet	Status
I	Agility in the Balance: Control, Autonomy, and Ambidexterity in Agile Software Development	Structured literature review	ICIS	Published

II	Journey Towards Agility – A Retro- and Prospective Review	Structured literature review / CAA	HICSS	Published
III	Do As You Want Or Do As You Are Told? Control vs. Autonomy in Agile Software Development Teams	Literature review / qualitative field study	HICSS	Published
IV	"Loosening the Reins": Balancing Control and Autonomy in Information Systems Development	Qualitative field study	HICSS/ISJ	Published / Submitted, 1st round
V	"Directing Self and Others": An Empirical Study of Control in Agile Information Systems Development	Quantitative field study	ICIS/ISR	Rejected / Submitted, 1st round
Legend: ICIS = International Conference on Information Systems, HICSS = Hawaii International Conference on System Sciences, JIT = Journal of Information Technology				

Table 1-1: Overview about included research papers

However, research is rarely an undertaking of an individual but rather the product of cooperation and teamwork. Consequently, all studies presented are outcomes of collaborations between different researchers and/or research teams. Table 1-2 offers details about the contributions of different researchers.

Study	Agility in the Balance: Control, Autonomy, and Ambidexterity in Agile Software Development	Journey Towards Agility – A Retro- and Prospective Review	Do As You Want Or Do As You Are Told? Control vs. Autonomy in Agile Software Development Teams	„Loosening the Reins": Balancing Control and Autonomy in Information Systems Development	"Directing Self and Others": An Empirical Study of Control in Agile Information Systems Development
Authors	T. Dreesen; S. Hansen	T. Dreesen; P. Diegmann; B. Binzer; C. Rosenkranz	T. Dreesen; T. Schmid	T. Dreesen; C. Rosenkranz; P. Hennel; S. Hansen	T. Dreesen; C. Rosenkranz; P. Hennel; S. Hansen
Research Design	T. Dreesen; S. Hansen	T. Dreesen; P. Diegmann; C. Rosenkranz	T. Dreesen	T. Dreesen; C. Rosenkranz S. Hansen	T. Dreesen C. Rosenkranz
Data Collection	T. Dreesen	T. Dreesen; P. Diegmann; B. Binzer;	T. Dreesen; T. Schmid	T. Dreesen; P. Hennel;	T. Dreesen; P. Hennel
Data Analysis	T. Dreesen	T. Dreesen; P. Diegmann; B. Binzer; C. Rosenkranz	T. Dreesen; T. Schmid	T. Dreesen; P. Hennel; C. Rosenkranz	T. Dreesen
Communi-cation	T. Dreesen*; S. Hansen	T. Dreesen*; P. Diegmann	T. Dreesen*	T. Dreesen; P. Hennel* C. Rosenkranz	T. Dreesen; C. Rosenkranz
* Indicates the presenter when submitted to a conference					

Table 1-2: Authors and their contributions

This introductory paper proceeds as follows. Chapter 2 presents an overview of related work at the intersection between digital technology and entrepreneurship. Chapter 3 describes the empirical studies of this dissertation. Thereafter, Chapters 4, 5, and 6 present three empirical studies. Chapter six summarizes the findings, highlight their contributions, and points to direction for future research.

1.2 Theoretical Background and Related Work

1.2.1 Information Systems Development

For more than 55 years, research put emphasis on the investigation of the adoption of information technologies (IT) by people and organizations, which has been part of what we nowadays understand as IS implementation (Kaplan and Duchon 1988; Moore and Benbasat 1991). Generally, most of the available conceptual explanations of the term IS can be distinguished to what degree they focus on either social or technical concerns (Alter 2008; Falkenberg et al. 1998; Geiger et al. 2012). Socio-technical theory is one attempt which considers both perspectives as it distinguishes between a technical and social (sub) system (Trist 1963). The technical subsystem encompasses both 'technology' (i.e., tools to generate outputs based on certain inputs) and 'process' (i.e., necessary procedures to perform economic activities), while the social subsystem encompasses 'people'[2] (who are directly involved in the IS) and a hierarchical reward and reporting 'structure', in which those people are embedded (Lui et al. 2007). To properly perform its functionalities an IS needs these four components to interact with each other in a productive manner (Lui et al. 2007; O'Hara et al. 1999) (see Figure 1-2).

This research follows this viewpoint on IS and considers IS as socio-technical systems. Based on these assumptions, an IS can simply be described as a compound of people and machines (hardware including software, networks, communication devices) which creates or utilizes information and which is interconnected by other communication relationships (Hansen et al. 2019). A synonym often found in literature is "software", but software is nevertheless not simply synonymous with IS. Based on the above, the essential difference is that IS comprises several components in which "software" represents only one element of many. In fact, the majority of academics go beyond a primarily technical view, in which for example the "people" factor is seen as an elementary component of IS (e.g, Davis 2000; Hansen et al. 2019; Hevner et al. 2004; Kaplan and Duchon 1988; O'Brien and Marakas 2009).

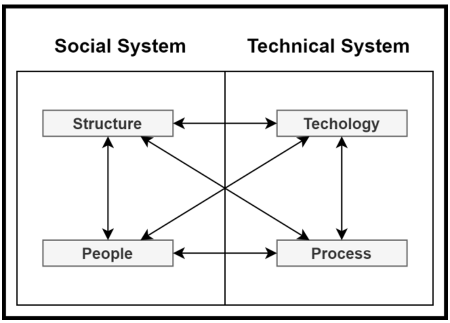


Figure 1-2: IS as socio-technical systems (Lui et al. 2007)

[2] sometimes also referred to as 'actors' (e.g. Falkenberg 1998, Lyytinen et al. 1998).

From an organizational perspective, IS are developed and implemented for the purpose of improving the effectiveness and efficiency of an organization (Hevner et al. 2004; Markus 1983; Nissen and Jin 2007), in particular, they are supposed to provide an organization with services which are needed for operations and management (Davis 2000; Falkenberg et al. 1998; O'Brien and Marakas 2009). Compared to ISD, the discipline of software engineering pursues a more narrowed goal, in which the emphasis is more on software, quality, and tools (Pressman 2010). For example, Pressman describes software engineering as an approach, which “encompasses a process, methods for managing an engineering software and tools” (2010, p. 13). Summarized, as ISD is a socio-technical process rather than a purely technical one and therefore must deal with both organizational and technical issues, ISD is even more complex and challenging to manage (Lee and Xia 2005). Although there have been a variety of different methodologies in the past, ranging from sequential, disciplined, iterative, or emergent approaches (Hummel et al. 2015), the success rate of ISD projects has historically been low (Jorgensen 2019; Lee and Xia 2005). Failures such as budget overruns, significant delays in time, or even organizational rejections have been quite common in ISD (Lyytinen et al. 1998). One of the reasons identified was the lack of flexibility in the methodologies, which is essential for responding to change (Austin and Devin 2009; Conboy 2009; Keil et al. 2013; Lee and Xia 2005; Lee and Xia 2010b; Misra et al. 2009). In contemporary business environments, where the needs of consumers and business professionals change rapidly and continue to evolve over time, the ability to respond quickly to changing user requirements has become essential for ISD success (Lee and Xia 2010a; Maruping et al. 2009a; Vidgen and Wang 2009). To address these issues, so-called ‘agile ISD methodologies’ have emerged (Beck 1999; Cockburn et al. 2001; Highsmith and Cockburn 2001; Schwaber and Beedle 2001)

1.2.2 Agile Information Systems Development

Agile ISD nowadays has become a collective term for a variety of different methodologies, such as Scrum, eXtreme Programming (XP), and Crystal (e.g., Martin 1991; Poppendieck and Poppendieck 2003; Schwaber 1995; Stavru 2014), which collectively emphasize an iterative and incremental development model, as well as close collaboration between stakeholders and involvement of clients (Dingsøyr et al. 2012; Larman 2003b; Williams and Cockburn 2003). Another common feature that characterizes these methodologies is their emphasis on significant flexibility and autonomy for project teams (Hoda et al. 2013; Wood et al. 2013). In market environments where technologies rapidly emerge, market structures, and customer preferences change quickly, agile ISD approaches promise to enable teams to counteract emergent needs in a timelier manner than traditional structured development approaches (Conboy and Fitzgerald 2007).

Once the choice for the use of an agile approach has been made, one is inevitably confronted with the selection of suitable so-called agile practices (Hummel et al. 2015; Pelrine 2011; Recker et al. 2017; Sarker and Sahay 2003). Agile practices can be described as methods-in-action and generative rules that are adapted to fit an ISD team’s specific context (Highsmith et al. 2001). Examples of agile practices are pair programming (code is written with two programmers at one machine) and collective code ownership (anyone can change any code anywhere in the system at any time) from XP; similarly, popular Scrum practices include daily scrums (a daily stand-up meeting in which all project participants briefly review the status of their work) and user stories (a methodology to define broad, user-centered requirements while enabling creativity) (Cohn 2010; Harris et al. 2009b; Tripp et al. 2016).

But for all the multitude and broad range of these methodologies, they all have in common that they are rooted in the values and principles of the Agile Manifesto (Beck et al. 2001a). Most prominent are four underlying basic 'values' (Beck et al. 2001a) which can be found in most agile ISD methodologies. Accordingly, agile ISD should value individuals and interactions over processes and tools, working software over comprehensive documentation, customer collaboration over contract negotiation, and responding to change over following a plan (Beck et al. 2001a). Each of these principles have been subject to research in some sort: for instance, in regard to individuals and interactions, research has investigated the effects of communication in agile ISD teams (Hummel et al. 2013), in regard to working software, extant literature investigated the influence of pair programming on software quality (Balijepally et al. 2009), in regard to customer collaboration, the funding process has been studied (Cao et al. 2013), and the ability to respond to change has been subject of studies as well (e.g., Fitzgerald et al. 2006b; Lee and Xia 2005; Sarker and Sarker 2009).

Surprisingly, in fact, there are few studies that look at agile methodologies as such, despite the increasing popularity of agile methodologies and the 20th birthday of the agile manifesto; many other studies focus on individual or organizational phenomena and effects of these methods. Examples are the use and effects of specific agile practices (e.g., Balijepally et al. 2009; Maruping et al. 2009a; Maruping et al. 2009b; Niederman et al. 2018; Recker et al. 2017) or effects regarding whole projects or organizations, such as the introduction of agile ISD methodologies to teams (e.g., Cao et al. 2009b), the use of hybrid methodologies or tailoring of agile methodologies to a team's specific needs (Karlsson and Ågerfalk 2009; Lee and Xia 2010a; Wang et al. 2012). Another stream in literature puts emphasis on outcome variables in agile process models such as success (e.g., Chow and Cao 2008; Dikert et al. 2016; Lindsjørn et al. 2016; Wood et al. 2013) or job satisfaction (Melnik and Maurer 2006; Tripp and Armstrong 2014; Tripp and Riemenschneider 2014). This literature mostly focuses on specific methodologies such as Scrum or XP (Fruhling and de Vreede 2006) or specific practices, for instance pair programming (Cao et al. 2013). Extant research focusing on success and failure of agile ISD in general exists but is rare (Jorgensen 2019; Lee and Xia 2010a; Nguyen 2016; Recker et al. 2017). Two important streams in research which are closely connected to the topic of success in agile ISD are the matter of control of the development process (e.g., Cram et al. 2016b; Gregory et al. 2013b; Hoda et al. 2013) and the empowerment of agile teams, granting them autonomy and flexibility needed in order to responded to unforeseen changes (Batra et al. 2017; Hoda et al. 2012; Masood et al. 2020; Xu and Shen 2016).

1.2.3 Autonomy, Control and Performance

1.2.3.1 Team Autonomy

Flexibility, being agile and a permanent readiness to respond to change in agile ISD approaches are all aspects that are reflected in the concept of team autonomy (Larman 2003b; Lee and Xia 2010a). Research can draw on a plethora of definitions of team autonomy and related concepts, some of which have a large common ground or are nearly the same, including self-organization (Chow and Cao 2008; Highsmith et al. 2001; Hoda et al. 2013), self-management (Sharp and Robinson 2004), and team empowerment (Larman 2003b; Maruping and Magni 2012). Therefore, to understand the subsequent course of this work, it is important to agree on a common definition of the underlying term. Following extant research, we define *team autonomy* as "the degree of discretion and independence granted to the team in scheduling the work, determining the procedures and methods to be used, selecting and deploying resources, hiring and firing team members, assigning tasks to team members, and carrying out assigned tasks" (Lee and Xia 2010a, p. 90). Despite a multitude of studies on the topic of autonomy in teams (e.g., Cordery et al. 2010; Gerwin and Moffat 1997; Moe et al. 2019), there is still no integrated overview of the extent to which autonomy is beneficial for a team. So far, findings seem ambiguous; on the one hand, autonomy in project teams seems to reduce productivity and performance (e.g., Langfred 2004; Maruping et al. 2009a; Yun et al. 2005), while other studies see autonomy as an important factor enabling teams to respond to new challenges and opportunities or even to increase team performance (e.g., Cordery et al. 2010; Lee and Xia 2005; Vidgen and Wang 2009). Furthermore, the interactions between autonomy at the individual level and at the team level are not well understood, for example, it remains unclear if autonomy on an individual level is needed to establish autonomy on a team level (e.g., Langfred 2004). Even more ambivalent results exists when we investigate different relationships of team autonomy associated with control related concepts. For example, studies show mixed results on whether formal control has a positive or negative effect on team autonomy. Table 1-3 provides an excerpt of exemplary studies that have focused on investigating the influence of either control on team autonomy or the effect of team autonomy on performance.

Effect	Studies
Team autonomy fosters performance	Cordery et al. (2010); David Gefen (2002); Gefen and Keil (1998)
Team autonomy reduces performance	Langfred (2004); Maruping et al. (2009a); Yun et al. (2005)
Team autonomy is positively influenced by informal controls	Henderson and Lee (1992); Maruping et al. (2009a); Kirsch et al. (2002); Remus et al. (2016)
Team autonomy is negatively influenced by formal controls	Barker (1993); Piccoli et al. (2004); Remus et al. (2016); Robey et al. (2000)
Team autonomy is not influenced by formal controls	Adler and Borys (1996); Cordery and Tian (2017); Dalton (1959); Feldman (1989)
Team autonomy is positively influenced by an enabling control style	Adler and Borys (1996); Wiener et al. (2016)

Table 1-3: Overview of team autonomy related effects

1.2.3.2 Control

Due to a general unspecific knowledge about the influence of team autonomy and a not exclusive positive effect of team autonomy in particular, the issue of *control* becomes a prominent question (Venkatesh et al. 2018; Wallace et al. 2004). Generally, the term 'control' has been treated differently in research, having different meanings dependent on different contexts (Flamholtz et al. 1985; McHugh 2011). This ambiguity is compounded by the fact that a number of terms are used synonymously with 'control', for example, 'governance', 'power' or 'command' (Chua and Myers 2018; Cole et al. 2013; Collier 2005; Druskat and Wheeler 2003; Kirkman et al. 2009; McHugh 2011; Simard and Lapalme 2019). This research adopts an interpretation of control, which focuses on an organizational perspective of control, which emphasizes performance evaluation and rules and procedures that have to be followed in order to achieve organisational goals. Consequently, we define control broadly as

"any process in which a person or group of persons or organization of persons determines … what another person or group or organization will do" (Tannenbaum 1962, p. 239)

As this definition suggests, the exercise of control necessarily implies certain limits on the ideal of team autonomy. Control itself encompasses a variety of different mechanisms and forms, and even recently has been supplemented with additional elements through the expanded theoretical framework of IS project control (Wiener et al. 2016). Despite a range of calls for further research on the impact of control and team autonomy in ISD efforts (Chua et al. 2012; Wiener et al. 2016), the existing evidence remains ambiguous, especially in the field of ongoing development of agile ISD methodologies (Cram and Brohman 2013; Dreesen et al. 2020). For example, few studies have investigated control modes and their effects on agile ISD team characteristics, such as team autonomy, or their enactment within agile practices (Cram et al. 2016a).

One well-known and widespread body of knowledge concerning control and its underlying mechanisms is represented by what is often referred to as *control theory* in recent literature (Jaworski 1988; Kirsch 1996; Maruping et al. 2009a; Ouchi 1977; Snell 1992). Having its origin in organizational and management science, control has been focused by organizational and agency theorist long ago (e.g., Eisenhardt 1985; Ouchi 1977) before it drew attention for managing ISD projects (Henderson and Lee 1992; Kirsch 1996). Control is often viewed in a behavioral sense, i.e., as any "attempt to ensure that individuals working on organizational projects act according to an agreed-upon strategy to achieve desired objectives" (Kirsch 1997, p. 216). This view implies to allow for a broader view on different control approaches and is in line with prior insights, for example Kirsch (1997) or Henderson and Lee (1992). A major advantage of this approach is that it deliberately abandons a cybernetic perspective in which, in principle, the desired outcomes are assumed to be known and control mechanisms are aligned on these assumptions. However, these outcomes are often vague in an environment such as ISD (and in particular in agile ISD), which underscores the need for a broader consideration of control (Kirsch 1997). In addition, this view implies a distinction of at least two different roles which take place in a typical control relationship, i.e., the role of a *controller* and *controllee*. The controller is usually equipped with the ownership of certain tasks (for example along the ISD process) and desires their accomplishment by assigning them to the controllee, thus, both roles resemble the ideas of a principal and agent in agency theory (Eisenhardt 1989; Wiener et al. 2016). However, these controller-controllee dyads in a control relationship may manifest in different forms and can be differentiated between a) hierarchical relationships and b) lateral relationships (Kirsch et al. 2002). Hierarchical relationships are typically characterized by having a 'superior' and a 'subordinate' (e.g., a team leader and their corresponding employees). Lateral relationships may comprise more than one individuum and may therefore include different groups of people, for an example, an organization's business department (controller) who assigns tasks for fulfillment to a corresponding IT unit (controllee) within the company (Kirsch et al. 2002; Wiener et al. 2016). A special case, and thus a third distinction, is the client-vendor relationship, which usually occurs in outsourced ISD environments (e.g., Tiwana and Keil 2009). Building upon these different views on typical controller-controllee relationships, it is important to mention, that generally groups of individuals may act as a controller (e.g. quality management departments, steering committees) and vice versa act as controllees. (e.g., agile ISD teams, vendors) (Wiener et al. 2016).

The most popular control concept underlying control theory is a strict distinction between formal and informal modes of control (Kirsch 1997). A *control mode* can be described as a certain type of control mechanism, for example demanding monthly reports, specifying milestones or attending socialization activities, whereas five different modes can be distinguished and have been developed over time. Formal control modes include *input, behavior* and *outcome* control, while *clan* and *self*-control belong to what is known as informal controls (Ouchi 1977; Ouchi 1978; Ouchi and Johnson 1978). Table 1-4 briefly summarizes each control mode by its core characteristics, in the following each mode will be explained more in detail.

Control Mode		Characteristics
Formal	Input Control	Measurable actions prior to implementation of an activity e.g., recruitment, training programs or manpower allotments
	Behavior control	Emphasizes behaviors, processes and procedures that must be followed, and offering rewards contingent on the adherence to the prescriptions.
	Outcome control	Involves outlining project goals and offering rewards contingent on their accomplishment. Emphasizes outputs regardless of the process used.
Informal	Clan control	Socializes team members into sets of valued norms. Emphasizes reinforcement of acceptable behaviors through shared rituals and experiences.
	Self-control	Provides autonomy to individuals to determine what actions are required and how to execute them. Emphasizes self-regulation of goals and self-monitoring of progress.

Table 1-4: Summary of control modes

Formal control is viewed as a performance evaluation strategy, its exercise provides guidance and establishes structure, assisting the development team in task execution (Eisenhardt 1985; Kirsch et al. 2002; Remus et al. 2016). Input control regulates the requirements needed for achieving the desired outcomes, typical examples are selection and training of (personnel) staff (Jaeger and Baliga 1985). Implementing input control may be beneficial, as it potentially avoids outcome related issues a priori, for example performance deficits. On the other hand, its implementation may have less effect or even be detrimental as input control only manages “potential”, there is no certainty about “what can be actually will be” (Snell 1992, p. 297). Behavior control aims at specifying rules and procedures which (if followed) will lead to the desired organizational outcomes, controllers in this context reward controllees based on the degree of compliance to these rules and sanction them in turn if deviations occur (Kirsch 1997). Behavior control is usually implemented when rules and procedures needed are known and controllee behavior can be observed (Eisenhardt 1985; Kirsch 1996; Snell 1992). An alternative for using behavior control, controllers can control “outputs” (Ouchi 1977) or outcomes. The main difference compared to behavior control is the solely definition of desired outputs, independent of how they can be achieved, or in other words, decoupled from the “behavior” that is likely to generate these outcomes (Kirsch 1997; Snell 1992). Controllers continuously evaluate interim and final outcomes and reward and sanction the controllee based on the achievement of the outcomes (Kirsch 1997; Remus et al. 2016). Summarized, “[…] input control regulates the antecedent conditions of performance—the knowledge, skills, abilities, values, and motives of employees; behavior control regulates the transformation process; and output control regulates results" (Snell 1992, p. 927). Importantly, traditional ISD approaches, such as structured or "waterfall" development models, rely heavily on such formal control mechanisms (Kirsch 1996; Kirsch 1997; Kirsch et al. 2002).

By contrast, informal controls are based on social or people strategies (Eisenhardt 1985; Jaworski 1988; Kirsch 1997). Developers benefit from informal controls by gaining greater discretion with regard to how tasks are accomplished (Henderson and Lee 1992; Kirsch et al. 2002; Maruping et al. 2009a; Tiwana and Keil 2009). As formal controls tend to present a 'mechanistic' view on the control process, they usually neglect interpersonal or self-regulating mechanisms, that also adjusts behavior (Kirsch 1997). In contrast to formal controls, informal control modes are typically undocumented and can be distinguished by their level of aggregation, that is clan control (group level) and self-control (individual level) (Jaworski 1988; Kirsch 1997). Clan control is characterized as "more powerful than either bureaucratic controls or market controls" (Macintosh and Quattrone 2010, p. 174) as their members hold similar beliefs and values, thus, having a strong sense of solidarity to the clan. They are subtle and elusive in nature and implementation is useful when organizational goals are ambiguous and not well understood (Macintosh and Quattrone 2010; McHugh 2011). The exercise of clan control allows the development team to identify important project goals and to determine collectively how to attain them (Maruping et al. 2009a). The exercise of self-control on the other hand, similarly, enables flexibility in pursuit of objectives, focusing on the role of the individual rather than that of the group. Self-control represents "the extent to which an individual exercises freedom or autonomy to determine both what actions are required and how to execute these activities" (Henderson and Lee 1992, p. 760). This emphasizes the definition of goals and processes for successful task accomplishments by the individuals themselves, while these may or may not be formalized (Kirsch 1997). Building upon this, informal controls, such as clan control and self-control, provide the promise of greater autonomy, which is seen as an important antecedent for responding to changing user requirements (Gerwin and Moffat 1997; Maruping et al. 2009a). Looking at the underlying capabilities of the different control modes and weighing their advantages and disadvantages against each other, it is not surprising that controllers do not limit themselves to single control modes but rather exercise them in concert, representing a so-called control portfolio (Kirsch 1997).

Despite of exclusively focusing on core elements of control theory as highlighted by Kirsch (1996; 1997; 2004), this research also looks beyond control modes by specifically considering extensions made by the expanded theoretical framework of IS project control (Wiener et al. 2016). Wiener et al. criticized in their study that extant research solely focuses on controlling portfolio configuration ('what' control modes are used) (2016). They argue that only few studies investigate 'how' controls can be put into practice – that is, what is broadly known as the concept of *control enactment* (Gregory et al. 2013a; Tiwana and Keil 2009). Specifically, control enactment is the interaction between a controller (the person exercising control) and a controllee (the target of control) (Wiener et al. 2016). In this context, *control style* is considered an elementary concept in the process of control enactment, as it significantly shapes the interaction of a control dyad (Wiener et al. 2016). Control style can be defined "as the manner in which the interaction between the controller and the controllee is conducted" (Wiener et al. 2016, p. 755). Two contradictory control styles are particularly noteworthy – *authoritative* and *enabling* (Adler and Borys 1996; Gregory et al. 2013a). An *authoritative* control style is employed if strict behavioral compliance is desired, granting the controllee limited discretion in taking action (Wiener et al. 2016). An *enabling* control style, on the other hand, is used to achieve compliant behavior while granting flexibility in decision making to deal with uncertainties in daily work procedures (Adler and Borys 1996; Remus et al. 2016). Two major characteristics can be distinguished with regards to an enabling control style. First of all, the “repair” feature which facilitates responses from controllees and appreciates controllee feedback about real work contingencies, as well as deviations from controller instructions when necessary. Second, the “transparency” feature, which is concerned with the visibility of control activities and the overall project context. For example, a controller provides the controllee with the rationale of the enacted controls, regular feedback on performance, and relevant and updated contextual information (the "big picture" or the rationale of the controls enacted) (Remus et al. 2019; Remus et al. 2020; Remus et al. 2016; Wiener et al. 2016).

In addition to the aforementioned control components, the concept of *control congruence* becomes important, which can be understood as the “level of agreement” or “degree of understanding” between a controller’s and controllee’s perception of the exercise of control (Narayanaswamy et al. 2013, p. 192; Wiener et al. 2016). The level of agreement regarding the appropriateness of controls is called *perceptual* congruence, whereas the degree of shared understanding of control measures is known as *communicational* congruence (Narayanaswamy et al. 2013; Ouchi 1978). Thus, control congruence may influence the quality of the whole control enactment process (Wiener et al. 2016).

In sum, limited guidance exists on how agile ISD teams should be governed with respect to the relationship between control and team autonomy, with significant ambiguity regarding how much team autonomy and how much control is needed, or what the appropriate balance between the two is (Cram et al. 2016a; Dreesen et al. 2020). Accordingly, we follow the call of Wiener et al. (2016) for further research on the inconclusive and partly contradictory results regarding control in ISD (Choudhury and Sabherwal 2003; Tiwana and Keil 2009), the role of different control styles, the interplay between formal and informal control mechanisms (Persson et al. 2011; Tiwana 2010), as well as their relationship to team autonomy (Gerwin and Moffat 1997).

1.2.3.3 Performance

Lastly, control occurs not without a reason, moreover control is enacted in ISD projects in order to steer outcomes and to achieve desired results or states. The question of outcomes brings us to another critical concept in the consideration of agile ISD dynamics, namely *team performance,* which is defined as the degree to which a team achieves its goals and how well its outputs match the team’s mission (Hackman 1987; Zellmer-Bruhn and Gibson 2006). Based on prior research, we anticipate that the various facets of control discussed have a significant impact on measures of performance (Goh et al. 2013; Harris et al. 2009a; Mahadevan et al. 2015; Maruping et al. 2009a; Persson et al. 2011). A variety of studies investigated different control effects on performance in terms of ‘efficiency’ (did goal achievement complete on time and within budget?) and ‘quality’ (did outcomes meet their intended purposes?). However, the exact nature of those relationships remains very much in question, not only in the context of agile ISD (Cram and Brohman 2013). Most studies take a rather one-sided view of control-performance relationships by focusing only on direct effects between individual modes and performance consequences (Wiener et al. 2016). Table 1-5 provides an excerpt of exemplary studies that have focused on investigating the influence of control on one of the aforementioned performance dimensions.

Effect type	Description	Studies
Direct effects of specific control modes	Distinct control modes such as input, behavior, outcome, clan or self-control influence performance.	Gopal and Gosain (2010); Henderson and Lee (1992); Keil et al. (2013); Maruping et al. (2009a); Tiwana (2010); Tiwana and Keil (2009)
Effects of control mode portfolios	Combinations of two or more control modes influencing performance.	Chua et al. (2012); Persson et al. (2011); Remus et al. (2020); Srivastava and Teo (2012); Tiwana (2010); Wiener et al. (2015)
Moderated effects of control	Studies which show the control effects being moderated	Gopal and Gosain (2010); Harris et al. (2009a); Keil et al. (2013); Mähring et al. (2018); Maruping et al. (2009a); Remus et al. (2019); Shinkle et al. (2021); Srivastava and Teo (2012); Venkatesh et al. (2018)
Indirect effects of specific control modes	Control effects on other variables which in turn influence performance	Liu et al. (2008); Wang et al. (2006); Yadav et al. (2007)

Table 1-5: Effects of control on performance[3]

There are only less and more recent studies like the one by Syed et al., which take into account not only classical control modes but also other influencing factors such as the relationship between different control tactics (control styles) and project performance (Syed et al. 2021). However, there are still limited studies available examining the role of control and its impact on performance in agile ISD environments, emphasizing the need for further research on this topic.

1.2.3.4 Summary

Table 1-6 summarizes the key concepts for understanding the basic terms and relationships. A complete description of all concepts is given in Appendix A.

Key Concept	Description	References
Control mode	Distinct type of control mechanism such as input, behavior, outcome, clan or self-control.	Jaworski (1988); Kirsch (1997); Ouchi (1979)
Control style	Commonly described as the way how a controller interacts with a corresponding controllee, whereas two contradictory control styles can be distinguished – *authoritative* and *enabling*. An *authoritative* control style is employed if strict behavioral compliance is desired, granting the controllee limited discretion in taking action. An *enabling* control style, on the other hand, is used to achieve compliant behavior while granting flexibility in decision making to deal with uncertainties in daily work procedures.	Adler and Borys (1996); Gregory et al. (2013a); Remus et al. (2016); Wiener et al. (2016)

[3] The table is based on an existing overview as provided by Wiener et al. (2016) and has been extended by additional recent quantitative studies. The original summary contains deeper information such as the direction of effects, distinction between different performance dimension as well as a further differentiation of studies in internal and outsourced ISD settings.

Control congruence	Defined as the "level of agreement" or "degree of understanding" between a controller's and controllee's perception of the exercise of control. These two different types of congruence are usually referred to as *perceptual* congruence (perceived appropriateness of controls) and *communicational* congruence (shared understanding of control measures).	Narayanaswamy et al. (2013, p. 192); Wiener et al. (2016)
Team autonomy	"degree of discretion and independence granted to the team in scheduling the work, determining the procedures and methods to be used, selecting and deploying resources, hiring and firing team members, assigning tasks to team members, and carrying out assigned tasks"	Lee and Xia (2010a, p. 90)
Team performance	The degree to which a team achieves its goals and how well its outputs match the team's mission.	Hackman (1987); Zellmer-Bruhn and Gibson (2006)

Table 1-6: Summary of key concepts used in this dissertation

1.3 Research Design

1.3.1 Theoretical Perspective and Research Approach

A plethora of different research approaches for conducting research in social sciences exists, with each having their own strengths and weaknesses (Recker 2012; Saunders et al. 2016; Yin 2003a). Survey, grounded theory, case study or action research are just some of them (Creswell 2009; Myers 2013). Researchers face a variety of fundamental decisions at the outset of a research endeavor that significantly influence the overall research design. For example, decisions must be made regarding how theory development is going to be approached (e.g., deductive vs. inductive), what sort of methods should be utilized (e.g., quantitative vs. qualitative) or what kind of strategies should be followed (e.g., survey-based vs. case study-based). Of course, these are just some of the questions that have to be answered, in general, researchers need to think at least about important components of a research design, represented, for example, as different layers of the so-called '*research onion'* following Saunders et al. (2016). Figure 1-3 shows the 'research onion' (Saunders et al. 2016), which illustrates essential aspects that have to be considered.

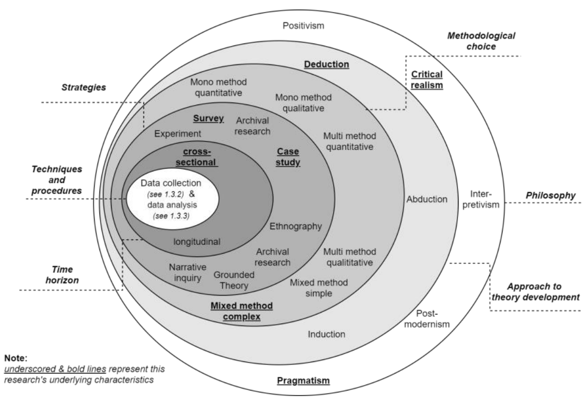


Figure 1-3: The research onion (Saunders et al. 2016)

In the following sections, I utilize and refer to some of these layers to describe the underlying research design and justify some of the choices. The underlying research design was shaped to explain and predict the phenomenon of interest (e.g, Recker 2012) – that is, exploring and explaining the effects of control, embodied in agile practices, on team autonomy and team performance in agile ISD teams. The focus of designing an appropriate research design was on a methodological choice (see Figure 1-3), i.e., the application of either quantitative, qualitative or a mix of both. However, the choice of appropriate research method(s) is closely related to an underlying philosophy; for example, surveys and experiments are inherently more positivistic, while others such as ethnography or action research are more interpretive (Creswell 2009; Saunders et al. 2016). To select appropriate methods, Yin (2003a) emphasizes the need to consider a) the type of the underlying RQ(s), b) the extent of control an investigator has over behavioral events, and c) the degree of focus on contemporary as opposed to historical events (see Table 1-7).

Strategy	Form of Research Question	Control of Behavioral Events?	Focus on Contemporary Events?
Experiment	How, why?	Yes	Yes
Survey	Who, what, where, how many, how much?	No	Yes
Archival Analysis	Who, what, where, how many, how much?	No	Yes/No
History	How, why?	No	No
Case Study	How, why?	No	Yes

Table 1-7: Relevant situations for different research strategies (Yin 2003a)

Applying Yin's proposed framework for selecting appropriate methods has the following implications for this research. First of all, as presented in section 1.1.1, the RQs put emphasis on the '*how*', e.g., how control is achieved in agile ISD and how this affects agile teams and agile projects. Furthermore, it can be assumed that basically no control of *behavioral events* is possible (agile team members cannot be influenced, manipulated or be assigned with tasks) and the research team will only act in the role of a 'silent observer'. Finally, the focus is on *current events*, the study examines how control is currently (or will be in the future) executed in practice in combination with agile methods, a historical outline, e.g., to identify developing trends, is not in the scope of this work.

Summarizing these aspects, the framework recommends the use of a mixed-methods approach, or more precisely, the use of surveys and case studies to gain knowledge. However, mixed-methods approaches come with a variety of advantages as well as disadvantages (Venkatesh et al. 2016). Some of the challenges include the need for extensive data collection, time-consuming procedure of analyzing both textual and numeric data, and lastly, the requirement that researchers need to be skilled to handle quantitative and qualitative methods (Creswell 2009). A common argument advocating the use of mixed-method approaches is that a focus on exclusively quantitative or qualitative methods may not be appropriate or even sufficient. In terms of the relevant philosophical perspectives, this represents a pragmatism viewpoint, which views an exclusive adoption of one philosophical position as unhelpful, allowing for 'freedom' of positions in order to accomplish research in the best way possible (Nastasi et al. 2010). In other words, both quantitative and qualitative research are appreciated by pragmatists and "the exact choice will be contingent on the particular nature of the research" (Saunders et al. 2016, p. 169). Recent research in the field of agile ISD made use of this aforementioned point of view, such as the investigation agile ISD agility by Lee & Xia, who underscore that the integration of both qualitative and quantitative methods overcomes weaknesses of each by maintaining (statistical) objectivity while providing a deeper understanding of contexts (Lee and Xia 2010a). This research follows this view and sees the use of both qualitative and quantitative methods as useful supplements rather than complements. Moreover, this research utilized the guidelines for conducting mixed-methods research as proposed by Venkatesh et al. (2016). Table 1-8 summarizes the decisions that have been made to properly conduct a mixed-method approach and investigates some of the aspects more in detail (see also Appendix B – Decision tree for (mixed) method choice for more information).

Step	Description	Comment on Decision
1: Decision on "appropriateness"	RQs, purposes, paradigmatic views, and contexts must be considered to decide on the appropriate-ness of a mixed-methods approach	• *RQ*: more than one RQ, mainly "how" questions, dependent on each other, predetermined • *Purpose*: 1) completeness (i.e., to gain a complete picture of phenomena, 2) developmental (i.e., to develop hypotheses that can be tested in the next strand of a study), 3) corroboration and triangulation (i.e., to enhance credibility of inferences from a single approach), 4) compensation (i.e., to eliminate weaknesses of one approach by using another one) • *Epistemological perspective*: multiple paradigms, mainly pragmatism and critical realism • *Paradigmatic assumption*: dialectic paradigm stance (i.e., important paradigm differences should be intentionally used together to engage meaningfully difference) and alternative paradigm stance (initiation of a paradigm that embraces and promotes the mixing of methods)
2: Develop "strategies" for research design	Strands/phases or research, priority of the methodological approach, design investigating strategies, mixing strategies and time orientation decisions relate to each other, but can also be independent	• *Strands/phases:* multistrand design (i.e., each phase of the research project encompassed all stages from conceptualization to inference • *Priority of methodological approach:* equivalent status design (i.e., both qualitative and quantitative approaches are used equally to investigate the phenomena of interest) • *Design investigation strategies:* mainly confirmatory, with minor exploratory aspects • *Mixing strategies*: fully mixed-method design (i.e., qualitative and quantitative research is involved through all components of a study, for example objective, data collection and analysis, type of inference etc.) • *Time orientation:* sequential exploratory (i.e., the study's qualitative phase follows a quantitative phase)
3: Develop strategies for collecting and analyzing data	Development of data collection (e.g. participants or number of participants) and data analysis	• *Sampling Design strategies:* basic mixed-methods probability sampling strategy (i.e., sampling units that are representative of the population were randomly selected) • *Data-collection strategies:* qualitative (semi-structured, non-numeric data) and quantitative (predetermined closed questions, numeric data) strategies • *Data-analysis strategies:* sequential qualitative-quantitative data analysis (i.e., one analyzes qualitative data then quantitative)
4: Draw "meta-inferences"	Inferences can be developed inductively, deductively or abductively	• *Theoretical reasoning:* mainly deductive theoretical reasoning, as strong theories exist, and hypotheses are tested ("confirmatory design investigation strategy")
5: Assessing "quality" of meta-inferences	Examination of inference quality including design and explanatory qualities	• *Inference quality:* design quality (i.e., the extent to which appropriate approaches to answer the RQs have been applied) addressed e.g. through consideration of other guidelines such as Yin's framework for selecting appropriate research methods (Yin 2003a) and explanatory quality (i.e. the degree to which credible interpretations of the obtained results have been made, e.g. through usage of certain reliability measures such as Cronbach's alpha or Proportional Reduction in Loss (PRL)) have been considered simultaneously

Table 1-8: Guidelines for conducting mixed-methods research (Venkatesh et al. 2016)

Summarized, this research follows a mixed-method approach for reasons such as completeness, enhancing credibility or compensation of one method's potential weakness, addressing pragmatism and critical realism philosophies. While the underlying investigation strategy is mostly confirmatory, we follow the guidelines of Creswell (2009) by using a sequential exploratory strategy (i.e., within a first phase we collect and analyze qualitative data to adjust our research model and propositions, followed by a second phase of quantitative data collection and analysis that builds on the results of the first phase). Data is collected utilizing qualitative and quantitative methods, while each data collection approach addresses different populations at one single point of time (cross-sectional design). Inferences are drawn by mainly deductive theoretical reasoning, as established and well-developed theories exist (e.g. control theory;Jaworski 1988; Kirsch 1996; Maruping et al. 2009a; Ouchi 1977; Snell 1992), have been used as theoretical lens and on which hypotheses have been tested.

1.3.2 Data Collection Approaches

The previous section informed about this study's overarching research design and introduced into mix-method approaches in general, implying that different methods have been used to collect data. The following sections provide insights about how data was collected. Table 1-9 lists the main characteristics of each study's underlying data collection approach and provides references to its methodological guidelines. However, more detailed information about collection (and analysis) approaches for each study are provided within the paper summary sections (see section 1.4.).

Study	Summary of Data Collection Approach	References
2) Journey Towards Agility – A Retro- and Prospective Review	▪ Stuctured literature review including an extensive keyword search (one search term) performed within title, keywords and abstract of an outlet ▪ Outlets: focus on primarily high quality, peer-reviewed literature, published in journals of the "Senior Scholars' Basket of Journals" and the AIS Toplist (IS, Management and Computer Science) and also articles of prominent conferences (e.g., HICSS, ICIS, ICSE) ▪ Exclude criteria: non-English language, no full text available, not research-focused (e.g., an opinion or commentary), not within our focused timespan of 01.01.1985 to 31.12.2017, not investigating agile ISD. ▪ after removing duplicates, our final set of articles consisted of 678 journal articles and 698 articles in conference proceedings, totaling up to N=1,376 articles	Levy and Ellis (2006); Miles and Huberman (1994a); Webster and Watson (2002a); Saldaña (2016)
3) Do As You Want Or Do As You Are Told? Control vs. Autonomy in Agile Software Development Teams	▪ Stuctured literature review including an extensive keyword search (one search term) performed within title, keywords and abstract of an outlet ▪ Databases such as EBSCOhost, INFORMS, Science Direct or ProQuest have been considered ▪ Exclude criteria: not peer reviewed, non-English language, not applicable to ISD/agile ISD and control; there was no restriction for the publishing year of the articles. ▪ after removing duplicates, the final set of articles consisted of N=28 relevant articles ▪ Additionally, data was gathered within a single case of eight different student development teams, the team level served as the unit of analysis. ▪ Inquiry followed a combination of intensity and convenience sampling strategy ▪ Semi-structured interviews (one per team) have been conducted, resulting in a final set of N=8 interview transcripts.	Levy and Ellis (2006); Miles and Huberman (1994a); Patton (1990); Webster and Watson (2002a)
4) „Loosening the Reins": Balancing Control and Autonomy in Information Systems Development	▪ Five cases, three large insurance companies (two of them are international companies while one focuses on the German market only. ▪ Semi structured interviews (N=37) and project documentation (e.g. project briefs, project plans, ressource plannings, organigrams etc.) were primary data sources. ▪ Regarding interview data, different roles (e.g. software developers, software and business analysts, test managers, project managers, agiles coaches etc.) have been considered. ▪ Data collection took place between July 2018 and November 2020	Dubé and Paré (2003); Lee (1991); Myers (2013); Yin (2003a);

	▪ Interviews were conducted using a guideline (derived from literature on control and teamwork) and lasted 60 minutes in average.	
5) "Directing Self and Others": An Empirical Study of Control in Agile Information Systems Development	▪ Survey data was obtained from an overall of 286 individuals participating in 89 agile ISD projects, which were conducted by mainly medium to large international companies (focus on development, improvement, customization, or the implementation of information systems). ▪ Distinction between supervisors (representing team-level of investigation) and corresponding team members (representing individual level), thus two different questionaires were developed and rolled out. ▪ Both supervisor and team member responses had to built matched-pairs, resulting in a set of N=148 completed matched-pairs data records (supervisor and corresponding team members) from 66 different teams (supervisors). ▪ Data collection took place between between August 2020 and March 2021 in different countries across three continents ▪ Software tool 'Qualtrics' was used for data collection, both surveys were provided the respondents via a link included in an invitation email.	Ko et al. (2005); Recker (2012); Saunders et al. (2016); Straub (1989)
Note: *Study I is not listed as it represents conceptual groundwork and serves as a proposal for the dissertations underlying research design*		

Table 1-9: Summary of data collection approaches

1.3.2.1 Structured Literature Review

For Studies II and III a literature review approach was part of the data collection process. The approach followed the guidelines of Webster and Watson (2002a) and Levy and Ellis (2006) to ensure a structured procedure. A SLR can be defined as "the use of ideas in the literature to justify the particular approach to the topic, the selection of methods, and demonstration that this research contributes something new" (Hart 1998, p. 1). Some researchers even go a step further and see the goal of an SLR as identifying links between existing studies, for example, "to explain how one piece of research builds on another" (Shaw 1995, p. 326), or as an essential prerequisite for building a "firm foundation for advancing knowledge" (Webster and Watson 2002a, p. xiii). In Study II, the aim of the SLR was the creation of a huge database of research articles, which dealt with the topic of agile ISD. Therefore, the search was rather broad in nature, that is, we included a broad range of journal articles and conference proceedings having their focus on either computer science or information systems related topics (e.g., outlets like Information Systems Research (ISR), IEEE Software or Hawaii International Conference on System Sciences (HICSS)). We conducted a keyword search and limited our exclude criteria to a minimum. The purpose of the literature review of Study III was to identify agile practices that implied a relationship with the exercise of control. Specifically, articles should be retrieved that provide information on the ability of agile practices to be associated with modes of control (Kirsch 1996; Kirsch 1997).

1.3.2.2 Case Study Research

A case study can be understood as an "in-depth inquiry into a topic or phenomenon within its real-life setting" (Yin 2003a). Generally, these in-depth inquiries can be designed to describe what happens and to explain for what reason(s) something is happening (Dubé and Paré 2003; Dubois and Gadde 2002; Saunders et al. 2016).

Study III employs a single case study design in order to explore potential and different relationships of agile practices (Fitzgerald 1997; Harris et al. 2009b; Recker et al. 2017; Tripp and Armstrong 2014) and control modes (Kirsch 1996; Kirsch 1997; Kirsch et al. 2002; Ouchi 1979; Ouchi 1980). The case is represented by eight student teams, who participated in development projects with different industry partners and applied Scrum as an agile approach. Objective data such as logs, project schedules, code repositories have been accessed and analyzed as well as field observations were conducted.

Study IV follows a positivist epistemology and employs a deductive and theory testing embedded multiple case study design (Dubé and Paré 2003; Lee 1991). In particular, nine teams in five projects across five different organizations have been investigated following the guidelines of Lee (1991); Myers (2013, pp. 24-25); Yin (2003a, p. 52), wheras each case represents one different case organization. Semi-structured interviews and project documentation were the primary mechanisms of data collection. From July 2018 to November 2020 a total sum of 37 face-to-face interviews have been conducted with an average duration of about 60 minutes. An interview guideline was used, but not shared with participants beforehand, as it only served as a broad guide for further and partly spontaneous discussion. More details about data collection can be found in the corresponding paper summary section (1.4.4).

1.3.2.3 Survey Data

In order to test the research model and hypotheses as proposed by Study V, survey instruments for two different online questioniares were developed. The use of two distinct questionnaires reduces potential problems arising from single informant and common method bias (Ko et al. 2005). The underlying sample data was collected via online questionnaires from an overall of 286 individuals participating in 89 agile ISD projects, resulting in a final set of 148 completed matched-pairs data records (supervisor and corresponding team members), from 66 different teams.

1.3.3 Data Analysis Approaches

After data has been collected, the next step within a research endeavor is usually to process and analyze the data that is, to make them useful by turning them into the researcher's desired information (Creswell 2009; Recker 2012; Saunders et al. 2016). For qualitative data, this often means to analyze data inductively building from particulars to themes and interpret the meaning of the data. In contrast, quantitative data is often used for testing objective theories by, for example, measuring variables and examining relationships among them, using statistical procedures (Creswell 2009). This research utilizes both, qualitative and quantitative data, thus, the essential analyses procedures are introduced in this section. The following table (Table 1-10) summarizes different approaches, how quantitative and qualitative data was investigated, and results have been derived. Similar to our data collection approaches, detailed data analysis per study is provided within the paper summary section (1.4).

Study	Summary of Data Analysis Approach	References
2) Journey Towards Agility – A Retro- and Prospective Review	General: Inductive SLR and qualitative assessment of data for the identification of agile ISD related research topics (history and gaps), based on a data set of 1376 research articles.	Levy and Ellis (2006); Webster and Watson (2002a) Blei et al. (2003); Srivastava and Sahami (2009) Aggarwal and Zhai (2012); Debortoli et al. (2016b);Miles and Huberman (1994a); Saldaña (2016)
	Computer aided analysis (CAA): The study employed a CAA tool for *topic modelling,* which was used to discover and reveal hidden topics shared across research. In particular, *Latent Dirichlet Allocation* was performed in which each document is seen as a mixture of different topics and each topic has certain probabilities of generating keywords. Coding: The study utilized a two-step coding process, beginning with *descriptive coding* to further summarize the data's contents and followed by *pattern coding* to develop major themes from data. The outcome of the coding process is a final set of 26 topics and eight topic groups.	
	General: SLR for the identification of control-related agile practices, combined by a single case study of 8 student development teams, following an exploratory approach.	

3) Do As You Want Or Do As You Are Told? Control vs. Autonomy in Agile Software Development Teams	SLR: A concept-driven and systematic literature review was employed based upon a final set of 28 research articles, a concept-matrix brought together agile practices and their relation to control related concepts. These various descriptions served as initial seed codes for coding of our interview data in step 2. Coding: Likewise, as in Study II the study utilized a two-step coding process, beginning with *descriptive coding* to further summarize the data's contents and followed by *pattern coding* to develop control related concepts and categories from data. This enabled us to link agile practices to different modes of control. Consequently, the outcome of the coding process is a final set of eight agile practices and their corresponding control modes.	Theoretical lens(es): Lenses of 'agile practices' (Recker et al. 2017; Tripp and Armstrong 2014), and 'control theory' (Kirsch 1996; Kirsch 1997) including the extension of 'emergent outcome control' (Harris et al. 2009a) served as guidelines and helped to build meaningful categories for mapping practices to control modes	Levy and Ellis (2006); Miles and Huberman (1994a); Saldaña (2016); Webster and Watson (2002a); Wolcott (1994, p. 55)
4) „Loosening the Reins": Balancing Control and Autonomy in Information Systems Development	General: Theory-testing, embedded, multiple-case study of nine teams in five projects across five different organizations, following a hypothetico-deductive logic in the case methodology, emphasizing "falsification" over "confirmation" through deduction.		Benbasat et al. (1987); Lee (1991); Myers (2013); Yin (2003a);
	Coding: Within a first step, *initial coding* was performed served as a starting point to get first ideas of concepts within the cases. To reduce and abstract the multitude of codes, *structural coding* was employed. To get more insights about the manifestation of the control and team related concepts (e.g., intensity), *magnitude coding* was applied. Step two followed *hypothesis coding* in order to identify statements in the interviews to support or refute this study's propositions. Coding techniques and checklists were afterwards used to connect data with constructs from our model, and the propositions.	Theoretical lens(es): Lenses of 'agile practices' (Recker et al. 2017), the 'expanded theoretical framework of IS project control' (Wiener et al. 2016), and 'control theory' (Kirsch 1996; Kirsch 1997) served as guidelines and delivered initial seed codes. Within the second coding step, they served for coding the interview data, providing descriptions for a codebook.	Lee (1991); Miles and Huberman (1994a); Saldaña (2016); Russell Bernard (2002); Weber (1990a);
5) "Directing Self and Others": An Empirical Study of Control in Agile Information Systems Development	General: Theory testing through hypothesis testing, matched pairs questionaire based field study, utilizing 148 matched-pair datasets.		Heumann et al. (2012); Tashakkori and Teddlie (2009) Dijkstra and Henseler (2015); Hair et al. (2017); Jöreskog (1967); Rigdon et al. (2017)
	SEM: In this study data was assessed utilizing a co-variance based SEM approach. The model was estimated using the maximum likelihood algorithm while considering multilevel data in form of nested data (team members) within teams (represented by supervisors)	Theoretical lens(es): Lenses of 'agile practices' (Recker et al. 2017), the 'expanded theoretical framework of IS project control' (Wiener et al. 2016), and 'control theory' (Kirsch 1996; Kirsch 1997)	

Table 1-10: Summary of data analysis approaches

1.3.3.1 Coding Strategies

In this research different coding strategies have been applied as a problem-solving technique. A code in qualitative inquiry is usually a term or short phrase, which is assigned to language-based, audio or even visual data (Saldaña 2016). Consequently, 'coding' or in other words, 'assigning' these labels to data can be understood as a "process of organizing the material into chunks or segments of text in order to develop a general meaning of each segment" (Creswell 2009, p. 227). Coding has an iterative character, as it is very often done in different cycles. For example, recoding can be done using the same first cycle method but with a slightly adapted perspective or theoretical lens, while a second cycle method may describe those processes (or codes) that might be employed during the second (and perhaps third, fourth…) review of data (Saldaña 2016). Table 1-11 provides a brief description of different coding methods which are employed by the different studies. How the different coding techniques were used and what purpose they served is explained in more detail in the corresponding paper summaries (see section 1.4).

Method	Cycle	Description	Study?
Descriptive coding	First	Sometimes called 'topic coding', this approach summarizes with a single word or short phrase the basic topic of a passage of data (Saldaña 2016). The outcome of descriptive coding can be compared to a 'base vocabulary' or 'bread and butter' for further analysis (Turner 2002).	Study II; Study III
Pattern coding	Second	Pattern coding is appropriate for the development of major themes from data (Miles and Huberman 1994a; Saldaña 2016). The codes generated in this step support the identification of emergent themes and therefore are helpful for grouping and summarizing them into a smaller number of sets, themes, or constructs (Miles and Huberman 1994a, p. 69).	Study II; Study III;
Initial coding	First	Initial coding (sometimes referred to as 'open coding') serves as a 'starting point' by guiding researchers in their further analytic endeavors. Passages of qualitative data is split into discrete parts, investigated and compared to other (initial) codes within the coding system. Initial codes are usually tentative and provisional in nature (Saldaña 2016; Strauss and Corbin 1998)	Study IV
Structural coding	First	Structural Coding usually adds labels representing a topic of inquiry to a segment or passage of data that relates to a specific research question used to frame the interview. This approach is particularly useful, when multiple participants, standardized or semi-structured data-gathering protocols, hypothesis testing, or exploratory investigations to gather topics lists have to be utilized (Saldaña 2016).	Study IV
Magnitude coding	First	Magnitude coding usually labels additional codes to existing codes for providing more information, e.g., about its intensity, frequency or presence. For example, magnitude codes indicating some evaluative content could range from "agree", to "neither agree/nor disagree" to "completely disagree". Magnitude Codes can be placed in tables or matrices for further analyses (Miles and Huberman 1994a; Saldaña 2016)	Study IV
Hypothesis coding	Second	Hypothesis Coding uses predetermined list of codes onto qualitative data to assess a researcher-generated hypothesis and is therefore appropriate for hypothesis testing. Hypothesis Coding can also be applied in middle or final stages of qualitative research projects to confirm or falsify any assertions or theories (Saldaña 2016; Weber 1990b)	Study IV

Table 1-11: Summary of coding methods

1.3.3.2 Structural Equation Modelling

Study V proposes a research model which includes insights from Study I to IV and addressed the need for a quantitative evaluation of the former results. The research model was transformed into a structured equation model (SEM) and estimated using the maximum-likelihood algorithm, following a covariance-based (CB) SEM approach. The data analysis approach has taken into account multilevel data, as it exists in our case in form of nested data (team members) within teams (represented by supervisors). The model met most of the common fit indices and represented an overall good fit as recommended by Hu and Bentler (1999) and Kenny et al. (2014).

1.4 Paper Summary

1.4.1 Study I – Agility in the Balance: Control, Autonomy, and Ambidexterity in Agile Software Development

The first study of this dissertation has three goals. First, this study aims to highlight the need to conduct research on the topic of control in agile ISD by showing important gaps (e.g., Chua et al. 2012; Cram and Brohman 2013; Wiener et al. 2016) in literature and justifying why this is an important topic. One of the central arguments is that agile ISD projects can benefit from both, team autonomy (e.g., Lee and Xia 2010a; Vidgen and Wang 2009) and the exercise of control (e.g., Maruping et al. 2009a; Persson et al. 2011), and that both need to be considered and applied in agile ISD, without each negatively affecting the other. Additionally, the study relies on the assumption that different control modes are embodied in different agile practices, allowing for 'control' through the application of these practices (for further details see 'Study III'). In line with the overarching research question "*How do control styles, agile practices, and modes of control affect each other and how do they influence an agile ISD team's autonomy and performance?*", this research addresses the elementary control and team related concepts by posing the following two broad research questions[4]:

(1) To what degree do modes and styles of control, as enacted through agile practices, influence team autonomy and team task performance?

(2) What are the effects of team autonomy and team task performance on ISD ambidexterity in agile development environments?

The study's underlying method is a SLR following the guidelines of Webster and Watson (2002a) as well as of Levy and Ellis (2006). Based on the identified set of research articles dealing with control in agile ISD an extensive concept-matrix was developed which a) summarized all relevant control and team related concepts for agile ISD and b) summarized gaps and future research directions of this topic.

The resulting insights allowed to achieve the studies second goal that is, proposing a preliminary research model, which a) identifies important concepts to explain how autonomy and control affect agile ISD and b) how these concepts relate to each other and are subject to specific interdependencies (see Figure 1-4).

[4] The concepts mentioned in the research questions and in the remainder of this paper are introduced and explained in more detail in Section 1.2.3, so a detailed explanation will not be provided here.

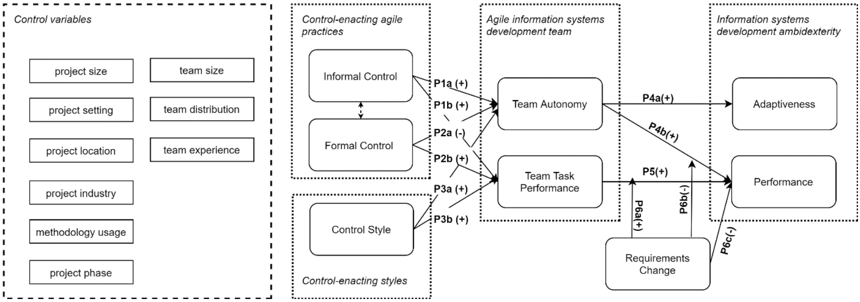


Figure 1-4: Proposed research model (Study I)

Third, to investigate the proposed research model and its corresponding propositions, this study recommends using a mixed method approach, incorporating both qualitative and quantitative research methods, as introduced in Section 1.3.. Table 1-12 summarizes the research model's underlying theoretical assumptions:

No.	Description
P1	Greater use of informal controls positively impacts (a) team autonomy and (b) task performance.
P2	Greater use of formal control negatively impacts (a) team autonomy, while it positively affects (b) task performance.
P3	Greater degrees of an enabling control style positively affect (a) team autonomy and (b) task performance.
P4	Greater degrees of team autonomy positively affect (a) adaptiveness and (b) project performance.
P5	Task performance positively impacts project performance.
P6	When requirement changes are high rather than low, (a) the positive relationship of task performance on (agile ISD) performance is strengthened, (b) the positive relationship of team autonomy on (agile ISD) performance is mitigated and (c) project performance is negatively affected.

Table 1-12: Summary of propositions (Study I)

In particular, the study proposes to follow a sequential exploratory strategy (Creswell 2009), i.e. to start by conducting in-depth case studies (Yin 2003b) in order to collect and analyze qualitative data to adjust our research model and propositions. Within a second step, quantitative data is collected and analyzed that utilizes the insights derived from qualitative data of the first step). According to Hunter (2004), the recommended approach considers to test the feasibility of the preliminary research model in a manageable environment, before testing the model within a wider field. Regarding qualitative data, the study suggests to code data starting with pre-defined concepts derived from the model and to extend them with open coding to provide sufficient space for the identification of novel and hitherto unrevealed insights. As the primary quantitative data collection instrument, a survey-based approach is suggested, objective data sources will complement the survey data. Again, a pre-test of all survey instruments is suggested before extending it to a larger sample of national and international agile ISD projects. For the analysis of the entire quantitative data set, the study suggests using structural equation modeling (SEM)[5].

To summarize, this study represents an overarching overview about this dissertation by introducing to the topic and underlying research question, proposing a preliminary research model, and finally from a methodological perspective, approaches of how to operationalize this model and how to analyze the so obtained data in order to answer the before introduced research questions.

1.4.2 Study II – Journey Towards Agility – A Retro- and Prospective Review

The second study of Part I of this dissertation aims at gaining answers to the following questions:

(1) What research topics were addressed within the last three decades by agile ISD research and
(2) how do these topics differ in terms of available publications and their distribution over time?

Due to an apparent wide variety of topics covered by agile ISD research, comprising technical aspects (e.g., Balijepally et al. 2009), sociological or psychological factors (e.g., Maruping et al. 2015a), and range from an individual level to an organizational level (e.g., Zheng et al. 2011), a clear assignment of existing streams of research cannot be identified. Although there exist few summarizing or aggregating literature reviews, these are often limited in scope or are even outdated and do not include recent research in this field. Consequently, we conducted a structured and comparative literature review as described by the guidelines of Levy and Ellis (2006) and Webster and Watson (2002b), followed by computer-aided topic modeling (Aggarwal and Zhai 2012; Debortoli et al. 2016b) on the extant body of knowledge of agile ISD.

Our data collection process started by performing an extensive keyword search within leading journals. We set a focus on primarily high quality, peer-reviewed literature, published in journals of the 'Senior Scholars' Basket of Journals' and the AIS Toplist (including leading journals not only from IS but also Management and Computer Science). Additionally, we included articles of prominent conferences (e.g., HICSS, ICIS, ICSE). We constrained to agile ISD investigating articles of English language, with full text available, research-focused (e.g., opinion or commentaries were excluded), and articles within our focused timespan from years 1985 to 2017. In total, after removing duplicates and articles which did not meet our language requirements, our final set of articles consists of 678 articles matching our search indicators for agile ISD in journals and 698 articles in conference proceedings, totaling up to 1,376 articles.

Following to the data collection, we analyzed all articles with the help of the *Scikit-learn* software library (Pedregosa et al. 2011), a computer-aided analysis and text mining tool. From within the Scikit-learn suite of machine learning tools, we specifically used *topic modelling* (Aggarwal and Zhai 2012; Debortoli et al. 2016b), which is an easy-to-use technique to discover topics shared across research and therefore

[5] Although, the original setup of the study intended to include different data collection points in time (start, middle and end of projects), the measurement of qualitative as well as quantitative data was carried out at one point of time in both cases with the goal of gaining deeper insights (see Section 1.3 for further details).

to help in answering this study's research questions. In particular, research found topic modelling suitable for efficiently revealing hidden topics by classifying, summarizing, and clustering of large amounts of text (Maowen et al. 2012; Srivastava and Sahami 2009) and topic trends over time (Alghamdi and Alfalqi 2015). We utilized *Latent Dirichlet Allocation* (LDA; Blei et al. 2003) as implemented in Scikit-learn as a specific topic modeling approach. LDA has been used in various research studies (e.g. Chen et al. 2016) and has been suggested as a suitable and helpful tool for research (Debortoli et al. 2016b). In this approach each document is seen as a mixture of different topics and each topic has certain probabilities of generating keywords. Keywords are allowed to occur in more than one topic.

Finally, and to make sense out of our sets of keywords, we applied a two-step coding cycle approach with different coding strategies as an exploratory problem-solving technique and to link our keywords to patterns, resulting in meaningful topic descriptions following the guidelines of Saldaña (2016). The outcome of this approach was a set of algorithm-generated keywords, each representing a different topic covered by certain research articles. Giving that as a starting point, *descriptive coding* was performed in a first coding cycle. Descriptive coding primarily leads to a categorized summary of the data's contents and builds the groundwork for second cycle coding and further analyses (Wolcott 1994, p. 55). So, this cycle's purpose was 1) to reduce complexity by summarizing and merging keywords (e.g., keywords which were seen as synonyms) and 2) to generate summarizing phrases, which are conducive to approximating a meaningful topic description of our different sets of keywords. Building on this, we applied *'pattern coding'* as a second cycle coding method. Its purpose was the further reduction of complexity and the extraction of major themes from data (Miles and Huberman 1994a; Saldaña 2016). Within this step, the authors tried to assign topics to broader topic groups. Like first cycle coding, pattern coding was conducted twice until consensus was reached. We completed the coding process within a final step, in which we did some post-coding activities such as fine-tuning of the wording and alphabetical order of the results. The outcome of the coding process is a final set of 26 topics and eight topic groups.

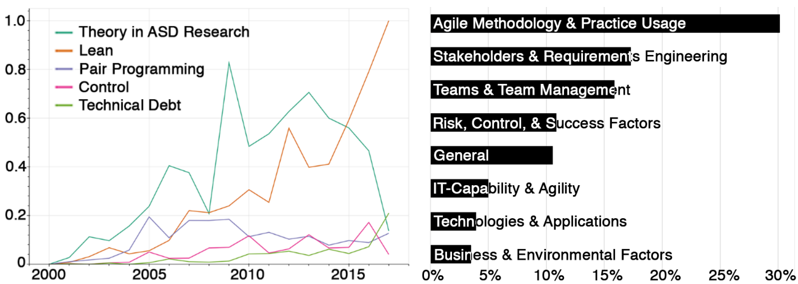


Figure 1-5: Topics over time and topic groups distribution

Based on our set of topics and their corresponding topic groups, we found several important insights. First, we were able to reveal research foci over the last three decades of research in the field of agile ISD. For example, the topic group, 'Agile Methodology & Practice Usage', summarizes the 'basics' of agile ISD and includes keywords centered around agile ISD methodologies, concepts, practices, management, and tasks. This topic group is ranked first in terms of the overall distribution (32.35%). In contrast, the topic group 'Risk, Control & Success Factors in Agile' entails risk assessment, quality and success factors, as well as control related content and is ranked average compared to the other groups (see Figure 1-5). Second, we were able to identify key outlets and articles, for example by comparing the number of publications of more computer science-oriented conferences (e.g., ICSE, XP/Agile) against information systems oriented conferences (e.g. AMCIS, ICIS).

Lastly, we were able to uncover trends and their implications to research by uncovering gaps and providing future research directions in this field. In particular, we provided topics that gained popularity over time (e.g., 'Lean') or lost attraction (e.g., 'Theory in agile ISD research') over time. In terms of potential research gaps, we found the 'social' aspect of 'socio-technical systems' needs to be embraced more by researchers. Similar to the first point of our research agenda, our data shows clearly a lack of research on this aspect of agile ISD, as no single topic group focuses on social aspect. For example, this study shows that research on the effects of agile ISD on control (see Topic Group 'Risk, Control, & Success Factors in Agile') could complement existing similar information systems research streams and answer calls for research (e.g., Lee and Xia 2010b; Wiener et al. 2016). Particularly, because this topic was also found to have developed consistently only at a rather low level over time, this contributed in our motivation to conduct further studies on the topic of control in agile ISD.

1.4.3 Study III - Do as You Want or Do as You Are Told? Control vs. Autonomy in Agile Software Development Teams

The third study of this dissertation focuses on the control enactment process by investigating how control can be exercised through agile practices and how these practices affect either formal or informal controls. The study's underlying research question is as follows:

> *"How can control be enacted in agile software development (ASD) projects through specific agile practices and how do they affect different types of control (i.e., formal and informal control) within an agile ISD team?"*

The study starts from the basic assumption that agile practices embody 'control' as such, in particular that different modes of control can be addressed by agile practices. To further explore this assumption and get insights in order to answer the research questions above, this research project followed a three-step data analysis approach as shown in Figure 1-6.

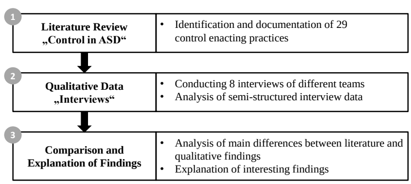


Figure 1-6: Research design of Study III

Within a first step we conducted a concept-driven and systematic literature review based on the approaches of Levy and Ellis as well as Webster and Watson (Levy and Ellis 2006; Webster and Watson 2002a). In contrast to Study II, the search was more narrowly designed. The search term of the initial keyword search was designed to only include articles, which deal with topics of agile ISD and control. Based upon the results of the keyword search, we proceeded with a reference, author and keyword backward search. Again, based on the results of the previous step, we accomplished our search process by conducting a reference and author forward search, revealing a final set of 28 research articles investigating agile ISD and some form of control. Based upon this set of articles we identified 29 agile practices, which could be related to some form of control. In detail, we distinguished between practices, which could be associated with at least one control mode, containing formal outcome- and behavior-control as well as informal self- and clan-control (see Table 1-4: Summary of control modes for further information). As a theoretical lens, we utilized mainly control theory (e.g., Kirsch 1996; Kirsch 1997; Kirsch 2004). Additionally, we included the concept of *emergent outcome control* (EOC) as an extension to common control modes (Harris et al. 2009a; Harris et al. 2009b). Harris et al. (2009a) proposed the concept of EOC to achieve a better product-market match, as they argue formal outcome control to be insufficient in agile environments. EOC therefore uses scope boundaries and ongoing feedback to "define the allowable space for exploration" and "check on decision as they are made throughout the development process" (Harris et al. 2009b, p. 405). Our revealed agile practices do not belong to a distinct agile methodology but cover several agile methodologies such as Kanban, Scrum or XP. Within the final set of 29 practices, we identified 17 of them which affect behavior control and clan control, followed by 15 practices that are suitable to enable outcome control. 11 practices could be linked to address EOC, while 12 practices found to support self-control in agile ISD teams. No practices could be found that enabled input control.

In step 2 we wanted to evaluate whether our findings of step 1 can be applied to practice. The study utilized a single case study design to get more insights about practices (Fitzgerald 1997; Harris et al. 2009b; Recker et al. 2017; Tripp and Armstrong 2014) and there potential relationship to control modes (Kirsch 1996; Kirsch 1997; Kirsch et al. 2002; Ouchi 1979; Ouchi 1980). The 'case' is represented by a lecture course in which eight teams of an average of five student software developers work on different software development projects. All teams had to apply the agile methodology Scrum and to use certain agile practices like daily-stand ups, user stories or backlog prioritization (Cohn 2010; Cram and Brohman 2010; Harris et al. 2009b; Tripp et al. 2016). Deviations from these requirements were sanctioned by the instructors. Generally, instructors acted as proxy product owners, representing the customers needs, while on the other hand they served as agiles coaches, supporting the student teams in the proper application of the methodology. From a control perspective, the lecturors represented the controller part in a controller-controllee dyad, whereas the student teams took on the role of the controllees. This 'case' is an interesting object of study, as it underlies a special control relationship in an educational environment, in which small agile teams of students are controlled by the faculty. Since one of the authors was also a lecturer, time, effort, money, and time could be saved because 'access' to the case was granted immediately. Thus, the sampling strategy can be seen as a mix of intensity/theoretical and convenience sampling (Dubé and Paré 2003; Miles and Huberman 1994a; Patton 1990). Semi-structured interviews (usually two or three per team members at the same time, providing consolidated answers) have been conducted by the second author within his master thesis. A guideline was used to perform the interview, which contained a first part of open questions and one second smaller part consisting more structured questions. All interviews took 45 to 60 minutes in average, resulting in a final set of eight interview data transcripts. We used a concept matrix based on different categories to get a better picture of how to differentiate between practices that enable different types of control (or control modes). Step 2 focused on a selection of eight agile practices. These practices were chosen for two reasons: (1) the selected practices are supported by literature to enact different control modes and (2) the selected practices cover a broad range of control modes according to control theory (Choudhury and Sabherwal 2003; Kirsch 1997). Consequently, most of the selected practices have their origin in Scrum, XP, and custom hybrid approaches as they represent more than two-thirds of agile methodologies used in software projects (VersionOne 2016).

Based on the results of both steps, we were able to compare both results in step 3 in order to identify major findings and insights. Summarized, we identified within our review's results a set of 23 agile practices that can be linked with the enactment of formal control types such as outcome or behavior control. In contrast, we found only 20 agile practices suitable for fostering informal control types such as clan control and self-control. Whereas 12 practices are dedicated to formal control types, there are 3 practices that affect informal control types only. The study revealed that the usage of a certain sets of common agile practices potentially enacts high amounts of formal control within an agile ISD project. This let the authors conclude that the enactment of formal control can be seen as a supplement rather than a complement of informal controls and formal controls as an important counterbalance to flexibility and discretion granted by high degrees of team autonomy. From a practical point of view, this list of practices allows practitioners working in agile ISD projects to evaluate the existing practices for general suitability and implementation fit within their projects. Table 1-13 shows an excerpt of the complete list by concentrating on agile practices and their associated control modes based upon an empirical investigation.

No.	Agile Practice	Control Mode		#	FREQ.
1	User stories	Formal	BC, OC	7	5,71
2	Iteration Retrospective	Formal	BC	2	4,28
		Informal	CC	5	
3	Burndown charts	Formal	BC, OC	4	4,14
		Informal	CC	3	
4	Pair programming	Informal	CC	7	4
5	Backlog prioritization	Formal	BC, OC	4	3,85
		Informal	CC	3	
6	Code reviews	Informal	CC	7	3,71
7	Daily stand-ups	Formal	BC	2	3
		Informal	CC, SC	5	
8	Collective code ownership	Informal	CC	7	2,85
LEGEND: Control Modes: BC = Behavioral Control, CC = Clan Control, EOC = Emergent Outcome Control, OC = Outcome Control, SC = Self-Control; Freq.: frequency of usage (6 is high)					

Table 1-13: Agile practices associated to control modes (step 2)

1.4.4 Study IV - „Loosening the Reins”: Balancing Control and Autonomy in Information Systems Development

The focus of this study lies on the exploration of control enactment in agile teams and how these controls affect team autonomy and team performance. The study builds on the findings of Study III (see section 1.4.3) by asserting that common formal and informal modes of control can be addressed or implemented through the application of agile practices (e.g., Harris et al. 2009b; Persson et al. 2011). In addition to control modes, the study also considers two other important aspects within the control enactment concept as suggested by Wiener et al. (2016). First, control style ("the manner in which the interaction between the controller and the controllee is conducted") and second, the concept of control congruence, (degree of understanding" between a controller's and controllee's perception of the exercise of control) (Narayanaswamy et al. 2013, p. 192; Wiener et al. 2016). Moreover, the study claims that all three elements of the control enactment process influence an agile team in a different way. Generally, control is enacted to steer outcomes to achieve desired results, consequently team performance (the degree to which a team achieves its goals) is investigated (Hackman 1987; Zellmer-Bruhn and Gibson 2006). In contrast, extant research suggests a negative relationship between control and team autonomy, as it potentially limits the degree of discretion enjoyed by a team, for example, by putting managers in the position of most decision making (Gerwin and Moffat 1997; Piccoli et al. 2004; Remus et al. 2016). Summarized, the central research question guiding this study is:

"To what degree do control styles, control congruence, and modes of control embodied in agile practices influence team autonomy and team performance of agile teams?"

No.	Description	References (excerpt)
P1	Team autonomy is positively influenced by informal control.	*Henderson and Lee (1992); Kirsch et al. (2002)*
P2a	Team performance is positively influenced by informal control	*Chua et al. (2012); Gopal and Gosain (2010)*
P2b	Team performance is not affected by informal control	*Choudhury and Sabherwal (2003); Tiwana and Keil (2009)*
P3a	Team autonomy is negatively influenced by formal control	*Piccoli et al. (2004); Robey et al. (2000)*
P3b	Task autonomy is not influenced by formal control	*Cordery and Tian (2017); Feldman (1989)*
P4a	Team performance is positively influenced by formal control	*Kirsch et al. (2002); Remus et al. (2016)*
P4b	Team performance is not influenced by formal control	*Domberger (1998); Nickerson and Zenger (2004)*
P5	Control congruence is increased by an enabling control style	*Adler and Borys (1996); Wiener et al. (2016)*
P6	Team autonomy is positively influenced by an enabling control style	*Adler and Borys (1996); Wiener et al. (2016)*
P7a	Team performance is positively influenced by an enabling control style	*Choudhury and Sabherwal (2003); Remus et al. (2016)*
P7b	Team performance is positively influenced by an authoritative control style	*Maruping et al. (2015b); Wiener et al. (2015)*
P8	Team performance is positively influenced by control congruence	*Narayanaswamy et al. (2013); Nelson (2005)*

P9a	Team performance is positively influenced by team autonomy	*Cordery et al. (1991); Stewart (2006); Wall et al. (1986)*
P9b	Team performance is negatively influenced by team autonomy	*Langfred (2004); Maruping et al. (2009a)*

Table 1-14: Summary of propositions of Study IV

To answer this question, a research model is proposed, integrating insights of control theory (e.g., Kirsch 1996) as well as novel extensions made by the expanded theoretical framework of IS project control (Wiener et al. 2016). The study provides 14 propositions based on recent and ambiguous findings in literature, including certain rival assertions (see Table 1-14). The study follows a positivist epistemology and employs a deductive theory testing and embedded multiple case study design (Dubé and Paré 2003; Lee 1991). In terms of data collection, nine teams of five projects across five different organizations were subject to our investigation, consequently, each case represents one different case organization (Lee 1991; Myers 2013, pp. 24-25; Yin 2003a, p. 52). Out of these five cases, three of them are large insurance companies, whereas two of them are operating internationally and one focuses on the German market only. The remaining two cases are represented by two mid-sized software developing companies, both focusing on German clients only. The cases in this study were sampled following a theoretical replication logic as we expected the different case conditions to engender contrasting results (Dubé and Paré 2003; Yin 2003a). Cases were selected for the following reasons: the banking and insurance industry is regarded as culturally conservative, highly regulated, and capital-intensive (Gomber et al. 2018) and seen relatively slow in adopting innovation (Bohn 2018; Cappiello 2018; McKinsey 2017). Consequently, we expect a comparatively high degree of hierarchy and formal control within the organizations. The medium-sized software development companies are expected to reflect significantly flatter hierarchies and less formal control. Based on these distinctions, we expect to observe different characteristics of the control portfolio as well as the control styles exercised (and thus different results) across the case settings. Primary data sources are an overall amount of 37 interviewees and corresponding project documentation if available (e.g. project briefs, project plans, resource planning documents, organigrams etc.). Data collection activities took place between July 2018 and November 2020. Each interview lasted 60 minutes in average and was semi-structured in nature. Different roles such as software developers, software and business analysts, test managers, project managers, agile coaches (and more) have been considered.

Study IV is following a hypothetico-deductive logic in the case methodology, emphasizing ‘falsification’ over ‘confirmation’ through deduction. Consequently, different coding methods were utilized. The coding process started with *initial coding*, which served as a starting-point by providing ‘first ideas’, about certain control-related, team-related (e.g., linked to team autonomy or team performance) or even yet unidentified phenomena within the cases (Saldaña 2016; Strauss and Corbin 1998). To abstract and reduce the multitude of codes at the same time, *structural coding* was done, primarily in order to relate different codes to specific themes, for example, linkage of codes to different modes of control according to control theory (e.g.,Kirsch 1996; Kirsch 1997). After categories and subcategories could be formed in this step, which are representative for our relevant control-related and team-related concepts, the next step was to try to learn more about their manifestation within the different cases. We therefore applied *magnitude coding*, providing more information about the intensity, for example of the occurrence of formal control within a case. We distinguished between 'high', 'moderate' and 'low' manifestations of our constructs. Finally, we employed a fourth coding technique called *hypothesis coding*. In this step, we sought to identify statements in the interviews to support or refute our propositions. Essentially, hypothesis coding involves *pattern matching*, which enhances internal validity (Yin 2003a). This involves qualitative, logical deduction (Lee 1989) wherein an empirically-based pattern is compared against a predicted pattern derived from (rival) theoretical perspectives (e.g., Markus 1983). Intercoder reliabilities were calculated for each code drawing on the *proportional reduction in loss (PRL)* reliabilty measure, which is well above the recommended cut-off level (0.70) with a value of 0.95 (Rust and Cooil 1994). After reliabilities were calculated, the coders discussed and reached an agreement on remaining coding discrepancies in a face-to-face meeting. Figure 1-7 illustrates the coding process.

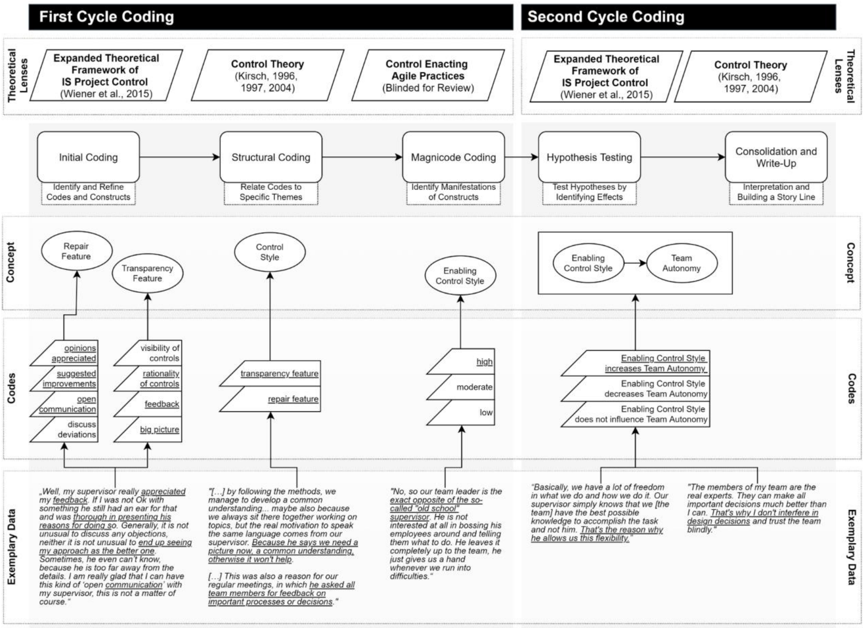


Figure 1-7: Coding process of Study IV

Figure 1-8 presents this study's proposed research model and shows results of the tested hypotheses. Regarding the control enactment process, the degree (ranging from 'low', 'moderate' to 'high') to which each of the concepts was identified within a first step. In a second step, questions how these concepts relate to each other and how they influence team mechanisms, such as autonomy and performance were adressed. During the overall assessment, the study distinguishes between three levels of support for claims in the data: a) 'unsupported', b) 'weakly supported', and c) 'supported'. A hypothesis is considered 'unsupported' if no evidence is found or if less than half of all cases provide supporting evidence. In turn, a hypothesis is considered 'supported' if greater than half of all cases show clear supporting evidence. 'Weak support', however, is present when there are attenuated or indirect indications that nevertheless leave some room for interpretation. The different colors used in Figure 1-8 inform about when a hypothesis was supported (green), weakly supported (yellow) or not supported (red).

Summarized, we found support for nine propositions whereas no evidence could be found for any falsifying (rival) propositions. Although the influence of *formal and informal controls* on team autonomy through the exercise of agile practices remains ambiguous (weak support only), it could be observed that such controls enabled by agile practices positively impact team performance (P2a & P4a).

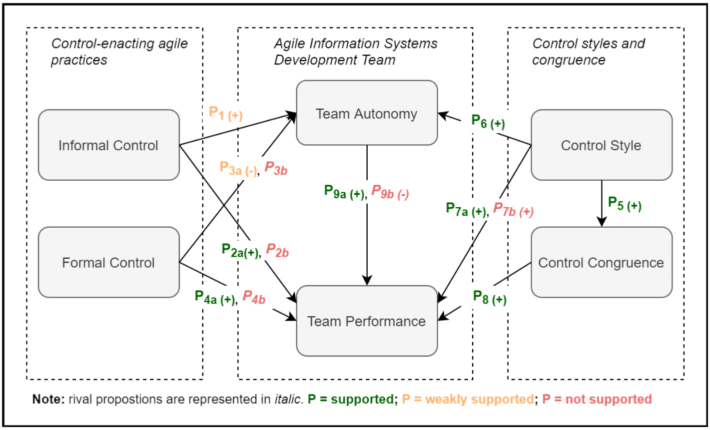


Figure 1-8: Research model and results of Study IV

Moreover, our findings reveal that *control styles* play an important role in establishing control portfolios and have a significant impact on agile teams. Three implications can be gleaned: 1) the presence of an enabling control style reduces the likelihood of an authoritative control style, 2) an enabling control style promotes a shared understanding (communicational congruence) and increases perceived appropriateness (perceptual congruence) of the controls enacted (P5) (Murungi et al. 2019; Narayanaswamy et al. 2013), and 3) the results underscore a positive influence of an enabling control style on both team autonomy (P6) and team performance (P7a), supporting prior findings in the literature. Regarding team autonomy, the two characteristics of 'repair' and 'transparency' in particular have a positive effect on flexibility and thus team autonomy (Adler and Borys 1996; Wiener et al. 2016). Likewise, these characteristics facilitate additional exchange of knowledge, regular feedback, and close collaboration within a controller-controllee dyad, leading to increasing team performance (Adler and Borys 1996; Remus et al. 2016).

Lastly, the findings underscores the importance of the concept of control congruence when control is exercised within agile teams. The results reveal that, in four of the five cases, a high level of control congruence positively impacts team performance (P8). This observation resonates with the results of recent studies, which indicate that control congruence contributes significantly to the quality of controls adopted and avoids negative socio-emotional effects, such as decreased job satisfaction (Narayanaswamy et al. 2013; Tripp et al. 2016).

1.4.5 Study V - "Directing Self and Others": An Empirical Study of Control in Agile Information Systems Development

The fifth and final study in this dissertation builds on the findings of the previous studies and undertakes a quantitative evaluation. In particular, Study V investigates the relationship between agile practice usage, control and team autonomy in agile ISD. Consequently, this study's central research question is as follows:

"How do control styles, agile practices, and modes of control influence an agile ISD team's autonomy?"

We transferred our research model into a structural equation model (SEM) and estimated this model using the maximum-likelihood algorithm, following a covariance-based (CB) SEM approach. Again, the study builds on the idea that formal and informal modes of control can be addressed by the usage of agile practices (e.g., Harris et al. 2009b; Persson et al. 2011). Therefore, the research model of this study considers the four control modes of clan, self-, behavior, and outcome control, reflected as single constructs. In context of agile practice application, the study considers three agile practices, namely, retrospective, pair programming, and code revisions. The choice of these practices was made mainly for three reasons, first, we sought to identify practices that could be identified in literature to potentially address control, either one distinct control mode or even a combination of more than one. Second, a pragmatic and common categorization of agile practices in the literature distinguishes between a) 'technical' or 'developmental' practices (e.g. continuous integration), b) 'management practices' which emphasize rules and behaviors for the exchange of information in meetings (e.g., daily stand-up meetings), and finally c) those practices that prescribe 'standards and norms' which are socialized among team members and that should be followed (Recker et al. 2017; Tripp and Armstrong 2014). Third, the selected practices had to be broadly used by our investigated teams and to be popular in general (VersionOne 2020). 'Agile Practice Usage' was developed as a second-order construct which included the aforementioned agile practices in order to best reflect agile working.

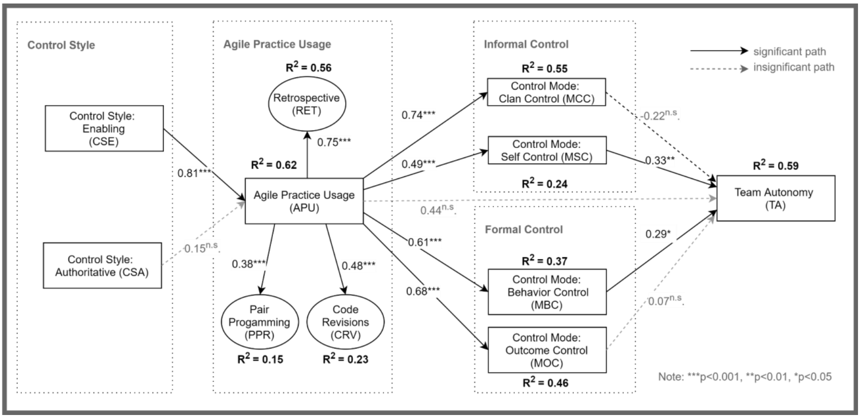


Figure 1-9: Research model and results of Study V

In addition to control modes, the study also considers 'control style' ("the manner in which the interaction between the controller and the controllee is conducted") as an important concept within the control enactment process (Wiener et al. 2016) (see section 1.2.3.2. for further details). We included both extreme endpoints of this continuum as independent variables in our measurement model (enabling style vs. authoritative style), as we believe that measuring both opposing styles provides more clarity and contributes to the accuracy of the measurement model. 'Team autonomy' is the outcome variable of interest and serves as the dependent variable in the measurement model.

In terms of data collection (and analysis), the study distinguishes between *supervisors* (representing measurements on a team-level of investigation) and corresponding *team members* (representing measurements within an individual level). Two separate questionaires were developed. The sampling of agile ISD projects and respondents had to fulfill three requirements:

First, projects had to be ISD projects. Second, each project had to follow an agile approach whereby no particular methodology was prescribed. Third, both the supervisor and their corresponding team members must have jointly operated in this project. The research participants identified in this way include, among others, the roles of developer, scrum master, or team leader. The firms that carried out these projects can be categorized medium to large international companies. A major part of the projects focused on the development, improvement, customization, or the implementation of information systems. In sum, data was obtained from an overall of 286 individuals participating in 87 agile ISD projects. These projects took place between August 2020 and March 2021 in different countries across three continents. Completed survey instruments from both supervisor and team member matching-pair for each project were required, dropped, if only one part of this dyad was available. In total, we received completed questionnaires from 66 different teams (supervisors), resulting in 148 completed matched-pairs data records (supervisor and corresponding team members). Qualtrics was used for data collection, both surveys were provided the respondents via a link included in an invitation email.

For data analysis, a covariance-based (CB) SEM approach was used instead of a composite-based (e.g., partial least squares) for mainly two reasons: first, CB SEM is said to provide more accurate and more consistent outcomes (Dijkstra and Henseler 2015; Hwang 2009) .Second, CB SEM is the method of choice when strong and established theories exist and the focus lies on confirming theoretically assumed relationships (Hair et al. 2017; Jöreskog 1967; Rigdon et al. 2017). The first step has been the preparation of data, which was mainly done with tools and corresponding libraries in the R environment. These tasks involved the conversion of data types, the management of incomplete data records or building of our matched pairs of supervisor vs. team member responses. After our data was 'analysis ready', we used MPlus as a tool of our choice because it has shown to be very efficient in handling multilevel data as it exists in form of nested data (i.e., team members in teams as represented by supervisors). We tested our model for goodness of fit and determined the most common fit indices as a result. With values such as the *Root Mean Square Error of Approximation* (RMSEA) of 0.046, *Comparative Fit Index* (CFI) of 0.958 or *Tucker-Lewis Index* (TFI) of 0.953 the overall fit of our model is good (Hu and Bentler 1999; Kenny et al. 2014). In terms of internal reliability, we used an omega coefficient instead of an alpha coefficient (Cronbach's alpha), as alpha is increasingly regarded as an inappropriate measure of internal consistency reliability (Dunn et al. 2013, p. 5). The values of both McDonald's omega and for composite reliability meet the recommended thresholds (ω and CR > 0.7) for indicating convergent validity. Loadings on the designated variables were commonly higher than the recommended value of 0.7, except for items PPR3, MSC3 and TA1 (see section 4.5 for details), which were just below this threshold (Hair et al. 2017). We decided to keep the items because they belong to validated and established measures. Based on the square root of the average variance extracted (AVE) value of a latent construct being larger than its squared correlation with any other latent construct in the model, we ensure both convergent and discriminant validity in our data. Summarized, we passed relevant cutoffs for reliability, convergent validity, and discriminant validity (Fornell and Larcker 1981; Hair et al. 2017; McDonald 1999).

Our results show that an enabling control style (CSE) is strongly associated with agile practice usage and serves as an important antecedent for agile practice usage (APU). In contrast, an authoritative style seems to have no effect on APU as we could not find any relationship between both variables. We found agile practices generally to influence each of the control modes positively, which underscores some of our findings in previous studies (e.g., Study III) and the general potential of agile practices to enact control. Concerning the influence on team autonomy, we found the informal control mode 'self-control' to have a positive effect, but surprisingly not 'clan control'. In addition, 'behavior control' (as a formal control mode) positively influences team autonomy, which is also unexpected as previous studies assume that formal controls do hinder team autonomy (e.g., Gerwin and Moffat 1997; Piccoli et al. 2004). Although we could not find any evidence for having agile practice usage directly influence team autonomy, we identified a complete mediation effect by behavior control and self-control, which act as mediators, emphasizing the interplay and interdependency of agile practices and control enactment. Table 1-15 summarizes our results and links them to this study's propositions.

No.	Proposition	Results
P1a	A predominantly enabling control style promotes agile practice usage	Supported
P1b	A predominantly authoritative control style does not promote agile practice usage	Supported
P2a	Agile practice usage promotes formal controls	Supported
P2b	Agile practice usage promotes informal controls	Supported
P2c	Agile practice usage promotes team autonomy	Rejected
P3a	Informal controls positively influence team autonomy	Partially supported
P3b	Formal controls negatively influence team autonomy.	Partially rejected

Table 1-15: Support of propositions of Study V

Our study revealed the strong linkage between the use of agile practices and control modes. Controllers (e.g., managers, team leads, supervisor) can utilize agile practices in agile ISD settings to achieve, for example, desired employee behavior while at the same time facilitating team autonomy in order to keep team members capable to response to unforeseen changes. From a theoretical point of view, this study contributes to our knowledge regarding control enactment within agile teams. This study combines novel concepts such as control styles with fundamental concepts from control theory like control modes and extends them by agile practices and team autonomy. Our results show that formal controls are also possible in agile projects, but also demonstrates that classical and authoritative control styles will reach their limits in projects where volatile requirements exist and agility is needed.

1.5 Discussion

After having presented the results of the individual studies within the scope of this dissertation, the following chapter goes into more detail about their contributions. First, the individual outputs and their significance for the follow-up studies will be discussed (section 1.5.1). Subsequently, their value and contribution to research (section 1.5.2) as well as the implications for practice are summarized (section 1.5.3). The chapter concludes with a list of open issues that were identified but not addressed in the dissertation. Based on this, the last chapter suggests starting points for follow-up studies (section 1.5.4).

1.5.1 Comparison and Dependencies

Table 1-16 compares the underlying research designs and procedures for data collection and data analysis of all studies in broad summary. It also explains about each study's intention and purpose and finally summarizes their output. As the studies' research designs, data collection and analysis approaches as well as their essential findings have been described in the previous sections, consequently, this part will briefly explain on their outputs and in relation to the other studies.

Study	*1) Agility in the Balance: Control, Autonomy, and Ambidexterity in Agile Software Development*	*2) Journey Towards Agility – A Retro- and Prospective Review*	*3) Do As You Want Or Do As You Are Told? Control vs. Autonomy in Agile Software Development Teams*	*4) "Loosening the Reins": Balancing Control and Autonomy in Information Systems Development*	*5) "Directing Self and Others": An Empirical Study of Control in Agile Information Systems Development*
Design	SLR	SLR	SLR / Single Case Study	Multiple case study research	Matched-pairs questionnaire-based field study
Data collection	Research articles (peer-review), more than one search (term)	Research articles (peer-review, N=1376)	Research articles (peer-review) / semi-structured interviews (N=8)	Semi-structured interviews (N=37)	Survey-based (N=148)
Data analysis	Concept-centric-matrix based, iterative approach of applying, analyzing, synthesizing and evaluation literature	CAA; different coding strategies	Different coding strategies	Different coding strategies; deductive theory testing	Hypothesis testing
Purpose	- Motivation / gap - Theoretical background - preliminary research model and - Research design	- Emphasis on motivation / gap - Extension of theoretical foundation	- Explore how control can be enacted through agile practices	- Increase understanding of how to enact control and how this influences team autonomy and team performance	- Quantitative substantiation of previous findings and claims
Link to main RQ?	Highlights the relationship between control modes, control style, autonomy and performance (both on team and project level)	No direct link, but addresses RQ indirectly by identifying existing control and agile ISD focused literature from different disciplines	Reveals the linkage between agile practices and different control modes *("how do modes and practices affect each other")* according control theory (incl. EOC)	Addresses the question of *"how control and agile practices affect autonomy and performance in agile ISD teams"*	Connects control modes, control styles and agile practices to team autonomy *("how do they influence team autonomy?")*

Contribution	Fosters knowledge of the interaction of control, team autonomy, team performance and ISD ambidexterity	Research foci and agenda for agile ISD (including the topic of control in agile ISD) in general, and emphasis on the motivation to conduct further research on this topic in particular	Extends current knowledge of how to enact control through agile practices	Enhances our knowledge a) of the effect of control enacting practices on team performance and b) of the influence of an enabling control style on control congruence, team autonomy and team performance	First, by showing how control is enacted through agile practice usage, second, by demonstrating how control styles influence agile practices usage, and third, by showing the effect of formal and informal modes on team autonomy
Major output for follow-up study	Preliminary research model	Improved preliminary research model	Control enacting practices	Relationships within the control enactment process including agile practices and the effect on autonomy and performance	Quantitative evaluation of findings related to the control enactment process

Table 1-16: Comparison of all studies

The aim of the first study was to introduce the underlying topic of this dissertation, and in particular to address the problem in more detail. One of the most important results of this study is the proposal of a preliminary research model, which is suitable to answer the research question (see section 1.4.1 for details). Study II builds upon the results of Study I by a) revealing more gaps in literature related to control in general and on the topic of control in agile ISD in particular and b) by identifying additional relevant research studies which provide further theoretical ground for the preliminary research model. Consequently, the main output of Study II is an improved preliminary research model. Study III focuses on a small part of the overall question, namely whether and how agile practices can enact different types of control. The result is a list of control enacting practices, based on insights from literature and evaluated by interview data from agile development teams. Study IV advances in investigating the topic by including additional control related concepts such as control styles and control congruence. The resulting picture of the interdependencies between the control enactment process, agile practices as well as autonomy and performance in agile teams, tested with qualitative data, finally serves Study V as the basis for a quantitative substantiation of previous findings and claims. In doing so, the latest study brings control styles and team autonomy into closer focus. The result is a quantitatively tested model that incorporates a fragment of the research model proposed by Study I and provides new and valid insights in this area (see also section 1.5.4. for details what has not been empirical tested as proposed by Study I's preliminary research model).

1.5.2 Contribution to Research

Within *Part I* of this dissertation, we identified research topics on agile ISD covered by relevant journals and prestigious conferences. Summarized, the findings provide an overview of topics, which attracted the attention of the research community, and which dealt with agile ISD methodologies within the last thirty years (Study II). The study demonstrated by relying on the concept of topic modeling that computer-aided topic clustering can help to outline the current state of agile ISD research and is generally suitable for similar purposes. With the help of computer-aided analysis, we showed how to process large amounts of data and to uncover topics within these texts. By further qualitative analysis, we were able to gain deeper insights into the history of agile ISD research and to uncover the topics in our body of knowledge regarding agile ISD research. One of the identified topic groups was titled "Risk, Control and Success Factors", which turns away from the more technology-driven issues and puts its focus on social or team related aspects and requirements in the agile ISD domain. In particular, the topic "Control in agile ISD" was identified as a topic that constantly evolved over time but gained in importance since 2016. Generally, we claim that this topic needs to be more embraced by researchers and have therefore taken it as an occasion to dive deeper in the corresponding matter. In this context, Study I served as a 'hotbed' by providing a preliminary research model and providing different propositions, which can be helpful to further investigate the topic of control in agile ISD.

Part II aimed at a literature-based identification and empirical evaluation of suitable control practices for agile ISD. Although the importance of control on the quality of ISD project outcomes is generally known and wide-spread in literature (Goh et al. 2013; Harris et al. 2009a; Mahadevan et al. 2015; Maruping et al. 2009a; Persson et al. 2011), there had been no focused literature review so far that sheds light upon the question how far control can be applied through agile practices, while providing team autonomy at the same time. The study closes this gap by providing detailed results derived from literature and qualitative data. Building upon our work we can extend our understanding on how agile ISD teams can be governed, especially in regard to the relationship between control and autonomy. Our list of agile practices and their impact on specific control modes revealed several interesting findings related to the topic of control usage in such projects. The study identified a set of 23 agile practices which can be linked with the enactment of formal control, in contrast, only 20 agile practices were found to foster informal control types. Whereas 12 practices are dedicated to formal control types, only three practices could be identified to address informal control modes only. The results show a surprising heavyweight on practices that enact more rigid (formal) controls instead of informal controls as one could assume. This let the authors conclude that the enactment of formal control, and thus, structure and boundaries are part of the agile ISD process, acting as counterbalance to team autonomy.

The contribution to research of *Part III* is threefold: first, the results revealed controls enabled by agile practices generally positively influence team performance. Nevertheless, the positive influence of informal controls on team performance is in line with the results reported in the literature as recent studies emphasize their performance-enhancing effect in the context of specific ISD projects (Chua et al. 2012; Tiwana and Keil 2009). In particular, the use of self-control provides developers with discretion regarding how tasks will be accomplished (Henderson and Lee 1992; Kirsch et al. 2002). For example, self-controlling team members align their resources and choose methods for goal achievement without the involvement of a project leader (Henderson and Lee 1992; Maruping et al. 2009a). Similarly, the informal mechanism of clan control establishes an environment where the controllee has the freedom to make use of her own skills and knowledge to accomplish specific tasks, leading to better team performance (Chua et al. 2012; Gopal and Gosain 2010). Formal controls, on the other hand, provide some degree of guidance and structure, which supports the execution of tasks and leads to better team performance (Remus et al. 2016). However, the effects of control enacting practices on team autonomy remain ambigious, which can be related to the state of transition of agile methodologies to a large-scale context in three cases, which led to only moderate levels of informal controls and autonomy across the cases (see section 1.5.4 for more details).

Second, the concept of 'control styles' plays an important role within the control enactment process. In detail, three important implications have emerged:

a) *the presence of an enabling control style reduces the likelihood of an authoritative control style*. This suggests that managers do not shift between both extrema of a control style, instead, the enabling and authorative control styles appear to be largely mutually exclusive and indicative of a broader cultural context. This is in line with the findings of Wiener et al. (2016) and Gregory and Keil (2014) that both styles are seldom found within a single controller but rather are commonly represented by two different managers with contrasting styles.

b) *an enabling control style promotes a shared understanding (communicational congruence) and increases perceived appropriateness (perceptual congruence) of the controls enacted* (Murungi et al. 2019; Narayanaswamy et al. 2013). The repair characteristic, on the one hand, may contribute to a generally better understanding by usually integrating regular feedback mechanisms (Gregory et al. 2013a). The transparency feature, on the other hand, may establish a 'big picture' perspective (Wiener et al. 2016), which in turn might lead to both an increased shared understanding of the rationale of controls and increased perceived appropriateness of controls.

c) *an enabling control style positively influences both team autonomy and team performance*. In detail, the two characteristics of 'repair' and 'transparency' in particular have a positive effect on flexibility and thus team autonomy (Adler and Borys 1996; Wiener et al. 2016). Likewise, these characteristics facilitate additional exchange of knowledge, regular feedback, and close collaboration within a controller-controllee dyad, leading to increasing team performance (Adler and Borys 1996; Remus et al. 2016).

Third, this study's findings suggest that the concept of control congruence is important when control is exercised within agile teams. The results reveal that it is likely that a high level of control congruence positively impact team performance. The authors' conclusion resonates with the results of recent studies, which indicate that control congruence contributes significantly to the quality of controls adopted and avoids negative socio-emotional effects, such as decreased job satisfaction (Narayanaswamy et al. 2013; Tripp et al. 2016).

Part IV extends the current knowledge regarding control enactment within agile teams especially by examining control and agile practice usage simultaneously (e.g., Maruping et al. 2009a). The study empirically proved that control styles determine the degree to which agile practices are going to be used. In detail, the results confirm that a predominantly enabling control style has a positive effect on the use of agile practices, and therefore generally serves as an important antecedent for agile practice usage. This is corroborated by the fact that we found no evidence for having a predominantly authoritative control style favoring the use of agile practices. Moreover, the study demonstrated that agile practice usage and control modes can be linked to each other. We found that both formal and informal controls are commonly used in agile teams and confirm recent findings of Sun and Schmidt (2018), who claimed that agile practices and control mechanisms facilitate each other. Although the study could not prove that agile practice usage directly influences team autonomy, a complete mediation behavior control and self-control could be identified as mediators (full mediation), putting more emphasis on the interdependency between agile practices and control enactment.

Lastly, the study showed that team autonomy is positively influenced by the informal control mode 'self-control', as well as by the formal control modes behavior control. We did not expect such results, as the literature so far assumed that formal controls do not so much promote autonomy as they hinder it (e.g., Gerwin and Moffat 1997; Piccoli et al. 2004). This finding implies that in agile ISD environments strong manifestations of a control style driven by transparency and feedback may still be able to promote desired degrees of freedom by establishing autonomy when formal controls must be enacted (e.g., see Harris et al. (2009a) and their concept of 'emergent outcome control' or Mahadevan et al. (2015, p. 80) for 'hybrid control'. To sum it up, the choice of control style and the use of agile practices occupy elementary roles in the concept of control enactment, thus decisively enabling the coexistence of control and team autonomy at the same time.

1.5.3 Implications for Practice

From a practical point of view, this research project offers several implications for practitioners challenging to control agile ISD teams.

1) As our knowledge on the question of how to enact control through agile practices in practice remained scarce, Study III is, by certain means, able to cover this gap by providing a first overview of suitable practices which are able to address different types of controls. This list of practices, allows practitioners working in agile ISD projects to evaluate the existing practices for general suitability and implementation fit within their projects. Managers and other responsible employees who are in charge of the governance of agile ISD teams can draw on our list of control enacting practices and choose between a variety of options, depending on the desired control mechanism that needs to be established. Study V contributes to these findings by empirically demonstrating that agile practices are capable of enacting control. Although no distinction was made between specific practices and their targeted effect on control modes in the empirical study, Study V nevertheless demonstrated that the use of agile practices in general can effectively address both formal and informal control modes. As an implication of this, managers, team leaders, or others who are in charge of exercising control no longer need to dispense on highly formalized controls in agile teams; agile practices (including management practices such as planning meetings or retrospectives, development practices such as pair programming or standard and norms emphasizing practices such as code reviews) are able to set predefined boundaries in which desired behavior can be achieved or control can be maintained.

2) Based on the results of Study IV combined with our insights on control enacting practices (Study III and V), managers or others who are in charge of managing agile ISD teams are able to select appropriate control modes and control styles in order to simultaneously achieve desired levels of team autonomy and team performance. In detail, controllers benefit from highly formalized controls in agile teams when enacted through selected agile practices. Formal controls issued in this manner can provide a structured framework in which desired behavior can be achieved or control of corresponding outcomes can be maintained, thus, positively affecting team performance. Similarly, a positive effect emerges when

informal controls are addressed, in other words, managers can draw from the entire control mode portfolio if the goal is to influence team performance positively.

3) Study IV and V showed that the choice of an enabling control style shows an overall positive effect within an agile team. The associated characteristics of 'repair' and 'transparency' not only promote the implementation of agile practices (Study V), a heavyweight on an enabling style also creates the foundation on which team autonomy is established, congruence of controls can be achieved and lastly, team performance can be increased. Summarized, the results showed that choosing an enabling control style as well as enacting control by using agile practices allow managers to minimize risk in the team by providing structure and managing behavior and outcomes, while still giving the team the ability to be responsive to unpredictable changes.

1.5.4 Limitations and Future Research Directions

Similar to its individual components, the dissertation is subject to a number of limitations, which will be discussed in more detail below. In this context, we also suggest possible directions for further research.

As for *Part I* and especially *Study II*, one limitation lies in the approach itself for gaining our results: Although we were able to utilize an automated approach to assess large amounts of data, the final topics (and topic groups) have been derived by applying qualitative methods. In detail, the approach provides a certain degree of freedom, so it cannot be ruled out that another research team would have identified slightly modified or even different topics. Moreover, it should be noted, however, that the interpretation in our discussion is based on the results of this topic modeling and not on statements of different authors. Therefore, our concluding statements might be of speculative nature. A frequent effort in the investigation of current developments in agile ISD research is helpful in keeping researchers focused and aware of trends, topics, and corresponding gaps. As we are convinced that this study's approach is suitable to assist in this challenging task, future research might expand on our approach by adding more and even more recent outlets as well as updating the conclusions based on an up-to-date database of research articles to further extend the applicability and generalizability of the findings as presented in this study. Replication studies utilizing these techniques might improve the confidence in our results and our conclusions.

Study III examined the relationship between agile practices and control modes according to classical control theory. Although initial findings regarding corresponding links were obtained, the results are subject to further limitations. By starting our literature review with a keyword search and also by following the guidelines of Levy, Ellis and Webster, Watson (Levy and Ellis 2006; Webster and Watson 2002a) in regard to forward and backward search, we tried to incorporate all past studies. Nevertheless, within the data analysis, we partially identified incongruity of different control mode definitions. This is complicated by the fact that further concepts and additions to the classic control modes have been developed over the course of time, some of which combine, adapt or supplement classic modes with further, possibly new concepts. For instance, Harris et al. (2009a; 2009b) focus on the concept of *emergent outcome-control* as an alternate view on outcome-control, Gregory focuses on three archetypal purposes of control (Gregory et al. 2013b), and Mahadevan introduces the concept of hybrid control, which is a compromise between well-known formal controls and the emergent outcome control concept (Mahadevan et al. 2015). Moreover, some associations are blurry and leave some room for interpretation, some are even contradictory, resulting in an in an overall lack of transparency on the associations between agile practices and control modes. Based on this limitation, we recommend further research, with focus on the comparison of control modes according to control theory. As our empirical part concentrated on qualitative fieldwork with students only, we suggest that more empirical, possibly even quantitative studies should be conducted on this topic, which extend the field of participants to non-student roles such as senior developers, managers or certified scrum-masters on both, client and vendor site.

Another limitation lies in the ambigious relationship between team autonomy and control modes. Although, positive effects of control modes on team performance could be proved, the results of Study IV could not reveal, if any specific mode contributes in any way to influence team autonomy. The reasons for this may be manifold, e.g., because the study mainly observed cases with moderate levels of informal control, or because the timing of data collection - especially for the three insurance cases - occurred in the midst of the transformation to an agile organization, i.e., old hierarchical structures still prevailed in some cases. For these reasons, we call for replication of our study in different contexts, with organizations of different sizes, industries, national settings, and overall levels of agility. We also suggest to use quantitative methods as we are convinced that these may help to reveal the different effects of certain modes on team autonomy.

Moreover, our quantitative study (Study V) investigated the effect of agile practice usage on control modes by only considering three exemplary different practices. Although we are confident that these practices very well represent different management practices (e.g., planning meetings) and development practices (continuous integration), we cannot exclude other practices would have induced different results. For this reason, future studies are needed to build on our research design while including different and more agile practices, enhancing our knowledge of how these practices (and in this way enacted controls) influence outcome variables such as team autonomy or team performance.

In addition, the dissertation did not consider all elements of the expanded control enactment framework by Wiener et al. (2016) or even more novel concepts which have evolved in the recent past. For example, the consideration of a communicational as well as perceptual congruence of controls between the controller and the controllee, and thus a consequent possible 'loss of control', could provide interesting insights, especially when outcomes such as team or project performance are of interest in the study (Huisman and Iivari 2006; Narayanaswamy et al. 2013; Ouchi 1979; Wiener et al. 2019). Other examples are the concepts of control legitimacy (Cram and Wiener 2018), control purpose (Wiener et al. 2019), or control amounts (Rustagi et al. 2008; Wiener et al. 2017; Wiener et al. 2016) (see Appendix A – Glossary of concepts for more details).

Finally, our study focused on explaining the relationships and mechanisms between agile practices, control, and outcome variables at a strictly 'team' level (e.g., team autonomy or team performance). The dissertation did not consider cross-level effects on a project level, for example, the effect of team performance of different teams on project performance) which can contribute to our general understanding of the topic.

1.6 Conclusion

How do we enact control in agile settings? Is control beneficial to agile ISD teams, even when flexibility is needed? Can control and autonomy co-exist in agile ISD teams? Is it more important how controllers exercise control than what type of control should be applied? And how can agile practices assist within the whole control enactment process? This research project addressed these and other questions. Although some of the answers still remain obscured, this dissertation approaches towards an integrated view by gaining first insights into the complex interplay between control and autonomy in agile teams.

With this study, we aimed to demonstrate our understanding of the use of agile practices and their connection to formal and informal controls. We found that both formal and informal controls are commonly applied in agile teams and confirm recent findings on this topic, which indicate that agile practices and control mechanisms facilitate each other. We have shown that the choice of a control style significantly determines how agile practices are implemented and controls are enabled in an agile environment while it is still possible to achieve the desired degree of autonomy within a team. The results show that formal controls are also possible in agile projects, but also demonstrates that classical and authoritative control styles will reach their limits in projects where volatile requirements exist and agility is needed. Last but not least, our results show that the choice of control style and the use of agile practices occupy essential roles in the concept of control enactment and thus significantly enable the preconditions for control and team autonomy to coexist.

In summary, this dissertation consequently represents a small piece of the big puzzle on the complex topic of "control in agile ISD". Our findings offer considerable room for further research, particularly through some of the shortcomings identified in this study and thus, associated recommendations for future research, which will help to increase our knowledge on control in agile ISD.

2 Part I: Justification and Theoretical Underpinning

Title	**Agility in the Balance: Control, Autonomy, and Ambidexterity in Agile Software Development (Study I)**	**Journey Towards Agility–A Retro-and Prospective Review (Study II)**
Author Names	Tim Dreesen (Corresponding Author) Sean Hansen	Tim Dreesen (Corresponding Author) Phil Diegmann Björn Binzer Christoph Rosenkranz
University	University of Cologne Faculty of Management, Economics and Social Sciences, Professorship for Integrated Information Systems Rochester Institute of Technology Saunders College of Business, Department of MIS, Marketing, and Digital Business	University of Cologne Faculty of Management, Economics and Social Sciences, Professorship for Integrated Information Systems
Abstract	*Agile methodologies for information systems development (ISD) are still drawing the attention of the research community. These methodologies promise to increase an ISD team's adaptiveness in such a way that ISD teams are able to respond and react to changing user requirements. Existing studies on team autonomy in agile ISD, however, imply that these projects potentially can benefit from different elements of control. Our objective is to improve the understanding of how to enact control through agile practices, and how these practices influence team autonomy and task performance in successful agile ISD projects in terms of project performance and project quality. This is achieved by developing a preliminary research model that is based on a solid theoretical foundation. As a theoretical framework, we employ ISD ambidexterity and extend it with context-specific insights from control theory. In consequence, we suggest several propositions for future testing.*	*After more than 15 years since the Agile Manifesto and extensive research on agile software development (ASD) for nearly three decades, a comprehensive body of knowledge is available and is constantly growing. ASD is considered an effective way for managing software development projects in environments characterized by rapidly changing requirements. This study aims to shed light on the existing knowledge on ASD by applying a structured literature review and computer aided analysis consisting of distinct text mining techniques. We analysed a sample of 1,376 papers and provide results from articles among relevant information systems research as well as computer science conferences and journals. Based on our approach, we are able to (1) evaluate key articles and journals, (2) analyse the development of ASD research in the last three decades and, most importantly, (3) identify research foci of the past as well as gaps in our knowledge on ASD.*
Keywords	Ambidexterity, Agile Information Systems Development, Control Theory, Task Performance, Team Autonomy	Agile Software Development, Information Systems Development, Literature Review, Text Mining, Topic Modelling
Paper Status	Accepted, published in proceedings of the International Conference on Information Systems (ICIS) 2018	Accepted, published in proceedings of the 52nd Hawaii International Conference on System Sciences.
Preliminary/ alternate versions (already published)	none	Diegmann, Dreesen, Binzer and Rosenkranz - Journey Towards Agility: Three Decades of Research on Agile Information Systems Development (ICIS 2018)

2.1 Study I

2.1.1 Introduction

Over the past decade, agile approaches to *information systems development (ISD)* have emerged as a dominant paradigm (Baskerville et al. 2011; Highsmith et al. 2001). A core facet of agile ISD is the principle of *team autonomy* – providing individual team members and groups the freedom to self-organize and initiate action on their own. Yet extant research paints a conflicted picture of autonomy's impact on teams. On the one hand, team autonomy has been found to reduce productivity and performance in the context of project teams (Langfred 2004); on the other hand, it has been identified as an important factor in enabling teams to respond to change (Vidgen and Wang 2009). The practical impact of autonomy on performance in agile ISD teams is further complicated by the question of *control* – understood broadly to mean "any process in which a person or group of persons or organization of persons determines … what another person or group or organization will do" (Tannenbaum 1962, p. 239) As this definition suggests, the exercise of control necessarily implies certain limits on the ideal of team autonomy. Yet, substantial research suggests that control leads to better *task performance* within a team (Hackman 1987; Zellmer-Bruhn and Gibson 2006), even in agile ISD contexts (Harris et al. 2009a; Kirsch et al. 2002; Persson et al. 2011). Furthermore, the use of agile methods and the exercise of control have been shown to have a positive effect on project quality (Maruping et al. 2009a). Of course, control itself encompasses a range of different mechanisms and forms; most notably, there is a critical distinction between formal and informal modes of control (Kirsch 1997). Disentangling the nuances and effects of varied control modes in the enactment of agile practices may have substantial import for our understanding of effectiveness in agile ISD settings. Despite a range of calls for further research on the impact of control and team autonomy in ISD efforts (Chua et al. 2012; Wiener et al. 2016), the existing evidence remains ambiguous, especially in the field of evolving agile ISD methodologies (Cram and Brohman 2013). Few studies have investigated informal controls and their effects on ISD outcomes, such as software product quality, or their enactment within agile practices (Cram et al. 2016a). Accordingly, we follow the call of Wiener et al. (2016) for further research on the inconclusive and partly contradictory results regarding control in ISD (Choudhury and Sabherwal 2003; Tiwana and Keil 2009), the interplay between formal and informal control mechanisms (Persson et al. 2011; Tiwana 2010), as well as their relationship to team autonomy (Gerwin and Moffat 1997) and task performance (Kirsch et al. 2002). Consequently, the central research questions guiding our study are:

(1) To what degree do modes and styles of control, as enacted through agile practices, influence team autonomy and team task performance?

(2) What are the effects of team autonomy and team task performance on ISD ambidexterity in agile development environments?

To foster a deeper understanding of the interaction of team autonomy, control, and task performance in agile ISD, we propose a model for identifying and investigating effects of agile practices on task performance and autonomy in agile ISD teams. This model draws substantially upon theories of ISD *ambidexterity* (Tiwana 2010), highlighting our focus on the degree to which autonomy and task performance are held in balance in successful agile ISD efforts. The remainder of this research-in-progress paper is structured as follows. We give a brief overview of related work on the topic of agile ISD research, focusing the aspect of control investigated on a team level view. Next, we derive the proposed model and state corresponding propositions based upon previous literature. Finally, we provide further information on the proposed

2.1.2 Related Work and Theoretical Background

2.1.2.1 Agile Information Systems Development

Agile ISD is an umbrella term for a variety of distinct methodologies, such as Scrum or eXtreme Programming (XP) (e.g., Schwaber 1995), which collectively emphasize an iterative development model, close collaboration between stakeholders, and a lightweight approach to documentation (Cohen et al. 2004). In contrast to structured development's embedded resistance to change (at least beyond the point of requirements documentation), the lightweight practices of agile ISD are envisioned to embrace and learn from change in pursuit of increased customer value (Conboy 2009). Thus, in a business environment where available technologies, market structures, and customer preferences change rapidly, agile ISD approaches are intended to enable ISD teams to react to emergent needs in a timely manner (Lee and Xia 2010a). This capability reflects what Gibson and Birkinshaw (2004) refer to as *adaptiveness* – the degree to which an individual or group is able to reconfigure its activities to correct misalignments with organizational goals, for example, due to the refinement of project requirements, technological, or environmental changes (Tiwana 2010). In order to respond effectively to change, agile ISD methodologies emphasize a variety of specific *agile practices* (e.g., Lee and Xia 2010a). Examples of agile practices from XP are *pair programming* (all production code is written with two programmers at one machine) and *collective code ownership* (anyone can change any code anywhere in the system at any time). Similarly, popular Scrum practices include *daily scrum meetings* (a daily stand-up meeting in which all project participants briefly review the status of their work) and *user stories* (a method to define broad requirements while enabling creativity) (e.g., Harris et al. 2009b).

2.1.2.2 Control and Autonomy in Agile Information Systems Development

In agile ISD and its associated technical processes, key *social practices and principles* have to be considered as well (e.g., Hummel et al. 2015). These socially-oriented practices and principles include, for example, an emphasis on collaboration between business and technical stakeholders (Lee and Xia 2010a), a preference for informal face-to-face communication and interactions (Hummel et al. 2015), and the promotion of self-organization and reflection within teams (Vidgen and Wang 2009). Importantly, these social facets have significant implications for the concept of managerial control (Remus et al. 2016). Research on control in ISD has produced a wealth of valuable insights. Kirsch's control theory (1996, 1997, 2004) offers a particularly valuable perspective. With respect to ISD teams, control theory distinguishes formal control types (e.g., input, behavior, and outcome controls) from informal control types (e.g., self-control and clan control) (Kirsch 1996). Table 2-1 summarizes key control modes, which are often exercised in concert rather than in isolation, representing a so-called control portfolio (Kirsch 1997).

Control Mode		Characteristics
Formal	Input Control	Measurable actions prior to implementation of an activity e.g., recruitment, training programs or manpower allotments
	Behavior control	Emphasizes behaviors, processes and procedures that must be followed, and offering rewards contingent on the adherence to the prescriptions.
	Outcome control	Involves outlining project goals and offering rewards contingent on their accomplishment. Emphasizes outputs regardless of the process used.
Informal	Clan control	Socializes team members into sets of valued norms. Emphasizes reinforcement of acceptable behaviors through shared rituals and experiences.
	Self-control	Provides autonomy to individuals to determine what actions are required and how to execute them. Emphasizes self-regulation of goals and self-monitoring of progress.

Table 2-1: Summary of control modes following (Kirsch 1996) (Study I)

It is well known that traditional ISD approaches rely heavily on formal control modes (Kirsch 1997; Kirsch et al. 2002; Tiwana and Keil 2009). By contrast, informal control potentially provides developers with discretion in how tasks will be accomplished (Henderson and Lee 1992; Maruping et al. 2009a). Informal controls such as clan and self-control promise to enact autonomy, which is seen as an important antecedent for responding to changing user requirements (Gerwin and Moffat 1997; Maruping et al. 2009a). The exercise of clan control allows a development team to identify important project goals and to determine how to attain them on their own (Maruping et al. 2009a). The exercise of self-control – i.e., "the extent to which an individual exercises freedom or autonomy to determine both what actions are required and how to execute these activities" (Henderson and Lee 1992, p. 760) – similarly enables flexibility in pursuit of objectives, focusing on the role of the individual rather than that of the group. While most of the existing studies focus on controlling portfolio configuration ("what" control modes are used), few studies investigate "how" controls can be put into practice (Gregory et al. 2013a; Tiwana and Keil 2009). Consequently, we follow Wiener et al. (2016) in suggesting *control style* as a relevant concept for control-enactment. Control style can be defined as "the manner in which the interaction between the controller and the controllee is conducted" (Wiener et al. 2016). Related literature suggests two contradictory control styles – *authoritative* and *enabling* (Adler and Borys 1996; Gregory et al. 2013a). An *authoritative* control style is employed if strict behavioral compliance is desired, granting the controllee less discretion in how control is enacted (Wiener et al. 2016). An *enabling* control style, on the other hand, is used to achieve compliant behavior while granting flexibility in decision making to deal with uncertainties in daily work procedures (Adler and Borys 1996; Remus et al. 2016). The principle of control and the various ways of framing its effects have significant implications for the agile ideal of team autonomy. Following Lee and Xia (2010a, p. 90), we define *team autonomy* "as the degree of discretion and independence granted to the team in scheduling the work, determining the procedures and methods to be used, selecting and deploying resources, hiring and firing team members, assigning tasks to team members, and carrying out assigned tasks" (Lee and Xia 2010a, p. 90). As this definition underscores, team autonomy is intrinsically intertwined with the broader objectives of flexibility and adaptiveness in agile ISD (Larman 2003b), as well as the related concepts of self-organization (Highsmith et al. 2001), self-management (Langfred 2004), and team empowerment (Larman 2003b).

Next to team autonomy, the enactment of control is closely linked to the establishment of *task performance,* defined as the degree to which a team achieves its goals and how well its outputs match the team's mission (Hackman 1987). Although a variety of empirical studies analyze the effect of control and task performance on agile ISD project outcomes (Maruping et al. 2009a; Persson et al. 2011), results remain ambiguous (Cram and Brohman 2013). For example, in terms of product quality, Maruping et al. (2009a) suggest that agile ISD project teams can benefit from the implementation of certain control modes (especially outcome control) to create an environment in which agile practices can engender autonomy while clear performance goals and structures are maintained. On the other hand, Harris et al. (2009a) argue that formal outcome control is insufficient in agile environments and propose the novel concept of *emergent outcome control* as a way to achieve a better product-market match. Regarding informal controls, little research has investigated informal controls, such as clan and self-control, and their effects on outcomes (e.g., software product quality) (Cram et al. 2016a). This reinforces some of the findings of Wiener et al. (2016), asserting that earlier studies on control in IS produced inconclusive and partly contradictory results. For example, there is no consensus if informal control has a positive (Henderson and Lee 1992; Wiener et al. 2015) or negative impact (Tiwana 2010; Tiwana and Keil 2009) on project performance and project adaptiveness. This has led to calls for examining the extent to which individual agile practices affect project outcomes, and the interplay between control modes and agile practices (Maruping et al. 2009a).

2.1.3 Theory Development and Research Model

Figure 2-1 summarizes our research model on the interrelationship between control-enacting agile practices and control styles, team autonomy, and task performance, and how they influence adaptiveness and outcomes in terms of project performance.

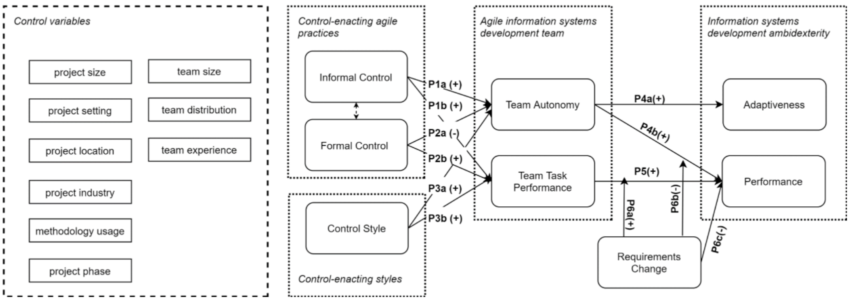


Figure 2-1: Proposed research model (Study I)

The research model has three components. First, as the relevant dependent variable for agile ISD, we focus on the construct of *ISD ambidexterity* (Tiwana 2010) because it combines both adaptation to changing demands and adherence to organizational objectives. Specifically, ambidexterity is "the capacity to simultaneously exhibit alignment and adaptiveness" (Gibson and Birkinshaw 2004, p. 211), where alignment is the degree to which the work of a group fulfills the project objectivities (e.g., client requirements or quality expectations) (Tiwana 2010). Alignment is a critical consideration in determining whether a project is completed on time and within budget – thus indicating a process performance dimension (Keil et al. 2013) – as well as if it fulfills requirements in terms of quality – indicating product performance (Henderson and Lee 1992). In our analysis, we opt for the term *project performance* in place of alignment to underscore the importance of both process and product performance dimensions (Tiwana and Keil 2009; Wiener et al. 2016). As the second constituent part of ambidexterity, *adaptiveness* describes “the capacity to reconfigure activities in the business unit quickly to meet changing demands in the task environment” (Gibson and Birkinshaw 2004, p. 209). By focusing on ambidexterity in our research model, we embrace a “both/and” rather than “either/or” perspective - the two dimensions are seen as interrelated but non-substitutable (Tiwana 2010).

Second, from a control-enactment perspective, we include both *control modes* and *control style* as independent variables in our research model. We suggest that different types of control can be exercised through different agile practices, that is, the method-in-action and generative rules, which are adapted to fit an agile ISD team’s specific context (Highsmith et al. 2001). While some extant research has mapped agile practices to either formal or informal control modes (e.g., Harris et al. 2009b; Persson et al. 2011), conclusive designations are challenging. Based on the results of an extensive structured literature review (*anonymous for review*), we identified a total set of 29 agile practices, including an analysis of their correspondence to specific control modes. Although most agile practices defy a straightforward classification by control mode, a subset of these practices offer clear indications of formal and informal control modes in their enactment. Table 2-2 provides an excerpt of control modes embodied in agile practices.

Next, we see *team autonomy* as an important mediating variable in our model, which is impacted by agile practices enacting formal or informal control, as it describes the extent to which a team underlies restrictions and interdependencies through control (Piccoli et al. 2004) or is granted discretion and independence (e.g., in scheduling the work or carrying out tasks) (Lee and Xia 2010a).

We also use *task performance* as a mediator, which is defined as the degree to which a team achieves its goals and how well its outputs match the team’s mission (Hackman 1987; Zellmer-Bruhn and Gibson 2006). Third, we consider *requirements change* as an important moderating construct, as ISD projects often face high level of uncertainty and volatility (Lee and Xia 2010a; Tiwana and Keil 2009) and therefore impact project performance (e.g., by cost overruns or poor product quality) (Maruping et al. 2009a). We now discuss propositions that link the concepts in our research model. In line with the argument of Wiener et al. (2016), we assert the need for greater consideration of the question of *control-enactment* – that is, *how* software project leaders are able to put distinct configurations of control portfolios into practice.

Practice	Description	Control Modes		References (Examples)
Backlog prioritization / estimation	Prioritize a collection of user stories. Assign IT estimation points.	Formal	BC, OC, EOC	Harris et al. (2009b); Mahadevan et al. (2015)
Burndown charts	A publicly displayed chart showing remaining work in the sprint backlog that is updated every day.	Formal	OC	Gregory et al. (2013b); Mahadevan et al. (2015)
		Informal	CC	Gregory et al. (2013b)
Code Reviews / Refactoring	After a piece of software code is developed or modified by a single software engineer, it is submitted to a code review system through which peer developers are invited to review and propose changes.	Formal	BC. OC, EOC	Harris et al. (2009a); Persson et al. (2011)
		Informal	SC, CC	Gregory et al. (2013b); Persson et al. (2011)
Collective Code Ownership	Anyone can change any code anywhere in the system at any time.	Informal	SC, CC	Maruping et al. (2009b); Persson et al. (2011)
Daily Scrum / stand-up	Daily meeting where members ex-plain briefly what they accomplished, what will be completed and indicate impediments preventing them from completing tasks.	Formal	BC, OC	Cram and Brohman (2013); Misra et al. (2009)
		Informal	SC, CC	Mahadevan et al. (2015)
Pair Programming	XP stipulates pair programming as a core practice, where two programmers, sharing the same computer, work collaboratively on all aspects of software development	Formal	BC, EOC	Harris et al. (2009b)
		Informal	SC, CC	Maruping et al. (2009a)
User stories	Documented set of requirements to be achieved by development	Formal	OC, EOC	Harris et al. (2009b); Mahadevan et al. (2015)

LEGEND: BC = Behavioral Control, CC = Clan Control, EOC = Emergent Outcome Control, OC = Outcome Control, SC = Self-Control

Table 2-2: Control modes embodied in agile practices (Excerpt, Study I)

Regarding the effects of formal and informal controls, several studies find that informal control usage provides high levels of autonomy in managing assigned work tasks – for example, by enabling the team to determine objectives, tasks, and monitoring activities to achieve project goals (Kirsch et al. 2002; Remus et al. 2016). Moreover, informal controls have been found useful in promoting effectiveness, and recent studies emphasize their performance-enhancing effect in the context of specific ISD projects (Chua et al. 2012; Tiwana and Keil 2009). In particular, the use of self-control provides developers with discretion with regard to how tasks are accomplished (Henderson and Lee 1992; Kirsch et al. 2002). For example, self-controlling team members are able to align their resources and choose methods for goal achievement without asking the project leader (Henderson and Lee 1992; Maruping et al. 2009a). Clan control can be promoted by establishing a collaborative culture within the team, allowing the controller to create an environment where the controllee has freedom to make use of her own skills and knowledge in order to accomplish certain tasks, leading to better task performance (Chua et al. 2012; Gopal and Gosain 2010). Consequently, we propose:

> **P1:** *Greater use of informal controls positively impacts (a) team autonomy and (b) task performance.*

In contrast, other studies find that formal controls "limit the team's autonomy" (Piccoli et al. 2004, p. 366) by overemphasizing work formalization (Barker 1993; Remus et al. 2016). For example, routine team progress reports and strict adherence to schedules and task assignments may hinder a team's effectiveness, as teams frequently turn to managers instead of solving problems on their own (Piccoli et al. 2004; Robey et al. 2000). Emphasizing functional specialization puts a manager in the position of controlling most decision making, leading to decreasing team autonomy (Gerwin and Moffat 1997). On the other hand, formal controls provide some degree of guidance and structure, which supports the execution of tasks (Remus et al. 2016). Such controls may provide clear directions and predefined workflows on how to perform certain tasks (Kirsch et al. 2002) or recommend proven techniques or practices (e.g., user stories), which in turn positively affect task performance (Remus et al. 2016). Thus, our second proposition is:

> **P2:** *Greater use of formal control negatively impacts (a) team autonomy, while it positively affects (b) task performance.*

As authoritative and enabling control styles can be seen "as end points on a continuum" (Wiener et al. 2016, p. 755) we follow Remus et al. (2016) and focus on an enabling style in our model. An enabling control style has two main characteristics, "repair" and "transparency" (Adler and Borys 1996). Together, both features establish an environment for the controllee, that is characterized by feedback, involvement in the control configuration, and some degree of freedom to "deviate from controller prescriptions [...] in order to respond to real-work contingencies" (Remus et al. 2016, p. 7). Additional exchange of knowledge, regular feedback, and close collaboration between controller and controllee leads to increasing task performance (Adler and Borys 1996; Remus et al. 2016). Conversely, a lack of information exchange and feedback mechanisms associated with an authoritative style leads to decreased task performance (Choudhury and Sabherwal 2003). We also suggest that an enabling control style increases team autonomy. An enabling style is likely to promote informal controls (such as clan control) which in turn positively affects team autonomy (see P1) (Wiener et al. (2016). This may be due to the repair and transparency characteristics, which allow for better knowledge exchange and continuous feedback loops (Adler and Borys 1996; Wiener et al. 2016). Hence, we propose:

> **P3:** *Greater degrees of an enabling control style positively affect (a) team autonomy and (b) task performance.*

However, agile ISD teams need to respond to changing user requirements (Conboy 2009). Much of the extant literature highlights the idea that "doing agile" is closely linked with the principle of autonomy. Indeed, a*gile ISD views team autonomy as an important antecedent of adaptiveness (Lee and Xia 2010a). In addition to its effect* on adaptiveness, team autonomy "brings decision making authority to the hands of the people who face and handle problems every day" (Lee and Xia 2010a, p. 90), suggesting an increased effectiveness in problem solving (Larman 2003b). *Thus, our fourth proposition is:*

> **P4:** *Greater degrees of team autonomy positively affect (a) adaptiveness and (b) project performance.*

Furthermore, task performance directly impacts project performance. Task performance of individuals "ultimately affects performance on the project level" (Remus et al. 2016, p. 2). For example, if formal control is enacted through agile practices, formal rules and procedures are established, assisting team members in efficient and effective task execution, which in turn positively impacts performance on a project level (Gopal and Gosain 2010; Remus et al. 2016). Therefore, we propose:

P5: *Task performance positively impacts project performance.*

Agile ISD teams tend to consider many alternative problem-solving approaches, creating the risk that they could possibly "lose sight of project objectives" (Maruping et al. 2009a, p. 383). To counteract this dilemma, high degrees of task performance can be achieved by establishing formal controls, emphasizing task outcomes and task behavior and mitigating the team's risk of failing to achieve predetermined project goals (Kirsch 1997; Maruping et al. 2009a). Without "guidance," teams may miss important objectives (Maruping et al. 2009a, pp. 383-384). However, there is also a downside regarding high degrees of autonomy when requirements continuously change over time (Maruping et al. 2009a). In environments with high requirements volatility, clan control is difficult to establish and self-control is minimally effective (Keil et al. 2013). Under these conditions where developers need to effectively coordinate their tasks, different approaches (e.g., coding processes) may be highly incongruent, adversely affecting performance and software quality (Maruping et al. 2009a). Finally, we assert that user requirements changes tend to inhibit agile ISD performance directly, because they may create a shifting standard for performance and the potential for conflicting outcomes. Indeed, requirements changes have consistently been identified as one of the top risks in ISD (e.g., Boehm 1991). Dynamic markets and changing user needs may also result in technical and managerial issues that can adversely affect performance measures (e.g., on-time completion or completion within budget; (Wiener et al. 2015). Consequently, our last proposition is:

> **P6:** *When requirement changes are high rather than low, (a) the positive relationship of task performance on (agile ISD) performance is strengthened, (b) the positive relationship of team autonomy on (agile ISD) performance is mitigated and (c) project performance is negatively affected.*

Our research design includes several control variables relating to a) agile ISD teams, i.e., team size, team distribution, team experience, and b) the project domain and specific project characteristics, i.e. size, setting (in terms of collocated, distributed or outsourced development), location (domestic vs. international), industry, methodology usage, project phase.

2.1.4 Proposed Research Design

We have chosen a research design that corresponds to our objective of explaining and predicting the phenomenon of interest (Recker 2012) – that is, exploring and explaining the effects of applying agile practices on team autonomy and task performance in agile ISD teams and predicting patterns concerning process and product performance as well as the effect on adaptiveness. To increase generalizability by considering divergent views (Creswell 2009) and to provide stronger evidence of our conclusions (Esteves and Pastor 2004), we will investigate our conceptualized model and the associated propositions using a *mixed method approach*, incorporating both qualitative and quantitative research methods. This has been shown to be a fruitful approach in investigating agile ISD (e.g., Lee and Xia 2010a). We follow the guidelines of Creswell (2009) by using a sequential exploratory strategy (i.e., within a first phase we collect and analyze qualitative data to adjust our research model and propositions, followed by a second phase of quantitative data collection and analysis that builds on the results of the first phase). Specifically, we will start by conducting in-depth case studies (Yin 2003b). We will test the feasibility of our preliminary research model in a manageable environment to elaborate our propositions and constructs of the model, before testing the model within a wider field (Hunter 2004). The qualitative data will be coded, starting with pre-defined concepts derived from our model and coupled with open coding to identify novel concepts. Following this qualitative analysis phase, we will conduct a quantitative analysis using structural equation modeling (SEM). The primary data collection instrument will be a questionnaire-based survey administered to practicing software development professionals. To the extent possible, objective data sources (e.g., logs from project management and tracking tools) will also be used, particularly for the measurement of project outcomes. We will design this study as a panel, and with data collection at multiple points in time (start, middle, and end of projects). Multivariate analysis will be used for analyzing the survey's results. To ensure generalizability beyond the data collection setting, we will apply the pre-tested research model on a larger scale by extending it to a larger sample of national and international agile ISD projects.

2.1.5 Contribution

The proposed research model describes how control can be enacted through agile practices while considering control styles, how these controls impact the autonomy and task performance within an agile ISD team, and the resulting effects on an agile ISD team's performance and adaptiveness. Based on the intended results, we are able to provide answers to our research questions and to enhance our knowledge on control in agile ISD projects from both a theoretical as well as practical point of view. We contribute to theory by fostering our knowledge of the interaction of control, team autonomy, and task performance in agile ISD as well as the overall influence on ISD ambidexterity. We provide practitioners guidance on how to enact control through agile practices, taking into account different control styles and how to achieve the degree to which autonomy and task performance are held in balance in successful agile ISD efforts.

2.2 Study II

2.2.1 Introduction

Interest in *agile software development (ASD)* methodologies has increased in recent years in both research and industry (Conboy 2009; Fitzgerald et al. 2006a; Lee and Xia 2010b). Based upon the principles of the Agile Manifesto (Beck et al. 2001b), different implementations, such as Scrum or eXtreme Programming (XP), have emerged and motivated a variety of research.

ASD has been applied to a wide range of projects: from small teams, situated in co-located offices (e.g., Cao et al. 2009b) to large scale, distributed, or outsourced projects (e.g., Sarker and Sarker 2009). In this context, ASD methodologies and practices have been implemented successfully but also unsuccessfully (Lee and Xia 2010b). Research also has investigated the customization and configuration of agile approaches, the so-called method tailoring (e.g., Fitzgerald et al. 2006a; Karlsson and Ågerfalk 2009; Wang et al. 2012). Due to the wide variety of topics covered by ASD research, ranging from rather technical aspects (e.g., Balijepally et al. 2009) to sociological or psychological factors (e.g., Maruping et al. 2015a), and from an individual level to an organizational level (e.g., Zheng et al. 2011), a clear categorization of existing streams of research is difficult to recognize. Additional difficulties arise because the concept of ASD, its exact definition, and its applicability are debated (Conboy 2009).

Motivated by this, our study's objective is twofold. First, we ask which topics of ASD research have been explored in the past and are currently investigated. Second, we want to identify topics that are not covered in current research and therefore still remain non-existent in extant literature. Consequently, the central research questions guiding our study are:

(1) What research topics have been addressed within the last three decades by ASD research and

(2) how do these topics differ in terms of available publications and their distribution over time?

To answer our research questions, we conducted a structured and comparative literature review as described by the guidelines of Levy and Ellis (2006) and Webster and Watson (2002b), followed by computer-aided topic modeling (Aggarwal and Zhai 2012; Debortoli et al. 2016b) on the extant body of knowledge of ASD.

The remainder of this paper is structured as follows. We give an overview about related work, targeting research on the field of ASD. Next, we describe our research design being used for data collection and analysis. Following, we present and discuss our findings. Finally, we provide an outlook for and point out future research directions.

2.2.2 Related Work and Background

2.2.2.1 Agile Software Development

In practice, approaches for developing software range from sequential approaches (Royce 1970a) to more cyclic, iterative approaches (Boehm 1988), that is ASD. During the last two decades, ASD methodologies such as eXtreme programming, rapid application development, or rapid prototyping complemented the iterative approach. Additionally, new management concepts associated with ASD, such as Scrum and Lean Software Management, have been proposed.

The four basic principles of the Agile Manifesto (Beck et al. 2001b) can be found in most ASD methodologies. According to the Agile Manifesto, ASD should value individuals and interactions over processes and tools, working software over comprehensive documentation, customer collaboration over contract negotiation, and responding to change over following a plan (Beck et al. 2001b). Each of these principles have been subject to research in some sort: for instance, in regard to individuals and interactions, research has investigated the effects of communication in ASD teams (Hummel et al. 2013), in regard to working software, extant literature investigated the influence of pair programming on software quality (Balijepally et al. 2009), in regard to customer collaboration, the funding process has been studied (Cao et al. 2013), and the ability to respond to change has been subject of studies as well (Fitzgerald et al. 2006a).

Moreover, next to the methodologies themselves, extant research so far has studied individual or organizational phenomena, such as the use and effects of specific agile practices (Balijepally et al. 2009; Maruping et al. 2009b), and effects regarding whole projects or organizations, such as the introduction of ASD methodologies to teams (e.g. Cao et al. 2009b). Furthermore, the use of hybrid methodologies or the tailoring of agile methodologies to a team's specific needs is covered by extant research (Karlsson and Ågerfalk 2009; Lee and Xia 2010b; Wang et al. 2012). Literature investigating the success and failure of ASD mostly focusses on specific methodologies, such as Scrum or XP (Fruhling and de Vreede 2006), or specific practices, for instance pair programming (Cao et al. 2013). Extant research focusing on success and failure of ASD in general exists, but is rare (Lee and Xia 2010b).

2.2.3 Existing Literature Reviews

By conducting a structured literature review, we assessed the current state of research regarding summarizing and aggregating literature reviews. We searched for articles containing "literature" and "review" as well as synonyms for ASD (i.e., scrum, xp or kanban) in the title, abstract, or keywords. The search was limited to a timeframe up to and including August 2016 and the outlets of the "Senior Scholars' Basket of Journals" edited by the Association for Information Systems and top conferences. We finished the search process with a resulting set of 15 relevant papers, of which none did a historic-holistic approach, meaning each of the structured reviews does not necessarily consider all agile methodologies, an explicit focus on software development or a broader and up-to-date timeframe. Instead, they focused on a specific field of interest, such as software engineering for ubiquitous systems (e.g., Guinea et al. 2016), individual acceptance, tailoring, or use of agile methods and practices (e.g. Campanelli and Parreiras 2015b; Inayat et al. 2015), general practices and challenges in agile requirements engineering (e.g., Inayat et al. 2015), or geographically distributed, large scale ASD and agility (Alzoubi et al. 2016; Dikert et al. 2016).

We can therefore conclude that few summarizing or aggregating literature reviews on the field of ASD research exists and that those articles are oftentimes specialized and limited in scope. For instance, ASD has been included in a summary for information systems offshoring (Strasser and Westner 2015). Other aggregating or summarizing literature focuses on the concept of agility itself (Conboy 2009), but only few provide an overview about existing studies (e.g., Dingsøyr et al. 2012; Dybå and Dingsøyr 2008). In sum, a clear categorization of existing streams of research is difficult to recognize.

2.2.4 Research Method

2.2.4.1 First Phase: Structured Literature Review

The approach of a structured literature review is chosen because of its applicability to gain an overview of the field and extant research and help to identify research gaps (Vom Brocke et al. 2015). The low number of review articles that are being published in the field further motivates the approach (Rowe 2014; Webster and Watson 2002b). Reviews are often a means to expose emerging issues to potential theoretical foundations, and because ASD itself is still a continuously emerging topic (Dingsøyr et al. 2012), this review aims at analyzing the extant research literature to summarize what has already been researched and what is left to be examined. To provide a comprehensive overview on current ASD topics and those topics that still have to be investigated, the existing literature is thoroughly examined, using a structured approach by following the guidelines of Levy and Ellis (2006) and Webster and Watson (2002b).

Initially, our data collection process started by performing an extensive keyword search within leading journals. We set a focus on primarily high quality, peer-reviewed literature, published in journals of the "Senior Scholars' Basket of Journals" and the AIS Toplist (including leading journals not only from IS but also Management and Computer Science). Additionally, we included articles of prominent conferences (e.g., HICSS, ICIS, ICSE). A complete set of all outlets is available from the authors on request. We defined a single search string for our keyword search to identify relevant articles in different databases: TIKEAB[6]:(software OR "information system") AND TIKEAB:(development OR engineering OR maintenance OR method* OR practic*) AND (TIKEAB:(agil* OR SCRUM OR XP OR "Extreme Programming" OR Kanban) NOT TIKE:(manufac*)) with TIKEAB searching in title ("TI"), keywords ("KE"), and abstract ("AB") and TIKE searching in title ("TI") and keywords ("KE").

As we aimed for an as broad and holistic overview as possible, we only applied minimal include and exclude criteria. We excluded those publications, which were either difficult to automatically analyze via text mining (e.g., non-English language or with no full text available) or which were not research-focused (e.g., an opinion or commentary). We decided to use a restriction for the publishing year of the articles, thus, articles that were published between January 1st, 1985 and December 31th, 2017 were included. January 1st, 1985 was chosen because the first article we found was from 1985 and all data was collected in August 2016, which is why we chose December 31st, 2017 as cap. Within the resulting set of papers, we further identified relevant articles for our project purpose ("in scope", i.e., investigating ASD) and dropped the others ("not in scope", i.e., not investigating ASD).

6 Concatenation describing the focus of the keyword search, for example "TIKEAB" indicates searches within title ("TI"), keywords ("KE") and abstract ("AB")

In total, after removing duplicates, our final set of articles consists of 678 articles matching our search indicators for ASD in journals and 698 articles in conference proceedings, totaling up to 1,376 articles. Further information concerning the distribution of results can be seen in Table 2-3.

Outlets	**Hits**
Conferences	
International Conference on Software Engineering	139
XP / Agile	132
Hawaii International Conference on System Sciences	98
International Conference on Global Software Engineering	52
Americas Conference on Information Systems	49
Journals	
IEEE Software	187
Journal of Systems and Software	99
Information and Software Technology	73
Computer	28
Communications of the ACM	23

Table 2-3: Distribution of results across top-five outlets of each type (Study II)

2.2.4.2 Second Phase: Computer Aided Analysis

Following to the data collection, we analyzed all articles with the help of *Scikit-learn* (Pedregosa et al. 2011), a computer-aided analysis and text mining tool. From within the Scikit-learn suite of machine learning tools, we specifically used *topic modelling* (Aggarwal and Zhai 2012; Debortoli et al. 2016b), which uncovers topics shared by different articles. We use this technique to easily discover topics shared across research and therefore to help in answering our research questions. Research found text mining and especially topic modelling to be helpful in discovering hidden topics by classifying, summarizing, and clustering of text (Maowen et al. 2012; Srivastava and Sahami 2009) and topic trends over time (Alghamdi and Alfalqi 2015). This semi-automated approach is especially helpful in analyzing large amounts of text (Maowen et al. 2012; Srivastava and Sahami 2009).

In order to analyze the extracted data, we first had to convert the articles into a compatible format by extracting text where available or by applying optical character recognition where no text was directly accessible. Furthermore, we annotated the extracted text with additional information, such as author, year, title, and outlet to enable further reaching analysis.

Following the data preparation, we utilized *Latent Dirichlet Allocation* (LDA; Blei et al. 2003) as implemented in Scikit-learn as a specific topic modeling approach. Within LDA, each document is seen as a mixture of different topics and each topic has certain probabilities of generating keywords. Keywords are allowed to occur in more than one topic. LDA has been used in various research studies (e.g. Chen et al. 2016) and has been suggested as a suitable and helpful tool for research (Debortoli et al. 2016b).

A too high number of topics to extract might lead to an excessive number of meaningless topics and a too low number might constrain the results unnecessarily; thus, the number of topics to be extracted is the most crucial parameter of the analysis (Debortoli et al. 2016b). Therefore, we used four different algorithms (Arun et al. 2010; Cao et al. 2009a; Griffiths and Steyvers 2004; Mimno et al. 2011) aimed at evaluating the quality of topic models to decide which number of topics leads to the optimal topic model. After testing and evaluating different numbers of topics, we settled on 34 topics, as it provided differentiated topics. Of these topics, 8 topics were discarded, as they covered less than 0.5% of all tokens (i.e., text), resulting in a final set of 26 topics. Furthermore, we decided against the use of lemmatization or stemming to avoid misleading keywords (e.g., "agil" instead of "agility" or "agile"). We opted to use n-grams (i.e., creation of consecutive words such as "agile software development"; in this setting, we decided to use 3-grams) to reduce the number of words with identical meanings but different lexical representations. To further refine the results, we used a list of stop words, which consisted of frequently found words, which added no meaning, such as "et al." or "journal". A complete list of all stop words used within our analysis can be provided by the authors on request.

2.2.4.3 Third Phase: Coding

Following Saldaña (2016) we applied different coding strategies as an exploratory problem-solving technique and to link our keywords to patterns, resulting in meaningful topic descriptions. At the core is the task of conceptualization, that is, "the process of grouping similar items according to some defined properties and giving the items a name that stands for that common link" (Strauss and Corbin 1998, p. 121). As coding can be seen as cyclical (Saldaña 2016), our coding process therefore can be distinguished between a first cycle coding and second cycle coding phase.

During the first cycle coding we started with "descriptive coding". Descriptive coding primarily leads to a categorized summary of the data's contents and builds the groundwork for second cycle coding and further analyses (Wolcott 1994, p. 55). All authors independently and individually made use of descriptive coding and compared all resulting topics against each other by comparing the included keywords per topic. Based on the keywords, a summarizing phrase was suggested. In case of matching topic phrases, no further action was needed. In case of differing topic phrases, the reasoning for each phrase was compared and alternatives were discussed. Subsequently, descriptive coding for differing phrases was repeated and consensus was reached.

We then applied "pattern coding" as a second cycle coding method. Pattern coding is appropriate for the development of major themes from data (Miles and Huberman 1994a; Saldaña 2016). These codes are helpful for aggregating and grouping themes into a smaller number of sets, themes, or constructs (Miles and Huberman 1994a, p. 69). Similar to first cycle coding, we then tried to group our descriptive codes into meaningful pattern codes – again first individually, followed by a discussion where needed. Again, pattern coding was conducted twice until consensus was reached.

We completed the coding process within a final step, in which we did some post-coding activities such as fine-tuning of the wording and alphabetical order of the results. The outcome of the coding process is a final set of 26 topics and eight topic groups.

2.2.5 Results

Figure 2-2 displays the number of articles published per year, as well as the number of articles published each year in the Senior Scholars' Basket. In Figure 2-3 a further distinction between publications focusing on either computer science or information systems research is made. To get more into detail, Table 2-3 shows the number of papers found for each outlet with at least five publications. Conferences and journals are displayed separately, but each are ranked by the number of publications in descending order. Table 2-4 lists our identified topics, the topic groups, the keywords contained in each topic, and the rank in terms of frequency of the individual topics. As can be seen from Table 2-4, we identified several topic groups because of the different foci of the topics themselves: while some topics comprise more general information such as concepts, principles, or methodologies related to ASD (see "Agile Methodology & Practice Usage" or "General"), others focus on an organizational perspective and link agile principles such as flexibility or agility to different contexts (see "IT Capability & Agility"); still others focus on managerial implications (see "Business & Environmental Factors") or put emphasis on certain aspects such as social or team related aspects and requirements engineering (see "Teams & Team Management" or "Stakeholders & Requirements Engineering") or risks and success factors (see "Risk, Control & Success Factors in Agile"). Furthermore, we identified a topic group containing research regarding technological aspects (see "Technologies & Applications in Agile").

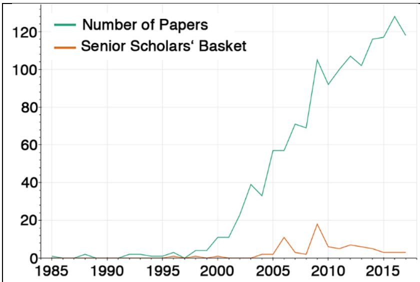


Figure 2-2: Number of papers published per year (Study II)

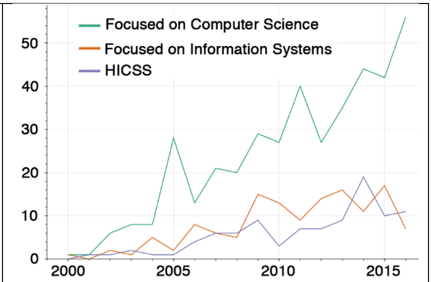


Figure 2-3: Number of papers in computer science or information systems focused outlets and HICSS per year (Study II)

Group	Topic		Rank
Agile Methodology & Practice Usage	1	Lean	2
	2	Large-Scale ASD	5
	3	Agile Architecture & Design	7
	4	Scrum	13
	5	Tests & Test-Driven-Development	14
	6	Pair Programming	17
	7	Extreme Programming	20
	8	Documentation	23

Business & Environmental Factors	9	Open Source	18
	10	Business, Transformation, Rules	25
	11	Technical Debt	26
General	12	Theory in ASD	1
IT-Capability & Agility	13	Supply Chain, Agility, Capabilities	9
Risk, Control, & Success Factors in Agile	14	Risk Management, Outsourcing, Project Management	8
	15	Effort Estimation, Success Metrics	10
	16	Control	19
Stakeholders & Requirements Engineering	17	User Participation & Design	4
	18	Requirements Engineering & Stakeholder Management	6
	19	Roles in ASD	11
	20	Requirements, Interdependencies, Prioritization	22
Teams & Team Management	21	Teams & Kanban	3
	22	Teaching and Learning Agile	12
	23	Communication in Distributed ASD	16
	24	Decision Making in ASD	21
Technologies & Applications in Agile	25	Cloud, Services, Security	15
	26	Big Data	24

Table 2-4: Identified topics, groups and ranks (Study II)

2.2.5.1 Research Foci Over the Last Decades

Although at first glance our topics presented in Table 2-4 seem to randomly comprise a lot of different and wide spread themes, further investigation and analysis of our results reveal distinct and meaningful patterns. The resulting topics, consisting of specific keywords, are overlapping but each one of them has its "raison d'être", as they represent themes that have been addressed in ASD research within the last decades.

As can be seen from Table 2-4, the first topic group, "Agile Methodology & Practice Usage", summarizes the "basics" of ASD. The keywords are centered around ASD methods, concepts, practices, management, and tasks. The second topic "Business & Environmental Factors" deals mainly with distinct business contexts such as open source while topic three contains a more general, theoretical perspective on ASD. The fourth topic group, "IT Capability & Agility", relates to a broader view on agile, namely organizational agility and IT capabilities. Similarly, "Risk, Control & Success Factors in Agile" entails risk assessment, quality and success factors, as well as control related content. "Stakeholders & Requirements Engineering" entails topics centered around different stakeholders, the process of requirements engineering, and generally speaking the involvement of users in the software development process. The topic group "Teams, & Team Management" is focused more on project management activities involving the team on a more abstract level. The last topic group "Technologies & Applications in Agile" relates to some technical and application-oriented facets, namely cloud technologies, security, and big data in ASD.

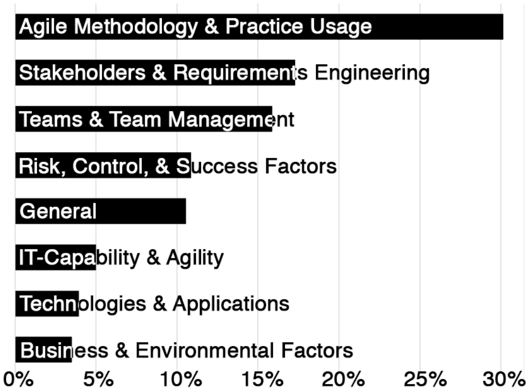


Figure 2-4: Topic group distribution (Study II)

Looking at the rankings of the topics and overall distribution (see Figure 2-4), one sees that ASD methodologies have been covered most (32.35%). While "Teams & Team Management" appears to be covered well (15.87%), actual team interaction (i.e., "Teaching and Learning Agile", "Communication in Distributed ASD", and "Decision Making in ASD") has been covered less so (3.72%, 2.79%, and 0.95% respectively) and most of the distribution stems from "Teams & Kanban" (8.41%). The ranking of topics of each group serves as a proxy in their distribution (ranks 3, 12, 16, and 21 for this example).

2.2.5.2 Key Outlets and Articles

Based on the number of publications per outlet displayed in Table 2-3, we clearly see that the computer science-oriented conferences (e.g., ICSE, XP/Agile) dominate the information systems-oriented conferences with nearly three times the number of publications (ICSE: 139 vs AMCIS: 49). The most prestigious information systems conference, the International Conference on Information Systems (ICIS), shows up second to last with 22 publications. This might hint at the orientation of extant ASD research being more technical and less managerial, social, or interdisciplinary (see also Figure 2-3).

Regarding the journal-based publications, the field is dominated by IEEE Software with 187 publications, followed by the Journal of Systems and Software (99) and Information and Software Technology (73). The most published-in journal of the Senior Scholars' Basket is the European Journal of Information Systems with 21 publications, ranked sixth, tied with IEEE Transactions on Software Engineering.

Looking at the most published-in outlets over time, one can identify different trends. While some outlets have been publishing ASD research early on (e.g., IEEE Software, Computer, ICSE, or HICSS), some started out later (e.g., ECIS, Journal of Systems and Software, or Information and Software Technology). While IEEE Software has been early on a very important outlet for ASD research, it has lost steadily since 2010 – but an upward trend started in 2016.

Looking at more recent publication statistics, especially XP/Agile, HICSS, and the Journal of Systems and Software appear to be the most up-and-coming outlets for ASD research. The trend for PACIS and ICIS appears to be declining.

Furthermore, topic modeling allows for identifying those papers, which cover each topic the most. It is important to note that "most covering" does not mean that these articles are the most influential or most important ones for this topic but rather are covering the topic most precisely in terms of the LDA model. We see that some topics are driven by the same authors repeatedly (e.g., "Effort Estimation, Success Metric" by Abrahamsson or "Pair Programming" by Balijepally), or that some authors are involved in different topics (e.g., Conboy in "Lean" and "Communication in Distributed ASD").

2.2.6 Discussion

2.2.6.1 Trends

By further investigation of our timeline regarding the distribution of published articles (see Figure 2-2, Figure 2-3), we recognize several interesting findings. First, ASD seems to strongly draw the interest of the research community starting around the year 2000, spiking at around 2003. Since then, there is a significant increasing slope of the graph, indicating that more articles have been published in the following years. Popular works published within this year are for example Williams and Cockburn's article "Agile Software Development: It's about Feedback and Change" (Williams and Cockburn 2003) and of course the "Agile Manifesto" (Beck et al. 2001b). All publications have in common that they deal with the topic of ASD from a methodology perspective, putting emphasis on concepts, principles, or detailed information concerning a specific approach. Some other articles published in the year 2003 deal with the topic of "virtual teams" (Edwards and Sridhar 2003). This is not surprising, since the concept of virtual teams is seen as an important antecedent for "doing agile" in organizations (Bowen and Maurer 2002; Domino et al. 2002).

Second, we identified a peak in our timeline between 2008 and 2009. One explanation for this may be the call for papers for special issue themes, such as "flexible and distributed ISD" in Information Systems Research (ISR) journal (Fitzgerald et al. 2006c) or previous works, which inspired further research, such as Larman's "Agile and Iterative Development: A Manager's Guide" (Larman 2003a) or Poppendieck and Poppendieck's "Lean Software Development: An Agile Toolkit" (Poppendieck and Poppendieck 2003). The ISR special issue was intended to build on the success of a previous special section of Communications of the ACM (Ågerfalk and Fitzgerald 2006a) and mini-track at HICSS in 2006 (Ågerfalk and Fitzgerald 2006b). Ågerfalk and Fitzgerald argued that "it became clear from these efforts that as a very active emerging area of research, there was an imminent need for a forum that allowed for the development and dissemination of full-research papers of the highest quality" (Ågerfalk et al. 2009,

p. 318). Similarly, a special issue of the European Journal of Information Systems was published in 2009 (Abrahamsson et al. 2009). It aimed at improving the understanding of various phenomena in ASD.

Consolidating this description of the trend in publications of ASD research, we suggest that ASD, while being a highly important topic to practice (Version One 2018), and despite a high and still growing number of publications, still lacks coverage in the top journals of information systems research as both curves drift further apart over time (see Figure 2-2).

2.2.6.2 Implications

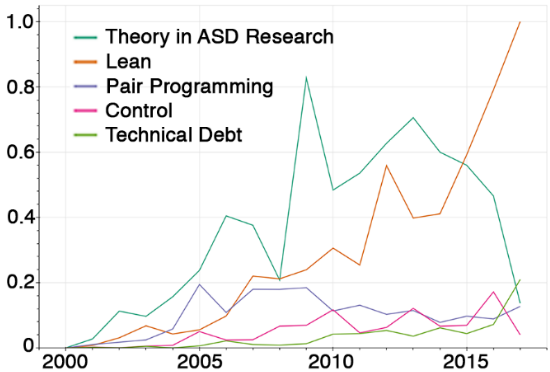


Figure 2-5: (Normalized) Distribution of selected topics over time (excerpt) (Study II)

Combining the outlined descriptions and looking at the evolution of topics present in research (see Figure 2-5), we found the majority (approximately 75%) of all mentioned topics gaining popularity over time. On closer examination of the data, however, we found topic-specific differences with regard to the respective trend development. First of all, the topic "Theory in ASD Research" is overall losing traction since its highest peak in 2009 and a smaller spike in 2013. Compared and in contrast to this trend, the topic "Lean" is overall showing a positive trend in topic distribution over time. While, from a trend development perspective, both topics, "Theory in ASD Research" and "Lean", are very much alike, we see a notable turning point in 2015, where for the first time "Lean" became distributed wider than theory-related topics. Moreover, the trend development of "Lean" represents by far the steepest slope compared to all other topics since 2015, indicating that this topic is not yet saturated but currently is the most discussed topic, with only temporary drops in its ascend to the top. This is interesting, since Dingsøyr et al. (2012) made this assumption in 2012: "A growing interest is evident at agile conferences on identifying ways to combine principles of lean development with software development" (Dingsøyr et al. 2012, p. 1218 p. 1218). Besides these examples for either strongly increasing and decreasing topic trend developments, we also found topics which have developed almost constantly over time: "Pair Programming" and "Control in ASD" are good examples. A striking feature of the latter topic is the peak in 2016, which can be explained by the extensive literature review and the call for further research on the topic of control by Wiener et al. (Wiener et al. 2016). Both topics are generally less often discussed but show a comparatively non-volatile behavior over time.

Regarding the overall coverage of different topics, the distribution over different outlets (see Table 2-3) the rankings of the topics (see Table 2-4), topic group distributions (see Figure 2-4), and the distributions over time (see Figure 2-5), we derive conclusions over gaps in the extant literature. The top three topics are about "Theory", "Lean" and "Teams & Kanban", indicating an emphasis on distinct methodology usage and team management in literature. Nearly all topic groups have at least one topic in the top 10, indicating some degree of coverage, with the only exceptions being the topic groups "Business & Environmental Factors" and "Technologies and Applications". As can be seen in Figure 2-5, "Technical Debt", as part of "Business & Environmental Factors", has only started to increase in coverage over the last two years, indicating an upward trend. In general, topics focusing on the above mentioned aspects of ASD are found first at rank 15, indicating a gap in current research. With regard to the latter the lack is not surprising since ASD is a socio-technical process rather than a purely technical one (Lyytinen et al. 1998) and consequently, the major problems of ASD projects are less technological as more sociological in nature (DeMarco and Lister 1987, p. 4). While other topics might touch on social aspects as well (e.g., "Communication in Distributed ASD" or "Teaching & Learning Agile"), these aspects are far less pronounced and of a more ancillary nature in these topics. Contrary to the fact that these topics are themselves of ancillary nature due to their low ranking and distribution and that these aspects appear to be peripheral matter to extant research, research acknowledges the importance of a not only technical but also social focus of ASD (Conboy et al. 2011; Maruping et al. 2015a).

In line with Dingsøyr et al. (2012) we observe a trend of increasing quantity and quality of ASD research and that some subfields (i.e., topics) in ASD research are more mature or saturated than others. Both, the findings from Dingsøyr et al. (2012) and the "top 10 burning questions" (Freudenberg and Sharp 2010) are reflected in our results: "Lean", "Effort Estimation, Success Metrics", "Agile Architecture & Design", or "Large-Scale ASD" are important topics, while "Pair Programming and "Extreme Programming" are becoming comparatively less important. Furthermore, Freudenberg and Sharp (2010) point out that sociological studies are important but currently mostly of peripheral appearance, which is clearly still the case and echoed by our results – a chance for ASD researchers.

To encourage ASD research to close these gaps, we propose the following research agenda. First, technologies and applications (see Topic Group "Technologies & Applications") as well as tool support (see Topic Group "IT Capability & Agility" and related topics) should be investigated further. The low rankings of the specific topics (see Table 2-4) and the low overall distribution (see Figure 2-4) paint a clear picture of an underrepresented research area. Studies on the effects of the use of tools such as versioning systems or coding tools would be valuable, as issues relating to, for instance, communication (e.g., Hummel et al. 2013) could be improved with improved understanding of the role of tools in ASD. Second, the "social" aspect of "socio-technical systems" needs to be embraced more by researchers. Similar to the first point of our research agenda, our data shows clearly a lack of research on this aspect of ASD, as no single topic group focuses on social aspect. For example, studies on the effects of agile SD on control (see Topic Group "Risk, Control, & Success Factors in Agile") or team-related issues (see Topic Group "Teams & Team Management") such as team composition or team diversity, could complement existing similar information systems research streams and answer calls for research (e.g., Lee and Xia 2010b; Wiener et al. 2016). Third, we encourage ASD researchers to increase the amount of self-reflecting and reviewing literature. By reflecting upon the current stage of ASD research, gaps become more apparent and by replicating extant research, trust in existing findings can be improved. We believe that the ASD research community specifically and the IS community in general would benefit greatly from extensive research on these three main points of our proposed research agenda.

It should be noted, however, that our discussion is based on the results of this topic modeling and not on statements of different authors. Therefore, our statements are of speculative nature and only backed by exemplary reasoning.

2.2.7 Conclusion and Outlook

Within this paper, we identified research topics on ASD covered by relevant journals and prestigious conferences. Our findings provide an overview of topics, which attracted the attention of the research community dealing with ASD methodologies over the last three decades.

Based on the topic modeling conducted on this data set, we demonstrated that computer-aided topic clustering can help to outline the current state of ASD research. With the help of computer-aided analysis, we were able to process large amounts of data and uncover topics within these texts. Further processing of this data and the results, as well as qualitative analysis, helped us gain deeper insights into the history of ASD research and uncover the topics in our body of knowledge regarding ASD research. Further, we waged an outlook into the future of ASD research by identifying less covered topics and looking for gaps in the topics covered by extant research. This might help other scholars in identifying new avenues and further extends the scientific community's knowledge about ASD.

We are confident that our study and results provide an appropriate degree of generalizability, completeness, and replicability. We described our procedure and sources to ensure replicability, while generalizability and completeness go together for this study. Due to the comprehensive literature basis provided by our structured literature review and the help of a computer-aided analysis, we are able to process extant research at large and discover topics. This research design facilitates generalizability and completeness.

Future research might expand on this research by adding more outlets or updating the conclusions based on more recent publications to further extend the applicability and generalizability of our findings. We also call for replication of our study to improve the confidence in our results and our conclusions. A continued effort in keeping track of the developments in ASD research might help in keeping researchers focused and aware of trends, topics, and gaps.

3 Part II: Control Modes and Agile Practices

Title	**Do As You Want Or Do As You Are Told? Control vs. Autonomy in Agile Software Development**
Author Names	Tim Dreesen (Corresponding Author) Thomas Schmid
University	University of Cologne Faculty of Management, Economics and Social Sciences, Professorship for Integrated Information Systems
Abstract	*Agile Software Development (ASD) projects still draw the attention of the research community. Agile methodologies promise to increase an ASD team's agility in such a way, that these teams are able to respond and react to changing user requirements. Existing studies on flexibility and autonomy in ASD projects, however, imply that these projects potentially can benefit from different elements of control. Our objective is to improve the understanding of how to enact control through agile practices, and how these practices affect either formal or informal control in ASD teams. Based on an extensive literature review, our study (1) provides an overview of adequate control-enacting agile practices and (2) compares the results with our empirical findings, derived from qualitative data.*
Keywords	Agile Software Development, Control Theory, Practices, Task Performance, Team Autonomy
Paper Status	Accepted, published in proceedings of the 51st Hawaii International Conference on System Sciences (2018) (IT and Project Management Minitrack)
Preliminary versions (already published)	none

3.1 Introduction

In today's software development practice the capability of rapid response to changing user requirements "has become increasingly critical for software development performance" (Lee and Xia 2010a). To address this crucial need, different agile software development (ASD) approaches have emerged during the 1990s and 2000s (Lee and Xia 2010a), for example, Scrum (Schwaber and Beedle 2001) or eXtreme Programming (XP) (Beck and Andres 2004), and are widely used in corporate settings. Whereas each ASD methodology may differ in terms of key principles and practices, they all have in common that they emphasize the importance of project teams that are empowered to make decisions, while the project manager's role has become rather team-supportive than team-directive (McAvoy and Butler 2009). Thus, although originally designed for small teams, ASD approaches are nowadays used even by large organizations, which tend to use scaling methodologies such as Scrum of Scrums or Scaled Agile Framework (VersionOne 2015).

Despite the popularity of ASD methodologies, projects using ASD still fail. For example, 94% of all organizations surveyed by a recent industry survey use ASD methodologies, but only half of them assess majority of their agile projects successful (VersionOne 2015). The most often mentioned reasons of project failure are a lack of experience regarding the use of agile methodologies (41%), a company philosophy or culture contrary to core agile values (46%), and missing management support (38%) (VersionOne 2016). Other studies come to similar results and conclude that agile projects have more or less the same fail rate today as in 2001 (Nguyen 2016). So despite proponents' view of ASD approaches, they are clearly not a "silver bullet" in and of themselves, overcoming long-known problems in software development (Fraser and Mancl 2008; Fraser et al. 2007). Because of the high popularity and still increasing use of ASD methodologies in practice and the notable number of unsuccessful projects, there is a need of identifying issues and proposing solutions to contribute to the enhancement of the success rate of ASD projects.

An often-mentioned trade-off that is seldom investigated may hold the key to answering this problem. It is known that a key factor of effectively managing any kind of software development project is controlling the development process and its results (Kirsch 1997; Orlikowski 1991; Wiener et al. 2016). ASD, however, is characterized by autonomously working teams, where this autonomy on the one hand enables them to respond to change but on the other hand, can be detrimental to the development process, for instance, when teams lose themselves in arguing how to tackle a problem rather than solving it (Maruping et al. 2009a). Acknowledging this apparent conflict between control and autonomy, and taking into account that ASD projects can and do fail (Nguyen 2016; VersionOne 2015), the question is in how far control and structure are needed in ASD projects, and how they can be applied with respect to the core principles of agile methodologies, especially to empower teams in decision making (McAvoy and Butler 2009). Only limited guidance exists on how ASD teams should be governed, especially in regards to the relationship between control and autonomy (Maruping et al. 2009a).

The goal of this research is to analyze common agile practices in ASD projects and, especially, to identify their impact on control and autonomy within ASD project teams. We agree with Wiener et al. (2016)

that more research is needed on control enactment in IS. In this review, we focus on a specific project context, that is ASD. Hence the following research question guides our study:

"How can control be enacted in ASD projects through specific agile practices and how do they affect different types of control (i.e., formal and informal control) within an ASD team?".

To answer our research question, we conducted a structured and comparative literature review on control enacting practices within ASD projects, based on the guidelines of Webster and Watson (2002a) and Levy and Ellis (2006). We analyzed the existing literature on ASD projects and identified a total set of 29 control enacting practices related to particular control modes. To empirically validate the literature review's results, we investigated agile practice usage and their impact on control and autonomy within 8 different ASD student teams by conducting semi-structured interviews. Based on the review's results and on our qualitative findings, we conducted an in-depth comparison of these practices concerning their suitability to enact control. The result of our study is a comprehensive summary of control enacting practices suitable for ASD projects.

The remainder of the paper is structured as follows. The next section provides information on the theoretical background, specifically on control theory, which serves us a theoretical lens, and the relation of control to ASD approaches. Section three introduces our research design with a description of the literature review as well as our data collection and analysis approach. Section four explains the results of our research with a focus on comparing control enacting practices and their impact on formal and informal control according to control theory. Section five summarizes our findings, explains the limitations of the study, and provides guidance for future research. Finally, section six provides a brief conclusion.

3.2 Theoretical Background

ASD is not only a technical process, but a social process as well (Balijepally 2005; Hummel et al. 2015; Pelrine 2011; Robinson and Sharp 2005; Sarker and Sahay 2003). This is why ASD project leaders must choose appropriate methods for managing both (Maruping et al. 2009a). An important aspect of the management process is the function of control (Kirsch 1997). Following Tannenbaum (1962), we define control in a broader way "to refer to any process in which a person or group of persons or organization of persons determines, that is, intentionally affects what another person or group or organization will do (Tannenbaum 1962, p. 239). We primarily rely on control theory by Kirsch (1996; 1997; 2004), which serves us as a theoretical lens. Although particular ASD methodologies are not specifically addressed within control theory (Cram and Brohman 2010), Kirsch points out that organizations in dynamic, changing environments may change control approaches through an ASD project's lifecycle, resulting in the implementation of appropriate control types (Kirsch 1996; 1997). Theory distinguishes formal control types such as input, behavior and outcome control from informal control types such as self-control and clan control as relevant to ASD teams (Kirsch 1996). Table 3-1 summarizes key control modes, which often are exercised in concert rather than in isolation, representing a so-called control portfolio (Kirsch 1997).

Control Mode		Characteristics
Formal	*Input Control*	Measurable actions prior to implementation of an activity e.g. recruitment, training programs or manpower allotments
	Behavior control	Emphasizes behaviors, processes and procedures that must be followed, and offering rewards contingent on the adherence to the prescriptions.
	Outcome control	Involves outlining project goals, and offering rewards contingent on their accomplishment. Emphasizes outputs regardless of the process used.
Informal	*Clan control*	Socializes team members into sets of valued norms. Emphasizes reinforcement of acceptable behaviors through shared rituals and experiences.
	Self-control	Provides autonomy to individuals to determine what actions are required and how to execute them. Emphasizes self-regulation of goals and self-monitoring of progress.

Table 3-1: Summary of control modes (Jaworski 1988; Kirsch 1996) (Study III)

The exercise of formal control provides guidance and structure, which assist the development team in task execution (Kirsch et al. 2002; Remus et al. 2016). It is well known that traditional software development (SD) approaches rely heavily on formal control mechanisms (Kirsch 1996; Kirsch 1997; Kirsch et al. 2002; Napier et al. 2009; Tiwana and Keil 2009). By contrast, informal control potentially provides developers with discretion regarding how tasks are accomplished (Henderson and Lee 1992; Kirsch et al. 2002; Maruping et al. 2009a). Generally, ASD methodologies rely more on informal controls rather than traditional formal controls (Cram et al. 2016a). Informal controls such as clan and self-control promise to enact autonomy, which is seen as an important antecedent for development teams being able to respond to changing user requirements (Gerwin and Moffat 1997; Maruping et al. 2009a). The exercise of clan control allows the team to identify important project goals and to determine how to attain them on their own (Maruping et al. 2009a). The establishment of self-control is similar, but focusses on the individual instead on a group of individuals. Self-control defines "the extent to which an individual exercises freedom or autonomy to determine both what actions are required and how to execute these activities" (Henderson and Lee 1992, p. 760).

ASD approaches view team autonomy as one of the essentials that affects agility (Larman 2003b; Lee and Xia 2010a). Prior literature provides various definitions of team autonomy and other closely related terms, for example, self-organization (Chow and Cao 2008; Highsmith et al. 2001), self-management (Sharp and Robinson 2004), or team empowerment (Larman 2003b; Maruping and Magni 2012). Following Lee and Xia (2010a, p. 90), we define team autonomy as the degree of discretion and independence granted to the team in scheduling the work, determining the procedures and methods to be used, selecting and deploying resources, hiring and firing team members, assigning tasks to team members, and carrying out assigned tasks (Lee and Xia 2010a, p. 90). Thus, ASD approaches are often seen as a counter-balance to the more rigid, formal, and structured SD approaches (Berente et al. 2015).

Next to team autonomy, the enactment of control is closely linked to the establishment of task performance, which is defined as the degree to which a team achieves its goals and how well its outputs match the team's mission (Hackman 1987; Zellmer-Bruhn and Gibson 2006). Although we find several empirical studies that analyze the direct effect of control and team task performance on ASD project outcomes such as product quality (Goh et al. 2013; Harris et al. 2009a; Mahadevan et al. 2015; Maruping et al. 2009a; Persson et al. 2011), results still remain ambiguous (Cram and Brohman 2013). For example, in terms of product quality Maruping et al. (2009a) suggest that ASD project teams can benefit from the implementation of control modes, especially formal outcome control, to create an environment in which agile practices can provide autonomy whilst at the same time clear performance goals and structures exist. On the other hand, Harris et al. (2009a) propose emergent outcome control as a new concept to achieve a better product-market match, as they argue formal outcome control to be insufficient in agile environments. Emergent outcome control therefore uses scope boundaries and ongoing feedback to "define the allowable space for exploration" and "check on decision as they are made throughout the development process" (Harris et al. 2009b, p. 405). Regarding informal controls, Cram et al. (2016a) argue that little research has investigated informal controls such as clan and self-control and their effects on outcomes (e.g., software product quality). This matches some of the findings of Wiener et al. (2016) who showed that earlier studies on control in IS produced inconclusive and partly contradictory results. For example, there is no consensus if informal control has a positive (Henderson and Lee 1992; Wiener et al. 2015) or negative impact (Tiwana 2010; Tiwana and Keil 2009) on project outcomes.

3.3 Research Design and Method

In line with our overarching research question *"How can control be enacted in ASD projects through specific agile practices and how do they affect different types of control (i.e., formal and informal control) within an ASD team?"*), our project followed a three-step data analysis approach (see Figure 3-1).

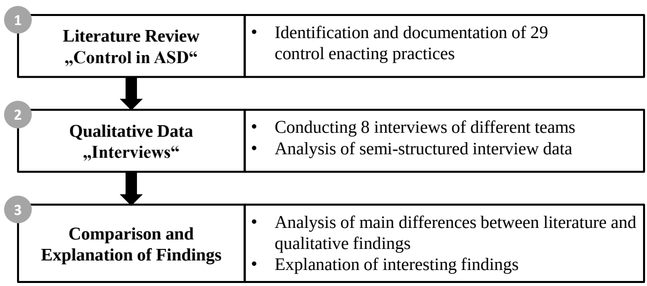


Figure 3-1: Analysis approach (Study III)

First, we conducted a concept-driven and systematic literature review based on the approaches of Levy and Ellis as well as Webster and Watson (Levy and Ellis 2006; Webster and Watson 2002a). The review started with a keyword search on control within ASD projects in general and control enacting agile practices in ASD projects in particular, followed by a backward and forward search. To achieve high quality results, only journals and conference articles listed in the top MIS journals and conferences ranking provided by the VHB (http://vhbonline.org/vhb4you/jourqual/vhb-jourqual-3/teilrating-wi/) were used. We defined a single search string for the keyword search (see Figure 3-2) to identify relevant articles in databases like EBSCOhost, INFORMS or ProQuest. There was no restriction for the publishing year of the articles. All search results were examined regarding title, abstract, and keywords. Within the resulting set of papers, we further identified relevant articles for our project purpose ("in scope") and dropped the others ("not in scope"). We subsequently proceeded with a reference, author and keyword backward search. Finally, a reference and author forward search identified our final set of articles for the data analysis phase. In total, our final set of articles consists of 28 articles on control in an agile environment. A brief summary of our literature search process can be found in Figure 3-2.

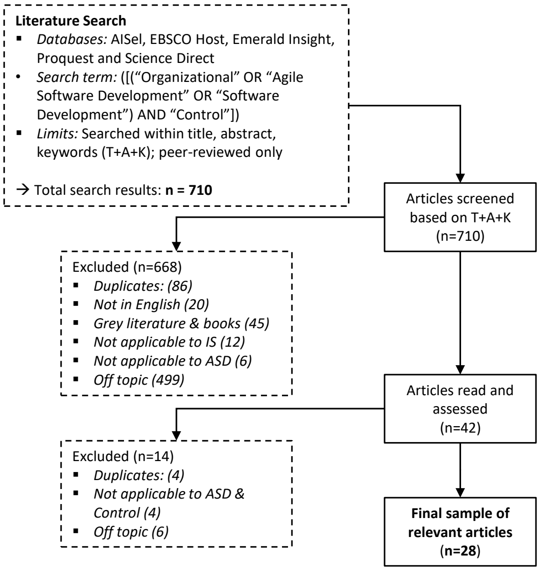


Figure 3-2: Literature search process (Study III)

Second, as part of our research design, we wanted to evaluate whether our findings of step 1 can be applied to practice by conducting semi-structured, one-to-one interviews with team members of 8 different development projects. One-to-one interviews allow gathering of rich data from people in different roles (Myers 2013). Furthermore, semi-structured interviews involve use of pre-formulated questions but allow improvisation for emerging topics during conversation. Each interview is based on an interview guide (Yin 2003b). All teams consist solely of students, participating in development projects with different industry partners. All development teams made use of the agile methodology *Scrum*. Objective data such as logs, project schedules, code repositories have been accessed and analyzed as well as field observations were conducted.

The results of the first and second step are set down in two tables, describing our findings of the literature review as well as from our collected qualitative data. We used a concept matrix that is based on several categories to structure the presentation of the results. The approach allowed us to differentiate between practices that enable different types of control (or control modes). Based on the concept matrix as well as both result tables, we were able to perform step 3 in order to identify major findings and insights.

3.4 Results

3.4.1 Control in Agile Software Development

The literature revealed 29 associations between agile practices and the defined control modes (Table 3-2). Due to space restrictions, a complete set of literature references has been neglected but is available from the authors on request. The associated control modes are based on control theory by Kirsch (e.g. (Kirsch 1996; Kirsch 1997; Kirsch 2004)) containing formal outcome- and behavior-control as well as informal self- and clan-control.

No.	Practice	Control Modes		#	References
1	Acceptance Testing	Formal	BC OC EOC	2	(Harris et al. 2009b; Ramesh et al. 2012)
2	Backlog prioritization / estimation	Formal	BC OC EOC	4	(Cram and Brohman 2013; Harris et al. 2009b; Mahadevan et al. 2015; McHugh et al. 2011a)
3	Book clubs	Formal	BC	1	(Gregory et al. 2013b)
		Informal	SC	1	(Gregory et al. 2013b)
4	Burndown Chart	Formal	OC	4	(Gregory et al. 2013b; Harris et al. 2009a; Mahadevan et al. 2015; McHugh et al. 2011a)
		Informal	CC	1	(Gregory et al. 2013b)
5	Code Review / Refactoring	Formal	BC OC EOC	3	(Harris et al. 2009a; Persson et al. 2011; Ramesh et al. 2012)
		Informal	SC CC	2	(Gregory et al. 2013b; Persson et al. 2011)
6	Coding Standards	Formal	OC	1	(Xu 2009)
		Informal	CC	1	(Maruping et al. 2009a)
7	Collective Code Ownership	Informal	SC CC	4	(Maruping et al. 2009b; Persson et al. 2011), (Cao et al. 2009b; Fitzgerald et al. 2006b)
8	Continuous Integration	Formal	BC EOC	2	(Harris et al. 2009a; Harris et al. 2009b)
		Informal	CC	1	(Harris et al. 2009b)
9	Co-location of Team Members	Formal	EOC	1	(Harris et al. 2009a)
10	Daily Stand-up	Formal	BC OC	2	(Cram and Brohman 2013; Misra et al. 2009)
		Informal	SC CC	8	(Babb et al. 2014b; Hoda et al. 2013; Mahadevan et al. 2015; McHugh 2011; McHugh et al. 2012; Misra et al. 2009; Tessem 2014; Vidgen and Wang 2009)
11	Defect Reporting	Formal	OC	2	(Cram and Brohman 2013; Gregory et al. 2013b)
		Informal	SC CC	1	(Gregory et al. 2013b)

No.	Practice	Control Modes		#	References
12	Energized Work	Formal	BC	1	(Harris et al. 2009b)
		Informal	CC	1	(Harris et al. 2009b)
13	Incremental Design	Formal	BC EOC	1	(Harris et al. 2009b)
14	Iterative Development	Formal	BC OC EOC	5	(Goh et al. 2013; Harris et al. 2009b; Karlström and Runeson 2005; Misra et al. 2009; Ramesh et al. 2012)
15	Iteration Planning	Formal	BC	2	(Mahadevan et al. 2015; Persson et al. 2011)
		Informal	SC CC	6	(Babb et al. 2014b; Mahadevan et al. 2015; McHugh 2011; McHugh et al. 2012; Tessem 2014; Vidgen and Wang 2009)
16	Iteration Retrospective	Formal	BC OC	2	(Gregory et al. 2013b; Mahadevan et al. 2015; Xu 2009)
		Informal	SC CC	5	(Babb et al. 2014a; Mahadevan et al. 2015; McHugh et al. 2012; Tessem 2014; Vidgen and Wang 2009)
17	Iteration Review	Informal	CC	1	(McHugh 2011)
18	Release Planning	Formal	OC	1	(Mahadevan et al. 2015)
19	On-Site Customer	Formal	BC	2	(Choudhury and Sabherwal 2003; Goh et al. 2013)
		Informal	CC	2	(Goh et al. 2013; Persson et al. 2011)
20	Open Workspace	Formal	BC OC	1	(Harris et al. 2009b)
		Informal	SC	2	(Harris et al. 2009b; Vidgen and Wang 2009)
21	Pair Programming	Formal	BC EOC	1	(Harris et al. 2009b)
		Informal	SC CC	4	(Harris et al. 2009b; Maruping et al. 2009a; Misra et al. 2009; Xu 2009)
22	Planning Game	Formal	BC OC	1	(Persson et al. 2011)
		Informal	SC CC	1	(Persson et al. 2011)
23	Practice Guides	Formal	BC OC	1	(Gregory et al. 2013b)
24	Sit Together	Formal	EOC	1	(Harris et al. 2009b)
		Informal	CC	1	(Harris et al. 2009b)
25	Slack	Formal	BC EOC	1	(Harris et al. 2009b)
26	Sustainable Pace	Informal	SC	2	(Vidgen and Wang 2009; Wood et al. 2013)
27	Unit Tests	Formal	OC	2	(Gregory et al. 2013b; Maruping et al. 2009a)
		Informal	SC CC	1	(Gregory et al. 2013b)
28	User Stories	Formal	OC	4	(Gregory et al. 2013b; Harris et al. 2009b; Mahadevan et al. 2015; Ramesh et al. 2012)
29	Whole Team	Formal	EOC	1	(Harris et al. 2009b)
		Informal	CC	1	(Harris et al. 2009b)

LEGEND: *Control Modes: BC = Behavioral Control, CC = Clan Control, EOC = Emergent Outcome Control, OC = Outcome Control, SC = Self-Control*

Table 3-2: Agile practices and associated control modes in literature (Study III)

The results are not limited to a distinct agile methodology; thus, they comprise practices for methodologies like Scrum or XP. From a control mode perspective, we identified 17 practices affecting behavior control and clan control, followed by 15 practices that are suitable to enable outcome control. Emergent outcome control can be enacted through 11 of our identified practices, whereas only 12 practices are said to support self-control in ASD teams (Table 3-3). We found no evidence in literature regarding practices that might affect input control.

Control Mode		Practices	#
Formal	Input Control	None	0
	Behavior control	1,2,3,5,8,10,12,13,14,15,16,19,20,21,22,23,25	17
	Outcome control	1,2,4,5,6,10,11,14,16,18,20,22,23,27,28	15
Informal	Clan control	4,5,6,7,8,10,11,12,15,16,17,19,21,22,24,27,29	17
	Self-control	3,5,7,10,11,15,16,20,21,22,26,27	12

Table 3-3: Practices per control mode (Study III)

Table 3-4 displays the results of the semi-structured interviews. We focused on an overall amount of eight distinct practices within qualitative data collection, as they imply to have effects on different types of control. These practices were chosen for two reasons: (1) the selected practices are supported by literature to enact different control modes and (2) the selected practices cover a broad range of control modes according to control theory (Choudhury and Sabherwal 2003; Kirsch 1997). Consequently, we focused on practices of Scrum, XP, and custom hybrid approaches as they represent more than two-thirds of agile methodologies used in software projects (VersionOne 2016).

No.	Agile Practice	Control Mode		#	FREQ.
1	User stories	Formal	BC, OC	7	5,71
2	Iteration Retrospective	Formal	BC	2	4,28
		Informal	CC	5	
3	Burndown charts	Formal	BC, OC	4	4,14
		Informal	CC	3	
4	Pair programming	Informal	CC	7	4
5	Backlog prioritization	Formal	BC, OC	4	3,85
		Informal	CC	3	
6	Code reviews	Informal	CC	7	3,71
7	Daily standups	Formal	BC	2	3
		Informal	CC, SC	5	
8	Collective code ownership	Informal	CC	7	2,85
LEGEND: *Control Modes: BC = Behavioral Control, CC = Clan Control, EOC = Emergent Outcome Control, OC = Outcome Control, SC = Self-Control; Freq.: frequency of usage (6 is high)*					

Table 3-4: Agile practices associated to control modes based on empirical data (Study III)

3.4.2 Comparison of Findings in Literature and Qualitative Data

Initially, we did not expect a high amount of control enacting agile practices specific to and dedicated for ASD. Within our total set of 29 practices, however, practices that focus on enabling formal control modes outnumber the overall amount of identified informal control enacting practices in ASD. The most frequently reported practices related to a particular type of formal control in ASD are, ordered by matches in literature, *iterative development, backlog prioritization/estimation, burndown chart* and *user stories*. Whereas *iterative development* and *backlog prioritization* seem to be suitable to enact behavior as well as outcome control, the usage of *burndown charts* and *user stories* in ASD are said to be applying outcome control only. On the other hand, the most frequently reported practices related to a specific type of informal control in ASD are *daily stand-up*, *iteration planning*, *iteration retrospective*, *pair programming* and *collective code ownership*. All these practices are suitable to foster both types of informal control, clan control as well as self-control.

Based upon the interview's results, all of the 8 agile practices could be assigned to control modes according to control theory. Only two practices could be assigned clearly, while the others were related more unambiguously. *Code reviews* and *pair programming* were both assigned to clan-control only. *User stories, retrospectives* and *collective code ownership* were said to support two different control modes. *Backlog prioritization, burndown charts* and *daily standups* even were associated to three different control modes. According to the results, self-control could only be enacted through *daily standups*, while almost every practice but *user stories* enact clan-control. 6 out of 8 practices were said to support behavior or outcome-control.

Table 3-4 also shows the frequency of usage of the same agile practices. The interviewees were told to rank agile practices on their frequency of usage inside the project they worked on. They could decide between a "0" that represents a non-existent usage or a scale from "1" to "6" with "1" representing the minimal level and "6" the maximal level of usage.

User stories, *retrospective* and *burndown charts* were used more frequently with a ranking between 4 and 5,71. A reason for the frequent usage of user stories is explained in the following quote.

> *"[…] the creation of user stories worked out quite well. Especially used for the initial planning to understand the whole requirements. What do they wish for and how will those requirements be developed? This was some kind of help for the whole team to understand what needs to be delivered in the future."*

It is noticeable that *user stories* were used by far the most with a frequent usage of 5,71 while *collective code ownership* got with 2,85 the lowest usage frequency. In contrast, *iteration retrospectives* ranked as the second important factor got a ranking of 4,28. Table 3-5 summarizes the overlapping and partially different results of step 1 and 2. With a focus on formal control, the review's results revealed that 23 out of 29 agile practices can be used to enact formal control. Similar results reflect our qualitative findings, 6 out of 8 practices are associated with formal control.

No.	Agile Practice	Interview results	SLR results
1	User stories	BC, OC	OC
2	Iteration Retrospective	BC, CC	BC, OC, CC, SC
3	Burndown charts	BC, OC, CC	OC, CC
4	Pair programming	CC	BC, CC, SC
5	Backlog prioritization	BC, OC, CC	BC, OC
6	Code reviews	CC	BC, OC, CC, SC
7	Daily standups	BC, CC, SC	BC, OC, CC, SC
8	Collective code ownership	BC, CC	CC, SC
LEGEND: *Control Modes: BC = Behavioral Control, CC = Clan Control, OC = Outcome Control, SC = Self-Control*			

Table 3-5: Comparison of control mode results (Study III)

3.5 Discussion

3.5.1 Summary of Findings and Implications

Building upon our pre-defined research question, the main goal of this research project was the literature-based identification and empirical evaluation of suitable control practices for ASD. Based on the results described in Section 4, we were generally able to provide answers to our research question and enhanced our knowledge on control in ASD projects from both a theoretical as well as practical point of view:

(1) *Providing future research directions for control-enactment and the effect on structure (formal control) and autonomy (informal control) in ASD teams.* Despite the known importance of control on the quality of SD project outcomes (Goh et al. 2013; Harris et al. 2009a; Mahadevan et al. 2015; Maruping et al. 2009a; Persson et al. 2011), there is so far no focused literature review that sheds light upon the question how far control and structure are needed in ASD projects, and how they can be applied through agile practices, while providing team autonomy at the same time. Our study closes this gap by providing detailed results derived from our three-step research approach as well as future research directions based on the existing research on ASD teams.

Building upon our work, and especially based on the differentiation of formal and informal control in ASD, we are able to extend our understanding on how ASD teams can be governed, especially in regards to the relationship between control and autonomy. Our list of agile practices and their impact on particular control modes revealed several interesting findings related to the topic of control usage in such projects.

We identified within our review's results a set of 23 agile practices that can be linked with the enactment of formal control types such as outcome or behavior control. In contrast, we found only 20 agile practices suitable for fostering informal control types such as clan control and self-control. Whereas 12 practices are dedicated to formal control types, there are 3 practices that affect informal control types only. This is surprising, since the underlying principles of agile methodologies (e.g. team autonomy) resemble more informal control types e.g. self-control. Following the Agile Manifesto, principles like "The best architectures, requirements, and designs emerge from self-organizing teams" or "Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done" (Beck et al. 2001a) provide evidence that informal control types are seen as much more important compared to the more formal and rigid control types like outcome control to agile methodologies. In contrast, we found out that the usage of a certain sets of common agile practices, which can be seen as the method-in-action (Fitzgerald 1997; Wang et al. 2012), potentially enacts high amounts of formal control within an ASD project. This leads us to conclude, that the enactment of formal control, and thus, structure within ASD teams, is necessary as it acts as an important counter-balance to team autonomy. The following quote focuses on enacted formal control and how it can help to improve the overall project outcome. This can help to get a better understanding for the overall need of formal control in ASD projects.

"We used daily standups as our meetings so we can discuss the progress of the project. Because we strictly performed these daily standups it was some kind of behavior controlling since everybody knows what you've done and what issues you are dealing with."

Moreover, we have recognized a lack of practices concerning the enactment of informal control types such as clan- and self-control. Although a lot of studies agree on the importance of team autonomy (Larman 2003b; Lee and Xia 2010a) or team empowerment in decision making (McAvoy and Butler 2009), our knowledge remains scarce about how to establish these principles in ASD teams. Our study provides first insights, that specific practices are well-suited to enact informal control. Especially the practice *daily stand-up*, having in sum most matches in literature, seems a very common enabler:

"We are also clan-controlled. We try to see each other every day and do the daily standups. It makes me think if I did not see my team today, I need to call them tonight and show them what I did and didn't work on today. "

Summing up, our research project revealed that, despite our general knowledge on suitable control-enacting practices for ASD, the exact relationship between the governance of control and autonomy within ASD teams and ASD project success is still unknown. Hence, we would recommend to increase the IS communities' research endeavor on this important topic. This could be done for example by an evaluation of control within ASD projects based on in-depth case study research. By applying such research methods in this context, we could further increase our understanding of how to implement the right kind of control within ASD projects.

(2) *Providing a first overview of control-enacting practices for ASD projects in practice*. As already mentioned beforehand, our knowledge on suitable control-enacting practices for ASD projects in practice remains scarce (see Section 1). Our study is, by certain means, able to cover this gap by providing a first overview of suitable practices in terms of exercising different types of controls. This list of practices, including references, allows practitioners working in ASD projects to evaluate the existing practices for general suitability and implementation fit within their projects. Hence, based on our list, we are able to provide first insights for practical application, which need to be amended by future research projects on this topic (e.g. in-depth empirical analysis of particular control practice suitability within different ASD projects settings).

3.5.2 Limitations and Future Research

While we were able to provide sufficient answers to our research question and enhance our knowledge on control in ASD projects, there are some limitations and corresponding future research directions that need to be acknowledged.

First, our research project considered relevant journals in the IS domain (based on the recommendations of the AIS and VHB) only. We did not take into account outlets, which focus for example on organizational control (in general) or cross-cultural studies. Hence, we cannot guarantee a complete analysis of the reference literature within our review. Nevertheless, due to the fact that ASD projects in particular are a phenomenon in the field of IS, we are quite sure, that our results are generalizable to a certain extend. However, we would recommend further literature reviews on this topic to even increase the coverage of the existing research on this topic.

Second, we need to address the topic of the broad perspective on *control* as a limitation of our research. By starting our literature review with a keyword search and also by following the guidelines of Levy, Ellis and Webster, Watson (Levy and Ellis 2006; Webster and Watson 2002a) in regards to forward and backward search, we tried to incorporate all past studies. Nevertheless, within the data analysis, we partially identified incongruity of different control mode definitions. While, for instance, Harris et al. (Harris et al. 2009a; Harris et al. 2009b) focus on the concept of *emergent outcome-control* as an alternate view on outcome-control in general, others still focus on the traditional outcome-control perspective closely related to classical control theory (e.g. (Kirsch 1997; Ouchi 1979)). The different associations result in a lack of transparency on the overall associations between agile practices and control modes. Based on this limitation, we would recommend further research, which explicitly focus on the comparison of control modes according to control theory in the light of suitable control enacting practices.

Third, one important limitation is the lack of experience regarding agile methodology use and strict role definition of all interviewees. A clearly defined role interpretation is fundamental for the usage of agile methodologies. The following quote provides an example of an interviewee's comment that supports this argument. In particular, the comment highlights weaknesses in the team-design which, in turn, leads to an emphasized development-mentality across all team members.

"I am not the scrum master. We are all part of the development team, even the scrum master. We do have a scrum master but everyone including the scrum master is also a developer and thus, responsible for creating and delivering working software every day [...]"

Furthermore, all the interviews were conducted with students, this means they generally lack experience compared to common employees working on an agile project. For example, 6 out of 8 interviewees worked on a project of this size and using an agile methodology for the first time. Thus, we recommend to extend future qualitative research to a wider field, comprising team participants such as senior developers, managers or certified scrum-masters on both, client and vendor site.

3.6 Conclusion

Uncertainty and changing user requirements in business and technology environments is ever-increasing. For companies, who want to stay competitive in SD, balancing control and autonomy to effectively deal with changing requirements has become an imperative, not an option. Given the complex relationships between control use and autonomy in ASD, project managers face difficult challenges in using control appropriately in ASD projects. While prior literature developed several frameworks to view control in ASD, little guidance is offered concerning which control modes are most efficient and how a control portfolio can be configured. Moreover, the body of knowledge lacks a comprehensive understanding on control enactment in general, e.g. how control and autonomy can be supported by utilizing agile practices. This research paper offers useful insights that are based on extant literature. Following Wiener et al. (Wiener et al. 2016), our goal was to examine how to enact distinct types of controls through selected agile practices. The results suggest agile practices are able to potentially enact distinct types of control and thus, supports project manager to choose suitable practices for their project. The authors conclude that agile methodologies are most efficient, when combined with formal control rather than exclusively informal control, such as clan and self-control. Control and autonomy in ASD are often viewed as negatively correlated. However, this research suggests why ASD can be flexible and controlled at the same time.

4 Part III: Modes, Styles and Congruence of Control in Agile Teams

Title	**„Loosening the Reins": Balancing Control and Autonomy in Information Systems Development**
Author Names	Tim Dreesen (Corresponding Author) Christoph Rosenkranz Phil Hennel Sean Hansen
University	University of Cologne Faculty of Management, Economics and Social Sciences, Professorship for Integrated Information Systems
Abstract	*Agile information systems development (ISD) strongly relies on social interaction and teamwork. Team processes and agile practices adopted by team members play an important role in the success of ISD projects. One crucial aspect is that agile practices promise greater responsiveness to change by granting higher levels of team autonomy. Existing studies on agile ISD, however, suggest that these projects can potentially benefit from various elements of control. In this study, we explore the enactment of control in agile teams and how control mechanisms influence team autonomy and team performance. In our analysis, we first propose a preliminary model based on extant ISD and teamwork literature to explain how the balancing of autonomy and control could work. We consider the ambiguous findings of the literature in the process by including and evaluating rival claims. We then present findings from testing this model in a multiple case study of five ISD projects in the insurance and software development industries. The evidence largely confirms our overall model and refutes rival hypotheses. Further, our findings suggest that the impact of control in agile teams is less a question of "what" controls should be exercised, but rather "how" controls are implemented in practice. The future direction we suggest for researchers and practitioners interested in control and ISD is to look beyond the traditional single-team, single-product context, when multiple teams must coordinate within a program, and team autonomy in projects may indeed become detrimental to overall program performance.*
Keywords	IS development, Project Management, Qualitative research, Control theory, Positivist case study, Agile methods
Paper Status	Submitted, under review in Information System Journal (ISJ)
Preliminary versions (already published)	Accepted, published in proceedings of the 53rd Hawaii International Conference on System Sciences (2020) (Best Paper Award: Software Technology Track - Agile and Lean: Organizations, Products and Development)

4.1 Introduction

In contemporary business environments, where the needs of consumers and business professionals change rapidly and continue to evolve over time, the ability to respond quickly to changing user requirements has become essential for information systems development (ISD) success (Lee and Xia 2010a; Maruping et al. 2009a; Vidgen and Wang 2009). Agile ISD approaches, such as Scrum (Schwaber and Beedle 2001) and eXtreme Programming (Beck and Andres 2004), have emerged in direct response to this growing demand for responsiveness and adaptability (Baskerville et al. 2011; Highsmith et al. 2001). In recent years, agile methodologies have become the *de facto* standard for ISD in industry, with most teams reporting the use of agile practices such as daily stand-up meetings (85%) or continuous integration (55%) (VersionOne 2020).

Despite the popularity of agile ISD methodologies and the purported higher performance of agile teams compared to older, structured ISD approaches, projects using agile ISD still fail, and the evidence of the approach's superior outcomes remains mixed (Niederman et al. 2018). Notwithstanding the growth of agile ISD, contemporary findings suggest that only 16% of software projects are successful (Jorgensen 2019). Indeed, project failure rates have changed little since 2001 when the "agile" designation was coined (Nelson 2005; Nelson 2007; Nguyen 2016).

All of this suggests that, despite the claims of some proponents, agile ISD has not been a "silver bullet" (Andrei et al. 2019; Fraser and Mancl 2008) for pernicious ISD challenges, with project success still depending upon a wide variety of factors (e.g., communication (Hummel et al. 2013), psychological safety (Hennel and Rosenkranz 2021), or agile practices (Recker et al. 2017)) affecting different facets of the ISD process (Siau et al. 2010). Contributing to this nuanced picture is the fact that there is not *one* agile ISD methodology adopted in an orthodox fashion, but rather a plethora of approaches, which are usually tailored to suit an organization's idiosyncratic needs (Campanelli and Parreiras 2015a; Fitzgerald et al. 2006b; Tripp and Armstrong 2018). Indeed, development teams may select specific agile *practices* from a portfolio of development techniques, project management practices, and standards and norms (Recker et al. 2017).

However, while each agile ISD methodology-in-use differs with respect to the key guidelines followed and practices employed, they generally share an emphasis on project teams that are empowered to make autonomous decisions and self-organize (Hoda et al. 2013; Masood et al. 2020; Xu and Shen 2016), with the project manager's role becoming more team-supportive than team-directive (McAvoy and Butler 2009; Remus et al. 2019). Accordingly, a critical premise for an ISD team to be considered 'agile' is the principle of *team autonomy* – that is, providing individual team members and groups the power to self-organize (Hoda et al. 2013; Moe et al. 2019) and the discretion of self-direction (Dikert et al. 2016; Moe et al. 2019). Despite its prominent role in agile ISD, studies have shown that team autonomy is not unambiguously beneficial. While autonomy is essential for teams to respond to novel challenges and opportunities (Lee and Xia 2005), it can also inhibit productivity and team performance (Langfred 2004; Maruping et al. 2009a; Yun et al. 2005). This is further complicated by two questions: (1) if autonomy should be granted on an individual level or promoted on a team level, and (2) how autonomy on one level affects the other level (e.g., Gregory et al. 2013a; Kirkman and Rosen 1999; Langfred 2000; Langfred 2004).

Due to the ambivalent results associated with team autonomy, the issue of *control* in agile ISD projects becomes increasingly important (Venkatesh et al. 2018; Wallace et al. 2004). Although the exercise of control necessarily implies certain limits on the ideal of team autonomy (Cordery and Tian 2017; Cram et al. 2016a; Orlikowski 1991; Shaw 1964), substantial research suggests that control leads to better performance within a team (Hackman 1987; Zellmer-Bruhn and Gibson 2006), even in agile contexts (Harris et al. 2009a; Kirsch et al. 2002; Maruping et al. 2009a; Persson et al. 2011). Furthermore, using (agile) ISD methodologies in combination with the exercise of some form of control has been shown to positively affect software quality (Grady 1993; Maruping et al. 2009a), team performance (Sun and Schmidt 2018), and client satisfaction (Shinkle et al. 2021).

Of course, control itself also is not a simple or singular concept, but rather one that encompasses a range of different mechanisms and forms. Most notably, there is a critical distinction between formal and informal modes of control (Keil et al. 2013; Kirsch 1997; Tiwana 2010). Other facets of the broad principle of control include the consideration of distinct control styles, the interaction between a controller and controllee, and the principle of control congruence (i.e., the shared understanding and perceived appropriateness of control mechanisms) (Wiener et al. 2016). Disentangling the nuances and effects of diverse control modes embodied in agile practices (anonymous for review), different styles used, and different enactments may have substantial importance for our understanding of the balance between control and team autonomy in agile ISD settings.

Despite a range of calls for further research on the effects of control and team autonomy in agile ISD (Chua et al. 2012; Wiener et al. 2016), the evidence remains ambiguous (Cram and Brohman 2013). Few studies have investigated different modes of controls (e.g., formal vs. informal) and their effects on performance and development outcomes (e.g., software product quality) or their enactment within agile practices (Cram et al. 2016a). Some studies have found that managers are still important for self-organizing teams (e.g., Garvin et al. 2013), using a more enabling control style (Remus et al. 2019). In sum, limited guidance exists on how agile ISD teams should be governed with respect to the relationship between control and team autonomy, with significant ambiguity regarding how much team autonomy and how much control are needed or the appropriate balance between the two (Cram et al. 2016a).

To address this gap, we examine the enactment of control and team autonomy in agile ISD (Choudhury and Sabherwal 2003; Tiwana and Keil 2009), the interplay between formal and informal control mechanisms (Persson et al. 2011; Tiwana 2010), and their relationship to team autonomy (Gerwin and Moffat 1997) and team performance (Kirsch et al. 2002). Consequently, the central research question guiding our study is:

> *"To what degree do control styles, control congruence, and modes of control embodied in agile practices influence team autonomy and team performance of agile teams?"*

In pursuing this question, we leverage both the long-standing insights of control theory (e.g., Kirsch 1996) and recently proposed novel conceptualizations of control in the context of ISD projects, specifically the concepts of control style and control congruence (Wiener et al. 2016). We found no existing studies dealing with control and agile practices that consider these extended control empowerment concepts and explore their impact on agile teams. Integrating these perspectives, we propose a preliminary model to explain the influence of control on agile ISD teams. Our objective is to deductively test concepts and relationships pertaining to control, team autonomy, and team performance. Specifically, we suggest that agile practices are likely to enact different control modes and, therefore, directly affect team performance and team autonomy. Moreover, we contend that, aside from the direct exercise of various modes of control, different control styles and degrees of control congruence influence both the behavior and outcomes of agile ISD teams. We also consider ambiguous findings in the extant literature by incorporating rival assertions. In so doing, we follow a positivist research epistemology and employ a theory testing, embedded multiple case study design (Dubé and Paré 2003; Lee 1991), following an approach similar to Sarker and Lee (2003). The focal cases reflect nine agile teams in five development projects across five different organizations.

The remainder of the paper is structured as follows. In Section 2, we provide an overview of related work on control and team autonomy in ISD. Building on this, we then develop a theoretical model, including competing, rival propositions, in Section 3. In Section 4, we describe our cases and the research methods used to assess and augment this model in detail. Subsequently, in Section 5, we present the results of our analysis. Finally, we discuss our results, implications, and limitations in Section 6.

4.2 Theoretical Background and Related Work

4.2.1 Agile Information Systems Development and Team Autonomy

Agile ISD is an umbrella term for a variety of distinct methodologies, such as Scrum, eXtreme Programming (XP), and Crystal (e.g., Martin 1991; Poppendieck and Poppendieck 2003; Schwaber 1995; Stavru 2014), which collectively emphasize an iterative development model, close collaboration between stakeholders, and a lightweight approach to project documentation (Cohen et al. 2004). Another common feature that characterizes these methodologies is their emphasis on significant flexibility and autonomy for project teams (Hoda et al. 2013; Wood et al. 2013). In agile ISD, the overall development process is not planned and scheduled upfront by an all-powerful project manager; rather, progress is made in short iterative phases, with decisions made collectively by the team as solutions evolve (Cockburn et al. 2001; Highsmith et al. 2001).

Flexibility and adaptiveness in agile ISD approaches are reflected in the concept of team autonomy (Larman 2003b; Lee and Xia 2010a). Prior literature provides various definitions of team autonomy and related concepts, including self-organization (Chow and Cao 2008; Highsmith et al. 2001; Hoda et al. 2013), self-management (Sharp and Robinson 2004), and team empowerment (Larman 2003b; Maruping and Magni 2012). Following extant research, we define *team autonomy* as “the degree of discretion and independence granted to the team in scheduling the work, determining the procedures and methods to be used, selecting and deploying resources, hiring and firing team members, assigning tasks to team members, and carrying out assigned tasks” (Lee and Xia 2010a, p. 90). As noted above, the emphasis on team autonomy in agile ISD stems from the assertion that the best outcomes emerge from teams that are given broad discretion in organizing and executing their work (Beck et al. 2001a).

In a business environment where available technologies, market structures, and customer preferences change rapidly, agile ISD approaches enable teams to react to emergent needs in a more timely manner than traditional structured development (Conboy and Fitzgerald 2007). When teams decide to apply an agile approach, various *agile practices* must be considered (Hummel et al. 2015; Pelrine 2011; Recker et al. 2017; Sarker and Sahay 2003). Agile practices can be described as methods-in-action and generative rules that are adapted to fit an ISD team's specific context (Highsmith et al. 2001). Examples of agile practices are *pair programming* (code is written with two programmers at one machine) and *collective code ownership* (anyone can change any code anywhere in the system at any time) from XP; similarly, popular Scrum practices include *daily scrums* (a daily stand-up meeting in which all project participants briefly review the status of their work) and *user stories* (a method to define broad, user-centered requirements while enabling creativity) (Cohn 2010; Harris et al. 2009b; Tripp et al. 2016).

4.2.2 Agile Information Systems Development, Control Enactment, and Team Performance

While agile ISD places emphasis on autonomous and self-organizing teams (Beck et al. 2001a) and many agile practices support such a self-governing approach (Lee and Xia 2010a), some degree of control must still exist (Harris et al. 2009a; Kirsch et al. 2002; Persson et al. 2011). Within our research, we define control broadly to mean "any process in which a person or group of persons or organization of persons determines [...] what another person or group or organization will do" (Tannenbaum 1962). In developing a theoretical assessment of the role of control in agile ISD contexts, we primarily draw upon Kirsch's control theory (1996; 1997; 2004), and specifically focus on extensions made by the expanded theoretical framework of IS project control (Wiener et al. 2016). Although particular ISD methodologies are not explicitly addressed within control theory (Cram and Brohman 2010), Kirsch points out that organizations in dynamic, changing environments may change control approaches over the course of an ISD project's lifecycle, resulting in the implementation of appropriate control types (Kirsch 1996; 1997).

With respect to agile ISD teams, theory distinguishes formal control types, such as input, behavior, and outcome control, from informal control types, such as self-control and clan control (Kirsch 1996). Table 4-1 summarizes key *control modes*, which are often exercised in concert rather than in isolation, representing a so-called control portfolio (Kirsch 1997).

Control Mode		Characteristics
Formal	Input Control	Measurable actions prior to implementing an activity, e.g., recruitment, training programs, or workforce allotments.
	Behavior control	Emphasizes behaviors, processes, and procedures that must be followed and offers rewards contingent on the adherence to the prescriptions.
	Outcome control	Involves outlining project goals and offering rewards contingent on their accomplishment. Emphasizes outputs regardless of the process used.
Informal	Clan control	Socializes team members into sets of valued norms. Emphasizes reinforcement of acceptable behaviors through shared rituals and experiences.
	Self-control	Provides autonomy to individuals to determine what actions are required and how to execute them. Emphasizes self-regulation of goals and self-monitoring of progress.

Table 4-1: Summary of control modes (Jaworski 1988; Kirsch 1996) (Study IV)

The exercise of formal control provides guidance and structure, assisting the development team in task execution (Kirsch et al. 2002; Remus et al. 2016). Importantly, traditional ISD approaches, such as structured or "waterfall" development models, rely heavily on such formal control mechanisms (Kirsch 1996; Kirsch 1997; Kirsch et al. 2002). By contrast, informal control provides developers greater discretion regarding how tasks are accomplished (Henderson and Lee 1992; Kirsch et al. 2002; Maruping et al. 2009a; Tiwana and Keil 2009). Thus, informal controls, such as clan control and self-control, provide the promise of greater autonomy, which is seen as an essential antecedent for responding to changing user requirements (Gerwin and Moffat 1997; Maruping et al. 2009a). The exercise of clan control allows the development team to identify important project goals and collectively determine how to attain them (Maruping et al. 2009a). The exercise of self-control similarly enables flexibility in pursuit of objectives, focusing on the role of the individual rather than that of the group. Self-control represents "the extent to which an individual exercises freedom or autonomy to determine both what actions are required and how to execute these activities" (Henderson and Lee 1992, p. 760).

Building on these principles, we posit that different types of control can be exercised through various agile practices. While some extant research has mapped agile practices to either formal or informal control modes (e.g., Harris et al. 2009b; Persson et al. 2011), conclusive determinations are challenging and elusive. Based on the results of an extensive structured literature review (*anonymous for review*), we identify a set of 29 distinct agile practices and analyze their correspondence to specific control modes. Although most agile practices defy a straightforward classification by control mode, a subset of these practices offer clear indications of formal and informal control modes in their enactment. For example, the use of the agile practice "user stories" can be seen as a formal control, as "they are a documented set of requirements (goals) to be achieved by development" (Gregory et al. 2013b, p. 5). Similarly, the use of "burndown charts" captures the progress of an agile team and is used to control for any deviations from an expected baseline (Gregory et al. 2013b; Mahadevan et al. 2015).

Appendix A[7] provides an excerpt of these practices and summarizes the corresponding control modes. In the following, we build on this subset.

[7] To access our appendices A to E see URL: https://osf.io/2b5f7/?view_only=7dca422b70884e878c9768cf76b12683

While most of the extant research focuses on controlling portfolio configuration ("what" control modes are used), few studies investigate "how" controls can be put into practice – that is, control enactment (Gregory et al. 2013a; Tiwana and Keil 2009). Specifically, *control enactment* is the interaction between a controller (the person exercising control) and a controllee (the target of control) – in other words, how the controller puts different modes of control into practice (Wiener et al. 2016). Consequently, we follow Wiener et al. (2016) in suggesting control style as another relevant concept for control enactment.

Control style can be defined "as the manner in which the interaction between the controller and the controllee is conducted" (Wiener et al. 2016, p. 755). Two contradictory control styles are particularly noteworthy – *authoritative* and *enabling* (Adler and Borys 1996; Gregory et al. 2013a). An *authoritative* control style is employed if strict behavioral compliance is desired, granting the controllee limited discretion in taking action (Wiener et al. 2016). An *enabling* control style, on the other hand, is used to achieve compliant behavior while granting flexibility in decision making to deal with uncertainties in daily work procedures (Adler and Borys 1996; Remus et al. 2016). These styles can be seen as ends of a continuum (Remus et al. 2016), and recent studies found that controllers can adopt an enabling or authoritative style at different times, for example, when performance issues become visible (Heumann et al. 2015).

In addition to modes and styles of control, a focus on "how" controls can be put into practice also directs attention to the concept of *control congruence*, which is the "level of agreement" or "degree of understanding" between a controller's and controllee's perception of the exercise of control (Narayanaswamy et al. 2013, p. 192; Wiener et al. 2016). The level of agreement regarding the appropriateness of controls is called *perceptual* congruence, whereas the degree of shared understanding of control measures is known as *communicational* congruence (Narayanaswamy et al. 2013; Ouchi 1978). Thus, control congruence may influence the quality of the whole control enactment process (Wiener et al. 2016).

Lastly, control is not enacted in a vacuum and for its own sake; we control ISD projects to steer their outcome to achieve desired results or states. The question of outcomes brings us to another critical concept of agile ISD dynamics, namely *team performance,* which is defined as the degree to which a team achieves its goals and how well its outputs match the team's mission (Hackman 1987; Zellmer-Bruhn and Gibson 2006). Based on prior research, we anticipate that the various facets of control discussed impact on both measures of team autonomy and team performance (Goh et al. 2013; Harris et al. 2009a; Mahadevan et al. 2015; Maruping et al. 2009a; Persson et al. 2011). However, the exact nature of those relationships remains very much in question, especially in the context of agile ISD (Cram and Brohman 2013).

4.3 Theory Development

In this section, we review the literature on control, team autonomy, and team performance to discern testable propositions regarding the role of agile ISD identified. Considering these inconclusive and partly contradictory results regarding control and the limited extant evidence concerning how control influences an agile ISD team, we propose a preliminary theoretical model to shed light upon these research gaps. The model in Figure 4-1 highlights the proposed interrelationships between control-enacting agile practices and control styles, control congruence, team autonomy, and team performance. To accurately capture the ambivalent state of the research discourse, we explicitly incorporate rival propositions in the model (italicized) where contradictory findings or rival assertions exist.

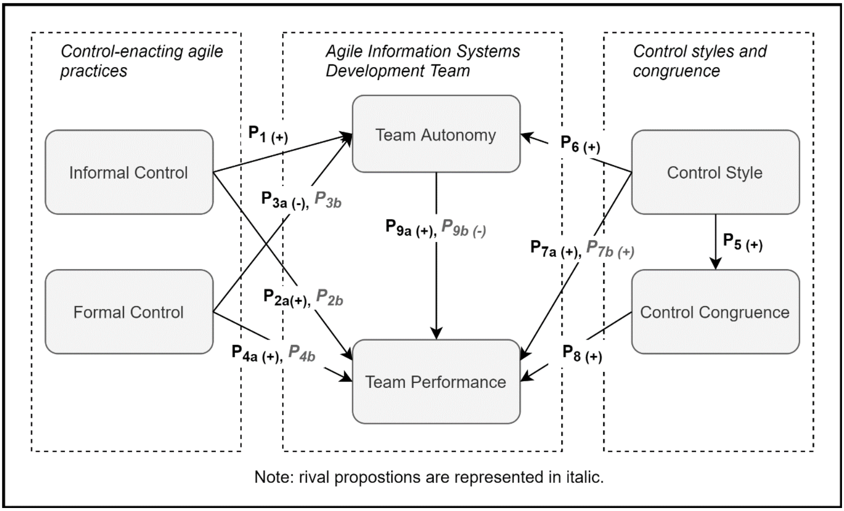


Figure 4-1: Proposed initial research model (Study IV)

From a control-enactment perspective, we include control modes in the form of control-enacting agile practices (see Appendix A), control style, and control congruence as independent variables in our research model. For a better to understanding, Table 4-2 briefly summarizes the constructs of our model.

Proposition / Construct	Definition	References
Control modes	Activities related to control are commonly categorized into so-called control modes, although a rough distinction can be made between formal and informal controls. Formal controls are enacted through a 'controller' via, e.g., plans, budgets, or other boundaries and are usually documented. Informal controls, on the other hand, are rarely documented and consist of norms and values which are in line with the organization's goals and are therefore promoted by a controller. By applying agile practices (see below), different control modes can also be addressed.	Dreesen and Schmid (2018); Harris et al. (2009b); Jaworski (1988); Kirsch (1996); Kirsch (1997); Ouchi (1979)

Control style	The way the interaction (e.g., how a manager sets certain rules or mandates due dates for status reports etc., according to organizational goals for his or her employees) between the controller and the controllee is conducted. Two opposite configurations can be distinguished: a) enabling and b) authoritative. An enabling style is characterized through continuous feedback, deviations from instructions, and a common understanding of the rationale of controls between the controller and controllee. An authoritative style, on the other hand, does not allow any deviations from instructions and enforces compliant controllee behavior.	Adler and Borys (1996); Gregory and Keil (2014); Remus et al. (2019); Remus et al. (2016); Wiener et al. (2016)
Control congruence	The degree of similarity between the controller (someone who exercises control) and controllee (someone who is controlled) perceptions of enacted controls. There are two types of perceptual congruence, communicational and perceptual congruence.	Cram and Wiener (2018); Narayanaswamy et al. (2013); Wiener et al. (2016); Wiener et al. (2019)
Agile practice	Are generative rules that are adapted to fit an ISD team's specific context. Recent literature categorizes practices to different such as 'management practices' (e.g., daily stand-up meetings), 'development practices' (e.g., pair programming), or 'standards and norms' (e.g., collective code ownership). Agile practices embody different control modes and thus can be utilized to enact control.	Cohn (2010); Harris et al. (2009b); Highsmith et al. (2001); Recker et al. (2017); Tripp et al. (2016)
Team autonomy	The extent to which discretion and independence are granted for the team to schedule the work, select adequate resources, choose suitable procedures and align resources to tasks.	Cordery et al. (2010); Hoda et al. (2012); Langfred (2000); Stewart and Barrick (2000)
Team performance	The degree to which a team achieves its goals and how well its outputs match the team's mission.	Hackman (1987); Zellmer-Bruhn and Gibson (2006)

Table 4-2: Summary of constructs and propositions (Study IV)

In this model, *team autonomy* and *team performance* are our dependent variables. In line with recent arguments regarding control in ISD (Wiener et al. 2016), we assert the need for greater consideration of control-enactment – that is, *how* software project leaders put distinct configurations of control portfolios into practice.

Regarding the effects of formal and informal controls, several studies find that informal control provides high levels of autonomy in managing assigned work tasks – for example, by enabling a team to determine objectives, tasks, and monitoring activities to achieve project goals (Kirsch et al. 2002; Remus et al. 2016). For example, self-controlling team members align their resources and choose methods for goal achievement without relying on a project leader (Henderson and Lee 1992; Maruping et al. 2009a). In particular, the use of self-control provides developers with discretion regarding how tasks are accomplished (Henderson and Lee 1992; Kirsch et al. 2002). Consequently, we propose:

P1: *Greater use of informal controls positively impacts team autonomy*.

Moreover, informal controls may help promote effectiveness, with some research emphasizing their performance-enhancing effect in the context of specific ISD projects (Chua et al. 2012; Tiwana and Keil 2009). For example, clan control can be promoted by establishing a collaborative culture within the team, allowing the controller to create an environment where the controllee has the freedom to make use of their own skills and knowledge in order to accomplish specific tasks, leading to better team performance (Chua et al. 2012; Gopal and Gosain 2010). Consequently, we propose:

P2a: *Greater use of informal controls positively impacts team performance.*

In contrast, there is also evidence in the literature that a positive effect of informal controls on team performance may not necessarily occur. For example, Tiwana and Keil (2009) found an non-significant effect on team performance, and concluded that the implementation of clan control in internal projects may be challenging because of a common discrepancy of shared goals, objectives, or priorities between controller and controllee (e.g., when both are represented by different departments such as business line functions and corresponding IT organizational units, and thus, mandatory social requirements for clan control are not met). This also applies to outsourced projects, in which a natural 'social distance' becomes a barrier to establishing clan control, and the embrace of shared values and beliefs is much less likely (Choudhury and Sabherwal 2003; Kirsch et al. 2002). Although Tiwana and Keil (2009) found some evidence for a more beneficial effect of self-control in terms of performance in internal projects, they also discovered that self-control in outsourced projects even hurts performance. In summary, based on these conflicting results, we formulate a rival proposition for P2a, which suggests that informal controls tend to show no or a negative effect on team performance:

P2b: *Greater use of informal controls does not or negatively impact team performance.*

A preponderance of extant research suggests a negative relationship between the exercise of formal control and the presence of team autonomy. This argument has a strong intuitive appeal, as formal controls can limit the degree of discretion enjoyed by a team (Piccoli et al. 2004, p. 366) by overemphasizing work formalization (Barker 1993; Remus et al. 2016). For example, routine team progress reports and strict adherence to schedules and task assignments may hinder a team's self-directedness, as teams frequently turn to managers instead of solving problems on their own (Piccoli et al. 2004; Robey et al. 2000). Furthermore, emphasizing functional specialization puts a manager in the position of controlling most decision-making, leading to decreasing team autonomy (Gerwin and Moffat 1997). Accordingly, a baseline expectation would hold that the use of formal control impedes the emergence of team autonomy. Therefore, we posit:

P3a: *Greater use of formal control negatively impacts team autonomy.*

At the same time, however, extant research suggests that the exercise of formal control has almost no discernible impact on levels of team autonomy (Adler and Borys 1996; Cordery and Tian 2017; Dalton 1959; Feldman 1989). The rationale underlying this assertion is that autonomy and control cannot be seen as empirically independent constructs, but rather as "inseparable aspects of managerial action", i.e., autonomy and managerial control have at least a minimal relation (Feldman 1989, p. 98). Dalton (1959) goes a step further, equating team autonomy with 'coerced freedom'; in other words, "freedom within a set of limited (controlled) circumstances" (Feldman 1989, p. 98). This suggests that bureaucratic (formal) controls are not necessarily dysfunctional (Adler and Borys 1996); rather, managerial control and the closely related exercise of authority and structural direction in the team may facilitate or even favor team autonomy (Cordery and Tian 2017). To address this conflicting insight, we offer the following rival proposition to P3a:

P3b: *Greater use of formal control does not impact team autonomy.*

Based on prevailing theory and evidence, the impact of formal control on team performance is likely to be quite different from its impact on team autonomy. In particular, formal controls provide some degree of guidance and structure, which supports the execution of tasks (Remus et al. 2016). Such controls may provide clear directions and predefined workflows on how to perform specific tasks (Kirsch et al. 2002) or recommend proven techniques or practices (e.g., user stories), which in turn positively affect team performance (Remus et al. 2016). Consequently, we suggest:

P4a: *Greater use of formal control positively impacts team performance.*

However, this expected positive effect of formal controls on team performance is not undisputed. For example, Tiwana and Keil (2009) could not find any positive effect of outcome control. Behavior control even showed contradictory results when looking at internal versus outsourced projects. They presume that penalties and sanctions of the controller commonly have little effect on controllees in case of deviations of formal controls, especially if the controller-controllee dyad exists through organizational units of one single company. In the context of outsourced projects, a realistic fear of "vendor interchangeability might obviate the need for outcome control because a controllee's prospects for obtaining future contracts provides a powerful disincentive from behaving opportunistically." (Domberger 1998; Lee et al. 2004; Nickerson and Zenger 2004; Tiwana and Keil 2009, p. 33). In light of these contradictory perspectives, we again present a rival proposition for the influence of formal control on team performance:

P4b*: Greater use of formal control does not impact team performance.*

As authoritative and enabling control styles can be seen "as end points on a continuum" (Wiener et al. 2016, p. 755), the two underlying facets of the enabling style – "repair" and "transparency" – are worth considering (Adler and Borys 1996). Both features establish an environment for the controllee that is characterized by feedback, involvement in the control configuration, and some degree of freedom to "deviate from controller prescriptions … in order to respond to real-work contingencies" (Remus et al. 2016, p. 7). In addition, the repair and transparency characteristics promote perceptual and communicational congruence (Adler and Borys 1996; Wiener et al. 2016). Taken together, this leads us to propose:

P5: *Greater degrees of an enabling control style positively impact control congruence.*

We also contend that, by its very nature, an enabling control style, with its emphasis on flexibility in decision making, will be more conducive to team autonomy (Wiener et al. 2016). This may be due to the repair and transparency characteristics, which allow for better knowledge exchange and continuous feedback loops (Adler and Borys 1996; Wiener et al. 2016). Therefore, we suggest:

P6: *Greater degrees of an enabling control style positively impact team autonomy.*

Additional exchange of knowledge, regular feedback, and close collaboration between controller and controllee leads to increasing team performance (Adler and Borys 1996; Remus et al. 2016). Conversely, a lack of information exchange and feedback mechanisms associated with an authoritative style leads to decreased team performance (Choudhury and Sabherwal 2003). Consequently, we posit:

P7a: *Greater degrees of an enabling control style positively impact team performance.*

P7a admits an alternative perspective. With respect to team performance, it may be beneficial if a controller uses a control style that restricts flexibility instead of promoting it. For example, Maruping et al. (2015b) found that a leadership style characterized by structuring, coordination, and management of the task pacing positively influences performance when time pressure prevails or is perceived as such by the team. This style shares common characteristics with an authoritative control style, as it can be seen as a rather top-down control style with strong formalization where deviations from controller prescriptions are seen as undesirable. However, studies of Wiener et al. (2016) and Gregory and Keil (2014) imply that neither the use of an authoritative nor the sole use of an enabling control style fulfills the requirements for achieving both performance and adaptiveness on a project level. Wiener et al. (2015) conclude that controllers with sufficient domain knowledge can achieve positive performance effects with an authoritative control style. This is in line with findings from team literature with a focus on performance-related outcome variables. For example, Hollenbeck et al. (2011) utilized structural contingency theory (SCT) (Burns and Stalker 1961) and argue centralized structures in teams (i.e., "the extent to which within-team decision authority lies solely with a team's leader" (Hirst et al. 2011, p. 626) may be more efficient, "because a central decision-maker may have a more holistic perspective on the task environment, may be well placed to ensure the dissemination of knowledge throughout the team, & may also provide a reliability check that reduces decision-making errors" (Cordery and Tian 2017, p. 114). In line with this conjecture, we formulate P7a's rival proposition:

***P7b*:** *Greater degrees of an authoritative control style positively impact team performance.*

Prior research indicates that team members' misunderstandings, poor relationships, and conflicts negatively influence the overall performance of a team (e.g. Nelson 2005; Nelson 2007). This raises the question of how congruent perspectives can be created between the controller and the controllee. For example, if controllers can establish perceptual congruence, this might be useful for obtaining feedback about the attempted control mechanisms. In line with Narayanaswamy et al. (2013), this might even "help to foster a climate in which disagreements can be discussed constructively and in turn boost team motivation" (Narayanaswamy et al. 2013, p. 211). Moreover, communicational congruence can be used to check communication mechanisms against their effectiveness, leading to transparency within the whole team and ensuring that both controller and controllee speak a common language in terms of objectives and tasks to be done to achieve these goals (Narayanaswamy et al. 2013). Consequently, we argue that a high level of control congruence positively impacts team performance, as it contributes significantly to the quality of the controls adopted and avoids negative socio-emotional effects such as decreased job satisfaction (Narayanaswamy et al. 2013; Tripp et al. 2016):

P8: *Greater degrees of control congruence positively impact team performance.*

Finally, one of the most ambivalent relationships of those considered concerns the influence of team autonomy on team performance. Team autonomy is central for agile ISD (see Section 4.2). Similarly, Studies of the impact of team autonomy on teams generally have concluded that increased team autonomy is positive for team performance and related factors, such as job satisfaction or well-being of team members (Cordery et al. 1991; Stewart 2006; Wall et al. 1986). Indeed, this perception if one of the central principles underlying the emphasis on team autonomy in agile ISD settings. At the same time, many studies suggest that the autonomy-performance relationship also is affected by other contextual factors, such as different working environments or types of work (Cordery et al. 2010; Haas 2010; Hyatt and Ruddy 1997; Stewart 2006). For example, Cordery et al. (2010) found that increasing team autonomy does indeed improve team performance. The extent to which performance increased, however, depends on the degree of uncertainty associated with the team task, suggesting that team autonomy yields higher benefits in uncertain contexts. Other evidence suggests that team autonomy may in fact inhibit productivity and performance in the context of project teams (Langfred 2004). Consequently, high levels of team autonomy can end up having negative effects (Barker 1993; Levy 2001; Pierce and Aguinis 2013). For example, Langfred (2007) argues that autonomous teams are particularly susceptible to conflict. Similarly, team autonomy means that development teams tend to consider many alternative problem-solving approaches, creating the risk that they could "lose sight of project objectives" and adversely impact performance and software quality (Maruping et al. 2009a, p. 383). In light of these competing arguments, rival propositions are certainly worth exploring in this relationship:

P9a: *Team autonomy positively affects team performance.*

P9b: *Team autonomy negatively affects team performance.*

4.4 Research Design and Method

4.4.1 Case Study Description

In order to evaluate the relationships between the different concepts and phenomena that we have introduced, we adopted a positivist epistemology and conducted a theory-testing, embedded, multiple-case study of nine teams in five projects across five different organizations (Lee 1991; Myers 2013, pp. 24-25; Yin 2003a, p. 52) (Table 4-3). We follow a hypothetico-deductive logic in our case methodology, emphasizing "falsification" over "confirmation" through deduction (Sarker and Lee 2003). An embedded multiple-case study approach for testing is particularly desirable when (a) the intent of the research is theory-testing, (b) extension of theory may be allowed (which is certainly the case with the existance of rival theories), and (c) higher degrees of the results' generalizability are sought (Benbasat et al. 1987), all of which applies to our study.

The cases in this study were sampled following a theoretical replication logic as we expect the different case conditions to engender contrasting results (Dubé and Paré 2003; Yin 2003a). All surveyed organizational units are based in Germany. We selected the cases for the following reasons: three of the cases, Apocorp, Dominsur, and Securefix, are set in large insurance companies – two of which (Apocorp and Securefix) are active internationally and one (Dominsur) nationally. Historically, the banking and insurance industry is regarded as culturally conservative, highly regulated, and capital-intensive (Gomber et al. 2018). Accordingly, firms in this sector are seen as relatively slow in adopting innovation (Bohn 2018; Cappiello 2018; McKinsey 2017). Consequently, we expect a comparatively high degree of hierarchy and formal control within the organizations. The two other cases, Unidevelop and Softac, are medium-sized software development companies. In light of the more dynamic and adaptive context of the IT sector, we expect both Unidevelop and Softac to reflect significantly flatter hierarchies and less formal control. Based on these distinctions, we expect to observe different characteristics of the control portfolio as well as the control styles exercised (and thus different results) across the case settings. Table 4-3 provides a short summary of the cases.

Characteristics	Case Organizations[1]				
	Apocorp	*Dominsur*	*Securefix*	*Unidevelop*	*Softac*
Description	Apocorp is a leading international insurance company. The company is in the process of agile transformation. Great efforts are being made to digitize large parts of the company within the next few years.	Dominsur focuses its core insurance business on the domestic market. Like Apocorp, Dominsur is in a phase of realignment. An agile approach will also be used here to digitize large parts of the product portfolio.	Securefix is a large European insurance company structured as a multinational group with several subsidiaries and business areas. Like the other insurance companies, Securefix undertakes an extensive digitalization approach	Unidevelop focuses exclusively on the development and deployment of software products (B2C and B2B). The focus is on web development and design. All of the employees are familiar with agile approaches to ISD.	Softac distributes a product consisting of hardware and software for the public sector as well as for clients in the private sector. The product is usually further customized for the special needs of the customer.
Industry	Insurance	Insurance	Insurance	Software Development	Software Development
Size	Large, international company	Large, national company	Large, international company	Small to medium, national	Medium, international
State of agile adoption	Agile transformation in process	Agile transformation in process	Agile transformation in process	Accomplished, 5-10 years of experience	Accomplished, more than 10 years of experience
Teams / Interviews	3 teams, 12 interviews including a project manager, a product owner, a scrum master, developers, and agile coaches	3 teams, 12 interviews, including two project managers, a product owner, a scrum master, developers, and agile coaches	One team, 5 interviews, including two project managers, one scrum master, a developer, and an agile coach	One team, 4 interviews including a project manager, a scrum master, and developers	One team, 4 interviews including two project managers and developers
[1] *Company names are anonymized for confidentiality purposes.*					

Table 4-3: Case overview (Study IV)

With respect to their ISD efforts, all of the insurance cases are in the process of organizational transformation initiatives, which started within the last three years. With the adoption and use of agile methods, all three companies have set themselves the goal of (a) digitizing the product portfolio and (b) achieving a better time-to-market for their products. All teams are working according to elements of the Kanban and Scrum methods. Unidevelop and Softac are more mature in their application of agile methods compared to the other organizations. Softac has many years of experience in the use of agile

ISD but, in contrast to Unidevelop, also has significant experience with non-agile methods (e.g., waterfall model or extended V-model). As a relatively young company, Unidevelop has only ever used agile practices for software development and consequently purports to develop software in a highly agile way.

4.4.2 Data Collection & Analyses

We collected data from various data sources and with different data collection methods. Semi-structured interviews and project documentation were the primary mechanisms of data collection. In all cases, we interviewed both project managers and project workers (see Table 4-3 last row for an overview of participating roles). Administrative documents, work descriptions, interview transcripts, and field notes were collected in a case study database. We collected data from July 2018 to November 2020 while conducting 37 face-to-face interviews across all cases. Our interview guides were derived from extant literature on control and teamwork. The interviews lasted on average about 60 minutes and were recorded and transcribed. The interview guide was not shared with the participants, and we only used it as a guide to the interview discussions. The aim was to encourage the participants to provide a narrative of their experiences as freely as possible. All insurance cases provided one team that was part of a bigger agile transformation task force, enabling us to gain an overview of all agile teams. All other participants from Apocorp and Dominsur were part of the development teams.

To achieve construct validity, we followed the case study guidelines recommended by Yin (2003a, pp. 33-37), including the use of multiple sources of evidence (multiple key informants and data collection types) and maintenance of a chain of evidence (project diary) during data collection. Furthermore, all key informants reviewed draft reports of the case study. With respect to internal validity, we took action in the form of pattern matching (linking the propositions and constructs to data from the case study diary) and explicit explanation-building. Since this case study was explicitly designed to evaluate the propositions of our model, we used replication logic in the setup of multiple case studies to ensure external validity. The multiple case study design was explicitly chosen to enable analytical generalization. To address reliability, we collected transcripts and created protocols from the interviews for each case in this study. Appendix C provides critical characteristics of the underlying research design and demonstrates how we assessed the case data.

Following Saldaña (2016), we applied multiple coding strategies and techniques. In the two-step coding process, we started to identify and refine our proposed constructs by means of *initial coding,* as described by Miles and Huberman (1994a) and Saldaña (2016). Initial coding served as a starting-point by providing ‘first ideas’, about certain control-related, team-related (e.g., linked to team autonomy or team performance) and especially even yet unidentified phenomena within the cases (Saldaña 2016; Strauss and Corbin 1998). The theoretical lenses of *agile practices* (Recker et al. 2017), the *expanded theoretical framework of IS project control* (Wiener et al. 2016), and *control theory* (Kirsch 1996; Kirsch 1997) served as guidelines in providing initial seed codes.

We integrated our findings from the initial coding effort and analyzed the different interdependencies and their impacts on newly identified constructs. To reduce the multitude of codes and to abstract at the same time, *structural coding* was done, primarily in order to relate different codes to specific themes, for example, linkage of codes to different modes of control according to control theory (e.g.,Kirsch 1996; Kirsch 1997). After our control-related and team-related categories and subcategories could be formed in this step, the next step was to learn more about their manifestation within the different cases. We therefore applied *magnitude coding*, providing more information about the intensity, for example of the occurrence of formal control. We therefore distinguished between three different degrees (high, moderate, low) to which specific controls could be identified, control styles were used, and control congruence between controllers and controllees was observed. These degrees were derived from the clarity of the statements made and their occurrence. For example, a high degree exists if more than half of the informants have made a clear statement; conversely, a low degree exists if no or few informants have made statements or such statements were not conclusive.

Next, we performed a second cycle of coding utilizing *hypothesis coding* (Russell Bernard 2002; Saldaña 2016; Weber 1990a). In this step, we sought to identify statements in the interviews to support or refute our propositions. In this step, we relied on theoretical lenses of control theory and their extension as proposed by Wiener et al. (2016) to identify potential effect on our dependent variables. Coding techniques and checklists (Miles and Huberman 1994a, pp. 170-244; Yin 2003a, pp. 109-138) were afterwards used to connect data with constructs from our model, and the propositions (cf. Appendix D and E for details; Appendix C gives a detailed overview of the attributes used to assess the case study's rigor). Essentially, hypothesis coding involves *pattern matching*, which enhances internal validity (Yin 2003a). This involves qualitative, logical deduction (Lee 1989) wherein an empirically-based pattern is compared against a predicted pattern derived from (rival) theoretical perspectives (e.g., Markus 1983). Consistent with hypothetico-deductive logic, this required us to search for patterns in the empirical material that were consistent (or inconsistent) with the patterns suggested by the theoretical propositions that were stated in a falsifiable and logically consistent manner. In our case, the comparison for P1, P5, P6, and P8 was with the null hypothesis (the absence of a relationship), and P2a (P2b), P3a (P3b), P4a (P4b), P7a (P7b), P9a (P9b) were compared with their corresponding rival propositions as stated within the brackets.

Figure 4-2 provides a detailed overview of the data analysis process. The first and second authors acted as coders and coded the transcripts to obtain coding reliability and frequency counts of how often control-related concepts (see Table 4-2) could be identified within each transcript. Intercoder reliabilities were calculated for each code drawing on the *proportional reduction in loss (PRL)* reliabilty measure proposed by Rust and Cooil (1994). This PRL reliability measure is similiar to Cronbach's alpha (using the same acceptable levels of reliability), but with advantages to more traditional reliability measures, such as Cohen's k, Cronbach's alpha, or the Perreault and Leigh measure, for assessment of qualitative data (Cohen 1960; Cronbach 1951; Perreault and Leigh 1989; Polonsky et al. 1997; Rust and Cooil 1994). The PRL was 0.95[8], which is well above the desired 0.70 cut-off level recommended (Rust

[8] Proportional interjudge/intercoder reliability (A) was 0.93, two judges, three categories, resulting in PRL = 0.95

and Cooil 1994). After reliabilities were calculated, the coders discussed and reached an agreement on remaining coding discrepancies in a face-to-face meeting.

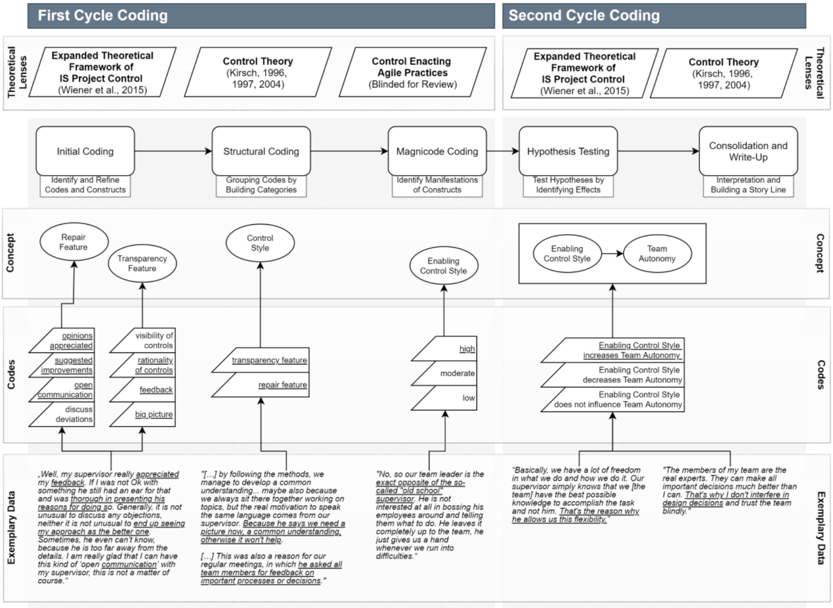


Figure 4-2: Coding process overview (Study IV)

4.5 Findings

We provide case vignettes with a mixture of full-scale narrative text (where a thoroughly "thick" description is needed), displays, and associated analytic text (Miles and Huberman 1994b, p. 243). Thus, the reader can reconstruct how the analysis developed, see how the explanation is grounded in the data, and assess the logical validity of our conclusions (Krathwohl 1998, p. 316). First, we present the identified control enactment concepts we observed in each of the cases studied. We then describe our coding of how these different concepts influence agile ISD teams in terms of the relationships pertaining to team autonomy and team performance.

4.5.1 Control Enactment

As the earlier theoretical review suggests, the distinction between "controllers" (i.e., ones exercising control) and "controllees" (i.e., those being controlled) is critical to any consideration of control dynamics. In all cases, control was exercised through managers (including top management) and scrum masters or product owners (controllers). Controllees were mostly developers, but also included other team members, such as software testers or architects. Each case revealed a distinct pattern concerning how (a) the concept of control enactment (e.g., certain degrees of evaluational control congruence) as a whole took place and (b) how these enacted controls impact the team. supports them with exemplary statements from the data. summarizes our control-related findings and supports them with exemplary statements from the data.

First, we found evidence of diverse formal and informal controls enacted across the cases. For example, top management exercised control through such aspects as team composition, the allocation of resources (e.g., the design of workspaces), or training (input control) (Jaworski 1988; Kirsch 1996), as well as the mandate to use an agile method, emphasizing processes and procedures that must be followed by teams (behavior control) (Kirsch 1997; Ouchi 1979).

> *"From the moment our top management decided to participate in the transformation program actively, it was very clear who was in charge. Since there were no roles between management and the team at that time, communication was straightforward from the top: 'These are my expectations of the output, and of course you have to adapt and to change your actions according to the plan.'" – Finch (Apocorp)*

Regarding the role of top management and their perceived control approach, a developer/manager at one of the insurers also noted:

> *"Well, management definitely tries to control it somehow, for example, to say, 'Yes, we're doing big-room planning in the short term and the decision has been made and there's no way around it.'" – Heath (Apocorp)*

In a clear point of contrast with our insurance industry cases, we found that Unidevelop and Softac teams tend to use fewer formal control mechanisms. There were specific guidelines regarding the applicability of concrete agile practices, but in general the teams in these firms could also decide in part which practices they would employ. For example, when asked what is expected of employees and what guidelines there are in terms of methods, practices, or tools, on developer noted:

> *"So kind of from the top or from management, there's no straightforward directives. But we have chosen the best things for ourselves, so to speak, and then consciously decided on these, and then, of course, you stick to them. If the idea now arises that you would like to use something else, that can, of course, be addressed. In terms of our tool usage, it has been proven over time that we mainly use Microsoft products, that is, tools like Visual Studio, VSTS, for software development." – Josh (Unidevelop)*

As we have noted, these agile practices imply certain control mechanisms and can in many cases be mapped to individual control modes (*anonymous for review*). Each of the practices summarized in Appendix A was observed across all cases. Also, it should be noted that in all cases an enabling control style, including the characteristics of "repair" and "transparency," was observed. However, we found that the predominance of an enabling control style is significantly lower in the case of Apocorp than in the other cases. We even have evidence of an authoritative control style in that case, undermining the repair and transparency features of the enabling control style:

> *"The team is managed with a rather 'strict hand' as far as the method is concerned! Um ... that means there is a less need-oriented adaptation of the process model." – Karen (Apocorp)*

A similar juxtaposition can be observed for the principle of control congruence. While all five cases reflect the pursuit of a common understanding between the controller and the controllee (communicational congruence), Apocorp demonstrates a deficit in the perceived appropriateness of some control mechanisms (perceptual congruence). For example, most of the interviewees of Apocorp observed or reported "resistance" within the team regarding the mandatory usage of agile practices:

"...oh God, not a retro again, it eats time, it eats capacity, I can't go on working then and really don't see the benefit." – Johanna (Apocorp)

	Apocorp	Dominsur	Securefix	Unidevelop	Softac
Control Mode					
Formal control	**High:** Teams use multiple practices that enact formal controls, especially behavior control. *"When they really start coding, of course, they have to follow certain guidelines in the group. There's no other way." – Claudia*	**High:** Teams use multiple practices that enact formal controls, including outcome and behavior controls. *"I have to make sure that the teams work according to the given model and process, e.g., [using] a reporting tool on which you can see how progress is." – Jane*	**High:** Teams use multiple practices that enact formal controls, like behavior control. *"There are also teams where it is stated that nothing may be committed to the repository without at least one other person having looked at it and had everything explained to them that was programmed there and also having seen tests and so on, so that is also a fixed rule." – Aaron*	**Moderate:** Teams use some practices that enact formal controls. *"We must use Scrum as the development process [...] any meetings are mandatory to keep everyone (including management) informed of development progress." – Josh*	**Moderate:** Teams use some practices that enact formal controls. *"You have to make sure that people are all working on the right things, i.e., coordination tasks, control tasks and also control as far as code quality is concerned in the end." – James*
Informal control	**Moderate:** Teams are partly responsible for making decisions and are provided with autonomy to do so. *"And we want them to be really self-organized and do the things that are best for the product and for the team... so that it works." – Claudia*	**Moderate:** Teams are partly responsible for making decisions and are provided with autonomy to do so. *"Control in the sense that we control ourselves as a team. How do we achieve the results we set ourselves for the sprint?" – Robert*	**Moderate:** Teams are partly responsible for making decisions and are provided with autonomy to do so. *"[We found out] that leadership is actually still necessary in agile ... It is no longer the functional direction, but that it is about being motivated and getting feedback ... [so that] each individual in this 'swarm' can achieve maximum performance." – Claudius*	**High:** Teams are mainly responsible for decision-making and provided with autonomy to do so. *"So there is basically little control from above here, i.e., from the management level, but more self-control or control by the team." – Josh*	**Moderate:** Teams are partly responsible for making decisions and are provided with autonomy to do so. *"So our daily routines were mainly shaped by the team. That wasn't predetermined [...] I expect each developer to work independently." – Tim*
Control Style					
Enabling-repair	**Moderate:** Regular controllee feedback is appreciated but not consistently demanded. *"So we [the transformation team] are always ready as contact persons [...] we offer always to approach us, so far nobody has done that." – Johanna*	**High:** Regular controllee feedback is provided and considered. *"And then we talked about it, and then I convinced them why such a [physical task] board makes sense from my point of view. But then I said, 'If you all want to remove it now, then we'll get rid of it.'" – Robert*	**High:** Regular controllee feedback is provided and considered. *"So in this respect, yes, control is important, but understood in the context of: I am interested in what my co-workers are doing, and I also want to give them regular feedback, because I want them to feel valued." – Pascal*	**High:** Regular controllee feedback is provided and considered. *"Decided finally, I think it's our boss. But on the basis of proposals we made to him. So we informed ourselves, proposed it, and then decided more or less as a team." – Miles*	**High:** Regular controllee feedback is provided and considered. *"Then I sit down with the developers and I know about what needs to be developed. And then we write down together how and what has to be built in detail." – Tim*
Enabling-transparency	**Moderate:** The rationale of controls sometimes remains unclear. *"I actually see the need that you realize that the rules you apply are appropriate to the situation, i.e., they should be supportive and not inhibitory." – Karen*	**High:** Reasons for enacted controls are well communicated. *"...but that we want to bring the knowledge into the teams and then also justify and advise why certain things make sense." – Bill*	**High:** Reasons for enacted controls are well communicated. *"So here again, talking to the individual people, that's the only way. In other words, really ask them what their fears are and where their difficulties lie." – Aaron*	**High:** Reasons for enacted controls are well communicated. *"So that other team members also know about different things, so it's not just the code now, but also how certain things are implemented. Certain quality standards that we might have or implementation standards will be listed there." – Kurt*	**High:** Reasons for enacted controls are well communicated. *"Yeah, we're trying to explain the background. To have such processes, what that means, why do you meet once and tell each other what you're doing? That's what we try to explain. That you don't just say something, but also understand why you do it in the end." – James*
Authoritative	**Moderate:** Control is partly enacted in an authoritative manner. *"The team is managed with a rather 'strict hand' as far as the method is concerned! Um...that means there is less need-oriented adaptation of the process model." – Karen*	**Low:** Limited evidence found for usage of an authoritative control style. *"And those who have been doing projects for 20 or 30 years now and were used to lead authoritatively, are now realizing that this is no longer the style of leadership with which one comes to success today." – Bill*	**Low:** Limited evidence found for usage of an authoritative control style. *"Scrum as a method is pretty much set, [but] everything else from Test Driven Development to Pair Programming and so on, is completely up to the teams. So even on the individual level, e.g. pair programming, if individual team members prefer to develop on their own, then they can do that too." – Aaron*	**Low:** Limited evidence found for usage of an authoritative control style. *"We team members can always suggest changes to routines or activities." – Josh*	**Low:** Limited evidence found for usage of an authoritative control style. *"Then I ask, "where is the problem" (if the person didn't say anything)? You have to tell me I have a problem. I'm not someone who just because you sit there looking sad, who asks, you need help. Because there is just too much to do for that." – Norton*

Control Congruence					
Communicational	**Moderate:** The need for a shared understanding is recognized, but controllers face difficulties in achieving it. *"It's important, but it actually has something to do with 'wanting to understand' and 'understanding the benefit of it.'" – Barbara*	**High:** Controllers ensure a shared understanding of the enacted controls. *"...that means you have to talk a lot, you have to explain why this is happening and what your goals are, that you mean well with it and that you still protect the people." – Cliff*	**High:** Controllers ensure a shared understanding of the enacted controls. *"So during the retro, we collectively work out how we want to work in the future ... But we are willing to cooperate in the team ... if we work something out together, we stand behind it." – Isabella*	**High:** Controllers ensure a shared understanding of the enacted controls. *"We [the whole team] actually agreed on Scrum, so I don't see any added value in changing the structure." – Rachel*	**High:** Controllers ensure a shared understanding of the enacted controls. *"You discuss these things with the developers. They also write down that you really know what was discussed." – Mary*
Perceptual	**Low:** Controls are often seen as "inappropriate." *"Oh God, not a retro again, it eats time, it eats capacity, I can't go on working then and really don't see the benefit." – Johanna*	**High:** Controls are usually perceived as 'appropriate.' *"And even if I come three times a day and change the priority and request information, they do it. I think that's great. There's no discussion...they understood that it's just like that in this project." – Jane*	**Moderate:** Controls are sometimes perceived as 'appropriate,' sometimes as 'inappropriate.' *[In response to the question of whether teams feel controlled:] "Yes, in the beginning, there was no clear understanding of why this was being done in the way it was, why it was good. Now, in retrospect, not anymore." – Pascal*	**High:** Controls are usually perceived as 'appropriate.' *"Of course, we have processes and rules that we have to adhere to [...] And I also see this basically formal process as important." – Kurt*	**High:** Controls are usually perceived as 'appropriate.' *"I feel very free here. I don't have the feeling that I am somehow restricted." – Mary*

Table 4-4: Findings for control modes, styles, and congruence (IVs) (Study IV)

4.5.2 Control and Influence on Agile Teams

While identifying specific control enactment concepts is important, equally substantial questions are how these concepts relate to one another and how they influence team mechanisms, such as autonomy and performance. Table 4-5 summarizes the extent to which we found evidence of an influence between control and team autonomy and team performance and lays the groundwork for the following hypothesis testing. In the course of the overall assessment, we distinguish between three levels of support for claims in the data: a) unsupported, b) weakly supported, and c) supported. A hypothesis is considered unsupported if no evidence is found or if less than half of all cases provide supporting evidence. Similarly, a hypothesis is considered supported if greater than half of all cases show clear supporting evidence. Unfortunately, not all statements can be clearly and directly assigned to a single hypothesis. Nevertheless, there are statements that implicitly suggest an indirect positive connection to a specific hypothesis. Weak support, therefore, is present when, for example, in a number of cases there are attenuated or indirect indications that nevertheless leave some room for interpretation.

Code group	Codes	Overall evaluation	Apo-corp	Dom-insur	Secure-fix	Unide-velop	Softac
Control Congruence	*...is increased by an enabling control style (P5)*	supported	(x)	X	X	X	(x)
Team Autonomy (TA)	*... is positively influenced by informal control (P1)*	weakly supported		(x)	X	(x)	(x)
	... is negatively influenced by formal control (P3a)	weakly supported		(x)	(x)	(x)	(x)
	... is not influenced by formal control (P3b)	not supported					
	...is positively influenced by an enabling control style (P6)	supported	X	X	X	X	X
Team Performance (TP)	*... is positively influenced by informal control (P2a)*	supported	(x)	X	X	X	(x)
	...is not affected by informal control (P2b)	not supported					

Code group	Codes	Overall evaluation	Apo-corp	Dom-insur	Secure-fix	Unide-velop	Softac
	... is positively influenced by formal control (P4a)	supported	(x)	X	X	X	X
	... is not influenced by formal control (P4b)	not supported					
	...is positively influenced by an enabling control style (P7a)	supported	X	X	X	X	X
	...is positively influenced by an authoritative control style (P7b)	not supported					
	...is positively influenced by control congruence (P8)	supported	(x)	X	X	X	X
	...is positively influenced by team autonomy (P9a)	supported	(x)	X	X	X	X
	...is negatively influenced by team autonomy (P9b)	not supported			(x)		(x)
'X' marks a clearly and frequently identified code supporting the corresponding proposition, while '(x)' marks a peripheral phenomenon or less clearly identified code							

Table 4-5: Dependent variables and hypotheses (Study IV)

Informants from Unidevelop and Softac stated that they experience high levels of autonomy within their teams. For example:

> *"The team itself has also been given a great deal of freedom from the management level. This means that from the very beginning it was up to the team to develop [software] what they thought was the right thing to do." – Josh (Unidevelop)*

In contrast, the informants from Apocorp and Dominsur felt somewhat restricted in their daily work routines.

> *"Well, I think they could be more autonomous and free, but they don't use it." – Claudia (Apocorp)*

> *"The degree of flexibility we have here helps. And I say 20 percent more flexibility, I think, would help even more." – Robert (Dominsur)*

From a control mode perspective, we found evidence across all cases that formal control is perceived as having a positive effect on team performance. Kurt (Unidevelop) argues:

> *"You need a certain amount of control to be able to keep the whole process under control and assess the process. Especially when it comes to meeting deadlines. Improving quality may also be another example. You must have a healthy level of both control and freedom."*

Similarly, we found substantial evidence of a positive influence of informal control on team performance (albeit slightly weaker than formal control). For example, Charles (Dominsur) and Miles (Unidevelop) state:

> *"Self-organization promotes motivation, communication and success (of a team)" – Charles (Dominsur)*

> *"Yeah, that's for sure. That's why we as a team decided back then that we would control all the pull requests from someone else, which means that another pair of eyes would look over it." – Miles (Unidevelop)*

In contrast, the effects of formal and informal control modes on team autonomy were less readily observable (i.e., reflected in a small number of codes and/or less clearly coded statements). Enabling control styles on the other hand, could clearly be observed to influence both team autonomy and team performance positively. Regarding team autonomy, Josh (Unidevelop) says:

> *"This means that from the very beginning, it was up to the team to develop what they thought was the right thing to do. This means that if we think that something is somehow beneficial, then we don't have to ask anybody; we can simply implement it. In the sense that we have complete freedom and as long as the result is right, everything is good."*

A positive effect of an enabling control style on team performance can be found in the following statement from Ben (Dominsur):

> *"Now, we (as people in charge) have even consciously taken ourselves back more and have simply tried to rely on the self-healing powers and self-responsibility of the team, to simply try it out. That actually worked quite well!"*

Finally, we see support in four of five cases regarding the positive influence of control congruence on team performance. Regarding a shared understanding of controls, Bill, a developer of Dominsur, argues:

> *"We still somehow speak a uniform language and not everyone else advises us in the team. Therefore, and I would say a bit of a success factor, it is important that we find a common line, that we develop common views on things [...] that is just important."*

Looking across all of the cases, we see significant support for several of the proposed control effects, while others reflect limited or countervailing evidence. These findings have important implications for both our theoretical understanding of the dynamics of control in agile ISD and the practical exercise of control in agile environments. In the next section, we explore these diverse implications in greater detail.

4.6 Discussion

The main goal of this research project is to shed light upon the question of control enactment in agile teams, taking into account the expanded theoretical framework of IS project control and explaining the impact on agile teams with respect to both autonomy and performance. Based on our findings, the comparison and contrast of the five cases significantly enhances our knowledge of control in agile ISD teams from both a theoretical and practical point of view. While substantial evidence supporting propositions P1, P2a, P3a, P4a, P5, P6, P7a, P8 and P9a was observed in most cases, we found no support or even falsifying indications for propositions P2b, P3b, P4b, P7b, P9b.

First, although the influence of formal and informal controls on team autonomy through the exercise of agile practices remains ambiguous, we certainly observe that such controls enacted by agile practices positively impact team performance. The ambiguity regarding team autonomy may relate to the fact that our case observations reflect mostly moderate levels (with the exception of Unidevelop) of informal controls, which are generally expected to provide high levels of autonomy in managing assigned work tasks (e.g., Kirsch et al. 2002; Remus et al. 2016). In the case of our insurance companies, these moderate levels of informal control can be partially explained by these firms still being in the process of agile transformation, suggesting the persistence of underlying hierarchies, structure, and formal processes. In one of the cases, an interesting note regarding self-control and a general rejection of some employees also became apparent, as the following statement indicates:

> *"Of course, there are also employees who specifically chose a job in the insurance industry. For example, because of the perception that the industry stands for values such as continuity and consistency rather than change and innovation. Employees who have been used to taking instructions and being guided for 20 years are suddenly expected to manage themselves and make their own decisions. Many are simply "not ready" for this or consciously do not want it." (Charles – Dominsur)*

Similarly, the idea that clan control can be challenging to implement in internal projects, as noted by Tiwana & Keil's (2009), could play a significant role in the limited degree of informal control observed. This is especially true if controller-controller dyads exist across different departments resulting in mismatches in priorities and organizational goals (e.g., goals of certain line functions versus objectives of corresponding IT departments). Having only moderate degrees of autonomy within teams that belong to insurance cases can also be explained by the fact that transitions of agile methods to a large-scale context took place in these cases. Although no specific large-scale agile framework, such as SAFe or Nexus, was used (Leffingwell 2020; Schwaber 2021), such agile multi-team settings require higher coordination and control mechanisms by mostly centralized units (Conboy and Carroll 2019; Dingsoeyr et al. 2019; Hobbs and Petit 2017). Consequently, common downsides of these approaches are threats to team autonomy, often resulting in lower flexibility and team responsiveness (Moe et al. 2019). Only Unidevelop, as a young company, seems to rely entirely on informal control mechanisms. Nevertheless, the positive influence of informal controls on team performance is in line with the results reported in the literature as previous studies emphasize their performance-enhancing effect in the context of specific ISD projects (Chua et al. 2012; Tiwana and Keil 2009). In particular, the use of self-control provides developers with discretion regarding how tasks will be accomplished (Henderson and Lee 1992; Kirsch et al. 2002). For example, self-controlling team members align their resources and choose methods for goal achievement without the involvement of a project leader (Henderson and Lee 1992; Maruping et al. 2009a). Similarly, the informal mechanism of clan control establishes an environment where the controllee has the freedom to make use of her own skills and knowledge to accomplish specific tasks, leading to better team performance (Chua et al. 2012; Gopal and Gosain 2010). Formal controls, on the other hand, provide some degree of guidance and structure, which supports the execution of tasks and leads to better team performance (Remus et al. 2016).

Second, our findings reveal that control styles play an important role in establishing control portfolios and have a significant impact on agile teams. In those cases where both characteristics of an enabling control style (repair and transparency) are readily observed, two critical implications can be gleaned: 1) the presence of an enabling control style reduces the likelihood of an authoritative control style, and 2) an enabling control style promotes a shared understanding (communicational congruence) and increases perceived appropriateness (perceptual congruence) of the controls enacted (Murungi et al. 2019; Narayanaswamy et al. 2013). While the former implication may appear unsurprising, it suggests that managers in the various case settings do not modify control styles based on the nature of tasks or areas of focus; instead, the enabling and authorative control styles appear to be largely mutually exclusive and indicative of a broader cultural context. This is in line with the arguments of Wiener et al.

(2016) and findings of Gregory and Keil (2014) that both styles are seldom found within a single controller but rather are commonly represented by two different managers with contrasting styles. The apparent influence of an enabling control style on both facets of control congruence is even more interesting, spurring speculation regarding the relative significance of the two dimensions of the enabling style (i.e., repair and transparency). The repair characteristic, on the one hand, may contribute to a generally better understanding (especially of the controls enacted) through the establishment of regular feedback mechanisms (Gregory et al. 2013a). On the other hand, the transparency feature of an enabling control style provides a "big picture" perspective (Wiener et al. 2016), which in turn might lead to both an increased shared understanding of the rationale of controls and increased perceived appropriateness of controls. Lastly, our results underscore a positive influence of an enabling control style on both team autonomy and team performance, supporting prior findings in the literature. Regarding team autonomy, the two characteristics of "repair" and "transparency" in particular have a positive effect on flexibility and thus team autonomy (Adler and Borys 1996; Wiener et al. 2016). Likewise, these characteristics facilitate additional exchange of knowledge, regular feedback, and close collaboration within a controller-controllee dyad, leading to increasing team performance (Adler and Borys 1996; Remus et al. 2016).

Third, our findings suggest that the concept of control congruence is important when control is exercised within agile teams. Our results reveal that, in four of the five cases, a high level of control congruence positively impacts team performance. This observation resonates with the results of recent studies, which indicate that control congruence contributes significantly to the quality of controls adopted and avoids negative socio-emotional effects, such as decreased job satisfaction (Narayanaswamy et al. 2013; Tripp et al. 2016).

While we believe this study provides valuable insights regarding the dynamics of control, team autonomy, and performance in agile ISD teams, there are a number of limitations worth noting. First, although we have intentionally employed a multi-case study research design to foster a rich investigation of team processes and enable exploratory theorizing regarding questions of control in these evolving organizational contexts, we recognize that this methodological approach implies limits on the generalizability of our findings. We, therefore, call for replication of our study in different contexts, with organizations of different sizes, industries, national settings, and overall levels of agility. Further, we used qualitative methods only, enabling us to focus on a single method and going into more detail but also limiting the reliability of our findings to a certain extent. By including quantitative methods and extending this analysis with a quantitative or mixed methods approach, future research could augment the reliability of our findings. Another limitation lies in the selection of participants. While all major roles of each team were interviewed, we did not conduct interviews with each and every team member. Perceptions of controls, styles, or congruence likely vary. Although the very consistent nature of the statements across interviews gives us significant confidence in the representativeness of respondents, we recognize that additional perspectives may not be represented. The final limitation is the influence of social desirability bias, as it is generally more socially desirable to report success rather than failure. Nederhof (1985) suggests postulating neutral questions. We tried to minimize the social desirability bias emerging from our questions. However, due to the clear preference for success over failure, social desirability bias was still likely to emerge from questions during our interviews.

4.7 Conclusion

In this paper, we explore the interplay between control modes, control styles, and control congruence and the resulting influence on team autonomy and team performance in agile ISD projects. We conducted a theory-testing, embedded, multiple-case study of nine agile ISD teams across five different case organizations. Our findings contribute to our understanding of control enactment in agile teams, taking into account the expanded theoretical framework of IS project control and explaining the impact on agile teams in terms of autonomy and performance. Based on our findings, our study enhances our knowledge of control enactment in a broader sense. Our insights revealed that a) control – independent of whether it is reflective of formal or informal control modes – positively influences team performance; b) an enabling control style is effective for enacting control in agile ISD environments as it facilitates team autonomy, team performance, and control congruence; and c) control congruence promotes team performance.

From a practical point of view, the results of our study can help managers or others who are in charge of managing agile ISD teams to select appropriate controls and control styles in order to simultaneously achieve desired team effects such as certain levels of autonomy and performance in the team. In particular, our study has two important implications for practice. First, managers, team leaders, or others who exercise control benefit from highly formalized controls in agile teams when enacted through agile practices. The formal controls issued in this manner can provide a structured framework in which desired behavior can be achieved or control of corresponding outcomes can be maintained, thus, positively affecting team performance. A similarly positive effect has also resulted from the use of informal controls, i.e., managers can draw from the entire control mode portfolio if the goal is to influence team performance positively. Second, the choice of an enabling control style shows overall positive effects in an agile team, as the associated characteristics of 'repair' and 'transparency' create the foundation on which team autonomy is established, congruence of controls can be achieved. Lastly, team performance can be increased.

We believe that the results of this study enhance our understanding of the dynamics and enactment of control in agile teams. In so doing, the research provides a valuable "seedbed" for further exploration of the control phenomenon in software development environments.

5 Part IV: Control and its Impact in Agile ISD Projects

Title	**“Directing Self and Others”: An Empirical Study of Control in Agile Information Systems Development**
Author Names	Tim Dreesen (Corresponding Author) Christoph Rosenkranz Phil Hennel Sean Hansen
University	University of Cologne Faculty of Management, Economics and Social Sciences, Professorship for Integrated Information Systems
Abstract	*This study investigates the relationship between agile practice usage, control and team autonomy in agile information systems development (ISD). Limited guidance exists on how agile ISD teams should be governed with respect to the relationship between control and team autonomy, with significant ambiguity regarding how control can be enacted without restricting an agile ISD's team being agile. We conducted a field study and based on matched-pair survey data of 148 supervisor-team member dyads, we considered the role of different control styles, the interplay between formal and informal control mechanisms as well as their relationship to team autonomy. We find that the choice of a control style significantly determines how agile practices are implemented and how agile practices and control mechanisms facilitate each other in an agile environment while it is still possible to achieve the desired degree of autonomy within a team.*
Keywords	Agile information systems development, agile practices, control enactment, control modes, control, control style, team autonomy
Paper Status	Submitted, under review in Information Systems Research (ISR) journal
Preliminary versions (already published)	Rejected for presentation at International Conference on Information Systems (ICIS) 2021

5.1 Introduction

Having marked the 20th anniversary of the Agile Manifesto and the subsequent decades of research on agile *information systems development* (ISD) (Diegmann et al. 2018), agile methodologies have now become the de facto standard for ISD in industry. Most contemporary ISD teams report the use of agile practices, such as iterative sprints, daily stand-up meetings, pair programming, and continuous integration (Baskerville et al. 2011; Highsmith et al. 2001; VersionOne 2020).

Despite the popularity of agile ISD and the purported higher performance of agile teams vis-à-vis structured and plan-driven ISD approaches, projects using agile ISD still fail, leaving evidence of the approach's superior outcomes mixed. For example, evidence indicates that, despite the growth of agile ISD, only 16% of software projects are successful (Jorgensen 2019) and industry-wide project failure rates have changed little since the "agile" designation was coined in 2001 (Nelson 2005; Nelson 2007; Nguyen 2016). These findings suggest that, despite the claims of some proponents, agile ISD has not been a "silver bullet" (Andrei et al. 2019; Fraser and Mancl 2008) for pernicious ISD challenges, with project success still depending upon a wide variety of factors, such as communication (Hummel et al. 2013), psychological safety (Hennel and Rosenkranz 2021), or self-organization (Hoda et al. 2013) affecting different facets of the ISD process (Siau et al. 2010).

One of the central challenges confronting all ISD teams is selecting from a wide variety of tools and techniques purported to support team functioning. In the agile context, teams may consider a broad portfolio of *agile practices,* including both targeted development techniques and more general project management practices (Recker et al. 2017), allowing teams to tailor their approach to their specific needs (Fitzgerald et al. 2006b). Despite this variety, agile ISD methodologies generally share an emphasis on the importance of project teams that are empowered to make autonomous decisions and self-organize their efforts to a significant extent (Hoda et al. 2013; Masood et al. 2020; Xu and Shen 2016), with the project manager's role becoming more team-supportive than team-directive (McAvoy and Butler 2009; Remus et al. 2019). Thus, an important antecedent for an ISD team to be considered 'agile' is the principle of *team autonomy* – providing individual team members and groups the power to self-organize (Hoda et al. 2013; Moe et al. 2019) and the discretion of self-direction (Dikert et al. 2016; Moe et al. 2019).

Although a considerable body of knowledge exists on the topic of autonomy in teams (e.g., Cordery et al. 2010; Gerwin and Moffat 1997; Moe et al. 2019), the benefits of team autonomy remain an open question. On the one hand, team autonomy has been found to reduce productivity and performance in the context of some project teams (e.g., Langfred 2004; Maruping et al. 2009a; Yun et al. 2005). On the other hand, it has been identified as an important factor in enabling teams to respond to novel challenges and opportunities or to increase team performance (e.g., Cordery et al. 2010; Lee and Xia 2005; Vidgen and Wang 2009). Moreover, the interdependencies between autonomy experienced at the individual and team levels are not well understood (e.g., Langfred 2004).

Due to the risks associated with team autonomy, the issue of *control* – broadly understood as "any process in which a person or group of persons or organization of persons determines … what another person or group or organization will do" (Tannenbaum 1962, p. 239) – becomes a prominent question

(Venkatesh et al. 2018; Wallace et al. 2004). As this definition suggests, the exercise of control necessarily implies certain limits on the ideal of team autonomy. Control itself encompasses a variety of different mechanisms and forms, with the concept's complexity being recently summarized in the expanded theoretical framework of IS project control (Wiener et al. 2016). The most popular control concepts are represented by a strict distinction between formal and informal modes of control (Kirsch 1997). Despite a range of calls for further research on the impact of control and team autonomy in ISD efforts (Chua et al. 2012; Wiener et al. 2016), the existing evidence remains ambiguous, especially in the field of ongoing development of agile ISD methodologies (Cram and Brohman 2013; Dreesen et al. 2020). Few studies have investigated control modes and their effects on agile ISD team characteristics, such as team autonomy, or their enactment within agile practices (Cram et al. 2016a).

Moreover, despite the emphasis on team autonomy, recent research highlights the continued significance of managers even in the context of self-organizing teams (e.g., Garvin et al. 2013), particularly with the application of a more enabling control style (Remus et al. 2019). In sum, limited guidance exists on how agile ISD teams should be governed with respect to the relationship between control and team autonomy, with significant ambiguity regarding how much team autonomy and how much control are needed, or what the appropriate balance between the two is (Cram et al. 2016a; Dreesen et al. 2020). Accordingly, we follow the call of Wiener et al. (2016) for further research on the inconclusive and partly contradictory results regarding control in ISD (Choudhury and Sabherwal 2003; Tiwana and Keil 2009), the role of different control styles, the interplay between formal and informal control mechanisms (Persson et al. 2011; Tiwana 2010), as well as their relationship to team autonomy (Gerwin and Moffat 1997). Consequently, this study's central research question is as follows:

"How do control styles, agile practices, and modes of control influence an agile ISD team's autonomy?"

To foster a deeper understanding of the interaction of control, agile practice usage, and team autonomy in agile ISD, we propose and test a model for identifying and investigating effects of control and agile practices on team autonomy in agile ISD teams. To the best of our knowledge, we are the first to take into account important elements of the expanded theoretical framework of IS project control (Wiener et al. 2016) and explore its impact on agile teams. Integrating these perspectives, we propose a preliminary model to explain the influence of control on autonomy of agile ISD teams. Our objective is to test the concepts and relationships pertaining to control, agile practice usage, and team autonomy. Specifically, we suggest that agile practices are likely to enact different control modes and therefore indirectly effect team autonomy. Moreover, we contend that, aside from the direct exercise of various modes of control through agile practices, different control styles influence the behaviour of agile ISD teams.

The remainder of this paper is structured as follows: We give a brief overview of related work on the topic of agile ISD research, focusing on the aspect of control from a team-level perspective. Next, we derive the proposed model and state corresponding propositions based upon extant research. This is followed by a description of the research method that we use to assess and augment the model. Subsequently, we present the results of our analysis and discuss the implications and limitations of the research.

5.2 Theoretical Background and Related Work

5.2.1 Agile Information Systems Development and Team Autonomy

Agile ISD is an umbrella term for a variety of distinct methodologies, such as Scrum, eXtreme Programming (XP), and Crystal (e.g., Martin 1991; Poppendieck and Poppendieck 2003; Schwaber 1995; Stavru 2014), which collectively emphasize an iterative development model, close collaboration between stakeholders, and a lightweight approach to project documentation (Cohen et al. 2004). Another common feature that characterizes these methodologies is their emphasis on significant flexibility and autonomy for project teams (Hoda et al. 2013; Wood et al. 2013). In agile ISD, the overall development process is not planned and scheduled upfront (e.g., by an all-powerful project manager); rather, progress is made in short iterative phases, with decisions made collectively by the team as solutions evolve (Cockburn et al. 2001; Highsmith et al. 2001). In a business environment where available technologies, market structures, and customer preferences change rapidly, agile ISD approaches enable teams to react to emergent needs in a more timely manner than traditional structured development approaches (Conboy and Fitzgerald 2007).

When teams decide to apply an agile approach, a variety of *agile practices* have to be considered (Hummel et al. 2015; Pelrine 2011; Recker et al. 2017; Sarker and Sahay 2003). Agile practices can be described as methods-in-action and generative rules that are adapted to fit an ISD team's specific context (Highsmith et al. 2001). Examples of agile practices include *pair programming* (code is written with two programmers at one machine) or *collective code ownership* (anyone can change any code anywhere in the system at any time) that originated in XP as well as *daily scrums* (a daily stand-up meeting in which all project participants briefly review the status of their work) or the use of *user stories* (a method to define broad, user-centred requirements while enabling creativity) from Scrum (Cohn 2010; Harris et al. 2009b; Tripp et al. 2016).

Flexibility and adaptiveness in agile ISD is reflected in the concept of team autonomy (Larman 2003b; Lee and Xia 2010a). Prior literature provides various definitions of team autonomy and related concepts, including self-organization (Chow and Cao 2008; Highsmith et al. 2001; Hoda et al. 2013), self-management (Sharp and Robinson 2004), and team empowerment (Larman 2003b; Maruping and Magni 2012). Following extant research, we define *team autonomy* as "the degree of discretion and independence granted to the team in scheduling the work, determining the procedures and methods to be used, selecting and deploying resources, hiring and firing team members, assigning tasks to team members, and carrying out assigned tasks" (Lee and Xia 2010a, p. 90). As noted above, the emphasis on team autonomy in agile ISD stems from the assertion that the best outcomes emerge from teams that are given wide discretion in organizing and executing their work (Beck et al. 2001a).

5.2.2 Control in Agile Information Systems Development

While agile ISD places an emphasis on autonomous and self-organizing teams (Beck et al. 2001a) and many agile practices support such a self-governing approach (Lee and Xia 2010a), some degree of control must still exist (Harris et al. 2009a; Kirsch et al. 2002; Persson et al. 2011). Within our research, we define control broadly to mean "any process in which a person or group of persons or organization of persons determines […] what another person or group or organization will do" (Tannenbaum 1962). In developing a theoretical assessment of the role of control in agile ISD contexts, we primarily draw upon Kirsch's control theory (1996; 1997; 2004) and specifically focus on extensions to it made by the expanded theoretical framework of IS project control (Wiener et al. 2016). Although specific ISD methodologies are not directly addressed within control theory (Cram and Brohman 2010), Kirsch points out that organizations in dynamic, changing environments may change control approaches over the course of an ISD project's lifecycle, resulting in the implementation of appropriate control types (Kirsch 1996; 1997).

With respect to agile ISD teams, theory distinguishes between two high-level modes of control: (1) formal control modes, focusing on the control of inputs, behaviors, and outcomes, and (2) informal control modes, reflected in processes of self-control and clan control (Kirsch 1996). Table 5-1 summarizes key *control modes*, which are often exercised in combination rather than in isolation, resulting in a diversified control portfolio (Kirsch 1997).

Control Mode		Characteristics
Formal	Input Control	Measurable actions prior to implementation of an activity e.g., recruitment, training programs or manpower allotments.
	Behavior control	Emphasizes behaviors, processes and procedures that must be followed, and offering rewards contingent on the adherence to the prescriptions.
	Outcome control	Involves outlining project goals and offering rewards contingent on their accomplishment. Emphasizes outputs regardless of the process used.
Informal	Clan control	Socializes team members into sets of valued norms. Emphasizes reinforcement of acceptable behaviors through shared rituals and experiences.
	Self-control	Provides autonomy to individuals to determine what actions are required and how to execute them. Emphasizes self-regulation of goals and self-monitoring of progress.

Table 5-1: Summary of control modes (Kirsch 1996; Jaworski 1988) (Study V)

The exercise of formal control provides guidance and structure, assisting the development team in task execution (Kirsch et al. 2002; Remus et al. 2016). Importantly, traditional ISD approaches, such as structured or "waterfall" development models, rely heavily on such formal control mechanisms (Kirsch 1996; Kirsch 1997; Kirsch et al. 2002). By contrast, informal control provides developers greater discretion with regard to how tasks are accomplished (Henderson and Lee 1992; Kirsch et al. 2002; Maruping et al. 2009a; Tiwana and Keil 2009). Thus, informal controls, such as clan control and self-control, provide the promise of greater autonomy, which is seen as an important antecedent for responding to changing user requirements (Gerwin and Moffat 1997; Maruping et al. 2009a). The exercise of clan control allows the development team to identify important project goals and to determine collectively how to attain them (Maruping et al. 2009a). The exercise of self-control – i.e., "the extent to which an individual exercises freedom or autonomy to determine both what actions are required and how to execute these activities" (Henderson and Lee 1992, p. 760) – similarly enables flexibility in pursuit of objectives, focusing on the role of the individual, rather than that of the group.

Building on these principles, we posit that different types of control can be enacted through the use of various agile practices. Based on the results of an extensive structured literature review (*anonymous for review*), we identify a set of 29 distinct agile practices and analyse their correspondence to specific control modes. A given agile practice can be linked to a formal control mode, an informal mode, or both. For example, the studies of Xu (2009) and Maruping et al. (2009a) identify pair programming as an enabler of informal controls, while the study of Harris et al. (2009b) links it to both formal and informal control modes. Gregory et al. (2013b) argue that agile project retrospectives implicitly exercise formal control (by formalizing outcomes within a meeting), while also strengthening team socialization and raising awareness of norms, thus, promoting informal control. This is in line with Mahadevan et al. (2015) who found retrospectives to address neither solely formal or informal controls, representing a hybrid form of control. Thus, while some extant research has mapped agile practices to either formal or informal control modes (e.g., Harris et al. 2009b; Persson et al. 2011), conclusive determinations remain challenging and elusive.

While most of the extant research focuses on control portfolio configuration ("what" control modes are used), few studies investigate "how" controls can be put into practice – that is, control enactment (Gregory et al. 2013a; Tiwana and Keil 2009). Specifically, *control enactment* is the interaction between a controller (the person exercising control) and a controllee (the target of control) – in other words, the way in which the controller puts different modes of control into practice (Wiener et al. 2016). Within the domain of control enactment, *control style* can be defined "as the manner in which the interaction between the controller and the controllee is conducted" (Wiener et al. 2016, p. 755). Two contradictory control styles are particularly noteworthy – *authoritative* and *enabling* (Adler and Borys 1996; Gregory et al. 2013a). An *authoritative* control style is employed if strict behavioural compliance is desired, granting the controllee limited discretion in taking action (Wiener et al. 2016). Conversely, an *enabling* control style is used to achieve compliant behaviour while granting flexibility in decision making to deal with uncertainties in daily work procedures (Adler and Borys 1996; Remus et al. 2016). These styles can be seen as ends of a continuum, and a good controller can adopt an enabling or authoritative style at different times, depending upon the situation at hand.

5.3 Theory development

In light of the inconclusive and partly contradictory results regarding control, the lack of understanding with regard to its relationship to team autonomy, and the limited extant evidence concerning how control influences an agile team, we propose a preliminary theoretical model in order to shed light upon these research gaps. Our research model in Figure 5-1 shows the proposed interrelationships between control style, control-enacting agile practices, modes of control expressed, and team autonomy. From a control-enactment perspective, we include both archetypal manifestations of control style as independent variables, and agile practices, control modes, and team autonomy as dependent variables in our research model.

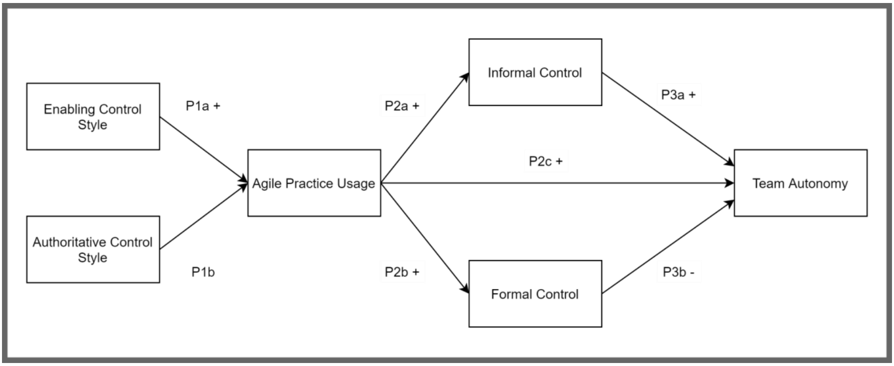


Figure 5-1: Research Model (Study V)

In this model, we conceive of team autonomy as the critical dependent variable of interest. In line with recent arguments regarding control in ISD (Wiener et al. 2016), we assert the need for greater consideration of control-enactment – that is, *how* software project leaders are able to put distinct configurations of control portfolios into practice.

Recent studies have shown the importance of control styles as essential components of the control enactment process (e.g., Remus et al. 2019; Remus et al. 2016; Wiener et al. 2017). An enabling control style has two main characteristics, "repair" and "transparency" (Adler and Borys 1996). Together, both features establish an environment for the controllee that is characterized by feedback, involvement in the control configuration, and some degree of freedom to "deviate from controller prescriptions … in order to respond to real-work contingencies" (Remus et al. 2016, p. 7). Consequently, we contend that, by its very nature, an enabling control style, with its emphasis on flexibility and bilateral communication, will be more conducive to agile practice usage (Wiener et al. 2016). This may be due to the repair and transparency characteristics, which allow for better knowledge exchange and continuous feedback loops (Adler and Borys 1996; Wiener et al. 2016). In contrast, we build on findings of Gregory et al. (2013a) who found authoritative control to generally deteriorate and intentionally neglect transparency, shared understanding, and communication – all three of which are important characteristics of agile methodologies (Hummel et al. 2013; Hummel et al. 2015). Hence, we propose:

P1: *Greater degrees of an enabling control style (a) increases the usage of agile practices while (b) a predominantly authoritative style does not promote agile practices usage.*

We further propose that specific agile practices are particularly helpful for project leaders because they potentially engender different modes of control. To date, few studies have attempted to assign some of these practices to specific control modes (e.g., Harris et al. 2009b; Mahadevan et al. 2015; Maruping et al. 2009a; Sun and Schmidt 2018). Based on our own prior analysis, only a few practices can be attributed precisely to a specific control mode, and most of the results of previous studies reflect contradictory findings (*anonymous for review)* (see Appendix A for more details). Yet, most studies agree that agile practice usage generally has the effect of promoting autonomy in teams (e.g., Hoda et al.

2013; Masood et al. 2020; Moe et al. 2019; Tripp et al. 2016). For example, planning meetings "provide teams with ownership and autonomy over the team goals and their personal goals" (McHugh et al. 2011b). Therefore, our second proposition is as follows:

P2: *The use of the agile practices (a) promotes formal controls, (b) informal controls, and (c) positively influences team autonomy.*

Regarding the effects of formal and informal controls, several studies find that informal control provides high levels of autonomy in managing assigned work tasks – for example, by enabling a team to determine objectives, tasks, and monitoring activities to achieve project goals (Kirsch et al. 2002; Remus et al. 2016). In particular, the use of self-control provides developers with discretion regarding how tasks are accomplished (Henderson and Lee 1992; Kirsch et al. 2002). For example, self-controlling team members are able to align their resources and choose methods for goal achievement without relying on a project leader (Henderson and Lee 1992; Maruping et al. 2009a). Similarly, clan control can be promoted by establishing a collaborative culture within the team, allowing the controller to create an environment where the controllee has freedom to make use of their own skills and knowledge in order to accomplish certain tasks (Chua et al. 2012; Gopal and Gosain 2010).

In contrast, some studies find that formal controls "limit the team's autonomy" (Piccoli et al. 2004, p. 366) by overemphasizing work formalization (Barker 1993; Remus et al. 2016). For example, routine team progress reports and strict adherence to schedules and task assignments may hinder a team's effectiveness, as teams frequently turn to managers instead of solving problems on their own (Piccoli et al. 2004; Robey et al. 2000). Furthermore, emphasizing functional specialization puts a manager in the position of controlling most decision making, leading to reduced team autonomy (Gerwin and Moffat 1997). Consequently:

P3: *(a) Greater use of informal controls positively impacts team autonomy, while (b) greater use of formal controls negatively influences team autonomy.*

5.4 Methodology

To test the model and propositions outlined above, survey instruments for two different online questionniares were developed. Specifically, we distinguished between *supervisors* (representing measurements on a team-level of investigation) and corresponding *team members* (representing measurements at an individual level), and accordingly developed two distinct questionnaires. Within the supervisor questionnaire we asked about team performance-related issues, while we asked team members about control aspects (e.g., control styles and modes), methodology usage, and the perceived degree of autonomy within the team. Both groups of respondents were asked to indicate how they feel about the congruence of the controls enacted. The use of two distinct questionnaires reduces potential problems arising from single informant and common method bias (Ko et al. 2005).

5.4.1 Data Collection

To identify appropriate projects and research participants, we reached out to a large international consulting firm, which has more than 500,000 employees and conducts projects all over the world. We identified agile ISD projects and respondents that satisfied the following criteria: First, projects had to be ISD projects. Second, each project had to follow an agile approach, but no particular methodology was prescribed. Third, both the supervisor and their corresponding team members must have jointly operated in this project. The research participants identified in this way include, among others, the roles of developer, scrum master, and team leader.

We obtained our sample data from a total of 286 individuals, participating in 89 agile ISD projects. These projects took place between August 2020 and March 2021 in different countries across three continents. Completed survey instruments from both supervisor and team member matching-pairs for each project were required, with responses dropped if only one part of this dyad was available. In total, we received completed questionnaires from 66 different teams (supervisors), resulting in 148 completed matched-pair data records (supervisor and corresponding team members). The client firms for which these projects were conducted can be categorized as medium to large international organizations. A central facet of each project focused on the development, improvement, customization, and/or implementation of information systems. For data collection, we used the online survey tool Qualtrics. Both surveys were provided to the respondents via a link included in an invitation email.

Individual Variables		Results	Individual Variables		Results
Education	general qualification for university entrance	2.04%	*Years of employment*	< 1	11.11%
	Bachelors's degree	48.98%		1 to 5	57.78%
	Master's or Diploma degree	38.78%		5 to 10	22.22%
	not specified	10.20%		10 to 15	17.78%
Years of employment	< 1	11.11%	*Roles*	Architect	3.92%
	1 to 5	57.78%		Business Analyst	5,88%
	5 to 10	22.22%		Developer	37.25
	10 to 15	17.78%		Scrum Master	15.69%
				Tester	9.8%
				other	13.73%

Table 5-2: Descriptive statistics of final survey sample (Study V)

Just over half of the projects (51.28 %) took place in an international setting, while 46.15% of the other projects consisted of national teams (in this context, within a single Central European country), for 2.6% of the projects we have no information. With respect to the applied methods within these projects, more than half of the projects (54.76%) applied the Scrum methodology, 4.7% employed Kanban, and 28.57% reflected a mixture of both Scrum and Kanban. All other projects (11.97%) used different types of agile hybrids (e.g., Scrum/Waterfall methodology).

Regarding team size, the study reflected a preponderance of smaller teams in line with the focus on agile methods, with 6.82% of the projects having fewer than five team members, 45.45% between five and 10 members, 31.82% between 10 and 25 members, 11.36% between 25 to 50, and 4.55% with more than 100 members.

The phases of the project in which the survey took place are distributed as follows: 13.95% of the projects were in an early stage, 74.42% of the projects were in a middle stage, and 11.63% of the projects were in final or end stage.

In terms of the distribution of the teams, only 16% of the projects included teams who were strictly co-located (e.g., on client's site), 44% of the projects had at least some team members geographically distributed (e.g., different offices), and 28% of the projects had team members geographically distributed and parts of the project work was outsourced to external vendors.

Further detailed information about the individual respondents' variables is provided in Table 5-2: Descriptive statistics of final survey sample (Study V).

5.4.2 Operationalization

We built a measurement model consisting of 11 latent variables, with one construct designed as a second-order construct. To operationalize our latent variables, we used only reflective perceptual measures for our constructs. All items are based on a five-point Likert scale (ranging from 'strongly disagree' to 'strongly agree'). The independent variables are "Control Style: Enabling" (CSE) and "Control Style: Authoritative" (CSA), representing both extremes of a controller's possible style to put distinct controls into practice. Regarding CSE, we used the existing scale of Remus et al. (2016). We self-developed items to measure CSA, as we found few existing measurements that met not our requirements. This is probably due to the novelty of the concept itself, but also due to the fact that most of existing studies focus on the emergence of an enabling control style (e.g., Remus et al. 2019; Remus et al. 2016). Since the two styles are seen as two endpoints of a continuum (Wiener et al. 2016), the argument is that the absence of one style (e.g., an enabling control style) implies the manifestation of the other style (in the example, an authoritative style). Nevertheless, we believe that measuring both opposing styles provides more clarity and contributes to the accuracy of the measurement model (Remus et al. 2019).

To measure the second-order construct "Agile Practice Usage," we had to determine which practices best match to the agile operations in the teams we investigated. Similar to previous studies (Maruping et al. 2009b), not all observed teams used the same agile methodology, such as Scrum or XP. Even among teams with an espoused methodology, the projects were not necessarily executed in a by-the-book fashion; rather, methodologies were largely adapted to meet the idiosyncratic needs of the various projects. Consequently, a comprehensive measurement of all existing practices was not feasible. We therefore imposed certain requirements on practices: First, we sought to identify practices that could be identified in literature to potentially address control, indicating either one distinct control mode or even a combination of more than one. Second, as agile practices can be distinguished as having a more technical (e.g., development practices such as continuous integration), a managerial focus emphasizing rules and behaviors for the exchange of information in meetings (e.g., daily stand-up meetings) or even those practices that prescribe 'standards and norms' which are socialized among team members and that should be followed (Recker et al. 2017; Tripp and Armstrong 2014), we sought to account for these three types of practices. Lastly, we sought to ensure that the selected practices were broadly used by our investigated teams and are popular in general (VersionOne 2020). As a result, we chose the management practice of “retrospectives” (RET), “code revisions” (CRV) as a standard and norms representing practice, and the development practice of “pair programming” (PPR). We used an existing reflective scale for measurement developed by Tripp et al. (2016) for both retrospectives and pair programming but self-developed items for the agile practice code revisions and designed them as first-order constructs in the model.

“Control modes” were measured by existing scales from the extant literature relating to each of one of the specific modes. Outcome control used items proposed by Maruping et al. (2009a), behavior control was measured by suggestions made of Tiwana and Keil (2009) and Kirsch et al. (2002), clan control used scales as proposed by Kirsch et al. (2002), and self-control was measured by a mixture of existing studies of Kirsch et al. (2002), Maruping et al. (2009a) and Remus et al. (2016).

To measure the dependent variable “Team autonomy” we drew on existing scales from Lee and Xia (2010a).

5.5 Data Analysis and Results

We transferred our research model into a structural equation model (SEM) and estimated this model using the maximum-likelihood algorithm. We favored a covariance-based (CB) SEM approach over a composite-based approach, such as partial least squares, for two reasons: (1) CB SEM is said to provide more accurate and consistent outcomes (Dijkstra and Henseler 2015; Hwang 2009), and (2) CB SEM is the method of choice when strong and established theories exist and the focus lies on confirming theoretically-assumed relationships (Hair et al. 2017; Jöreskog 1967; Rigdon et al. 2017). We used the MPlus 8.6 software for data analysis (Muthén and Muthén 1998), because it is suitable to handle multilevel data, as it exists in our case in form of nested data (team members) within teams (represented by supervisors). We also evaluated our results with the R package lavaan 0.6-7 (Rosseel 2012) and used different packages and tools within the R environment to prepare and manipulate data to make them ‘analysis-ready’ (e.g., dropping of incomplete data records, converting data types, and combining data in order to build our matched pairs of supervisor vs. team members).

Altogether, we used a final sample of 148 team member observations combined with 66 observations of corresponding supervisor responses. As the latter also accounts for the total number of observed teams, this results in a total sum of 66 observed teams with a cumulative total of 214 observations, resulting in an average team size of just over three respondents per team.

We tested our model for goodness of fit and determined the most common fit indices as a result. As can be seen in Table 5-3 the overall fit of our model is good (Hu and Bentler 1999; Kenny et al. 2014).

Fit indicator	Cutoff*	Result	Interpretation
Root Mean Square Error of Approximation (RMSEA)	< 0.06	0.046	Good fit
Comparative Fit Index (CFI)	> 0.95	0.958	Good fit
Tucker-Lewis Index (TFI)	> 0.95	0.953	Good fit
Standardized Root Mean Square Residual (SRMR)	< 0.08	0.074	Good fit

Table 5-3: Model Fit Indices (Study V)

* following Hu and Bentler (1999)

In terms of internal reliability, we used an omega coefficient instead of an alpha coefficient (e.g., Cronbach's alpha), as alpha is increasingly regarded as an inappropriate measure of internal consistency reliability (Dunn et al. 2013, p. 5). Table 5-4 summarizes McDonald's omega, composite reliability (CR), R^2, standard deviation, means, and corresponding item loadings of the latent variables. The values of both McDonald's omega and composite reliability meet the recommended thresholds (ω and CR > 0.7) for indicating convergent validity. Loadings on the designated variables were commonly higher than the recommended value of 0.7, except for items PPR3, MSC3 and TA1, which were just below this threshold (Hair et al. 2017). Instead of dropping the items, we decided to keep them because they belong to validated and established measures.

Construct	ω	CR	R2	SD	Mean	Item	Loading	Source of Items
AP: Retrospective (RET)	0.913	0.791	0.558	0.814	4.264	RET1	0.769	Tripp et al. (2016)
						RET2	0.749	
						RET3	0.722	
AP: Pair Programming (PPR)	0.864	0.844	0.145	1.158	2.484	PPR1	0.846	Tripp et al. (2016)
						PPR2	0.920	
						PPR3	0.662	
AP: Code Revisions (CRV)	0.977	0,979	0.230	1.200	3.936	CRV1	0.941	self-developed
						CRV2	0.958	
						CRV3	0.963	
						CRV4	0.977	
Control Style: Enabling (CSE)	0.796	0.792	n.a.	0.725	4.097	CSE1	0.811	Remus et al. (2016)
						CSE2	0.702	
						CSE3	0.727	
Control Style: Authoritative (CSA)	0.851	0.852	n.a.	1.00	2.295	CSA1	0.771	self-developed
						CSA2	0.824	
						CSA3	0.837	
Control Mode: Self Control (SC)	0.903	0.879	0.242	1.072	3.450	MSC1	0.961	Maruping et al. (2009a), Kirsch (1996)
						MSC2	0.713	
						MSC3	0.687	
Control Mode: Clan Control (CC)	0.916	0.909	0.549	0.969	3.962	MCC1	0.901	Kirsch et al. (2002)
						MCC2	0.895	

						MCC3	0.835	
Control Mode: Outcome Control (OC)	0.899	0.897	0.460	0.778	4.036	MOC1	0.891	Maruping et al. (2009a)
						MOC2	0.901	
						MOC3	0.791	
Control Mode: Behavior Control (BC)	0.908	0.909	0.367	0.995	3.493	MBC1	0.927	Kirsch et al. (2002), Tiwana and Keil (2009)
						MBC2	0.777	
						MBC3	0.920	
Team Autonomy (TA)	0.787	0.792	0.588	0.916	3.486	TA1	0.687	Lee and Xia (2010a)
						TA2	0.802	
						TA3	0.752	

Table 5-4: Construct Statistics and Item Loadings (Study V)

In addition, Table 5-5 provides values for construct correlations and the evaluation of the Fornell-Larcker criterion. Based on the square root of the average variance extracted (AVE) value of a latent construct being larger than its squared correlation with any other latent construct in the model, we conclude both convergent and discriminant validity is present in our data. In summary, the data achieves the relevant thresholds for reliability, convergent validity, and discriminant validity (Fornell and Larcker 1981; Hair et al. 2017; McDonald 1999).

We reduced the likelihood of common method bias and separated measurement by distributing different questionnaires at different points in time to supervisors and team members, respectively. Second, we reviewed correlations among constructs, as high correlations among constructs reflect a concern for common method bias. In line with the suggestions of Pavlou et al. (2007), the correlations do not exceed the recommended cutoff of 0.90, indicating that common method bias is not an issue in our data.

Construct	RET	PPR	CRV	CSE	CSA	SC	CC	OC	BC	TA
RET	**0.558**									
PPR	0.285	**0.650**								
CRV	0.358	0.183	**0.921**							
CSE	0.579	0.296	0.372	**0.560**						
CSA	-0.037	-0.019	-0.024	-0.241	**0.658**					
SC	0.367	0.187	0.236	0.381	-0.024	**0.711**				
CC	0.554	0.283	0.355	0.575	-0.037	0.364	**0.770**			
OC	0.507	0.259	0.325	0.526	-0.034	0.333	0.503	**0.744**		
BC	0.453	0.231	0.291	0.470	-0.030	0.298	0.449	0.411	**0.770**	
TA	0.490	0.250	0.314	0.509	-0.033	0.572	0.388	0.481	0.579	**0.560**

Table 5-5: Construct Correlations (Study V)

Figure 5-2 depicts the results of our structural model. Based upon the proposed relationships in our research model, our data reveals that an enabling control style (CSE) is strongly associated with agile practice usage (APU) (β = 0.81, p < 0.001), while an authoritative control style (CSA) shows no effect on agile practice usage (APU). Our data reveals that agile practice usage generally promotes control by addressing the full range of different control modes: all relationships are highly significant with p < 0.001 and all path coefficients reflect values of 0.49 or greater, except for the direct relationship between APU and team autonomy (TA), which was non-significant with values of β = 0.44, p > 0.05. Finally, team autonomy is positively influenced by certain informal and formal control modes. While behavior control shows a positive and significant association (MBC, with β = 0.29, p < 0.05) outcome control (MOC) shows not to be significant with β = 0.07, p > 0.05. With regards to informal modes, self control (MSC) is the sole informal control mode that is significantly and positively associated (β = 0.33, p < 0.01). Clan control (MCC) was not significantly related to team autonomy.

We also performed a post-hoc check for mediation for each of the control modes to determine indirect effects of agile practice usage on team autonomy. The analysis revealed a full mediation by self control (β = 0.31, p < 0.05) and behavior control (β = 0.33, p < 0.05). The model explains 59% of the variance in team autonomy, 62% in agile practice usage, 55% in clan control, 24% in self control, 37% in behavior control and 46% in outcome control, respectively.

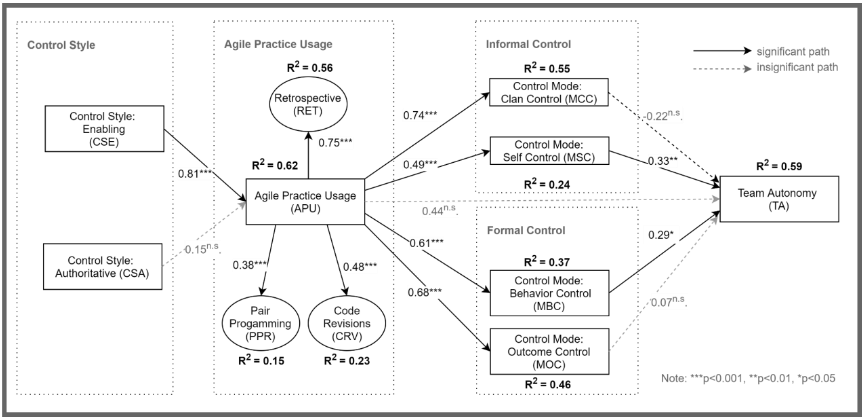


Figure 5-2: Structural Model Results (Study V)

Table 5-6 shows effect sizes for our tested and significant relationships. Summarized, we identified effects sizes ranging from 'small' to 'very large'. 'Very large effects' are concerned with associations between agile practice usage (APU) and corresponding control modes (MCC, MSC, MBC, MOC), while 'small effects' are concerned with associations between control modes (MSC, MBC) and team autonomy (TA). Lastly, we found a medium sized effect size for the relationship between an enabling control style (CSE) and agile practice usage (APU).

Relationship	Effect size	Interpretation*
CSE → APU	0.509	medium > 0.5
APU → MCC	1.536	very large > 1.2
APU → MSC	1.220	Very large > 1.2
APU → MBC	1.429	very large > 1.2
APU → MOC	1.259	very large > 1.2
MSC → TA	0.256	small > 0.2
MBC → TA	0.234	small > 0.2

Table 5-6: Effect Sizes (Study V)

* following Cohen (1988), Fritz et al. (2012), and Sawilowsky (2009)

5.6 Discussion

5.6.1 Summary of Findings

Our structural model substantiates most of our propositions while refuting others (see Table 5-7). Regarding our first proposition, we can confirm that a predominantly enabling control style has a positive effect on the use of agile practices (P1a), and therefore generally serves as an important antecedent for agile practice usage. This is corroborated by the fact that we found no evidence for a predominantly authoritative control style favoring the use of agile practices (P1b). Consequently, both propositions are supported by our results. This is line with a majority of studies dealing with leadership in self-organizing teams, which describe essential characteristics of suitable leadership approaches for self-organizing teams such as being 'light touch', 'adaptive', or 'feedback-driven' (Hoda et al. 2012; Takeuchi and Nonaka 1986). All of these are qualities that are taken up by the attributes 'repair' and 'transparency' of an enabling control style (Remus et al. 2019; Wiener et al. 2016).

In terms of our second proposition, we found agile practice usage generally to have a positive influence on control, independent of a specific control mode. In this context, our results show an almost equally distributed and strong effect on both informal and formal controls, supporting P2a and P2b. This result reinforces prior findings that (1) control modes can be addressed through the use of agile practices, and (2) that individual practices potentially can be indicative of multiple control modes (Dreesen and Schmid 2018; Gregory et al. 2013b; Harris et al. 2009b; Mahadevan et al. 2015; Maruping et al. 2009a; Xu 2009). We suspect that if additional practices are included in our second-order construct agile practice usage, the distribution of effects across the different modes of control will further equalize. However, this needs to be tested empirically in follow-up studies.

Although we do not find evidence of a direct effect between agile practice usage and team autonomy, we identified a complete mediation effect by behavior control and self-control, emphasizing the interplay and interdependency of agile practices and control enactment.

Proposition	Finding	Interpretation
P1a	A predominantly enabling control style promotes agile practice usage	Supports P1a
P1b	A predominantly authoritative control style does not promote agile practice usage	Supports P1b
P2a	Agile practice usage promotes informal controls	Supports P2a
P2b	Agile practice usage promotes formal controls	Supports P2b
P2c	Agile practice usage does not increase team autonomy on its own	Rejects P2c
P3a	Self control positively influences team autonomy	Partially supports P3a
P3b	Behavior control is positively associated with team autonomy, not negatively.	Partially Rejects P3b

Table 5-7: Interpretation of Results (Study V)

Lastly, team autonomy is positively influenced by the informal control mode 'self control,' but our results show no association between 'clan control' and team autonomy. We did expect such a result regarding self control, as literature indicates that the use of self-control provides developers with discretion regarding how tasks will be accomplished (Henderson and Lee 1992; Kirsch et al. 2002). For example, self-controlling team members align their resources and choose methods for goal achievement without the involvement of a project leader (Henderson and Lee 1992; Maruping et al. 2009a). Similarly, based on our data we assumed the informal mechanism of clan control to positively influence team autonomy. Previous studies found clan control to establish an environment where the controllee has the freedom to make use of her own skills and knowledge to accomplish specific tasks (e.g, Chua et al. 2012; Gopal and Gosain 2010). Regarding formal controls, we expected team autonomy generally to be negatively influenced. However, our data reveal different results. In our data, behavior control positively influences team autonomy. Although our data show a slightly positive association between outcome control and team autonomy ($\beta = 0.07$), no valid statement can be made here as the relationship is not significant with a value of $p > 0.05$. We did not expect such results, as the extant research suggests that formal controls hinder rather than promote team autonomy (e.g., Gerwin and Moffat 1997; Piccoli et al. 2004). Apparently, the choice of control style has a significant effect on how control modes actually work in an agile team. Strong manifestations of a control style driven by transparency and feedback may still be able to promote autonomy when behavior control is enacted. This corresponds to the findings of Harris et al. (2009a) and their concept of 'emergent outcome control', which combines restrictiveness (scope boundaries) with opportunities for dynamic feedback. Similarly, the concept of 'hybrid control', as utilized in Mahadevan et al. (2015, p. 80) ("the middle ground between structured a priori control mechanisms used in in the Waterfall approach and less structured, more-fluid emergent control mechanisms primarily used in pure Agile-development scenarios") or Gregory et al. (2013a) (i.e., simultaneous orientation toward improving efficiency and effectiveness and developing a shared understanding) both show that in agile environments the goals of formal control and desire of establishing degrees of freedom in an agile team are not contradictory.

5.6.2 Implications for Research and Practice

From a theoretical point of view, our research extends the body of knowledge of control in ISD in general, and in agile ISD in particular as it puts emphasis on the control enactment process within agile teams while going beyond the prevalent focus of control modes in past research (e.g., Choudhury and Sabherwal 2003; Kirsch 1997). In doing so, our work draws on control theory while incorporating important elements of the expanded theoretical framework of IS project control (Wiener et al. 2016). Our study's results connect to findings of previous studies, which found especially the concept of "control style" to be the preferable choice to predict and explain control consequences like performance or job satisfaction (Remus et al. 2019; Remus et al. 2020; Remus et al. 2016). We further extend this research with a focus on (agile) practices. To our knowledge, very little research exists examining control and agile practice usage simultaneously (e.g., Maruping et al. 2009a). On the one hand, these studies focus on specific agile methods and thus specific practices (e.g., Harris et al. 2009b; Maruping et al. 2009a; Maruping et al. 2009b) and therefore neglect a holistic view of agile practice usage in development teams, on the other hand, other studies follow an exclusively view on traditional control modes (e.g., Gopal and Gosain 2010; Kirsch 1997; Kirsch et al. 2002; Mahadevan et al. 2015; Maurer and Tiwana 2012; Tiwana 2010; Tiwana and Keil 2009). In addition, our study directly relates the concept of control style to the overall use of agile practices. Our results show that an enabling control style has a direct and immediate and beneficial effect on the use of agile practices. One explanation may be that the characteristics "transparency" and "repair" seem to be in line with the agile values and thus the practices themselves. In particular, a general emphasis on communication, continuous feedback, but also the absence of direct influence (as in a coercive style) seems to have a promoting effect.

Additionally, our research emphasizes the central role of agile practice usage within the control enactment process, as it links control style and control modes and also explains, how control can be enacted through agile practices in general. We are convinced that the use of any kind of agile practices (management, development or standards and norms representing practices) is a powerful tool for controllers to exercise control, at least when it is enacted through an enabling control style and at the same time the goal is to preserve team autonomy as much as possible. We thus tie in with previous findings that suggest that the choice of an appropriate control style significantly influences the effectiveness of control modes (e.g., Gregory et al. 2013a; Harris et al. 2009a; Mahadevan et al. 2015). These insights might open an avenue for further research on this topic, which extends our research with a more focused view on different leadership approaches (e.g., Druskat and Wheeler 2003; Maruping et al. 2015b; Rahmani et al. 2018; Spiegler et al. 2021).

Summarized, our research provides a rationale for how autonomy can be established in an agile team, specifically when control is enacted through agile practices by means of an enabling control style. Our study contributes to our understanding of control modes and their effects on agile ISD team characteristics, in this context especially team autonomy, and their enactment within agile practices.

Our study has three important implications for practice. First, managers, team leaders, or others who are in charge of exercising control no longer need to appeal to highly formalized controls in agile teams; agile practices (both management practices, such as planning meetings or retrospectives, and technical practices, such as pair programming or code reviews) are able to provide a structured framework in which desired behavior can be achieved.. Second, the choice of an enabling control style facilitates the implementation of practices in an agile team, as the associated characteristics of 'repair' and 'transparency' create the foundation on which agile methodologies are based. Third, choosing an enabling control style as well as enacting control using agile practices allows managers to minimize risk in the team by providing structure and managing behavior and outcomes, while still giving the team the ability to be responsive to unpredictable changes.

5.6.3 Limitations and Future Research Directions

The limitations of this study also recommend some opportunities for further research on this topic. The first limitation is the relatively small sample size. As noted by most of the researchers undertaking the effort to gather matched-pair data, collecting complete and consistent dyads of data records is a challenging task (e.g. Ko et al. 2005). However, our final sample of 148 complete matched-pairs (66 teams respectively) was sufficient to achieve trustworthiness regarding the operationalization of our model as well as to gain meaningful insights. Nevertheless, larger samples allow for more extensive analyses and inclusion of additional (dependent) variables of interest, such as job satisfaction (Tripp et al. 2016) or similar. Second, we constrained the design of our construct 'agile practice usage' to only three agile practices. Although we are convinced that these practices serve the study by (a) representing the distinct types of agile practices (i.e., management practices, practices representing standards and norms and development practices) well, and by (b) elegantly showing their relationship to different modes of control, we cannot exclude in the end that other practices would have induced different results. We would therefore encourage future studies to build on our research design while including other (different) agile practices. Third, we did not include all elements of the expanded control enactment framework by Wiener et al. (2016). For example, the consideration of a communicational as well as perceptual congruence of controls between the controller and the controllee, and thus a consequent possible 'loss of control', could provide interesting insights, especially when outcomes such as team or project performance are of interest (Huisman and Iivari 2006; Narayanaswamy et al. 2013; Ouchi 1979; Wiener et al. 2019). Again, we believe this could be considered as a further variable of interest in future studies. Finally, our study focused on explaining the relationships and mechanisms between agile practices, control, and team autonomy at a strictly 'within team' level. Cross-level considerations (e.g., the effect of different control styles on team performance or a between-team or between-project investigation) can contribute to our general understanding of the topic.

5.7 Conclusion

With this study, we aimed to advance our understanding of the use of agile practices and their connection to formal and informal controls. We found that both formal and informal controls are commonly used in agile teams and confirm recent findings on this topic, which indicate that agile practices and control mechanisms facilitate each other (e.g., Sun and Schmidt 2018). We have shown that the choice of a control style significantly determines how agile practices are implemented and controls are enabled in an agile environment, while it is still possible to achieve the desired degree of autonomy within a team. The results show that formal controls are feasible in agile projects, but also demonstrates that classical and authoritative control styles will reach their limits in projects where volatile requirements exist and agility is needed. Our results reveal that the choice of control style and the use of agile practices occupy elementary roles in the concept of control enactment, thus decisively enabling the coexistence of control and team autonomy at the same time.

6 ACKNOWLEDGEMENTS

This research was partially supported by a grant from the German Research Foundation (DFG) under record no. RO 3650/10-1.

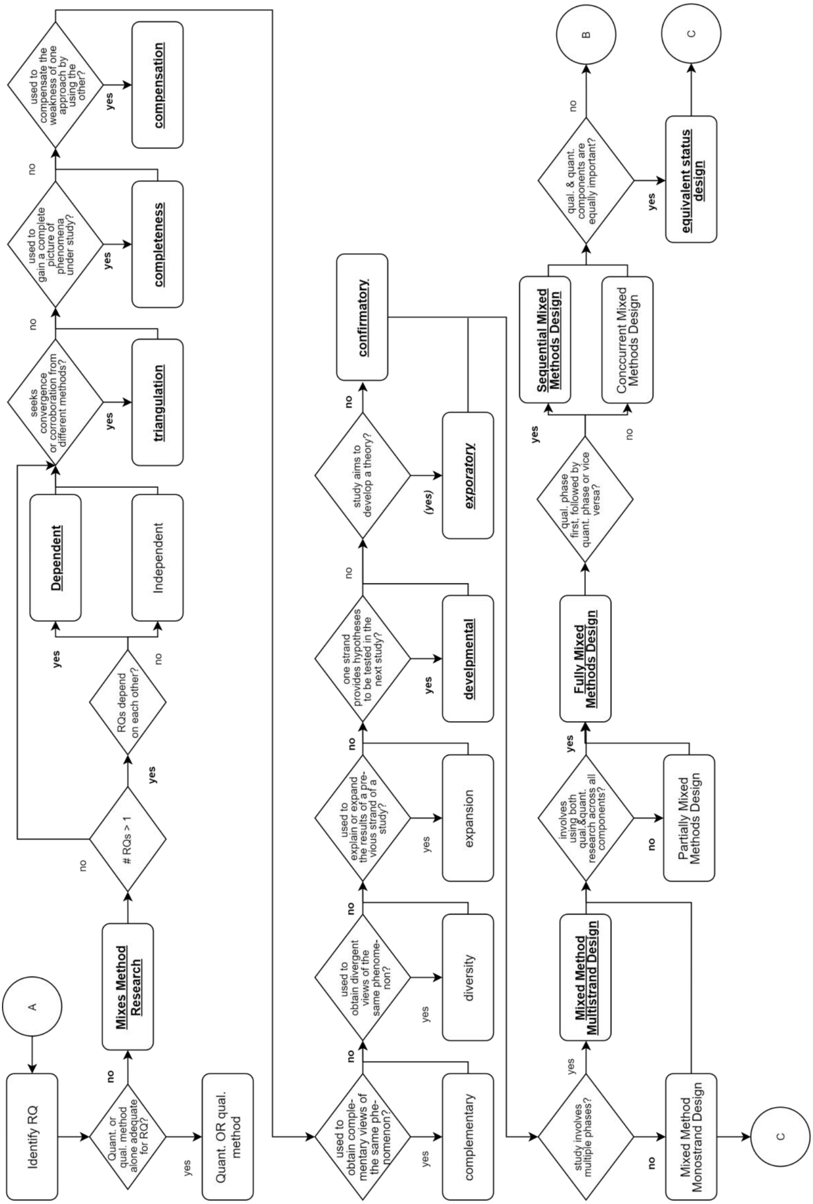
Appendix

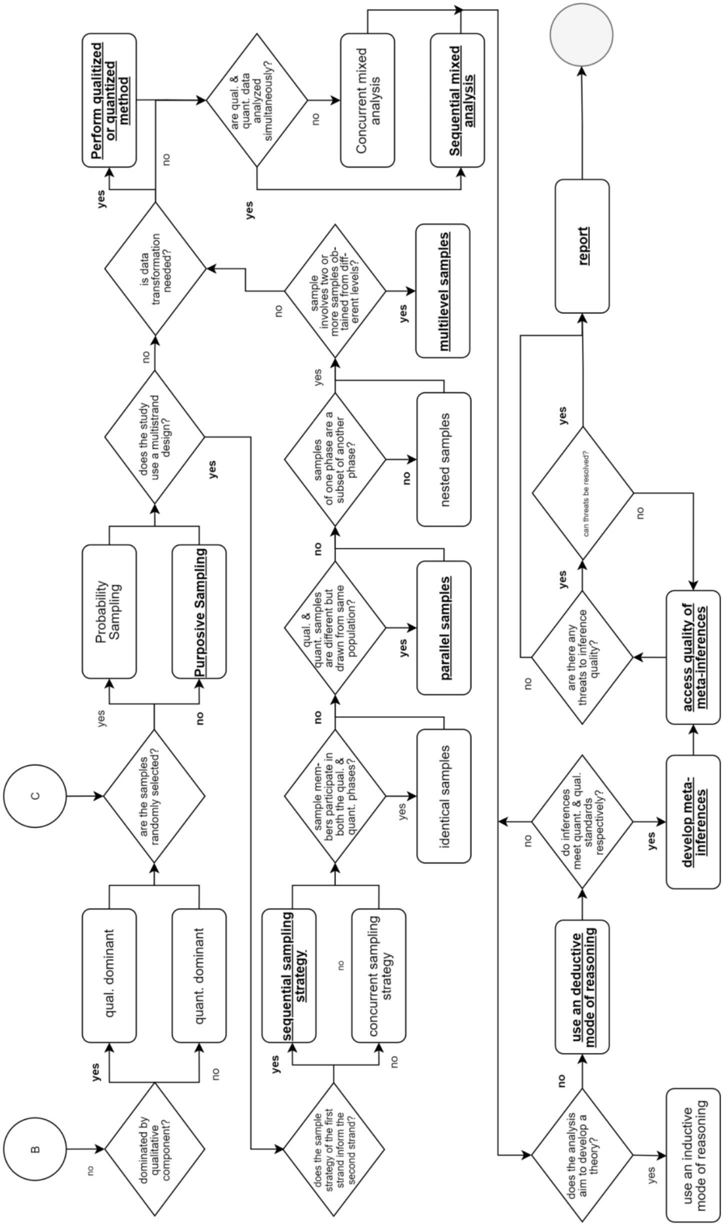
Appendix A – Glossary of concepts

Key Concept	Description	References
Control activity	Sometimes referred to as "controller activity". As noted Considers the choice of a control mode (what) and control style (how) as well as the degree of control enacted.	Cram and Wiener (2018); Murungi et al. (2019)
Control amount	Quantification of how much control is exercised via control portfolios, i.e., the variety of mechanisms used to exercise control over and the extent to which each of those mechanisms is used.	Rustagi et al. (2008); Wiener et al. (2017); Wiener et al. (2016)
Control choices	Sometimes referring to as "the choice of control portfolios", control choice describes the selection of control modes, control amounts and control styles within and/or across projects	Gopal and Gosain (2010); Kirsch (2004); Wiener et al. (2016)
Control congruence	Defined as the "level of agreement" or "degree of understanding" between a controller's and controllee's perception of the exercise of control. These two different types of congruence are usually referred to as *perceptual* congruence (perceived appropriateness of controls) and *communicational* congruence (shared understanding of control measures).	Narayanaswamy et al. (2013, p. 192); Wiener et al. (2016)
Control dynamics	Describe changes of a control portfolio in projects over time	Choudhury and Sabherwal (2003); Gregory et al. (2013a); Kirsch (2004); Murungi et al.
Control enactment	Focuses on the controller–controllee interactions and the implementation controls and therefore differs from the term "exercising control" as it constrains to a controller-perspective only	Wiener et al. (2016)
Control legitimacy	"[…] refers to the perception by subordinates that controls used within an organizational setting are appropriate in terms of justice, autonomy, group identification, and competence development […]"	Cram and Wiener (2018, p. 2)
Control mechanism	Any kind of activity initiated by the purpose of control, which establishes a certain mode of control	Kirsch (1997)
Control modes	Distinct type of control mechanism such as input, behavior, outcome (formal controls), clan or self-control (informal controls).	Jaworski (1988); Kirsch (1997); Ouchi (1979)
Control purpose	Can be understood as "as the intentions that underlie the controller's configuration (i.e., control modes) and enactment (i.e., control style) of controls" and can further distinguished between value-appropriation and value-creation purposes.	Wiener et al. (2019, p. 6)
Control style	Commonly described as the way how a controller interacts with a corresponding controllee, whereas two contradictory control styles can be distinguished – *authoritative* and *enabling*. An *authoritative* control style is employed if strict behavioral compliance is desired, granting the controllee limited discretion in taking action. An *enabling* control style, on the other hand, is used to achieve compliant behavior while granting flexibility in decision making to deal with uncertainties in daily work procedures.	Adler and Borys (1996); Gregory et al. (2013a); Remus et al. (2016); Wiener et al. (2016)
Control theory	"[…] theory [which] attempts to explain how one person or group in an organization can ensure that another person or group works toward and attains a set of organizational goals."	Kirsch (1996, p. 1)

Control type	Describes the purpose of the selected control mechanisms. Three types can be distinguished: a) procedural type, i.e., oriented toward improving efficiency and effectiveness (e.g. status reports), b) social type, i.e., oriented toward developing shared understanding (e.g. knowledge exchange activities) or c) hybrid, i.e., serving both goals simultaneously (e.g., onsite visits)	Gregory et al. (2013a); Jaworski (1988)
Controllee	Role in the controller-controllee dyad who is the subject of a control order	Chua and Myers (2018); Estevam et al. (2020)
Controller	Role in the controller-controllee dyad who exercises control over a controllee, i.e., taking some action in order to regulate or adjust the behavior of the controllee	
Emergent outcome control	Extension of classic control modes; combines restrictiveness (scope boundaries; formal control) with opportunities for dynamic feedback (informal control)	Harris et al. (2009a)
Hybrid control	Defined as "the middle ground between structured a priori control mechanisms used in in the Waterfall approach and less structured, more-fluid emergent control mechanisms primarily used in pure Agile-development scenarios"	Mahadevan et al. (2015, p. 80)
Team autonomy	Understood as the "degree of discretion and independence granted to the team in scheduling the work, determining the procedures and methods to be used, selecting and deploying resources, hiring and firing team members, assigning tasks to team members, and carrying out assigned tasks"	Lee and Xia (2010a, p. 90)
Team performance	The degree to which a team achieves its goals and how well its outputs match the team's mission.	Hackman (1987); Zellmer-Bruhn and Gibson (2006)

Appendix B – Decision tree for (mixed) method choice




B
no
dominated by qualitative component?
yes
no
qual. dominant
quant. dominant
C
are the samples randomly selected?
yes
no
Probability Sampling
Purposive Sampling
does the study use a multistrand design?
yes
no
is data transformation needed?
yes
no
Perform qualitized or quantized method
are qual. & quant. data analyzed simultaneously?
no
yes
Concurrent mixed analysis
Sequential mixed analysis
does the sample strategy of the first strand inform the second strand?
yes
no
sequential sampling strategy
no
concurrent sampling strategy
sample members participate in both the qual. & quant. phases?
yes
no
identical samples
qual. & quant. samples are different but drawn from same population?
yes
no
parallel samples
samples of one phase are a subset of another phase?
no
yes
nested samples
sample involves two or more samples obtained from different levels?
yes
no
multilevel samples
does the analysis aim to develop a theory?
yes
no
use an inductive mode of reasoning
use an deductive mode of reasoning
do inferences meet quant. & qual. standards respectively?
yes
no
develop meta-inferences
access quality of meta-inferences
are there any threats to inference quality?
yes
no
can threats be resolved?
yes
no
report

Appendix C – Interview Guideline (Study IV)

Success Factors in Agile Software Development

A study conducted by the University of Cologne

Interview Protocol (Excerpt)

Chief Investigators:

Dr. Christoph Rosenkranz
E-Mail: rosenkranz@wiso.uni-koeln.de
Professorship of Information Systems and
Integrated Information Systems
University of Cologne
Pohligstraße 1
50969 Cologne, Germany
Telephone: +49 (0) 221-470-5368

Tim Dreesen
E-Mail: dreesen@wiso.uni-koeln.de
Professorship of Information Systems and
Integrated Information Systems
University of Cologne
Pohligstraße 1
50969 Cologne, Germany
Telephone: +49 (0) 221-470-5373

Phil Diegmann
E-Mail: diegmann@wiso.uni-koeln.de
Professorship of Information Systems and
Integrated Information Systems
University of Cologne
Pohligstraße 1
50969 Cologne, Germany
Telephone: +49 (0) 221-470-5369

General Information

This interview is intended to explore your agile software development (ASD) process. We will be asking you about specific, current projects and we will try to understand the way you do things and why. Basically, we want to understand how control is enacted in agile software development and how the team members interact – we want to see how this affects the development team and the whole project the team is working on.

Background

- Please tell us about yourself as well:
    a. Background
    b. Job title and Role
- Please tell us about the information systems/software development [*change business unit title as appropriate for the firm*] unit within your organization:
    - What is the overall structure of the unit?
        - How many systems development teams do you have? [*If multiple, what is the basis for differentiation?*]
        - What are the key roles [formal or informal] on the teams?
        - What are the backgrounds of the team members?
    - Is the use of certain tools and practices mandatory?

- What discretion do project team members have in choosing the technologies and practices they will use?
- Is there a formal development methodology espoused by the organization?
    - If so, to what degree is adherence to this methodology enforced?
    - What guidance or direction regarding software development is provided by the executive level of the organization?
- Please describe what you perceive as the most important success factor for your team with respect effective and efficient software development within your organization.

- Tell me about the goal project(s) you are actively involved with:

Let's take the first project…. *(focus on this if only one, but repeat these questions for each project if applicable)*

Assessment of Current Practice – Activities and Routines

- How far along is the project? How's it going? At a very high level, could you walk me through the history of the project and the future plans for the project? [*Probe freely for details*] Tell me about the structure of your team [*probe for backgrounds, gender*, …] and the regular activities within your team – who does what and why? [This question may reference the earlier overview of team structure]
- What are the (key) roles (e.g. Scrum master, agile coach, project manager) or positions on this project?
    - What activities or routines (i.e., agile practices) do you see as central to the project? [probe for agile practices: iterations, daily stand up meetings, pair programming, collective code ownership, …]
- How do team members communicate within the project?
    - Which media or tools are used for this [also probe for formal/informal communication mechanisms such as face-to-face vs. diagrams and specifications]
    - Are there any expectations with respect to who should or may speak to whom?
    - Do team members talk freely to one another – Do they talk only about work-related or also about personal topics? Do team members know each other personally?
- Who defines, selects, and oversees the activities and routines that are used on the project? [Ask for examples!]
    - Who can become a ………….. (insert position)? [probe also for how an individual may leave or must leave a position]
    - What tasks, decisions, and obligations are associated with being a (insert position)? [*Choice rules*]
- How do those individuals ensure that the activities and routines are carried out in the way that they prefer? [*Control enactment*]
    - *Do* they take measures to ensure that things are done in a certain way?
    - If so, do they allow for modifications to the routines based on the preferences of team members?
- [*Using the response to the previous question*] How would you characterize the interaction between [*controller's title*] and the other members of the team? [*Control style*]
    - How often do they communicate with each other?
    - What is the nature of the discussion or communication?
    - [If necessary based on the preceding probes] In these interactions, would you describe [the controller] as acting more like a boss or a collaborator?
    - Are deviations from certain instructions tolerated or is there a chance that they can be adapted through frequent communication (feedback)? *[enabling style - repair]*
    - When team members are directed, is it always clear why specific actions are necessary? Is the "big picture" communicated? [enabling style - transparency]

- In your perception, *why* does the teamwork in the way that it does? Is it a formal rule, an informal convention, or was it always done this way? Can you provide some examples?
- Are these work practices and ways of interacting similar to other projects that are going on right now? Is it the same as historical ways of doing things? [*Probe on how and why things vary over time and across context*]. Can you provide some examples?
- Has anyone proposed changes to the work practices or ways of interacting employed on this project? If so, why? Can you provide some examples?

Psychological Safety and Trust / Personal Perceptions of Control

- How do you think team members feel in your team? Do they feel free to express unconventional or new ideas/voice concerns/raise tough issues?
- Do you think team members feel valued?
- Do you think it is easy for team members to ask each other for help?
- Do you think team members feel that their mistakes might be held against them?
- Do you think team members trust each other?
- Do you feel controlled? (To what degree would you say that your daily work routines or results are controlled by one or more other members of the team?) [*Perception of control*]
    - If you feel controlled, are you always aware of what is expected of you in terms of results*? [Control Congruence - communicational]*
    - Do you always agree with these controls or do you sometimes think they are inappropriate? [Control *Congruence - evaluational]*
- To what degree would you say that *you* control the activities or outcomes of your colleagues and why? How would you say that you achieve that control? [*Control enactment*]
- Do you think team members have always in mind what is best for the team?
- How are your personal/project outcomes judged? Is this somehow linked to your pay schedule?
- Do you think that flexibility and/or personal discretion are important for the overall outcomes of your team? If yes, how and why? [Team Autonomy and Performance]
- Do you perceive any tension between the need for control and the allowance of flexibility in the team's daily work routines? [Balancing Control]
- Do you think that some control is beneficial to the overall outcomes of your team? If yes, how and why does control improve the team's performance? [Control Performance]

Requirements Changes

- Did you perceive requirements changing during the project? Did the requirements identified at the beginning of the projects differ from those toward the end? In your opinion, what were reasons for it? [*Presence of requirements changes*]
- Changes over time – Did you recognize, that requirements fluctuated quite a bit in early phases of the project?
  (Did you recognize that requirements fluctuated quite a bit in later phases of the project?). Why do you think changes occurred at the beginning of the project? [*dependent on the question beforehand*, "…in the end of the project?"]
- Performance – Do you think that these changing user needs may also result in technical and managerial issues? What are these issues in detail? Can these issues increase the risk for on-time completion or completion of the project within budget?
- During times of high time pressure – Did someone "take the lead" to organize the team or did everyone proceed as usual? Did people change in their behaviour or role enactment?

Diversity and Resilience

- Do you think the team is able to recover quickly (i.e., using little to time, resources, …) from unforeseen crises / events / shocks (e.g., requirement changes)?
- If an unforeseen crisis / event / shock occurs, how does the team react? Do people act differently? Do routines change?
- How do you perceive the diversity of your team – regarding skillsets / regarding gender, ethnicity, culture, …
- Do you think diversity is helpful (or harmful) for your team?
    - During normal operations or
    - During shock or unforeseen crisis?

Appendix D – Complete survey items list (Study V)

Supervisor Questionnaire				
Group	**Subgroup / Construct**	**Item Reference**	**#**	**Question Text**
Requirements Changes	Response Extensiveness	Lee & Xia (2010)	1	To what extent did the team actually incorporate requirements changes in each of the following categories? ... (System) Scope
			2	... (System) Input Data
			3	... (System) Output Data
			4	... Business Rules & Processes
			5	... (Data) Structure
			6	...User Interface
	Response Efficiency	Lee & Xia (2010)	1	How much additional effort was required by the team to incorporate the following changes? (Effort includes time, cost, personnel, and resources.) ...(System) Scope
			2	... (System) Input Data
			3	... (System) Output Data
			4	... Business Rules & Processes
			5	... (Data) Structure
			6	...User Interface
	Adaptiveness	Tiwana 2010	1	Please indicate to what extent the team was able to incorporate new requirements. If no new requirements were incorporated, choose (1). If all new requirements were incorporated, choose (5). The extent to which it was possible to incorporate new requirements and design changes in response to changing client needs during ... ... requirements analysis.
			2	... high-level (conceptual) design.
			3	... detailed (technical) design.
			4	... development *(and coding)*.
Performance and Control 1	Software Functionality	Lee & Xia (2010)	1	The software delivered by the project achieved its functional goals
			2	The software delivered by the project met end-user requirements
			3	The capabilities of the software fit end-user needs
			4	The software met technical requirements
	Performance (Team)	Zellmer-Bruhn & Gibson 2006	1	This team achieves its goals.
			2	This team accomplishes its objectives.
			3	This team meets the requirements set for it.
			4	This team fulfills its mission.
			5	This team serves the purpose it is intended to serve.
Performance and Control 2	Performance (Project) / Alignment	Tiwana & Keil 2010	1	The extent to which the project development process was effective in successfully fulfilling the client's project needs.
			2	The extent to which the project development process was effective in successfully fulfilling the client's project objectives.
			3	The extent to which the project development process was effective in successfully fulfilling the client's project quality expectations.
			4	The extent to which the project development process was effective in successfully fulfilling the client's project functional requirements.
	Requirements Change	Maruping et al. 2009	1	Requirements fluctuated quite a bit in early phases of the project
			2	Requirements fluctuated quite a bit in later phases of the project
			3	Requirements identified at the beginning of the project were quite different from those identified toward the end

	Control Congruence - communicational	self-developed	1	To what extent are you or is your team... ...aware of controls that have been applied to you or your team?
			2	...misinterpreting these controls?
			3	... not recognizing controls, that have been applied to you or your team?
			4	...aware of the objectives of your team?
Team Member Questionnaire				
Group	**Subgroup/ Construct**	**Item Reference**	**#**	**Question Text**
Agile Practices	Burndown Charts	Tripp et al. 2016	1	Our team utilizes visual indicators (charts, graphs, etc.) of how well we are progressing DURING a work cycle.
			2	We use visual tools that allow team members to easily tell if the work is being completed on schedule.
			3	We plot our "work completed" against "work planned" on a chart.
	Stand-up Meetings	Tripp et al. 2016	1	Our team has a short meeting every day to discuss what is going on with the project.
			2	Each day, all team members share with the team what they are working on.
			3	The team discusses issues together daily.
	Retrospective	Tripp et al. 2016	1	On a regular basis, our team reflects on previous work and looks for ways to improve team performance.
			2	At the end of each work cycle, the team asks itself "what went well during the last work cycle".
			3	At the end of each work cycle, the team asks itself "what could be improved during the next cycle".
	Unit Testing	Tripp et al. 2016	1	Our team has a separate set of "test" code that is written specifically to test the "real" code.
			2	Every programmer is responsible for writing unit tests for the code he or she writes.
			3	Programmers are responsible to personally run a set of unit tests until they all run successfully before "checking in" changes.
	Continuous Integration	Tripp et al. 2016	1	Our team integrates code changes as soon as possible.
			2	The team has a process that automatically rebuilds the software several times a day.
			3	The team is automatically notified of any issues related to the automated compiling, deployment, or testing of code
	Automated Build	Tripp et al. 2016	1	Our team uses a script or other code to automatically compile the code.
			2	Our team uses a script or other code to automatically build the software package.
			3	Our team uses a script or other code to generate release notes or other documentation.
	Coding Standards	Tripp et al. 2016	1	The naming and structure of our code is consistent.
			2	Our team uses standards for consistent code formatting.
			3	It is important to the team's success that all of the code be formatted consistently.
	Refactoring	Tripp et al. 2016	1	Whenever our team sees the need, we improve the design of the code we have written previously.
			2	Every member of the team attempts to improve the structure of the code when making a change.
			3	If we find code that is not used, we remove it.
	Pair Programming	Tripp et al. 2016	1	When new software is being developed, two programmers concentrate on the code being written.
			2	We develop our code using pair programming.
			3	Our code is created by two people working together at a single computer.
	Collective Code Ownership	Maruping et al. 2009	1	Anyone on this team can change existing code at any time
			2	*If anyone wants to change a piece of code, they need the permission of the individual(s) that coded it.**
			3	Members of this team feel comfortable changing any part of the existing code at any time.
	User Stories	self-developed	1	Our team creates user stories to keep track of requirements.
			2	To keep an overview of the requirements, the team uses user stories.

			3	User stories help the team to keep track of requirements and to include the customer's voice in the development.
			4	Use stories are used to support the team in reducing complexity of the requirements.
	Code Reviews / Inspections	self-developed	1	Regular code inspections and code reviews help the team to keep the code clean.
			2	To keep the quality of the code high, the team performs regular code reviews or code inspections.
			3	By reviewing and inspecting the code regularly, the team assures a high quality level of the code.
			4	The team utilizes code inspections and code reviews to decrease the number of defects and increase reliability and maintainability of the code.
Requirement Changes	Adaptiveness	Tiwana 2010	1	Please indicate the extent to which the team was able to incorporate new requirements. If no new requirements were incorporated, choose (1). If all new requirements were incorporated, choose (5). The extent to which it was possible to incorporate new requirements and design changes in response to changing client needs during ... ... requirements analysis.
			2	... high-level (conceptual) design.
			3	... detailed (technical) design.
			4	... development and coding
	Requirements Changes	Maruping et al. 2009	1	Please indicate the frequency of the following scenarios or situations on a scale ranging from "never" (1) to "always" (5). Requirements fluctuated quite a bit in early phases of the project
			2	Requirements fluctuated quite a bit in later phases of the project
			3	Requirements identified at the beginning of the project were quite different from those identified toward the end
Safety & Resilience	Psychological Safety	Detert & Edmondson, 2011; Majchrzak & Jarvenpaa, 2010; Pearsall & Ellis, 2011; Schaubroeck et al., 2011; Edmondson, 1999	1	Members of my team are able to bring up problems and tough issues.
			2	It is safe to take a risk in my team.
			3	In my team, my unique skills are valued and utilized.
			4	*It is difficult to ask other members of my team for help.**
			5	*If you make a mistake on this team, it is often held against you.**
			6	*In my team, it is not advisable to bring up problems and tough issues.**
			7	If you make a mistake on this team, it is never held against you.
			8	No one on this team would deliberately act in a way that undermines my efforts.
	Norm Clarity	Lenberg & Feldt (2018)	1	*Standards for member behavior in this team are vague and unclear. **
			2	It is clear what is, and what is not, acceptable member behavior in this team.
			3	Members of this team agree about how members are expected to behave.
	Resilience	Chakravarty et al. (2013)	1	This team can be characterized as resilient.
			2	The team always finds a way to make things work.
			3	The team has the ability to absorb shocks.
			4	The team builds capabilities to defend against a wide range of scenarios.
			5	The team is pliable, in that we can adjust to abnormal conditions and then bounce right back when conditions come back to normal.
Time Pressure & Leadership	Temporal Leadership	Maruping et al. 2015	1	To what extent does the team leader remind members of important deadlines?
			2	To what extent does the team leader prioritize tasks and allocate time to each task?
			3	To what extent does the team leader prepare and build in time for contingencies, problems, and emerging issues?
			4	To what extent does the team leader pace the team so that work is finished on time?

			5	To what extent does the team leader urge members to finish sub-tasks on time?
			6	To what extent does the team leader set milestones to measure progress on the project?
			7	To what extent is the team leader effective in coordinating the team to meet customer deadlines?
	Time Pressure	Maruping et al. 2015	1	To what extent is your team facing lot of pressure to complete tasks on time?
			2	To what extent is your team not being afforded much time to complete your tasks?
			3	To what extent is your team provided short amount of time to complete your tasks?
			4	To what extent is your team dealing with short task durations?
	Control Congruence - communicational	self-developed	1	To what extent are you or is your team aware of controls that have been applied to you or your team?
			2	*To what extent are you or is your team misinterpreting these controls? **
			3	*To what extent are you or is your team not recognizing controls, that have been applied to you or your team? **
			4	To what extent are you or is your team aware of the objectives of your team?
	Control Congruence - evaluational	self-developed	1	To what extent did you or your team agree to controls, that have been applied to you or your team?
			2	*To what extent did you or your team find these controls inappropriate, that have been applied to you or your team? **
			3	*To what extent did you or your team completely disagree to controls, that have been applied to you or your team? **
			4	To what extent did you or your team know about the rationale of the applied controls?
Autonomy & Control	Team Autonomy	Lee & Xia (2010)	1	The project team was allowed to freely choose tools and technologies.
			2	The project team had control over what they were supposed to accomplish.
			3	The project team was granted autonomy on how to handle user requirements changes.
			4	The project team was free to assign personnel to the project.
	Control Style "Enabling" - Repair feature	Remus 2016	1	My team and I are able to identify a well operating development process.
			2	My team and I are able to identify opportunities to improve the development process.
			3	My team and I are allowed to deviate from defined procedures.
			4	My team and I are allowed to fix problems in the development process.
			5	Superiors of my team appreciate feedback to real work contingencies.
	Control Style "Enabling" - Transparency feature	Remus 2016	1	*The development procedures are communicated as lists of flat assertions of duties. **
			2	My team and I have insights into development processes by getting information about their key components and by having information about best practices.
			3	My team and I are expected to merely implement the communicated work instructions.
			4	My team and I are provided with an understanding of the rationale behind the development processes by the superior.
			5	My team and I get regular feedback about my performance.
			6	My team and I are aware of how my own tasks fit into the entire work product.
			7	The contextual information I have access to enables me to interact creatively with the broader project organization and its environment.
			8	My team and I are regularly informed about other project contexts in order to interact creatively with my organization and environment.
	Control Style "Authoritative"	self-developed	1	My team and I are not allowed to deviate from defined procedures which my superiors established.
			2	My team and I are not aware of the rationale behind certain development processes.
			3	My team and I do not get feedback about the performance during the development process.
			4	My team and I are enforced to comply with the rules and processes of the superiors in case of deviations from these regulations.

			6	My team and I are given less information about other project activities or contexts.
			7	My team and I follow procedures that are formulated as lists of flat assertions of duties.
	Input Control	Remus 2016 (Based on Yu and To 2011 and Snell 1992)	1	Other team members and I consist of professionals out of different divisions.
			2	My superiors encourage employees to further enhance their capabilities.
			3	My superiors select team members not only by professional competence but also by personality and personal values.
			4	My superiors emphasize the internalization of the goals, values, and norms of the organization.
			5	Other team members and I get rewarded based on the level of individual skills.
	Outcome Control	Maruping et al. 2009	1	The performance of the team will be evaluated by the extent to which project goals have been accomplished, regardless of how the goals were accomplished.
			2	Project goals are outlined at the beginning of a project.
			3	Significant weight will be placed upon timely project completion.
			4	Significant weight will be placed upon project quality.
			5	Significant weight will be placed upon project completion to meet client requirements.
			6	Pre-established targets are used as benchmarks for the team's performance evaluations.
	Behavior Control	Tiwana & Keil 2010; Kirsch 2002	1	My team is expected to follow an written sequence of steps regarding the accomplishment of project goals.
			2	My team is expected to follow an written sequence of steps to ensure the system meets user department requirements.
			3	My team is expected to follow an written sequence of steps to ensure the success of this project.
			4	My team is assessed on the extent to which we followed existing written procedures and practices during the development process.
			1	The client expected me to follow an written sequence of steps toward the accomplishment of project goals.
			2	The client assessed the extent to which existing written procedures and practices were followed during the development process.
	Clan Control	Kirsch 2002	1	Members of my team and I actively participated in project meetings to understand the project team's goals, values, and norms.
			2	Members of my team and I placed a significant weight on understanding the project team's goals, values, and norms.
			3	Members of my team and I attempted to be a "regular" member of the project team.
			4	Members of my team and I attempted to understand the project team's goals, norms, and values.
	Self Control	Kirsch 2002; Maruping et al. 2009; Remus 2016	1	Tangible rewards given to the team are (or will be) dependent on whether individuals on the teamwork on their own, without much direction from others.
			2	Individuals on this team are rewarded for their individual performance.
			3	Individual task performance is rewarded on this team.
			1	I set specific goals for myself on the project without the involvement of others
			2	I defined specific procedures for this project's activities without the involvement of others.

REFERENCES

Abrahamsson, P., Conboy, K., and Wang, X. 2009. "'Lots Done, More to Do': The Current State of Agile Systems Development Research," *European Journal of Information Systems* (18), pp. 281-284.

Adler, P.S., and Borys, B. 1996. "Two Types of Bureaucracy: Enabling and Coercive," *Administrative Science Quarterly* (41:1), p. 61.

Ågerfalk, P.J., and Fitzgerald, B. 2006a. "Flexible and Distributed Software Processes: Old Petunias in New Bowls," in: *Communications of the ACM*. Citeseer, pp. 27-34.

Ågerfalk, P.J., and Fitzgerald, B. 2006b. "Understanding Flexible and Distributed Software Development Processes," in: *Proc. Thirty-Ninth Hawaii Internat. Conf. System Sciences*. IEEE Computer Society Press Washington, DC, p. 1.

Ågerfalk, P.J., Fitzgerald, B., and Slaughter, S.A. 2009. "Flexible and Distributed Information Systems Development: State of the Art and Research Challenges," *Information Systems Research* (20:3), pp. 317-329.

Aggarwal, C.C., and Zhai, C. 2012. *Mining Text Data*. Springer Science & Business Media.

Alghamdi, R., and Alfalqi, K. 2015. "A Survey of Topic Modeling in Text Mining," *International Journal of Advanced Computer Science and Applications (IJACSA)* (6:1), pp. 147-153.

Alter, S. 2008. "Defining Information Systems as Work Systems: Implications for the Is Field," *European Journal of Information Systems* (17:5), pp. 448-469.

Alzoubi, Y.I., Gill, A.Q., and Al-Ani, A. 2016. "Empirical Studies of Geographically Distributed Agile Development Communication Challenges: A Systematic Review," *Information & Management* (53:1), pp. 22-37.

Andrei, B.-A., Casu-Pop, A.-C., Gheorghe, S.-C., and Boiangiu, C.-A. 2019. "A Study on Using Waterfall and Agile Methods in Software Project Management," *Journal of Information Systems & Operations Management*), pp. 125-135.

Arun, R., Suresh, V., Madhavan, C.V., and Murthy, M.N. 2010. "On Finding the Natural Number of Topics with Latent Dirichlet Allocation: Some Observations," in: *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, pp. 391-402.

Austin, R.D., and Devin, L. 2009. "Weighing the Benefits and Costs of Flexibility in Making Software: Toward a Contingency Theory of the Determinants of Development Process Design," *Information Systems Research* (20:3), pp. 462-478.

Babb, J., Hoda, R., and Norbjerg, J. 2014a. "Embedding Reflection and Learning into Agile Software Development," *IEEE Software* (31:4).

Babb, J., Hoda, R., and Nørbjerg, J. 2014b. "Embedding Reflection and Learning into Agile Software Development," *IEEE Software* (31:4), pp. 51-57.

Balijepally, V. 2005. "Collaborative Software Development in Agile Methodologies - Perspectives from Small Group Research," in: *AMCIS 2005 Proceedings*.

Balijepally, V., Mahapatra, R., Nerur, S., and Price, K.H. 2009. "Are Two Heads Better Than One for Software Development? The Productivity Paradox of Pair Programming," *MIS Quarterly* (33:1), pp. 91-119.

Barker, J.R. 1993. "Tightening the Iron Cage: Concertive Control in Self-Managing Teams," *Administrative Science Quarterly* (38:3), pp. 408-437.

Baskerville, R., Pries-Heje, J., and Madsen, S. 2011. "Post-Agility: What Follows a Decade of Agility?," *Information and Software Technology* (53:5).

Batra, D., Xia, W., and Zhang, M. 2017. "Collaboration in Agile Software Development: Concept and Dimensions," *Communications of the Association for Information Systems* (41:1), p. 20.

Beck, K. 1999. *Extreme Programming Explained : Embrace Change*. Harlow: Addison-Wesley.

Beck, K., and Andres, C. 2004. *Extreme Programming Explained: Embrace Change*, (2. ed.). Addison-Wesley Professional.

Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R.C., Mellor, S., Schwaber, K., Sutherland, J., and Thomas, D. 2001a. "Manifesto for Agile Software Development." Retrieved 11.11.2015, 2015, from http://www.agilemanifesto.org

Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R.C., Mellor, S., Schwaber, K., Sutherland, J., and Thomas, D. 2001b. "Manifesto for Agile Software Development." *Manifesto for Agile Software Development* Retrieved February 24, 2016, from http://www.agilemanifesto.org/

Benbasat, I., Goldstein, D.K., and Mead, M. 1987. "The Case Research Strategy in Studies of Information Systems," *MIS quarterly*), pp. 369-386.

Berente, N., Hansen, S.W., and Rosenkranz, C. 2015. "Rule Formation and Change in Information Systems Development: How Institutional Logics Shape Isd Practices and Processes," *System Sciences (HICSS), 2015 48th Hawaii International Conference on*: IEEE, pp. 5104-5113.

Blei, D.M., Ng, A.Y., and Jordan, M.I. 2003. "Latent Dirichlet Allocation," *Journal of machine Learning research* (3:Jan), pp. 993-1022.

Boehm, B.W. 1988. "A Spiral Model of Software Development and Enhancement," *IEEE Computer* (21:4), pp. 61-72.

Boehm, B.W. 1991. "Software Risk Management: Principles and Practices," *IEEE Software* (8:1).

Bohn, J. 2018. "Digitally-Driven Change in the Insurance Industry--Disruption or Transformation?," *Journal of Financial Transformation* (48), pp. 76-87.

Bowen, S., and Maurer, F. 2002. "Process Support and Knowledge Management for Virtual Teams Doing Agile Software Development," in: *Proceedings 26th Annual International Computer Software and Applications*. pp. 1118-1120.

Brooks, F., and Kugler, H. 1987. *No Silver Bullet*. April.

Burns, T., and Stalker, G.M. 1961. "The Management of Innovation. London," *Tavistock Publishing. Cited in Hurley, RF and Hult, GTM (1998). Innovation, Market Orientation, and Organisational Learning: An Integration and Empirical Examination. Journal of Marketing* (62), pp. 42-54.

Campanelli, A.S., and Parreiras, F.S. 2015a. "Agile Methods Tailoring - a Systematic Literature Review," *The Journal of Systems and Software* (110), p. 85.

Campanelli, A.S., and Parreiras, F.S. 2015b. "Agile Methods Tailoring - a Systematic Literature Review," *The Journal of Systems and Software* (110), pp. 85-100.

Cao, J., Xia, T., Li, J., Zhang, Y., and Tang, S. 2009a. "A Density-Based Method for Adaptive Lda Model Selection," *Neurocomputing* (72:7-9), pp. 1775-1781.

Cao, L., Mohan, K., Ramesh, B., and Sarkar, S. 2013. "Adapting Funding Processes for Agile It Projects: An Empirical Investigation," *European Journal of Information Systems* (22:2), pp. 191-205.

Cao, L., Mohan, K., Xu, P., and Ramesh, B. 2009b. "A Framework for Adapting Agile Development Methodologies," *European Journal of Information Systems* (18:4), pp. 332-343.

Cappiello, A. 2018. *Technology and the Insurance Industry: Re-Configuring the Competitive Landscape*. Springer.

Chen, C., Huang, E., and Yan, H. 2016. "Detecting the Association of Health Problems in Consumer-Level Medical Text," *Journal of Information Science* (44:1), pp. 3-14.

Choudhury, V., and Sabherwal, R. 2003. "Portfolios of Control in Outsourced Software Development Projects," *Information Systems Research* (14:3).

Chow, T., and Cao, D.-B. 2008. "A Survey Study of Critical Success Factors in Agile Software Projects," *The Journal of Systems and Software* (81:6).

Chua, C.E.H., Lim, W.-K., Soh, C., and Sia, S.K. 2012. "Enacting Clan Control in Complex Projects: A Social Capital Perspective," *MIS Quarterly* (36:2), pp. 577-601.

Chua, C.E.H., and Myers, M.D. 2018. "Social Control in Information Systems Development: A Negotiated Order Perspective," *Journal of Information Technology* (33:3), pp. 173-187.

Cockburn, A., Highsmith, J., and Boehm, B. 2001. "Agile Software Development: The People Factor," *Computer* (34:11), pp. 131-134.

Cohen, D., Lindvall, M., and Costa, P. 2004. "An Introduction to Agile Methods," *Advances in computers* (62:03), pp. 1-66.

Cohen, J. 1960. "A Coefficient of Agreement for Nominal Scales," *Educational and psychological measurement* (20:1), pp. 37-46.

Cohen, J. 1988. *Statistical Power Analysis for the Behavioral Sciences*. Hillsdale, N.J.: L. Erlbaum Associates.

Cohn, M. 2010. *Succeeding with Agile : Software Development Using Scrum*. Upper Saddle River, NJ: Addison-Wesley.

Cole, M.S., Carter, M.Z., and Zhang, Z. 2013. "Leader-Team Congruence in Power Distance Values and Team Effectiveness: The Mediating Role of Procedural Justice Climate," *Journal of Applied Psychology* (98:6), pp. 962-973.

Collier, P.M. 2005. "Entrepreneurial Control and the Construction of a Relevant Accounting," *Management Accounting Research* (16:3), pp. 321-339.

Conboy, K. 2009. "Agility from First Principles: Reconstructing the Concept of Agility in Information Systems Development," *Information Systems Research* (20:3), pp. 329-355.

Conboy, K., and Carroll, N. 2019. "Implementing Large-Scale Agile Frameworks: Challenges and Recommendations," *IEEE Software* (36:2), pp. 44-50.

Conboy, K., Coyle, S., Xiaofeng, W., and Pikkarainen, M. 2011. "People over Process: Key Challenges in Agile Development," *IEEE Software* (28:4), pp. 48-48.

Conboy, K., and Fitzgerald, B. 2007. "Agile Drivers, Capabilities, and Value: An over-Arching Assessment Framework for Systems Development," in *Agile Information Systems: Conceptualization, Construction, and Management,* K.C. Desouza (ed.). Massachusetts, USA: Elsevier, pp. 207-221.

Cordery, J.L., Morrison, D., Wright, B.M., and Wall, T.D. 2010. "The Impact of Autonomy and Task Uncertainty on Team Performance: A Longitudinal Field Study," *Journal of organizational behavior* (31:2-3), pp. 240-258.

Cordery, J.L., Mueller, W.S., and Smith, L.M. 1991. "Attitudinal and Behavioral Effects of Autonomous Group Working: A Longitudinal Field Study," *Academy of Management Journal* (34:2), pp. 464-476.

Cordery, J.L., and Tian, A.W. 2017. "Team Design," in *The Wiley Blackwell Handbook of the Psychology of Team Working and Collaborative Processes*. pp. 103-128.

Cram, W.A., Brohman, K., and Gallupe, R.B. 2016a. "Information Systems Control: A Review and Framework for Emerging Information Systems Processes," *Journal of the Association for Information Systems* (17:4), p. 216.

Cram, W.A., and Brohman, M.K. 2010. "Beyond Modes: A New Typology of Isd Control," *Internation Conference on Information Systems (ICIS)*, St. Louis, Missouri, USA.

Cram, W.A., and Brohman, M.K. 2013. "Controlling Information Systems Development: A New Typology for an Evolving Field," *Information Systems Journal* (23:2).

Cram, W.A., Brohman, M.K., Chan, Y.E., and Gallupe, R.B. 2016b. "Information Systems Control Alignment: Complementary and Conflicting Systems Development Controls," *Information Management* (53:2), pp. 183-196.

Cram, W.A., and Wiener, M. 2018. "Perceptions of Control Legitimacy in Information Systems Development,").

Creswell, J.W. 2009. *Research Design: Qualitative, Quantitative, and Mixed Methods Approaches*. SAGE Publications, Incorporated.

Cronbach, L.J. 1951. "Coefficient Alpha and the Internal Structure of Tests," *psychometrika* (16:3), pp. 297-334.

Dalton, M. 1959. "Men Who Manage,").

David Gefen, C.M.R. 2002. "Implementation Team Responsiveness and User Evaluation of Customer Relationship Management: A Quasi-Experimental Design Study of Social Exchange Theory," *Journal of management information systems* (19:1), pp. 47-69.

Davis, G.B. 2000. "Information Systems Conceptual Foundations: Looking Backward and Forward," in *Organizational and Social Perspectives on Information Technology*. Springer, pp. 61-82.

Debortoli, S., Müller, O., Junglas, I., and vom Brocke, J. 2016a. "Text Mining for Information Systems Researchers: An Annotated Topic Modeling Tutorial," *Communications of the AIS*).

Debortoli, S., Müller, O., Junglas, I., and vom Brocke, J. 2016b. "Text Mining for Information Systems Researchers: An Annotated Topic Modeling Tutorial," *Communications of the Association for Information Systems (CAIS)* (39:7), pp. 110-135.

DeMarco, T., and Lister, T. 1987. *Peopleware: Productive Projects and Teams*. Dorset House Publishing Co., Inc. New York, NY, USA.

Diegmann, P., Dreesen, T., Binzer, B., and Rosenkranz, C. 2018. "Journey Towards Agility: Three Decades of Research on Agile Information Systems Development," *Internation Conference on Information Systems (ICIS)*, San Francisco, CA, USA.

Dijkstra, T.K., and Henseler, J. 2015. "Consistent Partial Least Squares Path Modeling," *MIS Quarterly* (39:2), pp. 297-316.

Dikert, K., Paasivaara, M., and Lassenius, C. 2016. "Challenges and Success Factors for Large-Scale Agile Transformations: A Systematic Literature Review," *Journal of Systems and Software* (119), pp. 87-108.

Dingsoeyr, T., Falessi, D., and Power, K. 2019. "Agile Development at Scale: The Next Frontier," *IEEE Software* (36:2), pp. 30-38.

Dingsøyr, T., Nerur, S., Balijepally, V., and Moe, N.B. 2012. "A Decade of Agile Methodologies: Towards Explaining Agile Software Development," *Journal of Systems and Software* (85:6), pp. 1213-1221.

Domberger, S. 1998. *The Contracting Organization: A Strategic Guide to Outsourcing: A Strategic Guide to Outsourcing*. OUP Oxford.

Domino, M., Hevner, A., and Collins, R.W. 2002. "Applying Agile Software Development Processes to Global Virtual Teams: A Study of Communication Modalities," in: *Proceedings of the 2002 ACM SIGCPR conference on Computer personnel research*. Kristiansand, Norway: ACM, pp. 76-78.

Dreesen, T., Diegmann, P., Binzer, B., and Rosenkranz, C. 2019. "Journey Towards Agility–a Retro- and Prospective Review," *Proceedings of the 52nd Hawaii International Conference on System Sciences*, Maui, HI, USA.

Dreesen, T., Diegmann, P., and Rosenkranz, C. 2020. "The Impact of Modes, Styles, and Congruence of Control on Agile Teams: Insights from a Multiple Case Study," *53rd Hawaii International Conference on System Sciences (HICSS)*, Maui, HI, USA.

Dreesen, T., and Schmid, T. 2018. "Do as You Want or Do as You Are Told? Control Vs. Autonomy in Agile Software Development Teams," *Proceedings of the 51st Hawaii International Conference on System Sciences*.

Druskat, V.U., and Wheeler, J.V. 2003. "Managing from the Boundary: The Effective Leadership of Self-Managing Work Teams," *Academy of Management Journal* (46:4), pp. 435-457.

Dubé, L., and Paré, G. 2003. "Rigor in Information Systems Positivist Case Research: Current Practices, Trends, and Recommendations," *Management Information Systems Quarterly* (27:4).

Dubois, A., and Gadde, L.-E. 2002. "Systematic Combining: An Abductive Approach to Case Research," *Journal of business research* (55:7), pp. 553-560.

Dunn, T., Baguley, T., and Brunsden, V. 2013. "From Alpha to Omega: A Practical Solution to the Pervasive Problem of Internal Consistency Estimation," *British Journal of Psychology* (105).

Dybå, T., and Dingsøyr, T. 2008. "Empirical Studies of Agile Software Development: A Systematic Review," *Information and software technology* (50:9), pp. 833-859.

Edwards, H.K., and Sridhar, V. 2003. "Analysis of the Effectiveness of Global Virtual Teams in Software Engineering Projects," in: *HICSS 2003*. IEEE.

Eisenhardt, K.M. 1985. "Control: Oorganizational and Economic Approaches," *Management Science* (31:2), pp. 134-150.

Eisenhardt, K.M. 1989. "Agency Theory: An Assessment and Review," *Academy of management review* (14:1), pp. 57-74.

Estevam, A., Dennehy, D., and Conboy, K. 2020. "Using Flow Tools to Enact Control in Software Development Projects: A Cross-Case Analysis," *Information Systems Frontiers*).

Esteves, J., and Pastor, J. 2004. "Using a Multimethod Approach to Research Enterprise Systems Implementations," *Electronic Journal of Business Research Methods* (2:2), pp. 69-82.

Falkenberg, E., Hesse, W., Lindgreen, P., Nilsson, B., Oei, H., Rolland, C., Stamper, R., Van Assche, F., Verrijn-Stuart, A., and Voss, K. 1998. "A Framework of Information System Concepts," International Federation for Information Processing.

Feldman, S.P. 1989. "The Broken Wheel: The Inseparability of Autonomy and Control in Innovation within Organizations," *Journal of Management Studies* (26:2), pp. 83-102.

Fitzgerald, B. 1997. "The Use of Systems Development Methodologies in Practice: A Field Study," *Information Systems Journal* (7:3), pp. 201-212.

Fitzgerald, B., Hartnett, G., and Conboy, K. 2006a. "Customising Agile Methods to Software Practices at Intel Shannon," *European Journal of Information Systems* (15:2), pp. 200-213.

Fitzgerald, B., Hartnett, G., and Conboy, K. 2006b. "Customising Agile Methods to Software Practices at Intel Shannon," *European Journal of Information Systems* (15:2).

Fitzgerald, B., Slaughter, S.A., and Ågerfalk, P.J. 2006c. "Call for Papers - Information Systems Research Special Issue: Flexible and Distributed Information Systems Development," *Information Systems Research* (17:3), pp. 323-325.

Flamholtz, E.G., Das, T., and Tsui, A.S. 1985. "Toward an Integrative Framework of Organizational Control," *Accounting, organizations and society* (10:1), pp. 35-50.

Fornell, C., and Larcker, D.F. 1981. "Structural Equation Models with Unobservable Variables and Measurement Error: Algebra and Statistics," *Journal of Marketing Research* (18:3), pp. 382-388.

Fraser, S., and Mancl, D. 2008. "No Silver Bullet: Software Engineering Reloaded," *IEEE Software* (25:1), pp. 91-94.

Fraser, S.D., Brooks, F.P., Fowler, M., Lopez, R., Namioka, A., Northrop, L., Parnas, D.L., and Thomas, D. 2007. ""No Silver Bullet" Reloaded: Retrospective on "Essence and Accidents of Software Engineering"," in: *Companion to the 22nd ACM SIGPLAN conference on Object-oriented programming systems and applications companion*. Montreal, Quebec, Canada: ACM.

Freudenberg, S., and Sharp, H. 2010. "The Top 10 Burning Research Questions from Practitioners," *Ieee Software* (27:5), pp. 8-9.

Fritz, C.O., Morris, P.E., and Richler, J.J. 2012. "Effect Size Estimates: Current Use, Calculations, and Interpretation," *Journal of experimental psychology: General* (141:1), p. 2.

Fruhling, A., and de Vreede, G.-J. 2006. "Field Experiences with Extreme Programming: Developing an Emergency Response System," *Journal of Management Information Systems* (22:4), pp. 39-69.

Garvin, D.A., Wagonfeld, A.B., and Kind, L. 2013. "Google's Project Oxygen: Do Managers Matter?,").

Gefen, D., and Keil, M. 1998. "The Impact of Developer Responsiveness on Perceptions of Usefulness and Ease of Use: An Extension of the Technology Acceptance Model," *ACM SIGMIS Database: the DATABASE for Advances in Information Systems* (29:2), pp. 35-49.

Geiger, D., Rosemann, M., Fielt, E., and Schader, M. 2012. *Crowdsourcing Information Systems-Definition Typology, and Design*. AISeL.

Gerwin, D., and Moffat, L. 1997. "Withdrawal of Team Autonomy During Concurrent Engineering," *Management Science* (43:9), pp. 1275-1287.

Gibson, C.B., and Birkinshaw, J. 2004. "The Antecedents, Consequences, and Mediating Role of Organizational Ambidexterity," *Academy of Management Journal* (47:2), pp. 209-227.

Goh, J.C.-L., Pan, S.L., and Zuo, M. 2013. "Developing the Agile Is Development Practices in Large-Scale It Projects: The Trust-Mediated Organizational Controls and It Project Team Capabilities Perspectives," *Journal of the Association for Information Systems* (14:12).

Gomber, P., Kauffman, R.J., Parker, C., and Weber, B.W. 2018. "On the Fintech Revolution: Interpreting the Forces of Innovation, Disruption, and Transformation in Financial Services," *Journal of Management Information Systems* (35:1), pp. 220-265.

Gopal, A., and Gosain, S. 2010. "The Role of Organizational Controls and Boundary Spanning in Software Development Outsourcing: Implications for Project Performance," *Information Systems Research* (21:4), pp. 960-983.

Grady, R.B. 1993. "Practical Results from Measuring Software Quality," *Communications of the ACM* (36:11), pp. 62-68.

Gregory, R.W., Beck, R., and Keil, M. 2013a. "Control Balancing in Information Systems Development Offshoring Projects," *MIS Quarterly* (37:4), pp. 1211-1237.

Gregory, R.W., and Keil, M. 2014. "Blending Bureaucratic and Collaborative Management Styles to Achieve Control Ambidexterity in Is Projects," *European Journal of Information Systems* (23:3), pp. 343-356.

Gregory, T., Mathiassen, L., and Sambhara, C. 2013b. "Chains of Control in Agile Software Development," in: *AMCIS 2013 Proceedings*.

Griffiths, T.L., and Steyvers, M. 2004. "Finding Scientific Topics," *Proceedings of the National academy of Sciences* (101:suppl 1), pp. 5228-5235.

Guinea, A.S., Nain, G., and Le Traon, Y. 2016. "A Systematic Review on the Engineering of Software for Ubiquitous Systems," *The Journal of Systems and Software* (118), pp. 251-276.

Haas, M.R. 2010. "The Double-Edged Swords of Autonomy and External Knowledge: Analyzing Team Effectiveness in a Multinational Organization," *Academy of Management Journal* (53:5), pp. 989-1008.

Hackman, J.R. 1987. "The Design of Work Teams," in *Handbook of Organizational Behavior,* J.W. Lorsch (ed.). Englewood Cliffs, NJ: Prentice-Hall, pp. 315-342.

Hair, J.F., Hult, G.T.M., Ringle, C.M., and Sarstedt, M. 2017. *A Primer on Partial Least Squares Structural Equation Modeling (Pls-Sem)*.

Hansen, H.R., Mendling, J., and Neumann, G. 2019. *Wirtschaftsinformatik*. De Gruyter Oldenbourg.

Harris, M.L., Collins, R.W., and Hevner, A.R. 2009a. "Control of Flexible Software Development under Uncertainty," *Information Systems Research* (20:3), pp. 400-420.

Harris, M.L., Hevner, A.R., and Collins, R.W. 2009b. "Controls in Flexible Software Development," *Communications of the Association for Information Systems* (24:1).

Hart, C. 1998. "Doing a Literature Review. Releasing the Social Science Research Imagination,").

Henderson, J.C., and Lee, S. 1992. "Managing I/S Design Teams: A Control Theories Perspective," *Management Science* (38:6), pp. 757-777.

Hennel, P., and Rosenkranz, C. 2021. "Investigating the "Socio" in Socio-Technical Development: The Case for Psychological Safety in Agile Information Systems Development," *Project Management Journal* (52:1), pp. 11-30.

Heumann, J., Wiener, M., and Remus, U. 2012. "Power Distance in Information Systems Offshoring Projects - a Control Theory Perspective," in: *ICIS 2012 Proceedings*.

Heumann, J., Wiener, M., Remus, U., and Mähring, M. 2015. "To Coerce or to Enable? Exercising Formal Control in a Large Information Systems Project," *Journal of Information Technology* (30:4), pp. 337-351.

Hevner, A.R., March, S.T., Park, J., and Ram, S. 2004. "Design Science in Information Systems Research," *MIS Quarterly* (28:1), pp. 75-105.

Highsmith, J., and Cockburn, A. 2001. "Agile Software Development: The Business of Innovation," *Computer* (34:9), pp. 120-127.

Highsmith, J., Cockburn, A., and Boehm, B. 2001. "Agile Software Development: The Business of Innovation," *Computer* (34:9), pp. 120-123.

Hirst, G., Van Knippenberg, D., Chen, C.-h., and Sacramento, C.A. 2011. "How Does Bureaucracy Impact Individual Creativity? A Cross-Level Investigation of Team Contextual Influences on Goal Orientation–Creativity Relationships," *Academy of Management Journal* (54:3), pp. 624-641.

Hobbs, B., and Petit, Y. 2017. "Agile Methods on Large Projects in Large Organizations," *Project Management Journal* (48:3), pp. 3-19.

Hoda, R., Noble, J., and Marshall, S. 2012. "Developing a Grounded Theory to Explain the Practices of Self-Organizing Agile Teams," *Empirical Software Engineering* (17:6), pp. 609-639.

Hoda, R., Noble, J., and Marshall, S. 2013. "Self-Organizing Roles on Agile Software Development Teams," *IEEE Transactions on Software Engineering* (39:3), pp. 422-445.

Hollenbeck, J.R., Ellis, A.P., Humphrey, S.E., Garza, A.S., and Ilgen, D.R. 2011. "Asymmetry in Structural Adaptation: The Differential Impact of Centralizing Versus Decentralizing Team Decision-Making Structures," *Organizational Behavior and Human Decision Processes* (114:1), pp. 64-74.

Hu, L.t., and Bentler, P.M. 1999. "Cutoff Criteria for Fit Indexes in Covariance Structure Analysis: Conventional Criteria Versus New Alternatives," *Structural Equation Modeling: A Multidisciplinary Journal* (6:1), pp. 1-55.

Huisman, M., and Iivari, J. 2006. "Deployment of Systems Development Methodologies: Perceptual Congruence between Is Managers and Systems Developers," *Information & Management* (43:1), pp. 29-49.

Hummel, M., Rosenkranz, C., and Holten, R. 2013. "The Role of Communication in Agile Systems Development," *BUSINESS AND INFORMATION SYSTEMS ENGINEERING* (5:5), pp. 343-355.

Hummel, M., Rosenkranz, C., and Holten, R. 2015. "The Role of Social Agile Practices for Direct and Indirect Communication in Information Systems Development Teams," *Communications of the Association for Information Systems* (36).

Hunter, M.G. 2004. *Qualitative Research in Information Systems: An Exploration of Methods in the Handbook of Information Systems Research*. Hershey et al. : Idea Group.

Hwang, H. 2009. "Regularized Generalized Structured Component Analysis," *Psychometrika* (74:3), pp. 517-530.

Hyatt, D.E., and Ruddy, T.M. 1997. "An Examination of the Relationship between Work Group Characteristics and Performance: Once More into the Breech," *Personnel Psychology* (50:3), pp. 553-585.

Inayat, I., Salim, S.S., Marczak, S., Daneva, M., and Shamshirband, S. 2015. "A Systematic Literature Review on Agile Requirements Engineering Practices and Challenges," *Computers in human behavior* (51), pp. 915-929.

Jaeger, A.M., and Baliga, B.R. 1985. "Control Systems and Strategic Adaptation: Lessons from the Japanese Experience," *Strategic Management Journal* (6:2), pp. 115-134.

Jaworski, B.J. 1988. "Toward a Theory of Marketing Control: Environmental Context, Control Types, and Consequences," *Journal of Marketing* (52:3), pp. 23-40.

Jöreskog, K.G. 1967. "Some Contributions to Maximum Likelihood Factor Analysis," *Psychometrika* (32:4), pp. 443-482.

Jorgensen, M. 2019. "Relationships between Project Size, Agile Practices, and Successful Software Development: Results and Analysis," *IEEE Software* (36:2), pp. 39-43.

Kaplan, B., and Duchon, D. 1988. "Combining Qualitative and Quantitative Methods in Information Systems Research: A Case Study," *MIS Quarterly* (12:4), pp. 571-587.

Karlsson, F., and Ågerfalk, P. 2009. "Exploring Agile Values in Method Configuration," *European Journal of Information Systems* (18:4), pp. 300-316.

Karlström, D., and Runeson, P. 2005. "Combining Agile Methods with Stage-Gate Project Management," *IEEE Software* (22:3), pp. 43-50.

Keil, M., Rai, A., and Liu, S. 2013. "How User Risk and Requirements Risk Moderate the Effects of Formal and Informal Control on the Process Performance of It Projects," *European Journal of Information Systems* (22:6), pp. 650-672.

Kenny, D.A., Kaniskan, B., and McCoach, D.B. 2014. "The Performance of Rmsea in Models with Small Degrees of Freedom," *Sociological Methods & Research* (44:3), pp. 486-507.

Kirkman, B.L., Chen, G., Farh, J.-L., Chen, Z.X., and Lowe, K.B. 2009. "Individual Power Distance Orientation and Follower Reactions to Transformational Leaders: A Cross-Level, Cross-Cultural Examination," *Academy of management journal* (52:4), pp. 744-764.

Kirkman, B.L., and Rosen, B. 1999. "Beyond Self-Management: Antecedents and Consequences of Team Empowerment," *Academy of Management Journal* (42:1), pp. 58-74.

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

Kirsch, L.J. 1997. "Portfolios of Control Modes and Is Project Management," *Information Systems Research* (8:3), pp. 215-240.

Kirsch, L.J. 2004. "Deploying Common Systems Globally: The Dynamics of Control," *Information Systems Research* (15:4), pp. 374-396.

Kirsch, L.J., Sambamurthy, V., Dong-Gil, K., and Purvis, R.L. 2002. "Controlling Information Systems Development Projects: The View from the Client," *Management Science* (48:4), pp. 484-499.

Ko, D.-G., Kirsch, L.J., and King, W.R. 2005. "Antecedents of Knowledge Transfer from Consultants to Clients in Enterprise System Implementations," *MIS Quarterly* (29:1), pp. 59-85.

Krathwohl, D.R. 1998. *Methods of Educational and Social Science Research: An Integrated Approach*. New York, NY, USA: Longman.

Langfred, C.W. 2000. "The Paradox of Self-Management: Individual and Group Autonomy in Work Groups," *Journal of Organizational Behavior* (21:5), pp. 563-585.

Langfred, C.W. 2004. "Too Much of a Good Thing? Negative Effects of High Trust and Individual Autonomy in Self-Managing Teams," *Academy of management journal* (47:3), pp. 385-399.

Langfred, C.W. 2007. "The Downside of Self-Management: A Longitudinal Study of the Effects Tf Conflict on Trust, Autonomy, and Task Interdependence in Self-Managing Teams," *Academy of Management Journal* (50:4), pp. 885-900.

Larman, C. 2003a. *Agile and Iterative Development: A Manager's Guide*. Pearson Education.

Larman, C. 2003b. *Agile and Iterative Development: A Manager's Guide*. Pearson Education.

Lee, A.S. 1989. "Case Studies as Natural Experiments," *Human Relations* (42:2), pp. 117-137.

Lee, A.S. 1991. "Integrating Positivist and Interpretive Approaches to Organizational Research," *Organization Science* (2:4), pp. 342-365.

Lee, G., and Xia, W. 2005. "The Ability of Information Systems Development Project Teams to Respond to Business and Technology Changes: A Study of Flexibility Measures," *European Journal of Information Systems* (14:1).

Lee, G., and Xia, W. 2010a. "Toward Agile: An Integrated Analysis of Qualitative and Quantitative Field Data on Software Development Agility," *MIS Quarterly* (34:1), pp. 87-115.

Lee, G., and Xia, W. 2010b. "Toward Agile: An Integrated Analysis of Quantitative and Qualitative Field Data on Software Development Agility," *MIS Quarterly* (34:1), pp. 87-115.

Lee, J.-N., Miranda, S.M., and Kim, Y.-M. 2004. "It Outsourcing Strategies: Universalistic, Contingency, and Configurational Explanations of Success," *Information Systems Research* (15:2), pp. 110-131.

Leffingwell, D. 2020. "Scaled Agile Framework® (Safe) 5.0." from https://www.scaledagileframework.com/

Levy, P.F. 2001. "The Nut Island Effect. When Good Teams Go Wrong," *Harvard business review* (79:3), pp. 51-59, 163.

Levy, Y., and Ellis, T.J. 2006. "A Systems Approach to Conduct an Effective Literature Review in Support of Information Systems Research," *Informing Science* (9), pp. 181-212.

Lindsjørn, Y., Sjøberg, D.I.K., Dingsøyr, T., Bergersen, G.R., and Dybå, T. 2016. "Teamwork Quality and Project Success in Software Development: A Survey of Agile Development Teams," *Journal of Systems and Software* (122), pp. 274-286.

Liu, J.Y.-C., Chen, V.J., Chan, C.-L., and Lie, T. 2008. "The Impact of Software Process Standardization on Software Flexibility and Project Management Performance: Control Theory Perspective," *Information and Software Technology* (50:9-10), pp. 889-896.

Lui, T.-W., Piccoli, G., and Desouza, K. 2007. "Degrees of Agility: Implications for Information Systems Design and Firm Strategy," *Agile information systems: Conceptualization, construction, and management*), pp. 122-133.

Lyytinen, K., Mathiassen, L., and Ropponen, J. 1998. "Attention Shaping and Software Risk—a Categorical Analysis of Four Classical Risk Management Approaches," *Information Systems Research* (9:3), pp. 233-255.

Macintosh, N.B., and Quattrone, P. 2010. *Management Accounting and Control Systems: An Organizational and Sociological Approach*. John Wiley & Sons.

Mahadevan, L., Kettinger, W.J., and Meservy, T.O. 2015. "Running on Hybrid: Control Changes When Introducing an Agile Methodology in a Traditional "Waterfall" System Development Environment," *Communications of the Association for Information Systems* (36:1).

Mähring, M., Wiener, M., and Remus, U. 2018. "Getting the Control Across: Control Transmission in Information Systems Offshoring Projects," *Information Systems Journal* (28:4), pp. 708-728.

Maowen, W., Dong, Z.C., Weiyao, L., and Qiang, W.Q. 2012. "Text Topic Mining Based on Lda and Co-Occurrence Theory," in: *Computer Science & Education (ICCSE), 2012 7th International Conference on*. IEEE, pp. 525-528.

Markus, M.L. 1983. "Power, Politics, and Mis Implementation," *Communications of the ACM* (26:6), pp. 430-444.

Martin, J. 1991. *Rapid Application Development*. Macmillan Publishing Co., Inc.

Maruping, L.M., and Magni, M. 2012. "What's the Weather Like? The Effect of Team Learning Climate, Empowerment Climate, and Gender on Individuals' Technology Exploration and Use," *Journal of Management Information Systems* (29:1), pp. 79-115.

Maruping, L.M., Venkatesh, V., and Agarwal, R. 2009a. "A Control Theory Perspective on Agile Methodology Use and Changing User Requirements," *Information Systems Research* (20:3), pp. 377-400.

Maruping, L.M., Venkatesh, V., Thatcher, S.M., and Patel, P.C. 2015a. "Folding under Pressure or Rising to the Occasion? Perceived Time Pressure and the Moderating Role of Team Temporal Leadership," *Academy of Management Journal* (58:5), pp. 1313-1333.

Maruping, L.M., Venkatesh, V., Thatcher, S.M.B., and Patel, P.C. 2015b. "Folding under Pressure or Rising to the Occasion? Perceived Time Pressure and the Moderating Role of Team Temporal Leadership," *Academy of Management Journal* (58:5), pp. 1313-1334.

Maruping, L.M., Zhang, X., and Venkatesh, V. 2009b. "Role of Collective Ownership and Coding Standards in Coordinating Expertise in Software Project Teams," *European Journal of Information Systems* (18:4), pp. 355-371.

Masood, Z., Hoda, R., and Blincoe, K. 2020. "How Agile Teams Make Self-Assignment Work: A Grounded Theory Study," *Empirical Software Engineering* (25:6), pp. 4962-5005.

Maurer, C., and Tiwana, A. 2012. "Control in App Platforms: The Integration-Differentiation Paradox," in: *ICIS 2012 Proceedings*.

McAvoy, J., and Butler, T. 2009. "The Role of Project Management in Ineffective Decision Making within Agile Software Development Projects," *European Journal of Information Systems* (18:4), pp. 372-383.

McDonald, R.P. 1999. *Test Theory: A Unified Treatment*. Mahwah, NJ, US: Lawrence Erlbaum Associates Publishers.

McHugh, O. 2011. "A Study of Clan Control in Agile Software Development Teams."

McHugh, O., Conboy, K., and Lang, M. 2011a. "Using Agile Practces to Influence Motvaton within It Project Teams," *Scandinavian Journal of Information Systems* (23:2), pp. 59-84.

McHugh, O., Conboy, K., and Lang, M. 2011b. "Using Agile Practices to Influence Motivation within It Project Teams," *Scandinavian Journal of Information Systems* (23:2).

McHugh, O., Conboy, K., and Lang, M. 2012. "Agile Practices: The Impact on Trust in Software Project Teams," *IEEE Software* (29:3).

McKinsey. 2017. "Digital Disruption in Insurance: Cutting through the Noise ".

Melnik, G., and Maurer, F. 2006. "Comparative Analysis of Job Satisfaction in Agile and Non-Agile Software Development Teams," *XP*, Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 32-42.

Miles, M.B., and Huberman, A.M. 1994a. *Qualitative Data Analysis : An Expanded Sourcebook*. Thousand Oaks: Sage Publications.

Miles, M.B., and Huberman, A.M. 1994b. *Qualitative Data Analysis: A Sourcebook of New Methods*. Beverly Hills, CA, USA: Sage.

Mimno, D., Wallach, H.M., Talley, E., Leenders, M., and McCallum, A. 2011. "Optimizing Semantic Coherence in Topic Models," in: *Proceedings of the conference on empirical methods in natural language processing*. Association for Computational Linguistics, pp. 262-272.

Misra, S.C., Kumar, V., and Kumar, U. 2009. "Identifying Some Important Success Factors in Adopting Agile Software Development Practices," *The Journal of Systems and Software* (82:11).

Moe, N.B., Dahl, B., Stray, V., Karlsen, L.S., and Schjødt-Osmo, S. 2019. "Team Autonomy in Large-Scale Agile," *Proceedings of the 52nd Hawaii International Conference on System Sciences*.

Moore, G.C., and Benbasat, I. 1991. "Development of an Instrument to Measure the Perceptions of Adopting an Information Technology Innovation," *Information systems research* (2:3), pp. 192-222.

Murungi, D., Wiener, M., and Marabelli, M. "Ups and Downs in Is Projects–the Dynamic Interplay between Control and Emotions,").

Murungi, D., Wiener, M., and Marabelli, M. 2019. "Control and Emotions: Understanding the Dynamics of Controllee Behaviours in a Health Care Information Systems Project," *Information Systems Journal*), pp. 1-25.

Muthén, L.K., and Muthén, B.O. 1998. *Mplus: The Comprehensive Modeling Program for Applied Researchers: User's Guide*. Muthén & Muthén.

Myers, M.D. 2013. *Qualitative Research in Business & Management*. London: SAGE.

Napier, N.P., Keil, M., and Tan, F.B. 2009. "It Project Managers' Construction of Successful Project Management Practice: A Repertory Grid Investigation," *Information Systems Journal* (19:3), pp. 255-282.

Narayanaswamy, R., Grover, V., and Henry, R.M. 2013. "The Impact of Influence Tactics in Information System Development Projects: A Control-Loss Perspective," *Journal of Management Information Systems* (30:1), p. 191.

Nastasi, B.K., Hitchcock, J.H., and Brown, L.M. 2010. "An Inclusive Framework for Conceptualizing Mixed Methods Design Typologies: Moving toward Fully Integrated Synergistic Research Models," *Handbook of mixed methods in social & behavioral research* (2), pp. 305-338.

Nederhof, A.J. 1985. "Methods of Coping with Social Desirability Bias: A Review," *European journal of social psychology* (15:3), pp. 263-280.

Nelson, R.R. 2005. "Project Retrospectives: Evaluating Project Success, Failure, and Everything in Between," *MIS Quarterly Executive* (4:3), p. 5.

Nelson, R.R. 2007. "It Project Management: Infamous Failures, Classic Mistakes, and Best Practices," *MIS Quarterly executive* (6:2).

Nguyen, D.S. 2016. "Success Factors That Influence Agile Software Development Project Success," *American Scientific Research Journal for Engineering, Technology, and Sciences* (17:1), pp. 172-222.

Nickerson, J.A., and Zenger, T.R. 2004. "A Knowledge-Based Theory of the Firm—the Problem-Solving Perspective," *Organization Science* (15:6), pp. 617-632.

Niederman, F., Lechler, T., and Petit, Y. 2018. "A Research Agenda for Extending Agile Practices in Software Development and Additional Task Domains," *Project Management Journal* (49:6), pp. 3-17.

Nissen, M., and Jin, Y. 2007. "Co-Evolution and Co-Design of Agile Organizations and Information Systems through Agent-Based Modeling," *Agile information Systems. Conceptualization, construction, and Management", Elsevier/Butterworth-Heinemann*).

Nurdiani, I., Börstler, J., and Fricker, S.A. 2016. "The Impacts of Agile and Lean Practices on Project Constraints: A Tertiary Study," *Journal of Systems and Software* (119), pp. 162-183.

O'Brien, J., and Marakas, G. 2009. "Introduction to Information Systems." McGraw-Hill, Inc.

O'Hara, M.T., Watson, R.T., and Kavan, C.B. 1999. "Managing the Three Levels of Change," *Information Systems Management* (16), pp. 63-70.

Orlikowski, W.J. 1991. "Integrated Information Environment or Matrix of Control? The Contradictory Implications of Information Technology," *Accounting, Management and Information Technologies* (1:1), pp. 9-42.

Ouchi, W.C. 1977. "The Relationship between Organizational Structure and Organizational Control," *Administrative Science Quarterly* (22:1).

Ouchi, W.G. 1978. "The Transmission of Control through Organizational Hierarchy," *Academy of Management Journal (pre-1986)* (21:2), pp. 173-192.

Ouchi, W.G. 1979. "A Conceptual Framework for the Design of Organizational Control Mechanisms," *Management Science* (25:9), pp. 833-849.

Ouchi, W.G. 1980. "Markets, Bureaucracies, and Clans," *Administrative science quarterly*), pp. 129-141.

Ouchi, W.G., and Johnson, J.B. 1978. "Types of Organizational Control and Their Relationship to Emotional Well Being," *Administrative Science Quarterly* (23:2).

Patton, M.Q. 1990. *Qualitative Evaluation and Research Methods*. SAGE Publications, inc.

Pavlou, P.A., Liang, H., and Xue, Y. 2007. "Understanding and Mitigating Uncertainty in Online Exchange Relationships: A Principal-Agent Perspective," *MIS quarterly*), pp. 105-136.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., and Dubourg, V. 2011. "Scikit-Learn: Machine Learning in Python," *Journal of machine learning research* (12:Oct), pp. 2825-2830.

Pelrine, J. 2011. "On Understanding Software Agility: A Social Complexity Point of View," *Emergence : Complexity and Organization* (13:1), pp. 26-37.

Perreault, W.D., and Leigh, L.E. 1989. "Reliability of Nominal Data Based on Qualitative Judgments," *Journal of marketing research* (26:2), pp. 135-148.

Persson, J.S., Mathiassen, L., and Aaen, I. 2011. "Agile Distributed Software Development: Enacting Control through Media and Context," *Information Systems Journal* (22:6), pp. 411-434.

Piccoli, G., Powell, A., and Ives, B. 2004. "Virtual Teams: Team Control Structure, Work Processes, and Team Effectiveness," *Information Technology & People* (17:4).

Pierce, J.R., and Aguinis, H. 2013. "The Too-Much-of-a-Good-Thing Effect in Management," *Journal of Management* (39:2), pp. 313-338.

Polonsky, M.J., Carlson, L., Grove, S., and Kangun, N. 1997. "International Environmental Marketing Claims: Real Changes or Simple Posturing?," *International Marketing Review*).

Poppendieck, M., and Poppendieck, T. 2003. *Lean Software Development: An Agile Toolkit*, (1 ed.). Addison-Wesley Longman, Amsterdam.

Pressman, R.S. 2010. *Software Engineering: A Practitioner's Approach*. New York: McGraw-Hill Higher Education.

Rahmani, M., Roels, G., and Karmarkar, U.S. 2018. "Team Leadership and Performance: Combining the Roles of Direction and Contribution," *Management Science* (64:11), pp. 5234-5249.

Ramesh, B., Mohan, K., and Lan, C. 2012. "Ambidexterity in Agile Distributed Development: An Empirical Investigation," *Information Systems Research* (23:2), pp. 323-340.

Recker, J. 2012. *Scientific Research in Information Systems: A Beginner's Guide*. Springer Publishing Company, Incorporated.

Recker, J., Holten, R., Hummel, M., and Rosenkranz, C. 2017. "How Agile Practices Impact Customer Responsiveness and Development Success : A Field Study," *Project management journal*. (48:2).

Remus, U., Pregenzer, M., and Wiener, M. 2019. "Control Style Matters: Improving Control Agreement and Transmission in Internal Information Systems Development Projects," *European Conference on Information Systems (ECIS)*, Stockholm & Uppsala, Sweden.

Remus, U., Wiener, M., Saunders, C., and Mähring, M. 2020. "The Impact of Control Styles and Control Modes on Individual-Level Outcomes: A First Test of the Integrated Is Project Control Theory," *European Journal of Information Systems* (29:2), pp. 134-152.

Remus, U., Wiener, M., Saunders, C., Mähring, M., and Kofler, M. 2016. "Control Modes Versus Control Styles: Investigating Isd Project Control Effects at the Individual Level," in: *ICIS 2016 Proceedings*.

Rigdon, E.E., Sarstedt, M., and Ringle, C.M. 2017. "On Comparing Results from Cb-Sem and Pls-Sem

Five Perspectives and Five Recommendations," *Marketing: ZFP – Journal of Research and Management* (39:3), pp. 4-16.

Robey, D., Khoo, H.M., and Powers, C. 2000. "Situated Learning in Cross-Functional Virtual Teams," *Technical Communication* (47:1), pp. 51-66.

Robinson, H., and Sharp, H. 2005. "The Social Side of Technical Practices," in *Extreme Programming and Agile Processes in Software Engineering*. Springer, pp. 100-108.

Rosseel, Y. 2012. "Lavaan: An R Package for Structural Equation Modeling," *Journal of Statistical Software* (48:2), pp. 1 - 36.

Rowe, F. 2014. "What Literature Review Is Not: Diversity, Boundaries and Recommendations," *European Journal of Information Systems* (23:3), pp. 241-255.

Royce, W.W. 1970a. "Managing the Development of Large Software Systems: Concepts and Techniques," in: *Proceedings of WesCon*.

Royce, W.W. 1970b. "Managing the Development of Large Software Systems: Concepts and Techniques," *Proceedings of IEEE WESCON*: Los Angeles, pp. 1-9.

Russell Bernard, H. 2002. "Research Methods in Anthropology-Qualitative and Quantitative Approaches,").

Rust, R.T., and Cooil, B. 1994. "Reliability Measures for Qualitative Data: Theory and Implications," *Journal of Marketing Research* (31:1), pp. 1-14.

Rustagi, S., King, W.R., and Kirsch, L.J. 2008. "Predictors of Formal Control Usage in It Outsourcing Partnerships," *Information Systems Research* (19:2), pp. 126-143.

Saldaña, J. 2016. *The Coding Manual for Qualitative Researchers*. London: SAGE.

Sarker, S., and Lee, A.S. 2003. "Using a Case Study to Test the Role of Three Key Social Enablers in Erp Implementation," *Information & Management* (40:8), pp. 813-829.

Sarker, S., and Sahay, S. 2003. "Understanding Virtual Team Development: An Interpretive Study," *Journal of the Association for Information Systems* (4:1).

Sarker, S., and Sarker, S. 2009. "Exploring Agility in Distributed Information Systems Development Teams: An Interpretive Study in an Offshoring Context," *Information Systems Research* (20:3), pp. 440-462.

Saunders, M., Lewis, P., and Thornhill, A. 2016. *Research Methods for Business Students*. Harlow; Munich: Pearson.

Sawilowsky, S.S. 2009. "New Effect Size Rules of Thumb," *Journal of Modern Applied Statistical Methods* (8:2), p. 26.

Schwaber, K. 1995. "Scrum Development Process," *Conference on Object-Oriented Programming Systems, Languages, and Applications*, pp. 117-134.

Schwaber, K. 2021. "Nexus Guide," *The Definitive Guide to scaling Scrum with Nexus: The Rules of the Game https://www. scrum. org/resources/nexusguide*).

Schwaber, K., and Beedle, M. 2001. *Agile Software Development with Scrum*. Prentice Hall PTR.

Schwaber, K., and Sutherland, J. 2013. "The Scrum Guide." 2018, from https://www.scrumguides.org/docs/scrumguide/v1/scrum-guide-us.pdf

Sharp, H., and Robinson, H. 2004. "An Ethnographic Study of Xp Practice," *Empirical Software Engineering* (9:4), pp. 353-375.

Shaw, J. 1995. "A Schema Approach to the Formal Literature Review in Engineering Theses," *System* (23:3), pp. 325-335.

Shaw, M.E. 1964. "Communication Networks," in *Advances in Experimental Social Psychology*, L. Berkowitz (ed.). Academic Press, pp. 111-147.

Shinkle, G.A., Yang, M.M., Yang, F., Elshaw, J.J., and Schleicher, D.J. 2021. "Enacting Formal Controls in Information System Development: Process Guidance and Goal Importance," *Information & Management*), p. 103430.

Siau, K., Long, Y., and Ling, M. 2010. "Toward a Unified Model of Information Systems Development Success," *Journal of Database Management* (21:1).

Simard, M., and Lapalme, J. 2019. "Self-Organizing Is Not Self-Managing: A Case Study About Governance Challenges in an Agile It Unit and Its Scrum Projects," *Proceedings of the 52nd Hawaii International Conference on System Sciences*.

Snell, S.A. 1992. "Control Theory in Strategic Human Resource Management: The Mediating Effect of Administrative Information," *Academy of Management Journal* (35:2), pp. 292-327.

Spiegler, S.V., Heinecke, C., and Wagner, S. 2021. "An Empirical Study on Changing Leadership in Agile Teams," *Empirical Software Engineering* (26:3), p. 41.

Srivastava, A.N., and Sahami, M. 2009. *Text Mining: Classification, Clustering, and Applications*. CRC Press.

Srivastava, S.C., and Teo, T.S.H. 2012. "Contract Performance in Offshore Systems Development: Role of Control Mechanisms," *Journal of Management Information Systems* (29:1), pp. 115-159.

Stavru, S. 2014. "A Critical Examination of Recent Industrial Surveys on Agile Method Usage," *Journal of Systems and Software* (94), pp. 87-97.

Stewart, G.L. 2006. "A Meta-Analytic Review of Relationships between Team Design Features and Team Performance," *Journal of Management* (32:1), pp. 29-55.

Stewart, G.L., and Barrick, M.R. 2000. "Team Structure and Performance: Assessing the Mediating Role of Intrateam Process and the Moderating Role of Task Type," *The Academy of Management Journal* (43:2), pp. 135-148.

Strasser, A., and Westner, M. 2015. "Information Systems Offshoring: Results of a Systematic Literature Review," *Journal of Information Technology Management* (26:2), pp. 70-142.

Straub, D.W. 1989. "Validating Instruments in Mis Research," *MIS quarterly*), pp. 147-169.

Strauss, A., and Corbin, J. 1998. *Basics of Qualitative Research*. London: Sage.

Sun, W., and Schmidt, C. 2018. "Control Mechanisms and Agile Methodology Use: Data from the Industry," *Journal of Computer Information Systems* (58:3), pp. 234-243.

Syed, T.A., Wiener, M., Mehmood, F., and Abdelrahman, M. 2021. "Control-Style Ambidexterity and Information Systems Project Performance: An Expanded View of Control Activities," *European Journal of Information Systems*), pp. 1-22.

Takeuchi, H., and Nonaka, I. 1986. "The New New Product Development Game," *Harvard business review* (64:1), pp. 137-146.

Tannenbaum, A.S. 1962. "Control in Organizations: Individual Adjustment and Organizational Performance," *Administrative Science Quarterly* (7:2), pp. 236-257.
Tashakkori, A., and Teddlie, C. 2009. *Foundations of Mixed Methods Research: Integrating Quantitative and Qualitative Approaches in the Social and Behavioral Sciences*. Sage publications.
Tessem, B. 2014. "Individual Empowerment of Agile and Non-Agile Software Developers in Small Teams," *Information & Software Technology* (56:8), pp. 873-890.
Tiwana, A. 2010. "Systems Development Ambidexterity: Explaining the Complementary and Substitutive Roles of Formal and Informal Controls," *Journal of Management Information Systems* (27:2), pp. 87-127.
Tiwana, A., and Keil, M. 2009. "Control in Internal and Outsourced Software Projects," *Journal of Management Information Systems* (26:3), pp. 9-44.
Tripp, J.F., and Armstrong, D.J. 2014. "Exploring the Relationship between Organizational Adoption Motives and the Tailoring of Agile Methods," *2014 47th Hawaii International Conference on System Sciences*, pp. 4799-4806.
Tripp, J.F., and Armstrong, D.J. 2018. "Agile Methodologies: Organizational Adoption Motives, Tailoring, and Performance," *Journal of Computer Information Systems* (58:2), pp. 170-179.
Tripp, J.F., Riemenschneider, C., and Thatcher, J.B. 2016. "Job Satisfaction in Agile Development Teams: Agile Development as Work Redesign," *Journal of the Association for Information Systems* (17:4), pp. 267-307.
Tripp, J.F., and Riemenschneider, C.K. 2014. "Toward an Understanding of Job Satisfaction on Agile Teams: Agile Development as Work Redesign," *HICSS 2014*: IEEE, pp. 3993-4002.
Trist, E. 1963. *Organizational Choice; Capabilities of Groups at the Coal Face under Changing Technologies*. London, UK.: Tavistock Publications.
Turner, B.A. 2002. "Patterns of Crisis Behaviour: A Qualitative Inquiry," in *Analyzing Qualitative Data*. Routledge, pp. 209-229.
Venkatesh, V., Brown, S.A., and Sullivan, Y.W. 2016. "Guidelines for Conducting Mixed-Methods Research: An Extension and Illustration," *Journal of the Association for Information Systems* (17:7).
Venkatesh, V., Rai, A., and Maruping, L.M. 2018. "Information Systems Projects and Individual Developer Outcomes: Role of Project Managers and Process Control," (29:1), pp. 127-148.
Version One. 2018. "12th Annual State of Agile Report," Version One.
VersionOne. 2015. "The 9th Annual State of Agile Survey,"), p. 16.
VersionOne. 2016. "The 10th Annual State of Agile Survey,"), p. 16.
VersionOne. 2020. "The 14th Annual State of Agile Report."
Vidgen, R., and Wang, X. 2009. "Coevolving Systems and the Organization of Agile Software Development," *Information Systems Research* (20:3), pp. 355-377.
Vom Brocke, J., Simons, A., Riemer, K., Niehaves, B., Plattfaut, R., and Cleven, A. 2015. "Standing on the Shoulders of Giants: Challenges and Recommendations of Literature Search in Information Systems Research," *Communications of the association for information systems* (37:9), pp. 205-224.
Wall, T.D., Kemp, N.J., Jackson, P.R., and Clegg, C.W. 1986. "Outcomes of Autonomous Workgroups: A Long-Term Field Experiment," *Academy of Management Journal* (29:2), pp. 280-304.
Wallace, L., Keil, M., and Rai, A. 2004. "How Software Project Risk Affects Project Performance: An Investigation of the Dimensions of Risk and an Exploratory Model," *Decision Sciences* (35:2), pp. 289-321.
Wang, E.T., Shih, S.-P., Jiang, J.J., and Klein, G. 2006. "The Relative Influence of Management Control and User–Is Personnel Interaction on Project Performance," *Information and Software Technology* (48:3), pp. 214-220.
Wang, X., Conboy, K., and Pikkarainen, M. 2012. "Assimilation of Agile Practices in Use," *Information Systems Journal* (22:6), pp. 435-456.
Weber, R.P. 1990a. *Basic Content Analysis*. Sage.
Weber, R.P. 1990b. *Basic Content Analysis*. Sage.
Webster, J., and Watson, R.T. 2002a. "Analyzing the Past to Prepare for the Future: Writing a Literature Review," *MIS Quarterly* (26:2), p. 11.

Webster, J., and Watson, R.T. 2002b. "Analyzing the Past to Prepare for the Future: Writing a Literature Review," *MIS Quarterly* (26:2), pp. xiii-xxiii.
Wiener, M., Cram, A., and Remus, U. 2017. "The View from the Top–How Senior Executives Exercise Control over Information Systems Projects to Enhance Performance," in: *European Conference in Information Systems (ECIS)*. Guimarães, Portugal: pp. pp. 1423-1438.
Wiener, M., Mähring, M., Remus, U., and Saunders, C. 2016. "Control Configuration and Control Enactment in Information Systems Projects: Review and Expanded Theoretical Framework," *MIS Quarterly* (40:3), pp. 741-789.
Wiener, M., Mähring, M., Remus, U., Saunders, C., and Cram, W. 2019. "Moving Is Project Control Research into the Digital Era: The "Why" of Control and the Concept of Control Purpose," *Information Systems Research*).
Wiener, M., Remus, U., Heumann, J., and Mähring, M. 2015. "The Effective Promotion of Informal Control in Information Systems Offshoring Projects," *European Journal of Information Systems* (24:6), pp. 569-587.
Williams, L., and Cockburn, A. 2003. "Agile Software Development: It's About Feedback and Change," *Computer* (36:6), pp. 39-44.
Wolcott, H.F. 1994. *Transforming Qualitative Data : Description, Analysis, and Interpretation*. Thousand Oaks, Ca.: Sage.
Wood, S., Michaelides, G., and Thomson, C. 2013. "Successful Extreme Programming: Fidelity to the Methodology or Good Teamworking?," *Information & Software Technology* (55:4), pp. 660-673.
Xu, P. 2009. "Coordination in Large Agile Projects," *The Review of Business Information Systems* (13:4), pp. 29-43.
Xu, P., and Shen, Y. 2016. "Empowering Leadership, Transactive Memory Systems and Agility in Software Development Teams: A Theoretical Framework," in: *AMCIS 2016*.
Yadav, V., Adya, M., Nath, D., and Sridhar, V. 2007. "Investigating An'agile-Rigid'approach in Globally Distributed Requirements Analysis," *PACIS 2007 Proceedings*), p. 12.
Yin, R.K. 2003a. *Case Study Research: Design and Methods*, (3th ed.). Thousand Oaks, CA, USA: SAGE Publications Inc.
Yin, R.K. 2003b. *Case Study Research: Design and Methods*. SAGE Publications.
Yun, S., Faraj, S., and Sims Jr, H.P. 2005. "Contingent Leadership and Effectiveness of Trauma Resuscitation Teams," *Journal of Applied Psychology* (90:6), p. 1288.
Zellmer-Bruhn, M., and Gibson, C. 2006. "Multinational Organization Context: Implications for Team Learning and Performance," *Academy of Management Journal* (49:3), pp. 501-519.
Zheng, Y., Venters, W., and Cornford, T. 2011. "Collective Agility, Paradox and Organizational Improvisation: The Development of a Particle Physics Grid," *Information Systems Journal* (21:4), pp. 303-334.

DECLARATION (ERKLÄRUNG)

Eidesstattliche Erklärung nach § 6 der Promotionsordnung vom 16. Januar 2008

„Hiermit erkläre ich an Eides statt, dass ich die vorgelegte Arbeit ohne Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Die aus anderen Quellen direkt oder indirekt übernommenen Aussagen, Daten und Konzepte sind unter Angabe der Quelle gekennzeichnet. Bei der Auswahl und Auswertung folgenden Materials haben mir die nachstehend aufgeführten Personen in der jeweils beschriebenen Weise entgeltlich/ unentgeltlich geholfen:

- Datenerhebung und Analyse
    - Phil Hennel (geb. Diegmann)
    - Sean Hansen
    - Thomas Schmid
    - Björn Binzer
    - Christoph Rosenkranz
- Prüfung ("Review") der Manuskripte
    - Christoph Rosenkranz

Weitere Personen – neben den in der Einleitung der Arbeit aufgeführten Koautorinnen und Koautoren - waren an der inhaltlich-materiellen Erstellung der vorliegenden Arbeit nicht beteiligt. Insbesondere habe ich hierfür nicht die entgeltliche Hilfe von Vermittlungs- bzw. Beratungsdiensten in Anspruch genommen. Niemand hat von mir unmittelbar oder mittelbar geldwerte Leistungen für Arbeiten erhalten, die im Zusammenhang mit dem Inhalt der vorgelegten Dissertation stehen. Die Arbeit wurde bisher weder im In- noch im Ausland in gleicher oder ähnlicher Form einer anderen Prüfungsbehörde vorgelegt. Ich versichere, dass ich nach bestem Wissen die reine Wahrheit gesagt und nichts verschwiegen habe."

Ort, Datum

Ratingen, 10.01.2022

Unterschrift

CURRICULUM VITAE

Tim Dreesen (Dipl. Wirt.-Inf.)

Anschrift: Am Kiefernhain 3b, 40885 Ratingen

Geburtsdatum / -ort: 14.08.1979 in Mülheim a. d. Ruhr

Staatsangehörigkeit: deutsch

Familienstand: verheiratet, 1 Kind

XING: www.xing.com/profile/Tim_Dreesen

LinkedIn: www.linkedin.com/in/tim-dreesen-25b599a0

BERUFSERFAHRUNG

11/2020 bis heute

Technology Consultant ALDI Nord, Advanced Technology Solutions, Essen. Tätigkeiten u.a.:

- Optimierungen der IT Supply Chain Solutions
- Entwicklung von IT Lösungen im Bereich Outbound, Logistics & Transportation
- Projektmanagement innerhalb eines digitalen Transformationsprograms
- Verantwortung für den Bereich Datenmigration der neuen Systemlandschaft

05/2015 bis 10/2020

Wiss. Mitarbeiter Universität zu Köln, Cologne Institute for Information Systems (CIIS), Köln. Tätigkeiten u.a.:

- Verantwortlich u.a. für Praxispartnerkooperation mit dem Fachbereich Wirtschaftsinformatik
- Vorbereitung, Durchführung und Nachbereitung von Lehrveranstaltungen der Wirtschaftsinformatik
- Management der IT-Landschaft des Fachbereichs
- Verantwortung gegenüber studentischen Hilfskräften und Tutoren (z.B. Einstellung, Personalgespräche, Betreuung)
- Organisation von Seminaren, Workshops und Events

11/2010 bis 04/2015

Senior Consultant Mieschke Hofmann und Partner GmbH im Bereich Technology & Integration Solutions, Essen. Tätigkeiten u.a.:

- Verantwortlich für den Bereich Migration und Master Data Governance in internationalen SAP-Rollout Projekten

- Durchführung von Anforderungsanalysen inkl. Ausarbeitung von fachlichen und technischen Spezifikationen
- Fachliche (Teil-)Projektleitung insb. Planung und Risiko Management
- Evaluierung neuer Technologien und Trends, mit starkem Fokus auf Projekt- und Kundenorientierung.
- Unterstützung bei Pre-Sales/ Akquisetätigkeiten und Angebotserstellung, inkl. Aufwandschätzungen

07/2003 bis 12/2004

Informatikkaufmann Stadtsparkasse Oberhausen im Bereich Organisation, Oberhausen. Tätigkeiten u.a.:

- Aufgaben im Bereich der System- und Netzwerkadministration
- Ist-Analyse und Aufbau einer Sicherheitsstrategie
- First und Second Level Support im Bereich aller eingesetzten Soft- und Hardwarelösungen
- Vorbereitung und Durchführung von Anwenderschulungen

BILDUNGSWEG

05/2015 bis heute

Promotion im Bereich Wirtschaftsinformatik (Dr. rer. pol.) Cologne Institute for Information Systems (CIIS), Universität zu Köln, Köln

- Thema: "Agile Entwicklung von Informationssystemen"
- Fokus: Steuerung und Kontrolle von agilen Teams, insb. in Hinblick auf a) die Erhöhung von Projekterfolgen und b) bei gleichzeitiger Wahrung von Agilität der Teams.
- Publikationsliste: www.dreesen-online.de/publication

01/2005 bis 02/2010

Diplomstudiengang Wirtschaftsinformatik Universität Duisburg – Essen, Essen

- Fokus: "Betriebliche Kommunikationssysteme"
- Abschlussarbeit: „Einführung von CRM – Entwicklung eines ganzheitlichen Vorgehensmodells"
- Note: sehr gut (1,5)

08/2000 bis 06/2003

Ausbildung zum Informatikkaufmann Stadtsparkasse Oberhausen, Oberhausen

- Abschluss Informatikkaufmann (IHK)
- Berufsschule Heinz-Nixdorf-Berufskolleg, Essen

08/1990 bis 06/1999

Allgemeine Hochschulreife Städt. Gymnasium Broich, Mülheim a. d. Ruhr

AKADEMISCHE DIENSTE

Lehre

- Dozent der Lehrveranstaltung „Capstone Project Information Systems“ in der Wirtschaftsinformatik (Fallstudienbasierte Softwareentwicklungsprojekte kleiner Teams auf Basis realer Probleme der Praxispartner)
- Trainer für Zertifizierungen im Bereich SAP ERP (TERP10) und S/4Hana (TS410) für Studierende
- Betreuung von Seminar- und Abschlussarbeiten (Diplom, Bachelor, Master)

Reviewer

- *Journals:* Information Systems Journal, Project Management Journal
- *Conferences:* International Conference on Information Systems, European Conference on Information Systems, Americas Conference on Information Systems, Hawaii Conference on System Sciences, Wirtschaftsinformatik, Australasian Conference on Information Systems, International Research Workshop on IT Project Management

Mitgliedschaften

- Association for Information Systems (AIS)
- Special Interest Group for IT Project Management (SIGITPROJMGMT)

WEITERE KENNTNISSE

Zertifizierungen

- SCRUM Foundation (TÜV Süd)
- ITIL Foundation Zertifikat in IT Service Management
- Integrierte Geschäftsprozesse mit SAP ERP (TERP10)

Sprachkenntnisse

- Deutsch (Muttersprache)
- Englisch (fließend in Wort und Schrift)

Führerschein

- Klasse B

INTERESSEN

Badminton, Darts, Squash, Tennis, Sci-Fi Literatur, Reisen, Gitarre

Ratingen, 08. Januar 2022