VESTIGE OF THE PRESENT: SOCIO-TECHNICAL FACTORS IN THE CONSTRUCTION OF LEGACY PERCEPTIONS OF INFORMATION SYSTEMS ARTIFACTS

By

JULIA STACHOFSKY

A dissertation submitted in partial fulfillment of
the requirements for the degree of

DOCTOR OF PHILOSOPHY

WASHINGTON STATE UNIVERSITY
Carson College of Business

MAY 2024

© Copyright by JULIA STACHOFSKY, 2024
All Rights Reserved

© Copyright by JULIA STACHOFSKY, 2024
All Rights Reserved

To the Faculty of Washington State University:

The members of the Committee appointed to examine the dissertation of JULIA STACHOFSKY find it satisfactory and recommend that it be accepted.

Michelle Carter, Ph.D., Chair

Deborah Compeau, Ph.D.

Robert E. Crossler, Ph.D.

Richard D. Johnson, Ph.D.

ACKNOWLEDGMENT

While my name is on the first page, I could not have done this alone. Foolishly, I will attempt the impossible task of thanking everyone who helped me along the way. If I forgot to thank you in this section, rest assured it is not personal, and I will feel bad about it! Secretly, I am a sentimental person, so bear with me on the length.

First, I would like to thank my dissertation chair, Michelle Carter. I could not have asked for a more intelligent and supportive mentor to chair my dissertation. Her sharp insights undergird both this work and my personal development as a scholar. Additionally, the most challenging moments of my life overlapped with my time as a graduate student, and I received nothing but immediate support and kindness in response from Michelle. I know it is gauche to flatter your advisor too much in the acknowledgments, but I will close by saying that working with her has sincerely been the most fulfilling, intellectually engaging, and enjoyable experience of my professional career, and I am grateful for their mentorship.

I would also like to thank Debbie Compeau, another beacon of support during my time at WSU. I am grateful for her help in funding this research. Her rule of not allowing me to flippantly say, “Where is the IT artifact?” in my first-year seminar has probably had the single most significant impact on my scholarship. Her teaching course also helped me find my footing and confidence as an instructor. Teaching went from something I was afraid of to the part of the job I love most. I cannot think of a more thoughtful and qualified person to have at the helm of Carson College of Business.

I would also like to thank Rob Crossler. I admittedly threw him a few curveballs during his time as department chair, but throughout it all, he was supportive and respectful during a personally tumultuous time. I am also very grateful for his continued guidance and mentorship, not just on my dissertation committee but on the papers we have co-authored together and continue to work on.

I would also like to thank Rich Johnson for his mentorship while I was his teaching assistant and later for his support as a committee member on this dissertation. Rich has been good about reminding me to not neglect the social aspect of the legacy phenomenon in this dissertation. I also wish to thank Vicky Reibenspiess, who has been a wonderful mentor since joining WSU. Some of the most valuable and enjoyable moments in my PhD have been discussing research and teaching with her. I know we are both somewhat bad about attending conferences, but I hope we manage to cross paths at one soon!

I would also like to thank both Mauricio Featherman and Joe Compeau, who were very helpful in my early days of the program and continued to provide support throughout my time here. Being their teaching assistant helped me find my way in teaching, and I appreciate their mentorship and kindness. I would also like to thank Youngjin Kwon and Long Nguyen. They have both been very kind to me and immediately fit in with the positive culture of our department. I also would like to thank my fellow Ph.D. students Ben, Carlos, Deepika, Demi, Elliott, Fang, GS, Jalal, Jasper, JT, Kamrul, Kassie, Kesha, Natalie, Rebecca, Shaun, Shirley, Sikdar, Victoria, and Zonayed. I have grown especially close with Meg, Fang, Shirley, and JT in my time at WSU and am very grateful for our continued friendship.

I would be remiss if I did not also thank the excellent department administrators with whom I have had the pleasure of working: Lael, Whitney, Minie, and Ashlee. Minie and Ashlee especially have helped navigate the academic bureaucracy and are the most wonderful, kind, and caring colleagues to work with. I am also thankful to Chuck Munson and Titus Nordquist for their help with navigating the administrative side of the graduate school processes.

There are people outside of my immediate department that I would also like to thank. I want to give thanks to my undergraduate IS advisor, Lori Baker-Eveleth. She was the first person to give me an opportunity to work on research as an undergrad and has been an incredible mentor throughout my IS education. I hope she enjoys her well-earned retirement! I would also like to thank my undergraduate economics advisor/meme expert, Dan Hickman, and Professor Stefanie Ramirez, for their kindness and advice throughout my undergraduate and graduate studies.

I want to thank Assefaw Gebremedhin for involving me in the Cybersecurity Education and Research project, which has been a fantastic opportunity to mentor bright and engaged students. I am also grateful to Jennifer Henrichsen for allowing me to mentor her two student teams. I would also like to thank Christian Schaupp for his support and patience as a co-author on my first big research project at WSU. I am also grateful to the Thomas S. Foley Institute for Public Policy and Public Service for funding that research via the Alice O. Rice. Graduate Fellowship. I also thank France Bélanger for her ongoing mentorship on our current research project. I am also grateful to Allen Lee, Julien Malaurent, Jason Thatcher, Andrew Burton-Jones, Suprateek Sarker, and Andreas Eckhardt for taking the time to visit WSU and give guidance to the doctoral students during my time in the program.

This dissertation also benefitted immensely from two doctoral consortiums I had the privilege to attend. First, I would like to thank Atreyi Kankanhalli and Elena Karahanna for organizing the consortium at AMCIS 2023 and Dorothy Leidner for her feedback on my dissertation. I would also like to thank Sue Brown and Michelle Carter for organizing the consortium at HICSS 2024 and Sirkka Jarvenpaa and Yasser Rahrovani for their valuable feedback on my dissertation. Mentoring PhD students can be thankless work, but I am grateful to all the organizers and mentors who made those events possible. This dissertation is better for it, and I hope to pay that service forward.

I would also like to thank my friends for their patience and support. My friends in Bunni Server, RNG Walk, OS Paradise, Spencer, Thomas, Madison, Lars, Tony, Brandi, and many others have all provided much-needed relief from work throughout these five years. I am also grateful to Meg and David Barbieri for their support in letting me stay at their home while traveling for interviews. I am so glad to have met them during the Ph.D. and happy to call them both my friends. I would also like to thank my friend Jenn for all their support and friendship. I especially wish to thank my best friend Brad, who has been unrelenting in his support and borne the brunt of my yammering and complaining. While also reminding me that I cannot always use my dissertation as an excuse to skip out on life and friends. I deeply cherish our friendship, and working with him is the only thing I miss about industry.

To close, I thank my family for their support throughout this endeavor. Despite not having the opportunity to complete their own college educations, my parents instilled the importance of education in my life from an early age. Education has consistently been the most enriching and fulfilling aspect of my life, and I am eternally grateful for their support in pursuing it. I would also like to thank my aunt Stacie for always being there for me, my grandparents for helping me through my undergraduate studies, our newest family member Gabe for listening to me yap about research, and Lucy for sparing me the trauma of my elderly childhood dog dying while writing my dissertation.

A final acknowledgement goes to my younger sisters, Lindsey and Kaylee, whom I love more than anything else in the world. They make me proud every day, and their support throughout the last five years means more to me than I can ever hope to express.

VESTIGE OF THE PRESENT: SOCIO-TECHNICAL FACTORS IN THE CONSTRUCTION OF LEGACY PERCEPTIONS OF INFORMATION SYSTEMS ARTIFACTS

Abstract

by Julia Stachofsky, Ph.D.
Washington State University
May 2024

Chair: Michelle Carter

Legacy systems are formerly adequate incumbent information systems perceived as insufficient through a combination of social and technical factors. Legacy systems continue to be an expensive and challenging information technology asset for organizations to manage. However, much of the existing information systems literature does not focus on end-of-life information systems phenomena. This dissertation responds to this need for research on end-of-life information systems phenomena from a behavioral perspective by reviewing the information systems literature on legacy systems, developing a definition of legacy systems as a socio-technical construction, creating a scale for measuring a legacy perception of a system, and developing four additional scales for measuring the characteristics of information technology artifacts. New scales and statistical models were tested through a survey of IT managers in the United States.

Findings from this research provide support for legacy perception as a new construct. Results also suggest that system age is not a key influencer of legacy perception, but system capability shortcomings and a lack of system support availability are key influences. This

research also models interactions of the physical structures of legacy systems, finding that integration and complexity positively influence the adaptability of legacy system artifacts. As well as the adaptability of an artifact and state tracking abilities have a positive influence on representational fidelity. This study also finds that a legacy perception of a system positively influences both system investment behaviors and intentions to replace a system. Implications for theory and practice and opportunities for future research are discussed.

TABLE OF CONTENTS

LIST OF TABLES

Page

LIST OF FIGURES

LIST OF EQUATIONS

Dedication

For what was.

Hope I did alright with the place.

CHAPTER ONE: INTRODUCTION

Legacy systems remain an integral part of organizations' information technology (IT) portfolios, quietly running the operations of most organizations. It can be tempting to dismiss legacy systems as archaic drains on organization resources, but it is important to note that these systems persist in part because they continue to provide significant value to organizations (Gholami et al., 2017; Light, 2003) and only persisted long enough to be considered legacy systems in the first place due to being successful systems (A. J. O'Callaghan, 1999). Regardless of whether they are seen as a negative asset, legacy systems remain an essential management problem. Practitioners have continually reported high relevance and investment in legacy system issues (e.g., Kappelman et al., 2016, 2019, 2020, 2021, 2022; Luftman, 2005; Luftman et al., 2006), since being identified as a key IT management area by the Society of Information Management (Brancheau et al., 1996). In this research, I define legacy systems as formerly adequate incumbent information systems perceived as insufficient through a combination of social and technical factors.

Organizations continually struggle with legacy systems management, with some estimates suggesting that as many as 74% of mainframe modernization projects fail to be completed (2020 Mainframe Modernisation Business Barometer Report, 2020). Migration failures are also trending upwards, impacting over 40% of organizations (Konkel, 2016). However, the information systems (IS) literature is largely silent on managing these systems, leaving the topic to a computer science (CS) literature focused on the technical minutia of migration and reverse engineering. Despite the importance of these systems within organizations, much of the IS literature has focused on the early stages of the IS lifecycle, with significantly less attention given to post-implementation IS phenomena (C. Edwards, 1984; Furneaux & Wade, 2011; Soliman & Rinta-Kahila, 2019).

This dissertation attempts to address this dearth of IS research on legacy systems while also building a theoretical foundation for continued research in this area. Drawing on insights from representation theory (Wand & Weber, 1995), I propose a model that focuses on the structures of the *legacy system itself* as the focal phenomenon and how its internal technical structures interact within the system and with the external social systems. Most representation theory research focuses on deep structure (Recker et al., 2021) and fails to consider that deep structure is necessarily intertwined with surface and physical structures (Recker et al., 2021; Wand & Weber, 1995). Deep structures are implemented through physical hardware and software and accessed through user interfaces; they cannot exist without those physical and surface structures (Wand & Weber, 1995). As such, this research considers all structures of the legacy system artifact when theorizing, not just the deep structure model.

I also aim to answer the call for more research where technology is a primary focus and unit of analysis (Matook & Brown, 2017; Orlikowski & Iacono, 2001; Tiwana, 2019). I argue that deep engagement with technical artifacts does not need to be limited to design research. This dissertation develops five new scales for measuring IT artifact characteristics to operationalize technical concepts in the context of behavioral research. These scales are used in this research to study the structures of legacy systems but are general enough that they can be used in many other behavioral research contexts as well.

Legacy systems are, however, not entirely a technical phenomenon (Gibson et al., 1998; Light, 2003). Within the legacy systems domain, many social factors, such as replacement risk, system support, and investment (Furneaux & Wade, 2017), influence management decision-making. Furthermore, user interactions and organizational contexts can shape whether a system is perceived as legacy (Alvarez, 2000). In this vein, I argue that while the legacy system is a technical artifact, it cannot be understood without theorizing about the social interactions surrounding it. A system only becomes legacy when attributed the label by a social actor. It is a social construction based on real technical phenomena. As such, an IS understanding of legacy systems necessitates a socio-technical perspective (Bostrom & Heinen, 1977a, 1977b; Sarker et al., 2019).

Legacy systems as a topic is quite broad, encompassing various areas such as migration, replacement, integration, extraction, strategy, and security, to name a few. This dissertation will specifically focus on how IT managers form perceptions of legacy. Essentially, I explore why people view a system as legacy. I argue that while a legacy system is a technical artifact, the label of "legacy" is a construction made by a social actor based on perceptions of that artifact and the external social environment. There are many different operational definitions of a legacy system, but no investigation into what leads to a system being perceived as legacy. In this research, I address the following questions:

RQ1: What socio-technical factors result in the formation of a legacy perception of an information system?

RQ2: How does a legacy perception of a system impact replacement intentions and investment in an information system?

I first review the IS literature to identify potential factors. Those factors are then used in two different structural equation models to understand how those factors influence a legacy perception and interact with each other, such as in different layers of the technical structure.

Legacy perception is further studied to understand the impact a legacy perception has on system replacement intentions and investment decisions.

While this dissertation does not cover all aspects of legacy systems, it aims to address an essential behavioral aspect of the phenomenon and lay the groundwork for richer IS theorizing. Through this research, the IS field will better understand the specific factors that influence the perception of legacy, potentially resolving existing conflicting definitions of a legacy system and identifying areas of focus for future legacy systems research. For practice, knowing which factors most influence legacy perceptions can assist with IT management and development decisions around system design and allocation of resources. In the legacy systems context, available labor with experience is significantly constrained (*The Aging IT Workforce and Legacy Application Modernization*, 2024), requiring strategic decisions on allocating development resources.

Findings from this research provide support for legacy perception as a new construct. Four additional constructs are also developed: integration, connectivity, state, and adaptation. However, results for these measures are more mixed. In particular, there is a lack of discriminant validity between connectivity and the IT characteristic measure of integration and socio-technical characteristic of use representational fidelity. Further research and scale testing for measuring technical characteristics in behavioral surveys is necessary.

Surprisingly, results also suggest that system age is not a key influencer of legacy perception. However, system capability shortcomings and a lack of support availability are key influences. Within the physical structures of legacy systems, integration and complexity positively influenced the adaptability of legacy system artifacts, contradicting the theorized model. The adaptability of a legacy systems artifact and state tracking abilities also positively influenced a system's representational fidelity. This study also finds that a legacy perception of a system positively influences system investment behaviors and intentions to replace a system.

The remainder of this dissertation is broken into six chapters. In Chapter Two, I will discuss the theoretical foundations of this work: the representation theory of information systems, the systems thinking framework of IT artifacts, and the complexity theory of technology. I then review existing definitions of legacy systems in both the IS and CS literature. These definitions and the theoretical foundation are used to develop a new definition of legacy systems via theoretical propositions. Chapter Three is an overall literature review of the IS work on legacy systems categorized by topic. Chapter Four develops two models of legacy perception and hypotheses for each model. Chapter Five will cover the methodology for testing the models, and Chapter Six will report the results. The dissertation closes with Chapter Seven, discussing theoretical contributions, practical implications, and opportunities for future research. Additional scale development and measurement model statistics can be found in the appendices.

CHAPTER TWO: THEORETICAL FOUNDATION

The foundations of this dissertation are built based on a representation theory conceptualization of information systems (Recker et al., 2021; Wand & Weber, 1990, 1995), a systems theory framework of IT artifacts (Goldkuhl, 2013b; Matook & Brown, 2017), and a complexity theory of technology (Arthur, 2009; Arthur & Polak, 2006). The following sections cover each foundational theory, the compatibility of synthesizing these theories, and define a legacy system within the constraints of these theoretical foundations.

Representation Theory of Information Systems

The first fundamental assumption of representation theory is that information systems are phenomena that can be studied by themselves, divorced from social context (Wand & Weber, 1990). More specifically, they posit, "An information system is an artifactual *representation* of a real-world system as perceived by someone, built to perform information processing functions." (Wand & Weber, 1990, p. 62). What is unique about this definition of an information system is that an information system is purely technical; it is a *type* of IT artifact. It is an artifact that *interacts* with a social subsystem. The social subsystem is not a component of the information system itself. This contradicts other conceptions of IS artifacts in the field, such as those that consider the combination of technical, social, and information artifacts to be an IS artifact (A. S. Lee et al., 2015). The representation theory conception of an IS artifact only consists of the technical and information artifacts.

It should be noted that representation theory does not suggest that those other social aspects do not matter. Representation theory is clear that the representation embedded within the system is always based on human *perception* of reality, not reality itself (Wand & Weber, 1995). The theory argues that as an artifact, an information system can be ontologically separated and meaningfully studied. It is a thing in the world that can be observed, premised on a scientific realist materialist foundation (Bunge, 1977, 1979). The representation theory view is quite skeptical of IS theory that suggests humans and material objects cannot be separated ontologically, especially regarding agential realism (R. Weber, 2020, 2023). The information system interacts with other components, but those components can be understood as separate, consisting of their own properties and interactions like the information system itself.

The second fundamental assumption underlying representation theory is that the more accurately a real-world system is represented in an information system, the more useful the information system will be (Wand & Weber, 1995). Thus, information systems development aims to make the internal representation as accurate as possible. The real-world system is represented by three different structures within the information system. Figure 1 models the interactions of the physical, surface, and deep structures as they relate to each other and reality.

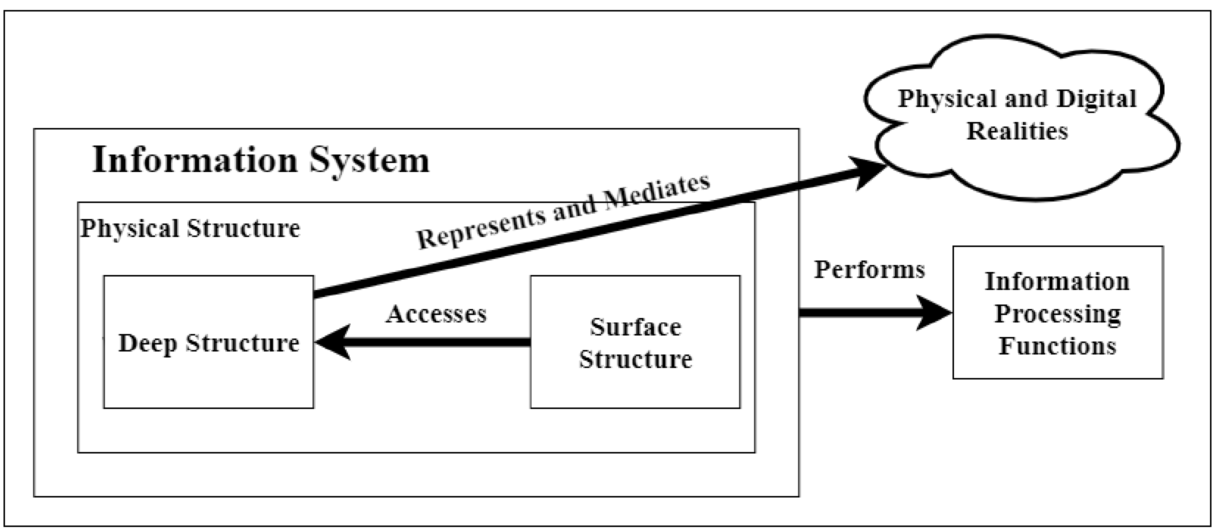

Figure 1: Representation Theory of Information Systems

The first structure is the physical structure, which is the combination of hardware, software, and design methodologies used to construct the information system (Recker et al., 2019; Wand & Weber, 1995). The two subsequent structures, surface structure and deep structure, are abstractions implemented on top of the physical structure (Wand & Weber, 1995). The only aspect of the information system that exists in physical reality is the physical structure. It is the manifestation of the artifact. The second structure is the surface structure, which is how users of the information system access the deep structure of the system (Recker et al., 2019; Wand & Weber, 1995). This is essentially the user interface components by which the user interacts with the system. If the interaction is not transparent, then the effective use of the system is diminished (Burton-Jones & Grange, 2013).

The final structure, which most representation theory research focuses on (Recker et al., 2021), is the deep structure of the information system. The deep structure represents the real-world system within the information system structure (Recker et al., 2019; Wand & Weber, 1995). The deep structure is an abstraction that is supported by the components of the physical and surface structures. The focus on deep structure in the literature is related to the assumption above, as the more accurate the real-world system is represented in the system structure, the more useful the information system is (Wand & Weber, 1995).

While many of the theoretical principles above hold, some assumptions have changed in light of digital reality (Baskerville et al., 2020; Recker et al., 2021). The assumption that the deep structure of an information system represents a physical, real-world system is only partly true. As more of the human experience has been computed, information systems must also represent digital real-world systems (Recker et al., 2021). Information systems also mediate representations, translations, executions, and changes between physical and digital real-world systems (Recker et al., 2021). However, those digital realities being represented still manifest in the physical world in some capacity through physical structures.

Most research that applies representation theory is done in the conceptual modeling literature (Recker et al., 2019, 2021). However, recent theoretical (Burton-Jones & Grange, 2013) and empirical (Burleson, 2016; Burleson et al., 2021) works have adapted the foundational concepts of representation theory to other contexts. My research is situated within this ongoing project to adapt representation theory concepts to non-conceptual modelling-based IS research. I place an increased emphasis on the physical structure of the information system. Since an information system is an engineered artifact and a specific type of IT artifact, I will now explain the theory of IT artifacts that underlies this research.

Systems Thinking Framework of IT Artifacts

This research holds a systems view of the IT artifact (Goldkuhl, 2013b; Matook & Brown, 2017). Specifically, I use the following definition of an IT artifact:

> "An IT artefact is a physical artefact based on technology. Every running IT artefact relies on some hardware. The software and hardware can be seen as an integrated whole. Without the software, the hardware is just an empty shell. Without hardware, the software is just symbolic expressions. But together they are machines with the power to execute intentionally designed information-processing tasks." (Goldkuhl, 2013b, p. 93).

Most notable about this definition of the IT artifact is that it conceptualizes the artifact as an integrated whole of subsystems (Matook & Brown, 2017). It is only through the *combination* of the hardware and software subsystems that an IT artifact is constructed. Additionally, in contrast to the ensemble (Goldkuhl, 2013a; Orlikowski & Iacono, 2001) and sociomaterial (Orlikowski, 2010) perspectives which would suggest the social and technical cannot be meaningfully separated, this definition of the artifact identifies the social subsystem as external to the technical subsystem. Figure 2 provides a summary of the systems thinking framework of IT artifacts.

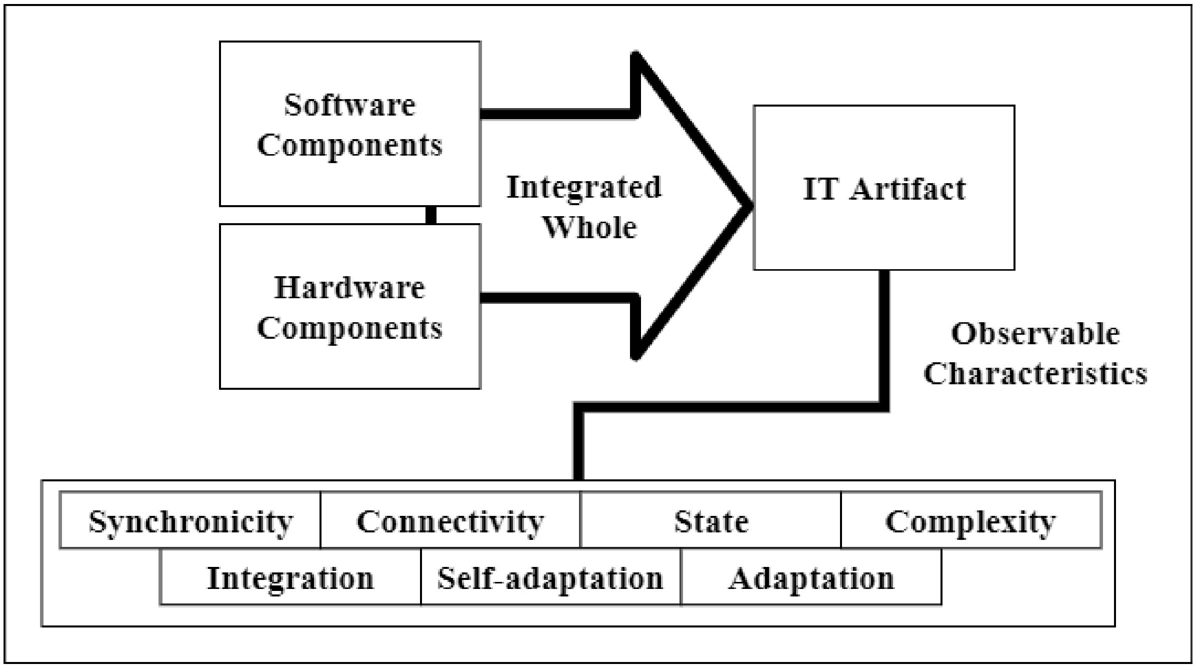

Figure 2: Systems Thinking Framework of IT Artifacts

Matook and Brown (2017) propose seven different characteristics of IT artifacts based on systems thinking. *Integration* is a measure of the aggregation of the internal IT artifact parts, which can be highly integrated or highly fragmented (Matook & Brown, 2017). *Connectivity* measures how connected the IT artifact is with external system parts and the environment outside of the system boundary, which can be highly connected or highly isolated (Matook & Brown, 2017). These characteristics are derived from two systems thinking concepts. The first is *system parts, wholeness, and system structure,* which suggest that systems must be understood as a whole, not only by their individual parts (M'pherson, 1974; Von Bertalanffy, 1956). The second systems thinking concept they use is *system boundary and environment*, which posits that all systems have boundaries that separate them from the outer environment, but that environment also influences the system itself (Ackoff, 1971).

State refers to the extent to which the IT artifact remembers its state, which can be completely stateless or completely stateful (Matook & Brown, 2017). This is derived from the concept of *system state and history of system state*, which is the notion that a system can have a state, that state can change, and that state change is remembered by the system (Ackoff, 1971; Beer, 1972). *Complexity* is the number of interdependent relations that make up the IT artifact, which can be less complex or highly complex (Matook & Brown, 2017). This characteristic is based on the systems thinking concept of *hierarchical order, wholeness, and complexity*, which is how interdependent the subsystems are upon each other to create the whole system (Boulding, 1956; Miller, 1965; Von Bertalanffy, 1972).

Synchronicity is the time aspect of input and output exchanges, which can be highly asynchronous or highly synchronous (Matook & Brown, 2017). The IT characteristic *self-adaptation* is defined as the ability of an IT artifact to change based on inputs from the environment or other system parts on its own, which can be highly non-adaptive to highly adaptive (Matook & Brown, 2017). Lastly, *adaptation* is defined as the ability of the IT artifact to change, which can be highly static or highly dynamic, with the difference being that the change is not done by the artifact itself (Matook & Brown, 2017). These characteristics are based on the systems thinking concepts of *transformation and feedback*, which is how the system processes inputs and outputs (Miller, 1965; Von Bertalanffy, 1956).

Thus far, I have established a theory of information systems and a framework of IT artifacts, with an information system being a specific type of IT artifact. The following section will discuss what I mean by technology at a more fundamental level. This is necessary because IT artifacts are technologies and, as such, are subject to the dynamics of how technologies are created and evolve. In particular, it is essential to articulate that the physical structure of an information system is not a singular technology but a combination of many interdependent technologies.

Complexity Theory of Technology

The theory of technology underlying this work is adapted from Arthur (2009), based itself on the foundations of complexity economics. Arthur defines technology as "…a means to fulfill a human purpose…", "…an assemblage of practices and components…", and "…the entire collection of devices and engineering practices available to a culture." (2009, p. 28). This definition of technology is broad, including engineered artifacts (e.g., accounting software) and conceptual artifacts (e.g., object-oriented software design patterns). Starting from this definition, Arthur builds a comprehensive theory of technology to understand the essence of technology itself, rather than the issues that surround and interact with technology, which is the emphasis of most technology literature (Arthur, 2009). For this research, I will cover the most relevant areas of Arthur's theory: phenomena, combination and structure, evolution, and structural deepening. Figure 3 summarizes the interrelation of these concepts, which will be explained throughout this chapter.

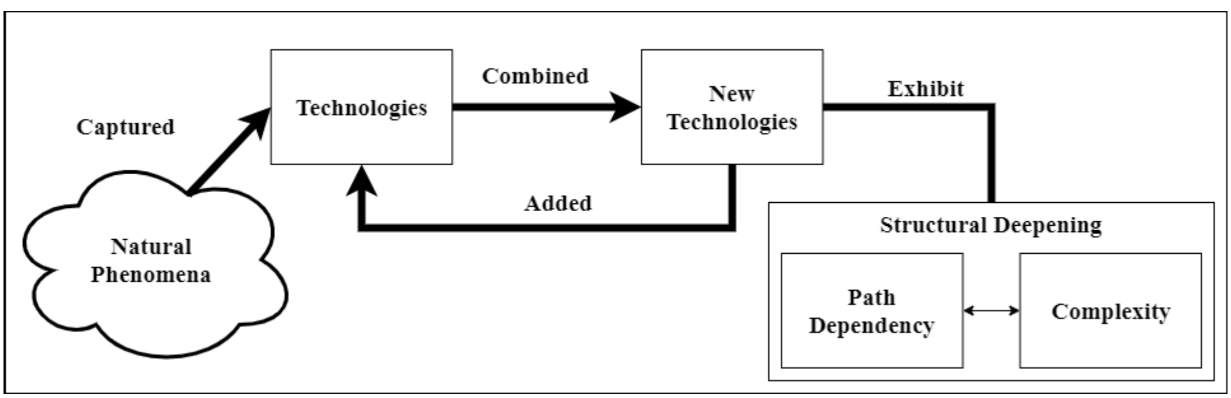

Figure 3: Complexity Theory of Technology

At the most foundational level of the theory of technology are phenomena, observable facts or events in the world (Arthur, 2009). A technology must be based on some phenomenon or set of phenomena that occur in nature, are exploited, and are used for a purpose (Arthur, 2009). This is similar to, but distinct from, the common assumption that technology is the application of science (Bunge, 1966; Iivari, 2020). Science is a method of phenomena discovery but is not the *only* method, and technologies can be created without the grounding of a kernel theory of the natural sciences (Arthur, 2009; Iivari, 2020).

Once a phenomenon or set of phenomena has been captured as a technology, they are added to the total stock of technologies. This means that those existing technologies can be combined to create new, more complex technologies (Arthur, 2009; Arthur & Polak, 2006). Similar to biological processes (Lenski et al., 2003), it is only through simpler technologies that more complex technologies evolve (Arthur & Polak, 2006). These more complex technologies are combined from component parts and assemblies, organized around a concept or principle (the purpose the technology serves), and consist of main assembly architectures that interact with many often equally complex subsystems supporting the main assembly (Arthur, 2009). Technologies, other than their most basic form, are not a singular entity, but a system of increasingly complex technologies that interact with each other and become the building blocks of newer technologies. This cycle continues repeatedly over time with increasing complexity becoming possible as the stock of technologies grow (Arthur, 2009).

The last main idea of the complexity theory of technology directly relevant to this work is the notion of structural deepening. Structural deepening describes the process by which existing technologies become more complex. Often, components in a system are replaced with newer (usually more complex) components. Additional components may also be added as workarounds to existing system limitations (Arthur, 2009). The technology evolves to meet new demands, but at a cost of complexity. All new components and assemblies added to the technology must account for the existing components, and the new complexity may lead to path dependencies and lock-in when trying to adapt the technology further in the future (Arthur, 2009).

Figure 4 synthesizes the theory of information systems, IT artifacts, and technology used for this work. The commonality between these theories is a shared theoretical base of general systems theory (Boulding, 1956; Von Bertalanffy, 1956), allowing for interconnections between these theories. The theory of technology is based on complexity economics, which is based on general systems theory concepts of hierarchical order, wholeness, and complexity (Arthur, 2009). The theory of IT artifacts derives the characteristics of the artifact from systems thinking concepts, which are applications of general systems theory concepts (Matook & Brown, 2017). Moreover, the theory of information systems applies general systems theory, emphasizing system representation, state tracking, and system decomposition (Recker et al., 2019; Wand & Weber, 1995).

Based on this shared theory base, I argue how information systems are constructed as artifacts. I posit that combinations of technologies create IT artifacts, which are, in turn, added back to the base of technologies to create more complex IT artifacts (Arthur, 2009; Matook & Brown, 2017). For example, at one point, a Linux server was considered a new IT artifact consisting of programming techniques, hardware technologies, and software operating systems code. However, many IT artifacts are now built on top of the Linux architecture in which it is a component combined with other technology subsystems to produce new artifacts. The theory of IT artifacts explains that software and hardware components are combined to create IT artifacts (Goldkuhl, 2013b; Matook & Brown, 2017). The theory of technology explains how that combinatory process happens and how the stock of software and hardware components emerge by which IT artifacts are created.

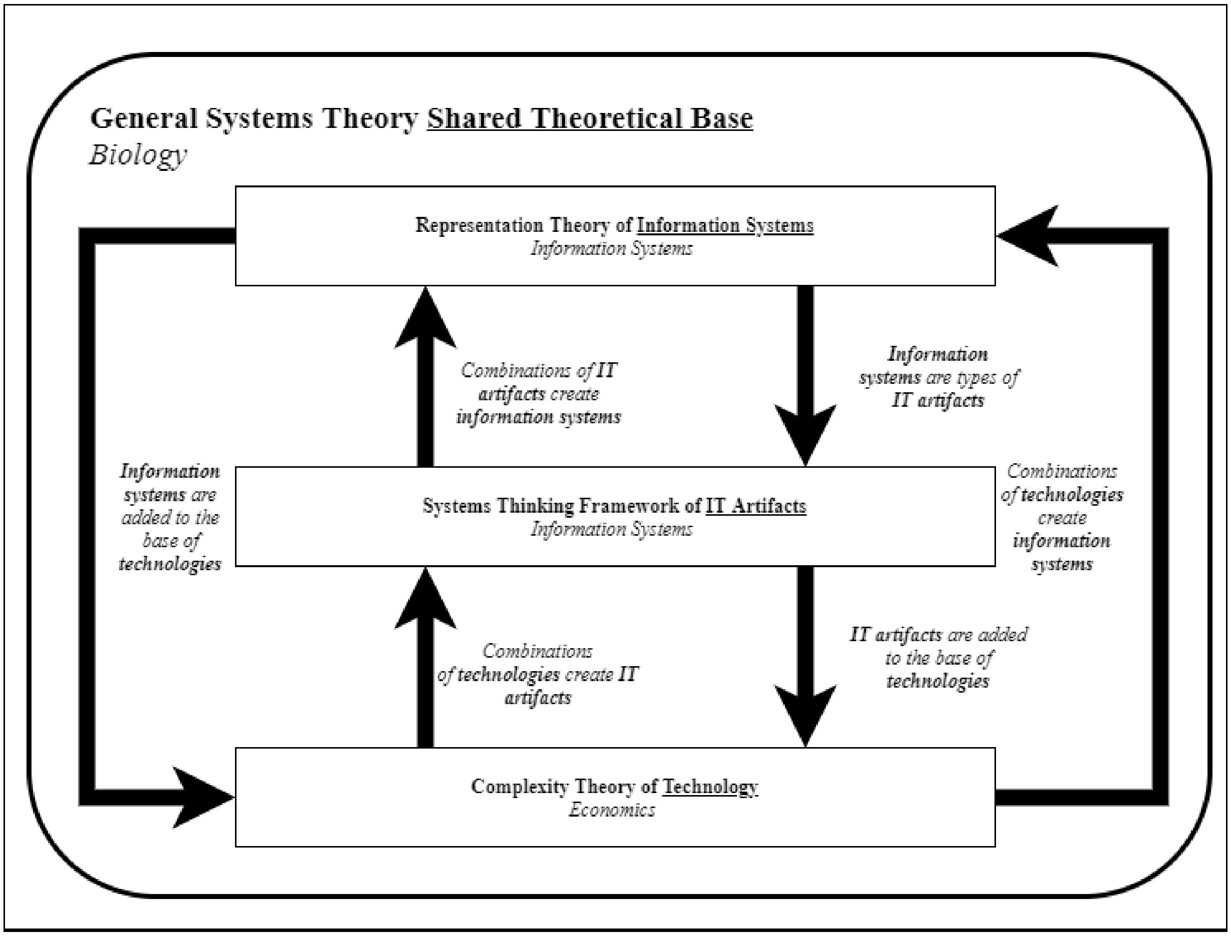

Figure 4: Theoretical Foundation Integration

Continuing with the theory of IT artifacts, an information system is considered a type of IT artifact. Specifically, it is an artifactual representation of real and digital world phenomena to perform information processing tasks (Recker et al., 2021; Wand & Weber, 1990, 1995). The physical structure of the information system is the IT artifact, but that structure is itself constructed from combinations of various IT artifacts and technologies (Arthur, 2009; Matook & Brown, 2017). Modern information systems, in particular, are often constructed by integrating various platforms and systems (Baskerville et al., 2022). The information system is a singular IT artifact, but the structure of that artifact is a combination of artifacts. Closing the theoretical loop, the theory of information systems and the theory of technology interact through combinations of technologies creating information systems, and information systems being added to the base of technologies by which new technologies can be created.

Defining a Legacy Information System – CS and IS Perspectives

Based on the established theoretical foundation above, I will now clearly define a legacy system as it is theorized in this research. First, it is essential to understand how the term is currently defined in the literature. Legacy systems have been researched in computing since the early 1980s (Smith, 1982), yet a clear, agreed-upon definition of a legacy system has yet to be reached. The most commonly cited definition comes from computer science: "large software systems that we don't know how to cope with but that are vital to our organization." (Bennett, 1995, p. 19). This definition is quite broad and could arguably be applied to other complex systems in organizations that one would not inherently consider legacy. Bennett (1995) further describes a typical legacy system as older, written in languages and techniques considered outdated, but were state of the art at the time of development. This is related to, but different from, the overall definition of legacy systems quoted above, but it has been the emphasis of most operational definitions of the term.

This notion of formerly adequate, now obsolete, technology is a common thread that runs throughout the definitions of legacy systems in the IS (Azadmanesh & Peak, 1995; Chirathamjaree, 2006; Mahapatra & Lai, 1998; Mallampalli & Karahanna, 2017; Tsai et al., 2022) and CS literature respectively (Carlson et al., 1996; Carvalho et al., 2019; Froscher et al., 1994; Rajlich & Adnapally, 1996; Wolfart et al., 2021; Yuming Zhou & Baowen Xu, 1999). Often, this obsolescence is framed within development and architecture paradigms such as procedural and object-oriented code (Datar & Schach, 1996; Masiero & Braga, 1999), flat and relational data models (Chirathamjaree, 2004, 2006), mainframe and client-server (Azadmanesh & Peak, 1995; Cimitile et al., 1997), monolithic and microservices (Carvalho et al., 2019; Wolfart et al., 2021), and locally hosted and cloud infrastructure (Gholami et al., 2017; S. Jain et al., 2016) to name a few examples.

It is clear that obsolete technology plays a role in how a legacy system is defined, but that obsolescence should be defined carefully. The framing of old design paradigms as inherently legacy runs into the issue of the modern design paradigm becoming associated with legacy systems. The CS literature, especially, has fallen into this trap. The legacy systems are modernized or replaced, “solving” the issue. However, within the next few years, papers begin emerging where the modernized system is now legacy. For example, by the late 1990s, researchers began grappling with adapting existing re-engineering techniques to systems built in an object-oriented paradigm despite it being a typical replacement architecture for legacy systems (Etzkorn et al., 1996; Etzkorn & Davis, 1997). From this emerges a subset of scholars that try to define a legacy system more clearly so that it is not subject to the whims of the constantly changing technology ecosystem.

In the CS literature, Edwards et al. (1999) challenge the assumption that legacy systems are inherently old and negative, emphasizing the business value legacy systems provide. This acknowledgment of the business value of a legacy system is not new (Bennett, 1995; Oca & Carver, 1998; Scandura, 1994; Weide et al., 1995), but the notion of a legacy system not necessarily being old is unique to this definition. They also posit that legacy systems include the broader social context in the definition, not just the technology, mirroring similar definitions emergent in the IS literature (Brooke, 2000; Kelly et al., 1999; Light et al., 1998). The authors further define a legacy system as “a group of interacting elements forming an entity where one or more elements impacts upon potential change” (H. M. Edwards et al., 1999, p. 14). While this solves the problem of not tying a definition to fickle technological trends, it returns to the issue of being vague. The definition could reasonably be applied to any complex system, technical or otherwise. The unique essence of what a legacy system is remains absent. This general definition shares commonality with perspectives in the IS literature, such as the notion that all systems are arguably legacy systems after implementation (Light, 2003). As well as the information infrastructures notion of the installed base and the existing socio-technical interactions that must be addressed during systems implementation (Hanseth & Lyytinen, 2010; Star & Ruhleder, 1996; Vestues & Rolland, 2021).

The late 1990s and early 2000s mark a similar focus on trying to clearly define a legacy system in the IS literature (Gibson et al., 1998; C. P. Holland et al., 1999; Kelly et al., 1999; Light et al., 1998; Light, 2003), firmly situated in the socio-technical roots of the discipline (Bostrom & Heinen, 1977a, 1977b; Sarker et al., 2019). Gibson et al. (1998) define legacy systems in terms of business strategic vision as systems that no longer have required functionalities for current and future business requirements, as well as the system itself being challenging to alter. The emphasis on business functionality (Brooke, 2002; Brooke & Ramage, 2001; Kelly et al., 1999; Pang, 2017) and the difficulty of changing the system (Bing Wu et al., 1997; Brodie & Stonebraker, 1995; de Kinderen & Kaczmarek-Heß, 2017; Light et al., 1998; Limaj et al., 2020; A. J. O’Callaghan, 1999) are common threads that run throughout both classic and modern definitions of legacy systems in the IS literature.

While many IS definitions are limited to the technical artifact itself and the business value of the artifact (Gibson et al., 1998; Mehrizi et al., 2019; Tsai et al., 2022; Warrell &

Stevens, 2003), other definitions take a more explicitly socio-technical approach to defining a legacy system. These definitions consider the social factors such as the people and business processes as well as the system they interact with to compose a legacy system (Brooke, 2002; Brooke & Ramage, 2001; Kelly et al., 1999; Light, 2003; Light et al., 1998; Soliman & Rinta-Kahila, 2019). Light (2003) further theorizes the importance of the characteristics of the legacy IS, interpretations of those characteristics, and temporal effects that influence interpretations and characteristics. Light et al. (1998) explicitly identify the technical artifact as legacy IT and the social components as business legacy, which combined make a legacy information system. However, it should be noted that the strictly artifactual view is generally more common in modern IS research (Limaj et al., 2020; Mallampalli et al., 2018; Mehrizi et al., 2019; Pang, 2017; Rinta-Kahila et al., 2023; Soliman & Rinta-Kahila, 2019; Tsai et al., 2022) with the legacy system interacting with the social system rather than encompassing the social system.

Defining a Legacy Information System – Theoretical Propositions

With this history of the term in mind, I will now outline my own definition of legacy systems that covers the main components mentioned previously and fits within the complexity, systems, and representation theory framework. I posit the following as a definition of legacy systems:

> *A formerly adequate incumbent information system perceived as insufficient through a combination of social and technical factors.*

Unique to this definition, I posit that legacy systems are a socio-technically constructed phenomenon. In the remainder of this section, I will justify this definition with clear propositions grounded in theory. I will also provide an example of a legacy system as I build up the propositions.

Proposition 1: The physical structure of a legacy system is constructed from combinations of technologies, potentially including other information systems.

Proposition 2: A legacy system is an IT artifact.

Proposition 1 is premised on the complexity theory of technology (Arthur, 2009). Since I posit that legacy systems are technical artifacts, they are subject to the combinatorial and evolutionary aspects of technological development (Arthur, 2009; Arthur & Polak, 2006). Proposition 2 delineates that a legacy system is computational and consists of some hardware and software combination (Goldkuhl, 2013b; Matook & Brown, 2017). An IT artifact is a specific type of technology within Arthur's (2009) overall theory of technology.

Proposition 3: A legacy system is a discrete technical artifact devoid of social context.

Proposition 4: A legacy system is an information system.

Proposition 3 is perhaps the most controversial point, but representation theory stipulates that information systems are technical artifacts that can be studied as unique entities divorced from their social context (Wand & Weber, 1990). It should be noted, though, that the system's structures have a social component, as they are technical representations of human perceptions of reality. Additionally, my research model looks at the impact of the IS on managerial decisions. It is divorced from the social context in the sense that the artifact is studied as a unique entity, but it can still *interact* with that social context and is engineered within that social context. Proposition 4 further narrows the definition of the artifact into an artifact that performs information processing functions per the stipulations of representation theory (Wand & Weber, 1990).

Proposition 5: The structures of a legacy system represent physical reality.

Proposition 6: The structures of a legacy system represent digital reality.

Proposition 7: The structures of a legacy system mediate physical and digital reality.

Proposition 5 is based on the assumption of representation theory that all IS must represent some real-world system within their deep structure, supported by the physical and surface structures of the system (Recker et al., 2019; Wand & Weber, 1995). Propositions 6 and 7 are based on recent extensions of representation theory to incorporate the notion of digital reality in the structures of an information system (Recker et al., 2021). Depending on their age, some legacy systems may have been developed in an era where they only had to represent physical, real-world systems as defined in classic representation theory (Wand & Weber, 1995). However, real-world systems also now consist of non-material computed objects that need to be represented or mediated between within the deep structure of an IS (Recker et al., 2021). Following the example, let us assume that the legacy system is an accounting system. This accounting system must represent the real-world ledger processes of the organization while also interfacing with other digital systems in the organizations embedded in those business processes.

Proposition 8: Legacy systems are incumbent systems.

Proposition 8 captures the temporal aspect of a legacy system (Light, 2003). While there is no pre-determined time threshold by which a system moves from non-legacy to legacy, the system must be implemented for some period of time before it can be considered legacy. This period is not immediately after implementation, making it distinct from other perspectives that suggest all implemented systems are legacy (H. M. Edwards et al., 1999; Light, 2003). Continuing with the example, let us assume that the legacy accounting system was implemented in the organization ten years ago and continues to operate.

Proposition 9: The "legacy" in legacy systems is socio-technically constructed.

Proposition 9 is the crux of my argument and a unique behavioral perspective in the literature on legacy systems. It draws from preliminary work on the importance of interpretations of system characteristics (Light, 2003) and narrative constructions about legacy systems in enterprise resource planning (ERP) implementations (Alvarez, 2000). Above I established that a legacy system is a tangible, technical artifact, which remains true. However, the system is attributed the legacy title by a social actor. Figure 5 provides a diagram of this socio-technical construction process. I will explicitly identify these social and technical factors and their interactions in the hypotheses development chapter.

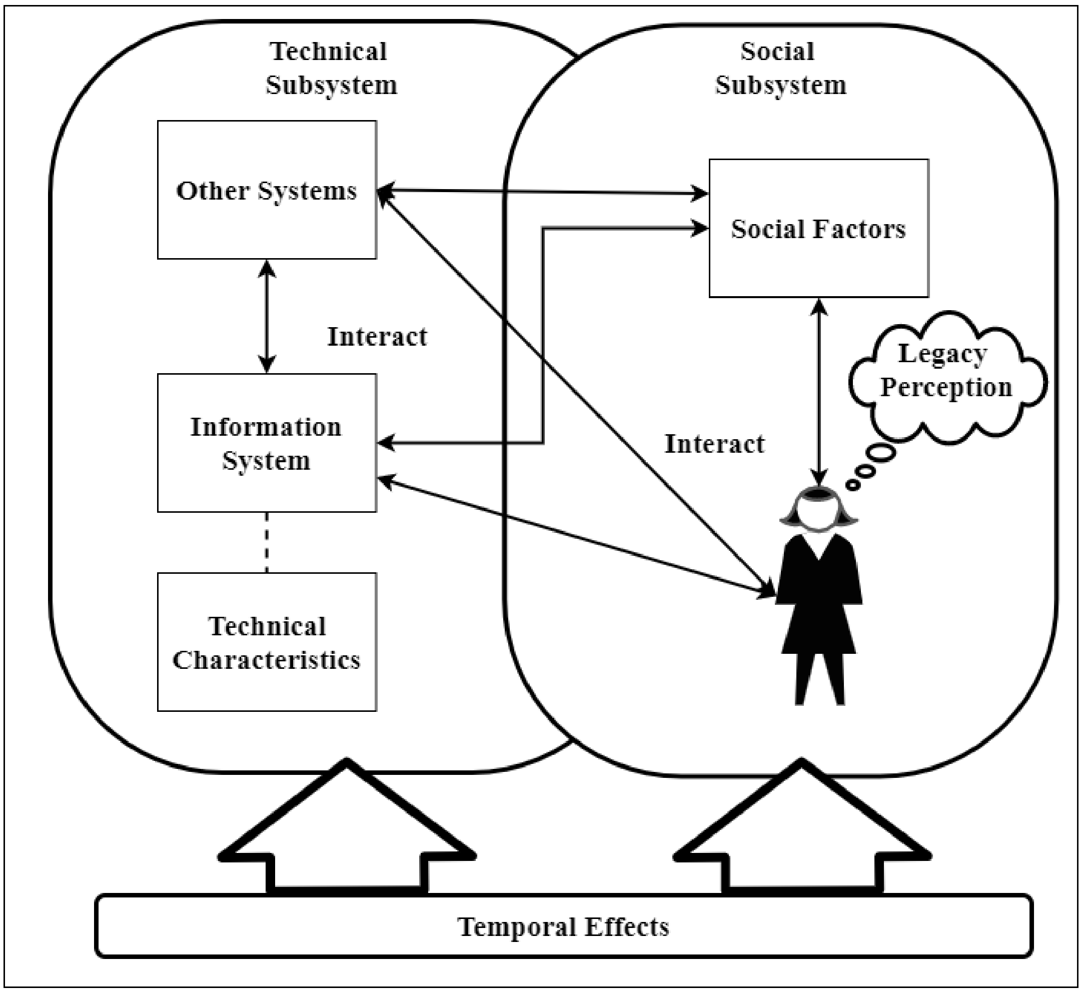

Figure 5: Socio-technical Construction Process

Continuing with the accounting system example, let us assume there exists a complex, degraded physical structure of the system. As a result of this structure, it is difficult to modify the system to accept inputs from other systems in the organization, as well as making modifications to the user interface. This lack of functionality has made the system insufficient for daily tasks, leading users to perceive it as legacy. Thus, the accounting system would be socio-technically constructed as a legacy system. As a counterexample, let us assume that the accounting system codebase features things such as application programming interfaces for system integrations, and developers have built wrappers to bypass existing non-user-friendly graphical interfaces. Now, despite the system's age and implementation using older technologies, there has been no impact on the daily tasks performed by the organization. In this case, the accounting system would be socio-technically constructed as *not* a legacy system.

A fair critique of this proposition is whether those system characteristics can be adequately measured divorced from the social actor. Collecting data about a system, such as complexity, degradation, and adaptability, necessarily requires perceptions of the system from a social actor. In some cases, "objective" secondary measures can be developed that measure characteristics without user perception of the artifact (e.g., Akhtyamov et al., 2018; Sarkar et al., 2007; Stachofsky et al., 2022; Tiwana, 2018). However, that is not always feasible or scalable in data collection contexts. I acknowledge this as a practical measurement challenge but remain convinced that, theoretically, proposition 9 remains true despite this challenge. Table 1 summarizes how each proposition relates to the theory it is derived from.

Proposition	Theoretical Foundation
P1, P2, P4	Complexity Theory of Technology -- (Arthur, 2009; Arthur & Polak, 2006)
P2, P4	Systems Thinking Framework of IT Artifacts -- (Goldkuhl, 2013a, 2013b; Matook & Brown, 2017)
P3, P4, P5, P6, P7	Representation Theory -- (Recker et al., 2021, 2021; Wand & Weber, 1990, 1995)
P8, P9	Legacy systems research on temporal effects, interpretations, and social constructions -- (Alvarez, 2000; Light, 2003)

Table 1: Propositions Summary

CHAPTER THREE: LITERATURE REVIEW

I focus on the IS literature on legacy systems for the literature review. Where relevant, the CS literature will also be discussed. Most of the overlap occurs in the technical areas of IS research, although even recent socio-technical works in the IS discipline (e.g., Rinta-Kahila et al., 2023) adopt conceptualizations of legacy systems from the software engineering literature (Bisbal et al., 1999). A full review of the CS literature is outside this dissertation's scope, but influential papers will be discussed throughout. There are some existing reviews in CS on legacy migration (Althani & Khaddaj, 2017; Bisbal et al., 1999), legacy web to mobile migration (Cajas et al., 2020) [1], and service-oriented architecture maintainability (Mishra et al., 2021). However, none of these reviews are comprehensive and remain an open opportunity for CS scholarship.

It should also be noted that this review does not include technical debt research. Technical debt is a closely related subject but is theoretically distinct from the literature on legacy systems (Alves et al., 2014; Holvitie et al., 2016). Legacy systems accumulate and create technical debt (Persson et al., 2023; Rinta-Kahila et al., 2023), but this is not unique to legacy systems. My search initially included the term, but ultimately, strictly technical debt papers were dropped after repeated instances of non-relevance to this work. However, papers that include both technical debt and legacy systems together are included in the review.

Review Methodology

Literature searching began on December 20th, 2021, and was periodically updated over the writing of this review. Paper counts and content included are accurate as of March 3rd, 2024. Journals were selected based on the Senior Scholar's List of Premier Journals in IS (*Senior*

[1] Not available in English at the time of this dissertation.

Scholars' List of Premier Journals, 2023). Additional journals from outside of the list were included to broaden the search. A search of conference publications was also included in this review to capture works in progress and research that did not move on to full journal publications. Relevant literature in citations that did not show up in these initial searches were also included in the review.

I searched the Association for Information Systems (AIS) eLibrary database with the following search terms: "Legacy System" OR "Legacy Systems" OR "Legacy Information System" OR "Legacy Software" OR "Legacy Hardware" OR "Legacy Code" OR "Legacy Program" OR "Legacy Information Systems" OR Obsolescence OR Obsolete OR deprecate. This search includes *Journal of the Association for Information Systems* and *MIS Quarterly* from the senior scholar's list, nine additional AIS journals, six AIS-affiliated journals, and many peer-reviewed conference proceedings. I also searched the same terms for the following journals not included in the AIS database: *Computers in Human Behavior, DATA BASE for Advances in Information Systems, Decision Support Systems, European Journal of Information Systems, Information & Management, Information and Organization, Information Systems Journal, Information Systems Research, International Journal of Information Management, Journal of Information Technology, Journal of MIS,* and *Journal of Strategic Information Systems*. In total, 218 papers were returned from the literature searches. After additional sorting and reading, this list was reduced to 94 relevant papers.

Literature Topics

Interest in legacy systems in the IS field was primarily sparked by the Y2K crisis (C. P. Holland et al., 1999), with the Communications for the Association of Information Systems publishing a special issue on the topic in 1999 (Coakes & Elliman, 1999; Giaglis, 1999; C. P.

Holland et al., 1999; Kavakli & Loucopoulos, 1999; Kelly et al., 1999; Lloyd et al., 1999; A. J. O'Callaghan, 1999; Randall et al., 1999). Much like the CS literature (e.g., A. Ahmad & Babar, 2014; Goeschka & Schranz, 2001; Wolfart et al., 2021), external technical trends (e.g., ERP, Cloud Technology) drive waves of interest in legacy systems research as innovations bump up against the constraints of extant technologies that were the innovations of the past. Rather than focus on individual technologies or events, I derive higher-level categories to describe each area of legacy research presented in Figure 6. The remainder of this chapter will discuss the literature organized around each category in this model.

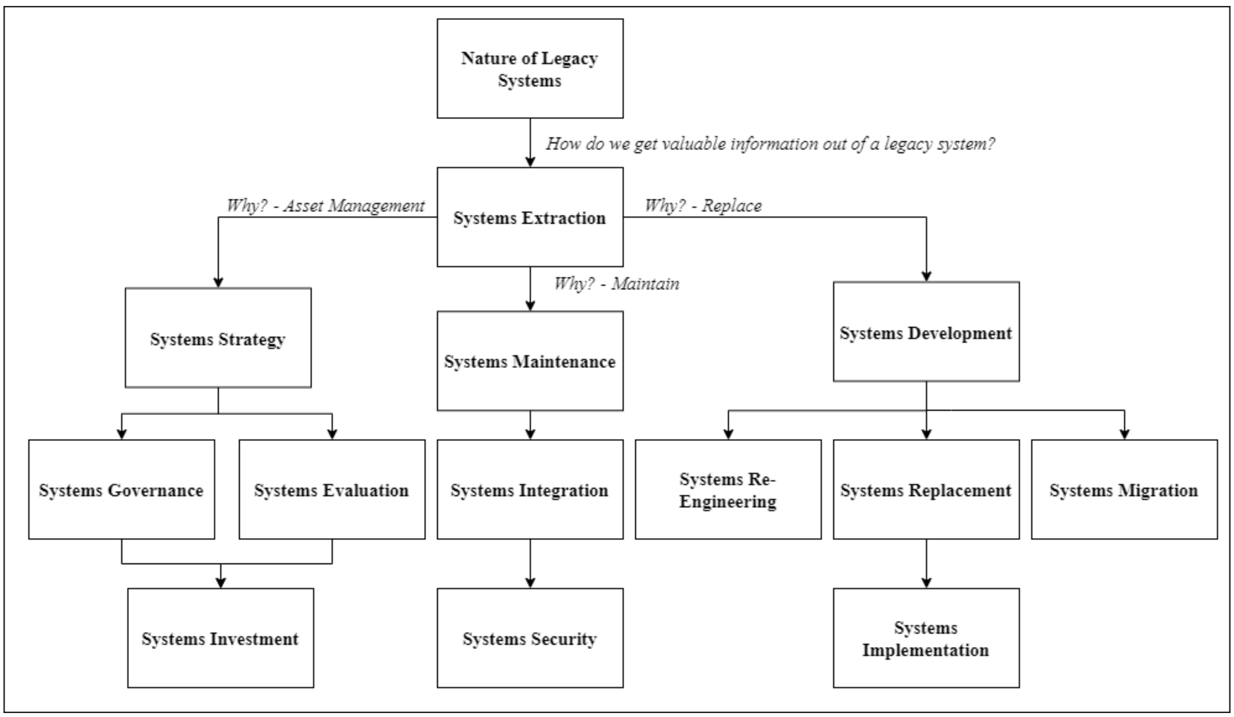

Figure 6: IS Legacy Systems Literature Map

This model starts with the category of *Nature of Legacy Systems*. This line of research is concerned with what a legacy system actually is. The second line of research is *Systems Extraction*, which concerns the processes by which value is extracted from a legacy system. This feeds into three main branches: *Systems Strategy*, which is concerned with managing legacy systems assets. Topics in this branch include *Systems Governance*, *Systems Evaluation*, and *Systems Investment*. The second branch is *Systems Maintenance,* primarily concerned with maintaining existing legacy systems. This also includes topics such as *Systems Integration* and *Systems Security*. The third branch is *Systems Development*, which focuses on replacing legacy systems. This includes topics such as *Systems Re-engineering*, *Systems Replacement*, *Systems Migration*, and *Systems Implementation*.

<u>Pre-IS Literature on Legacy Systems</u>

Before discussing the IS literature specifically, I will provide an overview of important CS works that predate the emergence of the legacy systems phenomena in IS (Azadmanesh & Peak, 1995; C. P. Holland et al., 1999). An early paper to discuss legacy systems in CS was published in the early 1980s (Smith, 1982), providing guidance to process control managers on replacing obsolete computing hardware in electrical systems. Despite originating in the CS literature, this paper also marks an early example of a management paper in legacy systems, discussing not only the technical aspects but also labor, vendor support, system availability, and expenditures, among other topics. These topics would be further explored in the IS literature in the context of enterprise systems over three decades later (Furneaux & Wade, 2017).

Another significant milestone in the literature was the application of virtualization to the domain of legacy systems, replicating a legacy system in a virtual environment to extend the life of the hardware testing system that used the software (Moorhead, 1993). Scandura (1994) marks early behavioral research in the legacy systems space, building code comprehension tools for visual representation to reduce the cognitive noise programmers face working on these systems. Bodeau (1994) marks an important milestone in security, in which their proposed systems-of-systems security engineering approach takes the role of securing legacy systems seriously as part of a more extensive system of organizational security practices.

Much of the technical literature is focused on lower-level concerns, but Antoniol et al. (1995) bring attention to legacy surface structures, namely the user interface. Their work provides an approach to migrate an application's user interface from character-based to graphical. Sneed (1995) marks an early example of economic analysis applied to the study of legacy systems, although quite practitioner-focused, proposing various portfolio and cost-benefit analysis methods for planning legacy system re-engineering projects. Most of the early CS literature emphasizes legacy systems being non-object oriented, aiming to re-engineer or migrate them into object-oriented systems. Etzkorn et al. (1996) is the first paper to grapple with the fact that object-oriented systems can also be legacy systems. This acknowledgment of how "new" paradigms eventually become "old" paradigms is an early example in the legacy CS literature of the constant churn and re-evaluation of migration tactics in light of innovations.

The final notable paper necessary for understanding the pre-IS legacy work is Bisbal et al.'s (1999) literature synthesis of legacy systems migration. They conceptualize their synthesis around coping strategies, identifying wrapping, maintenance, migration, and re-development, with each of these strategies having different levels of impact and number of changes on the legacy system. Bisbal et al.'s (1999) literature synthesis on legacy migration and Bennett's (1995) definition of a legacy system are two of the most influential papers in CS and IS on legacy systems.

<u>Literature Foundations: Systems Extraction and The Nature of Legacy Systems</u>

Shifting focus back to the IS literature, a small pocket of the literature focuses on the theoretical nature of legacy systems themselves, modeled as the starting point of Figure 6. Some key insights include noting that system age is not inherently problematic despite negative perceptions of legacy, legacy systems have business and technical dimensions, and continual modifications should prioritize maintainability to avoid structure degradation (Kelly et al., 1999). This literature also identifies the importance of temporal effects and interpretations (Light, 2003) and the dynamic nature of these systems (Kelly et al., 1999; Light, 2003; Rinta-Kahila et al., 2023). Preliminary empirical work in this area has also identified characteristics of legacy system artifacts as somewhat integrated, complex, and synchronous (Stachofsky, 2018) within a systems thinking lens (Matook & Brown, 2017).

Like the CS literature (e.g., Bennett, 1995; Carvalho et al., 2019; Cosentino et al., 2013; García-García et al., 2021), the IS literature recognizes that there is value embedded in the legacy systems which to be useful needs to be technically extracted in some way. This includes reusable programming components (Achee & Carver, 1995; Carvalho et al., 2019; Etzkorn & Davis, 1997; e.g., Quilici, 1995; Zhang et al., 2006), business logic (Cosentino et al., 2013; García-García et al., 2021; Ning et al., 1993; e.g., Petry, 1996) as some examples.

In Figure 6 this is the systems extraction topic. In the IS literature, three papers specifically focused on systems extraction. The first example is an economic paper showing how legacy systems can be a valuable source for data mining efficient routings for production (Jiao et al., 2007). The other two papers are technical in nature but use a similar data mining approach, demonstrating how heterogeneous legacy systems can be datamined to monitor business processes (Bhat & Goel, 2011) and model use case affordances and user behavior (Mesgari, 2018). Legacy systems can also be valuable for knowledge management functionalities (Stylianou & Savva, 2023). Together, the technical systems extraction and theoretical nature of legacy systems form the underpinnings of the IS literature that branch into other, more specific areas of study.

Reason for Extraction: Systems Strategy

The first branch connected to systems extraction in Figure 6 is systems strategy, which is further divided into systems evaluation, governance, investment, and risk management. This branch is primarily concerned with the strategic management of legacy systems as an IS asset based on the value extracted from the system. The first example of the strategy orientation in the IS literature comes from Gibson et al. (1998). They argue that misalignment between legacy systems and business strategy is due to the legacy system having an outdated internal business model that has become too difficult to change due to decades of modification (Gibson et al., 1998). In the parlance of representation theory, the deep structure of the system no longer accurately reflects the real-world system that the information system is meant to model as a function of inadequate physical structure (Wand & Weber, 1995). This perspective is similar to the findings that the flexibility of a legacy systems IT infrastructure has a direct influence on a firm’s capacity for information generation, information dissemination, and organizational responsiveness (Bhatt et al., 2010), a firm's overall level of business agility (van Oosterhout et al., 2006), opportunities for inter-organizational supply chain integration (Howard et al., 2004; Lu et al., 2006), business process change strategy (Light et al., 1998), and competitive advantage (Barnes et al., 2001).

The work of Brooke (2000, 2002; Brooke & Ramage, 2001) also marks important early milestones in the legacy systems strategy literature. Brooke (2000) argues that the literature on IS change had been overtly focused on technical issues rather than the human issues that emerge from technological change. In particular, they identify the importance of how information is viewed (perceptual or as a resource) as an extension to existing systems development frameworks at the time (Clegg et al., 1996). This would be further developed into the Software as a Business Asset (SABA) approach for legacy systems strategic decision-making (Brooke & Ramage, 2001), eventually culminating in an empirical validation of the usefulness of SABA in practice (Brooke, 2002). However, the most considerable contribution of this work to IS strategy remains the increased emphasis on socio-technical factors in legacy systems decision-making (Brooke, 2000) contrasting the techno-centric status quo (Brooke, 2002). This shares common ground with other early research highlighting the importance of mapping out internal and external legacy systems stakeholders when making legacy system decisions (Coakes & Elliman, 1999).

Another area of interest in legacy systems strategy is the evaluation of systems. In the context of mergers and acquisitions, business activity models have been used to evaluate what legacy systems should be supported after the merger (Orwig & Dean, 2007). Game theory modeling has also shown that systems like ERP are strategically more complex investments as all branches and departments must select the same system (O'Leary, 2000). The enterprise knowledge development framework was also developed to evaluate an organization's position in the face of rapid business process change (Kavakli & Loucopoulos, 1999). Similarly, identifying the obsolete knowledge within and around a legacy IS, which is necessary to prioritize systems maintenance decisions, has also been studied (Mehrizi et al., 2012; Shumaker et al., 2011). A managerial tension in this literature is that determining what knowledge is vital to preserve or necessary for middleware systems to reduce legacy dependency requires allocating additional resources to study and gain knowledge about the obsolete system (Mehrizi et al., 2012).

In the governance context, legacy systems have been found to limit process integration and data standardization, making adopting information governance practices and the existing ones more challenging (Tallon et al., 2013). Both topics feed into the literature on systems investment, the final destination for most strategic decision-making about legacy systems. Legacy system investment has been researched in the context of modeling software asset reuse (Vadapalli & Nazareth, 1998) and legacy software project risks (Warrell & Stevens, 2003). Empirical economic methods have also been used to show that in United States politics, there are higher investments in IT development and modernization of legacy systems when the presidency and Congress are controlled by the same party (Pang, 2017).

Reason for Extraction: Systems Maintenance

The second branch connected to systems extraction in Figure 6 is systems maintenance, which is further divided into systems integration and security. This branch is primarily concerned with maintaining an organization's legacy systems. The goal is to keep these systems updated with the changing socio-technical environment around them rather than replacing the system. This branch of the literature also marks the earliest example of the legacy systems concept in the IS literature (Azadmanesh & Peak, 1995), arguing the continued relevance of legacy mainframe computing functions even as end-user computing became more prominent. The theme of addressing the legacy technology in use, including the organizational practices that are attached to it (Randall et al., 1999), shares much in common with the information infrastructure literature’s emphasis on the installed base (Hanseth & Lyytinen, 2010; Star & Ruhleder, 1996; Vestues & Rolland, 2021).

As a general research topic, systems maintenance precedes legacy systems maintenance literature (e.g., Bateman & Wetherbe, 1978; Dekleva, 1992; C. Edwards, 1984; Moreton, 1990).

The key difference is that in the broader systems maintenance literature, scholars are not necessarily focusing on a legacy system; instead, they focus more on any system in a post-implementation state. There is an additional emphasis on the entire socio-technical system in which the software and hardware are embedded (C. Edwards, 1984). One example from the social system is adequately staffing an organization with individuals who understand the business environment and the technical system for effective system maintenance (Taylor et al., 1997). The team of people that maintain software and the tools they use are both essential aspects of the software maintenance process (Banker et al., 1998).

Some research has focused on formalizing software maintenance processes following a path from change management, design change, testing, and system release or integration (Moreton, 1990). However, this is not the only software maintenance process, and individual developer differences and preferences can strongly influence which methodologies for software maintenance are used (Edberg et al., 2012). Automating aspects of these processes has also been shown to make the software evolution process more manageable and sustainable as software artifacts grow in functionality (Barry et al., 2007). There is also an acknowledgment that not all maintenance is equal (Kemerer & Slaughter, 1997), with certain aspects of the artifact determining maintenance tasks. In general, older and larger systems are restructured and upgraded more frequently, highly complex systems are repaired often and are generally older and more prominent, and systems that are enhanced generally consist of strategic functionality in the organization (Kemerer & Slaughter, 1997). The level of maintenance efforts invested into a system is further explained by the overall organization portfolio of systems, size, age, and life expectancy of a system (Swanson & Dans, 2000).

Additional research identifies the importance of designing new systems with maintenance in mind to lower future maintenance effort and costs (Dennis et al., 2014) and project control balancing to promote technical debt remediation in systems maintenance projects (Ramasubbu & Kemerer, 2021). Software development practices impact maintenance performance (Banker et al., 1998; Dennis et al., 2014). For example, if pre-built packaged software is used instead of automatic code generators, system complexity is reduced, and the level of effort to enhance the software is also reduced (Banker et al., 1998). This is similar to findings in the recent legacy system literature suggesting that system customizations often generate more technical debt in a system (Rinta-Kahila et al., 2023).

A significant portion of the systems maintenance literature has adopted an economic lens. Some of the earliest work focused on developing economic cost models for the maintenance of computing equipment (Bateman & Wetherbe, 1978). Additional work has developed mathematical decision models to identify when to upgrade a system (Krishnan et al., 2004). Software maintenance has also been found to exhibit economies of scale, whereby batching together more minor modifications as a planned release can reduce maintenance costs significantly (Banker & Slaughter, 1997).

One area of interest in the legacy systems maintenance literature is how legacy systems become embedded in organizations. In the banking sector, non-technical reasons include a lack of human capital to replace the systems, regulatory burdens, and legacy organizational culture associated with the system (Limaj et al., 2020). Another source of legacy systems includes systems initially intended for personal use that become critical to a process, such as complex spreadsheets, which eventually are inherited by the next person in the position (Grossman et al., 2007). In the public sector, findings suggest that the bureaucratic processes and funding structures make it difficult for organizations to break out of maintenance logics even when a legacy system should be replaced (De Marco & Sorrentino, 2007). There is also critique of maintenance logics more generally, with some research suggesting that changes in the external technical environment can dramatically re-shape practices such that a system becomes legacy, even if the underlying technology has been upgraded and maintained such that the technical components are not obsolete (Chen, 2010).

Systems integration is one of the most actively researched topics within the systems maintenance literature. This literature focuses on how legacy systems are integrated with modern technologies and business processes in organizations. The technical portion of this literature consists primarily of designs for integrating systems. For example, research on development of middleware systems that integrate legacy systems to modern technologies using object-oriented models (Brook et al., 2000; Vergara et al., 2007), standardization layers (Shankaranarayan et al., 2000), query wrappers (Chirathamjaree, 2004, 2006), data wrappers (Crowley et al., 2013), semantic layer models (Buchmann & Karagiannis, 2016; Zalhan et al., 2019), application gateways (Sasso & Forcolin, 2009), and service-oriented architectures (Chou & Seng, 2009). Within the technical literature it is major external technical forces such as the internet (Brook et al., 2000; Chou & Seng, 2009; Shankaranarayan et al., 2000; Vergara et al., 2007), data modeling standards (Chirathamjaree, 2004, 2006), big data (Crowley et al., 2013; Zalhan et al., 2019), and smart devices (Brandt et al., 2018; Hauser & Gã, 2017) that often drive the interest and business need for interfacing with legacy systems (Thummadi et al., 2017).

Beyond middleware, technical systems integration researchers have also studied the mapping of data across legacy database schemas (Evermann, 2012) and the unique challenges of legacy components in the context of cyber-physical systems (Brandt et al., 2018; Hauser & Gã, 2017). In the context of cyber-physical, how mutable or immutable the legacy components of the system are essential, both the physical and the digital (Hauser & Gã, 2017). It can also be the case that an IT artifact can enhance a previously non-digital legacy component to produce new affordances, although the artifact's deep structure must represent that component's physical restrictions (Brandt et al., 2018). This literature stream aligns with recent theorization in the representation theory literature, arguing for more focus on the interaction between physical and digital (Recker et al., 2021) in light of the ontological reversal (Baskerville et al., 2020).

As a complement to the technical literature, the managerial lens is also applied to the topic of systems integration. This literature includes guidelines for obtaining data requirements from legacy systems (Aiken et al., 1999) and methods for integrating legacy systems with ERP systems in organizational processes (Li & Lau, 2005; Sharif et al., 2005). This literature also provides theoretical (Sulong et al., 2011) and empirical (Augusto et al., 2009) support for service-oriented architectures as methods to reuse and extend the life of legacy system software components. Artificial intelligence has also been proposed as a way to enrich legacy systems, predicated on system characteristics, user performance and training, task support and service features, and acceptance and adoption (Frick et al., 2020). At a broader level, managers have indicated legacy systems as essential but not critical due to coping strategies like virtualization, service migration, and middleware (Mattord & Bandyopadhyay, 2008). Public sector information systems management is uniquely challenging, with siloed government departments creating technical debt and high costs when integrating with legacy systems architecture (Persson et al., 2023).

The last topic in the systems maintenance branch is systems security. For the maintained systems, the goal is to keep them safe from new vulnerabilities even as vendor support for these systems drops over time. Research in this area has developed a universal markup language (UML) design methodology for modeling threats to legacy systems (Ingalsbe et al., 2008) and artifacts that emulate vulnerable legacy software to confuse attackers (Araujo et al., 2021). Empirical work has also tested the security by antiquity argument in which legacy systems are considered more difficult to exploit due to lack of documentation. This argument has been countered by results that show increased IT modernization spending and migration to the cloud decreased security incidents for government agencies (Pang & Tanriverdi, 2022).

Reason for Extraction: Systems Development

The final branch connected to systems extraction in Figure 6 is systems development, further divided into systems re-engineering, migration, systems replacement, and systems implementation. Research in this area is primarily concerned with how to replace legacy systems and the unique challenges of a systems development project in the legacy systems context. This is the most active area of current legacy systems research (e.g., Mehrizi et al., 2022; Rinta-Kahila et al., 2023; Tsai et al., 2022; Vestues & Rolland, 2021) in IS and also one of the earliest areas of study (Mahapatra & Lai, 1998).

In many ways, the legacy systems development findings mirror the fundamental systems analysis type issues that IS scholars have studied since the field's earliest days (Blumenthal, 1969; Dearden & McFarlan, 1966; Langefors, 1968; Mumford, 1965). Findings highlight the importance of business processes and IS being designed simultaneously to enable organizational change (Giaglis, 1999) and identifying the problems with a legacy process before moving forward with new systems development (Vidgen et al., 2017). More unique to the legacy IS systems development literature is design approaches that explicitly account for legacy systems (C. Holland & Light, 1999).

This includes using standard package suites such as ERP systems with minimal customizations from vendors. The best-of-breed approach implements integrations between the best offerings of various software vendors. The legacy systems forever approach wherein continual maintenance and development of a legacy system is conducted indefinitely rather than replaced. The new lamps for old approach where legacy systems are replaced in a way that their functionality and strategic alignment are mirrored in the replacement system. The ring fence approach where legacy core systems are maintained with other new systems being implemented to interface with them. Alternatively, in some cases, a complete re-engineering of legacy code to write an effectively new system entirely (C. Holland & Light, 1999). Although these development approaches for legacy systems have been identified, the empirical literature exploring their efficacy remains sparse. Although some research would indicate that commercial off the shelf software with minimal customizations is generally preferable to avoid architectural debt in a legacy system (Rinta-Kahila et al., 2023).

Systems re-engineering is one of the topics in legacy systems development focused on the complete re-writing of legacy system code. This topic area is relatively small, as the work is generally technical and better suited for CS journals. CS scholarship has noted that as the complexity and volume of technical artifacts increase, many reverse-engineering projects are not cost-effective, and the reverse-engineered code that is replaced will eventually need to be reverse-engineered again (Weide et al., 1995). As such, work heavily focused on reverse engineering of entire systems has shifted to more maintenance focuses (Baxter & Mehlich, 1997).

IS findings suggest that object-oriented code modules can mitigate poor software quality in legacy systems (Bevan, 2000). Software re-engineering patterns are also proposed as a way to address lock-in to inefficient business processes caused by legacy systems. Proposed technical design patterns include migrating functions and data through a divide and modernize approach, changing the system interfaces with wrappers and middleware, and modularity of phased processing where the system data is decoupled from other systems it receives and transmits data to (Lloyd et al., 1999). Proposed managerial re-engineering communication patterns include the war room approach, where all internal and external parties agree on deliverables and time scales, and the workshop approach, where long-term partnerships are built with suppliers that support a re-engineering project (Lloyd et al., 1999). Similar to the legacy system design approaches (C. Holland & Light, 1999), these legacy re-engineering design patterns remain largely unexplored in the legacy systems literature.

Another topic in legacy systems development is systems migration. Systems migration is an approach to legacy systems in which as much of the original code is preserved as possible, re-platformed rather than re-engineered or replaced (Bisbal et al., 1999). The technical side of this research has focused on migrating systems from one development paradigm to a newer paradigm such as object-oriented (A. O'Callaghan, 1998), service-oriented architecture (de Kinderen & Kaczmarek-Heß, 2017), and cloud computing (Fahmideh et al., 2019; Gholami et al., 2017). An essential contribution of these technical migration papers absent in the parallel literature in CS is the importance of organizational processes and business cases as essential for a successful migration, not just the technical changes (de Kinderen & Kaczmarek-Heß, 2017; Fahmideh et al., 2019; Gholami et al., 2017; A. J. O'Callaghan, 1999). This literature also contributes to conceptual modeling research with explicit process meta-modeling (Fahmideh et al., 2019; Gholami et al., 2017), design patterns (A. J. O'Callaghan, 1999), and modeling design language (de Kinderen & Kaczmarek-Heß, 2017) allowing for accurate representations of the legacy systems migration process.

Complementing the technical literature on migration is also behavioral and managerial literature. From a behavioral perspective, evidence suggests that individual risk perceptions reduce the likelihood of an organization migrating a legacy system from a closed to an open platform (Shim et al., 2009). Public sector legacy systems are also studied, with case studies suggesting that the same migration issues for general systems development exist in the public sector case, with the additional complexity that comes with balancing existing public-facing functions and adopting new system functions (Pilemalm et al., 2013). For cloud migration, a total cost of ownership framework has been developed as a tool for managers to assess the viability of cloud investment (Ramchand et al., 2018). Migration has also been studied from an information infrastructures perspective, highlighting the importance of decoupling applications from technical infrastructure, modularizing and platformizing applications, re-coupling of organization processes to newly decoupled systems, and the potential increased generativity that can emerge from a system if these actions are conducted successfully (Vestues & Knut, 2019; Vestues & Rolland, 2021).

The last two topic areas in the legacy systems development branch are systems replacement and the closely related systems implementation. Systems replacement refers to the processes related to replacing a legacy system with a new system. Systems implementation is more specifically concerned with how that new system is implemented, given the effects of the previous legacy system on the organization. Of the IS literature branches, this is the least technical, primarily focused on behavioral and managerial research.

One topic in legacy systems replacement is replacement strategy. Research in this area has shown that one method to promote ERP adoption is the social construction of myths about the legacy system dying (Alvarez, 2000). Representing the system as unintegrated and inaccessible and also ridiculing the system and support staff are all tactics that individuals in an organization may use to promote a new system (Alvarez, 2000). Replacing a system requires a simultaneous legitimization of the new system and a marginalization of the legacy system (Mehrizi et al., 2019). A method to evaluate ERP use maturity is also developed in which managing legacy systems and starting ERP projects mark the beginning stage of ERP maturity (C. P. Holland et al., 2000). Legacy systems have also been theorized as a driver of the lack of digital platforms in Europe, compared to countries like China that could start from a greenfield development approach (Hermes et al., 2020).

Regarding the replacement approach, some research would suggest that a radical approach to substitute an entire legacy system rather than a gradual replacement is a more efficient strategy for organizations (Madlberger, 2012; Rinta-Kahila et al., 2023). This is further supported by the fact that replacement projects can get caught in a middle state where both old and new architectures, as well as integrations between the two, adding complexity and taking resources away from data migration and process change concerns (Rinta-Kahila, 2018), although whether this middle state can be avoided with more complex system replacement projects is unclear.

Design is also critical for a successful replacement team, with cross-functional teams, operational knowledge from end-users, external experts, and adequate expertise in the core replacement project team, all critical success factors in a replacement project (Tsai et al., 2022). Those teams often experience task conflict due to IT resource constraints when designing legacy replacement systems. Team conflict resolution strategies such as compromising and collaboration can resolve design goal incongruences in those teams and develop a shared understanding of necessary replacement system development tasks (Tsai et al., 2023).

The other main topic in legacy systems replacement is the discontinuance process and issues within that process. Not all discontinuance literature is legacy systems related per se, but often, the system being discontinued is a legacy system. Furneaux and Wade's (2010, 2011) theory of organizational discontinuance underlies much of this literature. Furneaux and Wade (2011) identify change forces: system performance shortcomings, organizational initiatives, and environmental changes as positive influences on discontinuance intentions and continuance inertia and system investment, system embeddedness, and institutional pressures as negative influences on discontinuance intentions at the end of a systems life. Soliman and Rinta-Kahilia (2019) further refine the notion of discontinuance by theorizing five different types of discontinuance at different stages of the system's lifecycle. Discontinuance at the exposure stage is called rejection, discontinuance at the adoption phase is regressive discontinuance, discontinuance during the continued use phase is either quitting or temporary discontinuance, and at the end of the IS, use lifecycle replacement with a new system. For legacy systems, the discontinuance type of interest is replacement.

Much attention has been given to why legacy systems resist discontinuation. The escalation of commitment perspective theorizes that goal incongruence, information asymmetry, information ambiguity, side bets, and institutionalization are all drivers of legacy systems not being replaced, although this has not been tested empirically at this time (Mallampalli & Karahanna, 2017). Empirical results identify the importance of replacement risk and support availability as inhibitors of replacement intention, and system capability shortcomings as a driver of replacement intention (Furneaux & Wade, 2017). System complexity, institutional norms, and system investment are also related to these variables but do not directly affect replacement intention (Furneaux & Wade, 2017).

Organizational discontinuance processes can also lead to more technical debt acquisition, particularly in cases where organizations undergo a staggered replacement where the legacy system continues to operate at some level in the organization (Rinta-Kahila et al., 2023). This additional technical debt, usually embedded in system architecture, inhibits the digital options (Rolland et al., 2018) of the firm to replace the legacy system driven through path-dependent interactions of both social and technical inertia in the firm (Rinta-Kahila et al., 2023). At the individual level, results indicate legacy habits are multi-faceted, inhibiting users by keeping them attached to a legacy system serving as a bridge to work with the new system. They may also deter legacy system use due to dissatisfaction with legacy habits (Mehrizi et al., 2022). Impacts are also different depending on what level of the firm is examined. For example, a global project in an organization to replace a legacy system can trigger social inertia in localized pockets of a firm leading to additional customizations of the new system and integrations with the legacy system to appease local stakeholders. This social inertia often drives architecture changes that produce technical debt, further limiting options and entrenching the legacy system in the organization (Rinta-Kahila et al., 2023).

The process of discontinuing a legacy system from a vendor's perspective has also been studied empirically. In this context, four iterative phases of discontinuance are identified: realization, where the legacy IS is studied but remains the dominant IS, reversion where a legacy IS has further development to meet new requirements, handover where the momentum of the legacy IS is used to push towards the new IS, and marginalization where the development and use of the legacy IS is reduced (Mehrizi et al., 2019). Organizations also must give additional attention and resources to entirely discontinue a system (Mehrizi et al., 2019), similar to how determining what knowledge is essential to preserve to reduce legacy dependency requires allocating additional resources to studying a legacy system (Mehrizi et al., 2012).

The final topic in the systems development branch is systems implementation. This literature has identified training teams to transition users between the legacy and new system (Mahapatra & Lai, 1998), focusing on IT strategy, not just technical drivers (Zach, 2011), and knowledge of the legacy system data (Tona et al., 2012) as factors in system implementation success. Various research has also shown that users compare the new system with the legacy system (Mallampalli et al., 2018; Ng & Tan, 2004; Pan et al., 2001; Zach, 2011), which influences their benchmarking, habits and lens by which they process change (Pan et al., 2001). One potential way to address this is to design interfaces in the new system to match the legacy system design to increase acceptance (Pan et al., 2001). Research has also shown that user legacy system expertise can lead to actualizing affordances in the new IS that otherwise would not be discovered (Mallampalli et al., 2018). Although, habit and inertia with the legacy system may inhibit user acceptance of the new system (Polites & Karahanna, 2012).

One barrier system implementations face is symbolic attachment to legacy systems. IT support staff may take pride in maintaining the legacy system and its unique functionality if it is in-house developed while mistrusting an off-the-shelf software solution to meet the requirements currently met by the system (Ng & Tan, 2004). Other research outside the organizational context has found that fear of obsolescence is a barrier to whether an individual will adopt a new technology in the first place (Venkatesh & Brown, 2001). There are a few examples of research also exploring implementations where a legacy system is not entirely replaced. For example, firms that are satisfied with existing legacy systems still hold positive assessments of new ERP systems, implying the two can co-exist (Ifinedo, 2014). Some features of newer systems do not entirely capture the scope of legacy system functionality, so they are not necessarily superior options to legacy systems (Ofoegbu et al., 2011). In some cases, the legacy system may be replaced for political reasons rather than necessity, leading to end users developing workarounds to continue using the legacy system even after coercion from upper management (Bob-Jones et al., 2008). In extreme cases, user resistance may be so strong that the legacy system is re-implemented entirely (Grainger et al., 2009).

Literature Trends

Table 2 summarizes the IS literature review with citations for each paper reviewed and categorized for this dissertation. The Topic column refers to the topic from Figure 6 that a paper is assigned to. The Papers column includes the citations for each of the topics. Each paper is assigned only one topic. Systems integration was the most common topic with 22 papers, followed by systems replacement with 13 papers, systems implementation and systems strategy with 11 papers each, and systems migration with nine papers. Less common topics include systems maintenance with six papers, systems evaluation with five papers, systems extraction with four papers, nature of legacy systems, systems development, systems investment, and systems security with three papers, systems re-engineering with two papers, and systems governance with only one paper in the sample.

Topic	Papers
Nature of Legacy Systems *Papers:* **3**	(Kelly et al., 1999; Light, 2003; Stachofsky, 2018)
Systems Development *Papers:* **3**	(Giaglis, 1999; C. Holland & Light, 1999; Vidgen et al., 2017)
Systems Evaluation *Papers:* **5**	(Kavakli & Loucopoulos, 1999; Mehrizi et al., 2012; O'Leary, 2000; Orwig & Dean, 2007; Shumaker et al., 2011)
Systems Extraction *Papers:* **4**	(Bhat & Goel, 2011; Jiao et al., 2007; Mesgari, 2018; Stylianou & Savva, 2023)
Systems Governance *Papers:* **1**	(Tallon et al., 2013)
Systems Implementation *Papers:* **11**	(Bob-Jones et al., 2008; Grainger et al., 2009; Ifinedo, 2014; Mahapatra & Lai, 1998; Mallampalli et al., 2018; Ng & Tan, 2004; Ofoegbu et al., 2011; Pan et al., 2001; Tona et al., 2012; Venkatesh & Brown, 2001; Zach, 2011)
Systems Integration *Papers:* **22**	(Aiken et al., 1999; Augusto et al., 2009; Brandt et al., 2018; Brook et al., 2000; Buchmann & Karagiannis, 2016; Chirathamjaree, 2004, 2005; Chou & Seng, 2009; Crowley et al., 2013; Evermann, 2012; Frick et al., 2020; Hauser & Gã, 2017; Li & Lau, 2005; Mattord & Bandyopadhyay, 2008; Persson et al., 2023; Sasso & Forcolin, 2009; Shankaranarayan et al., 2000; Sharif et al., 2005; Sulong et al., 2011; Thummadi et al., 2017; Vergara et al., 2007; Zalhan et al., 2019)
Systems Investment *Papers:* **3**	(Pang, 2017; Vadapalli & Nazareth, 1998; Warrell & Stevens, 2003)
Systems Maintenance *Papers:* **6**	(Azadmanesh & Peak, 1995; Chen, 2010; De Marco & Sorrentino, 2007; Grossman et al., 2007; Limaj et al., 2020; Randall et al., 1999)
Systems Migration *Papers:* **9**	(de Kinderen & Kaczmarek-Heß, 2017; Fahmideh et al., 2019; Gholami et al., 2017; A. J. O'Callaghan, 1999; Pilemalm et al., 2013; Ramchand et al., 2018; Shim et al., 2009; Vestues & Knut, 2019; Vestues & Rolland, 2021)
Systems Re-engineering *Papers:* **2**	(Bevan, 2000; Lloyd et al., 1999)
Systems Replacement *Papers:* **13**	(Alvarez, 2000; Furneaux & Wade, 2017; Hermes et al., 2020; C. P. Holland et al., 2000; Madlberger, 2012; Mallampalli & Karahanna, 2017; Mehrizi et al., 2019, 2022; Rinta-Kahila, 2018; Rinta-Kahila et al., 2023; Tsai et al., 2022, 2023)
Systems Security *Papers:* **3**	(Araujo et al., 2021; Ingalsbe et al., 2008; Pang & Tanriverdi, 2022)
Systems Strategy *Papers:* **11**	(Barnes et al., 2001; Bhatt et al., 2010; Brooke, 2000, 2002; Brooke & Ramage, 2001; Coakes & Elliman, 1999; Gibson et al., 1998; C. P. Holland et al., 1999; Howard et al., 2004; Light et al., 1998; Lu et al., 2006; van Oosterhout et al., 2006)

Table 2: IS Literature Summary

Table 3 summarizes the theories and methods that were used in each paper. Papers were categorized by whether they had a Behavioral, Economic, Management, or Technical orientation. Papers can only be categorized in one orientation. Theories and methods used are grouped by the orientation of the papers in the table. Numbers in the parentheses refer to the number of times a theory or method appeared in the literature for its assigned group. N/A refers to any papers that do not explicitly identify a theory. Methods marked as Other generally refers to papers that do not have an empirical component and discuss legacy information systems issues conceptually.

Orientation	**Theories**	**Methods**
Behavioral *Papers:* **12**	Anchoring and Affordances (1) Conflict Survival Theory (1) Contingency Theory (1) DeLone and McLean IS Success Model (1) Escalation of Commitment (1) Information Elaboration Theory (1) Protection Motivation Theory (1) Risky Decision-Making (1) Social Constructionism (1) Socio-technical Conditions (1) Symbolic Interactionism (1) Theory of Planned Behavior (1)	Case Study (3) Ethnography (1) Mixed Methods (1) Proposal (1) Survey (5) Survey, Coding Interviews (1)
Economic *Papers:* **4**	Business Agility (1) Game Theory (1) N/A (1) Political Influence (1)	Data Mining (1) Mathematical Modeling (1) Panel Regression (1) Survey and Interviews (1)
Management *Papers:* **56**	Actor Network Theory (1) Capability Maturity Model (1) Competitive Advantage of Nations (1) Complexity Theory (1) Cultivation (1) Enterprise Knowledge Management (1) Information Infrastructures (2) IT Governance (1) IT Substitution (1) Multi-Agent Systems (1) N/A (33)	Case Study (28) Case Workshop (2) Cross-case Analysis (6) Group Case Discussion (1) Interviews (4) Mixed Methods (1) Other (9) Panel Regression (1) Survey (3) Theory Development (1) UML Design (1)

	Organizational Learning (1) Organizational Mythmaking (1) Organizational Path Theory (1) Organizational Scenarios (3) Organizational Unlearning (1) Practice Theory (1) Punctuated Socio-technical Information Systems Change Model (2) Resource-based View (1) Routine Activity Theory (1) Scaling Agility (1) Service-Oriented Architecture (2)	
Technical *Papers:* **23**	Affordances (1) Deception Steering (1) Linked Data (1) Metamodeling (2) Model-Driven Development (1) N/A (10) Object Oriented Design (3) Process Modeling (1) Semantic Stream Processing (1) Similarity as Interactive Activation and Mapping (1) Systems Thinking (1)	Case Study (2) Data Mining (2) Design (15) Design and Case Study (1) Other (2) Survey (1)

Table 3: IS Literature Theories and Methods

From this table, management-oriented legacy systems papers are the most common in IS research, with 56 papers. Technical is the second most active, with 23 papers and behavioral with 12 papers. Economic IS papers are the least common, with only four papers throughout the entire literature review. Theory and method usage is quite diverse, with no dominant overall paradigm. Case studies were the most common method, especially in management. Surveys were most common in behavioral research, and design was the most common method in technical research. There was no dominant methodology in the economic group. Forty-two of the papers, mostly in management and technical, do not have any identifiable theory at all. This may be partially explained by the more practical orientation of many of these legacy systems research studies.

While the amount of behavioral research is relatively small, that literature grouping strongly emphasizes theory more than other orientations. This dissertation would be classified as behavioral research under the topic of the Nature of Legacy Systems. This research engages significantly with the technical legacy systems literature but by adapting that literature to a behavioral context.

In some ways, the theory and methodological diversity of this literature are strengths. There are many different lenses being applied to the problem that are producing different insights. However, this also leaves the literature fragmented. The management and behavioral literature interact with each other to an extent, but the technical literature operates largely in isolation. This dissertation is partly an attempt to rectify this by integrating insights from the technical, managerial, and behavioral literature streams on legacy systems. One way I do this is by developing scales that can be used to measure IT artifact characteristics in the context of behavioral research. I think this is especially important as much of the modern IS literature on legacy systems accepts the technical view of legacy systems as artifacts (Limaj et al., 2020; Mallampalli et al., 2018; Mehrizi et al., 2019; Pang, 2017; Rinta-Kahila et al., 2023; Soliman & Rinta-Kahila, 2019; Tsai et al., 2022). If this view is accepted, the literature must have ways to measure and theorize about the artifact rather than just discussing the legacy system at a surface level or treating it as a context.

An interesting trend in the literature is the prevalence of case studies and other qualitative research methods. I think there are two potential factors here that are influencing this trend. The first is that legacy systems are incredibly complex and tied into nearly all aspects of an organization. Arguably, studying structures this complex necessitates deep, careful study of single cases with minimal generalization to other organizations whose legacy systems emerged under completely different conditions. I also believe that the largely atheoretical and disjoint literature has resulted in a lack of cumulative tradition from which to build. I see these qualitative works breaking necessary theoretical ground in many areas of legacy systems due to a general lack of theory development preceding the study. However, few papers pick up where those papers left off to continue exploring a given subtopic, leaving an unclear consensus on various aspects of legacy systems down to even the definition.

I do not say any of this to suggest that this qualitative work is not valuable. In fact, it is some of the best and most thorough theoretical scholarship the legacy systems literature offers. However, I think that to develop this literature further, finding methods that abstract and model some of the qualitative findings in quantitative research contexts could be valuable for generalizing in this literature. This dissertation partially does this by developing a legacy perception scale and scales to measure IT artifact characteristics. Nevertheless, this is merely a first step on one of many potential pathways forward. I believe if these disjoint literatures become better connected and build a cumulative tradition off of prior literature IS research can make more confident recommendations to practice on the management of legacy systems. This, of course, should be done with care, though, to not undermine the theoretical and methodological diversity the literature currently holds.

A general lack of economic papers is another challenge for legacy systems research. I believe the lack of economic literature can partially be explained by a general lack of available secondary datasets on legacy systems used in econometric modeling. Theoretical economic work on legacy systems can be done mathematically instead (e.g., O'Leary, 2000), but in general, this is less common in IS theorizing and less accessible. However, this literature could prove relevant in the near future. Some research has found clever ways to use the United States government budget and IT data reporting for panel regressions (Pang, 2017; Pang & Tanriverdi, 2022), and as those government datasets grow, longitudinal panel modeling may become more feasible. Additionally, in the United States context, legislation is in the works to require more reporting on legacy systems, potentially opening up new data sources in the future for public sector legacy systems (S.3897 - 117th Congress, 2022).

CHAPTER FOUR: HYPOTHESES DEVELOPMENT

Model 1: Legacy Perception as Second-order Formative

For this dissertation, I tested two different models, each with the goal of understanding legacy perception. Figure 7 presents the first approach. It shows that legacy perception is formed based on an actor's understanding of key characteristics of the technical and social environment and that this legacy perception further influences two key outcomes: replacement intention and system investment. I model legacy perception as a first-order reflective, second-order formative construct (Becker et al., 2012). This is denoted with a box with dashed lines to indicate it is a second-order construct.

At a measurement level, formative constructs align with constructivist theory (J. R. Edwards, 2011). Since I am positing that legacy perception is a socio-technical construction of the user, this constructivist foundation aligns with the theorization. However, I chose to model the first-order constructs as reflective. I argue that those first-order constructs are not constructions of the IT manager but measurable aspects of the IT manager's environment. The overall second-order model is presented in Figure 7. I posit that nine factors construct legacy perception in total.

Hypotheses testing for each first-order reflective construct with legacy perception will be evaluated with significance of model weights (J. F. Hair Jr. et al., 2022; Sarstedt et al., 2019) since those constructs are used as indicators of the second-order constructs. Chapter Five includes a more detailed discussion of how second-order formative constructs are evaluated statistically, but essentially, since those first-order reflective constructs become indicators, they are not *related* to legacy perception, but rather *compose* legacy perception. H1-H9 development should instead be interpreted as theoretical justifications for those first-order constructs as indicators of the second-order legacy perception construct.

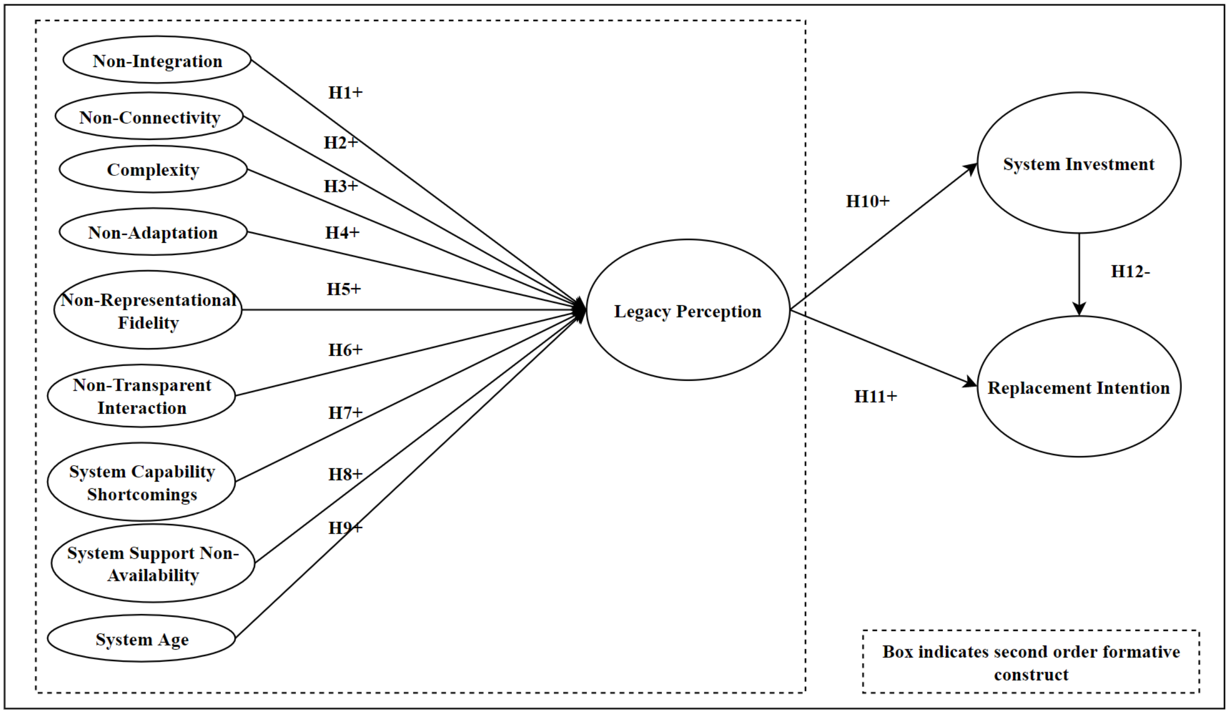

Figure 7: Model 1 – Second-order Formative

H1 through H4 are based on Matook and Brown's (2017) framework for delineating and theorizing IT artifact characteristics. With these hypotheses, I am focusing on how the artifact itself forms legacy perceptions. Matook and Brown (2017) propose seven different characteristics of IT artifacts based on systems thinking. I adapted four of these characteristics to my model but changed three to negative forms to maintain a positive relationship with legacy perception. Negative relationships between formative indicators and the latent construct are theoretically valid but are challenging to interpret and often result in suppression effects with the co-occurrence of positive indicators (Cenfetelli & Bassellier, 2009). For this reason I have chosen to model all relationships between the first-order reflective constructs and legacy perception as positive.

Table 4 summarizes how these constructs change from how Matook and Brown (2017) present them. Complexity is the only characteristic that does not change, as the hypothesis is positive for the original characteristic definition. For the other characteristics, they have been changed for positive hypotheses. I will still measure these characteristics in their original forms, but the items will be reverse-coded for the analysis. Scale endpoints in the table are from low end of the scale to high end of the scale.

Original Characteristic (Matook and Brown, 2017)	**Scale Endpoints** (Matook and Brown, 2017)	**Adapted Characteristic**	**Scale Endpoints**
Integration	Highly Fragmented to Highly Integrated	Non-Integration	Highly Integrated to Highly Fragmented
Connectivity	Highly Isolated to Highly Connected	Non-Connectivity	Highly Connected to Highly Isolated
Complexity	Less Complex to Highly Complex	No Change	No Change
Adaptation	Static to Dynamic	Non-Adaptation	Dynamic to Static

Table 4: IT Artifact Characteristic Adaptations

Integration is a measure of the aggregation of the internal IT artifact parts (Matook & Brown, 2017). In this study, I adapt this concept as non-integration with highly non-integrated systems being more fragmented. Fragmentation could be a theoretically distinct construct from integration, such as the difference between continuance and discontinuance (Furneaux & Wade, 2011; Soliman & Rinta-Kahila, 2019). By using non-integration I avoid this issue, while also resolving the issue of negative indicator weights in the construct definition (Cenfetelli & Bassellier, 2009).

Having a non-integrated system with loosely combined components makes it easier to remove and replace individual components without impacting the rest of the system. Various studies in the literature suggest modular, decoupled systems as replacements for legacy systems (Jermaine, 1999; Vestues & Knut, 2019; T. Wiggerts et al., 1997; T. A. Wiggerts, 1997). However, integrating many independent components leads to increased complexity and development of middleware systems and can be more challenging to manage and understand (K. Lee et al., 2022). These non-integrated systems often lack functionalities and synergies between artifact subsystems (Matook & Brown, 2017). As such, I hypothesize:

H1: The more non-integrated a system is, the more likely it is to be perceived as a legacy system.

Connectivity measures how connected the IT artifact is with external system parts and the environment outside of the system boundary (Matook & Brown, 2017). For this study, I adapt connectivity as non-connectivity to address the problem of negative indicators, with highly non-connective systems being more isolated from external system parts. Modern information systems often need to exchange information with other systems as part of their operation. This could be between other systems in an organization, but this can also include systems outside the organization's boundary (Matook & Brown, 2017). The ability to exchange this information is predicated on the system's physical structure. A system lacking connectivity, for example, would not have the necessary interfaces embedded in the system's physical structure.

In some cases, these functionalities are added to legacy systems post-implementation. For example, there is literature on adapting pre-internet systems for web connectivity functionalities (Abrahao & Prado, 1999; Sellink et al., 1999; Vergara et al., 2007). As information systems and digital reality (Recker et al., 2021) become more complex, the ability for artifacts to connect to one another becomes increasingly essential both technically and for business purposes. If a system lacks connectivity, it cannot, for example, be integrated with other system implementations in the organization, nor can it be transformed into a web-enabled artifact for internet functionalities. As such, I hypothesize:

H2: The more non-connective a system is, the more likely it is to be perceived as a legacy system.

Complexity refers to the number of interdependent relations that comprise the IT artifact (Matook & Brown, 2017). The complexity of legacy systems, making them difficult to maintain, is a common theme seen throughout the IS and CS literature (Fuentes et al., 2014; e.g., Gibson et al., 1998; Lei Wu et al., 2005; Rinta-Kahila, 2018). Previous descriptive research has also indicated legacy system artifacts, including accounting and ERP legacy systems, to be somewhat complex (Stachofsky, 2018). Some level of complexity is necessary for all systems (Moseley & Marks, 2006), especially if system developers wish to faithfully represent complex real-world systems within the structures of an IT artifact (Wand & Weber, 1995). However, this complexity comes with a cost.

In the long term, system complexity tends to expand regarding the number of components and interactions, requiring increased computational power (K. Lee et al., 2022; Schneberger & McLean, 2003). Additionally, the cognitive load on developers to fully understand the individual components and their interactions becomes more difficult as systems increase in complexity (K. Lee et al., 2022; Scandura, 1994). While the complexity in the system may have been implemented for a logically necessary purpose, that complexity will make attempts to modify that system more difficult and time-consuming. Thus, I hypothesize:

H3: The more complex a system is, the more likely it is to be perceived as a legacy system.

The flexibility of IT infrastructure is an enabler of organizational responsiveness and competitive advantage (Bhatt et al., 2010). This flexibility is based on the ability to change the artifact to meet a changing business context. If a system remains static, it can still meet the functional requirements of earlier iterations of a business problem or needs of that context. However, business problems will change over time, the technical environment will change, and users' expectations will change. If the system is not adaptable and can respond to those changes, then it will be insufficient to meet the business needs (Gibson et al., 1998; Kelly et al., 1999). Based on this, I hypothesize:

H4: The more non-adaptive a system is, the more likely it is to be perceived as a legacy system.

H5 and H6 are based on representation theory concepts (Recker et al., 2019; Wand & Weber, 1990, 1995), including the recent extensions of traditional representation theory to include digital reality (Recker et al., 2021). The two constructs I adapt from representation theory are based on the effective use conceptualizations from Burton-Jones and Grange (2013): representational fidelity, adapted as non-representational fidelity, and transparent interaction, adapted as non-transparent interaction. As with the IT artifact characteristic relationships, these constructs were logically negated to create a positive relationship with the formative construct.

Representational fidelity is defined as "the extent to which a user is obtaining representations from the system that faithfully reflects the domain being represented," and transparent interaction is defined as "the extent to which a user is accessing the system's representations unimpeded by its surface and physical structures" (Burton-Jones & Grange, 2013, p. 642). One thing to note about these definitions is that, theoretically, they are characteristics of *use*, not characteristics of *the artifact*. These constructs are socio-technical in nature, neither a pure evaluation of the user or the system. A strict reading of representation theory suggests representational fidelity could be evaluated via the deep structure as a system property divorced from social context (Wand & Weber, 1990, 1995).

However, as defined by Burton-Jones and Grange (2013), representational fidelity is a characteristic of system use, not the system. The focus is on what the users obtain from the system's deep structure, and users may operate at different levels of representational fidelity. Users may perceive different representational fidelity levels from the same system based on their usage patterns. Similarly, for transparent interaction, since the construct is focused on the user's ability to access a deep structure unimpeded by the physical and surface structures, there must be a use behavior to evaluate. In the context of an empirical behavioral model instead of a conceptual data model, some theoretical purity is sacrificed to measure these constructs practically. The primary difference is these measures evaluate representational fidelity via system use, instead of evaluating the conceptual modeling logic directly. There is support for measuring representational fidelity this way (Burleson et al., 2021) in behavioral research, although transparent interaction has not been tested.

Representation theory posits that a system with a more faithful representation of the real-world system within the deep structure of the information system will result in a more useful and effective information system (Burton-Jones & Grange, 2013; Wand & Weber, 1995). In addition to real-world physical systems, the faithful representation should include digital reality and the ability to mediate interactions between physical and digital realities (Recker et al., 2021). These faithful representations are the foundation of an effective and accurate business model being embedded into a legacy system's structure. Thus, I hypothesize:

H5: The more non-representationally faithful a system is, the more likely it is to be perceived as a legacy system.

Surface structure consists of components like the user interface that allow the users to access the deep structure of the information system (Recker et al., 2019; Wand & Weber, 1995).

A well-implemented physical and surface structure leads to a more transparent interaction with the system (Burton-Jones & Grange, 2013). In order for the deep structure of an information system to be utilized, there must be an adequate surface structure to access it (Recker et al., 2019; Wand & Weber, 1995). If that surface structure has significantly degraded, then the information system users will be less likely to extract value from the system. There is no way for the users to effectively access and interpret the business model embedded in the legacy system structure. Therefore, I hypothesize:

H6: The more non-transparent interactions with a system are, the more likely it is to be perceived as a legacy system.

An important aspect of the social system is the vendors that support information systems. Vendors providing after-sales support is essential for most large enterprise software projects and is a crucial component of software evolution (Ofoegbu et al., 2011). An organization's software often consists of heterogeneous vendor-specific solutions (Sasso & Forcolin, 2009), making that support even more critical and challenging to develop internally without intensive reverse engineering. When vendor support is available, organizations are less likely to replace their legacy systems (Furneaux & Wade, 2017). Over time, however, vendors reduce or end support entirely as they retire software products in their portfolio (Mehrizi et al., 2019; Schnappinger & Streit, 2021). Additionally, finding staff with required expertise and replacement components for the system can indicate a lack of support (Furneaux & Wade, 2011) I posit that a lack of support availability often signals to organizations that a system is at or near end of life. Therefore, I hypothesize:

H7: The more a system lacks support, the more likely it is to be perceived as a legacy system.

One of the primary reasons an organization may discontinue a system is if the system's capabilities no longer meet the requirements of the business (Furneaux & Wade, 2017). These shortcomings often emerge partly because vendors lack of support to develop new software features (Furneaux & Wade, 2017). These capability shortcomings indicate a widening gap between the deep structure representation of the business processes in an information system and the actual current processes, reducing the overall usefulness of the system (Burton-Jones & Grange, 2013; Wand & Weber, 1995). The emphasis on the lack of business functionality (Brooke, 2002; Brooke & Ramage, 2001; Kelly et al., 1999; Pang, 2017) underlies many definitions of legacy systems and leads to reduced capabilities and performances of organizations (Furneaux & Wade, 2017; Pang, 2017). Thus, I hypothesize:

H8: The more capability shortcomings a system has, the more likely it is to be perceived as a legacy system.

Perhaps the most common theme shared across all legacy systems definitions in the literature is that legacy systems are old incumbent systems (Bennett, 1995; Bisbal et al., 1999). Legacy systems have a strong temporal component (Light, 2003). Even if there is no pre-determined time threshold by which a system moves from non-legacy to legacy, the system must be implemented for some period of time before it can be considered legacy. The notion of obsolete technology is a common thread that runs throughout the definitions of legacy systems in the IS (Azadmanesh & Peak, 1995; Chirathamjaree, 2006; Mahapatra & Lai, 1998; Mallampalli & Karahanna, 2017; Tsai et al., 2022) and CS literature respectively. Empirical work has also shown a link between system age and the lower life expectancy of the system (Swanson & Dans, 2000). Therefore, I hypothesize:

H9: The older a system is, the more likely it is to be perceived as a legacy system

This final group of hypotheses is focused on the impacts of a system being perceived as legacy. One outcome of interest is system investment. Continual investment in a system is necessary for a system to evolve and meet new business needs. However, continuing to invest in a system also increases the replacement risk of a legacy system as well as the complexity of the artifact (Furneaux & Wade, 2017). Often, in their attempts to detach from a system, organizations will continue to invest more into the legacy system and incur additional technical debt (Rinta-Kahila et al., 2023). Although the legacy perception may signal to the organization that they should cease continual investment, the legacy system is a known asset and is the current system in place. The legacy system may be flawed, but it is in place, and providing value for the organization (Gholami et al., 2017; Light, 2003). Additionally, when an organization is prepared to replace a legacy system that process requires additional investment to complete effectively (Mehrizi et al., 2012, 2019). Therefore, I hypothesize:

H10: The more a system is perceived as legacy, the more likely it is to receive investment.

Replacement is often the end state of legacy systems in an organization. This assumes that the legacy perception is negative and signals to the organization to seek replacement of the system. Previous work has found that system capability shortcomings and lack of system support are drivers of replacement intentions (Furneaux & Wade, 2017), which are two of the factors forming my conceptualization of legacy perception. If a system is perceived as legacy, it suggests that, on some level, the organization's needs are no longer being met by the system and should be replaced.

However, the discontinuance process also results in increased allocation of resources to legacy systems (Mehrizi et al., 2012, 2019) as that replacement process is complex and expensive. Organizations also significantly increase their replacement risk with continual investment in a system, making them less likely to replace the system (Furneaux & Wade, 2017; Rinta-Kahila et al., 2023). A tension exists in that the system is receiving continual investment, increasing the risk of replacement for the organization. Thus, I hypothesize:

H11: The more a system is perceived as legacy, the more likely it is the organization intends to replace the system.

H12: The more investment a system receives, the less likely it is the organization intends to replace the system.

Model 2: Legacy Perception as First-order Reflective

Figure 8 presents the second model of legacy perception that will be tested. In this model, the interaction of different technical characteristics drives key characteristics of system inadequacy, which, along with factors of the social environment, influence legacy perception. State is the only new variable in this model as one of the technology characteristics, and legacy perception is now measured with reflective items instead of as a second-order formative construct. As with the previous model, this legacy perception impacts system investment and replacement intention.

Since every construct is a first-order construct, there is no concern over negative relationships for this model. As such, I have included the original names for each construct instead of the logically negated versions. It should be noted, though, that the items are the same. They are just no longer negated to maintain the positive relationship with legacy perception. Additionally, all hypotheses can be explicitly tested since the model is evaluated with a single structural model rather than a two-stage model where most constructs become indicators.

The primary benefit of this model is the ability to model relationships between the various characteristics. In Model 1, these constructs are used as indicators for the second-order construct, and while correlated to some extent, they could not have relationships with each other. In this model, I can capture how technical characteristics interact with each other and how they influence the socio-technical constructs. This model posits that the technical characteristics themselves are not driving legacy perception; instead, they determine the system's deep and surface structures, which are influencers of legacy perception. In addition to the system structure, the constructs from the social environment are included. They are still directly related to legacy perception aligning with Model 1.

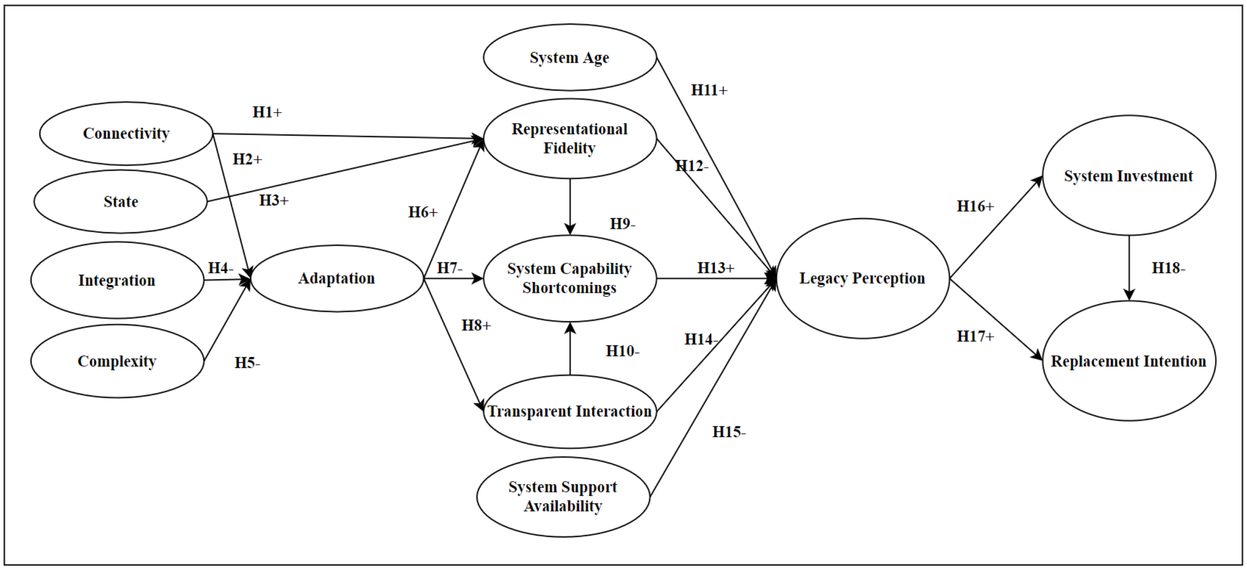

Figure 8: Model 2 – First-order Reflective

Overall, the two models are not necessarily competing models but instead reveal different things about the legacy perception phenomenon. The first is determining if legacy perception should be conceptualized as a second-order formative construct or a first-order reflective construct. Results will not be entirely conclusive since different factors could theoretically be selected for the formative construct in a similar study.

However, if constructs have a significant relationships in Model 2 with the reflective measure of legacy perception but have poor weights as indicators of the second-order formative construct, that would suggest that legacy perception may be better suited for reflective measurement, at least with the proposed factors. Those factors externally influence legacy perception but do not compose legacy perception. I am comparing the models in the sense of how to conceptualize legacy perception. If Model 1 is an accurate conceptualization of legacy perception, then Model 2 illuminates how those important factors interact. If Model 1 is an inaccurate conceptualization of legacy perception, then the reflective approach of Model 2 is a better explanation of the phenomenon.

The second reason testing both models is necessary is to capture relationships between the different first-order reflective measures. For example, Model 1 cannot model relationships between the technical, social, and socio-technical use characteristics because they are used as indicators of the second-order formative construct. One goal of this research is to study the interactions of internal structures of the legacy system. Model 2 allows for this and will provide insights into IT artifact characteristic interactions and representation theory (Matook & Brown, 2017; Wand & Weber, 1995)

H1 through H8 are relationships based on the IT artifact characteristics as they relate to each other and socio-technical use characteristics. Connectivity has two hypothesized relationships. The first is with representational fidelity. For a system to exhibit a faithful representation, it must have an accurate model of both physical and digital reality (Recker et al., 2021). Connectivity is necessary because it enables a device to mediate interactions in digital space and account for unique artifacts embedded in cyber-physical contexts (Brandt et al., 2018; Hauser & Gã, 2017). If there is no interface for a system to interact with other systems, then the internal deep structure will be incomplete in representing the digital interactions of a business process. Therefore, I hypothesize:

H1: The more connectivity a system has, the higher the representational fidelity will be.

The second hypothesis related to connectivity is between connectivity and adaptability of a system. Older systems are often integrated with other newly implemented systems in an organization. This is done through mechanisms like middleware systems (Buchmann & Karagiannis, 2016; Chirathamjaree, 2006; Chou & Seng, 2009; e.g., Vergara et al., 2007) and is often responding to a change in the technical environment and business needs (Thummadi et al., 2017). Those necessary system integrations cannot be built if a system lacks connectivity. Meaning that a system is not adaptable to change such that changes in the business process can be met (Kelly et al., 1999). Thus, I hypothesize:

H2: The more connectivity a system has, the higher the adaptability will be.

In this research, I develop a measure of state as an IT artifact characteristic (Matook & Brown, 2017). A stateful artifact tracks the current status and history of significant interactions with the artifact. An example of this would be an ecommerce system that tracks customer data and order processes. A fully stateless artifact does not retain or track any changes. An example would be a plain hypertext markup language website where the user does not login or have their actions recorded (Matook & Brown, 2017).

A key component of representation theory is the state-tracking model (Recker et al., 2019; Wand & Weber, 1988, 1990, 1995). The state tracking model consists of four components. Mapping real-world states to information system states, changing information system states to match changing real-world states, reporting of events that occur in the real-world system in the information system, and sequencing of events and state changes within the information system to match the real-world system (Wand & Weber, 1990). This meticulous and accurate state tracking enables the ability to create an accurate representation model (Thomas & Dhillon, 2012; Wand & Weber, 1990, 1995).

Generally, the representation theory literature has focused on the representation model rather than the state-tracking model (Thomas & Dhillon, 2012; Wand & Weber, 2017). However, given how critical the system state is to enable a representation model (Wand & Weber, 1990, 1995), neglecting the state-tracking model leaves a test of the theory incomplete. As such, I study the relationship between system state and representational fidelity to test one of the core claims of representation theory. Therefore, I hypothesize that:

H3: The more stateful a system is, the higher the representational fidelity will be.

In the context of this study, integration is a measure of the aggregation of the internal IT artifact parts (Matook & Brown, 2017), not a system integration between multiple artifacts. Monolithic IT artifacts can be more challenging to maintain compared to modular systems (Jermaine, 1999; Vestues & Knut, 2019; T. A. Wiggerts, 1997) as components cannot be easily removed or changed without making changes to numerous other portions of the artifact architecture. With a strongly integrated system, functionalities and synergies between artifact subsystems can emerge (Matook & Brown, 2017) but operate under the assumption that the system does not need to change significantly. If an organization's technology architecture or business requirements change dramatically, adapting to a highly integrated monolithic system is challenging. As such, I hypothesize:

H4: The more integrated a system's structure is, the lower the adaptability will be.

Complexity also affects the adaptability of a system. Complexity is required in all systems (Moseley & Marks, 2006), especially if system developers wish to represent complex real-world systems faithfully (Wand & Weber, 1995). However, complexity also makes systems more challenging to maintain (e.g., Fuentes et al., 2014; Gibson et al., 1998; Lei Wu et al., 2005; Rinta-Kahila, 2018). The cognitive load on developers to fully understand individual system components and their interactions becomes more difficult as systems increase in complexity (K. Lee et al., 2022; Scandura, 1994). Complexity increases by having systems with many interdependent relations (Matook & Brown, 2017), and the difficulty in understanding and modifying that system also increases, making adapting that system to new business requirements more difficult. Based on this, I hypothesize that:

H5: The more complex a system is, the lower the adaptability will be.

The adaptability of a system is hypothesized to be related to two socio-technical characteristics of use as well as the system's overall capabilities. Adaptability is considered an IT artifact characteristic of a system’s physical structure. All deep structure and surface structure representations depend on the implementation of the physical structure (Wand & Weber, 1995). If the physical structure of a system is not adaptable and resistant to change, this impacts the higher-level deep and surface structures. As the business context changes, the system must change to avoid having an outdated model in the deep structure of the business processes (Gibson et al., 1998; Kelly et al., 1999).

What elements of the user interface are available and how they are implemented will be determined by the physical structure. If the physical structure is not adaptable to change then the surface structure and transparent interaction will not be adequate. The system will struggle to meet business needs if the physical structure is not adaptable to changes in user interface requirements in the surface structure to enable access to that deep structure representation (Burton-Jones & Grange, 2013; Wand & Weber, 1995). This also manifests in overall system capability shortcomings (Furneaux & Wade, 2017), as developers cannot add new system functionalities to a system that is not adaptable to change. Based on this, I hypothesize that:

H6: The more adaptable a system is, the higher the representational fidelity will be.

H7: The more adaptable a system is, the lower the system capability shortcomings will be.

H8: The more adaptable a system is, the higher the transparent interaction will be.

H9 and H10 are premised on representation theory. The theory's core idea is that if a system is more faithfully modeling the real-world within its structure, the system will be more useful (Recker et al., 2019; Wand & Weber, 1995). Burton-Jones and Grange (2013) bring this core idea into the realm of effective use of information systems. While a traditional reading of representation theory would suggest deep and surface structures can be studied divorced from a social context (Wand & Weber, 1990, 1995), Burton-Jones and Grange (2013) argue that they must be understood as socio-technical characteristics of usage behavior. System use requires a user, system, and task (Burton-Jones & Grange, 2013; Burton-Jones & Straub, 2006). A user’s actions within a given scenario will influence their perceptions of the deep structure representation and the surface structure interactions necessary to access that representation.

In-order for a system to meet the business needs of an organization, it needs an accurate model of the real-world embedded in its structure (Recker et al., 2019; Wand & Weber, 1995). However, this is not sufficient. A user must also be able to utilize this representation effectively via the surface structure of the information system (Burton-Jones & Grange, 2013). A system that lacks an accurate real-world representation of the current business processes and the ability to access that real-world representation will lack the key capabilities necessary to support the business purpose of the system. Therefore, I hypothesize that:

H9: The more representationally faithful a system is, the lower the system capability shortcomings will be.

H10: The more transparently interactive a system is, the lower the system capability shortcomings will be.

H11 through H17 all overlap with relationships in Model 1. The corresponding hypothesis in Model 1 will be denoted as [H#]. They are written here again for clarity and completeness, but the full hypothesis justifications can be found in the previous section for Model 1.

H11 [H9]: The older a system is, the more likely it is to be perceived as a legacy system.

H12 [H5]: The more representationally faithful a system is, the less likely it is to be perceived as a legacy system.

H13 [H8]: The more system capability shortcomings a system has, the more likely it is to be perceived as a legacy system.

H14 [H6]: The more transparently interactive a system is, the less likely it is to be perceived as a legacy system.

H15 [H10]: The more a system is perceived as legacy, the more likely it is to receive investment.

H16 [H11]: The more a system is perceived as legacy, the more likely it is the organization intends to replace the system.

H17 [H12]: The more investment a system receives, the less likely it is the organization intends to replace the system.

In addition to the hypotheses, I include two control variables in Model 1 and Model 2. These controls are participant gender and age, associated with legacy perception. Since I argue that legacy perception is a construction of an IT manager, it is worth investigating if the characteristics of that IT manager influence the formation of legacy perception. The age of the participant could be particularly relevant as their perception of how old a system is could be in relation to how old they are themselves.

CHAPTER FIVE: METHODOLOGY

Sampling and Data Collection

For this dissertation, I conducted a survey of IT managers. I selected IT managers as the target sample because they are qualified enough to evaluate an information system's technical structures while also understanding its business context. If sampling only end-users, I would not be able to collect accurate data on the internal structure of the artifact, and if sampling only technical developers, I would not be able to collect accurate data on the social subsystem.

To calculate the necessary sample size, I use the inverse square root method for minimum sample size estimation in PLS-SEM (Kock & Hadaya, 2018). Equation 1 denotes the formula used, where $\widehat{N}$ is the estimated minimum sample size, $|\beta|_{min}$ is the absolute value of the statistically significant path coefficient with minimum acceptable magnitude in the model, z_x is the z-score for the significance level, and z_y is the z-score for the power level.

$$|\beta|_{min}\sqrt{\widehat{N}} > z_x + z_y \rightarrow \widehat{N} > \left(\frac{z_x + z_y}{|\beta|_{min}}\right)^2$$

Equation 1

I calculate this based on a minimum path coefficient of 0.20 (W. Chin, 1998), a statistical power level of 0.80, and a significance level of p = .05. Equation 2 shows the formula with these values included:

$$\widehat{N} > \left(\frac{1.645 + 0.842}{0.20}\right)^2$$

Equation 2

Based on this calculation, the minimum acceptable sample size is 155 participants. My final sample of 221 in Round 1 and 273 in Round 2 of data collection cleared these 155 minimum sample size requirements.

Procedure

Participants were asked to answer the survey based on one legacy system in their organization. I did not provide a definition of legacy to the participant, as I did not want to prime respondents on expected responses. Since legacy perception is being theorized as a socio-technical construction, the user must be able to construct what legacy means to them for this research. Participants were recruited from EMpanel Online (*EMpanel Online*, n.d.), a market research survey firm specializing in business-to-business. This firm was selected because they specialize in IT professionals. This includes a specialized respondent pool of IT managers which was used for this research.

The survey was conducted in two rounds but was not longitudinal, each round had different participants. The first round focused on scale development and was conducted on February 7, 2024. The second round was focused on model testing and was conducted on February 19, 2024. There were two filtering checks. First to get consent for data collection from the research participant and a second check to determine if they worked in IT management. Participants that did not consent or did not work in IT management had their surveys immediately closed. There was also one attention check question “If 2+2 = 4 select Somewhat disagree”, which immediately ended the survey of participants selected the wrong option.

The survey for both rounds was conducted online via Qualtrics XM (Qualtrics International Inc., 2017) and is available in full as Appendix D Exhibit 2. EMpanel distributed the link to participants who could take it on any device of their choice with an internet connection. Payment for data collection was made directly to EMpanel who handled compensation of participants internally.

Measures

The survey uses a mixture of adapted existing measures and entirely new measures. In this section, I will present each construct with the original item wordings and my adaptations grouped by the first-order constructs that form legacy perception and the dependent variables. A summary of each measure used and the scale source is presented in Table 5. Specific items can be found in Appendix A.

Construct	Source	Models
Integration	Self-developed based on Matook and Brown's (2017) definition	1 and 2
Connectivity	Self-developed based on Matook and Brown's (2017) definition	1 and 2
Complexity	Furneaux and Wade (2011) Technical Integration	1 and 2
State	Self-developed based on Matook and Brown's (2017) definition	2
Adaptation	Self-developed based on Matook and Brown's (2017) definition	1 and 2
Representational Fidelity	Adapted to legacy system context from Burton-Jones and Grange (2013)	1 and 2
Transparent Interaction	Adapted to legacy system context from Burton-Jones and Grange (2013)	1 and 2
System Support Availability	Furneaux and Wade (2017)	1 and 2
System Capability Shortcomings	Furneaux and Wade (2011); Modified to remove double-barreled questions.	1 and 2
System Age	Furneaux and Wade (2017)	1 and 2
System Investment	Furneaux and Wade (2011); Modified to remove double-barreled questions.	1 and 2
Replacement Intentions	Furneaux and Wade (2017)	1 and 2
Legacy Perception	Self-developed	2

Table 5: Summary of Measures

Most IT artifact characteristics proposed by Matook and Brown (2017) do not have existing scales in the literature. Exploratory research has measured these characteristics

(Stachofsky, 2018) but uses single-item measures and does not fully capture the theoretical nuance of the characteristics. Matook and Brown (2017) provide a clear literature review and definitions of each characteristic and what they consider the extreme points of each characteristic, allowing content for creating scales (Clark & Watson, 2019, 1995; MacKenzie et al., 2011). When possible, I adapted existing validated scales. For example, the complexity measure is adapted from Furneaux and Wade (2011) as it theoretically matches Matook and Brown's (2017) definition of complexity. However, for most of the artifact characteristics, new scales were developed.

Matook and Brown's (2017) definition of integration focuses on internal system components and how they connect together. A system can be highly integrated if the internal components are tightly coupled or highly fragmented if the components that construct the artifact are only loosely combined. Integration is usually discussed in the literature around connections between systems (e.g., K. Lee et al., 2022; Y. Weber & Pliskin, 1996). In this context, the focus is on the internal structure of the artifact, for which there are currently no measures. I developed scale items for integration, which are presented in Table A1.

In contrast to their integration characteristic, Matook and Brown's (2017) definition of connectivity is focused on external interactions. Highly isolated artifacts do not have connections to other systems. Artifacts that are highly connected connect with many other systems in the environment. Similar to integration, no scales exist for measuring connectivity for survey research. I developed scale items for connectivity, which are presented in Table A2.

I use the technical integration scale from Furneaux and Wade (2011) for complexity. The items are available in Table A3 and are not modified for this study. This scale captures the interdependencies of complexity, aligning with the source theory (Matook & Brown, 2017) that I draw from. Despite being named technical integration, all three items ask about complexity. In a later study, Furneaux and Wade (2017) developed a different complexity measure called system complexity. Unlike the 2011 scale, the 2017 scale does not ask about integration and connections of system components. I use the 2011 scale in this study as it is theoretically closer to Matook and Brown's conception (2017). However, the Furneaux and Wade (2017) scale items are available in Table A3.

System state is generally covered in the more technical and conceptual model-oriented IS literature (e.g., Wand & Weber, 1988, 1990, 1995). As such there are no scales currently for measuring system state in survey research. Matook and Brown's (2017) definition of state posits that IT artifacts exist on a spectrum between completely stateless and stateful. If a system is stateless, it does not store previous states of the system, and if it is stateful, the system maintains a record of previous system states. Based on this definition, I propose scale items for state presented in Table A4.

IT adaptability has been measured in behavioral information systems research, derived from adaptive structuration theory (Bhattacherjee & Harris, 2009). However, specific scale items have not been provided, nor does that theoretical lens align entirely with the adaptation defined by Matook and Brown (2017). Information systems adaptation has also been measured, although with specific metrics related to manufacturing processes (Frohlich & Dixon, 1999). Wang et al. (2013) have also developed an IT adaptability measure, but their measure conflates the overall IT department adaptability with the individual artifact.

None of these measures adequately capture what Matook and Brown (2017) mean by adaptability, so I developed new scale items for this research. Matook and Brown (2017) suggest that the extreme points of this characteristic are static systems when adaptation is low and dynamic when adaptation is high. The overall scale must capture the ability of an IT artifact to change. One thing to note is that the adaptation concept is neutral regarding what actor, technical, human, or self, is enacting change. It is merely capturing whether the artifact is capable of being changed. My scale items for adaptation can be found in Table A5.

Representational fidelity is adapted from the representational fidelity scale developed by Burton-Jones and Grange (2013) but was not empirically tested in their paper. One study has used a modified version of the construct in the context of Facebook messenger communication (Burleson et al., 2021). Results from that study suggest that representational fidelity is a reliable and valid construct when contextualized adequately to the information system being studied. Table A6 summarizes Burton-Jones and Grange's (2013) items for representational fidelity and my proposed contextualized items. Since I am collecting data from the IT manager on behalf of the organization the wording is shifted from individuals framing to a more general framing.

Table A7 summarizes Burton-Jones and Grange's (2013) items for transparent interaction and my proposed contextualized items. In their scale, positive and negative items are mixed. While scale mixing is common practice, it can often lead to issues of scale validity and reliability (Chyung et al., 2018). As a tradeoff, the potential for acquiescence bias is higher in my study, but the consensus of survey design research would suggest this is a reasonable tradeoff (Chyung et al., 2018). In my adaptation, I only use positively worded items.

System support availability is based on Furneaux and Wade’s (2011) scale for system support availability. The scale captures the extent to which resources are available to support a system. The original and adapted items are available in Table A8. System capability shortcomings is measured using Furneaux and Wade’s (2011) scale. The items measure whether a system meets the business requirements of the organization. The items remain unchanged except for one modification to an item and one additional item to remove a double-barreled question presented in Table A9. System age is measured in years with a one-item question similar to Furneaux and Wade's (2017) measure. The item is available in Table A10.

System investment is measured using a three-item scale from Furneaux and Wade (2011) instead of asking for a specific monetary value that most respondents may not have access to or may not be tracked by the organizations. The scale captures the level of investment an organization has made in a system. The items remain unchanged except for one modification to an item and one additional item to remove a double-barreled question presented in Table A11. The replacement intention scale also comes from Furneaux and Wade (2011), specifically the slightly revised scale used in Furneaux and Wade (2017). Items for both scales can be found in Table A12. I will use the 2017 scale in this study as it does not make assumptions about the replacement system ("another" instead of "competing") and changes "will be implementing" to "will be seeking to implement". The original wording implied that a replacement project was already underway.

The final scale is for legacy perception. This scale is unique to Model 2, as legacy perception is a second-order construct in Model 1. For this scale, I propose three items presented in Table A13. Processes for developing and testing the new scales can be found in Appendix B.

Analysis

Data analysis consisted of two rounds. The first round was focused on scale development concerns following the guidance of Mackenzie et al.'s (2011) scale development procedures. The analysis consisted of three rounds of item card sorting, reliability tests, discriminant and convergent validity tests, and a confirmatory factor analysis using covariance-based structural equation modeling (CB-SEM). A complete analysis of the scale development procedure and results are available in Appendix B.

After scales were sufficiently validated, the second analysis round was conducted through a second survey of EMpanel IT Manager participants. Data was analyzed using partial least squares structural equation modeling (PLS-SEM) for both models. I chose PLS-SEM for this dissertation due to my model's mixture of construct types. PLS-SEM is robust to mixtures of reflective and formative measures and single-item constructs (J. Hair Jr et al., 2017). Model 1 includes a single-item construct (system age), first-order reflective constructs, and a second-order reflective-formative construct. While it is possible to include these mixed constructs in CB-SEM, it requires additional modifications to construct specification (J. Hair Jr et al., 2017; J. F. Hair Jr et al., 2011). Additionally, PLS-SEM is ideal for cases with many constructs and indicators, and the goal is to identify key drivers of a construct (J. Hair Jr et al., 2017; J. F. Hair Jr et al., 2011).

In general, the differences between CB-SEM and PLS-SEM concerning parameter accuracy are often overstated (J. Hair Jr et al., 2017). Even in cases of moderate non-normality and small sample sizes, PLS-SEM and CB-SEM results are essentially the same concerning relationship significance (Goodhue et al., 2012). The differences remain in the handling of measurement error (Goodhue et al., 2012) between the variance approach of PLS-SEM and the covariance approach of CB-SEM. Considering the mixture of construct types and complexity of my models, I posit that PLS-SEM is the best choice overall for this research. Descriptive statistics were evaluated using SPSS version 27.0.0.0 (IBM Corp., 2020). Construct cross-loadings, PLS-SEM structural models, reliability, and CFA CB-SEM structural models were evaluated using SmartPLS version 4.1.0.0 (Ringle et al., 2024).

For Model 1, I chose to model legacy perceptions as formative because, with formative constructs, changes in measures cause changes in construct, but changes in the construct do not change the measures (Petter et al., 2007). If an IT manager no longer perceives a system as legacy, that would not fundamentally change anything about the system's state. Additionally, the indicators of legacy perception are characteristics of the construct, not manifestations of the construct. The indicators are not interchangeable, and dropping the indicators would fundamentally alter the construct domain of legacy perception (Jarvis et al., 2003).

Since Model 1 consists of a second-order formative construct consisting of first-order reflective constructs as indicators, the model was evaluated using a two-stage approach known as the extended repeated indicators approach (J. F. Hair Jr. et al., 2022; Sarstedt et al., 2019). In the first stage, the second-order formative construct (legacy perception) includes all of the individual indicators of the first-order reflective constructs. The first-order reflective constructs are also still part of the model and point to the second-order construct in the model. When estimating the PLS-SEM model, the result is not immediately meaningful, as the variance explained by the second-order construct is $R^2 = 1.00$ due to the overlap in indicators.

The standardized latent variable scores for each of the first-order reflective constructs for each data point are calculated as part of the first-stage of the model. These serve as a proxy to represent the first-order constructs (J. F. Hair Jr. et al., 2022; Sarstedt et al., 2019) and are saved as new variables in the dataset. Stage-two of the model is then estimated with these measures as formative indicators of the second-order construct instead of modeling individual reflective constructs. The second-order construct also removes the reflective indicators from stage one of the model since these are now captured with the latent variable scores. The PLS-SEM model is estimated again, with results showing the relationship between the second-order formative construct and any other dependent or independent variables. Figure 9 provides a simplified example of how the two-stage extended repeated indicators approach is modeled.

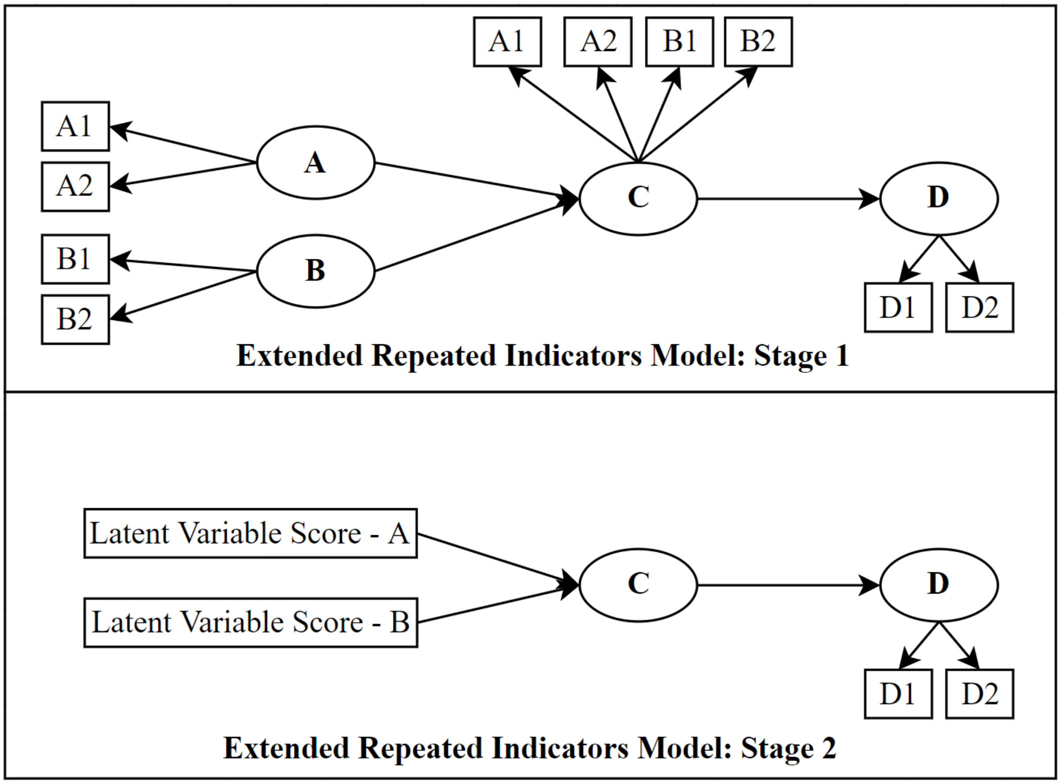

Figure 9: Example of Second-Order Formative First-Order Reflective Extended Repeated Indicators Two-Stage Model Approach

In this example, I model two first-order reflective constructs (A and B) that are indicators of a second-order formative construct (C). A and B each consist of two reflective indicators (A1, A2, and B1, B2). A fourth variable (D) is a first-order reflective construct with two indicators (D1 and D2) and is a dependent variable being predicted by the second-order formative construct, C. In stage one, A and B point to construct C, but construct C also includes all of the reflective indicators from A and B. In stage two, A and B are replaced with the standardized latent variable scores calculated from the stage one PLS-SEM model estimation and used as the formative indicators of C.

CHAPTER SIX: RESULTS

The results of this study are broken into two rounds of data collection. The first round is focused on measurement validation and testing of the new scales, adaptation, connectivity, integration, state, and the reflective version of legacy perception. Details on item generation and card sorting for these new scales are available in Appendix B. The second round is focused on testing the two proposed structural models. Supplementary statistics for both rounds can be found in Appendix C.

Round 1: Measurement Validation

After the scale development and card sort analysis, I conducted scale validity tests with a sample of IT managers recruited from the market research firm EMpanel Online (*EMpanel Online*, n.d.). The survey data was collected using the Qualtrics XM survey platform (Qualtrics International Inc., 2017) and analyzed using SPSS version 27.0.0.0 (IBM Corp., 2020) and SmartPLS version 4.1.0.0 (Ringle et al., 2024). Data was collected for all constructs to check for reliability, convergent, and discriminant validity. However, the confirmatory factor analysis (CFA) was only conducted on the newly developed scales from the card sort: adaptation, connectivity, integration, state, and legacy perception.

Data Cleaning

The survey was distributed to 309 participants. Fourteen participants declined the research consent check, four were removed for providing nonsensical answers in the text response box, ten did not meet the IT management experience requirement, and 31 failed the attention check in the survey. This left a total of 250 valid responses.

The next step was removing univariate outliers from the dataset. The *z*-scores were calculated for each indicator for the constructs. Then, the minimum and maximum *z*-scores were checked for each indicator to see if it fell in the acceptable range of -3.29 to 3.29 (Tabachnick & Fidell, 2018, p. 64). 20 of the 44 indicators had at least one *z*-score outside that range. Responses were checked for each of the problematic indicators, and 26 univariate outliers were removed in total.

I then checked for multivariate outliers for each construct with multiple items. Items for each construct were averaged into single measures. These measures were then used to calculate the Mahalanobis distance measure (Mahalanobis, 1936) and the Mahalanobis distance probability. Responses with a probability of less than 0.001 (Tabachnick & Fidell, 2018, p. 84) were considered multivariate outliers and dropped from the dataset. There were 11 responses that met this outlier criteria. However, eight had already been identified with the univariate outlier test, leaving only three additional responses to remove. In total, 221 valid responses remained for analysis.

Descriptive Statistics

21.3% of respondents were women ($n = 47$), 78.7% were men ($n = 174$), and no respondents were non-binary or had other gender identities. The average respondent age was 40.65, ranging from 19 to 73. 3.2% ($n = 7$) of respondents worked in lower management, 33.5% ($n = 74$) in middle management, 45.2% ($n = 100$) in upper management, 14% ($n = 31$) as a chief information officer, and 4.1% ($n = 9$) as chief information security officer. The age of the legacy systems ranged from 1 to 63 years, with an average system age of 17.81 years.

I conducted skewness and kurtosis tests to check that the data was normally distributed. Specifically, all constructs used were checked to establish that their skewness and kurtosis values were between -1 and +1 to establish normality (Cuttler, 2014). The test was conducted on the averaged single-measure versions of the constructs. Two constructs exhibited both skewness and kurtosis. System age had a skewness of 1.46 (*SE* = 0.16) and a kurtosis of 1.768 (*SE* = 0.326), suggesting a leptokurtic distribution with a positive skew. Replacement intentions had a skewness of -1.38 (*SE* = 0.16) and a kurtosis of 2.38 (*SE* = 0.33), suggesting a leptokurtic distribution with a negative skew (Cuttler, 2014). The positive skew for system age is not entirely unexpected, as the types of systems considered legacy in organizations typically will have been implemented for a more extended period. The meaning of the replacement intentions distribution is less clear theoretically and may pose problems as it violates assumptions of normality. Overall descriptive statistics are presented in Table 6.

Construct	Mean	Std. Error	Std. Deviation	Variance	Skewness	Std. Error	Kurtosis	Std. Error
Adaptation	5.32	0.07	1.10	1.20	-0.58	0.16	-0.33	0.33
Complexity	5.50	0.06	0.90	0.80	-0.60	0.16	0.17	0.33
Connectivity	5.62	0.06	0.83	0.69	-0.45	0.16	-0.11	0.33
Integration	5.65	0.05	0.77	0.59	-0.31	0.16	-0.42	0.33
Legacy Perception	5.94	0.05	0.80	0.63	-0.73	0.16	-0.08	0.33
Replacement Intentions	5.35	0.08	1.25	1.56	**-1.38**	0.16	**2.38**	0.33
Representational Fidelity	5.52	0.06	0.89	0.79	-0.58	0.16	-0.08	0.33
State	5.73	0.05	0.80	0.64	-0.58	0.16	-0.16	0.33
System Age	17.81	0.84	12.53	157.12	**1.46**	0.16	**1.77**	0.33
System Capability Shortcomings	5.08	0.08	1.14	1.31	-0.56	0.16	-0.13	0.33
System Investment	5.66	0.05	0.80	0.63	-0.23	0.16	-0.64	0.33
System Support Availability	5.04	0.08	1.26	1.58	-0.70	0.16	0.12	0.33
Transparent Interaction	5.25	0.07	1.08	1.16	-0.62	0.16	0.14	0.33

Table 6: Summary of Measures

Convergent and Discriminant Validity

My next step was to evaluate the loadings and cross-loadings of the constructs to check for convergent and discriminant validity, respectively. To generate these loadings and cross-loadings I test the entirety of Model 2 with PLS-SEM. The reason for this is Model 2 includes every construct, meaning that I can generate cross-loadings for every item and every construct, not just the newly developed constructs. The second round of data collection will also test the models with PLS-SEM so this approach is appropriate and ensures consistency across rounds. Additionally, the CB-SEM module in SmartPLS does not produce cross-loading results, only loadings for the latent construct. For Round 1 I am not conducting hypotheses testing, the model is only calculated to produce the loadings and cross-loadings for scale evaluation. The loadings and cross-loadings are presented in Table C1.

None of the cross-loadings were greater than 0.7 with another construct. For convergent validity, four items were of concern. The fourth item for connectivity had a loading of 0.57, the first and second items for system capability shortcomings had loadings of 0.69 and 0.68, respectively, and the first item for system support availability had a loading of 0.49. For system capability shortcomings, I chose to keep those items as they are very close to the 0.7 threshold and capture important content validity for the constructs (MacKenzie et al., 2011). However, the item loadings for system support availability and connectivity were much lower, so they are candidates for deletion if the reliability of the constructs are low.

I looked at items that loaded higher than 0.6 with constructs other than the theorized construct. The items are summarized in Table 7. Perhaps the most surprising loadings are AD3 and CON3, loading somewhat high on system support availability. Adaptation and connectivity are technical characteristics of the artifact, whereas system support availability is a measure from the social subsystem. Reviewing the items, there are no clear candidates for adjustment, and in the card sort, participants separated these constructs, so I did not make any changes to these items.

Item	Other Construct Loading
AD3: The system is easily modified.	SSA (0.63) | TI (0.60)
CON3: External systems can connect to this system easily.	AD (0.63) | SSA (0.63)
INT3: System components are integrated.	ST (0.63)
RF1: When employees use the system, they find the content it provides them is sufficiently complete.	ST (0.62) | TI (0.65)
ST1: The system keeps records of events.	INT (0.62)
ST4: The system saves information between uses.	RF (0.61)
SSA2: We can easily obtain the support resources necessary to continue operating this system.	TI (0.61)
SSA3: Support for this system is readily available.	TI (0.61)
TI1: When employees use the system, they have seamless access to the content they need.	INT (0.61) | RF (0.65)
TI3: I have difficulty obtaining the content I need because of physical characteristics of the device.	RF (0.61)
AD = Adaptation, CON = Connectivity, INT = Integration, RF = Representational Fidelity, ST = State, SSA = System Support Availability, TI = Transparent Interaction	

Table 7: Greater than 0.6 Loadings on Other Constructs

Less surprising are the loadings of CON3 on adaptation, INT3 on state, and ST1 on integration. These are all characteristics of the IT artifact and are, at some level, expected to be related. Regarding integration and connectivity, it may be due to system integration projects often involving connecting information systems. However, as theorized here (Matook & Brown, 2017), integration refers to internal artifact structure. I reviewed these items and saw no obvious candidates for wording changes. In the card sort, there were two incidents where a connectivity item was incorrectly placed with the integration group, but this contrasted with the other 42 incidents where the items were placed correctly.

There were also somewhat high loadings between S4 and representational fidelity, RF1 and state and transparent interaction, TI1 and integration and representational fidelity, TI3 and representational fidelity, and SSA2/SSA3 with transparent interaction. The higher loadings with state and representational fidelity are somewhat expected, as one of the foundations of representation theory is the state tracking model (Wand & Weber, 1995). Similarly, the surface structure, which is being measured via the transparent interaction variable is assessing the ability to access a faithful representation (Burton-Jones & Grange, 2013; Wand & Weber, 1995), so these constructs are closely linked, similar to how the characteristics of the IT artifact are distinct but related.

The loading of system support availability and transparent interaction is less clear as system support availability is a measurement of the external social subsystem. One explanation is that both constructs use the word "obtain" in multiple items, but the context in which they are used is distinct. Since representational fidelity, transparent interaction, and system support availability are all adapted from existing scales, I have chosen not to alter the items.

Reliability

To test reliability, I use Cronbach's alpha (Cronbach, 1951), composite reliability (Petersen et al., 2013), and average variance extracted (AVE) (Henseler et al., 2009) to test the internal consistency of the newly developed and adapted constructs. Table 8 shows the initial reliability scores. Constructs marked with "*" are existing scales adapted from the literature.

For the newly developed scales for this research (Adaptation, Connectivity, Integration, Legacy Perception, and State) all measures cleared the .7 threshold. However, I dropped CON4 (*The system communicates over a network.*) as it lowered the reliability of the construct and was also problematic regarding the convergent validity test loading at only 0.57 on the construct.

Construct	Cronbach's alpha	Composite reliability (rho_a)	Composite reliability (rho_c)	Average variance extracted (AVE)
Adaptation	0.77	0.78	0.86	0.60
Complexity*	**0.67**	0.71	0.82	0.60
Connectivity	0.70	0.71	0.82	0.53
Integration	0.75	0.75	0.84	0.57
Legacy Perception	0.73	0.74	0.85	0.65
Replacement Intentions*	0.85	0.86	0.91	0.77
Representational Fidelity*	0.81	0.81	0.87	0.63
State	0.78	0.79	0.86	0.61
System Capability Shortcomings*	**0.67**	**0.66**	0.80	0.50
System Investment*	0.79	0.80	0.86	0.62
System Support Availability*	0.77	0.90	0.84	0.66
Transparent Interaction*	0.77	0.77	0.87	0.68

Table 8: Round 1 Initial Reliability Scores

For the adapted measures, three were somewhat problematic. Complexity was slightly below the .7 threshold at 0.67 for Cronbach's alpha, but passed on other metrics. System capability shortcomings had a Cronbach's alpha of 0.67 and also was below the composite reliability (rho a) at 0.66. Both of these constructs have been used in previous research (Furneaux & Wade, 2017) and are close enough to the .7 threshold that I decided to retain them as is. System support availability has also been used previously (Furneaux & Wade, 2017), and the reliability of the construct dramatically improves from .77 to .84 if SSA1 (*We do not encounter difficulties in obtaining needed system support services.*) is dropped. This, paired with the low loading of 0.49 on the construct, led me to drop the SSA1 item, although this does reduce system support availability to a two-item scale.

All other developed and adapted constructs had reliability above the recommended .7 threshold for reliability and .5 threshold for AVE. Updated reliability scores are reported in Table 9. Constructs marked with "*" are existing scales adapted from the literature.

Construct	Cronbach's alpha	Composite reliability (rho_a)	Composite reliability (rho_c)	Average variance extracted (AVE)
Adaptation	0.77	0.78	0.86	0.60
Complexity*	**0.67**	0.71	0.82	0.60
Connectivity	0.72	0.72	0.84	0.64
Integration	0.75	0.75	0.84	0.57
Legacy Perception	0.73	0.74	0.85	0.65
Replacement Intentions*	0.85	0.86	0.91	0.77
Representational Fidelity*	0.81	0.81	0.87	0.63
State	0.78	0.79	0.86	0.61
System Capability Shortcomings*	**0.67**	**0.66**	0.80	0.50
System Investment*	0.79	0.80	0.86	0.62
System Support Availability*	0.84	0.86	0.93	0.86
Transparent Interaction*	0.77	0.77	0.87	0.68

Table 9: Round 1 Updated Reliability Scores

Confirmatory Factor Analysis: New Scales

As a final measurement validity check, I conducted a CFA for the new constructs of adaptation, connectivity, integration, state, and legacy perception using the CB-SEM module in SmartPLS 4.1.0.0 (Ringle et al., 2024). I chose to use CB-SEM for the CFA as it produces model fit statistics useful for evaluating the proposed scales. The specified measurement model with standardized loadings is presented in Figure 10. Correlation relationships are omitted from Figure 10 for readability, but when statistically tested the correlations between each combination of constructs were modeled. The model was estimated using the maximum likelihood approach. Construct correlations are reported in Table 10. The standardized outer loadings are reported in Table 11.

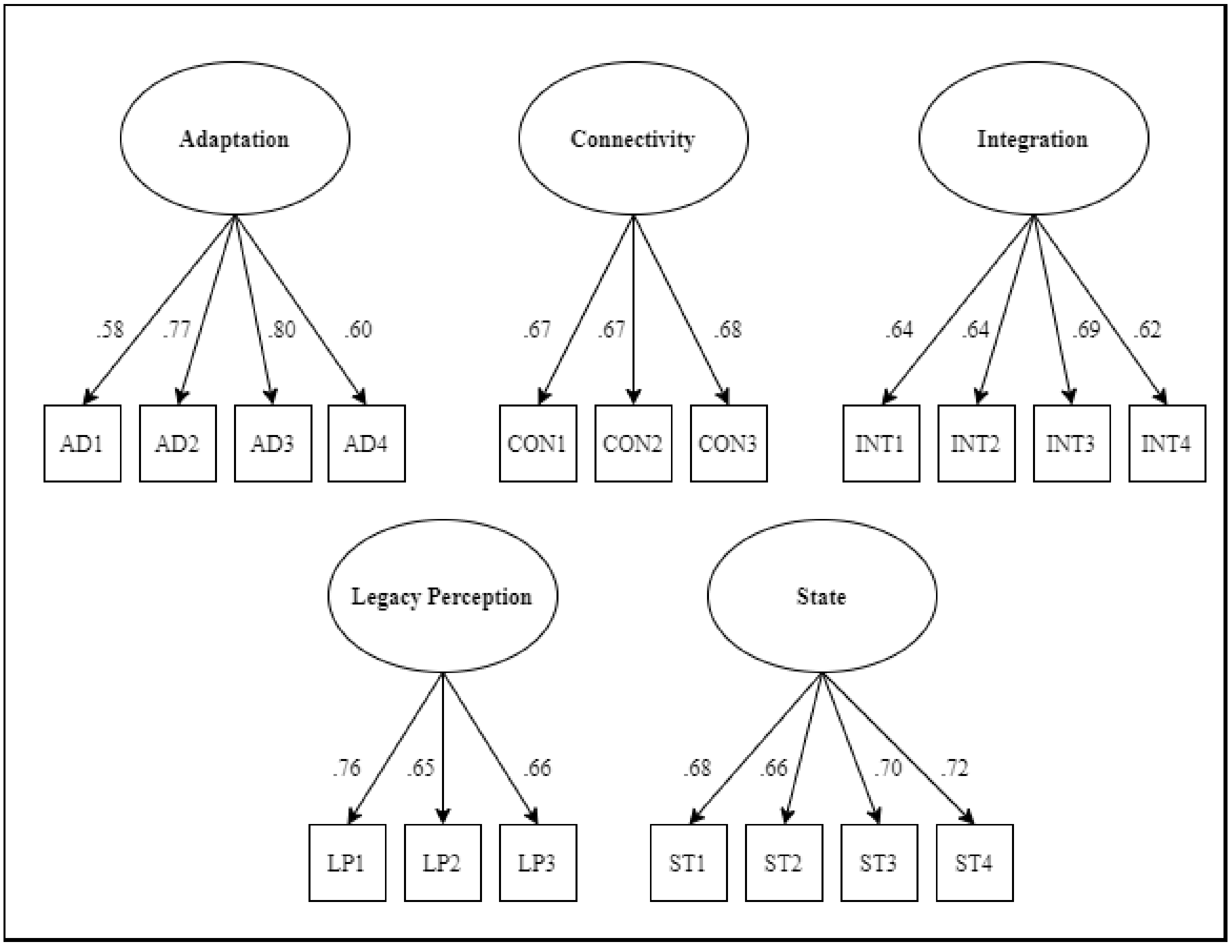

Figure 10: CFA Measurement Model

	Adaptation	**Connectivity**	**Integration**	**Legacy Perception**	**State**
Adaptation	**1.00**	0.85	0.71	0.09	0.69
Connectivity	0.85	**1.00**	0.89	0.31	0.84
Integration	0.71	0.89	**1.00**	0.47	0.93
Legacy Perception	0.09	0.31	0.47	**1.00**	0.34
State	0.69	0.84	0.93	0.34	**1.00**

Table 10: CFA Correlations

	Original sample	Sample mean	Standard deviation	T statistic	P value
AD1 <- Adaptation	0.58	0.59	0.08	7.72	< .001
AD2 <- Adaptation	0.77	0.76	0.07	11.20	< .001
AD3 <- Adaptation	0.80	0.79	0.05	16.46	< .001
AD4 <- Adaptation	0.60	0.60	0.07	8.56	< .001
CON1 <- Connectivity	0.67	0.68	0.07	10.35	< .001
CON2 <- Connectivity	0.67	0.67	0.07	9.86	< .001
CON3 <- Connectivity	0.68	0.68	0.06	12.22	< .001
INT1 <- Integration	0.64	0.64	0.06	11.79	< .001
INT2 <- Integration	0.64	0.64	0.05	13.58	< .001
INT3 <- Integration	0.69	0.69	0.04	17.65	< .001
INT4 <- Integration	0.62	0.62	0.05	11.53	< .001
LP1 <- Legacy Perception	0.76	0.76	0.06	13.39	< .001
LP2 <- Legacy Perception	0.65	0.65	0.06	10.95	< .001
LP3 <- Legacy Perception	0.66	0.66	0.06	11.82	< .001
ST1 <- State	0.68	0.67	0.05	13.27	< .001
ST2 <- State	0.66	0.66	0.05	12.84	< .001
ST3 <- State	0.70	0.70	0.04	17.15	< .001
ST4 <- State	0.72	0.71	0.04	17.15	< .001

Table 11: CFA Standardized Outer Loadings

The chi-square goodness of fit test was rejected $\chi^2 = 288.74$ $p < .001$, suggesting this factor model is poorly fit. However, the chi-square/df ratio of 2.31 (288.74/125) suggests a good fit and is less sensitive to sample size issues of the traditional chi-square test (Alavi et al., 2020; Wheaton et al., 1977). The Root Mean Square Error of Approximation of 0.08 with a 90% confidence interval of 0.065 to 0.089 suggests a good fit for this model. The Goodness of Fit Index and the Adjusted Goodness of Fit Index were 0.86 and 0.81, respectively, above the threshold of 0.8 for a good fit. The Comparative Fit Index was 0.9 matching the 0.9 threshold for good fit. The Standardized Root Mean Square Residual was 0.07, below the limit of 0.08, further supporting the model fit. The Normed Fit Index was 0.83 and the Tucker-Lewis Index was 0.87, which are close but slightly below the 0.9 threshold of good fit.

The standardized outer loadings of the CFA model were statistically significant and of acceptable fit, as presented in Figure 10 and Table 10. However, many loadings were in the 0.6 – 0.7 range, and one loading for adaptation was 0.58. The loadings are acceptable overall but not ideal. While the chi-square test hypothesis was rejected, the other fit indicators, convergent validity, discriminant validity, and construct reliability, suggest that the scales developed for this research are sufficiently validated.

Round 2: Model Testing

<u>Data Cleaning</u>

The survey was distributed to 351 participants for round two of data collection. Six of those 351 participants were filtered for having insufficient IT management experience. Thirty-one respondents were filtered for missing an attention check question in the survey. Three additional respondents were dropped for providing garbage text responses, and three were dropped for straightlining after data collection, leaving 308 valid responses. I then took these valid responses and did an additional check for outliers. For univariate outliers, I calculated *z*-scores for each indicator of the model constructs. Sixteen indicators had at least one respondent with a *z*-score outside the +/-3.29 range (Tabachnick & Fidell, 2018, p. 64). In total, 22 responses were removed for containing univariate outliers.

My next step was to check for multivariate outliers for all constructs that had multiple indicators (all constructs except for system age). Items for each construct were averaged into single measures. The covariance matrix of these measures for all responses was used to calculate the Mahalanobis distance measure (Mahalanobis, 1936) to determine the Mahalanobis distance probability. Responses with a Mahalanobis distance probability of less than 0.001 were considered multivariate outliers and dropped from the data analysis (Tabachnick & Fidell, 2018,

p. 84). In total, 15 responses were identified. However, six were already identified by the univariate outlier analysis, leaving nine multivariate outliers removed. In total, 31 outliers were removed, leaving 277 responses for final data analysis.

Descriptive Statistics

32.6% of respondents were women (*n* = 90), 67.0% of respondents were men (*n* = 185), no respondents were nonbinary or other gender identities, and there were two non-responses (0.4%). The average respondent age was 40.66, ranging from 22 to 69. 4.3% (*n* = 12) of respondents worked in lower management, 33.5% (*n* = 74) in middle management, 45.2% (*n* = 100) in upper management, 14% (*n* = 31) as a chief information officer, and 4.1% (*n* = 9) as chief information security officers. The age of the legacy systems ranged from 2 to 70 years, with an average system age of 20.1 years.

To check that the data are normally distributed, I conducted skewness and kurtosis tests. Specifically, the constructs in the research model were checked to establish that their skewness and kurtosis values were between -1 and +1 to establish normality. The test was conducted on the averaged single-measure versions of the constructs. None of the constructs displayed skewness or kurtosis, except system age, which exhibited both. System age had a skewness of 1.58 (SE = 0.15) and a kurtosis of 1.90 (SE = 0.29), suggesting a leptokurtic distribution with a positive skew (Cuttler, 2014). This matches what was found in Round 1 of data collection. However, replacement intentions did not have skewness or kurtosis issues despite exhibiting this in Round 1. The positive skew for system age is not entirely unexpected, as the types of systems considered legacy in organizations typically are incumbent pre-existing systems that have been implemented for a significant amount of time. Overall descriptive statistics are presented in Table 12.

Construct	Mean	Std. Error	Std. Deviation	Variance	Skewness	Std. Error	Kurtosis	Std. Error
Adaptation	4.83	0.08	1.28	1.64	-0.44	0.15	-0.60	0.29
Complexity	5.30	0.06	1.03	1.07	-0.55	0.15	-0.20	0.29
Connectivity	5.25	0.06	1.00	0.99	-0.48	0.15	-0.24	0.29
Integration	5.36	0.06	0.97	0.95	-0.73	0.15	0.46	0.29
Legacy Perception	5.88	0.05	0.88	0.77	-0.58	0.15	-0.23	0.29
Replacement Intentions	5.51	0.06	1.05	1.09	-0.67	0.15	0.33	0.29
Representational Fidelity	5.12	0.07	1.17	1.36	-0.69	0.15	-0.01	0.29
State	5.54	0.06	0.99	0.98	-0.92	0.15	0.87	0.29
System Age	20.10	0.93	15.53	241.12	**1.58**	0.15	**1.90**	0.29
System Capability Shortcomings	5.27	0.06	0.99	0.98	-0.53	0.15	-0.28	0.29
System Investment	5.57	0.05	0.87	0.76	-0.49	0.15	-0.02	0.29
System Support Availability	4.60	0.08	1.38	1.90	-0.30	0.15	-0.73	0.29

Table 12: Round 2 Descriptive Statistics

<u>Convergent and Discriminant Validity</u>

The loadings and cross-loadings for both stages of Model 1 are presented in Table C2 and Table C3. For Model 2, loadings and cross-loadings are presented in Table C4. The same convergent and discriminant validity issues were observed in Model 1 Stage 1 and Model 2. There were only minor differences in loadings due to the addition of state and first-order legacy perception measures to Model 2. Since Model 2 uses every construct and both models use the same dataset, I will primarily be evaluating the loadings and cross-loadings from Model 2 (Table C4) and Model 1 Stage 2 (Table C3).

For Model 1 Stage 2 replacement intentions and system investment both showed strong convergent validity. The indicators for replacement intentions ranged from 0.81 to 0.85. The indicators for system investment ranged from 0.78 to 0.82. Discriminant validity was also strong. All items except for one cross-loaded lower than 0.6. The one item was from the second order formative construct for legacy perception. The latent variable score for system capability shortcomings item cross-loaded at 0.65 on replacement intentions.

For Model 2, Indicators exhibited strong convergent validity to their construct, with all but and CON4 (0.67) and SCS4 (0.66) passing the 0.7 threshold for loadings. However, discriminant validity continues to be an issue with these constructs. SSA2 loaded at 0.73 on transparent interaction, and TI1 loaded at 0.71 on representational fidelity. I conducted this analysis again by lowering the threshold from 0.7 to 0.6, and the discriminant validity issues became more apparent. Potentially problematic indicators are highlighted in Table C4.

Two of the connectivity items (CON1 and CON3) load on adaptation. Two integration items (INT2 and INT3), three representational fidelity items (RF1, RF2, RF3), one system support availability item (SSA2), and one transparent interaction item (TI1) loaded on connectivity. Three representational fidelity (RF1, RF2, RF3) and one state item (S1) loaded on integration. Two connectivity items (CON1 and CON3), two integration items (INT2, INT3), one system support availability item (SSA2), and all three transparent interaction items are loaded on representational fidelity. One integration item (INT2) loaded on state. One adaptation item (A3), one connectivity item (CON3), and one transparent interaction item (TI1) loaded on system support availability. Lastly, one connectivity item (CON3), all four representational fidelity items, and two system support availability items (SSA2 and SSA3) are loaded on transparent interaction.

This suggests that although most items were unproblematic when using a 0.7 cutoff for discriminant validity, many indicators highly correlate with other constructs. At some level, this is expected. Especially with the IT artifact characteristics and representation theory constructs, as they all describe aspects of the information system and have some necessary theoretical overlap.

However, the amount of high cross-loadings remains a valid concern, so I ran an additional test of discriminant validity using the heterotrait-monotrait (HTMT) ratio matrix (Henseler et al., 2015) presented in Table 13.

	AD	**CX**	**CON**	**INT**	**LP**	**RI**	**RF**	**ST**	**SA**	**SCS**	**SI**	**SSA**	**TI**
Adaptation													
Complexity	0.52												
Connectivity	0.87	0.73											
Integration	0.64	0.78	**0.95**										
Legacy Perception	0.15	0.27	0.24	0.24									
Replacement Intentions	0.14	0.21	0.15	0.18	0.66								
Representational Fidelity	0.74	0.63	**0.91**	0.85	0.09	0.07							
State	0.65	0.73	0.85	0.87	0.36	0.18	0.76						
System Age	0.09	0.10	0.10	0.12	0.03	0.07	0.13	0.09					
System Capability Shortcomings	0.22	0.22	0.20	0.18	0.55	0.87	0.12	0.22	0.06				
System Investment	0.16	0.57	0.48	0.50	0.58	0.34	0.44	0.54	0.12	0.40			
System Support Availability	0.80	0.56	0.79	0.61	0.15	0.19	0.85	0.57	0.19	0.09	0.27		
Transparent Interaction	0.74	0.66	0.86	0.78	0.08	0.08	**0.97**	0.67	0.12	0.08	0.45	**0.96**	

Table 13: Model 2 HTMT Matrix

An HTMT value of >.90 indicates that there is not sufficient discriminant validity between the two constructs. Using this metric, there were discriminant validity issues between integration and connectivity (0.95), representational fidelity and connectivity (0.91), representational fidelity and transparent interaction (0.97), and system support availability and transparent interaction (0.96). Based on this and the problematic cross-loadings, I determined that I could not keep all the variables in the PLS-SEM models.

Ultimately, I chose to remove connectivity and transparent interaction from the models. Connectivity was removed because it could not be distinguished between two different constructs, representational fidelity, and integration. Representational fidelity is second to only legacy perception in theoretical importance to my arguments, so it makes more sense to keep it.

The connectivity and integration discriminant validity issue could be a disconnect between how integration is used colloquially in software development (i.e., building connections between systems) versus how Matook and Brown (2017) conceptualize it as a characteristic of internal artifact structure. Issues between connectivity and representational fidelity are less clear. Conceptually, there is no clear overlap. It could be related to the need for connectivity to mediate and represent digital reality interactions in representations (Recker et al., 2021), meaning it may already be captured (partially) by the representational fidelity items.

The other variable I removed was transparent interaction. While theoretically distinct, it was statistically indistinguishable from representational fidelity and system support availability. I believe the issue here is that a user determines if a system is representationally faithful by evaluating outputs from the surface structure (Burton-Jones & Grange, 2013). While a strict reading of representation theory argues that representational fidelity of the deep structure can be evaluated independently of use (Wand & Weber, 1995), other scholars have argued it is a socio-technical use characteristic (Burton-Jones & Grange, 2013). Since transparent interaction measures the ability of users to evaluate deep structure representations through surface structures, the two measures might be inseparable in the context of behavioral research.

As a result of dropping connectivity and transparent interaction constructs, some of the hypotheses in Models 1 and 2 had to be removed. For Model 1, H2 (Non-connectivity → Legacy Perception) and H6 (Non-transparent Interaction → Legacy Perception) were dropped. For Model 2, H1 (Connectivity → Representational Fidelity), H3 (Connectivity → Adaptability), H8 (Adaptation → Transparent Interaction), H10 (Transparent Interaction → System Capability

Shortcomings), and H14 (Transparent Interaction → Legacy Perception) were removed. I kept the remaining hypotheses numbering the same for consistency throughout the manuscript chapters. Updated models are presented in Figures 11 and 12.

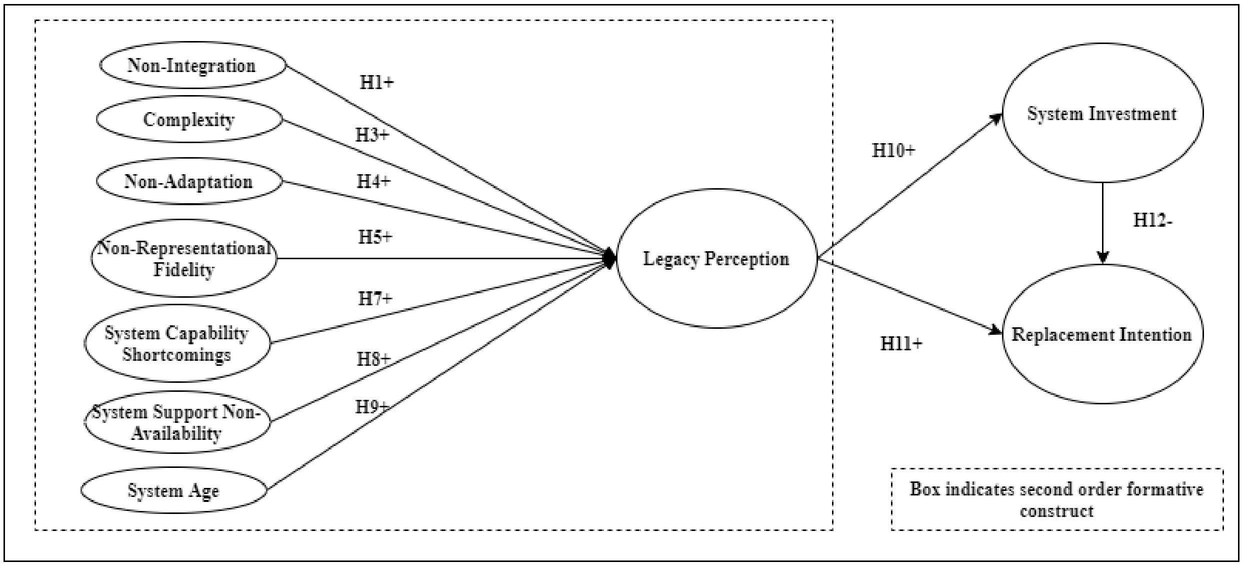

Figure 11: Updated Model 1 – Second-order Formative

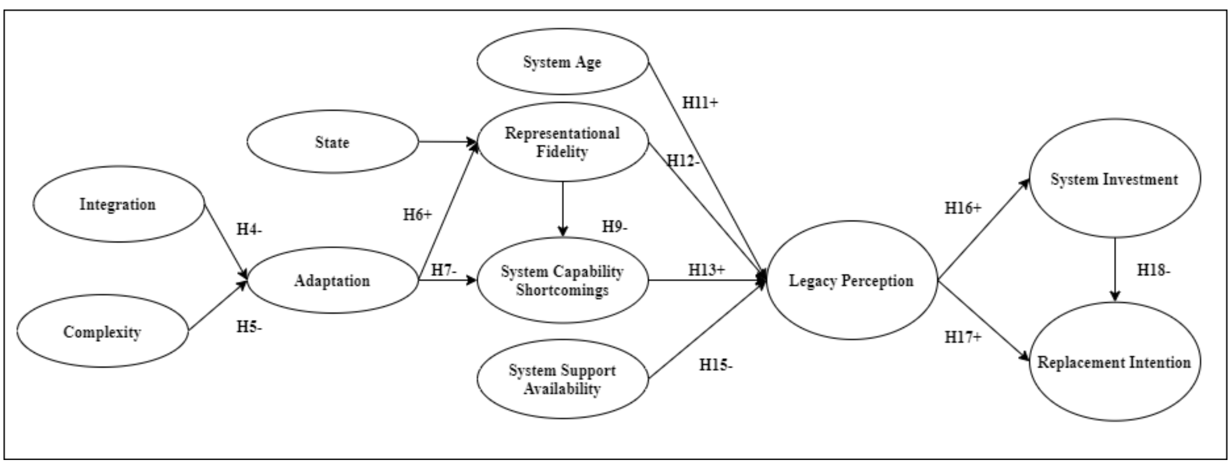

Figure 12: Updated Model 2 – First-order Reflective

To further assess construct-level discriminant validity I report the correlations between constructs for both models. Table 14 includes the correlations and square root of the AVE for Model 1 Stage 2. The same statistics are reported for Model 2 in Table 15. The bolded diagonal values are the square root of the AVE. The square roots of the AVEs were higher than the correlations with other constructs. Legacy perception in Model 1 was not included as it is a formative construct. This provides additional support for discriminant validity of the constructs in Model 2, but suggests a construct level discriminant validity issue with Legacy Perception in Model 1 (Fornell & Larcker, 1981). Legacy perception items in Model 1 will be discussed in the structural model section.

	Age	**Gender**	**LP**	**RI**	**SI**
Age	**1.00**				
Gender	-0.05	**1.00**			
Legacy Perception	-0.09	0.04	**NA**		
Replacement Intentions	0.03	0.00	0.63	**0.83**	
System Investment	-0.05	0.05	0.51	0.28	**0.80**

Table 14: Model 1 Stage 2 Construct Correlations

	AD	**Age**	**CX**	**Gen**	**INT**	**LP**	**RI**	**RF**	**ST**	**SA**	**SCS**	**SI**	**SSA**
Adaptation	**0.79**												
Age	-0.22	**1.00**											
Complexity	0.43	-0.13	**0.81**										
Gender	0.10	-0.05	0.08	**1.00**									
Integration	0.53	-0.09	0.62	0.02	**0.79**								
Legacy Perception	-0.02	0.00	0.17	-0.01	0.16	**0.83**							
Replacement Intentions	0.02	0.03	0.15	0.00	0.12	0.52	**0.83**						
Representational Fidelity	0.62	-0.14	0.51	0.07	0.71	0.01	-0.01	**0.83**					
State	0.53	-0.12	0.59	0.04	0.72	0.29	0.15	0.65	**0.82**				
System Age	0.08	0.03	0.08	-0.05	0.11	0.02	0.00	0.12	0.08	**1.00**			
System Capability Shortcomings	0.10	-0.08	0.15	0.04	0.13	0.44	0.65	-0.03	0.18	0.05	**0.73**		
System Investment	0.12	-0.05	0.41	0.05	0.39	0.46	0.27	0.36	0.45	0.11	0.34	**0.80**	
System Support Availability	0.62	-0.12	0.42	0.07	0.48	-0.14	-0.16	0.66	0.42	0.17	-0.07	0.17	**0.83**

Table 15: Model 2 Construct Correlations

Reliability

To test reliability, I use Cronbach's alpha (Cronbach, 1951), composite reliability (Petersen et al., 2013), and average variance explained to test the internal consistency of the newly developed and adapted constructs. All constructs had a reliability above the recommended .7 threshold. Although dropping SSA1 increases reliability from .80 to .83 for the system support availability scale. I kept the existing items since the scales were above the .7 cutoff. I dropped SSA1 in Round 1 because it also exhibited convergent validity issues. In Round 2 the convergent validity issues were less prominent, making it not necessary to drop the item. Reliability scores are reported in Table 16 for Model 1 Stage 2 and Table 17 for Model 2. Scales marked with "*" are existing scales adapted from the literature.

Construct	**Cronbach's alpha**	**Composite reliability (rho_a)**	**Composite reliability (rho_c)**	**Average variance extracted (AVE)**
Replacement Intentions*	0.77	0.78	0.87	0.69
System Investment*	0.81	0.82	0.88	0.64

Table 16: Model 1 Stage 2 Reliability

Construct	**Cronbach's alpha**	**Composite reliability (rho_a)**	**Composite reliability (rho_c)**	**Average variance extracted (AVE)**
Adaptation	0.80	0.80	0.87	0.63
Complexity*	0.74	0.83	0.85	0.65
Integration	0.81	0.83	0.87	0.63
Legacy Perception	0.77	0.78	0.87	0.69
Replacement Intentions*	0.77	0.78	0.87	0.69
Representational Fidelity*	0.85	0.85	0.90	0.69
State	0.84	0.84	0.89	0.68
System Capability Shortcomings*	0.71	0.73	0.82	0.53
System Investment*	0.81	0.82	0.88	0.64
System Support Availability*	0.80	0.92	0.87	0.69

Table 17: Model 2 Reliability

Common Method Bias

As an additional test of the measurement model validity, I checked for common method bias in both Model 1 and Model 2. This tests for variance attributable to the method, rather than the construct interactions (Podsakoff et al., 2003, 2024). For this test, I followed the guidelines of Kock (2015) by checking that variance inflation factors (VIF) statistics for the inner model are less than 3.3. Neither model had a VIF value > 3.3, with the highest VIF for Model 1 being 1.35 and the highest VIF for Model 2 being 1.80. Based on this test I do not find evidence for common method bias. However, there is disagreement on the validity of VIF measures (Kalnins & Praitis Hill, 2023). I did not collect marker variables so I was not able to do an additional test for common method bias (W. W. Chin et al., 2013). Model 1 Stage 2 VIF statistics are reported in Table 18, and Model 2 VIF statistics are reported in Table 19.

Relationship	**VIF**
Age -> Legacy Perception	1.00
Gender -> Legacy Perception	1.00
Legacy Perception -> Replacement Intentions	1.35
Legacy Perception -> System Investment	1.00
System Investment -> Replacement Intentions	1.35

Table 18: Model 1 Stage 2 VIF Statistics

Relationship	VIF
Adaptation -> Representational Fidelity	1.39
Adaptation -> System Capability Shortcomings	1.61
Age -> Legacy Perception	1.03
Complexity -> Adaptation	1.62
Gender -> Legacy Perception	1.01
Integration -> Adaptation	1.62
Legacy Perception -> Replacement Intentions	1.28
Legacy Perception -> System Investment	1.00
Representational Fidelity -> Legacy Perception	1.78
Representational Fidelity -> System Capability Shortcomings	1.61
State -> Representational Fidelity	1.39
System Age -> Legacy Perception	1.04
System Capability Shortcomings -> Legacy Perception	1.02
System Investment -> Replacement Intentions	1.28
System Support Availability -> Legacy Perception	1.80

Table 19: Model 2 VIF Statistics

Structural Model 1 Results

Model 1 was evaluated using a two-stage approach known as the extended repeated indicators approach (J. F. Hair Jr. et al., 2022; Sarstedt et al., 2019). Stage 1 of the model is only used to generate latent variable scores to be used as indicators in Stage 2. However, for completeness, the results of the Stage 1 model are available in Table C5. No bootstrapping is done for hypothesis testing at this stage, as hypothesis testing is conducted as part of the Stage 2 model.

For stage 2, to test the structural model, the significance of each hypothesized relationship was tested with a bootstrapping method using 5,000 sub-samples (J. Hair Jr et al., 2017). The tests were one-tailed, as directionality was specified in the theoretical model. The results of this model are presented in Figure 13.

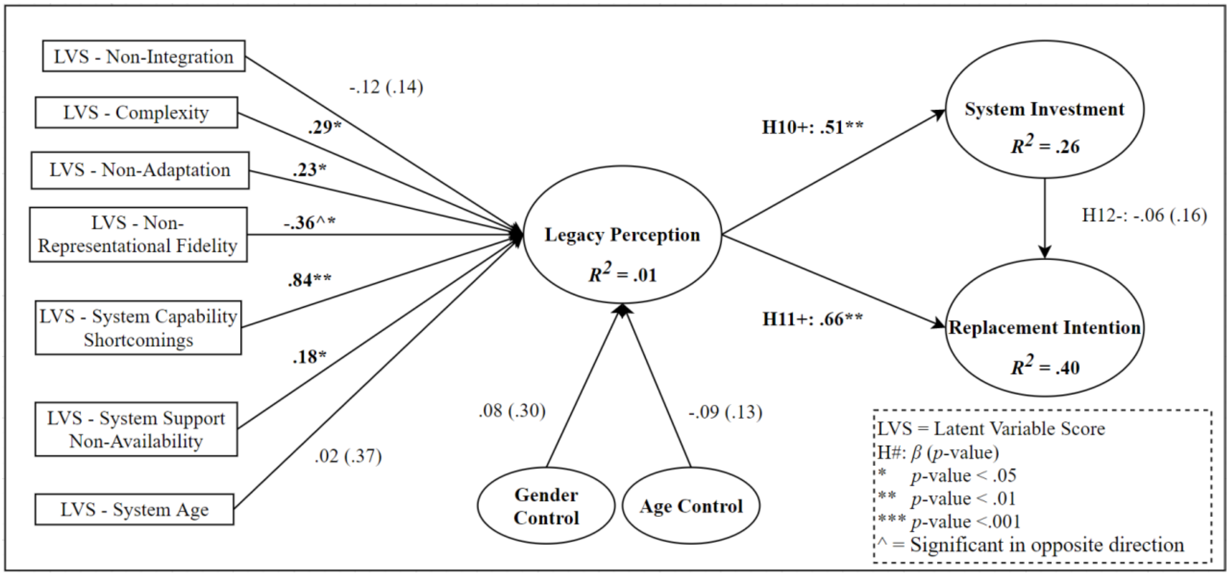

Figure 13: Model 1 Stage 2 Results

The results show support for H10 (β = .51, p = .006) and H11 (β = .66, p = .002). However, H12 was not supported (β = -.06, p = .16). The control variables of gender and age of the participant were not statistically significant. For the latent variable score indicators of legacy perception the weights for system age H9 (β = .02, p = .37) and non-integration H1 (β = -.12, p = .14) were not statistically significant. A lack of weight statistical significance is not enough on its own to justify removing an indicator from a formative model as there are concerns of undermining content validity of the construct, however (J. F. Hair Jr. et al., 2021a).

Non-representational fidelity was statistically significant, but negative H5 (β = -.36, p = .015). The other weights for complexity H3 (β = .29, p = .013), non-adaptation H4 (β = .23, p = .04), system capability shortcomings H7 (β = .84, p = .002), and system support non-availability H8 (β = .18, p = .046) were statistically significant. The overall explained variance for system investment was R^2 = .258, and for replacement intention, it was R^2 = .397. Legacy perception had a variance explained of R^2 = .01. However, this was only from the control variables since the other first-order reflective constructs were used as indicators in the model. Legacy perception indicators are summarized in Table 20.

Hypotheses	**Weights**	**Loadings**	**Supported?**
H1+ Non-Integration	-0.12 (.14)	-0.45 (.01)	No
H3+ Complexity	0.29 (.013)	0.50 (.007)	**Yes**
H4+ Non-Adaptation	0.23 (.04)	-0.14 (.11)	**Yes**
H5+ Non-Representational Fidelity	-0.36 (.015)	-0.29 (.03)	*Inverse*
H7+ System Capability Shortcomings	0.844 (.002)	0.88 (.002)	**Yes**
H8+ System Support Non-Availability	0.178 (.046)	-0.06 (.27)	**Yes**
H9+ System Age	0.016 (.37)	0.08 (.14)	No
Untested Hypotheses Due to Discriminant Validity Issues			
H2+ Connectivity			
H6+ Non-transparent Interaction			

Table 20: Model 1 Stage 2 Hypotheses

Structural Model 2 Results

Since all constructs in Model 2 were first-order reflective, a single PLS-SEM model was sufficient to evaluate the structural model. The significance of each hypothesized relationship was tested with a bootstrapping method using 5,000 sub-samples (J. Hair Jr et al., 2017). The tests were one-tailed, as directionality was specified in the theoretical model. The results of this model are presented in Figure 14.

The results show support for H3 (β = .45, p <. 001), H6 (β = .38, p < .001), H13 (β = .44, p < .001), H15 (β = -.22, p = .003), H16 (β = .46, p < .001), and H17 (β = .50, p < .001). Four hypotheses were significant but in opposite directions. Since these were one-tailed tests, they do not support the theorized model but suggest there are relationships between the specified constructs. This was the case for H4 (β = .42, p < .001), H5 (β = .17, p < .001), H7 (β = .19, p = .02), and H12 (β = .17, p = .004). The remaining hypotheses H9 (β = -.15, p = .07), H11 (β = .02, p < .37), H18 (β = .04, p < .26), and the control variable relationships of gender and participant age with legacy perception were not statistically significant. The path coefficients for complexity to adaptation, representational fidelity to legacy perception, and adaptation to system capability shortcomings were slightly below the .2 threshold used for the sample size calculations (W. Chin, 1998; Kock & Hadaya, 2018).

The overall explained variance for adaptation was R^2 = .29. For representational fidelity, was R^2 = .52. For system capability shortcomings, it was R^2 = .02. For legacy perception, was R^2 = .23. For system investment, R^2 = .22, and for replacement intentions, R^2 = .27. Table 21 summarizes the results of both models. Discussion of these results and their implications will be covered in the next chapter.

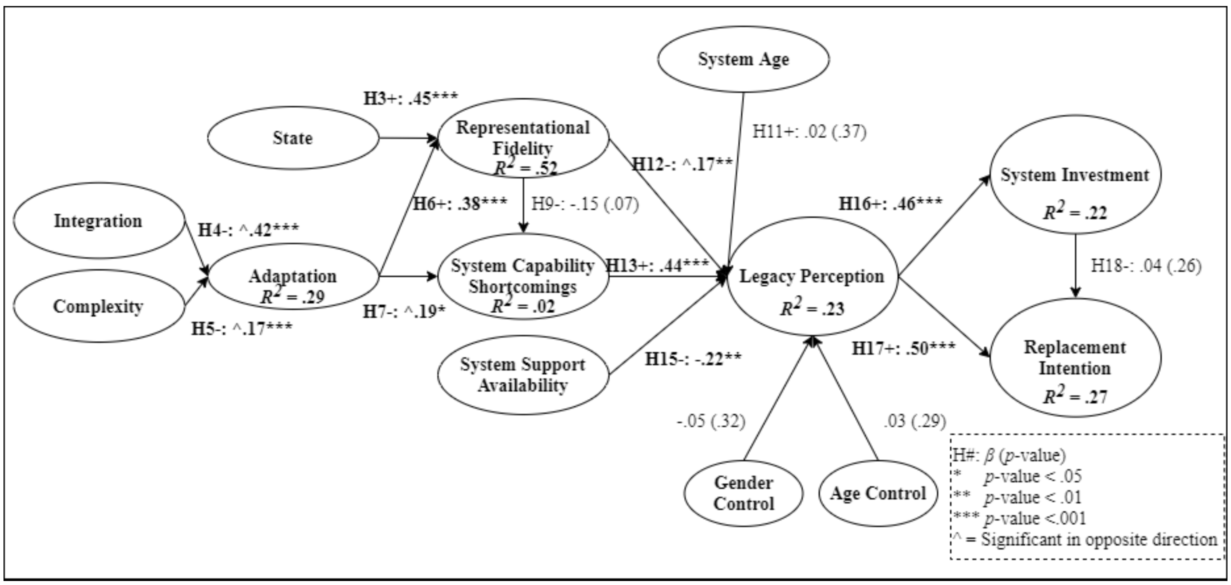

Figure 14: Model 2 Results

Hypotheses	Model 1 - Stage 2 Coeff. (*p*-value)	Model 2 Coeff. (*p*-value)	Supported?
H3+ ST→ RF	[redacted]	.45 (<.001)	**Yes**
H4- INT → AD	[redacted]	.42 (<.001)	*Inverse*
H5- CX → AD	[redacted]	.17 (<.001)	*Inverse*
H6+ AD → RF	[redacted]	.38 (<.001)	**Yes**
H7- AD → SCS	[redacted]	.19 (.02)	*Inverse*
H9- RF → SCS	[redacted]	-.15 (.07)	No
H11+ SA → LP	[redacted]	.02 (.37)	No
H12- RF → LP	[redacted]	.17 (.004)	*Inverse*
H13+ SCS → LP	[redacted]	.44 (<.001)	**Yes**
H15- SSA → LP	[redacted]	-.22 (.003)	**Yes**
H16+ [M1: H10+] LP → SI	.51 (.006)	.46 (< .001)	**Yes**
H17+ [M1: H11+] LP → RI	.66 (.002)	.50 (<.001)	**Yes**
H18- [M1: H12-] SI → RI	-.06 (.16)	.04 (.26)	No
Controls	**Coeff. (*p*-value)**	**Coeff. (*p*-value)**	**Significant?**
Age → LP	-.09 (.13)	.03 (.29)	No
Gender → LP	.08 (.30)	-.05 (.32)	No
Untested Hypotheses Due to Discriminant Validity Issues			
H1+ CON → RF			
H2+ CON → AD			
H8+ AD → TI			
H10- TI → SCS			
H14- TI → LP			

AD = Adaptation, CX = Complexity, CON = Connectivity, INT = Integration, LP = Legacy Perception, RF = Representational Fidelity, RI = Replacement Intentions, ST = State, SCS = System Capability Shortcomings, SI = System Investment, SSA = System Support Availability, TI = Transparent Interaction

Table 21: Hypotheses Summary Models 1 and 2

<u>Post hoc Test: System Age</u>

The non-significance of system age in Model 2 was an unexpected result. Since this measure also had issues with skewness and kurtosis, I investigated whether this was a potential cause of the non-significant result. I transformed the system age indicator using the base-10 logarithmic approach (West, 2022). This improved the skewness value from 1.58 to 0.10 and the kurtosis value from 1.90 to -0.18, resulting in a normal distribution.

I then re-calculated each PLS-SEM structural and bootstrap model with the transformed system age values. Model 2 remained insignificant with the transformed system age values (β = .05, p = .16). This suggests that this insignificant result is not due to the non-normal distribution of the indicator.

<u>Post hoc Test: Representational Fidelity</u>

Another unexpected result was the positive relationship between representational fidelity and legacy perception. I investigated this further to check for potential suppression effects. I first looked at the correlations for the four antecedents of legacy perception (representational fidelity, system capability shortcomings, system support availability, and system age) presented in Table 21.

	LP	RF	SA	SCS	SSA
Legacy Perception	**1.00**	0.01	0.02	0.44	-0.14
Representational Fidelity	0.01	**1.00**	0.12	-0.03	0.66
System Age	0.02	0.12	**1.00**	0.05	0.17
System Capability Shortcomings	0.44	-0.03	0.05	**1.00**	-0.07
System Support Availability	-0.14	0.66	0.17	-0.07	**1.00**

Table 22: Legacy Perception Antecedent Correlations

Most notable was the weak correlation between legacy perception and representational fidelity at 0.01. There was also a slight negative correlation with representational fidelity and system capability shortcomings, and a high correlation with system support availability. Running Model 2 with those two variables removed did not flip the sign negative for the hypothesized relationship. From this I conclude there were no suppression effects.

CHAPTER SEVEN: DISCUSSION

This chapter consists of five primary sections. First, I will discuss the results of the two models. I will then discuss theoretical contributions. This will be followed by future research opportunities and limitations. The next section will discuss implications of this research for practice. The chapter will close with a conclusion to the dissertation.

Discussion

The discussion will focus on four primary sections. The first section will focus on Research Question 1:*What socio-technical factors result in the formation of a legacy perception of an information system?* I will discuss the results of Model 2 focusing on the direct influences on legacy perception. The second section will focus on Research Question 2: *How does a legacy perception of a system impact replacement intentions and investment in an information system?* In this section I will explore the impacts of legacy perception on system investment and replacement intentions. I will then discuss the system characteristics relationships and their influence on the antecedents of the factors related to legacy perception.

Following this section will be a discussion of legacy perception itself as a construct. I will discuss the findings of Model 1 and Model 2 on formulating legacy perception as a second-order formative construct versus a first-order reflective construct. The discussion section will close with a broader discussion on measuring technical characteristics in behavioral research, focusing on the newly developed scales of adaptation, connectivity, integration, and state.

<u>Influences on Legacy Perception</u>

Model 2 posited four direct effects on legacy perceptions, from system capability shortcomings, system support availability, system age, and representational fidelity. The data supported two of these hypothesized relationships. System support availability was negatively related to legacy perceptions and system shortcomings was positively related. Representational fidelity was positively related to legacy perceptions, in opposition to my hypothesis, and system age was non-significant.

The factor that had the most influence on legacy perception was system capability shortcomings. This was the only factor that a high weight on legacy perception in Model 1 and was also significant in Model 2. The confirms that a lack of essential business functionality (Brooke, 2002; Brooke & Ramage, 2001; Kelly et al., 1999; Pang, 2017) and reduced capabilities and performances (Furneaux & Wade, 2017; Pang, 2017) are an important influence on a system being perceived as legacy. Similarly, system support availability was negatively associated with legacy perception. If system capability shortcomings are an internal signal to the organization that a system may be approaching legacy status, system support availability can be seen as an external signal from the environment as vendors drop support (Furneaux & Wade, 2017).

An unexpected result of influences on legacy perception was the relationship between representational fidelity and legacy perception. I hypothesized that this relationship would be negative, meaning that a system perceived as legacy would have a poor deep structure representation of reality. Instead, I found that representational fidelity positively influenced legacy perception. This is surprising as a high representational fidelity implies a more useful system (Burton-Jones & Grange, 2013; Wand & Weber, 1995), but a legacy system is often theorized to be misaligned with business needs (Gibson et al., 1998; Kelly et al., 1999).

In this case, the system accurately models the business process, and the IT manager is more likely to perceive the system as legacy. This suggests that perhaps legacy is not an inherently negative designation. It could be the case that this system has only remained in place because of the value it provides as a successful system (Gholami et al., 2017; Light, 2003; A. J. O'Callaghan, 1999) and a successful system relies on a faithful representation (Burton-Jones & Grange, 2013; Wand & Weber, 1995). Based on the rest of the results in this section, I find this explanation somewhat lacking. This would make more sense if the other relationships with legacy perception had been inverted. Then it would be a clear indicator that legacy is a positive, or at least neutral. Instead, the system capability shortcomings and support availability results strongly suggest that the legacy system fails to meet the organization's needs. Further study of this relationship and its implications for representation theory is needed to draw a conclusive explanation. A post-hoc test of the four antecedents to legacy perception did not show any suppressor effects for representational fidelity.

Perhaps the most interesting result for influences on legacy perception was the insignificance of system age. When developing these models, my concern was that system age would constitute the majority of variance explained by legacy perception, with the other theorized relationships being marginal at best. I conducted a post hoc test with a transformed measure of system age to see if the kurtosis and skewness issues were the culprit, yet the nonsignificant result persists.

The reason this is surprising is that the notion of legacy systems being old, obsolete systems is perhaps the most commonly held view across the literature (e.g., Azadmanesh & Peak, 1995; Bennett, 1995; Bisbal et al., 1999; Chirathamjaree, 2006; Mahapatra & Lai, 1998; Mallampalli & Karahanna, 2017; Tsai et al., 2022). I am hesitant to draw significant conclusions from a single study in the face of this consensus, but I think it suggests that the relationship between legacy and system age is significantly more complex than expected. One of the things that makes legacy systems a unique phenomenon is the temporal aspects (Light, 2003), and this result further highlights how nuanced that temporality is.

I do not think I would go as far as to say every system is already a legacy system (Light, 2003) or that age is not a factor in legacy (H. M. Edwards et al., 1999), but perhaps there is not a linear relationship between system age and legacy perception. Instead, this could be very contextual to the unique organizational context in which a system is implemented. The industry, type of system, stability of the business process, and many other things could influence how relevant the system age is in a situation. I also think it may not be age itself that is a factor, but system age is a proxy for all things that shift in the social and technical environment the longer a system is implemented.

However, an uncontrollable factor is removed from consideration if system age is insignificant in legacy perception formation. Managers who operate in three-dimensional space cannot control the flow of time. If system age is irrelevant in legacy perception formation, managers can focus on more feasibly addressable factors.

Legacy Perception Implications

For both Model 1 and Model 2, the relationship between legacy perception and increased system investment was significant. The same was true for the positive relationship between legacy perception and replacement intentions. Replacement intention and system investment have been studied in the context of barriers to system replacement (Furneaux & Wade, 2017), but this is the first research study to investigate how the legacy perception of a system influences these variables. My hypotheses assume that an organization will often continue to invest in these systems even though they simultaneously wish to replace them (Rinta-Kahila et al., 2023). This is partly because additional resources and investment are necessary for a systems replacement scenario to effectively discontinue the system (Mehrizi et al., 2012, 2019). These results also suggest that IT managers see the legacy label as a signal that the system should be replaced. This makes sense since the influences on legacy perception were things like a system lacking essential capabilities and vendor support not existing for the system.

The relationship between system investment and replacement intention was not significant. This was theorized to be negative, assuming that high system investment would make organizations more cautious about replacing a system due to the increased risk (Rinta-Kahila et al., 2023). The lack of significance suggests support for Furneaux and Wade's (2017) work on system replacement. They do not theorize a direct link between the two but rather suggest that system investment is only relevant to replacement intention in how it manifests as replacement risk and system complexity. Based on this research, the direct link between system investment and replacement intention is not supported.

<u>System Characteristics Influence on Legacy Perception Antecedents</u>

To start this section, I will discuss the results of the strictly physical structure interactions. This includes the relationship between integration and adaptation and complexity and adaptation. These IT artifact characteristics were hypothesized to have a negative relationship with adaptation, but the results show the opposite.

Integration was hypothesized to have a negative effect on adaptation based on extant literature arguing that modular systems are easier to maintain (Jermaine, 1999; Vestues & Knut, 2019; T. A. Wiggerts, 1997) since components could more easily be changed or replaced without affecting the rest of the system architecture. This inverse result could suggest that the functionalities and subsystem synergies present in a highly integrated artifact (Matook & Brown, 2017) make changing the system less difficult than expected. A system consisting of disparate module parts might be more difficult to manage, partly due to the cognitive load of understanding all the different modules (K. Lee et al., 2022; Scandura, 1994). While theoretically, a modular architecture should be easier to modify, the monolithic nature of a highly integrated system may be easier to understand and make changes to.

Much less clear is the fact that complexity had a positive relationship with adaptation. Of course, some level of complexity is required in every system (Moseley & Marks, 2006), especially when modeling a complex real business process is necessary (Wand & Weber, 1995). What is confusing about this result is the suggestion that such complexity would make a system more adaptable. Complexity has been found to make systems more difficult to maintain (e.g., Fuentes et al., 2014; Gibson et al., 1998; Lei Wu et al., 2005; Rinta-Kahila, 2018). This is primarily attributed to the many interdependent relations (Matook & Brown, 2017) and the cognitive challenge (K. Lee et al., 2022; Scandura, 1994) required to change a complex artifact.

However, a potential explanation emerges if one isolates to interactions strictly in the technical environment. It could be the case that, much like how complexity is needed to implement a deep structure representation of a business process, complexity is necessary to implement the technologies to make an adaptable system structure. For example, many organizations implement service-oriented architectures (Araujo et al., 2021; de Kinderen & Kaczmarek-Heß, 2017) for their information systems. This architecture is complex, consisting of many interdependent relations. However, as a tradeoff, the resulting systems are more maintainable and amenable to organizational changes (Mishra et al., 2021). In this case, the artifact has a complex architecture, but the complex architecture results in an adaptable system.

The findings also support the hypothesis that adaptation positively influences the representational fidelity of a system. This is an example of how a physical structure exerts influence on the deep structure of a system. If the system's physical structure cannot be changed, then the deep structure cannot be updated to match the current state of the real-world system and business processes. This result provides strong support for the representation theory axiom that a system’s deep structure is always limited by the characteristics of the software and hardware components by which it is constructed (Recker et al., 2019; Wand & Weber, 1995).

The significant relationship between state and representational fidelity provides strong empirical support for the state-tracking model of representation theory. Most representation theory research focuses on the representational model rather than the state-tracking model (Thomas & Dhillon, 2012; Wand & Weber, 2017). However, the state-tracking model is an important support structure of a representational model. It focuses on tracking and maintaining a faithful model of the world. If state is not accurately tracked, then meaningful and accurate representations cannot be constructed (Recker et al., 2019; Wand & Weber, 1995). This is one of the first studies to empirically test the theorized link between the state-tracking and representational models. This result suggests that state tracking abilities embedded in the physical structure of the IT artifact are a necessary antecedent to a faithful representation model (Thomas & Dhillon, 2012; Wand & Weber, 1990, 1995).

I also hypothesized that the capabilities of the system would be linked to the deep structure representation of a system (Burton-Jones & Grange, 2013; Wand & Weber, 1995). So, if a system were to have high representational fidelity, then the system should also have fewer system capability shortcomings. Instead, I found a nonsignificant result. One explanation is that the deep structure of the system is operating at a different level of abstraction than the capabilities of the system of the physical structure. The physical structure is used to implement the deep structure (Recker et al., 2019; Wand & Weber, 1995), but that is not the only purpose of the physical structure. Representational fidelity is primarily concerned with having an accurate conceptual model of the world embedded in the system. While there are many system capabilities that are needed to implement that model, not every capability of the system is necessary to construct that model. Nor is the system limited to just the capabilities needed to implement a deep structure representation.

For example, an organization may have an accurate business model embedded in its accounting system, indicating high representational fidelity in the deep structure of the system. Relevant business processes, general ledger, and transaction processes are well modeled in the deep structure and implemented using the physical structure hardware and software components. However, due to the physical structure's limitations, the system's performance is inadequate for the number of employees who now use the system as the organization has grown. Additionally, the development team would like to implement design best practices that are impossible with current hardware and software implementations.

In this scenario, the representational fidelity is high, and the system’s model of the world is accurate. However, the system still lacks the capabilities the organization would like. This disconnect between system layer abstractions and the multipurpose nature of the physical structure could explain the nonsignificant result. The representation theory literature has generally undertheorized physical structure concerns (Recker et al., 2021), arguing that a good information system design can be understood abstracted from a specific technical implementation (Wand & Weber, 1990). This result suggests that deeper scrutiny of the physical structure is necessary outside of its relationship with deep structure concerns.

Legacy Perception as a Construct

One of the reasons for testing Model 1 and Model 2 was to understand the nature of the legacy perception construct. Based on the results, I believe that legacy perception should be measured with a reflective first-order construct rather than a formative second-order construct. It was not my initial intent to compare the models, as the assumption was the indicators for Model 1 would be supported, and Model 2 would investigate the interactions of the indicators. However, the results of Model 1 did not support a formative approach to measuring legacy perception.

Two of the first-order reflective constructs, non-integration, and system age, had non-significant weights. Additionally non-representational fidelity had a significant weight, but in the negative direction. on the second-order construct. Complexity, non-adaptation, and system support non-availability were significant, but had relatively low weights. The only first-order construct that shows promise as an indicator of a second-order legacy perception construct is system capability shortcomings.

Of course, in the context of formative measurement, just looking at the weights is not enough, as dropping the lowest loading indicators would fundamentally alter the construct domain of legacy perception (J. F. Hair Jr. et al., 2021b; Jarvis et al., 2003). Arguably, the second-order construct is already fundamentally flawed due to the necessity of dropping non-transparent interaction and non-connectivity due to discriminant validity issues. Even if I had kept those indicators, I do not believe the results would have been notably different because they would have captured the same variance, considering the high HTMT scores. This, combined with low and negative weights, leads me to believe that a second-order formative measure of legacy perception is misspecified.

One potential argument against the first-order conceptualization is that the second-order formative version of legacy perception has higher explanatory power for replacement intention (R^2 = .40 versus R^2 = .27) and system investment (R^2 = .26 versus R^2 = .22). I believe this is an artifact of the second-order construct consisting of indicators previously found to be associated with replacement intention and system investment, such as system capability shortcomings and system support availability (Furneaux & Wade, 2017). In the first-order reflective model legacy perception does not include indicators from those constructs, and direct relationships are not modeled to replacement intentions or system investment from constructs other than legacy perception. Model 2 also isolates the specific effect of legacy perception on system investment and replacement intention since legacy perception is its own first-order construct with unique items.

While the Model 1 results in this study invalidated a second-order formative conception of legacy perception, the first-order reflective version of the measure shows some promise. Practically speaking, this version of the construct would be easier to include in other research models since it requires only three items to measure legacy perception rather than 34 items. The reflective version of legacy perception was also robust to reliability, discriminant validity, and convergent validity tests across both the scale development and model testing rounds of data collection.

Measurement of Technical Structures in Behavioral Research

In addition to hypothesized relationships, this research also focuses on developing measures for technical characteristics and structures in behavioral research. Results here mark a promising start but are overall mixed. The biggest issue I encountered with the technical measures was discriminant validity. I think, at some level, this is to be expected. When asking an individual about multiple related aspects of a single IT artifact, there will be some overlap. IT artifact characteristics do not exist in isolation (Matook & Brown, 2017).

Participants in the card sort were able to clearly separate the items, so I think participants recognize the characteristics conceptually as separate. However, this did not bear out in the data for round two, as the HTMT test found connectivity, integration, and representational fidelity to be indistinguishable. Similar issues existed for representational fidelity and transparent interaction, and transparent interaction and system support availability. Cross-loadings were not as high in round one, but they were still high enough to raise concerns about discriminant validity.

I believe the connectivity and integration discriminant validity issue could be a disconnect between how integration is used colloquially in software development (i.e., building connections between systems) versus how Matook and Brown (2017) conceptualize it as a characteristic of internal artifact structure. In the context of their model, connectivity would capture, conceptually, the concept of a system integration. This may suggest that the integration measure used in this study should be modified to remove loaded terms like integration while still capturing the essence of Matook and Brown's (2017) conceptualization. IT managers are also further removed from day-to-day operations of a system so developers may be better at distinguishing between the concepts. It could also be argued that a survey instrument is not capable of meaningfully measuring these characteristics. Instead, it may be necessary to utilize measures that collect data from the artifact itself, such as counts of module interconnections.

The issues between connectivity and representational fidelity are more confusing. At least conceptually, there is no clear overlap, and they are developed from two completely different theories other than a shared base of general systems theory (Burton-Jones & Grange, 2013;

Matook & Brown, 2017). It could be related to the need for connectivity to mediate and represent digital reality interactions in representations (Recker et al., 2021), but I find this explanation lacking and in need of additional study. Similarly, transparent interaction and system support availability are based on two separate theories (Burton-Jones & Grange, 2013; Furneaux & Wade, 2011). Although the culprit here could be the overlap in terms such as "obtain" in the item wordings for each, even if the thing being obtained is quite different, vendor support versus system content.

More problematic but less surprising is the discriminant validity issues between representational fidelity and transparent interaction. To my knowledge, this is the first empirical test of Burton-Jones and Grange's (2013) proposed transparent interaction measure as previous work has only included a measure of representational fidelity (Burleson, 2016; Burleson et al., 2021). I think measuring them together reveals a limitation in understanding representation theory artifact structures in the context of behavioral research.

A user determines if a system is representationally faithful by evaluating outputs from the surface structure (Burton-Jones & Grange, 2013). This means that the only way for a user to determine if a system's deep structure has representational fidelity in the first place is through a transparent interaction with the surface structure. While a strict reading of representation theory argues that representational fidelity of the deep structure can be evaluated independently of use (Wand & Weber, 1995), Burton-Jones and Grange's (2013) conceptualization, which the measures are based on, considers both representational fidelity and transparent interaction use characteristics.

A poor transparent interaction will always result in poor representational fidelity in this conceptualization. A good transparent interaction is always going to result in good representational fidelity. There is no way for the user to evaluate them separately because the surface structure is the lens by which they evaluate the deep structure. This is a case where these structures have a clear theoretical difference (Burton-Jones & Grange, 2013; Wand & Weber, 1995). They exist separately, but there is no feasible way to measure them separately in a behavioral study. I believe the closest one could get to a feasible measurement of both is to have someone evaluate a conceptual model for representational fidelity separately from the system implementation. Then, when measuring transparent interaction, one could measure the distance between the faithfulness of the model they observed versus what was actually able to be accessed via the surface structure. However, at that point, it is arguably a different measure than what was proposed (Burton-Jones & Grange, 2013).

On the positive side, the newly developed measures of adaptation, connectivity, integration, and state exhibited strong convergent validity and reliability across both rounds of data collection. The newly developed measures also exhibited acceptable but not great loadings in the CFA and passed on most model fit metrics. The measures were also successfully used in survey research and produced significant hypotheses in the context of the structural models. While there are certainly issues that need to be worked out on the margin concerning construct validity, the overall validity of measuring these very technical characteristics via a survey instrument is promising.

Theoretical Contributions

This work contributes to the IS literature in three primary ways. The first is to the legacy systems literature by reviewing the existing literature, conceptualizing a theory-grounded definition of legacy systems, conceptualizing legacy perception as a construction of an IT manager with measures for testing a perception formation, and testing factors that influence that legacy perception. The second is developing four new scales to measure technical characteristics in the context of behavioral research. The third contribution is empirically testing representation theory concepts in the context of behavioral research, including the link between physical structures and deep structures.

The review of the legacy systems literature integrates insights from the behavioral, technical, economic, and managerial IS research streams on legacy systems. These literatures, especially the technical literature, have mainly developed separately from each other, and as such, insights and integrations between them have not been identified. While further theoretical refinement is necessary for the literature review, it is a solid first step toward understanding the disjoint and disparate insights of the legacy systems literature in IS. My research models also integrate insights from the different literature streams in an attempt to synthesize and understand the findings of each branch of the literature.

As part of reviewing the literature in both IS and CS on how legacy systems are defined, I developed a new definition grounded in native IS theory. Despite describing a legacy system as a technical artifact, this definition introduces a uniquely behavioral perspective. I argue that while a legacy system is a tangible technical artifact, the label of legacy is constructed and applied to the system by an IT manager. I think this constructionist view aligns with the fact that not all users agree on what a legacy system is. However, I argue that this construction is based on real tangible aspects of the technology and social and environmental signals. Legacy as a label is constructed, but some factors influence whether that construction takes place. I believe this behavioral lens synthesizes the conflicting definitions in the literature, as it simultaneously explains why not every individual perceives a system the same way while also identifying what precisely about the artifact is likely to influence this construction. The social construction does not undermine the material nature of the artifact.

The results strongly support both the social and technical factors influence the formation of a legacy perception. The technical factor of system capability shortcomings had the strongest influence on legacy perception formation with a path coefficient of 0.44. System support availability was the social characteristics that had the largest influence on legacy perception with a path coefficient of -0.22. The findings also suggest that it is not system age that drives a legacy perception, system age may instead be a proxy for other things changing in the social and technical environments. The most important thing though is a legacy perception cannot be understood with only social or only technical factors. A legacy perception is socio-technical in nature.

As part of testing this, I developed two different conceptualizations for measuring legacy perception. The results provide strong support for a reflective measure of legacy perception that is influenced by different factors in the technical and social subsystems. The structural models provide insights into what factors influence a legacy perception of IT managers, providing further support for the proposed theorization of legacy as a formed perception of a system rather than a legacy system being an objective label. This scale has gone through both scale development and model testing and can be used by other legacy systems scholars to measure legacy perception in their own contexts.

An additional goal of this research was to address the need for more direct technology theorization. Tiwana (2019, p. 190) argues, "Similarly, units of analysis more unique to IS such as the system, IT artifact, an instantiation of a transaction, or a record remain uncommon in IS theorizing". This research attempts to respond to Tiwana by applying representation theory

(Recker et al., 2021; Wand & Weber, 1995), systems thinking (Goldkuhl, 2013b; Matook & Brown, 2017), and complexity theory (Arthur, 2009) to more deeply and directly theorize about an information system artifact. The legacy system itself is treated as a unit of analysis in this work. By taking this approach and centering the technology, I could theorize artifact structures and make claims about the artifact itself rather than treating it as a black box or a context.

One contribution towards this theoretical goal at a practical level is the operationalization of some of Matook and Brown's (2017) IT artifact characteristics, adaptation, connectivity, integration, and state for survey research. Previously, IS researchers could include the characteristics in overall theorizing but lacked research instruments to measure them. Similar instruments exist in the literature. However, most do not capture the purely technical aspects of Matook and Brown's (2017) conceptualization, often conflating portions of the social subsystem with the technical artifact characteristic or lacking rigorous scale development (Stachofsky, 2018). By developing new scales for the characteristics, more IS scholars can incorporate artifact characteristics into their theory and measurement more explicitly in any context, not just legacy systems.

Additionally, this work contributes strongly to the literature on representation theory. This literature is focused mainly on conceptual modeling research. Furthermore, most research has only considered the deep structure of a system (Recker et al., 2019), neglecting physical and surface concerns (Recker et al., 2021). While representational fidelity in the deep structure is the ultimate goal of representation theory, by neglecting physical and surface structure concerns the pathways to implementing that deep structure are not entirely clear.

In this research I tested the relationships between adaptation and state to representational fidelity. These are both characteristics of the physical structure of the system. The findings show that the physical structure and deep structure are interlinked, and also identifies two important drivers of representational fidelity which is the most important goal of representation theory (Recker et al., 2019; Wand & Weber, 1995). I also explored the antecedents of adaptation within the physical structure showing potential influences on the necessary technical characteristics to support representational fidelity.

Due to discriminant validity issues, I could not test surface structure concerns. Transparent interaction had not been empirically tested prior to this research, and these results show the challenges of including the measure in empirical behavioral research. This can be used as a first step towards developing an improved measure, if such measure is feasible in behavioral research. This research also provides strong support for the state-tracking model (Thomas & Dhillon, 2012; Wand & Weber, 2017) as an antecedent of the representation model. This confirms the theorized argument that accurate state tracking is necessary to implement and support an representation model in the deep structure of a system (Wand & Weber, 1995).

Limitations and Future Research

This work marks merely a first step in understanding legacy perception. The most obvious limitation is that this work focuses only on the legacy perception of an IT manager. Future research should incorporate more perspectives, such as system users and developers. Research that can collect data from multiple individuals for the same system would be especially valuable in studying potential conflicts of legacy perceptions between individuals and the ramifications for management decisions. The challenge is that not every potential group will be able to evaluate the same aspects of a system. For example, I chose IT managers for this context because I did not expect general users of a system to be able to answer questions about system architecture. Different user groups may have entirely different sets of factors they consider when forming a legacy perception. In particular, collecting data from developers who can better evaluate technical characteristics would be ideal for studying technical structures.

The overall variance explained of legacy perception in Model 2 was $R^2 = .23$. This suggests that many more factors in both the technical and social subsystems likely need to be explored as potential influences on legacy perception. This dissertation thoroughly explores the technical side of things given the theoretical underpinnings of the model, but it is admittedly sparse on measures of the social subsystem. I have selected what I think is most relevant in the social subsystem, but the theorization could be expanded. Additionally, the characteristics of the person evaluating a system and especially the characteristics of the organizational context may also be relevant.

A further limitation of this work is conceptualizing an information system as a technical artifact. I predicate this definition on the axioms of the complexity theory of technology (Arthur, 2009), systems thinking theory of IT artifacts (Goldkuhl, 2013b; Matook & Brown, 2017), and representation theory of information systems (Wand & Weber, 1995). This conflicts with other views arguing that the social system is an internal part of an information system, not an external one that interacts with the information system (e.g., A. S. Lee et al., 2015; Light, 2003). Future research may wish to explore legacy systems from a different lens to extract different insights.

Another limitation of this work is that I did not create scales for all the IT characteristics proposed by Matook and Brown (2017). Scales are still needed for self-adaptation and synchronicity. I did not develop them for this research as they did not seem theoretically relevant to my context. In a similar vein, exploring whether acceptable discriminant validity is achievable for all seven characteristics is another research opportunity, given that this dissertation found high cross-loadings with some of the characteristic constructs. It could be the case that measuring all seven at once may not be statistically feasible, and researchers may instead carefully choose the most relevant measures to their study.

I also think there are opportunities to measure other aspects of the IT artifact. The power of Matook and Brown’s (2017) conceptualization is that the characteristics are abstract. They can be applied to many contexts. However, I think this partially limits the explanatory power of developed models. If we take the legacy systems context as an example, perhaps developing a measure of technical degradation could be useful. While this measure would not be relevant to most IS research contexts, a measure of technical degradation could further explain why a system is perceived as legacy. This measure could capture information about failing hardware components in the legacy system. This would likely need to be collected through survey measurement, or potentially through logs that monitor hardware components.

For representation theory, this work builds upon the extant conceptual modeling literature and the limited empirical behavioral literature (Burleson, 2016; Burleson et al., 2021; Burton-Jones & Grange, 2013). As a result of building my arguments on that literature, I have mostly captured relationships related to the representation model, which makes up the vast majority of existing representation theory work (Wand & Weber, 2017). Future research should further investigate links to the state-tracking model and good-decomposition models of representation theory (Recker et al., 2019; Wand & Weber, 1990, 1995). While this dissertation begins exploratory measure and theorization of the state-tracking model, the good-decomposition model is not discussed at all. One challenge here is that, at least in the context of behavioral research, new measures will need to be developed to theorize and test these structures. Even the conceptual modeling literature is sparse on these topics with a few exceptions (Burton-Jones &

Meso, 2008, 2006; Thomas & Dhillon, 2012; Yang & Marquardt, 2009) due to the difficulty in operationalizing this aspect of representation theory (Wand & Weber, 2017).

Practical Implications

The primary value of this research to practitioners is identifying what factors influence the formation of legacy perception. The results suggest that IT managers that form a legacy perception of a system have higher intentions to replace that system. Knowing what factors are embedded in this socio-technical construction allows IT managers and developers to place specific focus on different aspects of the system and its context. In some cases, organizations could increase the system's longevity through strategic development choices. In other cases, organizations could recognize that it may be time to replace a system given the characteristics they have observed about the system. Legacy system replacement projects are costly endeavors with high chances of failure. If it is the case that a system is still useful and can be modified such that it is not perceived as legacy, that could avoid taking on a risky replacement too early.

The results of this study suggest that system age actually is not that important in determining whether a system is legacy or not. Instead, managers should focus on things such as system capability shortcomings as an internal signal and support availability for the system as an external environmental signal that a system may be legacy. When a system is not meeting capability requirements or vendors have dropped support, that could signal management that system replacement may be necessary. System age may instead be a proxy that gets blamed for these other issues that emerge over time. This is a surprising but promising result for IT managers because system age is not something they have control over, but the other factors that emerge they can partially address.

For development and system architecture, this research suggests that building highly adaptable systems is crucial for maintaining an accurate and up-to-date business model embedded in the system. System complexity and an integrated architecture are also not necessarily inhibitors of an adaptable structure and may be necessary to implement such a structure in the first place.

Each individual factor on its own is not necessarily an indication of a system being legacy. However, if IT managers observe both system capability shortcomings and a lack of system support availability it may suggest that it is time to replace a system. Alternatively, focus can be placed on controlling these factors. System support availability is largely an external factor driven by labor markets and vendor support. However, this research also suggests that system capability shortcomings are the most important aspect to address if the goal is for the system to not be perceived as legacy. If the underlying physical architecture of the system is highly adaptable IT managers can focus on directing development resources to develop the missing system capabilities.

Conclusion

This dissertation responds to the need for more IS research on post-implementation and end-of-life IS phenomena (C. Edwards, 1984; Furneaux & Wade, 2011; Rinta-Kahila, 2018). Specifically, this research investigates the social and technical factors that lead to a system being perceived as legacy, contributing to the behavioral research stream on legacy systems. I contribute to the legacy systems literature by reviewing the IS literature on legacy systems, developing a definition of legacy systems as a socio-technical construction grounded in native IS theory, and creating a scale for measuring legacy perception.

This dissertation provides support for measuring legacy perception as a reflective first-order construct. It also suggests that system age is not a key influencer of legacy perception, but system capability shortcomings and a lack of system support availability are important. This research also models interactions of legacy systems' physical structures, finding that integration and complexity positively influence the adaptability of legacy system artifacts. I also find that the adaptability of an artifact and state-tracking abilities positively influence representational fidelity. This study also finds that a legacy perception of a system positively influences both system investment behaviors and intentions to replace a legacy system.

In addition to identifying factors related to legacy perception, this research responds to the need for more explicit theorizing of technology in IS research (Tiwana, 2019). This research develops scales that IS scholars can use to incorporate technology theorizing into their models and provides an example of applying representation theory in a behavioral empirical context and testing core tenets of the theory. This work integrates insights of the technical, behavioral, and managerial literature on legacy systems. I hope this work can be a stepping stone for future research on perceptions of legacy systems and theorizing IT artifact characteristics and structures in behavioral IS research.

REFERENCES

2020 Mainframe Modernisation Business Barometer Report. (2020). Advanced. https://www.businesswire.com/news/home/20200528005186/en/74-Of-Organizations-Fail-to-Complete-Legacy-System-Modernization-Projects-New-Report-From-Advanced-Reveals

Abrahao, S. M., & Prado, A. F. do. (1999). Web-enabling legacy systems through software transformations. *Proceedings of International Workshop on Advance Issues of E-Commerce and Web-Based Information Systems. (Cat. No.PR00334)*, 87–90. https://doi.org/10.1109/WECWIS.1999.788193

Achee, B. L., & Carver, D. L. (1995). Identification and extraction of objects from legacy code. *1995 IEEE Aerospace Applications Conference. Proceedings*, *2*, 181–190 vol.2. https://doi.org/10.1109/AERO.1995.468926

Ackoff, R. L. (1971). Towards a System of Systems Concepts. *Management Science*, *17*(11), 661–671. https://doi.org/10.1287/mnsc.17.11.661

Ahmad, A., & Babar, M. A. (2014). A Framework for Architecture-Driven Migration of Legacy Systems to Cloud-Enabled Software. *Proceedings of the WICSA 2014 Companion Volume*. https://doi.org/10.1145/2578128.2578232

Ahmad, L. (2017). *Analysis of Modified Delphi Card Sorting*. Universität Paderborn.

Aiken, P., Yoon, Y., & Leong-Hong, B. (1999). Requirements-driven data engineering. *Information & Management*, *35*(3), 155–168. https://doi.org/10.1016/S0378-7206(98)00082-2

Akhtyamov, R., Vingerhoeds, R., & Golkar, A. (2018). Measures and approach for modernization of existing systems. *2018 IEEE International Systems Engineering Symposium (ISSE)*, 1–8. https://doi.org/10.1109/SysEng.2018.8544427

Alavi, M., Visentin, D. C., Thapa, D. K., Hunt, G. E., Watson, R., & Cleary, M. (2020). Chi-square for model fit in confirmatory factor analysis. *Journal of Advanced Nursing*, *76*(9), 2209–2211. https://doi.org/10.1111/jan.14399

Althani, B., & Khaddaj, S. (2017). Systematic Review of Legacy System Migration. *2017 16th International Symposium on Distributed Computing and Applications to Business, Engineering and Science (DCABES)*, 154–157. https://doi.org/10.1109/DCABES.2017.41

Alvarez, R. (2000). Examining an ERP Implementation through Myths: A Case Study of a Large Public Organization. *AMCIS 2000 Proceedings*, 8.

Alves, N. S. R., Ribeiro, L. F., Caires, V., Mendes, T. S., & Spínola, R. O. (2014). Towards an Ontology of Terms on Technical Debt. *2014 Sixth International Workshop on Managing Technical Debt*, 1–7. https://doi.org/10.1109/MTD.2014.9

Antoniol, G., Fiutem, R., Merlo, E., & Tonella, P. (1995). Application and user interface migration from BASIC to Visual C++. *Proceedings of International Conference on Software Maintenance*, 76–85. https://doi.org/10.1109/ICSM.1995.526529

Araujo, F., Sengupta, S., Jang, J., Doupé, A., Hamlen, K., & Kambhampati, S. (2021). *Software Deception Steering through Version Emulation*. Hawaii International Conference on System Sciences. https://doi.org/10.24251/HICSS.2021.243

Arthur, W. B. (2009). *The nature of technology: What it is and how it evolves*. Simon and Schuster.

Arthur, W. B., & Polak, W. (2006). The evolution of technology within a simple computer model. *Complexity*, *11*(5), 23–31. https://doi.org/10.1002/cplx.20130

Augusto, F., Ribeiro, J., & Gomes, R. (2009). Service-Oriented Architecture Adoption In A Portuguese Company: A case Study. *MCIS 2008 Proceedings*, 13.

Azadmanesh, M. H., & Peak, D. A. (1995). The Metamorphosis of the Mainframe Computer: The Superserver. *AMCIS 1995 Proceedings*, 7.

Balachandran, K. (n.d.). *kardSort* [Computer software]. https://kardsort.com/

Banker, R. D., Davis, G. B., & Slaughter, S. A. (1998). Software Development Practices, Software Complexity, and Software Maintenance Performance: A Field Study. *Management Science*, *44*(4), 433–450.

Banker, R. D., & Slaughter, S. A. (1997). A Field Study of Scale Economies in Software Maintenance. *Management Science*, *43*(12), 1709–1725. https://doi.org/10.1287/mnsc.43.12.1709

Barnes, D., Hinton, C. M., & Mieczkowska, S. (2001). Managing E-Operations for Competitive Advantage. *ICEB 2001 Proceedings*, 11.

Barry, E. J., Kemerer, C. F., & Slaughter, S. A. (2007). How software process automation affects software evolution: A longitudinal empirical analysis. *Journal of Software Maintenance and Evolution: Research and Practice*, *19*(1), 1–31. https://doi.org/10.1002/smr.342

Baskerville, R. L., Davison, R. M., Kaul, M., Malaurent, J., & Wong, L. H. (2022). Information systems as a nexus of information technology systems: A new view of information systems practice. *Journal of Information Technology*, *37*(4), 387–406. https://doi.org/10.1177/02683962221108757

Baskerville, R. L., Myers, M., & Yoo, Y. (2020). Digital First: The Ontological Reversal and New Challenges for IS Research. *MIS Quarterly*, *44*(2).

Bateman, B. L., & Wetherbe, J. C. (1978). Cost Analysis of Computer Maintenance Contracts. *MIS Quarterly*, *2*(4), 15. https://doi.org/10.2307/248901

Baxter, I. D., & Mehlich, M. (1997). Reverse engineering is reverse forward engineering. *Proceedings of the Fourth Working Conference on Reverse Engineering*, 104–113. https://doi.org/10.1109/WCRE.1997.624581

Becker, J.-M., Klein, K., & Wetzels, M. (2012). Hierarchical Latent Variable Models in PLS-SEM: Guidelines for Using Reflective-Formative Type Models. *Long Range Planning*, *45*(5–6), 359–394. https://doi.org/10.1016/j.lrp.2012.10.001

Beer, S. (1972). *Brain of the firm: A Development in Management Cybernetics* (2nd ed.). Herder and Herder.

Bennett, K. (1995). Legacy systems: Coping with success. *IEEE Software*, *12*(1), 19–23. https://doi.org/10.1109/52.363157

Bevan, J. (2000). Contributions of Object Oriented Software Design Towards Limiting the Problems Caused by a Lack of Software Engineering. *AMCIS 2000 Proceedings*, 6.

Bhat, J., & Goel, S. (2011). Mining Transaction Data for Process Instance Monitoring in Legacy Systems. *AMCIS 2011 Proceedings*, 11.

Bhatt, G., Emdad, A., Roberts, N., & Grover, V. (2010). Building and leveraging information in dynamic environments: The role of IT infrastructure flexibility as enabler of organizational responsiveness and competitive advantage. *Information & Management*, *47*(7–8), 341–349. https://doi.org/10.1016/j.im.2010.08.001

Bhattacherjee, A., & Harris, M. (2009). INDIVIDUAL ADAPTATION OF INFORMATION TECHNOLOGY. *Journal of Computer Information Systems*.

Bing Wu, D. Lawless, J. Bisbal, J. Grimson, V. Wade, D. O'Sullivan, & R. Richardson. (1997). Legacy systems migration-a method and its tool-kit framework. *Proceedings of Joint 4th International Computer Science Conference and 4th Asia Pacific Software Engineering Conference*, 312–320. https://doi.org/10.1109/APSEC.1997.640188

Bisbal, J., Lawless, D., Bing Wu, & Grimson, J. (1999). Legacy information systems: Issues and directions. *IEEE Software*, *16*(5), 103–111. https://doi.org/10.1109/52.795108

Blumenthal, S. C. (1969). *Management information systems; a framework for planning and development*. Prentice-Hall; WorldCat.org.

Bob-Jones, B., Newman, M., & Lyytinen, K. (2008). Picking Up the Pieces After a "Successful" Implementation: Networks, Coalitions and ERP Systems. *AMCIS 2008 Proceedings*, 13.

Bodeau, D. J. (1994). System-of-systems security engineering. *Tenth Annual Computer Security Applications Conference*, 228–235. https://doi.org/10.1109/CSAC.1994.367304

Bostrom, R. P., & Heinen, J. S. (1977a). MIS Problems and Failures: A Socio-Technical Perspective. Part I: The Causes. *MIS Quarterly*, *1*(3), 17. https://doi.org/10.2307/248710

Bostrom, R. P., & Heinen, J. S. (1977b). MIS Problems and Failures: A Socio-Technical Perspective, Part II: The Application of Socio-Technical Theory. *MIS Quarterly*, *1*(4), 11. https://doi.org/10.2307/249019

Boulding, K. E. (1956). General Systems Theory-The Skeleton of Science. *Management Science*, *2*(3), 13.

Brancheau, J. C., Janz, B. D., & Wetherbe, J. C. (1996). Key Issues in Information Systems Management: 1994-95 SIM Delphi Results. *MIS Quarterly*, *20*(2), 225. https://doi.org/10.2307/249479

Brandt, T., Feuerriegel, S., & Neumann, D. (2018). Modeling interferences in information systems design for cyberphysical systems: Insights from a smart grid application. *European Journal of Information Systems*, *27*(2), 207–220. https://doi.org/10.1057/s41303-016-0030-1

Brodie, M., & Stonebraker, M. (1995). *Migrating Legacy Systems: Gateways, Interfaces, & The Incremental Approach*. Morgan Kaufmann.

Brook, M., Gruhn, V., & Schoepe, L. (2000). Realizing an Integrated Electronic Commerce Portal System. *AMCIS 2000 Proceedings*, 8.

Brooke, C. (2000). A Framework for Evaluating Organizational Choice and Process Redesign Issues. *Journal of Information Technology*, *15*(1), 17–28. https://doi.org/10.1177/026839620001500103

Brooke, C. (2002). Information systems at CallCentre. *International Journal of Information Management*, *22*(5), 389–401. https://doi.org/10.1016/S0268-4012(02)00029-4

Brooke, C., & Ramage, M. (2001). Organisational scenarios and legacy systems. *International Journal of Information Management*, *21*(5), 365–384. https://doi.org/10.1016/S0268-4012(01)00023-8

Buchmann, R. A., & Karagiannis, D. (2016). Enriching Linked Data with Semantics from Domain-Specific Diagrammatic Models. *Business & Information Systems Engineering*, *58*(5), 341–353. https://doi.org/10.1007/s12599-016-0445-1

Bunge, M. (1966). Technology as Applied Science. *Technology and Culture*, *7*(3), 20.

Bunge, M. (1977). *Ontology I: The Furniture of the World*. Reidel.

Bunge, M. (1979). *Ontology II: A World of Systems*. Reidel.

Burleson, J. (2016). *Organization-Stakeholder Interaction Through Social Media: A Tri-Level Investigation, Categorization, and Research Agenda*.

Burleson, J., Grover, V., Thatcher, J. B., & Sun, H. (2021). A Representation Theory Perspective on the Repurposing of Personal Technologies for Work-Related Tasks. *Journal of the Association for Information Systems*, 34.

Burton-Jones, A., & Grange, C. (2013). From Use to Effective Use: A Representation Theory Perspective. *Information Systems Research*, *24*(3), 632–658. https://doi.org/10.1287/isre.1120.0444

Burton-Jones, A., & Meso, P. (2008). The Effects of Decomposition Quality and Multiple Forms of Information on Novices' Understanding of a Domain from a Conceptual Model. *Journal of the Association for Information Systems*, *9*(12), 748–802. https://doi.org/10.17705/1jais.00179

Burton-Jones, A., & Meso, P. N. (2006). Conceptualizing Systems for Understanding: An Empirical Test of Decomposition Principles in Object-Oriented Analysis. *Information Systems Research*, *17*(1), 38–60. https://doi.org/10.1287/isre.1050.0079

Cajas, V., Urbieta, M., Rossi, G., & Domínguez, F. (2020). Challenges of Migrating Legacies Web to Mobile: A Systematic Literature Review. *IEEE Latin America Transactions*, *18*(05), 861–873. https://doi.org/10.1109/TLA.2020.9082914

Carlson, A. C., Brook, W. R., & Haynes, C. L. F. (1996). Experiences with distributed objects. *AT&T Technical Journal*, *75*(2), 58–67. https://doi.org/10.15325/ATTTJ.1996.6771129

Carvalho, L., Garcia, A., Assunção, W. K. G., Bonifácio, R., Tizzei, L. P., & Colanzi, T. E. (2019). Extraction of Configurable and Reusable Microservices from Legacy Systems: An Exploratory Study. *Proceedings of the 23rd International Systems and Software Product Line Conference - Volume A*, 26–31. https://doi.org/10.1145/3336294.3336319

Cenfetelli & Bassellier. (2009). Interpretation of Formative Measurement in Information Systems Research. *MIS Quarterly*, *33*(4), 689. https://doi.org/10.2307/20650323

Chen, S.-H. (2010). E-Procurement System Adoption under Supply Chain Migration: A Case Study of Taiwanese Notebook Company. *PACIS 2010 Proceedings*, 13.

Chin, W. (1998). Commentary: Issues and Opinion on Structural Equation Modeling. *MIS Quarterly*, *22*(1).

Chin, W. W., Thatcher, J. B., Wright, R. T., & Steel, D. (2013). Controlling for common method variance in PLS analysis: The measured latent marker variable approach. In *New perspectives in partial least squares and related methods* (pp. 231–239). Springer.

Chirathamjaree, C. (2004). A Mediator to Integrate Databases and Legacy Systems: The Mediated Data Integration (MeDInt) Mediator. *ICEB 2004 Proceedings*, 7.

Chirathamjaree, C. (2005). An Approach to Integrate Heterogeneous Data Sources. *ICEB 2005 Proceedings*, 7.

Chirathamjaree, C. (2006). An Approach to Resolve Data Model Heterogeneities in Multiple Data Sources. *TENCON 2006 - 2006 IEEE Region 10 Conference*, 1–4. https://doi.org/10.1109/TENCON.2006.343819

Chou, T.-H., & Seng, J.-L. (2009). An intelligent multi-agent e-services method—An international telecommunication example. *Information & Management*, *46*(6), 342–350. https://doi.org/10.1016/j.im.2009.05.006

Chyung, S. Y. (Yonnie), Barkin, J. R., & Shamsy, J. A. (2018). Evidence-Based Survey Design: The Use of Negatively Worded Items in Surveys. *Performance Improvement*, *57*(3), 16–25. https://doi.org/10.1002/pfi.21749

Cimitile, A., Lucia, A. D., Lucca, G. A. D., & Fasolino, A. R. (1997). Identifying objects in legacy systems. *Proceedings Fifth International Workshop on Program Comprehension. IWPC'97*, 138–147. https://doi.org/10.1109/WPC.1997.601281

Clark, L. A., & Watson, D. (2019). Constructing validity: New developments in creating objective measuring instruments. *Psychological Assessment*, *31*(12), 1412–1427. https://doi.org/10.1037/pas0000626

Clark, & Watson. (1995). Constructing Validity: Basic Issues in Objective Scale Development. *Psychological Assessment*, *7*(3), 309–319.

Clegg, C., Coleman, P., Hornby, P., Maclaren, R., Robson, J., Carey, N., & Symon, G. (1996). Tools to incorporate some psychological and organizational issues during the development of computer-based systems. *Ergonomics*, *39*(3), 482–511. https://doi.org/10.1080/00140139608964476

Coakes, E., & Elliman, T. (1999). The Role of Stakeholders in Managing Change. *Communications of the Association for Information Systems*, *2*. https://doi.org/10.17705/1CAIS.00204

Cosentino, V., Cabot, J., Albert, P., Bauquel, P., & Perronnet, J. (2013). Extracting business rules from COBOL: A model-based framework. *2013 20th Working Conference on Reverse Engineering (WCRE)*, 409–416. https://doi.org/10.1109/WCRE.2013.6671316

Cronbach, L. J. (1951). Coefficient alpha and the internal structure of tests. *Psychometrika*, *16*(3), 297–334. https://doi.org/10.1007/bf02310555

Crowley, D. N., Dabrowski, M., & Breslin, J. G. (2013). Decision Support using Linked, Social, and Sensor Data. *AMCIS 2013 Proceedings*, 11.

Cuttler, C. (2014). *Student Guide to SPSS* (2nd ed.).

Datar, S., & Schach, S. R. (1996). Reuse of legacy software in object-oriented re-engineering: A case study. *Transactions of the South African Institute of Electrical Engineers*, *87*(3), 101–107.

de Kinderen, S., & Kaczmarek-Heß, M. (2017). Enterprise Modeling Support for SOA Migration. *13th International Conference on Wirtschaftsinformatik*, 4.

De Marco, M., & Sorrentino, M. (2007). Sowing the Seeds of is Cultivation in Public Service Organisations. *Journal of Information Technology*, *22*(2), 184–194. https://doi.org/10.1057/palgrave.jit.2000070

Dearden, J., & McFarlan, F. W. (1966). *Management information systems: Text and cases*. RD Irwin.

Dekleva, S. M. (1992). The Influence of the Information Systems Development Approach on Maintenance. *MIS Quarterly*, *16*(3), 355. https://doi.org/10.2307/249533

Dennis, A. R., Samuel, B. M., & McNamara, K. (2014). Design for Maintenance: How Kms Document Linking Decisions Affect Maintenance Effort and Use. *Journal of Information Technology*, *29*(4), 312–326. https://doi.org/10.1057/jit.2014.5

Edberg, D., Ivanova, P., & Kuechler, W. (2012). Methodology Mashups: An Exploration of Processes Used to Maintain Software. *Journal of Management Information Systems*, *28*(4), 271–304. https://doi.org/10.2753/MIS0742-1222280410

Edwards, C. (1984). Information Systems Maintenance: An Integrated Perspective. *MIS Quarterly*, *8*(4), 237. https://doi.org/10.2307/249094

Edwards, H. M., Mallalieu, G. M., & Thompson, J. B. (1999). Some insights into the maintenance of legacy systems within small manufacturing and distribution organisations in the UK. *Proceedings. Twenty-Third Annual International Computer Software and Applications Conference (Cat. No.99CB37032)*, 13–20. https://doi.org/10.1109/CMPSAC.1999.812508

Edwards, J. R. (2011). The Fallacy of Formative Measurement. *Organizational Research Methods*, *14*(2), 370–388. https://doi.org/10.1177/1094428110378369

EMpanel Online. (n.d.). Retrieved March 6, 2024, from https://empanelonline.com/

Endmann, A., Fischer, H., & Krökel, M. (Eds.). (2015). Casolysis 2.0: Flexible Auswertung von Card Sorting Experimenten. In *Mensch und Computer 2015 – Usability Professionals* (pp. 445–456). De Gruyter. https://doi.org/10.1515/9783110443882-056

Etzkorn, L. H., & Davis, C. G. (1997). Automatically identifying reusable OO legacy code. *Computer*, *30*(10), 66–71. https://doi.org/10.1109/2.625311

Etzkorn, L. H., Davis, C. G., Bowen, L. L., Etzkorn, D. B., Lewis, L. W., Vinz, B. L., & Wolf, J. C. (1996). A knowledge-based approach to object-oriented legacy code reuse. *Proceedings of ICECCS '96: 2nd IEEE International Conference on Engineering of Complex Computer Systems (Held Jointly with 6th CSESAW and 4th IEEE RTAW)*, 493–496. https://doi.org/10.1109/ICECCS.1996.558517

Evermann, J. (2012). Applying Cognitive Principles of Similarity to Data Integration – The Case of SIAM. *AMCIS 2012 Proceedings*, 11.

Fahmideh, M., Daneshgar, F., Rabhi, F., & Beydoun, G. (2019). A generic cloud migration process model. *European Journal of Information Systems*, *28*(3), 233–255. https://doi.org/10.1080/0960085X.2018.1524417

Fleiss, J. L. (1971). Measuring nominal scale agreement among many raters. *Psychological Bulletin*, *76*(5), 378–382. https://doi.org/10.1037/h0031619

Fornell, C., & Larcker, D. (1981). Evaluating Structural Equation Models with Unobservable Variables and Measurement Error. *Journal of Marketing Research*, *18*(1), 39–50.

Frick, N., Brünker, F., Ross, B., & Stieglitz, S. (2020). Design Requirements for AI-based Services Enriching Legacy Information Systems in Enterprises: A Managerial Perspective. *ACIS 2020 Proceedings*, 13.

Frohlich, M. T., & Dixon, J. R. (1999). Information Systems Adaptation and the Successful Implementation of Advanced Manufacturing Technologies. *Decision Sciences*, *30*(4), 921–957. https://doi.org/10.1111/j.1540-5915.1999.tb00914.x

Froscher, J. N., Kang, M., McDermott, J., Costich, O., & Landwehr, C. E. (1994). A practical approach to high assurance multilevel secure computing service. *Tenth Annual Computer Security Applications Conference*, 2–11. https://doi.org/10.1109/CSAC.1994.367324

Fuentes, V., Matias, J., Mendiola, A., Huarte, M., Unzilla, J., & Jacob, E. (2014). Integrating Complex Legacy Systems under OpenFlow Control: The DOCSIS Use Case. *2014 Third European Workshop on Software Defined Networks*, 37–42. https://doi.org/10.1109/EWSDN.2014.35

Furneaux, B., & Wade, M. (2010). The end of the information system life: A model of is discontinuance. *The DATA BASE for Advances in Information Systems*, *41*(2), 25.

Furneaux, B., & Wade, M. (2017). Impediments to Information Systems Replacement: A Calculus of Discontinuance. *Journal of Management Information Systems*, *34*(3), 902–932. https://doi.org/10.1080/07421222.2017.1373013

Furneaux & Wade. (2011). An Exploration of Organizational Level Information Systems Discontinuance Intentions. *MIS Quarterly*, *35*(3), 573. https://doi.org/10.2307/23042797

Gamer, M., Lemon, J., & Singh, I. F. P. (2019). *irr: Various Coefficients of Interrater Reliability and Agreement* (0.84.1) [Computer software]. https://cran.r-project.org/web/packages/irr/index.html

García-García, J. A., Maldonado, C. A., Meidan, A., Morillo-Baro, E., & Escalona, M. J. (2021). gPROFIT: A Tool to Assist the Automatic Extraction of Business Knowledge From Legacy Information Systems. *IEEE Access*, *9*, 94934–94952. https://doi.org/10.1109/ACCESS.2021.3093356

Gholami, M. F., Daneshgar, F., Beydoun, G., & Rabhi, F. (2017). Challenges in migrating legacy software systems to the cloud—An empirical study. *Information Systems*, *67*, 100–113. http://dx.doi.org/10.1016/j.is.2017.03.008

Giaglis, G. (1999). On the Integrated Design and Evaluation of Business Processes and Information Systems. *Communications of the Association for Information Systems*, *2*. https://doi.org/10.17705/1CAIS.00205

Gibson, N., Holland, C., & Light, B. (1998). Identifying Misalignment Between Strategic Vision and Legacy Information Systems in Organizations. *AMCIS 1998 Proceedings*, 4.

Goeschka, K. M., & Schranz, M. W. (2001). Client and legacy integration in object oriented Web engineering. *IEEE MultiMedia*, *8*(1), 32–41. https://doi.org/10.1109/93.923951

Goldkuhl, G. (2013a). From ensemble view to ensemble artefact—An inquiry on conceptualisations of the IT artefact. *Systems, Signs & Action*, *7*(1), 24.

Goldkuhl, G. (2013b). The IT artefact: An ensemble of the social and the technical? - A rejoinder. *Systems, Signs, & Actions*, *7*(1), 90–99.

Goodhue, Lewis, & Thompson. (2012). Does PLS Have Advantages for Small Sample Size or Non-Normal Data? *MIS Quarterly*, *36*(3), 981. https://doi.org/10.2307/41703490

Grainger, N., McKay, J., & Marshall, P. (2009). Learning from a Strategic Failure. *ACIS 2009 Proceedings*, 12.

Grossman, T. A., Mehrotra, V., & Özlük, Ö. (2007). Lessons from Mission-Critical Spreadsheets. *Communications of the Association for Information Systems*, *20*. https://doi.org/10.17705/1CAIS.02060

Hair Jr., J. F., Hult, G. T. M., Ringle, C. M., & Sarstedt, M. (2022). *A primer on partial least squares structural equation modeling (PLS-SEM)* (3rd ed.). Sage publications.

Hair Jr., J. F., Hult, G. T. M., Ringle, C. M., Sarstedt, M., Danks, N. P., & Ray, S. (2021a). Evaluation of Formative Measurement Models. In J. F. Hair Jr., G. T. M. Hult, C. M. Ringle, M. Sarstedt, N. P. Danks, & S. Ray (Eds.), *Partial Least Squares Structural Equation Modeling (PLS-SEM) Using R: A Workbook* (pp. 91–113). Springer International Publishing. https://doi.org/10.1007/978-3-030-80519-7_5

Hair Jr., J. F., Hult, G. T. M., Ringle, C. M., Sarstedt, M., Danks, N. P., & Ray, S. (2021b). *Partial Least Squares Structural Equation Modeling (PLS-SEM) Using R: A Workbook*. Springer International Publishing. https://doi.org/10.1007/978-3-030-80519-7

Hair Jr, J. F., Ringle, C. M., & Sarstedt, M. (2011). PLS-SEM: Indeed a Silver Bullet. *Journal of Marketing Theory and Practice*, *19*(2), 139–152. https://doi.org/10.2753/MTP1069-6679190202

Hair Jr, J., Hult, G. T., Ringle, C., & Sarstedt, M. (2017). *A primer on partial least squares structural equation modeling (PLS-SEM)* (2nd ed., Vol. 38). SAGE. http://www.tandfonline.com/doi/abs/10.1080/1743727X.2015.1005806

Hanseth, O., & Lyytinen, K. (2010). Design Theory for Dynamic Complexity in Information Infrastructures: The Case of Building Internet. *Journal of Information Technology*, *25*(1), 1–19. https://doi.org/10.1057/jit.2009.19

Hauser, M. M., & Gã, S. A. (2017). Designing Pervasive Information Systems: A Fashion Retail Case Study. *ICIS 2017 Proceedings*, 17.

Henseler, J., Ringle, C. M., & Sarstedt, M. (2015). A new criterion for assessing discriminant validity in variance-based structural equation modeling. *Journal of the Academy of Marketing Science*, *43*(1), 115–135. https://doi.org/10.1007/s11747-014-0403-8

Henseler, J., Ringle, C. M., & Sinkovics, R. R. (2009). The use of partial least squares path modeling in international marketing. In R. R. Sinkovics & P. N. Ghauri (Eds.), *New Challenges to International Marketing* (Vol. 20, pp. 277–319). Emerald Group Publishing Limited. https://doi.org/10.1108/S1474-7979(2009)0000020014

Hermes, S., Clemons, E., Schreieck, M., Pfab, S., Mitre, M., Böhm, M., Wiesche, M., & Krcmar, H. (2020). Breeding Grounds of Digital Platforms: Exploring the Sources of American Platform Domination, China's Platform Self-sufficiency, and Europe's Platform Gap. *ECIS 2020 Proceedings*, 20.

Holland, C., & Light, B. (1999). Generic Information Systems Design Strategies. *AMCIS 1999 Proceedings*, 4.

Holland, C. P., Light, B., Beck, P., Berdugo, Y., Millar, R., Press, N., & Setlavad, M. (2000). An International Analysis of the Maturity of Enterprise Resource Planning (ERP) Systems Use. *AMCIS 2000 Proceedings*, 8.

Holland, C. P., Light, B., & Kawalek, P. (1999). Focus Issue on Legacy Information Systems and Business Process Change: Introduction. *Communications of the Association for Information Systems*, *2*. https://doi.org/10.17705/1CAIS.00209

Holvitie, J., Licorish, S. A., Martini, A., & Leppänen, V. (2016). Co-Existence of the 'Technical Debt' and 'Software Legacy' Concepts. *1st International Workshop on Technical Debt Analytics*, 4.

Howard, M., Powell, P., & Vidgen, R. (2004). Interorganizational Collaboration and Value Creation in the Automotive Industry. *ECIS 2004 Proceedings*, 15.

IBM Corp. (2020). *IBM SPSS Statistics for Windows* (Version 28) [Computer software]. IBM Corp.

Ifinedo, P. (2014). An Examination of Information Technology Assets and Resources as Antecedent Factors to ERP System Success. *MCIS 2014 Proceedings*, 18.

Iivari, J. (2020). Editorial: A Critical Look at Theories in Design Science Research. *Journal of the Association for Information Systems*, *21*(3), 502–519. https://doi.org/10.17705/1jais.00610

Ingalsbe, J. A., Shoemaker, D., Mead, N. R., & Drommi, A. (2008). Threat Modeling the Enterprise. *AMCIS 2008 Proceedings*, 9.

Jain, A. K., & Dubes, R. C. (1988). *Algorithms for clustering data*. Prentice Hall.

Jain, S., Bhushan, M., Yadav, A., & Tiwari, D. (2016). Incremental Architectural Re-engineering for the Cloud: An Overview. *2016 Second International Conference on Computational Intelligence & Communication Technology (CICT)*, 655–659. https://doi.org/10.1109/CICT.2016.135

Jarvis, C. B., MacKenzie, S. B., & Podsakoff, P. M. (2003). A Critical Review of Construct Indicators and Measurement Model Misspecification in Marketing and Consumer Research. *Journal of Consumer Research*, *30*(2), 199–218. https://doi.org/10.1086/376806

Jermaine, C. (1999). Computing program modularizations using the k-cut method. *Sixth Working Conference on Reverse Engineering (Cat. No.PR00303)*, 224–234. https://doi.org/10.1109/WCRE.1999.806963

Jiao, J. (Roger), Zhang, L. (Linda), Pokharel, S., & He, Z. (2007). Identifying generic routings for product families based on text mining and tree matching. *Decision Support Systems*, *43*(3), 866–883. https://doi.org/10.1016/j.dss.2007.01.001

Kalnins, A., & Praitis Hill, K. (2023). The VIF Score. What is it Good For? Absolutely Nothing. *Organizational Research Methods*, 10944281231216381. https://doi.org/10.1177/10944281231216381

Kappelman, L., L. Johnson, V., Maurer, C., Guerra, K., McLean, E., Torres, R., Snyder, M., & Kim, K. (2020). The 2019 SIM IT Issues and Trends Study. *MIS Quarterly Executive*, *19*(1), 69–104. https://doi.org/10.17705/2msqe.00026

Kappelman, L., McLean, E., Johnson, V., & Torres, R. (2016). The 2015 SIM IT Issues and Trends Study. *MIS Quarterly Executive*, *15*(1).

Kappelman, L., McLean, E. R., Johnson, V. L., Torres, R., Maurer, C., Kim, K., Guerra, K., & Snyder, M. (2021). The 2020 SIM IT Issues and Trends Study. *MIS Quarterly Executive*, *20*(1).

Kappelman, L., Torres, R., McLean, E., Maurer, C., Johnson, V., & Kim, K. (2019). The 2018 SIM IT Issues and Trends Study. *MIS Quarterly Executive*, *18*(1). https://doi.org/10.17705/2msqe.00008

Kappelman, L., Torres, R., McLean, E., Maurer, C., Johnson, V., Snyder, M., Guerra, K., & Srivastava, S. (2022). *2022 Comprehensive Report: Results and Observations from the SIM IT Trends Study*. Society for Information Management.

Kavakli, V., & Loucopoulos, P. (1999). Modelling of Organisational Change Using the EKD Framework. *Communications of the Association for Information Systems*, *2*. https://doi.org/10.17705/1CAIS.00206

Kelly, S., Gibson, N., Holland, C. P., & Light, B. (1999). A Business Perspective of Legacy Information Systems. *Communications of the Association for Information Systems*, *2*. https://doi.org/10.17705/1CAIS.00207

Kemerer, C. F., & Slaughter, S. A. (1997). Determinants of software maintenance profiles: An empirical investigation. *Journal of Software Maintenance: Research and Practice*, *9*(4), 235–251. https://doi.org/10.1002/(SICI)1096-908X(199707/08)9:4<235::AID-SMR153>3.0.CO;2-3

Kock, N. (2015). Common Method Bias in PLS-SEM: A Full Collinearity Assessment Approach. *International Journal of E-Collaboration*, *11*(4), 1–10. https://doi.org/10.4018/ijec.2015100101

Kock, N., & Hadaya, P. (2018). Minimum sample size estimation in PLS-SEM: The inverse square root and gamma-exponential methods. *Information Systems Journal*, *28*(1), 227–261. https://doi.org/10.1111/isj.12131

Konkel, F. (2016). *Report: Almost Half of IT Migrations Doomed to Fail*. Nextgov.Com. https://www.nextgov.com/cio-briefing/2016/03/report-almost-half-it-migrations-doomed-fail/126574/

Krishnan, M. S., Mukhopadhyay, T., & Kriebel, C. H. (2004). A Decision Model for Software Maintenance. *Information Systems Research*, *15*(4), 396–412. https://doi.org/10.1287/isre.1040.0037

Langefors, B. (1968). *Theoretical analysis of information systems*. Studentlitteratur.

Lee, A. S., Thomas, M., & Baskerville, R. L. (2015). Going back to basics in design science: From the information technology artifact to the information systems artifact: From the IT artifact to the IS artifact. *Information Systems Journal*, *25*(1), 5–21. https://doi.org/10.1111/isj.12054

Lee, K., Han, K., Animesh, A., & Pinsonneault, A. (2022). Does IT Matter to Acquisitions? The Impacts of IT Distance on Post-acquisition Performance. *MIS Quarterly*, *46*(4).

Lei Wu, H. Sahraoui, & P. Valtchev. (2005). Coping with legacy system migration complexity. *10th IEEE International Conference on Engineering of Complex Computer Systems (ICECCS'05)*, 600–609. https://doi.org/10.1109/ICECCS.2005.29

Lenski, R. E., Ofria, C., Pennock, R. T., & Adami, C. (2003). The evolutionary origin of complex features. *Nature*, *423*(6936), 139–144. https://doi.org/10.1038/nature01568

Li, X., & Lau, S. K. (2005). A Multi-Agent Approach Towards Collaborative Supply Chain Management. *ICEB 2005 Proceedings*, 8.

Light, B. (2003). An Alternative Theory of Legacy Information Systems. *ECIS 2003 Proceedings*, 13.

Light, B., Holland, C., & Gibson, N. (1998). The Influence of Legacy Information Systems on Business Process Change Strategies. *AMCIS 1998 Proceedings*, 4.

Limaj, E., Bernroider, E., & Ivanova, M. (2020). Facing Legacy Information System Modernization in Scaling Agility in the Banking Industry: Preliminary Insights on Strategies and Non-technical Barriers. *ICIS 2020 Proceedings*, 10.

Lloyd, A. D., Dewar, R., & Pooley, R. (1999). Legacy Information Systems and Business Process Change: A Patterns Perspective. *Communications of the Association for Information Systems*, *2*. https://doi.org/10.17705/1CAIS.00224

Lu, X.-H., Huang, L.-H., & Heng, M. S. H. (2006). Critical success factors of inter-organizational information systems—A case study of Cisco and Xiao Tong in China. *Information & Management*, *43*(3), 395–408. https://doi.org/10.1016/j.im.2005.06.007

Luftman, J. (2005). Key Issues for IT Executives 2004. *MIS Quarterly Executive*, *4*(2).

Luftman, J., Kempaiah, R., & Nash, E. (2006). Key Issues for IT Executives 2005. *MIS Quarterly Executive*, *5*(2).

MacKenzie, Podsakoff, & Podsakoff. (2011). Construct Measurement and Validation Procedures in MIS and Behavioral Research: Integrating New and Existing Techniques. *MIS Quarterly*, *35*(2), 293. https://doi.org/10.2307/23044045

Madlberger, M. (2012). Reaching a Higher Level of Information Systems Integration: The Impact of Information Technology Substitution Strategies on Process Efficiency. *ECIS 2012 Proceedings*, 13.

Mahalanobis, P. C. (1936). On the generalised distance in statistics. *Proceedings of the National Institute of Sciences of India*, *2*(1).

http://library.isical.ac.in:8080/xmlui/bitstream/handle/10263/6765/Vol02_1936_1_Art05-pcm.pdf

Mahapatra, R., & Lai, V. (1998). Intranet-based Training Facilities ERP System Implementation: A Case Study. *AMCIS 1998 Proceedings*, 4.

Mallampalli, V., & Karahanna, E. (2017). Why Don't Systems Die? An Escalation of Commitment Perspective. *AMCIS 2017 Proceedings*, 5.

Mallampalli, V., Safadi, H., Faraj, S., & Univeristy, M. (2018). The Role of Anchoring in Actualizing IT Affordances in EMR Implementations. *Proceedings of the 51st Hawaii International Conference on System Sciences*, 11.

Masiero, P. C., & Braga, R. T. V. (1999). Legacy systems reengineering using software patterns. *Proceedings. SCCC'99 XIX International Conference of the Chilean Computer Science Society*, 160–169. https://doi.org/10.1109/SCCC.1999.810173

Matook, S., & Brown, S. A. (2017). Characteristics of IT artifacts: A systems thinking-based framework for delineating and theorizing IT artifacts. *Information Systems Journal*, *27*(3), 309–346. https://doi.org/10.1111/isj.12108

Mattord, H. J., & Bandyopadhyay, T. (2008). The Impact of Operating System Obsolescence on the Life Cycle of Distributed Teams. *AMCIS 2008 Proceedings*, 8.

Mehrizi, M. H., Rodon, J., & Zafarnejad, M. (2012). Managing Obsolete Knowledge: Towards a Clarified and Contextualized Conception of Unlearning. *Academy of Management Proceedings*, *2012*(1), 15473. https://doi.org/10.5465/AMBPP.2012.87

Mehrizi, M. H., Rodon Modol, J., & Zafar Nezhad, M. (2019). Intensifying to Cease: Unpacking the Process of Information Systems Discontinuance. *MIS Quarterly*, *43*(1), 141–165. https://doi.org/10.25300/MISQ/2019/13717

Mehrizi, M. H., van den Hooff, B., & Yang, C. (2022). Breaking or keeping the habits: Exploring the role of legacy habits in the process of discontinuing organisational information systems. *Information Systems Journal*, *32*(1), 192–221. https://doi.org/10.1111/isj.12341

Mesgari, M. (2018). Affordance-based Use Case Mining. *AMCIS 2018 Proceedings*, 9.

Miller, J. G. (1965). Living systems: Basic concepts. *Behavioral Science*, *10*(3), 193–237.

Mishra, A. K., Nagpal, R., Seth, K., & Sehgal, R. (2021). A Critical Review on Service Oriented Architecture and its Maintainability. *2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, 1–8. https://doi.org/10.1109/ICRITO51393.2021.9596285

Moorhead, G. (1993). New computers extend the life of deployed testers. *AUTOTESTCON 93*, 667–670. https://doi.org/10.1109/AUTEST.1993.396291

Moreton, R. (1990). A Process Model for Software Maintenance. *Journal of Information Technology*, *5*.

Moseley, B., & Marks, P. (2006). Out of the tar pit. *Software Practice Advancement (SPA), 2006*.

M'pherson, P. K. (1974). A perspective on systems science and systems philosophy. *Futures*, *6*(3), 219–239.

Mumford, E. (1965). Clerks and Computers: Study of the Introduction of Technical Change. *Journal of Management Studies*, *2*(2), 138–152. https://doi.org/10.1111/j.1467-6486.1965.tb00133.x

Ng, M. M. T., & Tan, M. T. K. (2004). Symbolic Processes in ERP versus "Legacy" System Support. *ECIS 2004 Proceedings*, 13.

Nielsen, F. (2016). Hierarchical Clustering. In F. Nielsen, *Introduction to HPC with MPI for Data Science* (pp. 195–211). Springer International Publishing. https://doi.org/10.1007/978-3-319-21903-5_8

Ning, J. Q., Engberts, A., & Kozaczynski, W. (1993). Recovering reusable components from legacy systems by program segmentation. *[1993] Proceedings Working Conference on Reverse Engineering*, 64–72. https://doi.org/10.1109/WCRE.1993.287778

Oca, C. M. de, & Carver, D. L. (1998). Identification of data cohesive subsystems using data mining techniques. *Proceedings. International Conference on Software Maintenance (Cat. No. 98CB36272)*, 16–23. https://doi.org/10.1109/ICSM.1998.738485

O'Callaghan, A. (1998). ADAPTOR: a pattern language for the re-engineering of systems to object technology. *IEE Colloquium on Understanding Patterns and Their Application to Systems Engineering (Digest No. 1998/308)*, 1/1-1/6. https://doi.org/10.1049/ic:19980541

O'Callaghan, A. J. (1999). Migrating Large-Scale Legacy Systems to Component-Based and Object Technology: The Evolution of a Pattern Language. *Communications of the Association for Information Systems*, *2*. https://doi.org/10.17705/1CAIS.00203

Ofoegbu, A., Griffiths, M., & Heinze, A. (2011). Themes and Challenges For Service Management Solutions In Small and Medium Enterprises (SMEs). *UK Academy for Information Systems Conference Proceedings 2011*, *UK Academy for Information Systems Conference*, 19.

O'Leary, D. (2000). Game Playing Behavior in Requirements Analysis, Evaluation, and System Choice for Enterprise Resource Planning Systems. *ICIS 2000 Proceedings*, 12.

Orlikowski, W. J. (2010). The sociomateriality of organisational life: Considering technology in management research. *Cambridge Journal of Economics*, *34*(1), 125–141. https://doi.org/10.1093/cje/bep058

Orlikowski, W. J., & Iacono, C. S. (2001). Research Commentary: Desperately Seeking the "IT" in IT Research—A Call to Theorizing the IT Artifact. *Information Systems Research*, *12*(2), 121–134. https://doi.org/10.1287/isre.12.2.121.9700

Orwig, R., & Dean, D. L. (2007). A Method for Building a Referent Business Activity Model for Evaluating Information Systems: Results from a Case Study. *Communications of the Association for Information Systems*, *20*. https://doi.org/10.17705/1CAIS.02053

Pan, S., Newell, S., Huang, J., & Cheung, A. (2001). Knowledge Integration as a Key Problem in an ERP Implementation. *ICIS 2001 Proceedings*, 9.

Pang, M.-S. (2017). Politics and Information Technology Investments in the U.S. Federal Government in 2003–2016. *Information Systems Research*, *28*(1), 33–45. https://doi.org/10.1287/isre.2016.0665

Pang, M.-S., & Tanriverdi, H. (2022). Strategic roles of IT modernization and cloud migration in reducing cybersecurity risks of organizations: The case of U.S. federal government. *The Journal of Strategic Information Systems*, *31*(1), 101707. https://doi.org/10.1016/j.jsis.2022.101707

Persson, P., Zhang, Y., Asatiani, A., Lindman, J., & Rudmark, D. (2023). Technical Debt in the Municipality Sector—The Missing Link with Citizens and Silofication. *14th Scandanavian Conference on Information Systems*. Scandinavian Conference on Information Systems, Porvoo, Finland.

Petersen, M. B., Skov, M., Serritzlew, S., & Ramsøy, T. (2013). Motivated reasoning and political parties: Evidence for increased processing in the face of party cues. *Political Behavior*, *35*(4), 831–854.

Peterson, R. (1994). A Meta-analysis of Cronbach's Coefficient Alpha. *Journal of Consumer Research*, *21*.

Petry, B. L. (1996). Getting the most out of legacy code: The uses of HyperCode within a typical IS organization. *Proceedings of the IEEE 1996 National Aerospace and Electronics Conference NAECON 1996*, *2*, 852–857 vol.2. https://doi.org/10.1109/NAECON.1996.517751

Petter, Straub, & Rai. (2007). Specifying Formative Constructs in Information Systems Research. *MIS Quarterly*, *31*(4), 623. https://doi.org/10.2307/25148814

Pilemalm, S., Andersson, D., & Mojir, K. Y. (2013). Migration of Legacy Systems in New Public Management – the Swedish Rescue Services Incident Reporting System. *ACIS 2013 Proceedings*, 12.

Podsakoff, P. M., MacKenzie, S. B., Lee, J.-Y., & Podsakoff, N. P. (2003). Common method biases in behavioral research: A critical review of the literature and recommended remedies. *Journal of Applied Psychology*, *88*(5), 879–903. https://doi.org/10.1037/0021-9010.88.5.879

Podsakoff, P. M., Podsakoff, N. P., Williams, L. J., Huang, C., & Yang, J. (2024). Common Method Bias: It's Bad, It's Complex, It's Widespread, and It's Not Easy to Fix. *Annual Review of Organizational Psychology and Organizational Behavior*, *11*(1), 17–61. https://doi.org/10.1146/annurev-orgpsych-110721-040030

Polites & Karahanna. (2012). Shackled to the Status Quo: The Inhibiting Effects of Incumbent System Habit, Switching Costs, and Inertia on New System Acceptance. *MIS Quarterly*, *36*(1), 21. https://doi.org/10.2307/41410404

Qualtrics International Inc. (2017). *Qualtrics XM* (February 2024) [Computer software]. Qualtrics International Inc. https://www.qualtrics.com

Quilici, A. (1995). Reverse Engineering of Legacy Systems: A Path Toward Success. *1995 17th International Conference on Software Engineering*, 333–333. https://doi.org/10.1145/225014.225046

R Core Team. (2023). *R Statistical Software* (4.3.1) [Computer software]. https://www.r-project.org/

Rajlich & Adnapally. (1996). VIFOR 2: A tool for browsing and documentation. *1996 Proceedings of International Conference on Software Maintenance*, 296–300. https://doi.org/10.1109/ICSM.1996.565019

Ramasubbu, N., & Kemerer, C. F. (2021). Controlling Technical Debt Remediation in Outsourced Enterprise Systems Maintenance: An Empirical Analysis. *Journal of Management Information Systems*, *38*(1), 4–28. https://doi.org/10.1080/07421222.2021.1870377

Ramchand, K., Chhetri, M. B., & Kowalczyk, R. (2018). *Towards A Comprehensive Cloud Decision Framework with Financial Viability Assessment*. 15.

Randall, D., Hughes, J., O'Brien, J., Rodden, T., Rouncefield, M., Sommerville, I., & Tolmie, P. (1999). Focus Issue on Legacy Information Systems and Business Process Change: Banking on the Old technology: Understanding the Organisational Context of "Legacy"

Issues. *Communications of the Association for Information Systems*, *2*. https://doi.org/10.17705/1CAIS.00208

Recker, J., Indulska, M., Green, P., Burton-Jones, A., & Weber, R. (2019). Information Systems as Representations: A Review of the Theory and Evidence. *Journal of the Association for Information Systems*, 735–786. https://doi.org/10.17705/1jais.00550

Recker, J., Lukyanenko, R., Jabbari, M., Samuel, B. M., & Castellanos, A. (2021). From Representation to Mediation: A New Agenda for Conceptual Modeling Research in a Digital World. *MIS Quarterly*, *45*(1), 32.

Ringle, C. M., Wende, S., & Becker, J.-M. (2024). *SmartPLS 4* (4.1.0.0) [Computer software]. https://www.smartpls.com/

Rinta-Kahila, T. (2018). Caught in between: How an Organization Became a Prisoner of Its Legacy System after IS Change. *AMCIS 2018 Proceedings*, 17.

Rinta-Kahila, T., Penttinen, E., & Lyytinen, K. (2023). Getting Trapped in Technical Debt: Sociotechnical Analysis of a Legacy System's Replacement. *MIS Quarterly*, *47*(1).

Rolland, K. H., Mathiassen, L., & Rai, A. (2018). Managing Digital Platforms in User Organizations: The Interactions Between Digital Options and Digital Debt. *Information Systems Research*, *29*(2), 419–443. https://doi.org/10.1287/isre.2018.0788

Roux, M. (2018). A Comparative Study of Divisive and Agglomerative Hierarchical Clustering Algorithms. *Journal of Classification*, *35*(2), 345–366. https://doi.org/10.1007/s00357-018-9259-9

S.3897 - 117th Congress: Legacy IT Reduction Act of 2022 (2022-03-22). (2022). https://www.congress.gov/bill/117th-congress/senate-bill/3897

Sarkar, S., Rama, G. M., & Kak, A. C. (2007). API-Based and Information-Theoretic Metrics for Measuring the Quality of Software Modularization. *IEEE Transactions on Software Engineering*, *33*(1), 14–32. https://doi.org/10.1109/TSE.2007.256942

Sarker, S., Chatterjee, S., Xiao, X., Copenhagen Business School, Elbanna, A., & Royal Holloway University of London. (2019). The Sociotechnical Axis of Cohesion for the IS Discipline: Its Historical Legacy and its Continued Relevance. *MIS Quarterly*, *43*(3), 695–719. https://doi.org/10.25300/MISQ/2019/13747

Sarstedt, M., Hair, J. F., Cheah, J.-H., Becker, J.-M., & Ringle, C. M. (2019). How to Specify, Estimate, and Validate Higher-Order Constructs in PLS-SEM. *Australasian Marketing Journal*, *27*(3), 197–211. https://doi.org/10.1016/j.ausmj.2019.05.003

Sasso, F., & Forcolin, M. (2009). A Cooperation Infrastructure For Communication Between Public Bodies; The Friuli Venezia Giulia Case. *MCIS 2009 Proceedings*, 9.

Scandura, J. M. (1994). Converting legacy code into Ada: A cognitive approach. *Computer*, *27*(4), 55–61. https://doi.org/10.1109/2.275004

Schnappinger, M., & Streit, J. (2021). Efficient Platform Migration of a Mainframe Legacy System Using Custom Transpilation. *2021 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, 545–554. https://doi.org/10.1109/ICSME52107.2021.00055

Schneberger, S. L., & McLean, E. R. (2003). The complexity cross: Implications for practice. *Communications of the ACM*, *46*(9), 216–225. https://doi.org/10.1145/903893.903940

Sellink, A., Sneed, H., & Verhoef, C. (1999). Restructuring of COBOL/CICS legacy systems. *Proceedings of the Third European Conference on Software Maintenance and Reengineering (Cat. No. PR00090)*, 72–82. https://doi.org/10.1109/CSMR.1999.756684

Senior Scholars' List of Premier Journals. (2023). Association for Information Systems. https://aisnet.org/page/SeniorScholarListofPremierJournals

Shankaranarayan, G., Balasubramanian, P. R., & Kang, C. (2000). Conceptualizing Architectures for E-Business Systems. *AMCIS 2000 Proceedings*, 5.

Sharif, A. M., Irani, Z., & Love, P. E. D. (2005). Integrating ERP using EAI: a model for post hoc evaluation. *European Journal of Information Systems*, *14*(2), 162–174. https://doi.org/10.1057/palgrave.ejis.3000533

Shim, S., Chae, M., & Lee, B. (2009). Empirical analysis of risk-taking behavior in IT platform migration decisions. *Computers in Human Behavior*, *25*(6), 1290–1305. https://doi.org/10.1016/j.chb.2009.07.004

Shumaker, J., Ward, K., & Blaskovich, J. (2011). Information System Knowledge Transfer Needs Assessment Instrument. *MWAIS 2011 Proceedings*, 7.

Smith, D. R. (1982). Obsolescence Evaluation of Process Control Computer Systems. *IEEE Transactions on Power Apparatus and Systems*, *PAS-101*(1), 191–194. https://doi.org/10.1109/TPAS.1982.317337

Sneath, P. H. A., & Sokal, R. R. (1973). *Numerical taxonomy: The principles and practice of numerical classification*. W. H. Freeman.

Sneed, H. M. (1995). Planning the reengineering of legacy systems. *IEEE Software*, *12*(1), 24–34. https://doi.org/10.1109/52.363168

Soliman, W., & Rinta-Kahila, T. (2019). Toward a refined conceptualization of IS discontinuance: Reflection on the past and a way forward. *Information & Management*, 103167. https://doi.org/10.1016/j.im.2019.05.002

Stachofsky, J. (2018). Delineating Legacy System IT Artifacts – Applying a Systems Thinking-based Framework. *2018 Proceedings of the Conference on Information Systems Applied Research*, *11*, 8.

Stachofsky, J., Gebremedhin, A., & Crossler, R. E. (2022). Cast to Vote: A Socio-technical Network Analysis of an Election Smartphone Application. *Digital Government: Research and Practice*, *3*(1), 1–17. https://doi.org/10.1145/3501031

Star, S. L., & Ruhleder, K. (1996). Steps Toward an Ecology of Infrastructure: Design and Access for Large Information Spaces. *Information Systems Research*, *7*(1), 25.

Stylianou, V., & Savva, A. (2023). Trends in KM-software and Other Non-KM-specific Software for Business Needs. *Proceedings of the 2023 Pre-ICIS SIGDSA Symposium*. Special Interest Group on Decision Support and Analytics, Hydrabad, India.

Sulong, M. S., Koronios, A., Gao, J., Abdul-Aziz, A., & Wongsim, M. (2011). Driving The Business Service-Oriented Architecture Enabled Initiative. *ICEB 2011 Proceedings*, 9.

Swanson, E. B., & Dans, E. (2000). System Life Expectancy and the Maintenance Effort: Exploring Their Equilibration. *MIS Quarterly*, *24*(2), 277. https://doi.org/10.2307/3250939

Tabachnick, B. G., & Fidell, L. S. (2018). *Using Multivariate Statistics* (Vol. 7). Pearson.

Tallon, P. P., Ramirez, R. V., & Short, J. E. (2013). The Information Artifact in IT Governance: Toward a Theory of Information Governance. *Journal of Management Information Systems*, *30*(3), 141–178. https://doi.org/10.2753/MIS0742-1222300306

Taylor, M., Moynihan, E., & Wood-Harper, T. (1997). Knowledge for Software Maintenance. *Journal of Information Technology*, *12*.

The Aging IT Workforce and Legacy Application Modernization. (2024). Deloitte. https://www2.deloitte.com/us/en/pages/technology/articles/aging-workforce-legacy-application-modernization.html

Thomas, M., & Dhillon, G. (2012). Interpreting Deep Structures of Information Systems Security. *The Computer Journal*, *55*(10), 1148–1156. https://doi.org/10.1093/comjnl/bxr118

Thummadi, B. V., Khapre, V. D., & Ocker, R. (2017). Unpacking Agile Enterprise Architecture Innovation work practices: A Qualitative Case Study of a Railroad Company. *AMCIS 2017 Proceedings*, 10.

Tiwana, A. (2018). Platform Synergy: Architectural Origins and Competitive Consequences. *Information Systems Research*, *29*(4), 829–848. https://doi.org/10.1287/isre.2017.0739

Tiwana, A. (2019). The curse of "reference" disciplines. *Journal of Information Technology*, *34*(2), 190–192. https://doi.org/10.1177/0268396219834140

Tona, O., Carlsson, S. A., & Eom, S. B. (2012). An Empirical Test of DeLone and McLean's Information System Success Model in a Public Organization. *AMCIS 2012 Proceedings*, 10.

Tsai, J. C.-A., Jiang, J. J., Klein, G., & Hung, S.-Y. (2022). Legacy information system replacement: Pursuing quality design of operational information systems. *Information & Management*, *59*(2), 103592. https://doi.org/10.1016/j.im.2022.103592

Tsai, J. C.-A., Jiang, J. J., Klein, G., & Hung, S.-Y. (2023). Task Conflict Resolution in Designing Legacy Replacement Systems. *Journal of Management Information Systems*, *40*(3), 1009–1034. https://doi.org/10.1080/07421222.2023.2229120

Vadapalli, A., & Nazareth, D. (1998). Software Reuse Management: Development of a Model in the Context of the Capability Maturity Model. *AMCIS 1998 Proceedings*, 4.

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

Venkatesh, V., & Brown, S. A. (2001). A Longitudinal Investigation of Personal Computers in Homes: Adoption Determinants and Emerging Challenges. *MIS Quarterly*, *25*(1), 71. https://doi.org/10.2307/3250959

Vergara, N. M., Linero, J. M. T., & Moreno, A. V. (2007). Model-driven component adaptation in the context of Web Engineering. *European Journal of Information Systems*, *16*(4), 448–459. https://doi.org/10.1057/palgrave.ejis.3000691

Vestues, K., & Knut, R. (2019). Making Digital Infrastructures More Generative Through Platformization and Platform- driven Software Development: An Explorative Case Study. *10th Scandinavian Conference on Information Systems*, 16.

Vestues, K., & Rolland, K. (2021). *Platformizing the Organization through Decoupling and Recoupling: A longitudinal case study of a government agency*. *33*, 33.

Vidgen, R., Shaw, S., & Grant, D. B. (2017). Management challenges in creating value from business analytics. *European Journal of Operational Research*, *261*(2), 626–639. https://doi.org/10.1016/j.ejor.2017.02.023

Von Bertalanffy, L. (1956). General system theory. *General Systems*, *1*(1), 11–17.

Von Bertalanffy, L. (1972). The History and Status of General Systems Theory. *Academy of Management Journal*, *15*(4), 21.

Wand, Y., & Weber, R. (1988). An Ontological Analysis of Some Fundamental Information Systems Concepts. *ICIS 1988 Proceedings*. International Conference on Information Systems, Minneapolis, MN.

Wand, Y., & Weber, R. (1990). Toward a Theory of the Deep Structure of Information Systems. *International Conference on Information Systems*, 11.

Wand, Y., & Weber, R. (1995). On the deep structure of information systems. *Information Systems Journal*, *5*(3), 203–223. https://doi.org/10.1111/j.1365-2575.1995.tb00108.x

Wand, Y., & Weber, R. (2017). Thirty Years Later: Some Reflections on Ontological Analysis in Conceptual Modeling. *Journal of Database Management*, *28*(1), 1–17. https://doi.org/10.4018/JDM.2017010101

Wang, E. T. G., Chiu, C.-H., & Chen, K.-X. (2013). Effect of IT Skills on IT Capabilities and IT-Business Alignment. *PACIS 2013 Proceedings*. Pacific Asia Conference on Information Systems, Jeju Island, Korea.

Warrell, D., & Stevens, K. J. (2003). Towards an Understanding of Risk Management in Legacy Information Systems. *ACIS 2003 Proceedings*, 11.

Weber, R. (2020). Taking the ontological and materialist turns: Agential realism, representation theory, and accounting information systems. *International Journal of Accounting Information Systems*, *39*, 100485. https://doi.org/10.1016/j.accinf.2020.100485

Weber, R. (2023). Sociomateriality, agential realism, and the metaphysics of accounting information systems: A response to Vosselman and De Loo. *International Journal of Accounting Information Systems*, *51*, 100639. https://doi.org/10.1016/j.accinf.2023.100639

Weber, Y., & Pliskin, N. (1996). The effects of information systems integration and organizational culture on a firm's effectiveness. *Information & Management*, *30*(2), 81–90. https://doi.org/10.1016/0378-7206(95)00046-1

Weide, B. W., Heym, W. D., & Hollingsworth, J. E. (1995). Reverse Engineering of Legacy Code Exposed. *1995 17th International Conference on Software Engineering*, 327–327. https://doi.org/10.1145/225014.225045

West, R. M. (2022). Best practice in statistics: The use of log transformation. *Annals of Clinical Biochemistry: International Journal of Laboratory Medicine*, *59*(3), 162–165. https://doi.org/10.1177/00045632211050531

Wheaton, B., Muthen, B., Alwin, D. F., & Summers, G. F. (1977). Assessing Reliability and Stability in Panel Models. *Sociological Methodology*, *8*, 84. https://doi.org/10.2307/270754

Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L., François, R., Grolemund, G., Hayes, A., Henry, L., Hester, J., Kuhn, M., Pedersen, T., Miller, E., Bache, S., Müller, K., Ooms, J., Robinson, D., Seidel, D., Spinu, V., … Yutani, H. (2019). Welcome to the Tidyverse. *Journal of Open Source Software*, *4*(43), 1686. https://doi.org/10.21105/joss.01686

Wiggerts, T. A. (1997). Using clustering algorithms in legacy systems remodularization. *Proceedings of the Fourth Working Conference on Reverse Engineering*, 33–43. https://doi.org/10.1109/WCRE.1997.624574

Wiggerts, T., Bosma, H., & Fielt, E. (1997). Scenarios for the identification of objects in legacy systems. *Proceedings of the Fourth Working Conference on Reverse Engineering*, 24–32. https://doi.org/10.1109/WCRE.1997.624573

Wolfart, D., Assunção, W. K. G., da Silva, I. F., Domingos, D. C. P., Schmeing, E., Villaca, G. L. D., & Paza, D. do N. (2021). Modernizing Legacy Systems with Microservices: A Roadmap. *Evaluation and Assessment in Software Engineering*, 149–159. https://doi.org/10.1145/3463274.3463334

Yang, A., & Marquardt, W. (2009). An ontological conceptualization of multiscale models. *Computers & Chemical Engineering*, *33*(4), 822–837. https://doi.org/10.1016/j.compchemeng.2008.11.015

Yuming Zhou & Baowen Xu. (1999). Extracting objects of Ada programs using module features. *Proceedings IEEE International Conference on Software Maintenance - 1999 (ICSM'99). "Software Maintenance for Business Change" (Cat. No.99CB36360)*, 23–29. https://doi.org/10.1109/ICSM.1999.792495

Zach, O. (2011). Exploring ERP System Outcomes in SMEs: A Multiple Case Study. *ECIS 2011 Proceedings*, 13.

Zalhan, P.-G., Silaghi, G. C., & Buchmann, R. A. (2019). Marrying Big Data with Smart Data in Sensor Stream Processing. *28th International Conference on Information Systems Development*, 12.

Zhang, Z., Yang, H., & Chu, W. C. (2006). Extracting Reusable Object-Oriented Legacy Code Segments with Combined Formal Concept Analysis and Slicing Techniques for Service Integration. *2006 Sixth International Conference on Quality Software (QSIC'06)*, 385–392. https://doi.org/10.1109/QSIC.2006.29

APPENDIX

APPENDIX A: SCALE ITEMS

* = Item dropped in Round 1 (scale development) data analysis
^ = Scale dropped in Round 2 (model testing) data analysis

Integration

Item	Developed for this Study based on Matook and Brown (2017) Definition
INT1	The system components are tightly combined.
INT2	Internal modules of the system work together.
INT3	System components are integrated.
INT4	System components are dependent on each other.

Table A1: Integration Items

Connectivity ^

Item	Developed for this Study based on Matook and Brown (2017) Definition
CON1	The system can interface with other systems.
CON2	The system is connected to other systems in the organization.
CON3	External systems can connect to this system easily.
CON4*	The system communicates over a network.

Table A2: Connectivity Items

Complexity

Item	Furneaux and Wade (2011) Technical Integration [Adapted as Complexity, items unchanged]
C1	The technical characteristics of this system make it complex.
C2	The system depends on sophisticated integration of technology components.
C3	There is considerable technical complexity underlying the system.
Item	**Furneaux and Wade (2017) System Complexity [Not used in Study]**
SC1	There is considerable complexity surrounding this system.
SC2	Implementing and operating this system is a complex undertaking.
SC3	In general, this system would be regarded as highly complex.

Table A3: Complexity Items

State

Item	Developed for this Study based on Matook and Brown (2017) Definition
S1	The system keeps records of events.
S2	The system stores data about previous interactions.
S3	The system stores data about previous system states.
S4	The system saves information between uses.

Table A4: State Items

Adaptation

Item	Developed for this Study based on Matook and Brown (2017) Definition
A1	The system can be changed.
A2	It is easy to change the system.
A3	The system is easily modified.
A4	Changing the system for new functionality is possible.

Table A5: Adaptation Items

Representational Fidelity

Item	Burton-Jones and Grange (2013)	Adaptation to Employees
RF1	When using the system, I find the content it provides me is sufficiently complete.	When employees use the system, they find the content it provides them is sufficiently complete.
RF2	When using the system, I find the content it provides me is sufficiently clear.	When employees use the system, they find the content it provides them is sufficiently clear.
RF3	When using the system, I find the content it provides me is sufficiently correct.	When employees use the system, they find the content it provides them is sufficiently correct.
RF4	When using the system, I find the content it provides me is sufficiently meaningful.	When employees use the system, they find the content it provides them is sufficiently meaningful.

Table A6: Representational Fidelity Items

Transparent Interaction ^

Item	Burton-Jones and Grange (2013)	Adaptation to Employees
TI1	I have seamless access to the content that I need	When employees use the system, they have seamless access to the content they need.
TI2	I have difficulty obtaining the content I need because of the system's interface	When employees use the system, they have no difficulty obtaining the content they need because of the system's interface.
TI3	I have difficulty obtaining the content I need because of physical characteristics of the device.	When employees use the system, they have no difficulty obtaining the content they need because of physical characteristics of the device.

Table A7: Transparent Interaction Items

System Support Availability

Item	Furneaux and Wade (2011)
SSA1*	We do not encounter difficulties in obtaining needed system support services.
SSA2	We can easily obtain the support resources necessary to continue operating this system.
SSA3	Support for this system is readily available.

Table A8: System Support Availability Items

System Capability Shortcomings

Item	Furneaux and Wade (2011)	Adaptation to Remove Double Barreled Questions
SCS1	There are notable limitations in the ability of this system to meet our needs.	There are notable limitations in the ability of this system to meet our needs.
SCS2	We would like to have many capabilities that are not supported by this system.	We would like to have many capabilities that are not supported by this system.
SCS3	The performance and functionality of this system is highly inadequate.	The performance of this system is highly inadequate.
SCS4		The functionality of this system is highly inadequate

Table A9: System Capability Shortcomings Items

System Age

Item	Furneaux and Wade 2017
SA1	Approximately how many years has the system in use in the organization?

Table A10: System Age Items

System Investment

Item	Furneaux and Wade (2011) [Items unchanged]	Adaptation to Remove Double Barreled Questions
SI1	Significant organizational resources have been invested in this system.	Significant organizational resources have been invested in this system.
SI2	We have committed considerable time and money to the implementation and operation of this system.	We have committed considerable time to the operation of this system.
SI3	The financial investments that have been made in this system are substantial.	We have committed considerable money to the operation of this system.
SI4		The financial investments that have been made in this system are substantial.

Table A11: System Investment Items

Replacement Intention

Item	Furneaux and Wade (2011)	Furneaux and Wade (2017) [Used in Study]
RI1	We plan to replace the system with a competing system.	We plan to replace this system with another system.
RI2	Our intention is to replace this system with an entirely different system.	Our intention is to replace this system with an entirely different system.
RI3	We will be implementing a replacement to this system.	We will be seeking to implement a replacement to this system.

Table A12: Replacement Intention Items

Legacy Perception (Reflective)

Item	Developed for this study
LP1	This system is considered a legacy system in the organization.
LP2	Other employees would consider this a legacy system.
LP3	I think the system is a legacy system.

Table A13: Legacy Perception Items

APPENDIX B: SCALE DEVELOPMENT CARD SORT

Following the guidelines of Mackenzie et al. (2011), I conducted scale development activities for adaptation, connectivity, integration, state, and the reflective version of legacy perception. These scales were chosen because items had to be created for this research as there are no existing measures in the literature. Conceptualization and item generation were based on the IT artifact characteristics established by Matook and Brown (Matook & Brown, 2017). There were no additional items in earlier rounds of scale development, only those present in Appendix A. The measurement model proposed for each construct is modeled as reflective. To assess the content validity of the developed items, I conducted three rounds of closed card sorting with three Ph.D. students, five IS faculty members, and three industry professionals using kardSort (Balachandran, n.d.) to collect responses. The email sent to participants can be found in Appendix D Exhibit 1, and individual items can be found in Appendix A.

Card Sorting Analysis

In the first round of card sorting with Ph.D. students, one participant placed the item "The system is connected to other systems in the organization." under *Integration* rather than *Connectivity*. This may be partly due to system integration projects often involving connecting information systems. However, in the context of IT artifact characteristics, integration describes an artifact's internal structures (Matook & Brown, 2017). Since this was isolated to one participant and one item, I did not change the item wording for future card sorts. Participants indicated that the items and constructs were clear overall.

In the second round of card sorting with IS faculty, one participant placed "The system stores data about previous system states." under *Legacy Perception* instead of *State*. That same participant also placed "The system stores data about previous interactions." in the *Fits Multiple*

Categories category. Similar to before, this was isolated to a single participant, and there was no apparent theoretical reason to change the construct items. However, it suggests that an artifact's historical state tracking may be partially intertwined with its legacy status. Overall, participants indicated that constructs and items were clear.

In the final round of card sorting with industry professionals, one participant put “System components are dependent on each other.” in the *Fits Multiple Categories* option. The same participant also placed “The system is connected to other systems in the organization.” under *Integration* rather than *Connectivity*. This was the same assignment one participant gave the item in the first round of card sorting. In the follow-up comments, one of the participants also noted, “While tightly integrated components don’t inherently carry a legacy perception, this could vary across organizations.” This comment does not affect the constructs and items themselves, as integration and legacy perception are separate constructs, but it does suggest that the hypothesized link between integration and legacy perception may be contextual.

Since each round of card sorting had the same items and categories, I combined the three rounds into a single dataset of 11 participants for further analysis. A cluster analysis was conducted using Casolysis 2.0 (Endmann et al., 2015) using a hierarchical single-link agglomerative cluster approach (Nielsen, 2016; Roux, 2018; Sneath & Sokal, 1973, p. 214). This approach initially treats every item as its own cluster via a blank graph representing a proximity matrix. Then, nearby clusters are merged into a single cluster, meaning that individual items consistently placed in the same category of the card sort begin to form their own cluster with each iteration through the proximity matrix. If enough iterations occur, all items will eventually be grouped in the same cluster (L. Ahmad, 2017; A. K. Jain & Dubes, 1988, p. 61). However, for this analysis, iterations stop once the distinct categories emerge.

The hierarchical clustering algorithm analysis reveals a clear point of separation for the five constructs at a linkage distance of 0.18, meaning that the specific items are closely related and precise to their given categories. These five categories hold until a linkage distance of 0.82 is reached, meaning they are not sensitive to slight variations in the data even as cluster thresholds become more lenient.

Fleiss' Kappa (Fleiss, 1971) was calculated in R 4.3.1 (R Core Team, 2023) using the Tidyverse (Wickham et al., 2019) and Interrater Reliability libraries (Gamer et al., 2019). R code for analysis can be found in Appendix D Exhibit 3. Fleiss' Kappa measures the level of agreement between more than two raters. In the context of this study, there were 19 items, 11 raters, and five categories, making it a good fit for this research. The Fleiss' Kappa value was 0.943 indicating near perfect agreement among the raters for item categorization with a z-score of 61.6 and p-value of $< .001$. This Kappa score along with the hierarchical cluster analysis provides strong empirical support for the proposed constructs and their respective items.

As a final test, I calculated the hit ratio for all the items presented in Table B1. The hit ratio measures the accuracy of item placement for the proposed constructs (Moore & Benbasat, 1991). Since there were 11 raters and 19 items, 209 total item placements took place. The raters placed 204 items correctly, resulting in an overall hit ratio of 97.61%. The diagonal in Table B1 represents the total correct item placements for each construct. The cluster analysis, Fleiss kappa score, and hit ratio matrix all support the use of these developed items for their associated constructs. Based on this, I moved onto a Round 1 of data collection to test the items.

		Actual Constructs							
		AD	**CON**	**INT**	**ST**	**LP**	**NA**	**TTP**	**% Hits**
Theoretical Constructs	**AD**	**44**						44	100.00
	CON		**42**	2				44	95.45
	INT			**43**			1	44	97.73
	ST				**42**	1	1	44	95.45
	LP					**33**		33	100.00
Total Placements	209	*AD = Adaptation, CON = Connectivity, INT = Integration, TS = State, LP= Legacy Perception NA = Unclear, Multiple, or No Fit, TTP = Theoretical Total Placement*							
Hits	204								
Overall Hits Ratio	97.61								

Table B1: Hit Ratio Matrix

APPENDIX C: SUPPLEMENTARY STATISTICS

	AD	CX	CON	INT	LP	RF	RI	ST	SA	SCS	SI	SSA	TI
AD1	**0.71**	0.37	0.53	0.44	0.31	0.41	0.11	0.45	-0.03	0.19	0.39	0.37	0.40
AD2	**0.82**	0.40	0.53	0.43	0.14	0.44	0.05	0.42	0.08	0.14	0.30	0.53	0.51
AD3	**0.83**	0.34	0.59	0.46	0.08	0.47	0.10	0.48	0.02	0.17	0.30	0.63	0.60
AD4	**0.73**	0.38	0.55	0.45	0.28	0.45	0.17	0.50	0.01	0.22	0.45	0.42	0.46
CX1	0.29	**0.71**	0.30	0.45	0.25	0.46	0.07	0.44	0.02	0.14	0.40	0.28	0.37
CX2	0.46	**0.83**	0.49	0.55	0.32	0.52	0.08	0.50	0.05	0.14	0.51	0.48	0.42
CX3	0.33	**0.78**	0.38	0.51	0.37	0.46	0.17	0.42	-0.13	0.21	0.51	0.31	0.35
CON1	0.51	0.42	**0.79**	0.54	0.28	0.48	0.11	0.49	-0.06	0.22	0.44	0.45	0.46
CON2	0.51	0.38	**0.81**	0.50	0.35	0.50	0.12	0.51	0.01	0.19	0.48	0.44	0.41
CON3	0.63	0.39	**0.72**	0.45	0.15	0.47	0.09	0.50	0.05	0.08	0.32	0.63	0.57
CON4	0.38	0.31	**0.57**	0.50	0.27	0.40	0.13	0.46	-0.13	0.06	0.43	0.30	0.40
INT1	0.43	0.56	0.46	**0.77**	0.32	0.47	0.11	0.54	-0.03	0.15	0.43	0.41	0.49
INT2	0.48	0.47	0.55	**0.76**	0.38	0.51	0.12	0.52	-0.05	0.19	0.49	0.36	0.44
INT3	0.45	0.48	0.55	**0.76**	0.33	0.58	0.20	0.63	-0.02	0.16	0.50	0.52	0.57
INT4	0.36	0.45	0.48	**0.72**	0.48	0.48	0.22	0.47	-0.03	0.22	0.53	0.34	0.41
LP1	0.23	0.42	0.34	0.45	**0.84**	0.38	0.27	0.35	-0.02	0.29	0.50	0.24	0.25
LP2	0.22	0.31	0.24	0.41	**0.80**	0.31	0.26	0.34	-0.02	0.28	0.47	0.15	0.22
LP3	0.16	0.24	0.28	0.32	**0.78**	0.31	0.24	0.23	-0.03	0.24	0.33	0.09	0.11
RF1	0.49	0.50	0.51	0.55	0.32	**0.80**	0.07	0.62	-0.02	0.10	0.54	0.49	0.65
RF2	0.51	0.49	0.52	0.55	0.30	**0.80**	0.01	0.58	-0.06	0.04	0.46	0.45	0.54
RF3	0.41	0.55	0.48	0.55	0.36	**0.81**	0.02	0.58	0.00	0.02	0.47	0.43	0.53
RF4	0.39	0.43	0.52	0.51	0.35	**0.76**	0.06	0.47	0.01	0.08	0.48	0.49	0.58
RI1	0.09	0.12	0.11	0.18	0.27	0.07	**0.89**	0.14	0.03	0.57	0.19	0.03	0.06
RI2	0.17	0.12	0.19	0.18	0.27	0.05	**0.86**	0.10	-0.02	0.58	0.18	0.06	0.11
RI3	0.10	0.11	0.09	0.19	0.30	0.02	**0.89**	0.11	-0.02	0.59	0.22	-0.02	0.11
ST1	0.43	0.52	0.50	0.62	0.35	0.58	0.12	**0.77**	0.05	0.08	0.51	0.42	0.45
ST2	0.41	0.39	0.47	0.50	0.26	0.52	0.06	**0.80**	-0.02	0.06	0.46	0.39	0.43
ST3	0.55	0.41	0.58	0.56	0.34	0.48	0.23	**0.74**	-0.04	0.19	0.40	0.46	0.42
ST4	0.49	0.50	0.56	0.57	0.26	0.61	0.02	**0.81**	0.02	0.11	0.53	0.49	0.47
SA1	0.02	-0.02	-0.03	-0.04	-0.03	-0.02	0.00	0.01	**1.00**	0.00	0.01	-0.03	-0.11
SCS1	0.10	0.13	0.11	0.16	0.32	0.08	0.33	0.10	-0.11	**0.69**	0.18	0.09	0.12
SCS2	0.09	0.16	0.16	0.26	0.35	0.14	0.59	0.17	-0.09	**0.68**	0.33	0.07	0.10
SCS3	0.27	0.15	0.15	0.13	0.15	0.00	0.42	0.06	0.14	**0.76**	0.12	0.13	0.11
SCS4	0.22	0.15	0.11	0.07	0.04	-0.03	0.54	0.03	0.11	**0.71**	0.09	0.12	0.06
SI1	0.39	0.45	0.47	0.50	0.48	0.50	0.15	0.48	0.01	0.24	**0.83**	0.36	0.39
SI2	0.36	0.46	0.47	0.48	0.45	0.48	0.20	0.48	-0.05	0.18	**0.78**	0.39	0.36
SI3	0.35	0.50	0.45	0.55	0.43	0.46	0.19	0.51	0.01	0.22	**0.78**	0.38	0.36
SI4	0.35	0.53	0.38	0.49	0.33	0.49	0.16	0.45	0.08	0.20	**0.75**	0.38	0.37
SSA1	0.40	0.28	0.42	0.34	0.02	0.35	0.01	0.30	-0.06	0.13	0.20	**0.49**	0.41

SSA2	0.60	0.40	0.59	0.50	0.21	0.52	0.04	0.51	-0.05	0.15	0.44	**0.94**	0.61
SSA3	0.59	0.50	0.60	0.51	0.18	0.56	0.00	0.55	0.01	0.11	0.45	**0.92**	0.61
TI1	0.56	0.49	0.60	0.61	0.22	0.65	0.08	0.54	-0.05	0.11	0.41	0.59	**0.83**
TI2	0.49	0.31	0.46	0.48	0.17	0.52	0.19	0.40	-0.12	0.18	0.32	0.46	**0.83**
TI3	0.54	0.41	0.52	0.49	0.21	0.61	0.00	0.45	-0.11	0.08	0.42	0.57	**0.82**

AD = Adaptation, CX = Complexity, CON = Connectivity, INT = Integration, LP = Legacy Perception, RF = Representational Fidelity, RI = Replacement Intentions, ST = State, SCS = System Capability Shortcomings, SI = System Investment, SA = System Age, SSA = System Support Availability, TI = Transparent Interaction

Table C1: Round 1 Cross-loadings

	LP	**CX**	**NAD**	**NCON**	**NINT**	**NRF**	**NSSA**	**NTI**	**NRI**	**SA**	**SCS**	**SI**
CX1	**-0.49**	**0.76**	-0.24	-0.39	-0.39	-0.37	-0.33	-0.37	0.16	0.11	0.16	0.40
CX2	**-0.64**	**0.85**	-0.45	-0.52	-0.59	-0.48	-0.43	-0.47	0.08	0.04	0.09	0.25
CX3	**-0.53**	**0.83**	-0.29	-0.41	-0.49	-0.35	-0.32	-0.39	0.16	0.06	0.16	0.42
NAD1	**0.55**	-0.29	**0.75**	0.50	0.42	0.43	0.40	0.34	-0.11	-0.08	-0.18	-0.14
NAD2	**0.65**	-0.30	**0.85**	0.59	0.39	0.53	0.58	0.55	0.01	-0.04	-0.08	-0.01
NAD3	**0.67**	-0.37	**0.84**	0.59	0.40	0.53	0.66	0.57	0.11	-0.09	0.01	-0.05
NAD4	**0.56**	-0.35	**0.72**	0.50	0.43	0.45	0.37	0.39	-0.08	-0.05	-0.10	-0.18
NCON1	**0.74**	-0.45	0.61	**0.83**	0.58	0.63	0.55	0.58	0.01	-0.10	-0.03	-0.21
NCON2	**0.59**	-0.43	0.34	**0.72**	0.56	0.50	0.34	0.40	-0.11	-0.07	-0.16	-0.35
NCON3	**0.74**	-0.39	0.68	**0.78**	0.54	0.61	0.65	0.62	-0.01	-0.05	-0.06	-0.20
NCON4	**0.58**	-0.39	0.40	**0.69**	0.56	0.46	0.32	0.38	-0.19	-0.05	-0.18	-0.38
NINT1	**0.67**	-0.53	0.40	0.59	**0.80**	0.53	0.41	0.49	-0.07	-0.14	-0.14	-0.38
NINT2	**0.71**	-0.51	0.48	0.63	**0.82**	0.62	0.45	0.54	-0.08	-0.11	-0.03	-0.28
NINT3	**0.70**	-0.46	0.47	0.63	**0.82**	0.61	0.47	0.54	-0.05	-0.07	-0.04	-0.26
NINT4	**0.56**	-0.44	0.28	0.48	**0.74**	0.46	0.26	0.40	-0.26	-0.02	-0.24	-0.36
NRF1	**0.76**	-0.42	0.55	0.64	0.60	**0.85**	0.62	0.67	0.05	-0.09	0.09	-0.27
NRF2	**0.74**	-0.37	0.52	0.62	0.60	**0.84**	0.55	0.67	-0.01	-0.08	0.00	-0.25
NRF3	**0.75**	-0.44	0.55	0.61	0.61	**0.83**	0.56	0.63	-0.04	-0.07	-0.03	-0.31
NRF4	**0.70**	-0.43	0.42	0.56	0.52	**0.79**	0.60	0.67	0.02	-0.16	0.03	-0.38
NSSA1	**0.57**	-0.28	0.47	0.42	0.38	0.50	**0.76**	0.57	0.13	-0.11	0.05	-0.06
NSSA2	**0.76**	-0.43	0.59	0.61	0.51	0.69	**0.91**	0.73	0.09	-0.11	0.04	-0.26
NSSA3	**0.68**	-0.40	0.58	0.56	0.38	0.58	**0.86**	0.65	0.14	-0.20	0.05	-0.22
NTI1	**0.76**	-0.44	0.55	0.60	0.56	0.71	0.71	**0.84**	0.04	-0.12	-0.02	-0.30
NTI2	**0.68**	-0.36	0.44	0.53	0.51	0.66	0.61	**0.85**	0.05	-0.02	0.07	-0.27
NTI3	**0.70**	-0.48	0.50	0.55	0.50	0.63	0.61	**0.82**	-0.06	-0.14	0.00	-0.32
RI1	-0.06	0.13	-0.04	-0.06	-0.11	0.02	0.13	0.00	**0.81**	0.01	0.54	0.18
RI2	-0.11	0.16	-0.02	-0.09	-0.16	-0.05	0.07	-0.03	**0.85**	0.07	0.57	0.28
RI3	-0.01	0.09	0.02	-0.07	-0.06	0.06	0.16	0.07	**0.82**	-0.07	0.50	0.22
SA1	**-0.15**	0.08	-0.08	-0.09	-0.11	-0.12	-0.17	-0.11	0.01	**1.00**	0.04	0.11
SCS1	**-0.11**	0.15	-0.05	-0.13	-0.13	-0.01	0.08	0.03	0.56	0.02	**0.80**	0.31
SCS2	**-0.09**	0.13	-0.01	-0.09	-0.14	-0.01	0.08	0.03	0.47	0.03	**0.67**	0.39

SCS3	**-0.06**	0.06	-0.13	-0.03	-0.04	0.09	0.02	-0.01	0.47	0.07	**0.70**	0.10
SCS4	**-0.10**	0.10	-0.14	-0.11	-0.06	0.04	-0.03	-0.01	0.39	0.02	**0.74**	0.09
SI1	-0.35	0.35	-0.08	-0.32	-0.36	-0.33	-0.17	-0.30	0.19	0.09	0.25	**0.82**
SI2	-0.33	0.35	-0.12	-0.29	-0.29	-0.28	-0.19	-0.27	0.22	0.09	0.26	**0.78**
SI3	-0.31	0.33	-0.09	-0.27	-0.32	-0.26	-0.14	-0.28	0.28	0.08	0.29	**0.81**
SI4	-0.33	0.35	-0.09	-0.28	-0.30	-0.29	-0.22	-0.29	0.21	0.09	0.23	**0.79**

NAD = Non-adaptation, CX = Complexity, NCON = Non-connectivity, NINT = Non-integration, LP = Legacy Perception, NRF = Non-representational Fidelity, RI = Replacement Intentions, SCS = System Capability Shortcomings, SI = System Investment, SA = System Age, NSSA = System Support Non-availability, NTI = Non-transparent Interaction

Table C2: Round 2 Model 1 Stage 1 Cross-loadings

	LP	**RI**	**SI**
LV Score - CX	**-0.50**	0.16	0.43
LV Score - NAD	**0.14**	-0.02	-0.12
LV Score - NCON	**0.39**	-0.09	-0.36
LV Score - NINT	**0.45**	-0.14	-0.40
LV Score - NRF	**0.29**	0.01	-0.36
LV Score - NTI	**0.29**	0.01	-0.36
LV Score - SA	**-0.08**	0.01	0.11
LV Score - SCS	**-0.87**	0.65	0.32
LV Score - NSSA	**0.07**	0.14	-0.22
RI1	-0.51	**0.85**	0.18
RI2	-0.57	**0.83**	0.28
RI3	-0.47	**0.81**	0.22
SI1	-0.44	0.19	**0.82**
SI2	-0.40	0.21	**0.78**
SI3	-0.45	0.27	**0.82**
SI4	-0.38	0.20	**0.78**

NAD = Non-adaptation, CX = Complexity, NCON = Non-connectivity, NINT = Non-integration, LP = Legacy Perception, NRF = Non-representational Fidelity, RI = Replacement Intentions, SCS = System Capability Shortcomings, SI = System Investment, SA = System Age, NSSA = System Support Non-availability, NTI = Non-transparent Interaction

Table C3: Round 2 Model 1 Stage 2 Cross-loadings

	AD	CX	CON	INT	LP	RF	RI	ST	SA	SCS	SI	SSA	TI
AD1	**0.74**	0.30	0.51	0.42	0.07	0.43	0.12	0.48	0.08	0.17	0.15	0.39	0.34
AD2	**0.86**	0.31	0.60	0.40	-0.13	0.53	-0.01	0.35	0.04	0.06	0.01	0.57	0.55
AD3	**0.85**	0.39	0.60	0.41	-0.07	0.53	-0.11	0.40	0.09	-0.02	0.05	0.64	0.57
AD4	**0.71**	0.35	0.51	0.44	0.08	0.45	0.09	0.46	0.05	0.12	0.18	0.33	0.40
CX1	0.24	**0.72**	0.38	0.39	0.21	0.37	0.16	0.38	0.11	0.17	0.40	0.29	0.37
CX2	0.45	**0.89**	0.52	0.59	0.07	0.48	0.08	0.55	0.04	0.08	0.25	0.42	0.48
CX3	0.29	**0.81**	0.40	0.49	0.19	0.35	0.15	0.47	0.06	0.16	0.42	0.28	0.39
CON1	0.61	0.46	**0.83**	0.59	0.06	0.63	-0.01	0.50	0.10	0.03	0.21	0.51	0.58
CON2	0.34	0.43	**0.70**	0.56	0.17	0.50	0.11	0.46	0.07	0.15	0.35	0.32	0.40
CON3	0.69	0.39	**0.81**	0.54	-0.03	0.61	0.01	0.50	0.05	0.04	0.20	0.62	0.62
CON4	0.39	0.40	**0.67**	0.56	0.26	0.46	0.19	0.57	0.05	0.19	0.38	0.28	0.38
INT1	0.39	0.53	0.59	**0.79**	0.09	0.53	0.07	0.51	0.14	0.14	0.38	0.37	0.49
INT2	0.47	0.52	0.62	**0.84**	0.16	0.62	0.08	0.65	0.11	0.04	0.28	0.42	0.54
INT3	0.47	0.47	0.63	**0.83**	0.05	0.61	0.05	0.59	0.07	0.04	0.26	0.44	0.54
INT4	0.28	0.45	0.47	**0.71**	0.27	0.46	0.25	0.52	0.02	0.25	0.36	0.24	0.40
LP1	-0.03	0.17	0.12	0.14	**0.85**	0.04	0.47	0.25	0.03	0.40	0.43	-0.11	-0.02
LP2	0.05	0.17	0.16	0.20	**0.80**	0.06	0.38	0.31	0.00	0.32	0.36	-0.07	-0.02
LP3	-0.08	0.08	0.03	0.07	**0.83**	-0.07	0.43	0.17	0.03	0.38	0.36	-0.16	-0.08
RF1	0.55	0.42	0.65	0.61	-0.06	**0.85**	-0.06	0.50	0.09	-0.09	0.27	0.58	0.67
RF2	0.52	0.38	0.62	0.60	0.00	**0.84**	0.01	0.55	0.08	-0.01	0.25	0.51	0.67
RF3	0.55	0.45	0.61	0.61	0.07	**0.84**	0.03	0.58	0.07	0.03	0.31	0.54	0.63
RF4	0.42	0.43	0.56	0.52	0.03	**0.78**	-0.03	0.50	0.16	-0.03	0.38	0.55	0.67
RI1	0.04	0.12	0.06	0.10	0.45	-0.02	**0.86**	0.13	0.01	0.54	0.18	-0.14	0.00
RI2	0.01	0.15	0.09	0.15	0.40	0.05	**0.80**	0.14	0.07	0.58	0.28	-0.07	0.04
RI3	-0.02	0.09	0.06	0.05	0.43	-0.06	**0.83**	0.10	-0.07	0.51	0.22	-0.18	-0.07
S1	0.39	0.46	0.51	0.58	0.24	0.53	0.12	**0.83**	0.03	0.19	0.37	0.28	0.44
S2	0.46	0.48	0.55	0.60	0.23	0.54	0.12	**0.85**	0.07	0.15	0.36	0.37	0.43
S3	0.46	0.54	0.59	0.62	0.20	0.51	0.11	**0.83**	0.08	0.13	0.38	0.34	0.44
S4	0.42	0.45	0.53	0.57	0.28	0.54	0.14	**0.78**	0.09	0.11	0.37	0.40	0.49
SA1	0.08	0.08	0.09	0.11	0.02	0.12	0.00	0.08	**1.00**	0.05	0.11	0.17	0.12
SCS1	0.05	0.15	0.12	0.12	0.36	0.01	0.56	0.18	0.02	**0.77**	0.31	-0.07	-0.03
SCS2	0.00	0.12	0.08	0.13	0.45	0.01	0.46	0.22	0.03	**0.75**	0.39	-0.09	-0.03
SCS3	0.13	0.06	0.03	0.04	0.22	-0.09	0.47	0.00	0.07	**0.71**	0.10	-0.01	0.01
SCS4	0.14	0.10	0.10	0.05	0.17	-0.04	0.39	0.06	0.02	**0.66**	0.09	0.02	0.01
SI1	0.07	0.34	0.31	0.36	0.38	0.33	0.18	0.35	0.09	0.26	**0.82**	0.13	0.31
SI2	0.11	0.33	0.28	0.29	0.41	0.28	0.21	0.38	0.09	0.28	**0.80**	0.15	0.27
SI3	0.09	0.32	0.26	0.31	0.36	0.26	0.27	0.37	0.08	0.32	**0.81**	0.10	0.28
SI4	0.09	0.33	0.27	0.30	0.33	0.28	0.20	0.32	0.09	0.25	**0.77**	0.18	0.29
SSA1	0.47	0.30	0.43	0.38	-0.15	0.50	-0.14	0.30	0.11	-0.06	0.06	**0.89**	0.57
SSA2	0.59	0.44	0.62	0.52	-0.05	0.69	-0.09	0.53	0.11	-0.05	0.26	**0.80**	0.73
SSA3	0.58	0.40	0.57	0.39	-0.10	0.58	-0.14	0.37	0.20	-0.05	0.22	**0.80**	0.65

TI1	0.56	0.45	0.61	0.56	-0.07	0.71	-0.04	0.46	0.12	0.01	0.30	0.67	**0.85**
TI2	0.44	0.36	0.54	0.51	-0.05	0.66	-0.05	0.42	0.02	-0.07	0.28	0.60	**0.84**
TI3	0.50	0.48	0.55	0.50	0.01	0.63	0.05	0.49	0.14	0.01	0.32	0.58	**0.82**

AD = Adaptation, CX = Complexity, CON = Connectivity, INT = Integration, LP = Legacy Perception, RF = Representational Fidelity, RI = Replacement Intentions, ST = State, SCS = System Capability Shortcomings, SI = System Investment, SA = System Age, SSA = System Support Availability, TI = Transparent Interaction

Table C4: Round 2 Model 2 Cross-loadings

Path	**Coefficient**
Age -> Legacy Perception	-0.001
Complexity -> Legacy Perception	-0.199
Gender -> Legacy Perception	0.000
Legacy Perception -> Replacement Intentions	0.021
Legacy Perception -> System Investment	-0.414
Non-Adaptation -> Legacy Perception	0.221
Non-Integration -> Legacy Perception	0.282
Non-Representational Fidelity -> Legacy Perception	0.31
System Age -> Legacy Perception	-0.023
System Capability Shortcomings -> Legacy Perception	-0.082
System Investment -> Replacement Intentions	0.29
System Support Non-Availability -> Legacy Perception	0.197

Table C5: Model 1 Stage 1 Path Coefficients

APPENDIX D: RESEARCH INSTRUMENTS

Exhibit 1: Card Sort Email

Hi everyone,

I hope you are well :) For my dissertation I am conducting a card sorting exercises as part of developing new scales. If you have time, I would greatly appreciate it if you could complete the card sorting exercise.

For this exercise you just need to drag the items to the category that best fits that item. There are 19 items in total. If you are unsure of a category definition you can click the blue “i” button on the category to see the definition. I have also included them in the below table.

Category	Definition
Adaptation	The extent to which the information technology can be changed
Connectivity	The extent to which an information technology is connected to other systems in and outside of the organization
Integration	The extent to which the **internal** components of an information technology work together
State	The extent to which an information technology remembers and stores interactions
Legacy Perception	The extent to which a system is perceived as legacy. Legacy is defined as "An incumbent information system that is perceived as insufficient through a combination of social and technical factors."
Unclear	Place item here if you are unsure what an item is saying.
Fits Multiple Categories	Item does not clearly belong to a single category.
Does Not Fit Any Category	Place item here if it does not fit with any category.

The card sort exercise is available here: https://study.kardsort.com/dissertation-scale-development

Exhibit 2: Survey

Principal Investigator: Michelle Carter, PhD, Principal Investigator, Associate Professor of Information Systems, Department of Management, Information Systems, and Entrepreneurship, Washington State University, Phone: (509) 335-6415 michelle.carter@wsu.edu
Julia Stachofsky, PhD Candidate, Co-principal Investigator, Department of Management, Information Systems, and Entrepreneurship, Washington State University, Phone: (208) 791-1990, julia.stachofsky@wsu.edu

Researchers: You are being asked to take part in a research study carried out by Dr. Michelle Carter, Dr. Deborah Compeau, and Julia Stachofsky at Washington State University (WSU). This form explains the research study and your part in it if you decide to join the study. Please read the form carefully, taking as much time as you need. Please contact us if you have any questions. You can decide not to join the study. If you join the study, you can change your mind later or quit at any time. There will be no penalty or loss of services or benefits if you decide to not take part in the study or quit later.

What is this study about? This research study is being done to understand legacy systems management in organizations. Taking part in the study will take about 10 minutes of your time.

What will I be asked to do if I am in this study? If you take part in the study, you will be asked to answer a set of survey questions related to a legacy system in your organization.

Are there any benefits to me if I am in this study? There are no direct benefits to you for taking part in this study. You will contribute to research that may help organizations better manage legacy systems.

Are there any risks to me if I am in this study? The main potential risk is that you may experience some discomfort in answering questions. Remember, though, that your replies are anonymous so that we cannot link them directly to you.

Will my information be kept private? The data for this study are being collected anonymously. Neither the researcher(s) nor anyone else will be able to link data to you. The results of this study may be published or presented at professional meetings, but the identities of all research participants will remain anonymous. The data for this study will be kept for a minimum of 3 years after the completion of the study.

Are there any costs or payments for being in this study? There will be no costs to you for taking part in this study.

Who can I talk to if I have questions? If you have questions about this study or the information in this form, please contact Julia Stachofsky (julia.stachofsky@wsu.edu) or any other member of the research team. This study, IRB #20367, has been certified as Exempt by the WSU Human Research Protection Program. If you have questions or concerns about your rights as a research participant, please contact the WSU Human Research Protection Program at irb@wsu.edu.

What are my rights as a research study volunteer? Your participation in this research study is completely voluntary. You may choose not to be a part of this study. There will be no penalty to you if you choose not to take part. You may choose not to answer specific questions or to stop participating at any time.

What does my consent mean? Your consent to participate in this research (indicated by clicking on the I CONSENT button below) means that

- You understand the information given to you in this form
- You have been able to ask the researcher questions and state any concerns
- The researcher has responded to your questions and concerns
- You believe you understand the research study and the potential benefits and risks that are involved.

○ I consent, begin the study. (1)

○ I do not consent. I do not wish to participate. (2)

What best describes your position in IT?

- ○ Non-managerial Role (6)
- ○ Lower Management (1)
- ○ Middle Management (2)
- ○ Upper Management (3)
- ○ Chief Information Officer (CIO) (4)
- ○ Chief Information Security Officer (CISO) (5)

Legacy_Definition How do you define the term "legacy system"?

____

____

____

____

____

When answering the following questions for this survey please consider **ONE** legacy system in your organization.

	0 7 14 21 28 35 42 49 56 63 70
To the best of your knowledge, approximately how many years has the system been in use in the organization? ()	

Please indicate the level at which you agree or disagree with the following statements based on the **ONE** legacy system you are evaluating.	Strongly disagree (1)	Disagree (2)	Somewhat disagree (3)	Neither agree nor disagree (4)	Somewhat agree (5)	Agree (6)	Strongly agree (7)
The system components are tightly combined. (1)	○	○	○	○	○	○	○
Internal modules of the system work together. (2)	○	○	○	○	○	○	○
System components are integrated. (3)	○	○	○	○	○	○	○
System components are dependent on each other. (4)	○	○	○	○	○	○	○
The system can interface with other systems. (5)	○	○	○	○	○	○	○
The system is connected to other systems in the organization. (6)	○	○	○	○	○	○	○
External systems can connect to this system easily. (7)	○	○	○	○	○	○	○

The system communicates over a network. (8)	○	○	○	○	○	○	○
The technical characteristics of this system make it complex. (9)	○	○	○	○	○	○	○
The system depends on sophisticated integration of technology components. (10)	○	○	○	○	○	○	○
There is considerable technical complexity underlying the system. (11)	○	○	○	○	○	○	○
The system keeps records of events. (12)	○	○	○	○	○	○	○
The system stores data about previous interactions. (13)	○	○	○	○	○	○	○
The system stores data about previous system states. (14)	○	○	○	○	○	○	○
The system saves information between uses. (15)	○	○	○	○	○	○	○
The system can be changed. (16)	○	○	○	○	○	○	○

It is easy to change the system. (17)	○	○	○	○	○	○	○
The system is easily modified. (18)	○	○	○	○	○	○	○
Changing the system for new functionality is possible. (19)	○	○	○	○	○	○	○
When employees use the system, they find the content it provides them is sufficiently complete. (20)	○	○	○	○	○	○	○
When employees use the system, they find the content it provides them is sufficiently clear. (21)	○	○	○	○	○	○	○
When employees use the system, they find the content it provides them is sufficiently correct. (22)	○	○	○	○	○	○	○

	Strongly disagree (1)	Disagree (2)	Somewhat disagree (3)	Neither agree nor disagree (4)	Somewhat agree (5)	Agree (6)	Strongly agree (7)
When employees use the system, they find the content it provides them is sufficiently meaningful. (23)	○	○	○	○	○	○	○
When employees use the system, they have seamless access to the content they need. (24)	○	○	○	○	○	○	○
When employees use the system, they have no difficulty obtaining the content they need because of the system's interface. (25)	○	○	○	○	○	○	○
When employees use the system, they have no difficulty obtaining the content they need because of physical characteristics of the device. (26)	○	○	○	○	○	○	○

We do not encounter difficulties in obtaining needed system support services. (27)							
We can easily obtain the support resources necessary to continue operating this system. (28)							
Support for this system is readily available. (29)							
There are notable limitations in the ability of this system to meet our needs. (30)							
We would like to have many capabilities that are not supported by this system. (31)							
The performance of this system is highly inadequate. (32)							
The functionality of this system is highly inadequate (33)							

Significant organizational resources have been invested in this system. (34)							
We have committed considerable time to the operation of this system. (35)							
We have committed considerable money to the operation of this system. (36)							
The financial investments that have been made in this system are substantial. (37)							
We plan to replace this system with another system. (38)							
Our intention is to replace this system with an entirely different system. (39)							
We will be seeking to implement a replacement to this system. (40)							

This system is considered a legacy system in the organization. (41)	○	○	○	○	○	○	○
Other employees would consider this a legacy system. (42)	○	○	○	○	○	○	○
I think the system is a legacy system. (43)	○	○	○	○	○	○	○
If 2+2 = 4 select "Somewhat disagree" (44)	○	○	○	○	○	○	○

System_Type What type of system is the legacy system you selected?

- ○ Accounting Information System (7)
- ○ Customer Relationship Management System (CRM) (10)
- ○ Decision Support System (DSS) (1)
- ○ E-Commerce System (4)
- ○ Enterprise Resource Planning System (ERP) (2)
- ○ Executive Information System (13)
- ○ Human Resource Information System (HRIS) (8)
- ○ Knowledge Management System (12)
- ○ Marketing Information System (6)
- ○ Medical Information System (14)

○ Office Automation System (9)

○ Supply Chain Management System (SCM) (11)

○ Other (Please Specify) (3)

____

Industry What sector of industry is your organization in?

____

Gender What is your gender?

○ Woman (1)

○ Man (2)

○ Non-binary (3)

○ Prefer to self-describe (5)

____

○ Prefer not to say (4)

Age What is your age?

	18 28 38 47 57 67 77 87 96 106 116
Age ()	

Exhibit 3: R Analysis Code

Card_Sort_Analysis.R

```
# Analysis file for conducting inter-rater reliability analysis

# Install and import packages
install.packages("irr")
install.packages("tidyverse")
library(irr)
library(tidyverse)

# Import combined casolysis file
setwd('C:/Users/stach/OneDrive/Documents/School/PhD/Dissertation/Scale_Develo
pment')
data <- read.csv("Combined_Card_Sort_Data.csv")

# Pivot data to a matrix where rows are items and columns are raters
data_matrix <- as.matrix(spread(data, Rater_ID, Rating)[,-1])

# Calculate Fleiss' Kappa and print
result <- kappam.fleiss(data_matrix)
print(result)
```