The Potential of Niklas Luhmann's General Theory of Social Systems for Research on Agile Methodologies

Completed Research Paper

Andreas Drechsler
University of Duisburg-Essen
andreas.drechsler@icb.uni-due.de

Tobias Trepper
University of Duisburg-Essen
tobias.trepper@icb.uni-due.de

Abstract

This paper highlights the potential of Niklas Luhmann's general theory of social systems for IS research on agile methodologies. Based on the common denominator of communication, a high degree of compatibility of Luhmann's systems theory with agile principles and values is found. Its value for IS research is illustrated by three sample research avenues: explaining social mechanisms in agile project environments, grounding agile methodologies in theory, and designing enhanced agile techniques for practice. IS researchers interested in explanatory and theory-building research can use Luhmann's theory as a novel lens to arrive at new insights about the effects of agile methodologies in project teams. Design science-oriented IS researchers and practitioners find opportunities for developing and enhancing agile methodologies and techniques in a theoretically grounded and empirically-validated way, based on systemic principles.

Keywords

Agile methodology, Social systems, Social theory, Luhmann, Communication, Autopoiesis

Introduction

Over the last fifteen years, agile methodologies such as Extreme Programming (Beck and Andres 2004) or SCRUM (Schwaber 2009) were developed in practice, in order to cope with common challenges of software projects. While they have received certain attention in IS research (Balijepally et al. 2009; Goh et al. 2013; Lee and Xia 2010; Maruping et al. 2009; Persson et al. 2012), so far, the selection of theories used has been based on their suitability for the aspects of interest. Examples are trust, expertise coordination, control theory, social facilitation, or individual and group information processing. Since agile methodologies were developed in practice, they lack a solid and coherent theoretical underpinning. This makes it challenging to find a suitable theoretical lens which is wide enough to provide a broader perspective on agile methodologies beyond the isolated analysis of selected issues. It also leads to a lack of theoretically grounded body of descriptive and prescriptive knowledge to propose and evaluate enhancements to agile methodologies and techniques by means of design science research (Gregor and Hevner 2013; Hevner et al. 2004).

Against this backdrop, it is now argued that the general theory of social systems, developed by the German sociologist Niklas Luhmann (1995), is a theory which is suitable for such a broad theoretical perspective. In order to show this, we first outline the fundamentals and principles of Luhmann's theory as well as the principles and values of agile methodologies. Afterwards, we analyze the extent of compatibility between them in greater detail and outline three sample avenues for further research: one for research to explain real-world phenomena in agile environments, one for research specifically aimed at theory building, and one for design-oriented research for novel agile methodologies and techniques. The paper ends with a brief discussion and conclusion.

Theoretical foundations

Luhmann's general theory of social systems

Fundamentals of systems theories

The term "systems theory" is an umbrella term for several theories of actions and effects in systems and society which all share a systemic view of the world. Systems theorists understand the world as a system consisting of subsystems, which are in more or less strong correlation with each other. This relationship is called "structured coupling" (Maturana 2002). However, the main proponents' perspectives on systems theories are – sometimes fundamentally – different and yet in some ways complementary. As an example, the concept of "autopoiesis" was developed in the early 1970s in the context of neurobiology (Maturana 2002), but also forms an integral part of Luhmann's general theory of social systems (Luhmann 1995). Luhmann's theory will be the center of attention in the remainder of this paper.

Types of systems and social systems

Luhmann (1995) differentiates four different types of systems: organic systems (e. g., living organisms), mental health systems (e. g, awareness), social systems, and machines (as long as they are able to observe). In contrast to other theories of social systems, individuals and groups of people do not form a system on their own. In contrast, a person is a collection of multiple systems. For Luhmann, social systems consist of communication. He distinguishes a hierarchy of three types of social systems: interactions, organizations, and societies. Interactions represent very short contacts, while societies are the largest and most complex persistent social systems whose boundaries are only bound by the limitations of communication. Organizations are in between interactions and societies; they are more complex than interactions, but still subsystems of societies.

Systems and operations

In Luhmann's (1995) view, a system can only survive only if it is able to operate. Each type of system differentiated above shares the same two basic principles of operation: Observing other systems and distinguishing between itself and the environment. For this reason, a machine is only a system when it is able to observe. A system comes into existence (and persistence) only if its operations are executed continuously, one after another. This concept of ongoing operations creating and recreating the system itself is called autopoiesis. A system is autopoietic when its components are produced and reproduced by the elements it consists of. Every single object of a system is defined through these objects and operations. In other words, there is no input and no output of any objects to and from the system. However, this does not mean that a system does not have a relationship to its environment. It only means that its objects do not pass the boundary between a system and its environment.

Social systems and communication

Luhmann (1995) states that communication is the main and only element each social system consists of and by which it reproduces itself. This implies an abstraction from humans and their minds. Each instance of communication goes through a three-point selection process performed by the participating sender and receiver. Each of these three steps has to be done synchronously to make communication happen as an emergent action. In the first selection process, the sender selects which information is going to be communicated. In the second selection, the actual message is selected and the third selection happens on the receiving end and consists of the selection of the understanding of the message through interpretation.

For each step, there are always several possible alternatives to select. The information to be communicated has to be chosen by the sender. After this decision the sender has to wrap this information into a message to be sent through an appropriate media. The choice of media leads to a further transformation of the initial piece of information. The receiver's task is now to understand the received message. However, the receiver can only base their understanding on their own perceptions of the received message and interpret the information using their own mental model of reality. This third selection is the most crucial one because it is the key for successful communication.

A closer look at the three steps leads to the conclusion that there is an endless number of possible outcomes for each of the step. Luhmann (1995) calls them "contingencies". On first sight, this makes a successful communication very unlikely: In each selection, one of several different possibilities is chosen (selecting what to send, selecting the media, selecting the interpretation of the perceived message). Since there is a sender and a receiver involved, there are even two sources for contingencies. Luhmann calls this "double contingency". However, this double contingency unexpectedly leads to the effect of lowering the uncertainty of communication because the sender and the receiver get (and need) to observe and respond to each other in order to ensure and verify a successful communication effort. This creates and re-creates social order, contributing to the persistence of the social system.

The initial uncertainty is further decreased by three forms of media: Speech lowers the uncertainty in the selection of information. Communication media – such as print, radio, or the internet – lowers the uncertainty while communicating. And symbolic or success media – such as money, power, love, law, or religion – can boost the identification of communication and avoid its refusal. Furthermore, all three types of selection are supported by the universal media types of sense and meaning.

For the self-preservation through continuous re-creation (autopoiesis) of a social system, a continuous flow of communication is necessary. Should there be no more communication happening inside a social system its existence comes to an end.

Systems and their environment

Luhmann (1995) differentiates the outside of a system into its direct environment and the other world. At the moment a system becomes aware of a part of the outer world this part turns into a part of the direct environment. This process is called differentiation, which is a direct operation of the system. In this perspective, a system is always different from its environment and, for a system, its environment only exists through its perspective. In order to be able to distinguish between themselves and their environment and to perceive and assess differences of any kind, systems need to observe their surrounding environment. As mentioned above, systems are not closed to external influences. However, the decision whether (and how) a perceived external observation of an influence will actually have any impact on the system and its operations is always taken by the system itself and cannot be predicted. In addition, this openness to the environment allows for a structural coupling between systems.

While the sections above only could summarize – and sometimes simplify – the fundamentals of Luhmann's theory, these fundamentals are sufficient to allow their relation and application to the principles and values of agile methodologies, which are outlined below.

World and management view, values, and principles of agile methodologies

Agile methodologies were developed in response to shortcomings of traditional project management for software projects (Highsmith 2002). A basic assumption of agile project management is that all projects are taking place in a chaotic and complex world. The management view in this context is no longer a "heroic" one as in traditional project management methodologies, but can rather be characterized as "post-heroic", with an emphasis on collaborative and interactive thinking. It is acknowledged that a project does usually not develop in a straight line, but can be rather chaotic. The corresponding methodology thus needs to be flexible. However, this does not mean that the anticipatory component, which is prevalent in classic project management, is completely excluded in agile project management. Furthermore, the management perspective on agile projects puts people first. Everyone's ability to interact, adapt, self-organize but also the desire to provide good performance and thus to create the best work possible is emphasized.

The four underlying values of agile project management are named in the agile manifesto (Beck et al. 2001a):

1. "Individuals and interactions over processes and tools
2. Working software over comprehensive documentation
3. Customer collaboration over contract negotiation

4. Responding to change over following a plan"

These values form the foundation for the twelve principles of the agile manifesto (Beck et al. 2001b):

1. "Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
4. Business people and developers must work together daily throughout the project.
5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
7. Working software is the primary measure of progress.
8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
9. Continuous attention to technical excellence and good design enhances agility.
10. Simplicity--the art of maximizing the amount of work not done--is essential.
11. The best architectures, requirements, and designs emerge from self-organizing teams.
12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly."

In sum, agile project management tries to cover the range of relevant "hard factors" as well as the "soft factors" through its processes, practices, and techniques, placing cooperation and communication above all. Here, agile methodologies provide a comprehensive framework, orchestrating several processes, practices, and techniques. For example, the SCRUM methodology prescribes "sprints" (project phases), "daily scrum meetings", or the role of the "SCRUM master" (Schwaber 2009). Here, however, a holistic application of a chosen methodology is assumed (Cockburn 2007). A phased introduction of individual aspects of a methodology can lead to problems, since using only certain elements of a methodology might lead to "loose ends" and overhead without yielding the benefits of a holistic application of the elements working together. In addition, all project participants must be highly skilled and be able to work in a disciplined way. Furthermore, the possibilities for formal project monitoring are limited because of the value emphasizing working software over documentation. As a result, agile project management should not be used in critical systems where a high degree of monitoring and control is needed (Cockburn 2007). Due to the high communication overhead and constant necessity for cooperation, agile project management is more suitable for small than for large project teams.

Extent of compatibility between Luhmann's theory and agile methodologies

In this section, the extent of compatibility between Luhmann's general theory of social systems and agile methodologies is discussed on three levels: world and management view, values, and principles. In order to do so, an agile project system is seen as a special instance of a social system in the sense of Luhmann's theory.

World and management view

As mentioned above, the key characteristic of the world view of agile methodologies lies in the assumption that projects take place in a chaotic and complex environment and agile project management is tasked with coping with chaos and complexities emerging over the duration of a project. This is well in line with

the systemic perspective: There, a project system observes emergent phenomena from its environment and decides whether and how to react, based on its observations and its inner logic. In the management view, people are put first in agile methodologies and qualities such as interaction, adaption, and self-organization are emphasized. In Luhmann's theory, people are not part of social systems. However, the emphasis on interaction provides the direct link to the constituent element of social (and project) systems: communication. Furthermore, continuous adaption and self-organization are two effects of the ongoing process of self-reproduction (autopoiesis) and the reactions of a project system on external influences. In conclusion, the world and management view of systemic and agile thinking are not congruent, but very closely aligned and compatible.

Values

The first value of the agile manifesto mentions individuals and interactions. In the systemic view, it is not the human individual, but their interaction (communication), and their abilities to observe and to differentiate, which makes the creation and existence of persistent project systems possible. Here, the systemic view can be considered an abstraction from the complex entity "human" that highlights their role as communicators keeping a project system in existence.

In the second agile value, working software is emphasized over documentation. To link this value to a systemic perspective, it makes sense to extend the perspective beyond the project system itself and consider the customer system as well. In a systemic view, these two systems are part of each other's environment and thus not able to influence each other directly; they can only "irritate" each other. For example, the customer system communicates requirements, and the project system can respond by communicating documentation or a working prototype. As shown above, the process of communication is subject to a "double contingency" and thus, communication success is not to be taken for granted. The customer system cannot escape this challenge when communicating requirements. In contrast, the project system can to a certain extent, when responding to the customer by selecting a different media: not spoken or written text, but a working software prototype. By doing so, the ambiguities in the third – interpretative – step of the communication process are reduced for the customer system since it can experience part of the future software which can be assumed to have less room for misinterpretations than written or spoken text. The argumentation for the link of the third agile value – collaboration over contract negotiation – to the systemic perspective goes along similar lines. Here, it is assumed that mutual and direct communication helps resolving the challenge of the double contingency of requirements communication better than written text in form of contracts.

For the fourth agile value – responding to change over following a plan – a direct link to the concept of autopoiesis of the project system can be drawn: As stated above, the reaction of social systems to external influences cannot be predicted from outside the system and follows their own inner logic. Thus, a project system can respond to an external influence – a request for change – by sticking to the original plan (and not changing) or by responding to change in form of self-reorganization as part of the ongoing process of autopoiesis. In systems theories, it is a basic assumption that systems which adapt to their environment are better suited to fulfill external requirements than systems which choose not do so.

Overall, it becomes obvious that there is a high congruence between agile values and the systemic perspective on social systems and that Luhmann's theory appears well suited to provide rationales for agile values.

Principles

Since the space available does not allow an in-depth discussion of every of the twelve principles listed, we group them according to the values discussed above, if possible. The respective links of the values to Luhmann's theory were already discussed in the previous section.

Agile Value	Agile Principles	Relation to Luhmann's theory
1: Individuals and interactions	5, 6	Communication as constituent element
2: Working software	1, 3, 7	Effectiveness of different media
3: Customer collaboration	4	Effectiveness of different media

4: Respond to change	2, 9, 11, 12	Autopoiesis of social systems

Table 1: Agile values, agile principles, and their relation to Luhmann's theory

Two agile principles cannot easily be associated with an agile value. Principle eight emphasizes striving for sustainable development and the need to maintain a constant pace over time. In a systemic perspective, this corresponds to a stable and sustainable structural coupling between the project system and the customer system. Principle ten speaks of simplicity being an essential quality for agile projects. This is the only principle not directly being relatable to Luhmann's systems theory. However, an even more general law of systems theories (cybernetics) characterizes simplicity as a relative quality of systems. Specifically, following the law of requisite variety, a system needs to have a greater or equal variety (and thus, internal complexity) than another system (or its environment) in order to control it (Conant and Ashby 1970). For social systems such as organizations, Weick (1995) concludes that organizational processes need to be of sufficient variety to control inputs of a certain variety. In case of too simple processes, important external inputs may not be processed by the organization, leading to inappropriate operations and responses to external stimuli. Applying this to the specific case of agile project organizations, agile methodologies and techniques should not be designed in a too simplistic way so that they are not able to deal with the variety of their environment (which includes the customer system). Thus, in a systemic perspective, the tenth agile principle needs to be criticized as neglecting the aspect of complexity and variety of the environment of the project system.

Overall, with the exception of a single principle out of fourteen, the agile world and management view, agile values, and agile principles were found to have a high extent of compatibility to the fundamental principles of Luhmann's general theory of social systems. On this foundation, three future high-level research directions for research on agile methodologies are outlined.

Sample applications of Luhmann's theory to research on agile methodologies

Explanatory research: A new theoretical lens for analyzing communication in agile software projects

As implied in the first ("interaction") and third ("collaboration") value, communication is of great importance in agile methodologies and it is assumed that face-to-face communication is superior to processes, tools, or contracts, in order to cope with the challenges of software development. At the same time, Luhmann's theory shows that successful unambiguous communication is a challenge due to the double contingencies which exist in any instance of communication and communication success cannot be taken for granted. And due to the general importance of communication and cooperation in agile methodologies, there is always the possibility of crucial misunderstandings and communication problems, threatening project success.

However, many of the theories (for example, social facilitation (Balijepally et al. 2009) or expertise coordination (Maruping et al. 2009)) employed in previous research on agile techniques or methodologies implicitly assume successful communication between members of the project system or the project and the customer system. Here, a specific lens on communication (and miscommunication)-related phenomena can enhance the understanding of prerequisites or context factors which inhibit or foster successful communication in agile contexts and actually allow the application of theories which assume successful communication. Due to the central role of mental models and the autopoietic nature of social systems in Luhmann's theory, an interpretive approach and methods with a high researcher involvement such as participant observation or ethnography appear suitable to conduct explanation-oriented research through a systemic lens.

Theory-building research: Luhmann's general theory of social systems as theoretical underpinning of agile methodologies

Due to the high extent of similarity between agile principles and values and the fundamentals of Luhmann's theory, it appears not only worthwhile to use it as novel explanatory lens for specific issues in agile projects, but also to use it as a starting point to formulate a theory of agile project management. Such

a research endeavor can follow the approach Anderson et al. (1994) chose to derive a theory of quality management based on Deming's fourteen principles of quality management (Deming 1981). They understood theory in the sense of Whetten (1989) who points out the importance of four elements for a theory: 1) variables, constructs, or concepts ("what?"), 2) the relationships between these elements ("how?"), 3) rationales and assumptions for the inclusion of the elements and the nature of their relations ("why?"), and 4) the context or boundary constraints for the applicability of the theory. Anderson et al. (1994) built an initial set of 37 constructs out of Deming's principles in a Delphi session and refined it to seven constructs. They added relationships through logical deduction, related their theoretical model to existing literature in order to account for the "why?" part, and, in the end, formulated testable propositions.

Analogous to the principles and values of agile methodologies, Deming's principles were formulated as prescriptive guidelines by practitioners (or rather, a single practitioner) with a practitioner audience in mind. They likewise lack a theoretical foundation, but proved effective and useful in practice. Therefore, the approach employed by Anderson et al. (1994) appears suitable to formulate a theory of agile project management as well. Due to the high extent of compatibility as shown above, Luhmann's theory appears to be an especially suitable candidate for a key building block of the "why" part of theory development. In addition, other theories employed in existing research on agile projects and methodologies are also candidates for underpinning specific aspects in the agile context.

Design-oriented research: Extending the agile technique of pair programming with systemic principles

While agile methodologies acknowledge the existence of a chaotic environment, they do not directly address the resulting challenges through readily applicable coping practices or techniques. Instead, they address them in an indirect way: By placing each individual and their interactions above processes and tools, agile methodologies provide an "arena" for cooperation so that the team members can tackle and solve the issues they identify through communication. But in turn, successful communication cannot be taken for granted as discussed above. Based on the compatibility between agile and systemic principles, it is now conceivable that it is possible to enhance existing agile techniques based on systemic principles, in order to provide readily applicable instruments to directly tackle these core issues. The development of such techniques (models or methods) falls into the realm of design science research (Gregor and Hevner 2013; Hevner et al. 2004).

One agile technique providing such an arena for cooperation and problem discussion and solving is pair programming. Pair programming is the act of two programmers working together on one workstation. One of the programmers is coding, while the other one controls the work and supports his partner (Beck and Andres, 2006). Based on a literature review and their own controlled experiments, Balijepally et al. (2009) conclude that pair programming indeed not only leads to the same or even higher software quality compared to individual programming, but also contributes to increased work satisfaction and confidence of the individuals forming the pair. However, in their experiment they relied on Information Systems students performing a pair programming exercise. In reality, with experienced programmers, it is conceivable that diverging perceptions of the "true way" of coding or other fundamental differences in the programmers' mental models can, in the worst case, lead to open and irreconcilable conflicts.

Therefore, from a systemic perspective, it appears useful to make both programmers aware of issues such as potential diverging perceptions of the truth or social conflicts and conduct pair programming as deliberate, ongoing, mutual irritation, as it is called in systemic terminology. This not only makes them aware of such issues and gives them an explicit mandate to deal with any upcoming issues themselves, but also includes the task to proactively take a constructive critical stance as well as to be open for a change to their mental models. The question is now whether systemic pair programming is indeed more effective than non-systemic pair programming and how the added task of mutual irritation contributes to its efficiency. Thus, the goal for design science research here is to develop specific, theoretically grounded guidelines for a systemic pair programming technique, evaluate them in practical settings, and refine them further, based on the evaluation results.

Discussion and conclusion

This paper highlighted the potential of Niklas Luhmann's general theory of social systems to contribute to the field of IS research through provision of a new lens to view social systems. It was shown that the theory is especially suited for research on agile methodologies and techniques due to a high extent of compatibility, ranging from the underlying world and management view to the four agile values and twelve agile principles. Even on this high level of abstraction, a shortcoming and potential improvement of an agile principle (no. 10) could be shown when analyzed through a systemic lens. To highlight the specific potential of Luhmann's theory for future applications in IS research on agile methodologies, three general research avenues were outlined and illustrated by means of a sample outline of a research question and research design. For explanation-oriented research this constitutes a focus on communication in agile projects, to be researched through an interpretive approach. It was also shown that Luhmann's theory can play a large role in theory-building research for agile project management, based on an analogous effort from the field of quality management (Anderson et al. 1994). And finally, for design-oriented research it provides a rationale for the proposition of design knowledge for the theoretically-grounded design and enhancement of agile techniques and methodologies, which need to be evaluated in practice afterwards. Furthermore, due to the abstract and general nature of Luhmann's theory and the importance of the social context and communication for many IS domains, the authors regard it as worthwhile to analyze the potential for applications beyond agile methodologies. It is also a task for future research to investigate relations, synergies or incommensurabilities of Luhmann's theory to other social theories.

REFERENCES

Anderson, J. C., Rungtusanatham, M., and Schroeder, R. G. 1994. "A Theory of Quality Management Underlying the Deming Management Method," *Academy of Management Review* (19:3), pp. 472–509.

Balijepally, V., Mahapatra, R., Nerur, S., and Price, K. H. 2009. "Are Two Heads Better Than One for Software Development? the Productivity Paradox of Pair Programming," *MIS Quarterly* (33:1), pp. 91–118.

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

Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J., and Thomas, D. 2001a. "Manifesto for Agile Software Development," Retrieved October 3, 2012, from http://www.agilemanifesto.org.

Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J., and Thomas, D. 2001b. "Principles behind the Agile Manifesto," Retrieved October 3, 2012, from http://agilemanifesto.org/principles.html.

Cockburn, A. 2007. *Agile software development: the cooperative game*, Upper Saddle River, NJ: Addison-Wesley.

Conant, R. C., and Ashby, R. W. 1970. "Every good regulator of a system must be a model of that system †," *International Journal of Systems Science* (1:2), pp. 89–97.

Deming, W. E. 1981. "Improvement of quality and productivity through action by management," *National Productivity Review* (1:1), pp. 12–22.

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

Gregor, S., and Hevner, A. R. 2013. "Positioning and Presenting Design Science Research for Maximum Impact," *MIS Quarterly* (37:2), pp. 337–A6.

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

Highsmith, J. 2002. *Agile software development ecosystems*, Boston et al.: Addison-Wesley.

Lee, G., and Xia, W. 2010. "Toward Agile: An Integrated Analysis of Quantitative and Qualitative Field Data on Software Development Agility," *MIS Quarterly* (34), pp. 87–114.

Luhmann, N. 1995. *Social systems*, Stanford, Calif.: Stanford University Press.

Maruping, L. M., Zhang, X., and Venkatesh, V. 2009. "Role of collective ownership and coding standards in coordinating expertise in software project teams," *European Journal of Information Systems* (18:4), pp. 355–371.

Maturana, H. 2002. "Autopoiesis, Structural Coupling and Cognition: A history of these and other notions in the biology of cognition," *Cybernetics & Human Knowing* (9:3-4), pp. 5–34.

Persson, J. S., Mathiassen, L., and Aaen, I. 2012. "Agile distributed software development: enacting control through media and context," *Information Systems Journal* (22:6), pp. 411–433.

Schwaber, K. 2009. *Agile Project Management with Scrum*, Microsoft Press.

Weick, K. E. 1995. *Sensemaking in organizations*, Thousand Oaks: Sage.

Whetten, D. A. 1989. "What constitutes a theoretical contribution?," *Academy of Management Review* (14:4), pp. 490–495.