

IT Project Management from a Systems Thinking Perspective

Pascal van Eck *María Laura Ponisio*

University of Twente, Department of Computer Science,
P.O. Box 217, 7500 AE Enschede, The Netherlands
{p.vaneck,m.l.ponisio}@utwente.nl

Abstract

Purpose – We propose a Systems Thinking approach to the study of IT project management and show how this approach helps project managers in controlling their projects. Systems Thinking is a holistic problem solving method in which system behaviour emerges from the interaction of system components.

Design/methodology/approach – To illustrate our proposal, we present an example model of the dynamics of IT outsourcing projects. The example model explains these dynamics in terms of feedback loops consisting of causal relations reported in the outsourcing literature.

Findings – The model provides insight in how coordination, trust, information exchange and possibilities for opportunistic behaviour influence each other and together influence delivery quality, which in turn influences trust.

Practical implications – The integration of these insights provided by applying the Systems Thinking perspective helps project managers to reason about how their choices influence project outcome, and therefore, to better control their projects.

Research implications – From the viewpoint of academic research, we discuss how the Systems Thinking perspective can serve as an additional tool in the study of IT project management, and how results obtained in a specific case can be generalized. To develop methods and tools specific for IT project management, applying the Systems Thinking perspective also calls for additional research in which this perspective is itself the object of study.

Originality/value – The value of our proposed approach is that it provides a way to integrate isolated theoretical insights into a practical tool for project control.

Keywords: Systems Thinking, Systems Dynamics, IT Project Management, IT Outsourcing, Coordination, Trust

1 Introduction

Many IT projects fail in one way or another: while the exact percentage depends on the definition of failure and way of measuring, failure rates reported are usually higher than 50% ([Gemino *et al.*, 2007](#)). In this paper, we define IT projects as projects in which some desired IT functionality is created for an internal, non-commercial or external, commercial customer organization. This can be in the form of in-house software development by software engineering or in the form of outsourcing to an external organization. The desired IT functionality can be implemented in the form of a new information system developed from scratch, or in the form of extra functions of an existing software application.

IT projects are complex systems that consist of different kinds of components, such as stakeholders, work products, procedures, and project controls. The interaction of these components determines the outcome of the project. Similar to [van Ekris \(2008\)](#), we use a broad definition of

Shorter version published as “IT Project Management from a Systems Thinking Perspective: A Position Paper”, eProceedings of the 3rd International Research Workshop on Information Technology Project Management (IRWITPM), Paris, France, December 12th-13th, 2008

© Centre for Telematics and Information Technology

project outcome: project outcome is any change in the material or mental state of the customer caused by the project. Project outcome is not determined at a single moment in time, but by a process consisting of multiple interactions with members of the customer organization. These interactions include the delivery of partial project results, consultations with members of the customer organization, and delivery of progress reports. In other words, the project outcome is governed by the dynamics of a complex system, which IT project management aims to control.

How can we control these dynamics? The dynamics of a complex system emerges from the interactions between the components of the system, and can often be explained by feedback loops within the system. For instance, consider a project in which partial project results have been rejected by users because they found that the results do not fit their way of working. This may lead to a decrease in trust in the project, which may lead to a decrease in knowledge sharing between the project and the (disappointed) project users, which in turn leads to subsequent project results that fit even less.

In other words, dynamics cannot be fully explained by isolated correlations between two variables, as this neglects feedback loops. However, current research in the Information Systems area mainly focuses on such isolated correlations, for instance the impact of top management support (independent variable) on project success (dependent variable). Therefore, a holistic approach that studies parts of the system not in isolation, but as the parts of a greater whole has the potential to extend current research insights in a way that practitioners need to successfully manage projects.

Systems Thinking (Sterman, 2000; Maani and Cavana, 2000), which we discuss in Section 2, is an approach to problem solving based on a holistic view in which a problem is explained as emerging from the dynamics of the underlying system. The position that we take in this paper is that applying Systems Thinking in the area of IT project management enriches the area, providing a way to capture the dynamics of IT projects. This helps project managers to understand how their decisions influence project success, which in turn helps them to stay in control.

Specifically, the contribution of this paper is a proposal to apply Systems Thinking in the area of IT project Management (Section 3). Using an application in the area of outsourcing projects as an example, we show how applying Systems Thinking in IT project management helps to understand dynamics (Section 4). In turn, a better understanding of the dynamics enables project managers to better understand the consequences of the control mechanisms that they choose and of the control actions they take. We conclude by presenting implications for research and practice (Section 5).

2 Systems Thinking: Background and related work

Systems Thinking is an approach to problem solving in which a problem is viewed as emerging from the interactions of the components of a system. Systems Thinking analyses a problem by modelling the relations between the components of the system and studying the behaviour that is jointly created by the interactions between these components. This creates a holistic view of the problem that helps to identify the dynamics of the system that results from interactions between its components. These dynamics would remain hidden in an analytic approach in which system components are studied in isolation.

The history of Systems Thinking can be traced back to research in general (mathematical) systems theory as developed in the mid of the previous century. Systems Thinking grew out of Forrester's work in which he applied general systems theory to organizational systems in business. Two paradigms can be distinguished in Systems Thinking: Systems Dynamics and Soft Systems Methodology (SSM). Systems Dynamics (the paradigm pioneered by Forrester) focuses on modelling feedback loops consisting of cause-and-effect relations between system components and on computer simulations to study system dynamics. Systems Dynamics primarily deals with quantitative models, but as early as 1983 qualitative approaches have been proposed (Wolstenholme and Coyle, 1983; Wolstenholme, 1983). Other examples of qualitative applications have been presented by Dangerfield (1999), and by Swart and Powell (2006). Richardson (1999) names qualitative

approaches to systems dynamics as the first trend in his 1999 reflection on the field. This paper follows the tradition of qualitative Systems Dynamics.

The Systems Dynamics approach is what Pollack (2007) calls a ‘hard’ approach to systems thinking. According to Maani and Cavana (2000), in the ‘hard’ approaches the systems model is seen as a representation of the real world. Further assumptions are that stakeholders agree on a “clear and single dimensional (single objective)” problem definition (Maani and Cavana, 2000). Maani and Cavana (2000) also note that in the ‘hard’ approaches, organizational and people issues are not taken into account, and data is usually quantitative. Wolstenholme (1993) puts this into perspective: while his approach is ‘hard’ in the sense that his model is seen as an objective representation of the real world and there is a clear and single-dimensional problem definition, data is qualitative and organizational issues are taken into account (the latter even turns out to be crucial in his case study). We concur with Wolstenholme (1993) and Pollack (2007) and assume that in ‘hard’ approaches, data is objective and based on empirical observations, but not necessarily quantitative. This is also in-line with Yin’s (2003) approach to case study research.

According to Maani and Cavana (2000) and Pollack (2007), in ‘soft’ approaches (of which Peter Checkland’s Soft Systems Methodology (Checkland, 1981) is probably the best-known example), the dynamic model is not seen as an objective representation of the real world, but as “a way of generating debate and insight about the real world” (Maani and Cavana, 2000). Thus, in the ‘soft’ approaches, a model is subjective and this reflects that different stakeholders may have different perceptions about how the real world behaves, and also about what the problem actually is.

In fact, both ‘hard’ and ‘soft’ approaches have a place in IT project management. A ‘soft’ approach has been used by Johnstone *et al.* (2006) to develop a holistic framework of conflict and conflict resolution in IT projects. His framework appears to contain one (short) feedback loop: governance provides process adjustments to resolve conflicts, and observes whether this helps. This loop seems to be a typical control loop that crosses the system boundary: where a controlling entity (‘governance’) observes a system-to-be-controlled (the IT project) and conducts control actions (the process adjustments). The results of Johnstone *et al.* can be the starting point for a subsequent ‘hard’ approach that further explains the behaviour of the system-to-be-controlled in terms of internal feedback loops.

Mutschler *et al.* (2007) and Mutschler and Reichert (2008) use a ‘hard’ approach to model the factors that influence implementation costs of what they call ‘process-aware information systems’: complex enterprise systems that monitor or control business systems. Workflow management systems and workflow management functionality within enterprise information systems are examples of process-aware information systems. The authors present quantitative, mathematical models of a large number of cost factors and use simulation to explain dynamic behaviour such as costs that rise exponentially over time, or costs that fluctuate around a certain level. Their work shows how a ‘hard’, quantitative approach can be used to model one aspect of project management (management of cost).

3 A Systems Thinking perspective on IT project management

Applying Systems Thinking to reason about (a problematic phenomena of) a system involves creating a model of the system. Sterman (2000) proposes a five-step, iterative process to do so. This process consists of the following steps: (i) problem articulation, (ii) formulation of the dynamic hypothesis, (iii) formulation of a simulation model, (iv) testing and (v) policy design and evaluation. These steps are conducted by problem stakeholders, possibly together with a consultant trained in Systems Dynamics. The hypothesis formulated in the second step is a conjecture of how feedback loops in the system explain the behaviour of the system. In quantitative systems dynamics, this hypothesis is then tested (step iv) using quantitative simulation techniques determined in step (iii) to assess the robustness to boundary cases, and to do sensitivity analysis. In qualitative Systems Dynamics, the hypothesis is validated using techniques such as obtaining expert feedback.

In this paper, we focus on what Sterman (2000) calls the dynamics hypothesis. This hypothesis

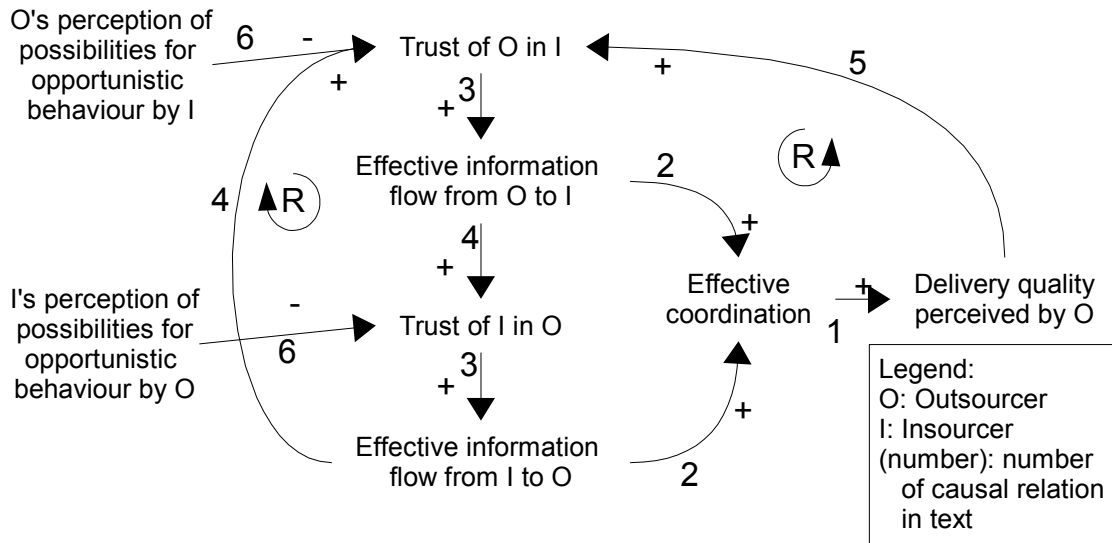


Fig. 1: Operational success of outsourcing: causal model. A '+' indicates that an increase in one variable causes an increase in the other. A '-' indicates that an increase in one variable causes a decrease in the other. Two 'reinforcing' (letter 'R') feedback loops are indicated.

explains the behaviour of the system in terms of interactions between its components and is usually formulated using some diagramming technique such as causal loop diagrams or stock and flow diagrams (Sterman, 2000; Maani and Cavana, 2000). A causal loop diagram (and also a stock and flow diagram, which is an extension of a causal loop diagram) consists of a number of variables and causal relations between them. A variable either describes some property of the system as a whole, or some part of it. We present an example causal loop diagram in Section 4.

The position of this paper is that we propose to apply Systems Thinking to IT project management. This means that we propose to focus on the dynamic behaviour of a project: the behaviour of (everything in) a project changes over time, and IT project management is about controlling this. Project managers do so by identifying feedback loops, creating causal models of the project and choosing control actions based on this model, similar to the process outlined above. We illustrate the kind of insight that this creates in the next section.

4 A causal model of IT outsourcing project success

In this section, we present, as an illustration of the Systems Thinking perspective, a causal model of dynamics in an IT outsourcing project. Our model, which is depicted in Figure 1, is based on the following vision of success of outsourcing projects. Outsourcing creates a relation between two economically independent actors (the outsourcer and the insourcer). The fact that the two actors are economically independent creates forces and tensions between them that in turn influence *delivery quality* as depicted in the model (words in italics refer to variables in Figure 1). During the project, both employ *information flows* to *coordinate* their activities. This information exchange, however, is influenced by the *trust* that they have in each other, which in turn is influenced by the *perception* of the outsourcer of the extent to which the insourcer is capable of *delivery*. As both actors are economically independent, we have to assume that each is self-interested and not unwilling to act in a way that advances its own interest in a way that is detrimental to the other (*possibilities for opportunistic behaviour*).

We claim that in outsourcing, there are at least two positive, or reinforcing, feedback loops

which explains why outsourcing projects have a tendency to get out of control: for instance, if trust erodes, parties become less open in their communication, which affects delivery, which in turn further erodes trust, and so on. Of course, the feedback loop can also work in the opposite direction: if trust increases (for instance, by decreasing opportunism), eventually delivery quality will increase, which in turn increases trust further. The point is that the feedback loops make the *dynamic* aspects of the project explicit. This is the primary benefit of applying the Systems Thinking perspective.

The next sections systematically describe the variables in the model as well as the relations between them. For each variable, we motivate its inclusion in the model. We do so by discussing two norms: the variable should be involved in an important causal relation and (with the exception of the outcome variable, which we discuss next) the variable should represent something that project managers can control in real-world situations.

4.1 Delivery quality: Operational success of IT outsourcing projects

With operational success, we mean that the project results in delivery of a solution, within time and within budget, that according to the customer has the quality agreed upon beforehand. Similar to [van Ekris \(2008\)](#), we view operational success as more than only delivering the right result on time and within budget. Instead, operational success of a project is determined during the entire process of executing the project. We use the term ‘delivery quality’ to refer to the quality of obtaining and delivering the project result over time.

Delivery quality as perceived by the outsourcer is in the model because it is the ultimate goal of at least the outsourcer and most probably also of the insourcer to deliver intermediate and ultimate project results that are perceived as quality results by the outsourcer. The precise operationalisation of delivery quality is case-specific: the relevant stakeholders in a particular case decide how they recognize, or even quantitatively measure, delivery quality. [Thomas and Fernandez \(2008\)](#) report results of a survey among 36 Australian companies with the aim to understand how these companies define project success. They found that project success is a multi-dimensional construct and identify several success criteria in three categories (‘project management’, e.g. on-time and on-budget, ‘technical’, e.g. system quality, and ‘business’, e.g. delivery of benefits). For this paper, we assume that quality of the products and services delivered by the insourcer can be operationalised in a sufficient way.

Delivery quality participates in two causal relations in [Figure 1](#): one with coordination and another one with trust. We discuss these relations in the sections devoted to these two variables below.

4.2 Coordination

We follow [Malone and Crowston \(1994\)](#) and define coordination as “managing dependencies between activities.” In the context of the current illustration, ‘activities’ are the common activities in software development, such as requirements engineering, programming, testing, and delivering the resulting software product. There are dependencies between these activities: certain activities can only start when others have finished, or certain activities may need access to the same resource. Coordination consists of all mechanisms employed and actions taken to ensure that these dependencies do not hamper the project.

[Thompson \(1967\)](#) distinguishes between three types of dependencies: sequential dependences (where one activity can only start after another has finished), pooled dependencies (where several activities consume or produce resources that are pooled), and reciprocal dependencies, where the result of one activity is used by another and the other way around. [Thompson](#) also distinguishes three types of coordination: coordination by plan, which is appropriate for sequential dependence, coordination by standardisation (pooled dependence), and coordination by mutual adjustment (reciprocal dependence).

In software development projects, all three forms of dependence and all three forms of coordination can be observed. The well-known standard software development processes such as Rational Unified Process (RUP) distinguish a number of phases (sequential dependence); applying a process such as RUP in a project is clearly a form of coordination by plan. In many projects, different groups of programmers make different modules in parallel. These modules contribute to a pool of modules that eventually comprises the software product, and all comply with standard interfaces. This is an example of pooled dependence and coordination by standardization. We discuss the third type, reciprocal dependence, in the section on information exchange below.

Causal Relation 1: An increase in effective coordination causes an increase in delivery quality as perceived by the outsourcer.

4.2.1 Research support

Mirani (2007), in a recent paper, examines the relation between what he calls procedural coordination in offshore “software tasks” in two case studies. He describes how in the organizations studied, coordination mechanisms are changed over time in different phases of the offshoring relationships and analyses the outcomes of these changes. He found that in one case study, when it was decided not to use the “global delivery model” (a set of coordination mechanisms) after moving from a pilot project to full-blown offshoring, numerous quality issues arose.

4.2.2 Value

We concur with Kraut and Streeter (1995) and mention that successful coordination is an important determinant of project success. Moreover, coordination is an area that is amendable to managerial control: project managers can choose coordination mechanisms in the project setup phase and can monitor whether coordination is successful when the project is carried out and intervene in a relatively easy way.

4.3 Information exchange

In our model, successful coordination is in turn influenced by the effectiveness of information exchange between the insourcer and the outsourcer. We take the position that reciprocal dependence is the dominant type of dependence in the interaction between outsourcer and insourcer. Every project milestone in which (part of) the final product is delivered is an opportunity for the outsourcer to provide the insourcers with feedback concerning the quality of the product, which in turn enables the insourcer to create a new version, which is another opportunity for the outsourcer to provide feedback, and so on. The appropriate coordination mechanism according to Thompson is mutual adjustment based on information exchanged between the outsourcer and the insourcer. This works better when the information exchanges is of better quality, which justifies the inclusion of the following causal relation in the model:

Causal Relation 2: An increase in the quality of information (regarding the project) exchanged between the insourcer and outsourcer causes an increase in effective coordination.

4.3.1 Research support

Relations between information exchange and coordination are numerous in the scientific research literature. Many observations made by Mirani (2007) boil down to information exchange; for instance, the author observed that coordination improved when the offshoring company started to use a ‘dedicated multi-tier Telecom infrastructure’, Bennett (2004) advises to ‘build a communications plan’ as a way to manage a successful relationship. These two results focus on facilitating direct human-to-human communication. Best-practice frameworks such as CMMI (CMMI Product

Team, 2006) also provide support for this relation: many processes in the CMMI are included to ensure that the right information flows from one stakeholder in a project to another at the right time.

4.3.2 Value

Like in the area of coordination, it is relatively easy for project managers to install control mechanisms that facilitate effective information exchange across the whole spectrum from facilitating direct human-to-human communication to formalized document flows in the project. Managers can invest in modern communication technology to facilitate meetings between stakeholders, increase travel budgets, place employees at each other's sites, or invest in tools such as shared workspaces for document exchange, workflow management systems, and bug trackers.

4.4 Trust

Although it is possible to facilitate information exchange by a vast number of social and technical means, whether the actors in an outsourcing relation actually use these means is influenced by the trust they have in each other. We follow Lewicki *et al.* (1998) and define trust of an actor X in another actor Y as the "confident positive expectations" that X has of the conduct of Y . Mcknight and Chervany (2001) specialize trust for e-commerce; they distinguish between dispositional, institutional, and interpersonal trust. When outsourcing is viewed as a special type of e-commerce, in our model the type of trust that fits best is interpersonal trust. Lewicki *et al.* (1998) characterise high trust in terms of hope, faith and confidence, assurance and initiative; low trust is the absence of this. Next to trust, Lewicki *et al.* (1998) also distinguish distrust. Distrust of an actor X in another actor Y is the "confident negative expectations" that X has of the conduct of Y . High distrust is characterised in terms such as fear, scepticism and cynism, and low distrust is the absence thereof. Trust and distrust are two separate dimensions, rather than opposites in one dimension. In our model, we only use trust; the causal relations turn low trust into high trust or the other way around, but they do not turn trust into distrust, nor the other way around. As we will discuss in the next section, the perception of possibilities for opportunistic behaviour can be interpreted as distrust.

Trust is a dynamic concept: the amount of trust that one actor has in another may change over time. Lewicki and Bunker (1996) distinguish three types of trust (calculus-based, knowledge-based and identification-based trust) and describe the "stepwise evolution of trust" over time. It is thus possible to speak of the increase or decrease of trust, which allows us to include the following causal relations.

Causal Relation 3: An increase in the trust that an actor X has in actor Y causes an increase in the effectiveness of information transmission from X to Y .

Causal Relation 4: An increase of the effectiveness of information transmission from an actor X to an actor Y causes an increase in the trust that Y has in X .

Causal Relation 5: An increase in delivery quality as perceived by the outsourcer causes an increase in the trust that the outsourcer has in the insourcer.

4.4.1 Research support

Levin and Cross (2004) argue that trust plays a mediating role in knowledge transfer. This implies that trust influences the effectiveness of information exchange and the other way around, as there can be no knowledge transfer without effective information exchange. Lewicki and Bunker (1996) discuss several mechanisms that play a role in trust development. These mechanisms also can be viewed as support for the causal relations. For instance, Lewicki and Bunker state that regular (and,

presumably, effective) communication is key in developing knowledge-based trust. Langfield-Smith and Smith (2003) have studied mechanisms used for trust building and control in outsourcing relationships. Their findings include that “outcome controls” (which we view as monitoring or measuring delivery quality) are applied to develop trust. Lander *et al.* (2004) report similar findings. In particular, they state that “The importance of communication as a trust-building mechanism had the highest overall rating as the most important of the ten mechanisms examined. Study participants identified communication as a trust-building mechanism more frequently than any other mechanism.” (Lander *et al.*, 2004, p. 517). Nicolaou and McKnight (2006) investigated trust in what they call inter-organizational systems and found that an increase in perceived quality of information exchange results in an increase in trust.

4.4.2 Value

Trust plays a central role in the causal model; it is the linking pin that connects all other variables. Trust can be manipulated by project managers via various means. As our model indicates, one way is to increase delivery quality. The literature on outsourcing discusses many other ways. Jarvenpaa and Leidner (1998), for instance, discuss how trust can be influenced on the inter-personal level (by making sure that project members meet face-to-face, etc.)

4.5 Perception of possibilities for opportunistic behaviour

The possibility that either, or both, of the actors involved in an outsourcing relation behaves opportunistically follows from Transaction Costs Analysis (TCA), an economic theory proposed by Coase (1937) and extended by Williamson (1975, 1981). Transaction Costs Analysis is rooted in an economic view of the world in which economic actors have bounded rationality¹ (Simon, 1985) and are not inherently ‘good’: economic actors are willing to bend the rules, breach contract, cheat, or engage in any other activity that advances them toward their goal, even if this harms other economic actors. Williamson calls this ‘opportunism’ (Williamson, 1993). Consequently, in TCA it is assumed that a contract never completely specifies a transaction upfront: due to the bounded rationality of the participants in the contract, not all possible future situations are foreseen. Moreover, due to the opportunism, participants in the contract will exploit any incompleteness that is to their advantage if needed. This holds for the whole spectrum of transactions, from explicit, formal contracts between independent companies on the market to implicit, informal contracts between workers and their supervisors within the hierarchy of a single organization.

Traditionally, Transaction Costs Analysis has been used in the context of companies that manufacture physical goods. Recently, a modified transaction costs model (Murray and Kotabe, 1999) has been used to analyse sourcing decisions in service industries. A further specialization is the application of TCA to analyse outsourcing of IT services (Aubert *et al.*, 2004). This (empirical) research shows that factors predicted by the TCA framework, such as uncertainty and the extent to which investments have to be made that are specific for a particular outsourcing deal, indeed explain the level of outsourcing observed in surveys covering large sets of companies.

The extent to which an economic actor has the possibility to act opportunistically differs for each relation that the actor engages in. This influences the amount of trust that one economic actor has in the other. We therefore include in our model a negative causal relation between possibilities for opportunistic behaviour and trust:

Causal Relation 6: An increase in the possibilities for opportunistic behaviour by an actor X as perceived by another actor Y causes a decrease of trust of Y in X .

¹ Bounded rationality means that an economic actor has only limited knowledge and reasoning power at its disposal when assessing a situation and choosing what to do to reach its goals. Therefore, its behaviour is suboptimal from the perspective of full information and unlimited reasoning or computational power.

4.5.1 Research support

This causal relation follows directly from Transaction Cost Analysis: if an economic actor can benefit from exploiting opportunism, he will do so. However, [Granovetter \(1985\)](#) questions this and argues that the fact that an organization is usually embedded in social networks avoids this (extreme) behaviour. We take the middle ground: embeddedness makes gross exploit of opportunistic behaviour unlikely, but we think that in practice, the possibility that one party exploits a small opportunity is large enough that economic actors try to avoid this. One way to do so is to cover this in contracts. An increase (even a small one) in the perceived likelihood that a partner exploits an opportunity (even a small one) then leads to a decrease in trust; this is exactly what the arrow between the two variables in the model represents.

4.5.2 Value

The perception of the possibilities for opportunistic behaviour can be controlled by project managers, for example by making sure that such possibilities are covered by the contract governing the relation. In our model, this is currently the only way to balance the forces of the feedback loops that involve trust.

4.6 Discussion

In this section, we illustrated how the Systems Thinking perspective reveals feedback loops that explain the behaviour of an outsourcing project over time. The variables of the model indicate which control actions a project manager can take, and provide insight in why the project behaves as it does. Thus, the dynamic model provides a means for project managers to reason about their projects as a starting point for decision making.

The causal model presented in this section is not the only possible model that explains the dynamics of an outsourcing project. We are aware of several (almost obvious) possible extensions, which we discuss next. In fact, from the perspective of Systems Thinking methodology, this is of minor importance. Our model is a consolidation of the experience that we gained in studying outsourcing and inter-organizational systems development with several industrial partners. Thus, the model is not specific for one particular IT outsourcing project. When Systems Thinking is applied in a specific project context, completeness of the model is tested for that specific context as part of the process outlined in [Section 3](#). Generalising to other possible context is usually not in the interest of the project stakeholders. Academic researchers, on the other hand, are interested in generalisation; we discuss this in [Section 5.1](#)

Possible extensions of the model include adding more variables. The literature on outsourcing and offshoring provides numerous suggestions, see [Dibbern *et al.* \(2004\)](#) for a comprehensive survey and [Lacity *et al.* \(2008\)](#) and [King and Torkezadeh \(2008\)](#) for contemporary reports on the state of the art. For instance, especially in the case of offshore outsourcing, cultural issues are important factors in the success of an outsourcing relation. Also knowledge is a factor: the outcome of an outsourced IT project is influenced by the knowledge the insourcer has of the project context at the outsourcer. Knowledge is related to two variables that are present in our model: information exchange (as a way to ‘build’ knowledge) and trust ([Levin and Cross, 2004](#)).

Another type of extension is related to our definition of project success. Our model focuses on operational success: a process is successful if the perceived delivery quality is sufficiently high. However, operational success does not necessarily imply unqualified, general success for both partners: it may be the case that an outsourcing project is successful in delivering a solution that turns out to be the wrong one from the perspective of the business strategy of one or both partners. Whether it is possible to extend the model with strategic considerations such as this is a topic for further research.

5 Conclusion

5.1 Implications for research

Systems Thinking has been developed as a problem solving approach and not as a scientific research method. The question then is: Can IT project management from a Systems Thinking perspective result in scientific knowledge? In this section, we discuss two roles that we see for academic research in the area of Systems Thinking and IT project management.

In this first role, Systems Thinking is a tool in the academic study of IT project management. While Systems Thinking is not itself a scientific method, it contains elements that are related to scientific methodology. As has been discussed in our presentation of the modelling process in Section 3, practitioners of Systems Thinking formulate a dynamic hypothesis of the cause of the problem at hand, for instance in the form of a causal loop diagram. They evaluate this hypothesis via simulation, sensitivity analysis, etc., for their specific case. The questions that are relevant for academic research is: Are these models more general than just the specific case from which they originate? How can results obtained in the context of the particular case be generalized?

In our opinion, generalization of results obtained by applying Systems Thinking to a specific IT project management case can follow several different routes. One route is to apply the same causal model to a different case. This is what Lee and Baskerville (2003) call TE-generalization (from Theory to Empirical description): “generalizing a theory, confirmed in one setting, to descriptions of other settings”. This route requires testing the model in the context of this different case as outlined by the process described in Section 3. The model thus becomes a kind of reference model that practitioners use as a starting point in the hypothesis formulation step.

The other route is to regard a causal model that is validated in a particular case as a representation of an empirical fact and derive from this a theory. This type of generalization underpins the case study research methodology of Yin (2003). Lee and Baskerville (2003) call this type of generalization ET (from Empirical description to Theory): generalization from description to theory. This type of generalization requires that the researcher carefully documents how the idiosyncrasies of the particular case are interpreted as instances of more general phenomena (Klein and Myers, 1999). We think that Systems Thinking is particularly useful for this: models such as causal loop diagrams describe the causal mechanism that *explains* the observed empirical fact.

The second role that we propose for academic research takes Systems Thinking itself as the object of study. In this role, academic research studies the application of Systems Thinking to IT project management with the aim to develop knowledge about best practices, and to develop new methods and tools that help practitioners to apply Systems Thinking in this context.

5.2 Implications for practice

In Section 4, we have illustrated how the Systems Thinking perspective on IT project management helps project managers to understand and reason about the causal mechanisms that explain project dynamics. Project managers can use this to decide which interventions to conduct to keep their projects under control. The Systems Thinking perspective extends existing knowledge about IT project management that is of a more static nature, such as knowledge of success factors and risks.

This practical implication applies to the phase in which the project is running and managers have to exercise control by using the control mechanisms that the project provides. We see a second practical implication in the setup phase of the project: the Systems Thinking perspective supports design of project control mechanisms. In the setup phase, stakeholders have to decide on which mechanisms to use to control the project. These mechanisms form a system, which is an artefact that needs to be designed by the project stakeholders. One step in design is identifying solution alternatives and choosing the one(s) that can be expected to reach the goal of the design. Designers base this expectation on knowledge of the behaviour of each solution alternative. The Systems Thinking perspective equips project stakeholders with the tools to obtain this knowledge:

the dynamic models explain how the different control mechanisms interact. Thus, the Systems Thinking perspective on IT project management not only allows project managers to control their projects, but also to design the set of control mechanisms that they employ to exercise control.

Acknowledgments.

We gratefully acknowledge the financial support of the Dutch Jacquard program for the project “QuadREAD”.

References

- Aubert, B. A., Rivard, S. and Patry, M. (2004), “A transaction cost model of IT outsourcing”, *Information & Management*, Vol. 41, pp. 921–932.
- Bennett, A. H. (2004), “Outsorcery: How to create phenomenal outsourcing relationships”, in *STC Proceedings*, Vol. 51, pp. 30–33.
- Checkland, P. (1981), *Systems Thinking, Systems Practice*, John Wiley & Sons Ltd.
- CMMI Product Team (2006), CMMI for development, version 1.2: Improving processes for better products, Technical Report CMU/SEI-2006-TR-008, Software Engineering Institute, Carnegie-Mellon University.
- Coase, R. H. (1937), “The nature of the firm”, *Economica*, Vol. 4 No. 16, pp. 386–405.
- Dangerfield, B. C. (1999), “System dynamics applications to European health care issues”, *The Journal of the Operational Research Society*, Vol. 50 No. 4, pp. 345–353. <http://www.jstor.org/stable/3010452>
- Dibbern, J., Goles, T., Hirschheim, R. and Jayatilaka, B. (2004), “Information systems outsourcing: a survey and analysis of the literature”, *SIGMIS Database*, Vol. 35 No. 4, pp. 6–102. DOI: [10.1145/1035233.1035236](https://doi.org/10.1145/1035233.1035236)
- Gemino, A. C., Reich, B. H. and Sauer, C. (2007), “Beyond chaos: Examining IT project performance”, in *eProceedings of the 2nd International Research Workshop on Information Technology Project Management (IRWITPM)*, pp. 29–38. <http://www.sigitprojmgmt.org/WorkshopProceedings2007.pdf>
- Granovetter, M. (1985), “Economic action and social structure: The problem of embeddedness”, *The American Journal of Sociology*, Vol. 91 No. 3, pp. 481–510. <http://www.jstor.org/stable/2780199>
- Jarvenpaa, S. L. and Leidner, D. E. (1998), “Communication and trust in global virtual teams”, *Journal of Computer-Mediated Communication*, Vol. 3 No. 4. <http://jcmc.indiana.edu/vol3/issue4/jarvenpaa.html>
- Johnstone, D., Huff, S. and Hope, B. (2006), “IT projects: Conflict, governance, and systems thinking”, in *Proc. 39th Annual Hawaii International Conference on System Sciences HICSS '06*, Vol. 8, pp. 197b–197b. DOI: [10.1109/HICSS.2006.235](https://doi.org/10.1109/HICSS.2006.235)
- King, W. R. and Torkezadeh, G. (2008), “Information systems offshoring: Research status and issues”, *MIS Quarterly*, Vol. 32 No. 2, pp. 205–225.
- Klein, H. K. and Myers, M. D. (1999), “A set of principles for conducting and evaluating interpretive field studies in information systems”, *MIS Quarterly*, Vol. 23 No. 1, pp. 67–93.

- Kraut, R. E. and Streeter, L. A. (1995), “Coordination in software development”, *Communications of the ACM*, Vol. 38 No. 3, pp. 69–81. DOI: [10.1145/203330.203345](https://doi.org/10.1145/203330.203345)
- Lacity, M. C., Willcocks, L. P. and Rottman, J. W. (2008), “Global outsourcing of back office services: lessons, trends, and enduring challenges”, *Strategic Outsourcing: An International Journal*, Vol. 1 No. 1, pp. 13–34. DOI: [10.1108/17538290810857457](https://doi.org/10.1108/17538290810857457)
- Lander, M. C., Purvis, R. L., McCray, G. E. and Leigh, W. (2004), “Trust-building mechanisms utilized in outsourced is development projects: a case study”, *Information & Management*, Vol. 41, pp. 509–528. DOI: [10.1016/j.im.2003.10.001](https://doi.org/10.1016/j.im.2003.10.001)
- Langfield-Smith, K. and Smith, D. (2003), “Management control systems and trust in outsourcing relationships”, *Management Accounting Research*, Vol. 14, pp. 281–307. DOI: [10.1016/S1044-5005\(03\)00046-5](https://doi.org/10.1016/S1044-5005(03)00046-5)
- Lee, A. S. and Baskerville, R. L. (2003), “Generalizing generalizability in information systems research”, *Information Systems Research*, Vol. 14 No. 3, pp. 221–243.
- Levin, D. Z. and Cross, R. (2004), “The strength of weak ties you can trust: The mediating role of trust in effective knowledge transfer”, *Management Science*, Vol. 50 No. 11, pp. 1477–1490. <http://search.ebscohost.com/login.aspx?direct=true&db=bsh&AN=15170097&site=ehost-live>
- Lewicki, R. J. and Bunker, B. B. (1996), Developing and maintaining trust in work relationships, in Kramer, R. M. and Tyler, T. R. (Eds.), *Trust in organizations: Frontiers of Theory and Research*, Sage Publications, Thousand Oaks, CA, USA, chapter 7, pp. 114–139.
- Lewicki, R. J., McAllister, D. J. and Bies, R. J. (1998), “Trust and distrust: New relationships and realities.”, *Academy of Management Review*, Vol. 23 No. 3, pp. 438–458. <http://search.ebscohost.com/login.aspx?direct=true&db=bsh&AN=926620&site=ehost-live>
- Maani, K. E. and Cavana, R. Y. (2000), *Systems Thinking and Modelling: Understanding Change and Complexity*, Prentice Hall, Auckland, New Zealand. <http://books.google.com/books?id=IF-YAAAACAAJ>
- Malone, T. W. and Crowston, K. (1994), “The interdisciplinary study of coordination”, *ACM Computing Surveys*, Vol. 26 No. 1, pp. 87–119. DOI: [10.1145/174666.174668](https://doi.org/10.1145/174666.174668)
- Mcknight, D. H. and Chervany, N. L. (2001), “What trust means in e-commerce customer relationships: An interdisciplinary conceptual typology.”, *International Journal of Electronic Commerce*, Vol. 6 No. 2, pp. 35–59. <http://search.ebscohost.com/login.aspx?direct=true&db=bsh&AN=6418522&site=ehost-live>
- Mirani, R. (2007), “Procedural coordination and offshored software tasks: Lessons from two case studies”, *Information & Management*, Vol. 44 No. 2, pp. 216–230. DOI: [10.1016/j.im.2006.12.001](https://doi.org/10.1016/j.im.2006.12.001)
- Murray, J. Y. and Kotabe, M. (1999), “Sourcing strategies of U.S. service companies: A modified transaction-cost analysis”, *Strategic Management Journal*, Vol. 20, pp. 791–809.
- Mutschler, B. and Reichert, M. U. (2008), “On modeling and analyzing cost factors in information systems engineering”, in *Proceedings 20th Int’l Conf. on Advanced Information Systems Engineering (CAiSE’08)*, Montpellier, France, Vol. 5074 of *Lecture Notes in Computer Science*, Springer Verlag, Berlin/Heidelberg, Germany, pp. 510–524. DOI: [10.1007/978-3-540-69534-9_38](https://doi.org/10.1007/978-3-540-69534-9_38)
- Mutschler, B., Reichert, M. U. and Rinderle, S. B. (2007), “Analyzing the dynamic cost factors of process-aware information systems: A model-based approach”, in *Proceedings of the 19th International Conference on Advanced Information Systems Engineering (CAiSE 2007)*, Trondheim,

- Norway, Vol. 4495 of *Lecture Notes in Computer Science*, Springer Verlag, Berlin/Heidelberg, Germany, pp. 589–603. DOI: [10.1007/978-3-540-72988-4_41](https://doi.org/10.1007/978-3-540-72988-4_41)
- Nicolaou, A. I. and McKnight, D. H. (2006), “Perceived information quality in data exchanges: Effects on risk, trust, and intention to use”, *Information Systems Research*, Vol. 17 No. 4, pp. 332–351. DOI: [10.1287/isre.1060.0103](https://doi.org/10.1287/isre.1060.0103)
- Pollack, J. (2007), “The changing paradigms of project management”, *International Journal of Project Management*, Vol. 25 No. 3, pp. 266–274. DOI: [10.1016/j.ijproman.2006.08.002](https://doi.org/10.1016/j.ijproman.2006.08.002)
- Richardson, G. P. (1999), “Reflections for the future of system dynamics”, *The Journal of the Operational Research Society*, Vol. 50 No. 4, pp. 440–449. <http://www.jstor.org/stable/3010465>
- Simon, H. (1985), “Human nature in politics: the dialogue of psychology with political science”, *American Political Science Review*, Vol. 79, pp. 293–304.
- Sterman, J. D. (2000), *Business dynamics: systems thinking and modeling for a complex world*, Irwin/McGraw-Hill. <http://books.google.com/books?id=VzXYAAAACAAJ>
- Swart, J. and Powell, J. H. (2006), “Men and measures: capturing knowledge requirements in firms through qualitative system modelling”, *Journal of the Operational Research Society*, Vol. 57, pp. 10–21. DOI: [10.1057/palgrave.jors.2601983](https://doi.org/10.1057/palgrave.jors.2601983)
- Thomas, G. and Fernandez, W. (2008), “Success in IT projects: A matter of definition?”, *International Journal of Project Management*, Vol. 26 No. 7, pp. 733–742. DOI: [10.1016/j.ijproman.2008.06.003](https://doi.org/10.1016/j.ijproman.2008.06.003)
- Thompson, J. D. (1967), *Organizations in action: social science bases of administrative theory*, The McGraw-Hill Book Company.
- van Ekris, J. (2008), “Customer perception of delivery quality: a necessary area for attention for project managers”, in Pahl, C. (Ed.), *Proceedings of IASTED International Conference on Software Engineering as part of the 26th IASTED International Multi-Conference on Applied Informatics, Innsbruck, Austria*, ACTA Press, pp. 268–275.
- Williamson, O. E. (1975), *Markets and hierarchies, analysis and antitrust implications: a study in the economics of internal organization*, first edn, The Free Press, New York, NY, USA. <http://lccn.loc.gov/74027597>
- Williamson, O. E. (1981), “The economics of organization: The transaction cost approach”, *American Journal of Sociology*, Vol. 87 No. 3, pp. 548–577.
- Williamson, O. E. (1993), “Opportunism and its critics”, *Managerial and Decision Economics*, Vol. 14 No. 2, pp. 97–107.
- Wolstenholme, E. F. (1983), “Modelling national development programmes – an exercise in system description and qualitative analysis using system dynamics”, *The Journal of the Operational Research Society*, Vol. 34 No. 12, pp. 1133–1148. <http://www.jstor.org/stable/2581837>
- Wolstenholme, E. F. (1993), “A case study in community care using systems thinking”, *The Journal of the Operational Research Society*, Vol. 44 No. 9, pp. 925–934. <http://www.jstor.org/stable/2584184>
- Wolstenholme, E. F. and Coyle, R. G. (1983), “The development of system dynamics as a methodology for system description and qualitative analysis”, *The Journal of the Operational Research Society*, Vol. 34 No. 7, pp. 569–581. <http://www.jstor.org/stable/2581770>
- Yin, R. K. (2003), *Case study research: Design and methods*, 3rd edn, Sage Publications, Thousand Oaks, CA, USA. http://books.google.com/books?id=BWea_9ZGQMwC