Dynamics of the Amount of Control in Offshore Software Development Projects

Robert Hartmann
Accenture
Anni-Albers Str. 11
80807 Munich
Germany
+49 175 5766471
robert.x.hartmann@accenture.com

Dr. Martin Wiener
University of Erlangen-Nuremberg
Lange Gasse 20
90403 Nuremberg
Germany
+49 911 5302-859
martin.wiener@wiso.uni-erlangen.de

Dr. Ulrich Remus
University of Canterbury
Private Bag 4800
Christchurch
New Zealand
+64 3 364-2618
ulrich.remus@canterbury.ac.nz

ABSTRACT

This paper investigates changes in the variety and intensity of formal and informal control mechanisms in offshore software development (OSD) projects. Based on a comparative case study approach our results confirm existing findings such as that the amount of control varies across different projects stages, but also contribute with new findings. For example, we found that particularly the quality of project deliverables in early project phases will lead to an increase of the amount of formal control. However, these quality problems do not necessarily lead to an increase of informal control. In return, an increase in quality of deliverables will subsequently decrease the amount of control. An important finding is that in contrast to prior studies our results do not support that the amount of control is directly related to project success. Altogether, our study contributes to the further understanding of the dynamics of the amount of control, its influencing factors and its relationship to project success.

Keywords

Offshore Software Development, Control Theory, Offshoring, Dynamics of Control, Amount of Control

1. INTRODUCTION

Offshoring of tasks or processes is among the most discussed topics in both practitioner and academic literature [1]. In past years, the offshoring industry grew significantly to $ 80 billion in 2008 [2]. It is predicted that this growth will continue in the next five years by a compound annual growth rate of around 6.4% [3], regardless of the economic downturn [2]. Offshoring tasks or entire processes are not new phenomena. Already in the early 90s it became popular with Kodak shifting the operation of its information center to a global provider which was partly located in India [4, 5].

This paper focuses on offshore software development (OSD), which is defined as the design, development and testing of software by a supplying organization located in a foreign, low-cost country. Nowadays, OSD is widely conducted, not only by Fortune 1000 companies. This has several reasons such as cost benefits, flexibility gains [6], increased project management and process quality by OSD providers [7]. While OSD offers a lot of opportunities and benefits, it also poses some specific challenges. Compared to in-house or domestically outsourced projects, offshore software projects are more prone to failure [8]. Apart from traditional software project risks, OSD is exposed to additional risks caused by language and cultural differences [9, 10, 11], geographic distance [9, 12, 13], knowledge transfer difficulties [14, 15, 16], and challenges with regard to control and coordination [17, 18, 19].

One approach for managing risks associated with OSD is the exercise of control. The OSD context imposes several unique challenges in terms of how the client monitors, evaluates, and rewards or sanctions the supplier [20, 21]. Due to differences in cultures and goals, both client and supplier are strongly influenced by opportunistic behavior of the project partner [22]. For instance, on the one hand, the client may be worried about the supplier delivering inadequate software quality and thus may try to increase control. On the other hand, the supplier may fear that the client will change requirements late in the project. Moreover, some clients prefer interpersonal interactions which the supplier may regard as too costly and restrictive. Differences in perspectives like those described may cause the two companies to drag project control in different directions [23]. Furthermore, geographical distance further complicates control. It limits the ability of both parties to meet on a regular basis. As a result, receiving regular feedback and socializing to build up and maintain interpersonal relationships is difficult. To summarize, finding the right balance out of a variety of different control mechanisms and 'customizing' these mechanisms to the specific project context remain among the biggest OSD challenges.

Previous research has mainly focused on the control variety and intensity within portfolios of control – the amount of control [24,

10th International Conference on Wirtschaftsinformatik,
16th - 18th February 2011, Zurich, Switzerland

25]. Prior research has also investigated the dynamics of control in terms of how control changes across different project stages [21, 25]. However, no research has tried to combine these two streams of research, even though this has important implications for managers of OSD projects, trying to exercise the right mix of control mechanisms in each of the project phases.

Thus, this paper tries to address this gap by investigating the changes in the variety and intensity of formal and informal control mechanisms across different phases of OSD projects. In particular, the paper tries to identify important factors that trigger changes in the amount of control in different project phases. Furthermore, these findings are linked to project success, suggesting refining the relationship between amount of control and success variables. Using a comparative case study approach, we propose seven propositions refining the relationship between the amount of control and its influencing factors across different phases of control.

2. THEORETICAL BACKGROUND

2.1 Exercise of Control

In accordance with previous studies, this papers views control in a behavioral sense, which means that the controllees are influenced to act according to objectives and goals set by the controller [26]. This view is based on agency and organization theories and is consistent with prior IS studies [20, 24, 25, 27].

Usually, a control situation involves an individual exercising control (the controller) and an individual being controlled (the controllee) [20]. However, this clear distinction becomes difficult in the OSD context [25]. For instance, the controller and the controllee may not be single individuals but teams of individuals representing their organizational unit or organization respectively. Furthermore, in an OSD project the supplier project manager may be controlled by the client and, in turn, may control the supplier project team members. Often, project managers prevent direct contact to the supplier's project team making it difficult to directly influence their behavior [20].

Category	Mode	Approach
Formal control	*Behavior*	Definition and monitoring of the process to achieve desired outputs
	Outcome	Specification and measurement of outputs (both interim and final)
Informal control	*Clan*	Reliance on the group, or clan, to monitor and control itself
	Self	Reliance on the individuals to monitor and control themselves

Table 1: Control modes

From a behavioral perspective four control modes can be distinguished (see Table 1). These four control modes can be grouped into formal and informal control modes. Formal control can be viewed as a performance evaluation strategy [28] and are split into behavior and outcome control. *Behavior control* is exercised when procedures and rules that are applied are pre-specified and if rewards are based on the extent to which the controlee follows the procedures [26, 28]. *Outcome control* can be exercised when targets or specific outcomes are specified and when the controlee is rewarded for meeting these given goals.

In contrast to formal modes of control, which ignore self-regulating and interpersonal dynamics that influence behavior, informal controls embrace social or people strategies [28, 29]. Informal controls consist of clan and self control. *Clan control* is likened to the cohesive practices of a group and is typified by the degree to which all members of a group are committed to achieving group goals. *Self control* is solely reliant on an individual's ability to monitor and control their own behaviors, with appropriate rewards and sanctions as required.

Various control mechanisms are available – some of them can also be used for exercising more than one control mode. Typically, the controller uses a portfolio of control consisting of several control mechanisms of different control modes [27]. This portfolio is subject to modifications and adoptions throughout the entire project [20, 25], frequently referred to as the dynamics of control [24, 25].

2.2 Amount of Control

This paper adopts Rustagi et al.'s notion of the amount of control. They define the amount of control as "the variety of mechanisms used by a client to exercise control over a vendor and the extent to which each of those mechanisms is used" [1, p. 129]. Consequently, it can be said that the amount of control consists of the two determinants: *variety* and *intensity*.

Most previous research has only focused on the variety of control mechanisms in offshored or outsourced software development [24, 25]. However, particularly little effort was undertaken to measure the variety per control mode as well as the control intensity within a portfolio/mode. Here, clients usually have mechanisms that they primarily rely on, whereas they use other mechanisms as supplement [1]. Although previous studies found empirical evidence that there is a positive relationship between the total amount of control and (project) performance [33, 34], there is no recent research taking into account the more precise definition of Rustagi et al.'s notion of the amount of control.

2.3 Dynamics of Control and Influencing Factors

Dynamics of control describe how the used control mechanisms change during the project. Choudhury and Sabherwal [25] and Heiskanen et al. [30] explored those dynamics of control through the lens of encounters and episodes. They distinguish between stable, long periods—called episodes—and disrupting events between those episodes—called encounters. For instance, in the OSD context, the completion of the requirements specification can be seen as such an encounter ending the episode of requirement determination [25]. Kirsch [24] argues that controllers usually build their starting control portfolio by investigating what formal control mechanisms are available. Then these control mechanisms are evaluated. As a result, suitable mechanisms are kept, inadequate mechanisms dropped, or new appropriate mechanisms added to the initial control portfolio. Subsequent changes in control choices are then triggered by factors in the project, stakeholder, and global contexts.

While there is considerable prior research on the factors that influence the exercise of control in general, there is only very little on the factors influencing the amount of control, in particular across different project phases. Typically, factors that influence the general choice of control can be categorized into controller and controllee [1, 26, 27, 28], project [20, 24, 25], relationship [4, 24, 25], and task characteristics [24, 26, 27, 43]. Factors in these

categories (e.g., project-related knowledge, project performance, project stage, resource availability, task complexity, and task uncertainty) may also impact the amount of control exercised in various project stages to various degrees, but still this has not been sufficiently discussed in literature. So far, current research indicates that the project stage would influence the choice of control mechanisms [20, 25, 31]. Often, in some phases certain mechanisms proof to be inefficient and are therefore removed. Often, during later stages of the project other phase-specific mechanisms are added (such as testing) [25]. Other authors [20, 25, 32] found a relation between project performance, control and certain triggering factors. They note that decreasing project performance, often represented by decreasing deliverable quality, would lead to an increase in control. For instance, Sabherwal and Choudhury [21, 25] state that behavior and clan control are introduced or increased in certain phases if problems occur during the project.

2.4 Project Success

Previous studies found empirical evidence that there is a positive relationship between the total amount of control and (project) performance [33, 34]. In the IS literature, two ways of measuring project success are popular [35]. The first method is to measure the extent to which the initial expectations are met. The second method is to determine the level of overall satisfaction with the offshoring agreement. This paper uses a mixture of both methods considering several success variables, such as delivery in time, project costs, project quality [36] and customer satisfaction with the offshoring agreement [35].

3. RESEARCH METHODOLOGY

In order to answer our research questions, we adopted a comparative case study approach, guided by the process described by Eisenhardt [37]. She draws from both interpretivists and positivists in developing her theory-building process. Similar to Kirsch's study on the dynamics of control [20], our research approach was designed to investigate pre-identified constructs from a positivist view as well as to surface new constructs in an interpretive manner [20]. This hybrid approach can be characterized as "soft positivism" or "scientific realism" [38].

Our rationale for selecting cases was based on purposeful maximum sampling in order to show different perspectives on the issue [44]. Therefore, our cases were heterogeneous in terms of scope, clients, suppliers, and outsourcing locations. In particular, cases were selected in which a significant number of employees were located either offshore or nearshore. Such projects tend to be particularly challenging for the project management, requiring a wide range of control mechanisms. In order to allow for comparison across cases only cases that followed a sequential project phase methodology were selected. This resulted in a selection of nine OSD project cases.

3.1 Cases

A short description of each case including general information about project volume, employees involved, outsourcing location, etc. can be found in the appendix (Table 2) as well as the amount of control in the corresponding project phases (Table 3).

About half of the cases included farshoring to countries in Asia such as India. Two cases examined nearshoring arrangements to Eastern European countries (Poland and Slovakia). The remaining cases covered nearshoring projects in Africa or Western Europe (e.g., Italy and Spain). One quarter of the projects involved captive offshoring, meaning that the supplier is a legal entity of the client. The other cases were about offshore outsourcing, connoting that the project was either outsourced to a multinational service provider with offshoring capabilities or a local vendor in an offshore/nearshore country.

Our study covers various project volumes, ranging from € 20.000 to more than € 100 million. A vast majority of the cases had a total project volume of less than € 5 million. Half of the clients negotiated time and material contracts. One client added a cap limit to the contract to minimize possible additional costs. All other clients had fixed price agreements with their vendors. The offshore team size varied from 4 to over 300. However, around half of the regarded projects involved less than 50 offshore employees. The length of the projects ranged from 4 months for a web-portal development to over 6 years for a SAP implementation involving multiple rollouts in different locations. Five projects had duration of less than 2 years.

3.2 Data Collection

Between February and April 2010, we interviewed 12 project and program managers. The semi-structured interviews followed Myers and Newman's guidelines for qualitative interviewing [39]. Due to the very large size of some projects, work stream leads were interviewed as well. The average interviewee had almost twelve years of IT working experience and around five years of offshore experience. The interviews lasted between one and two hours. Before the interviews, an interview guideline was sent to all participants. This guideline contained definitions of control, the research objectives and a set of sample questions. For simplifying the analysis of the data, the authors introduced a generic three phase project model consisting of the stages requirements determination, system development and system implementation. The interview partners were then asked how control was carried out in each project phase. In order to avoid interview bias, questions and the following discussions were adapted to the interview partner's specific context and role. We also interviewed multiple project team members in each case. This helped in refining and validating the findings. In case of any discrepancy in the statements or findings, they were solved together with the interview partners.

The interview itself was split into three parts. In the first part, general information about the interview partner and the project under study was gathered. The second part consisted of questions regarding the variety of control mechanisms applied and the intensity with which these mechanisms were exercised. For this reason, a non exhaustive list of control mechanisms for each control mode was used in the interviews. Here, the interviewee was asked open questions regarding the mechanisms. Not yet listed mechanisms were added to the list. After this open discussion part, the interviewer asked specifically for the remaining, not mentioned mechanisms in the list. The intensity was determined by asking the interview partner which mechanisms or modes were relatively important compared to others. Sometimes a top list was compiled together with the interviewee. Next, the influencing factors for the identified amount of control were discussed. The third part consisted of simple questions for evaluating the outcomes and the success of the respective offshore software project.

3.3 Data Analysis

For data analysis, the interviews were merged into nine case summaries. These were then checked by the interview partners for

correctness and completeness of the data. In order to avoid bias, other project team members reviewed the case summary in most cases. To achieve a more macro view on each case, we compared across cases. The following steps were carried out:

Step 1: Identification of amount of control per project phase

In order to identify the amount of control per project phase, we performed the following steps for each project and project phase:

First, the mechanisms of each control mode were determined and counted. The intensity of each control mode was estimated by deriving it from the cases (from very low to very high). Second, the variety of mechanisms was translated into a qualitative variable (see scale above). Here, we went through all projects to determine the maximum number of concurrently used mechanisms within one control mode. For instance, the highest number of mechanisms applied for outcome control was 10. Thus, 10 represented the highest achievable variety for this control mode, 0 the lowest. This method was applied for the other control modes as well. Third, by calculating the average between both variables we derived the amount of control from the identified mechanism variety and intensity. In cases where the amount was in between two values, it was rounded up. The results were verified by reviewing the cases again. Finally, the overall amount of control was calculated by using the average of all control modes. The overall amount of control was checked by again comparing the result with the case description.

Step 2: Identification of influencing factors

During the interviews, we asked about changes in control mechanisms and intensities in all project phases. We also asked if there were any concerns regarding too much or too little control exercised by the client. In addition, with the help of the interviewee we identified disrupting events or changing external or internal influences in the projects. This helped identifying influencing factors for the dynamics of control.

This procedure was followed for each of the project phases, enabling us to examine changes across different project phases. The factors influencing the amount of control were determined by analyzing answers to the open questions posed during the interviews. After a couple of interviews, the common drivers for the control amounts emerged and the open questioning was extended by direct questions regarding specific influencing factors.

Step 3: Drawing conclusions

In order to draw conclusions, the findings had to be aggregated on the case level. This was done with the help of overview charts. By examining these overview charts, patterns shared by the projects were identified. In addition, project characteristics and the information gathered during the interviews were used for explaining the control changes during the project. We also identified cross-case patterns, investigated salient features in the case descriptions, and tried to explain possible (statistical) outliers by using available case data. Finally, data on project success was mapped to the amount of control. All findings were subsequently compared with previous research in order to draw final conclusions and explain the findings.

4. AMOUNT OF CONTROL ACROSS PROJECT PHASES

Similar to prior research to map control to project phases a very generic three phase software project approach was used, consisting of requirements determination, the development and the subsequent implementation phase [4]. The macro view on all projects showed that the overall amount of control ranged from low to very high. The breakdown of all amounts can be seen in the appendix (see Table 3). Most projects used a rather high amount of control.

4.1 Requirements Determination Phase

All project except C4 and C11 applied control mechanisms already in the first project phase. The two exceptions had not yet brought offshore employees into their projects due to trainings or lack of need. C10 on the other hand, brought in their offshore team in the middle of the requirements analysis phase.

In particular, C2, C3, C6 and C7 used high amounts of outcome control. In contrast, C8 and C10 applied only medium amounts of control. C8's medium outcome control was the result of the vendor not thoroughly reviewing the results in that phase. C10 brought in the offshore team after half of the phase, when a lot of work had already been conducted. All sequential projects used project plans with deadlines and milestones. Other popular mechanisms included reviewing deliverables, controlling the client's prototypes, preliminary deliverables and functional specifications. This finding supports Choudhury and Sabherwal [25] findings that control portfolios in outsourced software projects are dominated by outcome control, especially in the beginning of the project.

Behavior control was also exercised to a high extent. In four cases the amount of control was high (C2, C3, and C6) to very high (C7). In most of these cases team travelling communication mechanisms were heavily used. In contrast, C8, C10 and C12 had rather low amounts of behavior control in place, mainly because the client was only able to directly influence the vendor management but not the individuals, a phenomenon which typically occur in OSD settings.

The amount of clan control varied significantly in the cases. On the one hand, projects such as C2, C3 and C7 utilized a broad range of clan control mechanisms. Temporary co-location was one of the key factors for these high amounts of clan control. On the other hand, other projects such as C6, C10 and C12 neglected clan control in the first phase. Reasons for this varied: in C6, for example, client and offshore teams were brought together later in the project; in C12 the project managers thought that clan controls were not necessary or even inefficient in this phase; and finally in C10 the vendor tried to hide the project staff behind anonymous services.

The amount of self control ranged from very high (C7) to very low (C10 and C12). In general, self control was the least used control mode. The data did not indicate a link to project size, strategic importance or any other variable, thus suggesting other factors being important predictors of self control. For example data from C7 shows that the high amount of informal control, in particular self control was strongly influenced by experiences gathered from prior OSD projects. In C10 the project manager had bad experiences in using informal control, so the project setup did not plan for these kinds of controls.

4.2 System Development Phase

In the system development phase, all projects utilized the near-/offshore resources for software development tasks. This phase was characterized by an increase of the amount of control.

As in the previous phase, *outcome control* remained the most important control mode. Moreover, in five of nine cases, the amount of outcome control was enhanced compared to the requirements determination phase (e.g. C2, C6, C7, C8 and C10). In the rest of the cases, the amount stayed on the same levels as in the previous phase. This increase is the result of new outcome control mechanisms being added to the existing portfolio. For instance testing, code reviews etc. supplemented existing outcome controls (see section 5). Another reason is that some projects ran into problems requiring an increase in outcome control.

The amount of *behavior control* increased in this phase as well. In C2, C6, C8 and C10 for instance, it increased, whereas it stayed on the same level in the other cases. Team travelling and intense communication were the drivers for this increase. In C4 and C11 exercising behavior control was difficult, because of the vendor trying to inhibit direct influence and monitoring of behavior. However, behavior control was still the second most popular control mode and our data shows a moderate increase of the amount of behavior control. This confirms Choudhury and Sabherwal's [25] findings that behavior control mechanisms are often added later in the project. For instance in C6 intense communication and collaboration also intensified the amount of behavior control, exercised by mechanisms such as imposing procedures guiding programming, documenting and testing.

Clan control increased in three of nine projects (C6, C8 and C10). This was caused by intensified team travelling during system development. In C7 the amount of clan control decreased, because the co-location was abandoned due to cost reasons. In C4 and C11 the offshore sources had just been added to the project. Due to the vendor preventing direct contact, the amount of clan control was rather low in these two projects.

Self control only increased in two projects (and decreased in one project). The overall amount remained low compared to the other control modes. In the other projects it remained untouched.

4.3 System Implementation Phase

In the system implementation, some projects (e.g. C6 and C8) slowly pulled out their offshore teams. No data on this phase was available in C2, because the project had not yet started with system implementation.

In three cases the amount of *outcome control* was reduced (C7, C10 and C11). In all other projects, the amount of control remained on a similar level. This may be explained by the removal of several outcome based controls initialized in the previous phase. For instance, in C11 extensive testing was abandoned, resulting in a lesser amount of control. In C7 on the other hand, the intensity of the existing mechanisms was reduced because of cost reasons.

Behavior control was also reduced (C3, C8 and C11). Again this was due to the fact that certain mechanisms of the previous phase were abandoned. For example in C4 and C9 co-location of project staff was not extensively required anymore.

The amount of *clan control* decreased in four projects (C7, C8, C10 and C11). This can be explained by the offshore sources not being involved to a high extent in the last phase. Socializing and co-location had no further use and thus, they were abandoned for those cases. However, in other projects, the amount stayed on the same level (C3, C4 and C7). The offshore sources were utilized in these phases for (multiple) roll-outs of the software. For this reason, during this phase they were still involved to a higher degree.

Self control was not reduced as strongly as the other control modes. It was solely reduced in C11 due to less team travelling and socializing. This can also be explained by the rather low level of self control in most projects throughout all phases.

5. DISCUSSION – INFLUENCING FACTORS

What are the factors behind the changes in variety and intensity of formal and informal control mechanisms across different phases of OSD projects? This was the second part of the research question posed at the beginning of the paper. The changing amount of control during the project can only partly be explained by the changing needs in each phase [e.g. 20, 40, 31]. In addition, the global context and the influences of involved stakeholders require alterations of the amounts of control during these phases. Further explanations and refinements are consolidated into the following propositions:

Proposition 1: Intensive testing leads to an increase of the amount of formal control

In the system development phase intensive testing took place. In six of nine projects, testing mechanisms, code reviews or the joint specification test cases were added with high intensity to the formal control portfolio (C2, C3, C6, C7, C10 and C11). For instance, in C10 a wide range of testing mechanisms was introduced, such as regular reviews of test plans, actual module and functional tests. These mechanisms were not necessary in the first phase. In C6 an automated testing system was used for assembly and module tests. During the night, this tool tested previously checked-in code. If errors occurred, the responsible person was notified by the system about errors or warnings. Most of these newly introduced mechanisms were dropped in the following phase, because they were not required for the user acceptance or functional system test, some others continued, such as the manual user interface test (C7).

Proposition 2: High communication intensity in the systems development phase leads to an increase of the amount of clan control

Compared to other phases we observed a relatively high amount of clan control in the development phase. In three out of nine projects it increased. A possible explanation might be the high communication intensity in this phase, as evidenced in C6, C7 and C8. For instance, in C6 communication increased as the relationship between the client and vendor teams became closer (see also proposition 4). Another possible explanation is that cultural differences typically emerge in phases with high communication needs, such as the systems development phase. These can be facilitated by the use of informal controls [20].

Proposition 3: Changing team involvement requires changes in the amount of formal and informal control

In some projects, the near-/ offshore teams were introduced later in the project (e.g. C4, C10, and C11). In other projects, the offshore teams were only marginally involved in the system implementation phase (C6 and C8). This changing team involvement required changes in the amount of formal and informal control. For example, in C6 the offshore team was only marginally involved in the system implementation phase. As a result the control mechanisms were significantly reduced. Since

no outcomes were delivered by the offshore team, no outcome control was exercised at all. Communication with the Indian team members took place on demand whenever issues arose and needed clarification. This finding can partly be explained by prior research stating that the choice of particular control mechanisms further depends on the knowledge of the stakeholders as well as the relationship between these stakeholders. Thus, with new team members joining and pulling out of the project, knowledge of controller and controlee and its relationships will change as well [4].

Proposition 4: Trust between client an OSD provider leads to a decrease of the amount of formal control and an increase in the amount of informal control

In particular, with new team members entering the project the relationship between controller and controlee might evolve towards a more trustful relationship [20]. Trust might lead to a decrease of formal controls and an increase in informal controls [20]. This became particularly evident in C11 where the vendor shifted some of his Indian employees onsite for improving communication and increasing productivity. During these team member stays, socializing among offshore and onsite employees was an important trust building mechanism and was facilitated in particular during lunches or private conversations. Altogether trust was built up, which in turn reduced the amount of formal control in this project.

Proposition 5a: Quality problems during the project lead to an increase of the amount of formal control

In total, four of the nine projects ran into quality problems during the project (C2, C3, C8 and C11). These were mainly caused by conflicting perceptions in quality or miscommunication. In some projects deadlines elapsed without delivery of satisfactory results from the vendor. As a result, the client increased the amount of formal controls in all projects but C8. In C8 the client trusted the vendor in solving the quality issues himself. While most clients increased both outcome and behavior control (C2 and C3) one client enhanced solely the amount of outcome control (C11).

This result is not surprising and various authors have argued that project performance problems influence control [25, 20, 41]. Rustagi et al. [1] findings support the claim that task uncertainty which may be caused by erratic quality is positively associated with the total amount of formal control. Moreover, Heiskanen et al. [32] investigated the influence of project performance problems on outcome control. They found that when quality problems arise, clients mitigate this by extending the amount of outcome control. Sabherwal and Choudhury [21, 25] partly support this finding as well. They found out that if problems arise, the client usually introduces more behavior and clan control. Interestingly, our data only supports a positive relationship between project performance problems and behavior control but not between project performance problems and clan control.

Proposition 5b: Quality problems during the project do not necessarily lead to an increase of the amount of informal control

Researchers have discussed the relationship between project performance problems and control [25, 20, 41]. Sabherwal and Choudhury [21, 25] stated that usually clan control is enhanced or introduced when problems occur in a project. This relation is not supported by our data. As described above, four projects ran into quality problems (C2, C3, C8 and C11). However, in our cases the amount of informal control seemed to be unrelated to those emerging quality issues. In C2 and C3 the amount of informal control remained on a similar level. The problems were instead mitigated with the help of formal control. In C8 the amount of control was slightly increased. This was done because the client trusted the vendor to solve the issues himself. C11 showed an increase of the amount of formal control after the problems occurred, while the informal control was minimally decreased. This was the result of the vendor trying to prevent direct governance and control between the client and his offshore sources.

Proposition 5c: Quality problems during the project lead to the introduction of new control mechanisms

As described above, in some phases entirely new mechanisms are introduced, whereas others are abandoned. A good example is the usage of additional deliverable control mechanisms. In a few cases the vendor was required to present prototypes for reviews (C7, C8 and C12). Other examples of this practice include C2, where additional new outcome control mechanisms were added, such as preliminary deliverables, which were then pre-checked by the client. As a result, due to poor quality of early deliverables the intensity of outcome control increased significantly.

Proposition 6: Good quality of deliverables reduces the amount of control

In a few cases (C3, C11) the quality of deliverables increased, after having tackled the quality problems encountered in prior phases. In general we found that the client reduced the amount of control after the project was running smoothly. One possible explanation for this is that the introduction of high amounts of control will also increase the costs associated with these controls. As soon as the project requires less attention, control is reduced for saving costs. This relation could also be explained by an increase of trust between client and vendor after the project was running smoothly [1].

Proposition 7: The amount of control is not directly related to project success

Finally, we made a more general observation, not necessarily linked to the changes of the amount of control across project phases – the link between the amount of control and project success.

Project success was determined according to the method described in section 2.4. The data indicates that there is no relation between project success and control. For instance, project success in highly controlled projects such as C1 and C10 was considered to be low by the project managers. This finding contrasts previous research [18, 33, 34, 42] that found a positive relation between control and project success.

A possible explanation might be that projects with quality or scope problems tend to exercise more formal control (see proposition 5). If the project management cannot solve the quality issues, the overall amount of control may remain on a high level. Thus, despite a high amount of control, the desired project success might not be achieved. Another interviewed manager (C9) stated that by monitoring the vendor too closely, possible problems are uncovered that would have remained undetected with low amounts of control. These detected problems are usually solved by the vendor before the deadlines and thus, they do not affect project success that much. This rather surprising result should be investigated further.

6. CONCLUSION AND OUTLOOK

While prior research has investigated antecedents of control in general, as well as the dynamics of control across project phases [1, 21, 24, 25], there has been only very little research on the dynamic changes in the variety and intensity of formal and informal control mechanisms across different stages of OSD projects. Thus, the unique contributions of this paper are (1) to have provided further empirical work on the still neglected area of 'amount of control', (2) to have taken on a dynamic perspective, by investigating the amount of control across different project phases, (3) to have identified important factors that trigger changes in the amount of control in these phases, and finally (4) to have challenged common assumptions that the amount of control is directly related to project success [18, 33, 34, 42]. In particular the latter is an important finding and suggests that this direct relationship needs to be revisited as there might be mediation or moderation effects responsible for these results.

The first most obvious finding confirms prior research that the amount of control varies as the project progresses [20. 25]. As such, the amount of control is strongly influenced by the project phase [e.g. 29, 40, 31]. Furthermore, our results provide a clearer picture on control modes being used, their intensity in each phase and how they change across phases. For example, in both the systems determination phase and the system development phase in seven out of nine projects the amount of formal control increased (C2, C6, C7, C8, and C10) or stayed on the same level (C3, C12). Our data also shows a moderate increase of the amount of behavior control, thus confirming Choudhury and Sabherwal's [21, 25] findings that behavior control mechanisms are often added later in the project.

Additionally, we could confirm that quality problems during the project lead to an increase of formal amount of control [1]. However, these problems do not necessarily lead to an increase of informal amount of control, in particular clan control [21, 25]. In return, an increase in the quality of deliverables will decrease the amount of control in subsequent project phases. In general it was interesting to see that in all projects self-control played a minor role. This certainly warrants further attention and calls for further in-depth studies to explore possible explanations for this finding. Furthermore, we found that intensive testing will lead to an increase of the amount of formal control, whereas trust will lead to a decrease of the amount of formal control. We also found that changing team involvement required changes in the amount of formal and informal control. So far, this finding has not been acknowledged in prior literature.

There are, of course, limitations to our study that may provide interesting paths for future explorations. First, we made some trade-offs in regard to measuring the amount of control. Here, the variety of control mechanisms was determined in relation to the maximal number of mode-specific mechanisms in all examined projects whereas the intensity of each mechanism was determined by relating this control mechanism to all other mechanisms used in a particular project. Second, this research made no distinction between far- and nearshoring within the same OSD project. In all three corresponding cases (C3, C6, and C11), far- and nearshore teams were using different control modes. Therefore, it might be interesting to further explore these differences with regard to control choices and amounts within the specific sub-projects. Finally, our data was drawn from a comparative case study approach resulting in a number of propositions. The next logical step would be to further refine some of these propositions and test these with the help of a large-scale quantitative study.

Altogether, this study contributes to the further understanding of the complex interplay between the amount of control, its influencing factors and its relationship to project success.

7. REFERENCES

[1] Rustagi, S.; King, W. R.; Kirsch, L. J. 2008. Predictors of Formal Control Usage in IT Outsourcing Partnerships. *Information Systems Research*, 19 (2), pp. 126-143.

[2] McKinsey 2009. Strengthening India's offshoring industry. *McKinsey on Business Technology*, Number 16 Summer 2009, pp. 19-22.

[3] Datamonitor 2009. *Industry Profile: Global IT Services.* Date April 2009, pp. 1-24.

[4] Rao, M. T. 2004. Key issues for global IT sourcing: Country and individual factors. *Information Systems Management*, 21 (3), pp. 16-21.

[5] Rajkumar, T. M.; Mani, R. V. S. 2001. Offshore software development. *Information Systems Management*, 18 (2), pp. 63-74.

[6] Amoribieta, I.; Bhaumik, K.; Kanakamedala, K.; Parkhe, A. D. 2001. Programmers abroad: A primer on offshore software development. *McKinsey Quarterly* (2), pp. 128-139.

[7] Herath, T.; Kishore, R. 2009. Offshore Outsourcing: Risks, Challenges, and Potential Solutions. *Information Systems Management*, 26 (4), pp. 312-326.

[8] Iacovou, C. L.; Nakatsu, R. 2008. A risk profile of offshore-outsourced development projects. *Communications of the ACM*, 51 (6), pp. 89-94.

[9] Dibbern, J.; Winkler, J.; Heinzl, A. 2008. Explaining variations in client extra costs between software projects offshored to India. *MIS Quarterly*, 32 (2), pp. 333-366.

[10] Hofstede, G. 2001. *Culture's consequences*, 2nd edition, SAGE Publications, Thousand Oaks, CA, USA.

[11] Krishna, S.; Sahay, S.; Walsham, G. 2004. Managing cross-cultural issues in global software outsourcing. *Communications of the ACM*, 47 (4), pp. 62-66.

[12] Carmel, E.; Agarwal, R. 2001. Tactical Approaches for Alleviating Distance in Global Software Development. *IEEE Software*, 18 (2), pp. 22-29.

[13] Carmel, Erran; Abbott, Pamela 2007. Why 'nearshore' means that distance matters. *Communications of the ACM*, Vol. 50 Issue 10, p40-46.

[14] Beulen, E.; Ribbers, P. 2007. Control in Outsourcing Relationships: Governance in Action. *HICSS 2007 Proceedings*, pp. 236b.

[15] Heeks, R.; Nicholson, B. 2001. Synching or Sinking: Global Software Outsourcing Relationships. *IEEE Software*, 18 (2), pp. 54-61.

[16] Sahay, S., Nicholson, B., Krishna, S. 2003. *Global It Outsourcing: Software Development across Borders*, Cambridge University Press, Cambridge, UK.

[17] Erickson, J. M; Ranganathan, C. 2006. Project Management Capabilities: Key to Application Development Offshore Outsourcing. *HICSS 2006 Proceedings*, Vol. 8, pp. 199b.

[18] Narayanaswamy, R.; Henry, R. M. 2005. Effects of culture on control mechanisms in offshore outsourced IT projects. *Proceedings of the 2005 ACM SIGMIS CPR conference*, pp. 139-145.

[19] Heeks, R.; Nicholson, B. (2001): Synching or Sinking: Global Software Outsourcing Relationships. *IEEE Software*, Vol. 18 Issue 2, p54-61.

[20] Kirsch, L. J. 2004. Deploying Common Systems Globally: *The Dynamics of Control. Information Systems Research*, 15 (4), pp. 374-395.

[21] Sabherwal, R.; Choudhury, V. 2009. Governance of remotely outsourced software development: A comparison of client and vendor perspectives. In R. Hirschheim, A. Heinzl, & J. Dibbern (Eds.) Information systems outsourcing: Enduring themes, new perspectives, and global challenges (pp. 187–222), Springer, Berlin, Germany.

[22]]Nicholson, B.; Jones, J.; Espenlaub, S. 2006. Transaction costs and control of outsourced accounting: Case evidence from India. *Management Accounting Research*, 17 (3), pp. 238-258.

[23] Heiskanen, A.; Newman, M.; Simila, J. 1996. Software Contracting: A Process Modeling Approach. *ICIS 1996 Proceedings*, Paper 3.

[24] Kirsch, L. J. 1997. Portfolios of Control Modes and IS Project Management. *Information Systems Research*, 8 (3), pp. 215-240.

[25] Choudhury, V.; Sabherwal, R. 2003. Portfolios of Control in Outsourced Software Development Projects. *Information Systems Research*, 14 (3), pp. 291-314.

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

[27] Kirsch, L. J.; Sambamurthy, V.; Dong-Gil K.; Purvis, R. L. 2002. Controlling Information Systems Development Projects: The View from the Client. *Management Science*, 48 (4), pp. 484-498.

[28] Eisenhardt, K. M. 1985. Control: Organizational and economic approaches. *Management Science*, 31 (2), pp. 134-149.

[29] Jaworski, B. J. 1988. Toward a Theory of Marketing Control: Environmental Context, Control Types, and Consequences. *Journal of Marketing*, 52 (3), pp. 23-39.

[30] Heiskanen, A., Newman M., Simila J. 1996. Software contracting: A process modeling approach. *ICIS 1996 Proceedings.* Paper 3.

[31] Xu, B.; Yan, X.; Lin, Z. 2005. Control in Open Source Software Development, *AMCIS 2005 Proceedings*. Paper 433.

[32] Heiskanen A.; Newman M.; Eklin, M. 2008. Control, trust, power, and the dynamics of information system outsourcing relationships: A process study of contractual software development. *The Journal of Strategic Information Systems*, 17 (4), pp. 268-286.

[33] Tannenbaum, A., S. 1962. Control in Organizations: Individual Adjustment and Organizational Performance. *Administrative Science Quarterly*, 7 (2), pp. 236-257.

[34] Henderson, J. C.; Soonchul, L. 1992. Managing I/S design teams: a control theories perspective. *Management Science*, 38 (6), pp. 757-777.

[35] Dibbern, J., Goles, T., Hirschheim, R., Jayatilaka, B. 2004. Information systems outsourcing: A survey and analysis of the literature. *The Data Base for Advances in Information Systems*, Vol. 35 Issue 4, 6–102.

[36] Fabriek, M.; Brand, M.; Brinkkemper. S.; Harmsen, F. Helms, R. 2008. Reasons for Success and Failure in Offshore Software Development Projects. *In 16th European Conference on Information Systems* (Golden W, Acton T, Conboy K, van der Heijden H, Tuunainen VK eds.), p446-457.

[37] Eisenhardt, K. M. 1989. Building Theories from Case Study Research. *The Academy of Management Review*, 14 (4), pp. 532-550.

[38] Madill, A.; Jordan, A.; Shirley, C. 2000. Objectivity and reliability in qualitative analysis: Realist, contextualist and radical constructionist epistemologies. *British Journal of Psychology*, 91 (1), pp. 1-20.

[39] Myers, M. D.; Newman M. 2007. The qualitative interview in IS research: Examining the craft. *Information and Organization*, 17 (1), pp. 2-26.

[40] Prifling, M.; Gregory, R.; Beck, R. 2009. Changing Psychological Contracts and their Effect on Control Modes in IT Offshore Outsourcing Projects - A Case from the Financial Services Industry. *HICSS 2009 Proceedings*, pp. 1-10.

[41] Prifling, M.; Gregory, R.; Beck, R. 2008. Project Management Techniques for Managing Cross-Cultural Differences in IT Offshore Outsourcing. *AMCIS 2008 Proceedings*, Paper 365.

[42] Henderson, J. C.; Soonchul, L. 1992. Managing I/S design teams: a control theories perspective. *Management Science*, Vol. 38 Issue 6, p757-777.

[43] Zhang, M.; Chand, D.; David, G. 2007. Exploring Control Modes in Globally Distributed IT Work Management. *AMCIS 2007 Proceedings*.

[44] Creswell, J. 2007. *Qualitative enquiry and research design*. Thousand Oaks, California: Sage Publications, Inc.

8. APPENDIX

Table 2. Case overview

		Case 2	Case 3	Case 4	Case 6	Case 7
Client	**Industry**	Apparel	Telecommunication	High Tech	Healthcare	Software Engineering
	Size	large	large	large	large	Small
	Headquarter	Germany	UK	Germany	Germany	Germany
Project Characteristics	**Description**	Re-development of e-commerce platform with enhancements in functionality.	Global SAP rollout project with a high degree of custom developments.	B2B enterprise application integration, migration and implementation project.	Software development for medical imaging software.	Development of an ajax, web-based system for travel businesses.
	Delivering Organization	Vendor: Indian based, multinational service company	Vendor: Multinational service provider	Vendor: Multinational service provider	Captive: Subsidiary in India	Captive: Subsidiary in Poland
	Cost	< 5 Mio. Euro	> 20 Mio. Euro	< 5 Mio. Euro	> 5 Mio. Euro	< 5 Mio. Euro
	Delivery Location	Offshore: India | Onsite: Germany	Offshore: India | Nearshore: Southern Europe	Nearshore: Slovakia	Offshore: India Nearshore: Eastern Europe	Nearshore: Poland
	Project Employees	around 75 (Vendor: 50 - 60 offshore | Client: 15 - 20)	600 (200 offshore, 100 nearshore)	15 – 30 (onsite: 3 – 8 | nearshore: 12 – 22)	Over 300 (around 2/3 offshore or nearshore)	around 30 (7 nearshore, 23 onsite)
	Duration	8 months	6 years	3 years	> 3 years	1 year
	Contract	Fix Price	Time & Material	Fix Price	Time & Material	Time & Material
	Problems	Quality	Quality	None	None	None
	Success Variables	**Expectations met:** negative **Costs:** rather positive **Timely Delivery:** positive **Satisfaction:** neutral	**Expectations met:** positive **Costs:** rather positive **Timely Delivery:** rather positive **Satisfaction:** neutral	**Expectations met:** rather positive **Costs:** positive **Timely Delivery:** positive **Satisfaction:** rather positive	**Expectations met:** neutral **Costs:** positive **Timely Delivery:** neutral **Satisfaction:** rather positive	**Expectations met:** rather positive **Costs:** negative **Timely Delivery:** neutral **Satisfaction:** neutral
	Success	neutral	rather positive	positive	rather positive	Neutral

		Case 8	Case 10	Case 11	Case 12
Client	**Industry**	Chemical	Telecommunication	Measurement Instruments	Advertising
	Size	medium	large	large	small
	Headquarter	Germany	Spain	Switzerland	Dubai
Project Characteristics	**Description**	Development of a customized solution for production control in the chemical industry.	Custom development and integration of billing related systems	SAP implementation project (most modules) with custom developments.	Arabic web portal development for commercial advertisements.
	Delivering Organization	Vendor: Indian company with onsite management team	Vendor: Multinational service provider	Vendor: Multinational service provider	Vendor: Small, Tunisian offshore provider with
	Cost	< 1 Mio. Euro	> 20 Mio. Euro	> 20 Mio. Euro	< 0.5 Mio. Euro
	Delivery Location	Offshore: India	Nearshore: Spain	Offshore: India Nearshore: Italy, Spain	Nearshore: Tunisia
	Project Employees	30 (+ 4 project management + 4 employees from Client)	Peak > 350 (Vendor: 110 onsite, > 100 nearshore)	185 (Vendor: 100 onsite/ Nearshore, 25 offshore)	4,5 (Vendor: 3,5 offshore developers)
	Duration	15 months	3 years	4,5 years	4 months
	Contract	Fix Price	Fix Price	Time & Material (with cap)	Fix Price
	Problems	Quality	Time	Quality	None
	Success Variables	**Expectations met:** neutral **Costs:** rather negative **Timely Delivery:** rather negative **Satisfaction:** neutral	**Expectations met:** rather negative **Costs:** negative **Timely Delivery:** negative **Satisfaction:** rather negative	**Expectations met:** rather positive **Costs:** rather positive **Timely Delivery:** rather positive **Satisfaction:** positive	**Expectations met:** rather positive **Costs:** neutral **Timely Delivery:** neutral **Satisfaction:** positive
	Success	neutral	rather negative	rather positive	rather positive

Table 3. Overview of the amounts of control across project stages

Project Phases	Case / Control Amounts	C2 Outcome	C2 Behavior	C2 Clan	C2 Self	C3 Outcome	C3 Behavior	C3 Clan	C3 Self	C4 Outcome	C4 Behavior	C4 Clan	C4 Self	C6 Outcome	C6 Behavior	C6 Clan	C6 Self	C7 Outcome	C7 Behavior	C7 Clan	C7 Self
Requirements Determination	Amount	◕	◕	◕	◔	●	◕	●	◕	○	○	○	○	◕	◕	◔	◑	◕	●	◕	●
System Development	Amount	●	●	◕	◔	●	◕	●	◕	●	◑	◔	◔	●	●	◑	◕	●	●	◑	◕
System Implementation	Amount	n/a	n/a	n/a	n/a	●	◑	●	◕	●	◑	◔	◔	○	○	○	○	◕	◕	◑	◕
Overall Control	Amount	◕				●				◑				◕				◕			

Project Phases	Case / Control Amounts	C8 Outcome	C8 Behavior	C8 Clan	C8 Self	C10 Outcome	C10 Behavior	C10 Clan	C10 Self	C11 Outcome	C11 Behavior	C11 Clan	C11 Self	C12 Outcome	C12 Behavior	C12 Clan	C12 Self
Requirements Determination	Amount	◑	◔	◔	◔	◑	◔	○	○	○	○	○	○	◕	◔	○	○
System Development	Amount	◕	◑	◑	◔	●	●	◕	◑	◕	◑	◔	◔	◕	◔	○	○
System Implementation	Amount	○	◔	◔	◔	●	●	◑	◑	◑	◔	○	○	◑	◔	○	○
Overall Control	Amount	◔				◕				◑				◔			