Controlling risk prior to offshore application development

Cor-Jan Jager, Stefan Vos, Michiel Borgers, Frank Harmsen, Sjaak Brinkkemper and Lidwien van de Wijngaert

Department of Information and Computing Sciences
Utrecht University

Technical Report UU-CS-2008-009
www.cs.uu.nl
ISSN: 0924-3275

Controlling risk prior to offshore application development

Jager, C.J., Vos, S., Borgers, M., Harmsen, F., Brinkkemper, S., & Wijngaert, L.

Abstract

This master thesis investigates operational risk occurrence in offshore (custom) application development projects and derives specific risk mitigation actions from them. This information can be integrated in an offshore readiness assessment aimed at assessing an IT organization for readiness in offshoring their application development. Risk described in literature was validated in expert interviews and related to project success in an online survey. From a scientific perspective a delicate extension of specific risk impacts is given to the undifferentiated manner in which risks are often described in scientific literature. An outsourcing IT organization or consultancy firm can use the results (to advice) on risk prioritization prior to offshoring application development.

*A focused literature study was conducted to derive risk (factors) and a risk categorization framework. A distinction was made between basic, residual, controllable and incontrollable risk variables leading to a total of 81 risk variables distributed among categories and integrated in an online survey. This survey was sent to project managers in offshore application development project from which 44 were returned. In order to derive a more comprehensible and manageable set of risk variables, PCA factor analyses were performed for the different risk variable categories and residual factor loadings were used as input for a linear regression analysis to explain project success. A loose translation of risk impact (correlation) * occurrence was made to derive weighing factors for an offshore readiness study.*

Results show that risks variables from all risk categories were meaningful to explain project success. Risks from a collaborative origin have a bigger impact on project success than application complexity risks. Most effective (significant) mitigation actions can be found in the provision of (contextual) documentation, sophistication and use of communication / application engineering tools, compliance to Project management method and compliance to CMMI process standards, but more delicately: different risks require different mitigation strategies. Derived weighing factors indicate that Knowledge management, Application complexity and Methods/tools should get the highest weights in the assessment. The practical feasibility of risk prioritization should be combined with an order of prosecution risk mitigation actions and a financial picture behind these actions.

Some limitations of this research are the relatively small sample size and the over representation of Indian-Dutch project managers. Further research could provide more in-depth analyses of risk variables. Another interesting follow-up research would be to compare risk between onshore and offshore projects.

Preface: Bills at the top

"Considering the current sad state of our computer programs, software development is clearly still a black art, and cannot yet be called an engineering discipline."
(Bill Clinton)

"If you rely too much on the people in other countries and other companies, in a sense that's your brain, and you are outsourcing your brain."
(Bill Gates)

Two quotes from the past. Although the content seems to be quite diverse and both Bills speak from a profoundly different background, their messages contain some sort of a warning, while the subjects in the two quotes pair up to constitute the main topic of this master thesis. **Software development** being one half of the topic addressed and **outsourcing,** or more specifically in context, **offshore** outsourcing, the other half.

The message in the quotes, provide an initial trigger towards the message that we intend to address in this work. Software development is clearly not an easy process. The statement in the first quote is an expression of this difficulty and although there may be some exaggeration in it, the core message seems valid.

Bill Gates warns for overconfidence in outsourcing and illustrates that organizations should be conscious about what and how they outsource and not rely *too* much on the (expertise of) people in other countries and companies.

These messages combined, provide in a nutshell the main topic of my master (Business Informatics) thesis research from Utrecht University: This thesis is all about offshore outsourcing (custom) software development and the risks involved in doing so. It is mainly concerned with risk identification and mitigation that can be done *prior* to the actual offshore outsourcing process from an outsourcing organization (clients) point of view. It is also important to notice that the perspective chosen in this thesis is that of Western (mostly North-American and Western European) society as point of departure.

The metaphorical picture on the front page of this work resembles the main viewpoint of this research. The ship (**the outsourcing organization**) is still in the docks and did not go offshore yet. It is even still under construction in the dock of origin (onshore) where it is being made offshore ready to withstand the rough seas (**risks**) in its offshore journey.

1. Laying the offshore outsourcing foundation

In this chapter, the research subject will be introduced briefly and we will discuss the context around the research and the triggers for research. At the end of the chapter, the motive for this research, its relevance both practical as scientific and the different steps in the research should be clear for the reader. The following paragraphs will be distinguished in this introductory chapter.

- **Subject introduction**: provides an introduction into the main research topic.
- **Research context**: deals with the major stakeholders in our research and the reason why they would be interested in the research topic.
- **Research questions**: Introduces the research questions we will attempt to answer.
- **Justification of research**: This paragraph deals with what will be added to current research with the defined research questions and what the societal value will be.
- **Overview of thesis**: provides a brief overview of what topics will be addressed in the following chapters to give an overview of how our story is build up.

1.1 Subject introduction: offshore outsourcing

There is quite some delicacy involved in talking about offshore outsourcing in general, but for this moment it is suffice to say that offshore outsourcing is bringing parts of your (IT) organization to (an) external vendor(s) located in another (farshore) country. A more thorough analysis of delicacies involved in talking about offshore outsourcing is given in chapter three.

Organizations are increasingly occupied with the (strategic) evaluation of offshore outsourcing options (Willcocks and Lacity, 2006). After the burst of the dotcom bubble and economic recession that followed, competition among organizations grew and along with it, a fast evolvement of the need for cost-cutting strategies like offshore outsourcing emerged (Lewin and Peeters, 2006). At the same time, several investments in education and infrastructure in low-wage countries like India were made rapidly and intensively (Khan, Currie, Weerakkody and Desai, 2003) and the amount of software experts from India increased sixteen fold from 1995 to 2002 (Layman, Williams, Damian and Bures, 2006). Offshore outsourcing in general is expected to grow at double digit growth rates (Beulen, Fenema and Currie, 2005). At the moment there are over 10.000 vendors in more than 175 countries that claim to offer some type of offshore outsourcing (Tsotra and Fitzgerald, 2007) from which India is by far the biggest player with about 70% market share in the whole offshoring market (Lewin et al., 2006) These figures speak for themselves, but an important remaining question lies in the rationale behind them. Relating back to the second quote from the preface, the question is why organizations decide to outsource parts of their (IT) services to an external vendor, since part of their organizational “brain” and the control over it, is lost. Why would organizations engage in this offshore outsourcing adventure at all? What are their expectations and are these expectations actually fulfilled?

1.1.1 Expectations

There are a number of expectations that organizations have with (offshore) outsourcing that makes up the actual trigger/reason for outsourcing parts of their organization. In this overview from literature the *most widely* addressed and *most important* expectations are described.

Cost reduction

As mentioned above, there was a specific need for cost cutting strategies in organizations after the dotcom bubble, and cost reduction is until date the most important (Capgemini, 2006; Young and Potter, 2006; Lewin et al., 2006) and most widely cited promise that organizations report as reason for outsourcing parts of their IT function (Bhattacharya, Behare and Gundersen, 2003; Khan et al., 2003; Dibbern, Winkler and Heinzl, 2006; Carmel, 1999; Carmel and Agarwal, 2002; Herbsleb and Grinter, 1999; Aspray, Mayadas and Vardi, 2006; Conchuir, Holmstrom, Agerfalk and Fitzgerald, 2006; Carmel and Tjia, 2005; Willcocks et al., 2006). It is believed that offshore vendors can provide services at lower costs due to economies of scale and better access to *lower cost* labor pools (Bhattacharya et al., 2003; Erber and Ahmed, 2005).

Enabling focus on core competencies
A second expectation that organizations have with offshore outsourcing is to be able to remain focused on their own core competencies (Khan et al., 2003; Bhattacharya et al., 2003; Capgemini, 2006; Prikladnicki, Audy and Evaristo, 2003; Carmel et al., 2005; Amberg, Schröder and Wiener, 2005; Yang and Huang, 2000; Kuni and Bhushan, 2006; Dibbern, Goles, Hirschheim and Jayatilaka, 2004). In increasingly competitive environments, outsourcing non critical or non core processes can unlock necessary internal resources (Erber et al., 2005; Carmel et al, 2005).

Access to IT skills
Another assumed benefit from offshore outsourcing is the access itself to skilled IT labor forces, not available or very expensive in the country of the outsourcing organization (Khan et al., 2003; Bhattacharya et al., 2003; Carmel et al., 2002; Carmel et al., 2005; Aspray et al., 2006; Conchuir et al., 2006; Damian and Moitra, 2006). Due to heavy education and infrastructure investments in IT developing countries like India and China (Khan et al., 2003; Erber et al., 2005), there is a large labor pool of skilled workers available in these countries (Bhattacharya et al., 2003; Erickson and Ranganathan, 2006).

Higher quality of services
Firms in developing countries like India and China agree almost without exception to international quality standard like CMMI (level 4|5) and ISO9000 (Dibbern, et al., 2006; Erber and Ahmed, 2005) which indicates their IT process maturity and quality management maturity respectively. The average quality level based upon such standards is much lower for most outsourcing organizations from Western countries. Improved quality of service is hence another important offshore outsourcing expectation (Capgemini, 2006; Khan et al., 2003; Aspray et al., 2006; Bhattacharya et al., 2003; Balaji and Ahuja, 2005; Chandrasekaran and Ensing, 2004).

Reduction of time to market
The expectation of round the clock or follow the sun service; thereby reducing time to market for newly developed systems is addressed by a number of authors (Damian et al., 2006; Herbsleb et al., 1999; Khan et al., 2003; Conchuir et al., 2006; Chandrasekaran et al., 2004; Carmel et al., 2005; Kuni et al., 2006; Casey and Richardson, 2006; Shami, Bos, Wright, Hoch, Kuan, Olsen and Olsen, 2004). The premise behind round the clock development is that distributed project teams can work across time zones on the same project within the standard working hours of their respective time zone. The bigger the time zone difference, the larger the potential benefit of the project time efficiency and, thereby time to market, that can be achieved. This expectation can solely be contributed to the *offshore* outsourcing variant with large time-differences between outsourcing IT organization and vendor organization.

1.1.2 Fulfillment of expectations

If all of the above triggers would indeed result in the objective pursued by the outsourcing organizations, there would not be an interesting research project. Unfortunately (or fortunately), the world is not that flat yet (Friedman, 2005).

Cost reduction
The pursued cost efficiency of (offshore) outsourcing is certainly not always achieved (Levina and Ross, 2003; Kuni et al., 2006; Aron and Singh, 2005). Aron et al. (2005) even report that half of the organizations that shifted processes offshore failed to generate the financial benefits they expected. These benefits are under a lot of pressure due to overhead in communication, control and coordination costs as a result of offshore collaboration (Conchuir et al., 2006). In some cases hidden costs like consultancy fees can also play a role in not realizing the intended cost reduction (Khan et al., 2003).

Enabling focus on core competencies
A problem with the expectation of being able to focus more on own core competencies is that it seems to be difficult for companies to distinguish core, critical and commodity processes ready for offshore outsourcing (Aron et al., 2005). Moreover freeing resources to focus on core competences by outsourcing has not always led to the desired result. Dibbern et al. (2004) address the risk of losing control over a strategic asset as Information Systems. When business requirements desire more innovative power from IT to remain competitive at the core business, an organization could be in trouble having outsourced all of their IT expertise. This could indicate that they have to invest heavily in rebuilding or even insourcing back their internal IS capability (Hirschheim and Lacity, 2000).

Access to IT skills
Conchuir et al. (2006) mention that the access to IT skills benefit is valid, but it could well be that when the increasing offshore outsourcing market is growing harder than IT skills investments in IT developing countries, this access will become pressured. From another viewpoint, Conchuir et al. (2006) notice a backside of these IT skills which is reflected in the socio cultural problems in collaboration that can emerge. Kshetri (2007) reasons along similar lines and states that outsourcing readiness of a country’s workforce is not only dependant on technological expertise, but also upon cultural and linguistic expertise.

Higher quality of services
It is difficult to evaluate a possible increased quality of service in offshore outsourcing arrangements, since it is very difficult to make this comparison objectively. However, it seems that quality standards do not lead to the desired result in all cases. Researchers contribute this to two causes. First: the cooperation between outsourcing organization and vendor organization consists of two parties. Although one of them (the vendor) might be superior in their quality standards, they are also dependent upon the outsourcing organization. Amberg and Wiener, (2005) and Willcocks et al. (2006) state that the gap between standards like CMM(I) for example should not be too large; an indication that the chain is as strong as its “weakest” link. Another reason that expected quality is not always met is that the addressed quality standards themselves do not solely guarantee quality. Dibbern et al. (2006) and Currie (2003) stress the importance of creativity, business knowledge and application domain knowledge, while Ramasubbu, Krishnan and Kompalli (2005) address the lack of distributed managing project capabilities not captured in these maturity standards.

Reduction of time to market
The promises of follow the sun benefits, making more efficient use of time zone differences are unrealistic (Bhat, Gupta and Murthy, 2006; Shami et al., 2004) due to required extra coordination and miscommunications (Carmel et al., 2005). Conchuir et al. (2006) stress that the time zone differences should not be seen as an advantage, but a disadvantage, because the lack of a shared collaboration window. Carmel (1999) reveals that in more than half of the offshore development projects, the expected time to market reduction was not achieved.

1.1 Expectations wrap up
The offshore outsourcing market is growing rapidly, partly due to high and quite diversified expectations. Unfortunately, like has been illustrated, these high expectations are not always justified.

This research embraces the *potential* of offshore outsourcing and attempts to gain more insight in the reason behind failed expectations on the operational level to *prepare* organizations better for their offshore outsourcing journey.

1.2 Research Context: Utrecht University and Capgemini

After the brief introduction into the relevance of the subject in organizations above, this section will deal with the relevance of the subject *in a research context*, describing the environment in which the research was conducted. This master thesis is mainly conducted with the interests of two main interest groups taken in consideration, positioning myself in the middle of these interests.

1.2.1 Utrecht University

The Master Business Informatics from the department of informatics in the faculty of science has as its generic core focus topics: (1) to improve organizations to take advantage of ICT and (2) improving ICT for organizations (Business Informatics, 2007). This thesis deals mainly with the first of those topics, referring back to the expectations addressed earlier: Offshore outsourcing deals with *improving organizations* to take advantage of their (outsourced) IT by enabling focus on core competencies or obtaining cost reductions. Two courses within this master program to which this work is mostly related are Method Engineering and Knowledge Management. The former teaches an approach to construct models combining a series of activities with associated deliverables. Such a model was constructed as is shown in chapter two. The latter has proven very useful for relevant background literature.

There are a number of students that have worked and published on offshore outsourcing research topics in the Business Informatics program. The topic is thus acknowledged to have importance from a departmental scientific point of view and researchers from the department are gradually increasing knowledge in the field of offshore outsourcing.

1.2.2 Capgemini

A main stakeholder is IT consultancy organization Capgemini, from where most of this research was conducted. Capgemini is a large information technology, consulting, outsourcing and professional services organization headquartered in Paris. Capgemini has operations in 30 countries worldwide and employs approximately 75,000 employees (Capgemini, 2007) of which 15.000 are located in acquired organizations in India. The company is active in a wide diversity of disciplines through four major divisions:

- **Technology services**: The technology branch is mostly concerned with delivering new IT solutions in collaboration with the client. Due to this rather strict separation in expertise, Technology Services also handles outsourcing deals, related to new application development.
- **Outsourcing services**: Outsourcing Services deals mostly with large outsourcing contracts in applications management, business process outsourcing, infrastructure outsourcing and transformational outsourcing (*not* in application development).

- **Consulting services:** Consulting services advices in Customer Relationship Management, financial and employee transformations, Global sourcing, Operational research, Supply Chain Management and work closely together with the other branches.
- **Local Professional services**: Some of the local expertise, regarding application and infrastructure services and high tech consulting, lays in the hands of subsidiary: Sogeti

My research is conducted from the *Technology Advisory Services* practice in the Netherlands, located in the Technology Services division described above. Technology Advisory Services consists of consultants, enterprise architects and transformation managers. Sourcing is an important segment within Technology Advisory Services and the practice has a number of master students that perform their master thesis research here, thereby increasing the level of practice expertise and giving a more solid scientific foundation on sourcing service offerings that are provided to clients. By this, Capgemini and its clients are important stakeholders in this research.

1.2.3 Offshore readiness at Capgemini

Technology Advisory Services has the intellectual ownership over the so-called: Rightshore™ Assessment Study, an assessment to evaluate Rightshore potential for specific application(s) (groups) of the outsourcing organization; the client for Capgemini in this context. First a brief introduction regarding Rightshore will be given.

Rightshore: Rightshore is the (trade-mark protected) global IT delivery model that Capgemini uses. It basically attempts to access the right IT service in the right place at the right price for Capgemini clients. It makes use of an intensive corporate wide network of onsite, onshore, nearshore (another country, but in close proximity of outsourcing organization) and offshore resources owned by Capgemini (Capgemini 2007b).

Rightshore Assessment Study

Technology Advisory Services uses an offshore readiness assessment, called the Rightshore Assessment Study by which outsourcing client organizations can structurally examine their software applications for Rightshore potential. Two main dimensions are being assessed: Application complexity and Organizational capability. The main assumption behind the RAS-framework is: Low application complexity and high organizational capability results in an application (score) that has Rightshore potential. Depending on the risk tolerance at the outsourcing client organization, the application is perceived as being an *actual* good Rightshore candidate. This is represented in figure 2 below (by the dotted square box for risk aversity). The assessment approach also calculates financial consequences of different alternative delivery options by using Net Present Values. The financial side is out of scope of this research.

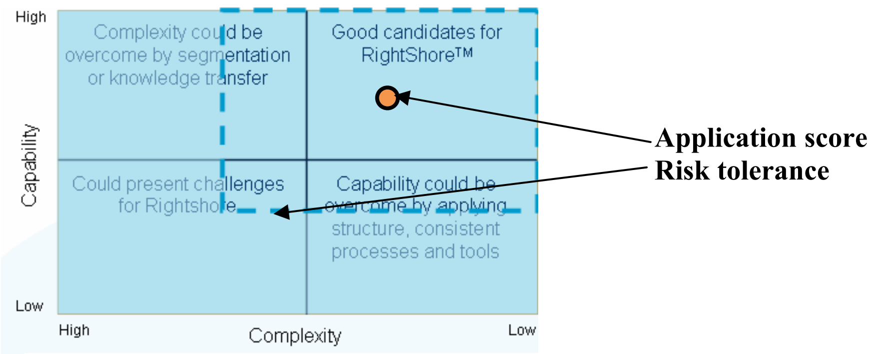


Figure 1: Rightshore Assessment Study

RAS has been derived from years of consulting and IT transformation consultancy experience and has great commercial potential, due to its financial base and relatively forthcoming presentation mode. This provides at the very least a good place to start initial conversation regarding an application landscape and in some cases an actual starting point for an offshore outsourcing project.

Our research will be based on offshore outsourcing application development risks exclusively, but the results in isolation can well be integrated into the assessment approach to evaluate the offshore variant.

1.2 Research context wrap up

After briefly describing the two main interests groups, it has become clear that both parties have an interest and stake in this research subject: The further development of a relevant scientific discipline from the Universities point of view, and increasing Capgemini's knowledge regarding offshore outsourcing risks, which enables them to better serve their clients.

I am positioned as the stakeholder in the middle, it is my primary priority to fully exploit the consensus in the two interests and find a balance in the possible conflicting ones.

1.3 Research Questions

At this point, the relevance of the research subject in society and for both stakeholder groups should be apparent. In this paragraph, we will zoom in further and introduce the concrete research questions as upbeat for the question what is currently lacking from both a scientific and practical point of view, which will be discussed in paragraph 1.4. The main question of this research is given below, followed by a subdivision of the three operational instruments of answering this question.

What impact does operational risk occurrence have in offshore application development projects and in what way can it be mitigated?

1. Is there an impact of controllable and incontrollable application and collaboration risk factor occurrence on project success in offshore outsourcing application development?

2. Does controllable risk have an impact on residual incontrollable risk occurrence?

3. How can an offshore readiness assessment be improved to include the results of question [1] and [2]?

It should be noted that the three questions are posed in terms of: Is there an effect? Although this will indeed be tested as the basic point of departure, the real value relates back to the main question: Within a significant effect, what is the relative contribution of different risk variables? These three research questions will form the backbone of the research in both content as well as thesis structure. We will refer to the specific concepts in the questions in the following chapters three and four mainly describing *scope* and *research constructs* concepts. The research questions themselves are also being subdivided and when combined answer the main research question. Chapter five introduces the latter subdivision in subquestions and hypotheses, when the *scope and research constructs* should be clear.

Scoping: Chapter three
These questions will be answered in chapter three and will direct the research in the desired direction. We will use literature research to answer these three questions.

Q1.a What is considered under outsourcing?
Q1.b What is considered under offshore outsourcing?
Q1.c What is considered under application development?

Research constructs: Chapter four
The following questions are dealt with in chapter four. They represent the research constructs that will be examined and are derived from a number of theoretical perspectives described in the next chapters. The constructs were validated in a number of expert interviews to check the constructs and their sub dimensions for possible biases, multiple interpretations and equal level of granularity.

Q1.d What is project success?
Q1.e What are application risk factors?
Q1.f What are collaboration risk factors?
Q1.g What are controllable risk factors?
Q1.h What are incontrollable risk factors?
Q3.a What is an offshore readiness assessment?

1.4 Research relevance

As illustrated above, much is known about expectations and often failed realization of these expectations with regard to offshore outsourcing. This paragraph attempts to explain why *the research questions* posed above provide a relevant contribution to at the one hand the research field and at the other hand adds value for Capgemini.

1.4.1 Scientific relevance of research

A lot has been written about *specific* risks factors in (offshore) outsourcing (Willcocks et al., 2006; Carmel et al., 2005; Beulen et al., 2005; Aubert, Patry and Rivard, 1998; Cramton, 2001; Carr et al., 1993; Khan et al,, 2003; Lee, 2001; Prickladnicki et al., 2003; Currie, 2003; Kshetri, 2007; Na, Simpson, Li, Singh and Kim, 2007) and there are a number of other scholars that mention them less explicitly.

The scientific contribution of this work is twofold: First, the scope of risks taken into account is broad, relating to different origins of risk in an operational setting. Most of the empirical evidence of risk factors described in literature comes from high level strategic risk analysis focusing either on offshore in general (Djavanshir, 2005), distributed work in general (Cramton, 2001), outsourcing risks in general (Bahli et al., 2005; Gonzalez, Gasco and Llopis, 2004) or onshore application development (Barki et al., 1993, Kraut and Streeter, 1995), some exceptions excluded (Beulen et al., 2005, Gopal, Mukhopadhyay and Krishnan, 2002). These authors use quite high-level container variables without sufficient neither discriminative explaining power though. Empirical evidence for the combined effect of offshore outsourcing application development risk in an operational setting is scarce.

Second, this thesis aims to assist IT outsourcing organizations with more advanced risk control and prioritization. Pfleeger (2000) stresses the importance of not only using expert judgments in risk weighing, but base them on historical data from projects. Boehm (1991) describes three different steps of a risk assessment that were followed here. Identification [1] and analysis [2] of risk is operationalized in such a way that the final step [3] of prioritization can be taken. Translating back empirically found relational strengths between risks and project success to specific risk factors that can be controlled, gives this paper its societal relevance. Moreover the contribution of different risk impacts is a delicate extension, reaching further than the undifferentiated manner in which risks are often described in scientific literature.

To conclude there is a lack of research on delicacy in risk factor impact and mitigation options from an outsourcing IT organization perspective. The current research field has basically come to just a big *list* of generic potential risks and best practices for mitigation in offshore outsourcing application development which are themselves valid, but lack a deeper understanding.

1.4.2 Societal relevance of research

Insight in the relevance of the risks assessed in the Rightshore Assessment Study, introduced earlier, would bring more delicacy in the tool. Such an increased delicacy would present the possibility for a more specific and accurate assessment for Rightshore potential, hereby providing the clients CIO or IT manager with a more funded ground for risk prioritization, prior to the decision of offshore outsourcing of the organizations application development.

Another point of value is provided by the manner in which Capgemini is executing and managing their offshore outsourcing projects. In some cases, Capgemini uses cases their Rightshore concept for handling a client's application development outsourcing. Clients can choose for this road, which is in itself a risk mitigation approach, due to the highly experienced and formalized approach that is being used through the local onshore delivery centers. In some cases however, the client decides to offshore parts of their application development more or less directly to India, with a project manager and team in between exclusively. The client advantage in this case is cost reduction, which is "paid" by more project risks involved. Competition for Capgemini is increasing however, with pure players like Tata, Wipro and InfoSys that can deliver IT services from India directly to the client without intervention and at lower costs. Insight in the most effective risk mitigation options would enable Capgemini to better prepare the client organization. A focused bounding and more accurate pinpointing of the risks involved offshoring their application development could be a mean to challenge the pure players.

1.4 Justification of research wrap up
To sum up: What is unknown until this date is the interaction between (different types of) risk (factors) and their respective relative importance. This thesis attempts to fill this gap departing from the perspective of the outsourcing organization and examining operational risk in offshore application development projects. Moreover, we will investigate different risk mitigation options by which IT organizations can more accurately control risk prior to offshore outsourcing.

1.5 Overview of thesis

The structure of this research follows the concepts in the three main research questions. The next chapter (Chapter two) will first describe the starting point of this research from a theoretical perspective applied to the practical trigger for this research: RAS. The purpose of this chapter is mainly to create the scientific starting point of outsourcing and derive a framework in which this study can be placed in a more generic context.

In chapter three will then be dealt with scoping of the research by the concepts described above. The goal is to canalize and direct the research into the right scope after which the derivation of research constructs will be made within this narrow scope. Chapter four addresses these constructs from a theoretical perspective and comes up with the two generic kinds of risks that will be examined. It also presents a subcategorization within them, resulting in our conceptual research framework. This will be endowed in Chapter five with hypotheses. After defining this framework, some methodological considerations will be made in Chapter six with regard to how the defined research constructs will be measured and researched. Chapter seven describes the results of the risk survey that was conducted. Chapter eight draws the conclusions from these results, after which this research work will be reflected upon in the discussion of Chapter nine.

2 Theoretical and practical perspectives on outsourcing

This chapter provides the points from which this research was started from a scientific and practical point of view. It describes some high-level theories on outsourcing and provides more background information with regard to the offshore readiness assessment: “RAS” that is used at Capgemini. From that point onwards, the research will be scoped to the operational project level in the sequential chapters.

2.1 Theoretical starting point: Different perspectives

In the introduction, some of the most addressed expectations with outsourcing were described from a practical point of view, but what lays underneath can be better described from a theoretical viewpoint. This theoretical perspective will be applied to help develop our research constructs from high-level theories. Resource-based View; RBV (Penrose, 1959; Wernerfelt, 1984), extended by the Dynamic Capabilities; DC (Teece, Pisano and Shuen, 1997) and Transaction Costs Economics; TCE (Williamson, 1981) are among the most widely cited theories used to deal with the concept of outsourcing. Resource based View is connected to outsourcing in multiple sources (Erickson et al., 2006; Singh et al., 2006; Lewin et al., 2006; Stark et al., 2006, Looff, 1996; Amberg et al., 2005; Dibbern et al., 2004; Dibbern et al., 2007; Levina et al., 2003). A rather new perspective, which extends the Resource based view of the firm: Dynamic Capabilities (Teece et al., 1997) is also considered in this chapter. Transaction Cost economics is used in outsourcing context in multiple researches (Dibbern et al., 2004; Sabherwal, 2003; Singh and Zack, 2006; Willcocks et al, 2006).

2.1.1 Resource-based View

The resource-based view of the firm sees a firm as a collection of productive resources, which can be both tangible and intangible (Jashapara, 2004). Resources and productivity are positioned to be two different sides of the same coin (Wernerfelt, 1984) and organizations compete with each other based on the portfolio of resources (Singh et al., 2006; Stark et al., 2006).

Barney, (1991) in Jashapara (2004) describes a number of characteristics of resources that can lead to competitive advantage, assuming resources are distributed heterogeneously across firms:

- Rarity: not widely held resources
- Valuable: resources promote efficiency and effectiveness
- Not imitable: the resources can not be replicated easily
- Not substitutable: other resources cannot fulfill the same function
- Not transferable: the resources cannot be bought in resource markets.

The assumption is that, if possible, acquiring or developing such kinds of resources leads to business value and competitive advantage (Balaji and Brown, 2005b). The notion of a company’s core competence is derived from this (Jashapara, 2004). Core competencies exist of unique resources or a combination of unique resources that cannot be leveraged by competition (Levina et al., 2003).

The next step towards using the resource-based view to explain make or buy decisions (Singh et al., 2006; Balaji et al., 2005b; Dibbern et al., 2004) is not a big one. Generally speaking: If the lack of resources in a specific part of IT is too large, this can be rectified with acquiring resources from outside firm boundaries using sourcing arrangements (Singh et al., 2006; Balaji et al., 2005b). This places vendor capabilities bridging the resource gap in a key position (Dibbern et al., 2006). The resources that can be leveraged by the vendor are not unique by nature, because they can be acquired by competing firms as well, so it seems impossible to achieve sustained competitive advantage merely by outsourcing (Singh et al., 2006). However, leverage of existing non outsourced resources (core competences) at the outsourcing organization *can* improve.

Unfortunately this is the point where the RBV stops; it focuses at stable resource potential, not focusing at potential to improve upon or manipulate resources. This is where another theoretical perspective can be placed: Dynamic Capabilities (Teece et al., 1997).

2.1.2 Dynamic Capabilities

The dynamic capability framework is an extension of the Resource Based View (Erickson et al., 2006). The term dynamic refers to the capacity to renew competences (Teece et al., 1997; Lee, 2001) Dynamic capabilities are defined as a firm's ability to integrate, build and reconfigure internal and external competences to address rapidly changing environments (Teece et al., 1997).

One of the key differences between Resource Based View and Dynamic Capabilities is that the latter looks at an organization as consisting of a bundle of capabilities (that can change) instead of a bundle of (static) resources (Balaji et al., 2005b). With this philosophy in mind, a step can be taken to attempt to improve the dynamic capabilities of the outsourcing firm to better exploit their resources.

The concept of dynamic capabilities is relatively new (Jashapara, 2004) and the applicability on outsourcing has not been widely addressed (Balaji et al., 2005b). However, departing from the primary definition of the dynamic capabilities, the ability to integrate, build and reconfigure internal and external competences to address rapidly changing environments, brings us close to outsourcing. The research field has illustrated that outsourcing; especially offshore outsourcing requires the outsourcing organization to have their internal processes/procedures on track (Capgemini, 2006, Carmel and Tjia. 2005; Na et al., 2007, Gopal et al., 2002; Herbsleb et al., 1999). The capabilities that the client organization should retain or develop can thus be seen as dynamic capabilities for reconfiguration of existing resources leading to a more successful exploitation of offshore outsourcing. In fact, this entails developing the potential to manipulate existing resources in such a way that they become offshore ready.

Besides the resources and capabilities an organization possesses, cost reduction was still the number one trigger for offshore outsourcing as mentioned in the first chapter. When focusing on the actual costs of outsourcing, we stumble upon another theoretical perspective: Transaction Cost Economics.

2.1.3 Transaction Cost Economics

Transaction Cost Economics (TCE) provides a basis for analyzing outsourcing from an economic point of view (Dibbern et al., 2006, Singh et al., 2006). An important objective of TCE is to identify conditions under which market governance is more cost efficient than hierarchical governance; (Looff, 1996) of producing internally (Dibbern et al., 2007) and assumes that in practice, markets are not perfect (Looff, 1996). Two kinds of costs are distinguished.

- Production costs: the costs directly associated with production (Singh et al., 2006, Williamson, 1981)

- Transaction costs: The costs involved in the transaction; the exchanges of goods or services between economic actors, who are technologically separate units, found inside or outside the organizations (Looff, 1996).

It is expected that minimizing the sum of these two costs determines the choice for producing an asset internally or in the market by outsourcing (Evaristo, Scudder, Desouze and Sato, 2004). In this context, it is supposed that some parties in a transaction behave opportunistically and take advantage of opportunities at the expense of the other party (Dibbern et al., 2007). This threat of opportunism is considered being dependent upon contingencies like:

- *Asset specificity:* The investments made to support a transaction have a higher value to that transaction compared to one for another purpose. This brings some power tension in the client - vendor relation to bear. Subdimensions of asset specificity are, found in Looff (1996):
    - Site specificity: transactions are only available at a certain location.
    - Physical asset specificity: refers to how specialized the equipment must be to complete the transaction.
    - Human asset specificity: refers to how specialized the required knowledge must be to complete the transaction (Williamson, 1981).
- *Uncertainty*: The amount of uncertainty in the transaction caused by a wide range of things like unpredictable markets, technological trends or contractual complexity (Singh et al., 2006).
- *Frequenc*y: Refers to the frequency with which parties transact. The more frequent the parties transact, the less costly the transaction will be over time.

In this research perspective, relating TCE explicitly to outsourcing, it is believed that in IS outsourcing, production costs are low, but transaction costs are high (Looff, 2006). In general TCE would suggest that relatively common and stable activities would be the most cost advantageous candidates for outsourcing (Singh et al, 2006). The uncertainty derived from the multiple contingencies described above, provides the scientific justification for *risk* in offshore outsourcing, since offshore outsourcing is merely a specific kind of outsourcing (Dibbern et al., 2006).

2.2 Practical starting point: Offshore readiness assessment

After a brief introduction in Section 1.2.3, this paragraph describes Capgemini's offshore readiness assessment RAS more thoroughly. A careful preparation of the offshoring journey means that the outsourcing organization inventories its own applications and systems (Carmel et al., 2005). RAS deals like any risk assessment with the prior *identification* of potential unsatisfactory outcomes (Bahli and Rivard, 2003). The RAS-assessment is performed prior to offshore outsourcing and in cooperation with the outsourcing organization. This enables an in depth analysis of the organization and its applications.

2.2.1 Risk assessments in practice

According to Boehm (1991), risk management consists of risk assessment and risk control. A risk assessment consists of three main steps: Identification, analysis and prioritization of risk. Two operationalizations of risk assessments are reviewed briefly here as upbeat for RAS.

ABN assesses offshore readiness of their applications by pre-determined risk categories like maintenance capacity, maturity of IT project Management, Infrastructure complexity, experience, support capacity and organizational flexibility. The results are presented in a spidergraph representing internal strengths and weaknesses (Carmel et al., 2005).

Wipro, a pure player from India uses a quantitative assessment of application complexity to calculate predicted transition time of an application to be brought offshore and calculate the ideal onshore-offshore mix. The input variables are the application profile, scope of work, expertise, stability, documentation, processes and tools (Kuni et al., 2006).

Risk assessments similar to RAS are thus performed in practice. The next subsection briefly describes the process and concepts of the RAS assessment.

2.2.2 Activities and concepts in the RAS Assessment

To give an impression of how the RAS assessment is being performed, a process-data model was constructed (See Appendix 12.4). The technique of performing this meta-modeling was adopted from the work of Weerd and Brinkkemper (2007).

There are two process flows in the RAS-assessment which start simultaneously. The left side of the model consists of a process flow. The left one of the two process flows resembles the development of a business case, which is part of the assessment approach. The result of this business case is an overview of costs and benefits, expressed in terms of Net Present Values. This financial side of the assessment is left out of scope. The right side of the process flow resembles the assessment with respect to content. All activities are connected to a concept in the right side of the model. They represent the deliverables and constructs of the assessment. Familiarities with Unified Modeling Language (UML) modeling are noticeable. The composition, aggregation and multiplicity constraints between concepts should be interpreted similarly. Rectangles with a black background represent closed concepts. They are not elaborated on further. Rectangles with an open white background are open concepts which are discussed in bigger detail.

The assessment sheet of RAS contains all 158 questions and 0 or more scores on these questions. The assessment score is the super class of Application score and Organization score which represent the two main pillars of the assessment. The next subparagraph elaborates on these concepts as upbeat for our contribution to RAS.

2.2.3 Pillars of RAS

The main pillars of the RAS assessment are the complexity of the application and the capability of the organization. Within these two main pillars, a subcategorization is made like illustrated in table 30 and 31 respectively and visualized in the process data model in Appendix 12.4.

Table 30 Application

Subcategory	Description
Architecture complexity: 12 items	Determines the complexity of the existing architecture. A distinction is made between technical, data and application architecture.
Business logic complexity: 10 items	Determines the complexity of the business logic supported by the application. Also confidentiality and importance of availability of the application (data) is considered here.
Application complexity: 18 items	Determines the complexity of the application in general. Aspects as application maturity, number of modules, GUI's, lines of code, annual releases are considered as well as level of customization.
Interface complexity: 9 items	Determines the complexity of the internal and interfaces by their number. Also the intensity with which transactions are made over these interfaces and the customization of the interfaces is taken into account.
Infrastructure complexity 14 items	Determines the complexity of all with regard to IT infrastructure, including maturity of servers, number of different development environments and languages as well as interaction patterns between application stakeholders.
Interaction complexity 24 items	Determines the complexity of interactions in all stages of the software life-cycle with regard to documentation availability, interactions and hand-off interaction coordination.
Support complexity 10 items	Determines the complexity of the support function. This depends on the required intensity and type of support. Also support toolsets and methodologies are taken in consideration.

Table 31 Organization

Subcategory	Description
Knowledge Management 11 items	Determines the knowledge management processes, tools and documentation that are available in the organization.
Methods 14 items	Determines the extent by which (development) methods, processes and tools exist in the organization and are sufficiently used.
Requirements 11 items	Determines the maturity of the requirements environment (to develop High Level Design requirements). Technical and business requirements are considered.
Test 15 items	Determines the maturity of the test environment; The degree by which application knowledge is required to test, the process, responsibilities and the way it is followed is considered.
People 10 items	Determines the structure of the IT organization with regard to people. Development programs and training is considered. Also experience with the application, methods, tools and processes is taken into account as well as employee turnover.

Weighing factors
The activity "validate / adjust scoring" is linked to the WEIGHING FACTOR concept. However in practice, there is no weighing performed to the subdimensions of the RAS constructs from the tables above. Every subdimension has a similar weight which brings us to the contribution of this

work for RAS. New weights of the subdimensions should be assigned in a RAS derivative RAS', customized for application development, based on the findings of risks in the projects.

2.3 Perspective views combined

Two remaining questions are: (1) Can the theoretical perspective be combined and applied to RAS and (2) what is its purpose in this thesis? Figure 2 shows the different perspectives in one picture and illustrates the first question. From the Resource based view perspective, an outsourcing decision is taken if the internal resources are insufficient compared to those of the vendor organization. The dynamic capability perspective has shown that these resources are not static and can be manipulated by developing relevant capabilities. The Transaction Cost economics perspective expects costs to be a dominant factor in the outsourcing decision. The outsourcing organization evaluates production costs and adds transaction costs of outsourcing. When these costs are lower than keeping them in-house, they will be outsourced.

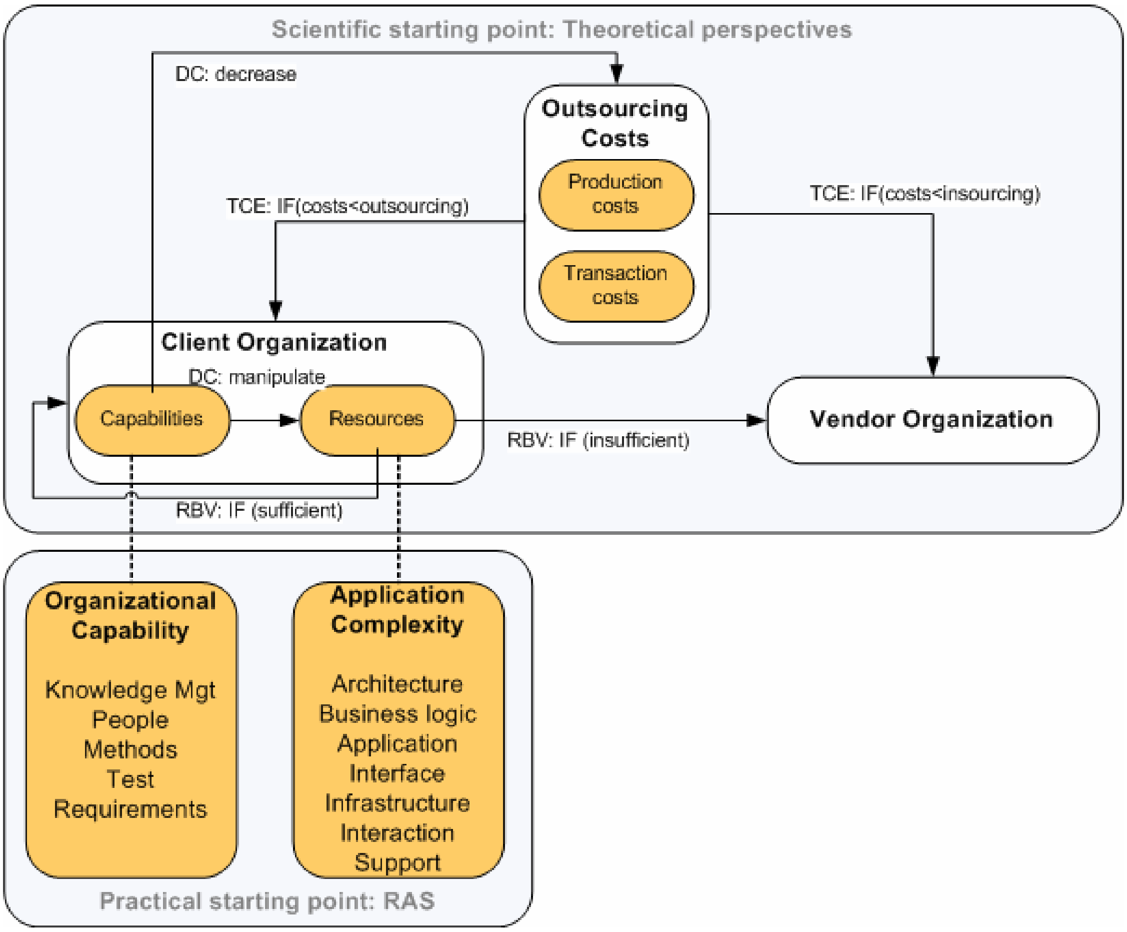


Figure 2: Perspective views combined

The combination of the three theoretical perspectives can be applied quite well to the two main pillars and assumptions of RAS. These assumptions are (1) that a high organizational capability leads to good offshore potential and (2) highly complex applications lead to bad offshore potential. The combined figure of perspectives illustrates this resemblance. As described, the capabilities of the outsourcing organization are being assessed in RAS. These capabilities can manipulate existing resources (among which applications). Capability leads to a positive manipulation of application complexity (the organization is better able to deal with application complexity), which decreases outsourcing costs through the dynamic capability advantage and decreases the risk in transaction costs. This enabler for outsourcing is assumption 1. The other assumption is that existing resources, narrowed down here to simple applications can result in outsourcing. The arrow from resources to the vendor organization should be read here like: “insufficient” in the sense of: that the application is imitable, non unique and transferable. When this is the case, an organization can seek alternatives in outsourcing. The application complexity partly deals with exactly these topics. “How strategic is the application, how dependent is the organization on it, the extent of unique knowledge required to manage it etc.

There is however an important trade-off point to be made here. Investing in capabilities means both decreasing outsourcing costs as well as being able better to manipulate existing resources which makes the applications better transferable. On the other hand, when the application is strategic and serves the core competences of the organization, the same argument could be used to insource the application back. Within the scope of this research, talking about non strategic applications, the RAS-assessment seems to have a theoretical base in the three theories presented though.

The answer to the second question, relating to the purpose of this chapter in the thesis is to reveal the importance of risk evaluation, to show from what sources it originates and to give a theoretical foundation for risk mitigation. This thesis is mainly about risks identification and mitigation prior to offshore outsourcing application development. Based on the Transaction Cost Economics, an organization evaluates costs to determine make or buy decisions. Costs refer here to all activities that require effort and in the end money. Prior to offshoring, the exact costs are not clear, but potential *risks* on costs have been evaluated in research. Mitigating the right risks, could thus increase the chance of success in outsourcing. Developing or retaining the right capabilities at the client organization, increases the potential of more successfully deploying resources to outsource other ones. This stems with our second research question in which we will investigate what controllable risk factors have the potential to reduce risks for an offshore outsourcing project.

3 Offshore Outsourced Application Development

This chapter is most of all important to elaborate upon the scoping concepts from our research questions and defines the exact scope that will be chosen regarding the concepts introduced in the preface of this thesis. By this, the research is channeled in the desired direction and secondary validity is limited to the scope defined.

Q1.a What is considered under outsourcing?
Q1.b What is considered under offshore outsourcing?
Q1.c What is considered under application development?

3.1 Outsourcing

When Eastman Kodak decided to outsource their data centre to IBM in 1989 (Tsotra and Fitzgerald, 2007), one of the first outsourcing contracts was made and not long afterwards, many organizations followed their example (Palvia, 1995). At the moment outsourcing IT services or in other cases even whole business or IT functions to external vendors is every day practice and in aggressive highly specialized markets even becomes an important mean to survive (Herbsleb et al., 1999). Different forms of outsourcing can be distinguished.

3.1.1 Business Process Outsourcing

It is necessary at this point to provide some scoping to the concept of outsourcing. There are a lot of studies that deal with outsourcing topics, but many fail to address the kind of outsourcing they are actually investigating. Organizations can outsource entire business functions like for example their logistics or their financial department. This form of outsourcing is often addressed by the term **Business Process Outsourcing** (BPO) and defined by (Kshetri, 2007) as:

"*Long term contracting of a firms (the client firm) non core business processes to an external service provider*".

3.1.2 Information Systems Outsourcing

The form of outsourcing described above involves in principle non IT processes, but in most cases there is (at least some) facilitation by IT (Khsetri, 2007). However, according to Dibbern et al. (2004) this kind of outsourcing is fundamentally different from pure **Information Systems (IS) outsourcing** due to the pervasive character of the latter. In this thesis, we will focus on IS outsourcing and embrace one of the most widely cited definitions for IS outsourcing (Yang et al., 2000) from Loh and Venkatraman (1992):

"*The significant contribution by external vendors in physical and/or human resources associated with the entire or specific components of the IT infrastructure in the user organization".*

In the remainder of this research, we will abbreviate "IS outsourcing" with just "outsourcing", although the distinction made here remains valid. This definition is quite generic and it should be, because the concept of IS outsourcing can still be very diverse. Complete IT-functions can be outsourced (Khsetri, 2007), but in most cases organizations outsource specific parts of their IT-function. (Tramacere and Marriott, 2005; Carmel et al. 2005; Willcocks et al. 2006), from which most is application related (Tramacere et al., 2005). In general it can be said that IS sourcing consists of infrastructure outsourcing and application outsourcing (Young et al., 2006) where infrastructure outsourcing can be thought more of the hardware side of IS outsourcing. Bhattacharya et al. (2003) provide the following definition: *Infrastructure outsourcers provide hardware, infrastructure integration, network services, security, system operations, and support* *services.* A useful way to describe *application outsourcing* is by using the software life cycle (Carmel et al., 2005)

3.1.3 Outsourcing in the software life cycle

IEEE (1990) defines the software life cycle as: "*The period of time that begins when a software product is conceived and ends when the software is no longer available for use"* (Carr, Konda, Monarch, Ulrich and Walker, 1993). The typical life cycle includes a: concept phase, requirements, design, implementation, test, installation + checkout, operation and maintenance and a retirement phase (Carr et al., 1993). A Gartner research from Tramacere et al. (2005) states that application development relating to the first steps in the software life cycle until installation and checkout is outsourced the most, followed by application maintenance and application management. A brief overview in reversed order of these three forms is provided below.

Application Management
Application management refers to maintenance, support and enhancements activities of the application portfolio (Levina et al., 2003). Basically this indicates that application management deals with all of the steps in the software life cycle. Outsourcing your application management thus indicates letting go all aspects of the application portfolio.

Application Maintenance
Application maintenance refers to maintenance of existing (legacy) applications (Tramacere et al. 2005), which have been developed years ago (Feiman, 2005). It also entails maintenance of newly developed applications.

Application Development
As stated, most of the IS outsourcing that occurs is application development outsourcing (Tramacere et al., 2005). Application development relates to the development of new applications or in most cases new modules that are build upon existing applications. Application Development is concerned with the Requirements analysis, design, construction (coding), testing and implementation (deployment) of software applications (Steenbeek, Wijngaert, Brand, Brinkkemper and Harmsen, 2005; Carmel et al., 2005). There is an important point of delicacy to be made here. In this thesis, emphasis and scope lays at *Custom* Application Development *as opposed to* component-based software development where the development is entirely based on pre-existing software components (Kotlarsky, 2005).

Application development has a strong project focus as compared to other kinds of outsourcing; e.g. Application maintenance or infrastructure management that represent more long-term arrangements (Carmel et al., 2005; Beulen et al., 2005; Erickson et al., 2006). Moreover it is non-routine and depending on many social interactions between IS professionals (Balaji et al., 2005; Dibbern et al., 2007). These characteristics seem to make this form of outsourcing such a willing research subject.

3.1.4 Outsourcing model

To sum up this section, an overview and hierarchy in the field of outsourcing is given in figure 3 below. This picture is based on the different definitions from literature described above and on the description of overlapping forms of different kinds of outsourcing.[1]

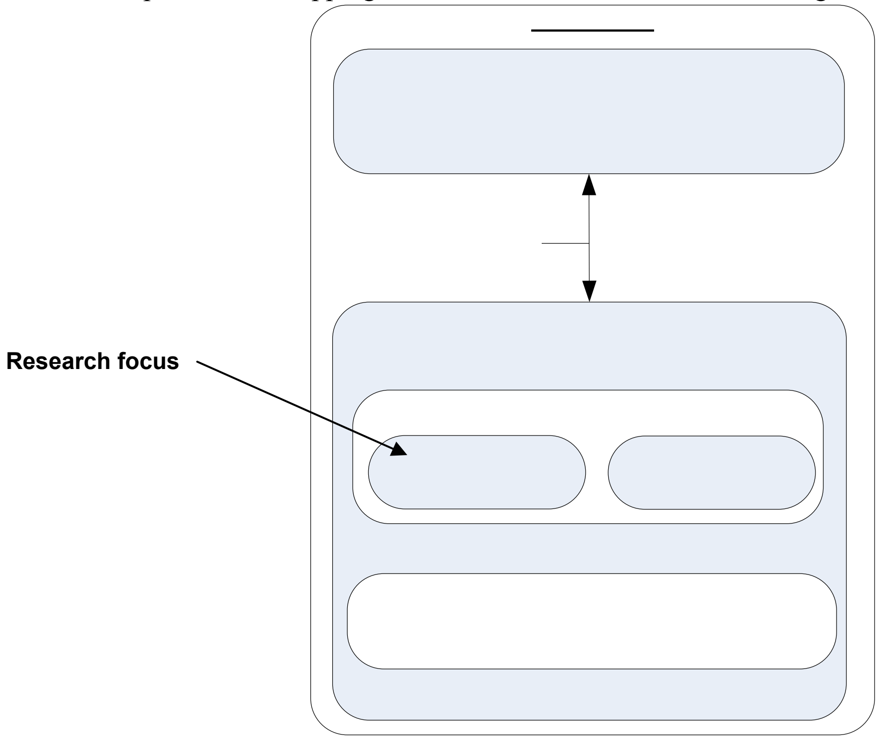


Figure 3: Outsourcing model

Outsourcing is represented as the super class, consisting of Business Process Outsourcing and IS outsourcing. The link between IS outsourcing and Business Process outsourcing is IT facilitation provided from the IT side, but BPO does not deal with pure IT processes. IS outsourcing is the super class of Infrastructure outsourcing and Application (Management) Outsourcing. The latter consists of and Application Maintenance and Application Development outsourcing. Our focus will be on Application Development Outsourcing due to its solid base in scientific literature and widely application in practice. The first part of the scoping has now been described. Another possible scoping that can be made is evaluating the location to outsource application development to.

3.2 Offshore outsourcing

During the 1990's a specific variant of outsourcing (Layman et al., 2006): offshore outsourcing began to increase attention (Lewin et al., 2006). Companies discovered a large supply of well-trained English-speaking IT-specialist in countries in South-East Asia. The dotcom bubble made the offshore industry develop even faster (Lewin et al., 2006; Carmel et al., 2005). Nowadays of the Fortune top 1000 companies, 60% is involved in some kind of offshore outsourcing (Beulen, 2007). In this thesis, we will focus on custom application development, where project teams work together in the same team over distance, across national, organizational (and cultural) barriers and

[1] Note that this representation is based on certain definitions picked from literature. Due to the lack of consensus, some researchers slightly deviate in their definitions and thereby would come to a respective slightly different picture.

cannot meat with each other regularly. The onshore part of the team is referred to in this research as the **Front-office** (both client as well as native service provider team that is located onshore). **Back-office** refers to the offshore part of the distributed development team where the development work is being offshored to. These terms are derived from Lacity et al. (2006). They are also used widely in practice at Capgemini in sourcing arrangements. Another reason to define these concepts explicitly is that it creates an unambiguous point of reference to be used in our research survey. Having defined outsourcing itself thoroughly in the previous section, this section briefly zooms in on the different alternatives that an organization can have in outsourcing. This distinction is based largely on the work of Steenbeek et al. (2005). Two dimensions will be addressed: outsourcing location and vendor selection.

3.2.1 Location selection

The first dimension that will be described is the outsourcing location. Steenbeek et al. (2005) distinguishes five alternatives of sourcing location options[2]. Similar categorizations are also embraced by Tsotra et al. (2007), Erber et al. (2005), Carmel et al. (2005) and are used in practice (Capgemini, 2007; McCarthy, Martorelli, Moore, Agosta and Ross, 2004; Young et al., 2006).

- **Onsite sourcing:** This is performing activities inside the organization at the client location. Onsite sourcing is not outsourcing.
- **Onshore sourcing:** Onshore sourcing relates to outsourcing activities to a vendor in the same country.
- **Nearshore sourcing:** Nearshore sourcing deals with outsourcing activities to a nearby country or region.
- **Offshore western sourcing:** Offshore western sourcing refers to sourcing to another continent or far away region with a Western culture. Note that in the preface of this research was indicated that the perspective of the western outsourcing organization was chosen.
- **Offshore sourcing:** Offshore sourcing is similar to offshore western sourcing, with the exception that the other continent or region has a non-western culture.

Roughly said, all other things being equal (ceteris paribus), the economic value for IT outsourcing organizations increases in the order (top-down) described; especially the cost reduction incentive. However, risk exposure follows the same line.

Working across national, organizational and cultural distances can cause problems (Carmel and Agarwal, 2001). Organizations can choose to retain their IT activities in-house completely or they might prefer to outsource their activities in the country of origin, only exceeding organizational boundaries. Another alternative might be to choose a vendor in a country nearby, usually against lower costs. For European countries: Ireland and countries in the east of Europe are possible candidates (Tramacere et al., 2005; Erber et al., 2005); and for North-American organizations, countries like Brazil and Mexico (Chandrasekaran et al., 2004) can be good nearshore country candidates. This way, national and cultural boundaries and time zones are only trespassed through a limited degree. For this reason some of the major Indian software development firms are building nearshore centers in proximity of their clients (Chandrasekaran et al., 2004). The third alternative deals with offshoring IT activities to another continent, where wages (and production costs) are even lower than in nearshore alternatives. This increases the physical distance and time zone differences between clients and vendors to an even bigger extent.

---

[2] Note in the referenced paper the term sourcing instead of outsourcing is mentioned, because it also includes insourcing in its categorization

In practice, the fifth alternative: Offshore non-western is the most relevant form of offshore outsourcing, because vendors in non western countries like China, but more importantly; India dominate offshore outsourcing from our defined Western perspective. (Lewin et al., 2006, McCarthy et al., 2004; Tramacere et al., 2005; Carmel et al., 2005). Their cultures are quite different from Western cultures (Carmel et al., 2005), creating potential communication and coordination problems (Carmel and Agarwal, 2001; Kobitzsch, Romback and Feldmann, 2001). This thesis deals exclusively with offshore outsourcing to non Western countries.

3.2.2 Vendor selection

Another dimension of the alternative options is the vendor selection process. Basically an outsourcing organization has three alternatives (Beulen et al., 2005),

- **Captive service providers:** Captive service provision involves insourcing. The client and the vendor are part of the same legal entity (wholly owned).
- **Native service providers:** Native service providers, provide services in different developing countries all over the globe e.g. India, while headquartered in the developed (Western) regions. These are global organizations like Capgemini, Atos, EDS, IBM and Logica CMG. In this research a distinction is made between native service provision with the Rightshore centre as intermediating party and without (Direct offshore).
- **Foreign service providers:** Foreign service providers operate directly from the developing countries and having only sales offices in the developed regions**.** A few examples of Foreign Service providers are Wipro, InfoSys, Cognizant, Tata and Xansa.

A similar categorization is described in Steenbeek et al. (2005), describing different ways of cooperating in outsourcing agreements. An important distinction made here, that should be added to the whole process of scoping is that of single versus multiple outsourcing, referring respectively to clients that choose *one* or *multiple* vendors for their outsourcing deal. Since application development is very project oriented and quite clearly defined, in practice there is only one offshore outsourcing vendor per application. Also, our research focus is based on operational risks instead of strategic risks, a distinction that will be discussed later. Single outsourcing is thus taken for granted within the research scope.

Some researchers only distinguish between captive sourcing versus non-captive sourcing (Lewin et al., 2006; Dibbern et al., 2007; Kobitzsch et al., 2001; Mirani, 2007; Carmel et al., 2002), but the distinction above from Beulen et al. (2005) proves to be a more useful distinction. The reason for this is that from a practical point of view, outsourcing organizations have the explicit choice in engaging in offshore outsourcing agreements through an intermediating party (native) *or* directly (foreign). Our research is being conducted from Capgemini. The focus will thus be at risks involved from a native service provider point of view.

3.3 Shoring and sourcing research scope

Figure 4 below illustrates the different shoring and sourcing alternatives within scope. The native service provider in the offshore non-Western box at the right represents Capgemini in India in both models. Our interest lies in the intermediate offshore delivery through the onshore **(upper model)** Rightshore centre, reviewed from the onshore as well as the offshore viewpoint. The other within scope form is that of direct offshore **(lower model)** service delivery from the native service provider (Capgemini) offshore in India.

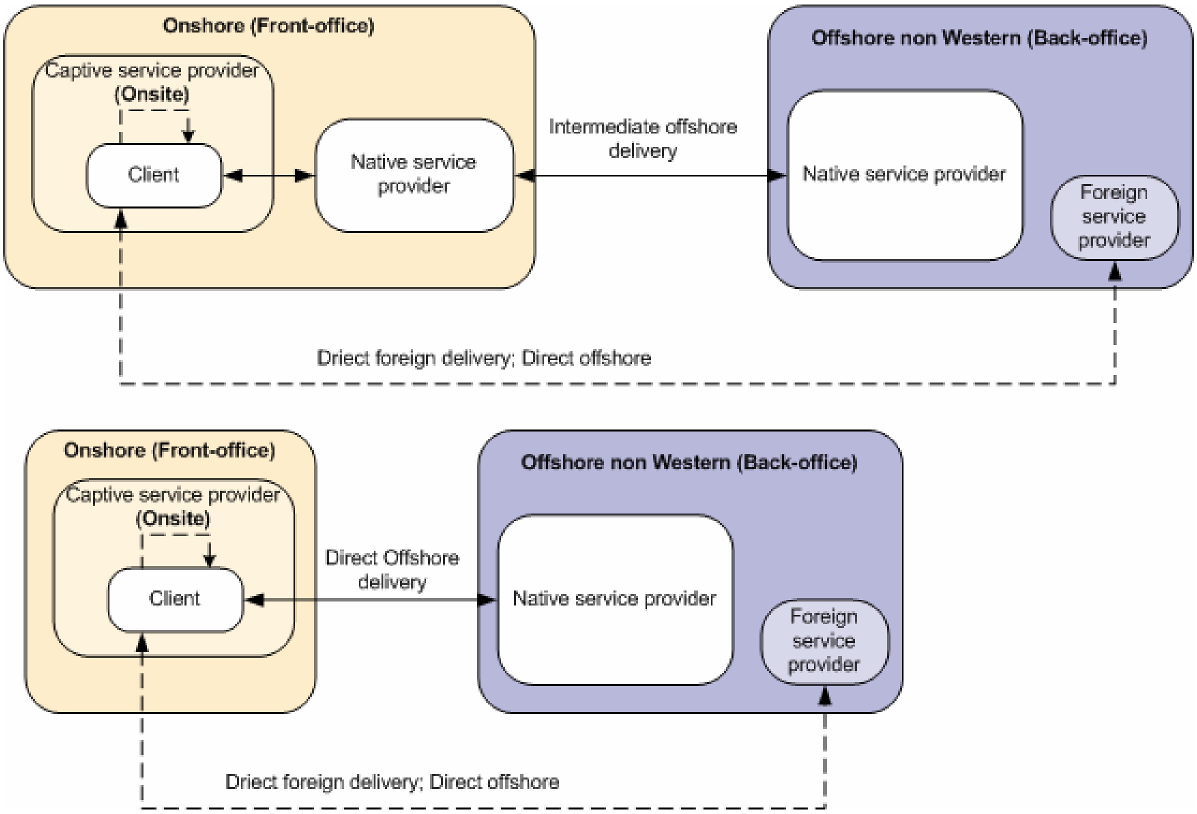


Figure 4: Shoring and sourcing options within scope

Having thoroughly described our research scope, we will conclude with the final topic of interest: **(Distributed) offshore outsourcing custom application development**. This is defined as:

The custom development of new application(s) (modules) across time zones, national, organizational and cultural boundaries in a team with an offshore (back-office) and an onshore (front-office) part, that do not have frequent physical communication, within a native service provision context.

4 Research constructs: Offshore outsourcing risks

Similar to the previous chapter, this chapter explains concepts in the three general research questions. However, rather than scoping constraints, this chapter deals with the actual development of aggregate research constructs that will be examined in the survey design. The following sub-questions will be handled in the following order:

Q1.d What is project success?
Q1.g What are controllable risk factors?
Q1.h What are incontrollable risk factors?
Q1.e What are application risk factors?
Q1.f What are collaboration risk factors?

The chapter concludes with the worked out version of the conceptual model below in figure 5. The arrows in the picture indicate an expected relation between concepts. The following paragraphs zoom in on the different concepts of these models, starting of with the dependent variable: Project success.

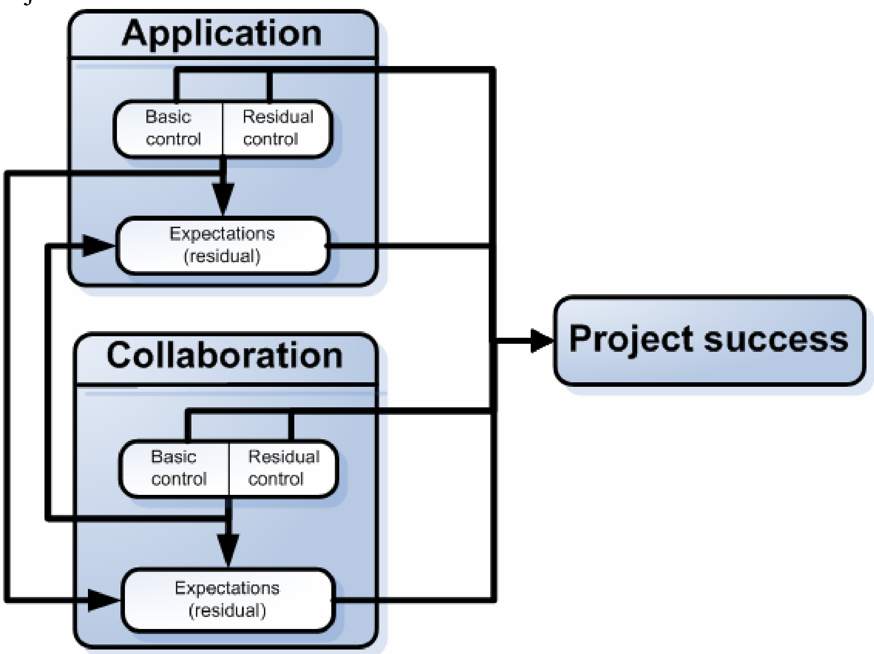


Figure 5: Conceptual model

4.1 Project success

Before talking about risk at all, our point of departure is *project success*. After all: risk refers to negative consequences; performance risk (Nidumolu, 1996) and it is good knowing, to what concept these negative consequences are actually referring to.

It should be noted that it is ambiguous to define project success, because success seems to be in the eye of the beholder, but attempts were made (Na et al., 2007; Prickladnicki, Audy and Evaristo, 2004; Balaji and Brown, 2005; Bhat et al., 2006; Jiang and Klein, 2000). By integrating them, the final definition of project success elements in this study can emerge. First, a distinction is made in talking about project success on the strategic, tactical and operational level.

On the one hand there is strategic success that the client organization has with offshore outsourcing. This form of success is not based on a specific project, but more on the total combination of projects in a strategic program. Na et al. (2007) and Carmel et al. (2005) emphasize the realization of cost reduction as one the most important factors for this kind of success. Being the primary trigger for offshore outsourcing, this seems a valid assumption, but other realizations of strategic offshore outsourcing triggers, addressed in the introduction might just as well bring strategic success. A study relating risk factors to strategic project success is done by Lee (2001).

On a tactical level, the client organization has the need for (new) functionality to be developed in an (existing) application. Project success on this level is reflected primarily by meeting the goals of this functionality, preferably against quality service levels, like addressed in Erickson et al. (2006). Two other important objective factors (Na et al., 2007) are project budget and time schedule. A project is typically budgeted as well as time-limited and sticking to the budget within time is an important part of project success (Jiang et al., 2000; Erickson et al., 2006; Na et al., 2007). An application that meets its functional goals and quality standards might prove to be less useful when it is delivered over time or with a (massive) exceeding of project budget. A final tactical factor to be considered is more subjective (Na et al., 2007): Satisfaction of client organization stakeholders with the end-product (Erickson et al., 2006; Balaji et al., 2005). A lack of stakeholder satisfaction might inhibit future project sourcing arrangements.

Project success on the operational level can also be subjective and is more about team member satisfaction (Bhat et al., 2006) with the development result and process. The front-office and back-office team might provide the outsourcing organization and its stakeholders the desired application. If the operational collaboration in the process of developing was extremely troublesome and team members were not at all satisfied with it, this can have its negative repercussions in future sourcing arrangements though. This kind of satisfaction can thus be also seen as a form of project success (Na et al., 2007).

Table 1 below describes the nine items embraced in this study for project success. A brief description of the item and the reference to the source it is addressed in is given.

Table 1 Project success

Item	Description	Source(s)
Purpose achieved	Measurement of extent by which the purpose of the developed application was achieved	(Erickson et al., 2006)
Quality achieved	Measurement of extent by which quality service levels of the developed application were achieved	(Erickson et al., 2006)
Budget	Measurement of extent by which the original budget was overrun (percentage)	(Na et al., 2007) (Jiang et al., 2000) (Erickson et al., 2006)
Time schedule	Measurement of extent by which the original time schedule was overrun (percentage)	(Na et al., 2007) (Jiang et al., 2000) (Erickson et al., 2006)
Client satisfaction	Measurement of degree by which the client organization was satisfied with the developed application	(Na et al., 2007) (Erickson et al., 2006) (Balaji et al., 2005)
Back-office satisfaction	Measurement of degree by which the back-office team was satisfied with the developed application	(Bhat et al., 2006) (Na et al., 2007)
Front-office satisfaction	Measurement of degree by which the front-office team was satisfied with the developed application	(Bhat et al., 2006) (Na et al., 2007)
Back-office process satisfaction	Measurement of degree by which the back-office team was satisfied with the process of developing the application	(Bhat et al., 2006) (Na et al., 2007)
Front-office process satisfaction	Measurement of degree by which the front-office team was satisfied with the process of developing the application	(Bhat et al., 2006) (Na et al., 2007)

The relevance of each of the three levels in this definition might change depending on the outsourcing organization and/or project. This study deals exclusively with tactical / operational project success, since it takes a specific project as central point of interest. There were no means in the study to evaluate and reflect upon the higher strategic success. In short:

Figure 6: Project success

Project success is determined by: the completion of a project within time and budget, while realizing the intended purpose, quality standards, stakeholder satisfaction and team satisfaction with both end results as the process leading to it.

4.2 The concept of risk

Risk is a very generic concept and it can relate to all kinds of fields. Since this thesis deals with risk as one of its core concepts, it should deserve a proper definition in the correct context.

4.2.1 Risk elements

It is good to indicate that the definition of risk itself is not even that important. It is more interesting to look at elements that make up the actual risk exposure. A widely cited definition provided by Boehm (1991) for risk exposure in a software context is:

*Risk exposure is the probability of an unsatisfactory outcome P(UO) * L(UO): the loss to the parties affected if the outcome is unsatisfactory*

This definition, or more specifically: the breaking down of risk in these two elements is embraced in many studies (Bhattacharya, 2003; Na et al., 2007; Bahli et al., 2005; Aubert et al., 1998; Barki et al., 1993; Carmel and Tjia, 2005). We will abbreviate this definition to make it more operational in context of this research, by not talking about probability of unsatisfactory outcome and potential loss, but rather **Occurrence** and **Impact** respectively.

An important conceptual contribution is adopted from Bahli and Rivard (2001). They place *another layer* in between occurrence and impact; that of the **Scenario**. By doing this, the distinction between risk and risk factor is made. The risk factor does not necessarily influence project success directly. Rather it influences the likehood of a scenario (risk) which on its turn influences project success. To give an example: Why would a standardized environment bring project success. There does not seem to be direct logical if-then relation. Instead, it is more logical to reason that the standardized environment (risk factor) decreases the likehood of a scenario in which more controlling communication is required to make clear what the task is about (risk). This on its turn is very closely related; has an impact on project success (e.g. exceeding time schedules). This study uses the term *"risk variable"* for indicating risk items in general (risk or risk factor) from this point onwards.

4.2.2 Risk types

One of the aims of this research is to give some foundation to an existing offshore readiness study, based on organizational capability. The distinction we choose to make in the research is according to these reasoning lines. Fed by expert interviews and research practitioners, a 2x2 categorization of generic risk types was made.

As addressed in Nidumolu (1996), software development project performance risk outcomes can be measured before the project or in the latter stages of the project. The latter is referred to as *residual performance risk.* It is assumed that risk exposure changes during the project (Nidumolu, 1996; Na et al., 2007). We included this line of reasoning into the first categorization. Risk exposure is different at different time intervals in the project. In practice this means: Before the project starts there is a *basic risk point of departure (basic risk)*, representing risk boundary potential. While the project proceeds and approaches its end, the risk exposure progresses based on the basic risk and its outcome in practice. This is *the residual performance risk*; *(residual risk*) which represent risks in practice. The residual risks in this research will not be measured during the project, but afterwards, based on compliance to standards and deviation of expectations of the potential risks the project started with. This will be explained more thoroughly below.

The other risk type categorization that is made in the research is also based partly on Na et al. (2007) and Nidumolu (1996). They state that risk consists of a controllable and an incontrollable part. A similar distinction was made by Pfleeger (2000), describing voluntary versus involuntary risks. The first type of risk (controllable) can be controlled using specific remedies, the latter can not. The concepts that will be embraced are *controllable* and *incontrollabl*e for both basic risk and residual performance risk. “Incontrollable” in our research context refers to a lack of *direct* manipulation potential. Incontrollable risks just have a different origin; they are in fact the real risks, similar to the scenario described above. They cannot be *influenced* on the *individual* risk level. This does not mean that they cannot be influenced at all; it is assumed that certain risk factors can affect the risk. In fact one of our hypotheses (H2) is that controllable risk factors can influence incontrollable risks.

More concretely, with controllable risk, we indicate the risk factors that can be controlled directly before or during the project; in most cases this refers to characteristics surpassing the specific project boundaries like availability and use of standards/methods/tools. The controllable risk will be the central measurement concept of organizational capabilities that the outsourcing organization can manipulate/change prior to offshore outsourcing. It is assumed that integrating the compliance *to* the basic capabilities in combination with these basic capabilities gives a good indication for organizational capability as a whole. Incontrollable risks refer to implicit project risks like inherent task-specific complexity and trust/motivation. These incontrollable risks are measured by means of deviating expectations.

The reason for using exceeding expectation for residual incontrollable risks is that it enables standardizing the outcomes of potential risks of projects to a set of project expectations, where the project and project success is based upon. This resembles the scenario discussed earlier. By doing this, a distinction between risk and risk factors (influencing the risk) is made. Moreover, without using this initial expectation set as a central point of reference, the effects of potential risks would become fully project dependent.

To give an example: When a project team manager knows before the project starts, that the external interface complexity of the application is very high, he would likely consider this in his project resources. In other words: This complexity is a *potential* hazard and by asking for the deviation in the complexity from expectations, the *actual* realization of the potential risk is measured. Sakthivel (2005) makes a similar claim indicating that 100.000 lines of code can be either a large application or a small one, all depending on previous experience with this kind of application size.

To sum up, the categorization (Basic risk, Residual risk) * (Controllable risk, Incontrollable risk) is represented in figure 7 below.

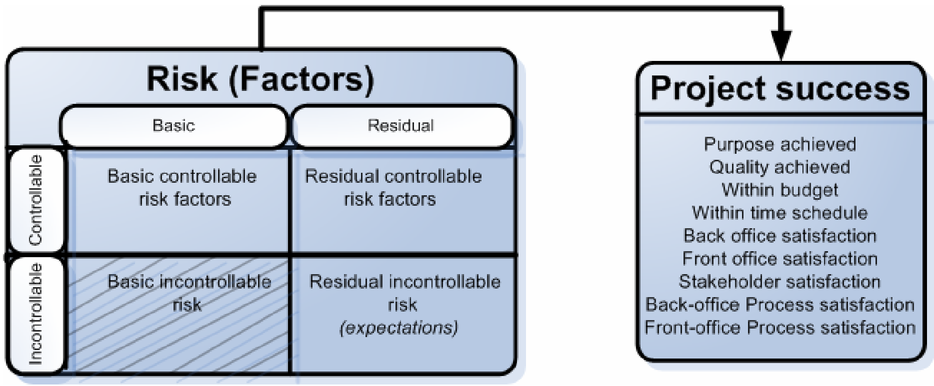


Figure 7: Risk (factor) types

After defining the concept of risk and describing the four different types of risk, the next paragraph is about the two different content types of risk: application and collaboration risk. By placing these research constructs in the right boxes, the actual content of the risk survey will be filled.

4.2 The concept of risk wrap up

- *Risk* consists out of two elements: occurrence and impact
- *Risk* in this study refers to the residual incontrollable risks measured in deviating project expectations. They have a potential effect on project success.
- *Risk factors* refer to basic and residual controllable risk, potentially influencing risk (and thereby eventually) project success.

4.3 Application risk variables

The first content-container research construct is application risk. Using the 2*2 matrix as the container content structure, different risks variables derived from scientific literature and expert interviews are introduced. For sake of readability, this chapter was written backwards compatible with the results from expert interviews that were conducted to validate the risk variables. The choice for inclusion of risk variables is primarily based on differentiated explaining factors.

4.3.1 Rationale behind application risk

The reason for distinguishing between application and collaboration risk factors follows from the way offshore application development projects are being conducted which deviates from "traditional application development". As written by some researchers, (it is believed that) traditional application development risks are being exacerbated by elements of the offshore distributed character of the project (Carmel, 1999; Prikladnicki et al., 2003; Sakthivel, 2005; Conchuir et al., 2006; Sabherwal, 2003; Cramton, 2001).

For this reason, we decided to isolate traditional application risk from the total project risk exposure. The other segment; collaboration risk is described in the next paragraph and represents risk factors that originate from this exacerbated risk pool. An important note on the latter however, is that our research intent is *not* to *compare* between offshore projects and onshore projects, but merely to compare the occurrence and impacts of these two different risk *origins* in an offshore context. This to potentially reveal that the offshore distributed character is more (or less) determining for project success than these traditional application development risks. This might also provide the basis for interesting further research options in comparing between offshore and onshore projects. The four different kinds of application risk variables, based on the 2*2 matrix introduced above are being described in the following subparagraphs. This will be mostly based on two generic (cost estimation) risk models.

The first of these models is the COCOMO (Constructive Cost Model) II model developed by Boehm, Abts, Brown, Chulani, Clark, Horowitz, Madachi, Reifer and Steece (2000b) which evolved from the original COCOMO model. The main purpose of this COCOMO II software cost estimation model is to estimate the costs, effort and schedule when planning a new software development activity (CSE, 2007).

The other source is the Taxonomy Based Risk Identification (Carr et al., 1993). The taxonomy is built up of risk Classes (main categorization of risks), Elements (subclasses within classes) and Attributes (which characterize the elements). We will mainly focus on the "Class" distinction to categorize and present (chapter six) different kinds of risks. Some of the elements and attributes are used as well, but the deviation at that level from Carr et al. (1993) is so large for a multitude of reasons, that we will not refer to these specific elements and attributes explicitly. Figure 8 below illustrates the filled in risk model with the operationalized application risk variable categories. These will be described in this paragraph.

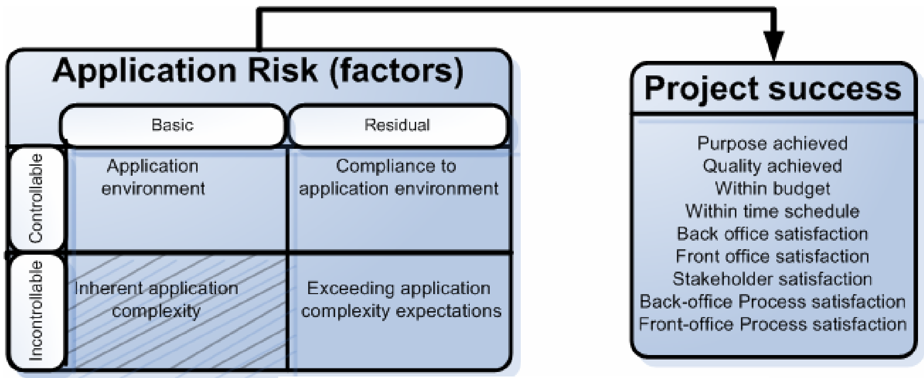


Figure 8: Application risk (factors)

4.3.2 Basic controllable application risk: Application Environment

The first kind of application risk variables are the basic controllable risks factors which relate mainly to the Development Environment that describe the methods, procedures and tools used to develop the application (Carr et al., 1993). Since this subparagraph deals with basic controllable risk factors, the questions derived from these factors refer to the a priori existence of (similarity in) methods, procedures and tools in the project between front-office and back-office.

The following tables (1-6) below summarize the basic controllable application risk factors used in this research. The three columns describe respectively: The risk factor, a brief description of the factor and the (scientific) source(s) it is addressed in. These sources do not always literally refer to the risk factor description, but do indicate the specific factor to be a valid risk factor within the scope of this research.

Service level Agreement
The Service Level Agreement describes project deliverables according to timelines, required quality service levels and role responsibilities (Carmel et al., 2005). A detailed Service level agreement has a positive effect on project success (Sakthivel, 2005; Currie, 2003; Favela and Pena-More, 2001; Beulen et al., 2005).

Table 2 Service Level Agreement factor

Risk factor	Description	Source(s)
Service Level Agreement (SLA) detailedness	Measurement of the level of detail in the service level agreement.	(Carmel et al., 2005) (Sakthivel, 2005) (Currie, 2003) (Favela et al., 2001) (Beulen et al., 2005)

Existing application understanding
Three retained application factors are mentioned. The retained application in this context refers to the (legacy) application in the front-office on which new modules are being built. In some projects, the building of new functionality has to be built upon other applications or modules and these factors can have an effect on the eventual project success.

High cohesion in the application code leads to better application understanding according to Boehms (2000; 2000b) COCOMO model. A High self-descriptiveness, referring to the quality of the existing application documentation, leads to better application understanding (Boehm, 2000; Boehm et al., 2000b; Carmel et al., 2005; Mirani, 2007, Herbsleb et al., 1999; Carr et al., 1993). The programmer's familiarity with the existing application leads to a better application understanding (Boehm, 2000; Boehm, 2000b; Carr et al., 1993).

Table 3 Existing application factors

Risk factor	Description	Source(s)
Existing application structuredness	Measurement of the structuredness of the existing application in terms cohesion.	(Boehm, 2000) (Boehm et al., 2000b)
Existing application self-descriptiveness of application code	Measurement of the self-descriptiveness of the application documentation.	(Boehm, 2000) (Boehm et al., 2000b) (Carmel et al., 2005) (Mirani, 2007) (Herbsleb et al., 1999) (Carr et al., 1993)
Existing application programmers familiarity at back-office	Measurement of the programmer's familiarity with the existing application.	(Boehm, 2000) (Boehm et al., 2000b) (Carr et al., 1993)

Architecture standards
Architecture refers to the coherent blueprint of the structure of the software or modules to be developed (Carr et al., 1993). Architecture planning and design is one of the nine Core IS capabilities that should be retained in house, according to the Feeny and Willcocks (1998) framework to enable sustainable exploitation of the IS function. Since this subparagraph deals with basic controllable risk, the architecture in this context refers to the standardization of

architectural blueprints. Based on expert interviews and a similar approach in the Rightshore Assessment Study, architecture was split in three parts.

Table 4 Architecture standard factors

Factor	Description	Source(s)
Data architecture standardization	The level in which the data architecture; the set of data linkages, processes and technologies the organization had selected for the creation of the application, was based on standards.	(Feiman, 2005) (Pries-Heje, Baskerville and Hansen, 2005)
Application architecture standardization	The level in which the application architecture; the relationships (internal and external) among the application components and modules, was based on standards.	(Feiman, 2005) (Pries-Heje et al., 2005)
Technical architecture standardization	The level, in which the technical architecture; the design and building blueprint of software architecture rationales with focus on the interaction between software and hardware, was based on standards.	(Feiman, 2005) (Pries-Heje et al. 2005)

Technical tool support

Kotlarsky (2005), in her dissertation mentions a number of software development tools, suggested to support global software development. Technical tool support in the context of controllable and basic risks refers to the level of sophistication in development tool support. A similar use of tool support is embedded in a software development risk assessment. (Barki et al. 1993; Kuni et al., 2006). To describe the tool support to current day standards, we embrace the COCOMO II description and operationalization described in Boehm et al. (2000b) and distinguish between application engineering tool support and testing tool support (Boehm et al., 2000b).

Table 5 Technical tool support factors

Risk factor	Description	Source(s)
Application engineering tool support	Measurement of sophistication in application engineering tool support that is advanced; consisting of strong mature pro-active lifecycle tools, well integrated with processes, methods and reuse (Boehm et al., 2000b).	(Kotlarsky, 2005) (Boehm et al., 2000b) (Sakthivel, 2005) (Barki et al., 1993) (Carmel et al., 2002)
Testing tool support	Measurement of sophistication in testing tool support, consisting of assertion checking, integration of automated analysis and test tools and model based process management (Boehm et al., 2000b).	(Kotlarsky, 2005) (Boehm et al., 2000b)

Development method maturity

Besides the type of development method (e.g: agile, waterfall), that is used in the project, a basic controllable risk is the similarity in maturity of using the development method between front-office and back-office (Evaristo et al., 2004; Prikladnicki et al., 2003). The focus with development methods with regard to risk factors often lies more on the type of development method used (Balaji, Ahuja and Ranganathan, 2006; Layman et al., 2006), rather than on the maturity of using the development method. However, based on some of the expert interviews that were conducted, it appeared that this difference in maturity seems to be an important perceived risk factor in application development projects and was thus included in the research.

Table 6 Development method factor

Risk factor	Description	Source(s)
Development method maturity similarity	Measurement of similarity in the level of maturity of the development method between front-office and back –office.	(Evaristo et al., 2004) (Prikladnicki et al., 2003)

Process maturity
Another controllable basic risk factor is process maturity. As discussed briefly in the introduction, most of the back-offices operate at CMMI level 4/5 (Dibbern, et al., 2006; Erber et al., 2005). A number of scholars addressed the CMM Process maturity gap (in specific Key Process Areas) as a risk factor in application development projects (Willcocks et al., 2006; Evaristo et al., 2004, Gopal et al., 2002; Amberg et al., 2005; Carmel et al., 2005). A general assumption in offshore outsourcing work is that using the same methods/tools/processes has a positive effect on the quality of a system (Sakthivel, 2005) and requires less effort to obtain common understanding. Like addressed in Willcocks et al (2006), Amberg et al. (2005) and Carmel et al. (2005) it is recommended to elevate process maturity (if applicable) to at most two levels of operations in the back-office team.

Based on expert interviews, we decided to include seven Key Process Areas from the Capability Maturity Model Integration in our research (CMMI, 2006). The choice was based first on the seven more basic Key Process Areas described on maturity level 2 in the continuous representation (CMMI, 2006). This choice was refined based on some expert interviews in which became clear that certain Process Areas were also useful to include and some more low-level processes were less relevant within our scope. We included Requirements Development (level 3) and Causal Analysis and Resolution (level 5) and excluded Process and Product quality assurance and Supplier Agreement Management. Within this risk category, we focus on basic controllable risks, thus on the a priori similarity (gap) in process maturity between front-office and back-office.

Table 7 Process maturity factors

Risk factor	Description	Source(s)
Project Planning process maturity similarity	Measurement of similarity in the process of establishing and maintaining plans that define project activities. (CMMI, 2006).	(Willcocks et al., 2006) (Amberg et al., 2005) (Carmel et al., 2005)
Causal Analysis and Resolution process maturity similarity	Measurement of similarity in the process of identifying causes for defects and other problems and take action to prevent them from occurring in the future (CMMI, 2006).	(Willcocks et al., 2006) (Amberg et al., 2005) (Carmel et al., 2005)
Requirements Development Process maturity similarity	Measurement of similarity in the process to produce and analyze customer, product and product component requirements (CMMI, 2006).	(Willcocks et al., 2006) (Amberg et al., 2005) (Carmel et al., 2005)
Requirements Management process	Measurement of similarity in the process to manage the requirements of the projects	(Willcocks et al., 2006) (Amberg et al., 2005)

maturity similarity	products and product components and to identify inconsistencies between those requirements and the projects plans and work products (CMMI, 2006).	(Carmel et al., 2005) (Gopal et al., 2002)[3]
Configuration Management process maturity similarity	Measurement of similarity in the process to establish and maintain the integrity of work products using configuration identification, configuration control, configuration status accounting and configuration audits (CMMI, 2006).	(Willcocks et al., 2006) (Amberg et al., 2005) (Carmel et al., 2005) (Gopal et al., 2002)[3]
Measurement and Analysis process maturity similarity	Measurement of similarity in the process to develop and sustain a measurement capability that is used to support management information needs (CMMI, 2006).	(Willcocks et al., 2006) (Amberg et al., 2005) (Carmel et al., 2005)
Monitoring and Control process maturity similarity	Measurement of similarity in the process to provide an understanding of the projects progress so that appropriate corrective actions can be taken when the project's performance deviates significantly from the plan (CMMI, 2006).	(Willcocks et al., 2006) (Amberg et al., 2005) (Carmel et al., 2005)

4.3.3 Residual controllable application risk: Compliance

Another kind of application risk variables are the residual controllable ones. Within our research context, this refers not so much to new risk factors, but rather *another dimension* of factors already described in 4.3.2: basic controllable risks.

When describing tools and processes, most scholars address the basic controllable risks, referring mainly to standardization, but skip the actual compliance to them in practice. Given the basic point of departure represented by the level of sophistication in the technical tool support and the process maturity, the question posed here is to what degree the tools were used sufficiently and consistently and to what extent team members complied with the used processes. For example: Cusick and Prasad (2006) stress the need to reassure compliance of coding standards. Kotlarsky (2005) makes a more general statement by mentioning the importance of compliance to common processes and tools in a distributed project. Below follows a tabular description (table 8-10) of the risk factors in this category. The source column has been left out, since the originating sources of the risk factors is the same as in the previous paragraph. The other dimension of the factor, namely compliance instead of basic level of sophistication was brought forward to be relevant very strongly in the expert interviews.

Table 8 Technical tool support compliance

Risk factor	Description
Sufficient use of Application engineering tool support.	Measurement of sufficient use of the application engineering tool support.
Consistent use of application engineering tool support.	Measurement of consistent use of the application engineering tool support.

[3] Refers to CMM Key Process Area; not CMMI

Sufficient use of Testing tool support.	Measurement of sufficient use of the testing tool support.
Consistent use of Testing tool support.	Measurement of consistent use of the testing tool support.

Table 9 Development method compliance

Factor	Description
Development method compliance	Measurement of compliance to the development method

Table 10 Process compliance

Risk factor	Description
Project Planning process compliance	Measurement of compliance to the process of establishing and maintaining plans that define project activities. (CMMI, 2006).
Causal Analysis and Resolution process compliance	Measurement of compliance to the process of identifying causes for defects and other problems and take action to prevent them from occurring in the future (CMMI, 2006).
Requirements Development process compliance	Measurement of compliance to the process to produce and analyze customer, product and product component requirements (CMMI, 2006).
Requirements Management process compliance	Measurement of compliance to the process to manage the requirements of the projects products and product components and to identify inconsistencies between those requirements and the projects plans and work products (CMMI, 2006).
Configuration Management process compliance	Measurement of compliance to the process to establish and maintain the integrity of work products using configuration identification, configuration control, configuration status accounting and configuration audits (CMMI, 2006).
Measurement and Analysis process compliance	Measurement of compliance in the process to develop and sustain a measurement capability that is used to support management information needs (CMMI, 2006).
Monitoring and Control process maturity compliance	Measurement of compliance to the process to provide an understanding of the projects progress so that appropriate corrective actions can be taken when the project's performance deviates significantly from the plan (CMMI, 2006).

4.3.4 Basic incontrollable application risk: Inherent complexity

The basic incontrollable application risks refer to the implicit application complexity, which is dependent upon the actual application to be developed. There has been made a deliberate choice **not** to include these risk factors in this research. The reason for this is twofold:

First, it is very difficult to integrate an objective measurement of inherent application complexity in a survey research, because this would require a number of viewpoints (e.g. architect, project manager, programmer) per project and a multitude of measurement items to be included.

Second: It is even more difficult to relate this complexity directly to project success, because the operationalization of project success itself depends among others on exceeding budget and time schedule. It is likely that some a priori known basic complexity would result in an increased amount of project resources (time, budget) dedicated to "compensate" for this complexity. This would thus distort the sec relation between complexity and project success.

Rather than looking for the complexity, we decided to make a fundamental choice to not look at the complexity, but instead ask for deviating, misaligned expectations in complexity which is dealt with in the next subparagraph.

4.3.5 Residual incontrollable application risk: Deviating expectations

The residual incontrollable risks refer to these deviating expectations, more specifically: exceeding (complexity) expectations. Our respondents consist of Project Managers which might not have the technical knowledge that enables them to accurately pinpoint implicit application complexity in the application. Generally they can however, give an indication of the amount of exceeding expectations on element of complexity.

Like addressed in the previous subparagraph, the project budget and time schedule are partly based on the expected difficulty (complexity) of the, to be developed application. Making use of these exceeding expectations standardizes the complexity measurement to the basic set of expectations a given project began with and based on which it should rollout according to planning. This enables a comparative impact measurement of exceeding complexity expectations on project success between projects. Important note is that this leaves thus in the middle the question of the *actual* application complexity impact on exceeding expectations. The following tables (10-16) summarize the residual incontrollable application risks.

Skills
The first set of risk factors relate to skills or capabilities of project team members. These risks were asked for the other side team skills; which reflects the expectations character better. Human skills on different aspects of the programming work to be done are not uncommon to be considered. Kuni et al. (2006) take human expertise and team skills in an assessment into account to calculate offshore – onshore percentages. Carr et al. (1993) mention a set of technical skills that might be lacking in an application development project. Boehm et al. (2000b) focus on personnel capability in the early stages of software development as relevant factors influencing cost estimation. Other than that, human skills in software development projects in general are addresses also in Jiang et al. (2000), Barki et al. (1993) and Aubert et al. (1998).

Since the length of our survey is limited, we decided to follow the COCOMO II approach and focused more on the different generic steps of software development (Boehm et al., 1993) Table 11 below shows the derived risks.

Table 11 Human skills

Risk	Description	Source(s)
Requirements development skills	Measurement of the extent by which requirements development skills in the other side team were according to expectations.	(Boehm et al., 2000b) (Carr et al., 1993)
Architecture design skills	Measurement of the extent by which architecture design skills in the other side team were according to expectations.	(Boehm et al., 2000b) (Carr et al., 1993)
Application coding skills	Measurement of the extent by which application coding skills in the other side team were according to expectations.	(Boehm et al., 2000b) (Carr et al., 1993)
Software testing skills	Measurement of the extent by which software testing skills in the other side team were according to expectations.	(Carr et al., 1993)

Deployment skills	Measurement of the extent by which deployment skills in the other side team were according to expectations.	(Carr et al., 1993)

Requirements
One of the most important global software development risks is related to the requirements phase of software development (Prikladnicki, Audy and Evaristo, 2006). The requirements phase asks for much communication between the front-office and back-office team (Sakthivel, 2005), which is exacerbated in a distributed team (Na et al., 2007). Prikladnicki et al. (2003) and Prikladnicki et al. (2006) opt for face to face requirements elicitation, because functional business requirements can easily be misunderstood due to the organizational and national distance (Na et al., 2007). In general stable business requirements (Gopal et al., 2002; Herbsleb et al., 1999; Na et al., 2007; Boehm et al., 2000b) and the need for detailed requirements (Prikladnicki et al., 2006, Cusick et al., 2006; Prikladnicki et al., 2004) are addresses to overcome the difficulties of global software development. Also the level of familiarity (precedented requirements) with similar requirements seems to have a positive impact on a project (Tiwana, 2004; Boehm et al., 2000b).

For the operationalization of these risks, we choose to distinguish between functional (business) and technical requirements. In the Taxonomy based Risk Identification, Carr et al. (1993) mention all the aspects of requirements described above and add a few. Based on the expert interviews that were conducted, we decided to include requirements completeness; the amount of unwritten requirements as another risk. The risks are summarized in table 12 below.

Table 12 Requirements

Risk	Description	Source(s)
Technical requirements stability	Measurement of the stability in the technical requirements compared to what was expected.	(Carr et al., 1993) (Boehm et al., 2000b)
Functional requirements stability	Measurement of the stability in the functional requirements compared to what was expected.	(Carr et al., 1993) (Boehm et al., 2000b) (Gopal et al., 2002) (Herbsleb et al., 1999) (Na et al., 2007)
Technical requirements completeness	Measurement of the amount of unwritten technical requirements compared to what was expected.	(Carr et al., 1993)
Functional requirements completeness	Measurement of the amount of unwritten functional requirements compared to what was expected.	(Carr et al., 1993)
Technical requirements detailedness	Measurement of the detailedness in the technical requirements compared to what was expected.	(Carr et al., 1993) (Prikladnicki et al., 2006) (Cusick et al., 2006) (Prikladnicki et al., 2004)
Functional requirements detailedness	Measurement of the detailedness in the functional requirements compared to what was expected.	(Carr et al., 1993) (Prikladnicki et al., 2006) (Cusick et al., 2006) (Prikladnicki et al., 2004)
Technical requirements precedents	Measurement of the amount of precedented technical requirements compared to what was expected.	(Boehm et al., 2000b) (Carr et al., 1993) (Tiwana, 2004)

Functional requirements precedents	Measurement of the amount of precedented functional requirements compared to what was expected.	(Boehm et al., 2000b) (Carr et al., 1993) (Tiwana, 2004)

Business logic

Besides the complexity of grasping the client's business processes due to distances, this risk is concerned with a somewhat related area; the complexity of the business rules that are incorporated in the new application. Dibbern et al. (2006) gives an example of an application were even the client organization has lost track of the business rules in an application. Tiwana (2004) uses business rules as a mean to assess back-office application domain knowledge.

Although not formally, but indirectly addressed as being risks by the authors above, business logic complexity is still included in the research. Primary reason is the frequent mentioning of the risk in expert interviews and the presence of the risk in the Rightshore Assessment Study.

Table 13 Business logic

Risk	Description	Source(s)
Business logic complexity	Measurement of the extent by which the business rules to be captured in the new application were more complex than expected.	(Dibbern et al., 2006) (Tiwana, 2004)

Architecture and interfaces

Partly based on earlier referenced authors the architectural design of the application can be more complex than expected. For the application architecture: the relationships among the application components and modules, we reference to internal and external interface complexity here respectively.

The internal interfaces are the points were the software system under development interacts with other components in the system under development (Carr et al., 1993). The external interfaces refer to the points were the software system under development interacts with other systems, sites or people (Carr et al., 1993).

The specification of interfaces is a critical part of software development efforts, especially (exacerbated) in a distributed team context (Herbsleb et al., 1999). Smith, Mitra and Narasimhan (1996) even question the suitability of big scope project for offshoring due to the large amount of external interfaces. The internal and external interface complexity is also part of the Rightshore assessment study. The derived risks are described in table 14.

Table 14 Architecture and interfaces

Risk	Description	Source(s)
Data architecture complexity	Measurement of the degree by which the data architecture was more complex than expected.	
Internal interface complexity	Measurement of the degree by which the internal interfaces were more complex than expected	(Herbsleb et al., 1999) (Carr et al., 1993)
External interface complexity	Measurement of the degree by which the external interfaces were more complex than expected.	(Herbsleb et al., 1999) (Smith et a., 1996)

Technical architecture complexity	Measurement of the degree by which the technical architecture was more complex than expected.	

Platform
Software and hardware platform complexity refers to two elements. First: the newness of the platform and second: The extent by which the unfamiliarity caused for unforeseen project efforts. This is based mainly upon Barki et al. (1993) and Boehm et al (2000b).

Software platform here refers to the set of compilers and assemblers and programming language supporting the development of the software system (Boehm et al., 2000b). Hardware platform complexity refers to an archaic development environment.

Aspray et al. (2006) relates software platform standardization to more successful offshoring. Pfleeger (2000) describes using a new software platform to be an important risk factor. In the COCOMO II model, a lack of (software) platform experience is a risk that contributes to software development project costs (Boehm et al., 2000b, Boehm, 2000). Table 15 shows the two derived risks.

Table 15 Platform complexity

Risk	Description	Source(s)
Hardware platform complexity	Measurement of the extent by which the hardware platform was more archaic than expected.	(Aspray et al., 2006)
Software platform complexity	Measurement of the extent by which the hardware platform unfamiliarity caused unforeseen project efforts.	(Boehm, 2000) (Boehm et al., 2000b) (Pfleeger, 2000) (Aspray et al., 2006)

Module diversity
The module diversity, the extent by which the content of modules was different from each other is addressed in Carmel et al. (2005) and Carr et al (1993). The latter mentions module cohesiveness as successful quality attribute.

Modules are defined here as parts of an assembly (Kotlarsky, 2005), providing a distinct portion of functionality. This indicates that they are inherently diverse. However, when the diversity in the type of modules is small, there might be learning effects to speed to process of development. Conversely, large diversity means re-inventing the wheel over again.

Table 16 Module diversity

Risk	Description	Source(s)
Module diversity	Measurement of the extent by which the content of the modules was more diverse than expected	(Carr et al. 1993) (Carmel et al. (2005)

Data transfer security
Data transfer security requirements are mentioned quite often as a risk, but more often than not, they are described more in the context of strategic risks or high-level inhibitors of outsourcing (Young et al., 2006; Aspray et al., 2006). Regardless of this more strategic origin, it seems that data security requirements can also be a risk in a more operational context (Carr et al., 1993; Lewin et al., 2006; Currie, 2006).

Table 17 Data transfer security

risk	Description	Source(s)
Data transfer security requirements	Measurement of the extent by which data security requirements were more stringent than expected.	(Carr et al., 1993) (Currie, 2003) (Lewin et al., 2006)

4.3.6 Out of scope application risk

A wide diversity of the traditional application development risks were considered, but all on a high and accessible level for project managers. In order to compare between individual projects, we decided to ask project managers with an overall view of the project to fill out our research survey. This has a backside in the level of depth by which questions could be answered. The real deviation from expectations of application complexity can be asked in depth, but the respondents would likely not be able to answer these questions accurately. Our purpose is to see which risk factors in general have the biggest impact. In depth-risk variables were therefore considered out of scope.

4.3 Application risk variables wrap up

- One risk variable category was left out of scope: basic incontrollable application risk

Three risk variable categories were discussed:

- *Basic control* refers here to basic controllable application risk factors: Control in the application environment
- *Residual control* refers here to residual controllable application risk factors: Compliance to the application environment
- *Expectations* refers here to residual incontrollable application risks; the exceeding application complexity expectations.

4.4 Collaboration risk variables

The second of the two large container boxes are Collaboration risk variables. This section will describe the logic behind this container box and describe the individual risk variables that are included in our research.

4.4.1 Rationale behind Collaboration risk

Despite the risks of software development that are present regardless of an onshore or offshore situation, the real distinction between onshore and offshore outsourcing lies in the collaboration factor. The concept that is discussed mostly relating to this distributedness in literature is that of virtual teams. Virtual teams are teams that operate across time, geographical locations and organizational boundaries linked by communication technologies (Casey et al., 2006). This definition fits the distributed software development teams in this study. It is generally assumed that virtual team performance is inhibited more than in collocated teams (Conchuir et al., 2006; Sakthivel, 2005; Casey et al., 2006).

An important contribution that is embraced, is made by Dixon and Panteli (2007), who do not speak of (static) virtual teams, but rather a *level* of virtuality in teams; resembling the (level of) discontinuity in geography, time, organization work practices and technology. Making the concept of virtual teams more dynamic, enables the opportunity to relate specific actions to obtain a decreased level of virtuality in the team, resembling co presence to a bigger extent.

Carmel (1999, 2005) presents global distributed software development as a centrifugal force that propels things outwards from an ideal situation, based on this virtuality and lack of physical interaction. These forces should be counterbalanced by centripetal mitigation forces (Prikladnicki et al., 2003; Carmel, 1999; Carmel et al., 2001), which are mainly referring to standardization and collaborative tools.

The original intent of this thesis was to present the factors influencing this level of virtuality in a team with the notion of social capital. "*Social capital refers to the sum of the actual and potential resources embedded within, available, through, and derived from the network of relationships possessed by an individual or social unit"*(Nahapiet and Ghoshal., 1998). It can be described in three dimensions:

- Structural: Members of the network can find each other; the facilitation of access to people and resources (Willcocks et al., 2006; Nahapiet et al., 1998; Jashapara, 2004).
- Relational: Refers to the sense of trust and motivation to be developed through the network ties, creating a sense of common motivation, purpose and benefit. Some related concepts are trust, norms, obligations and identification. (Willcocks et al., 2006; Nahapiet, 1998, Jashapara, 2004).
- Cognitive: Refers to the common understanding of interactions and work to be done. Related concepts: Shared codes and narratives (Willcocks et al., 2006; Nahapiet, 1998; Jashapara, 2004).

The reason to abandon the idea of departing from the theoretical lens of social capital is that the categorization cannot easily be generalized to the application risk variables. The actual categorization of risk variables that was used is very similar to that of the three different dimensions of social capital though. Basic and residual controllable risks refer to the structural dimension. Basic incontrollable risk refers mostly to the relational dimension, representing inherent trust and motivational distances in the virtual team. The residual incontrollable risk refers to the common understanding of work and interactions in the virtual team. The distinction in the different dimensions of social capital was embraced in the survey design, but from this point forward, the more generic categorization of risks described in section 4.2.2 is used, which is illustrated in figure 9 below.

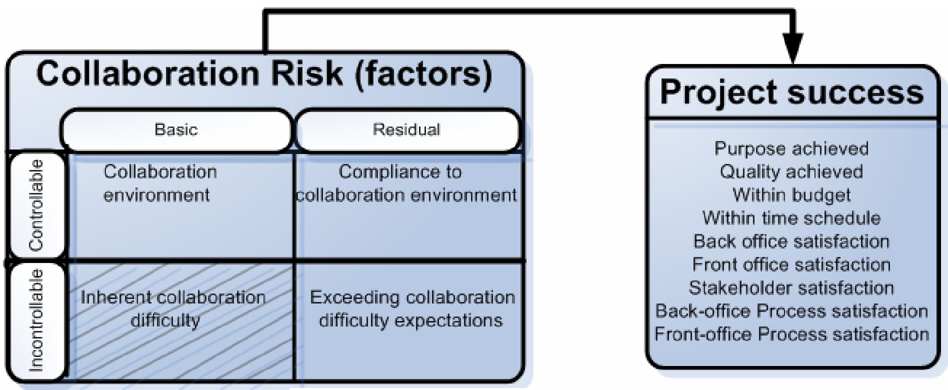


Figure 9: Collaboration risk (factors)

4.4.2 Basic controllable collaboration risk: Collaboration Environment

The first categorization of collaboration risks refer to the Collaborative environment in which the project was performed. The following tables (18-22) below summarize the basic controllable collaboration risk factors used in this research.

Project Management
A widely cited risk in any project is related to project management. Project management is the application of knowledge, abilities and techniques to plan activities that can reach the needs and expectations of all stakeholders involved in a project (Prikladnicki et al., 2003).

Besides the process maturity addressed in the previous paragraph, Dibbern et al. (2006) state that the CMMI processes (CMMI, 2006) areas do not fully grasp the total amount of project management activities. Erickson et al. (2006) found evidence that specific Project management capabilities were related to more project effectiveness. The relevance of these capabilities seems to have an exacerbated influence in a distributed context (Prikladnicki et al., 2003). IT project management maturity is described explicitly as a success factor in Carmel et al. (2005).

In this survey, we will not go into great detail with regard to the specific kind of methodologies (e.g. Prince 2 or PMBOK), but rather focus on similarity of maturity in the Project management method. Table 18 summarizes the Project management factors.

These factors solely focus on the basic point of departure; the compliance to project management method is not of importance here. This is relevant in the following subparagraph.

Table 18 Project Management

Risk factor	Description	Source(s)
Project Management maturity standard in front-office	Measurement of the maturity of the Project Management method used in the front-office	(Keil, Cule, Lyytinen and Schmidt, 1998) (Nidumolu, 1996) (Aspray et al., 2006) (Prikladnicki et al., 2003) (Prikladnicki et al., 2004)
Project Management maturity standard in back-office	Measurement of the maturity of the Project Management method used in the back-office	(Keil et al., 1998) (Nidumolu, 1996) (Aspray et al., 2006) (Prikladnicki et al., 2003) (Prikladnicki et al., 2004)
Differences between front-office and back-office method of Project Management.	Measurement of the differences between front-office and back-office in conducting Project Management.	(Feiman, 2005)

Experience
The following three factors in table 19 represent the working experience in both teams. Working experience with other side team members has been described to be an important success factor (Prikladnicki et al., 2006; Dibbern et al., 2006; Sakthivel, 2005; Carmel et al., 2005). To prevent misunderstandings, it is good to clarify that these risk factors do not refer to experience in the domain or managing the offshore contract like addressed in (Aubert et al., 1998; Gopal et al.,

2002; Prikladnicki et al., 2003). Instead, these factors solely refer to the experience of working over distance in a virtual team.

The latter risk factor in the table below refers not only to previous working experience, but also to previous experience in working with individuals. Expert interviews highlighted this to be an important success-factor. In one of the case studies described in Kotlarsky (2005), having this kind of a relation of knowing each other reduced the chance of miscommunications and conflicts.

Table 19 Working experience

Risk factor	Description	Source(s)
Front-office working experience	Measurement of the extent by which front-office team members had experience in working with a back-office team.	(Prikladnicki et al., 2006) (Sakthivel, 2005)
Back-office working experience	Measurement of the extent by which back-office team members had experience in working with a front-office team.	(Dibbern et al., 2006) (Sakthivel, 2005)
Individual working experience	Measurement of the extent by which team members had worked together as individuals between the back-office and front-office team.	(Kotlarsky, 2005)

Collaboration tools

Another combination of factors refers to collaboration tools; the way in which team members were able to communicate through sophisticated tools.

Assuming that asynchronous communication tools are at hand, the focus here lies on the availability of high level synchronous collaboration tools. Virtual teams with low cohesion require synchronous communication tools in order to build trust and relationships (Sakthivel, 2005). Using solely asynchronous communication tools creates potential misunderstandings (Prikladnicki et al., 2003) that might be prevented using synchronous communication tools (Carmel et al., 2001). The COCOMO II model also includes a “multisite” cost driver in its model, representing different level of distributed communication options (Boehm et al., 2000b). Since we were merely interested in the overall level of sophistication, the risk factor was not split in the survey design.

Table 20 Collaboration tools

Risk factor	Description	Source(s)
Synchronous communication tools	Measurement of the extent by which high level tools; video-conferencing options, conference calls, Internet Relay Chat synchronous communication tools were available	(Boehm et al. 2000b) (Evaristo et al., 2004) (Prikladnicki et al., 2003) (Carmel et al., 2001) (Sakthivel, 2005)

Application documentation

Similar to structuredness of the existing application factor described earlier, these factors deal with the pure access to relevant documentations. Not only documentation at the technical application level, but also on the level of access to contextual application knowledge.

Prikladnicki et al. (2006) consider the lack of documentation available for the back-office team an important risk factor. Omissions in documentation can cause great problems in application development projects (Herbsleb et al., 1999).

Mirani (2007) also stresses the necessity for relevant application documentation, but includes access to relevant contextual documentation. In highly task-specific development projects, like for example an insurance application, access to technical application information might not be enough. Governmental regulations or other country or organization specific contextual knowledge should be available for the back-office (Feiman, 2005; Dibbern et al., 2006).

Table 21 Application documentation

Risk factor	Description	Source(s)
Access to technical application documentation	Measurement of the extent by which team members had access to accurate application documentation	(Herbsleb et al., 1999) (Prikladnicki et al., 2006) (Mirani, 2007) (Boehm et al., 2000b)
Access to contextual business knowledge documentation	Measurement of the extent by which team members had access to accurate contextual business knowledge to be captured in the application.	(Mirani, 2007) (Feiman, 2005) (Dibbern et al., 2006)

Team set-up

The final two factors taken into consideration in the basic controllable category are related to the team set-up.

The first one is making use of liaisons, which are defined as a representative of client and supplier working at the site of their counterpart (Harmsen, Brand, Hillegersberg and Aydin, 2007). Working with liaisons has often been described as a best-practice (Carmel et al., 2001; Kobitzsch et al., 2001; Kotlarsky, 2005; Herbsleb et al., 1999) to a window of access to the offshore team (Mirani, 2007). Liaisons can be located permanently for the duration of the project, but also in specific phases of the project (Cusick et al., 2006). In an offshore context, cultural liaisons can bridge the gap between sites and prevent miscommunications based on cultural differences (Layman et al., 2006). In the survey, the extent by which teams made use of liaisons is brought forward.

Another aspect is predominantly related to the expert opinions. Two of the project managers interviewed mentioned a collaborative team training that was performed prior to the project kick-off and illustrated its effectiveness. A similar effect has been described in a case study by Prikladnicki et al. (2003, 2004).

Table 22 Team set-up

Risk factor	Description	Source(s)
Liaisons	Measurement of the extent by which the team made use of liaisons.	(Harmsen et al., 2007) (Carmel et al., 2001) (Kobitzsch et al., 2001) (Kotlarsky, 2005) (Herbsleb et al., 1999) (Mirani, 2007) (Cusick et al., 2006) (Layman et al., 2006)
Team Training	Measurement of the extent by which team members were trained to be aware of potential collaboration pitfalls	(Prikladnicki et al. 2003) (Prikladnicki et al. 2004)

4.4.3 Residual controllable collaboration risk: Compliance

A very small category of risk factors in this research are the residual controllable collaboration risks. They are represented in tables 23 and 24. Similar to paragraph 4.3.3, these factors do not represent different factors, but merely different dimensions of the risk factors introduced earlier. This is the reason that the source column is left out.

Table 23 Compliance project management method

Risk factor	Description
Project Management method compliance front-office	Measurement of compliance to the project management method in the front-office.
Project Management method compliance back-office	Measurement of compliance to the project management method in the back-office.

Use of communication tools
This factor refers to the sufficiency in use of synchronous communication tools. The consistency factor that was added as separate risk factor with application engineering tool support is not asked, since this is less relevant in the context of communication tools.

Table 24 Sufficient use synchronous communication tools

Risk factor	Description
Sufficient use of synchronous communication tools	Measurement of sufficient use (whenever necessary) of the synchronous communication tools.

4.4.4 Basic incontrollable collaboration risk: Inherent collaboration difficulty

Also with collaboration risk, the basic incontrollable type of risk variables is left out of scope. These kinds of factors would according to our risk typology be the risk factors that are inherent and constitute some kind of basic trust or motivational values of team members. These risk factors would be the basic point of departure for the incontrollable, implicit collaborative environment. There are some tangible basic collaboration risk factors like: prior experience, time-zone differences and to some extent even cultural distances using the five cultural dimensions of Hofstede (1996). These cannot really be depicted as "*sure*" factors influencing trust or team motivation on an assumed linear scale though. It thus seems nearly impossible to obtain a standardized measurement of basic incontrollable risks.

For this reason, we decided also to look at the effects (residuals) of risks with an incontrollable origin again by linking the risks to the original expectations that were present in the team.

4.4.5 Residual incontrollable collaboration risk: Deviating expectations

The residual incontrollable risks are thus the residual effects measured in deviating expectations. They are categorized and discussed in tables 25-28 below.

Relational factors
The first category consists of relational risks between the front-office and the back-office. Loss of Team cohesion is referred to by Carmel (1999) as a consequence of the centrifugal forces of

distributed software development (Prikladnicki et al., 2003). Substantial evidence is given that more distributedness; virtuality, leads to lessoned team cohesion (Herbsleb and Mockus., 2003). This can on its turn hinder trust and spread of information between front-office and back-office (Conchuir et al., 2006). Boehm et al. (2000b) take on a more generic concept of team cohesion as a cost driver in his COCOMO II model.

Another factor is the willingness or motivation to share information (Cramton, 2001; Ramasubbu et al., 2005). This factor is an explicit part of team cohesion in the COCOMO II model (Boehm et al., 2000b) and is related to an even more generic concept: trust in Kotlarsky (2005) and Herbsleb et al. (1999).

There might also be a motivational issue with regard to loss of intellectual property in strategic applications. Lewin et al. (2006) state that some companies perceive this to be a relevant risk, but it is perceived higher when companies have less offshoring experience. Harmsen et al. (2007) mention intellectual property to be a concern in strategic operations. It was included in the survey.

The fourth relation risk is related to pro-activity (Mirani, 2007). A lack of pro-activity in the back-office is on of the most widely mentioned issue addressed in the expert interviews with Dutch project managers to be an important risk factor. Requirements are followed rigidly without dispute, representing a lack of ability/responsibility to actively participate in handling project issues (Dibbern et al., 2006). The risk factor included in the research, is solely based on *back-office* pro-activity, since it was only addressed by Dutch Project managers and appears to be a "cultural" characteristic of an Indian back-office (Carmel et al., 2005). This will be considered when discussing external validity of the research.

Another risk considered here is speed. Herbsleb et al. (2003) describe the large effect global software development can have on development speed. It takes longer for issues to be dealt with, exacerbated by time differences between front-office and back-office.

The final risk: the important effect of proper collaboration skills in software development projects in general are addresses in Jiang et al. (2000).

Table 25: relational risks

Risk	Description	Source(s)
Team cohesion	Measurement of the extent by which the team members felt part of one team in comparison to what was expected.	(Conchuir et al., 2006) (Prikladnicki et al., 2003) (Herbsleb et al., 2003) (Carmel, 1999) (Boehm et al., 2000b)
Willingness to accommodate objectives	Measurement of the extent by which team members were willing to accommodate other side team member objectives	(Boehm et al., 2000b) (Cramton, 2001) (Ramasubba et al., 2005)
Intellectual property concerns	Measurement of the extent of reluctance in sharing strategic application information compared to what was expected.	(Lewin et al., 2006) (Harmsen et al., 2007)
Pro-activity in the back-office	Measurement of the extent by which back-office team members acted less proactive than was expected.	(Carmel et al., 2005) (Mirani, 2007) (Dibbern et al., 2006)

Speed of dealing with issues	Measurement of the extent by which the speed of dealing with issues between front-and back office was larger compared to what was expected	(Herbsleb et al., 2003)
Collaboration skills	Measurement of the extent by which collaboration skills in the other side team were according to expectations.	(Jiang et al., 2000)

Common understanding
The next three risks relate to a degree of common understanding between front-office and back-office. These factors partly represent the consequences of access to relevant task specific knowledge in the front office, but there is probably an omnipresent implicit risk of lack of understanding.

The lack of common understanding about work to be done was split in three. The first one refers to the underlying business processes to be captured in the application (Currie, 2003; Dibbern et al., 2006), which are owned by the front-office client organization. The second refers to contextual application knowledge (Mirani, 2007; Balaji et al., 2005) like for example governmental regulations, not owned by the outsourcing organization, but assumed known from the perspective of a specific region. The third refers to the technical programming work, which reflects the potential consequence of a lack of application documentation.

Table 26 Common understanding

Risk	Description	Source(s)
Front-office business processes understanding	Measurement of the extent by which there were more efforts than expected necessary to obtain common understanding about front-office business processes	(Herbsleb et al., 1999) (Currie, 2003) (Dibbern et al., 2006)
Contextual application knowledge understanding	Measurement of the extent by which there were more efforts than expected necessary to obtain common understanding about contextual application knowledge.	(Mirani, 2007) (Feiman, 2005) (Dibbern et al., 2006)
Programming work understanding	Measurement of the extent by which there were more efforts than expected necessary to obtain common understanding about the programming work.	(Herbsleb et al., 1999) (Prikladnicki et al., 2006)

Clarity
The clarity risk is highly related to the previous, but is more about the clarity about high level concepts like team roles and objectives.

A lack of clear role responsibilities has statistically been shown (Jiang et al., 2000) and described (Erickson et al., 2006; Ramasubbu et al., 2005; Balaji et al., 2005) to be a risk in software development projects. The underlying reason is confusion about who is performing which tasks and can be accounted for success or failure of that task. Project managers in the interviews mentioned an even exacerbated effect in virtual teams, due to a lack of coordination.

Two other risks relate to objectives. Ramasubbu et al. (2005) place "shared business goals" in a staged framework relating to collaboration readiness for distributed development. The business goals should be reflected in the new application. We thus translated this factor to application objective clarity. The other factor relates to consistency in individual team member objectives. Baht et al., (2006) and Casey et al. (2006) describe the importance of having similar objectives in the team. Also the expert interviews confirmed that consistency in team member's objectives is an important risk to be considered in virtual teams.

Table 27 Clarity risks

Risk	Description	Source(s)
Role responsibilities	Measurement of the extent by which there were unforeseen efforts necessary to clarity role team member role responsibilities	(Jiang et al., 2000) (Erickson et al., 2006) (Ramasubbu et al., 2005) (Balaji et al., 2005)
Application objective clarity	Measurement of the extent by which there were unforeseen efforts necessary to clarify the application objective	(Ramasubbu et al., 2005)
Team member objective consistency	Measurement of the extent by which consistency in individual team member objectives was according to expectations	(Bhat et al., 2006)

Cultural

By far the most widely cited risk of offshore outsourcing are put down in the container risk factor of culture (Carmel et al., 2005; Willcocks et al., 2006; Beulen et al., 2005; Layman et al., 2006; Lewin et al., 2006; Khsetri, 2007; Herbsleb et al., 1999; Carmel et al., 2001; Kobitzsch et al., 2001; Dibbern et al., 2006; Evaristo et al., 2004; Bhat et al., 2006); to really just be naming a few. It seems that scholars cannot neglect or get past cultural risks, but find difficulties in really grasping the concept. An option would be to operationalize the 5 dimensions of Hofstede (1996) in the context of offshore software development, but like Beulen et al. (2005) address: it is very difficult to define or manage cultural issues in global software development.

Our intent here is not to claim that we have found the strategy to grasp this. The cultural category described here is merely the result of an attempt to break different risk factors in a somewhat coherent piece. We do attempt to highlight some factors that are at the very least related to cultural factors and have been described in isolation as risk factors in literature. These risks were also rated highly relevant by expert interviews.

The first one is related to native language differences. Most front-offices and back-offices are able to speak the same language (mostly English). However, it appears that not having the same native language and the through delicacies in expressing oneself that goes along with it, can be a cause for problems (Khsetri, 2007; Bhat et al., 2006; Casey et al., 2006; Prikladnicki et al., 2003; Kobitzsch et al., 2001). An experiment performed between American MIT students and students from Mexico revealed similar problems in performing distributed tasks (Favela et al., 2001). Carmel et al. (2001) and Kshetri (2007) describe having the same native languages to be an advantage in distributed teams.

A similar risk relates to communication customs. Carmel et al. (2005) describe a number of distinctive elements of communicating in different countries which might cause confusion in a virtual team. Indian employees tend to be over positive and say yes as a form of politeness, where a Western employee might have said no. The other side of the coin is that Western employees tend to very dominant in interactions which might inhibit the foundation for an open debate about project issues. Casey et al. (2006) address the need to understand communication protocols in culturally distributed environments. Kotlarsky (2005) recommends using established communication protocols with ground rules in global distributed software development. A number of other scholars address different communication customs and associated problems between the lines of their work, hidden under the container factor of culture. Without an exception, all project manager in the expert interviews addressed communication customs to be a relevant risk for confusion in projects.

Another risk relates to misaligned ideas about the expected precision of project deliverables, like for example specific requirements. This factor is highly related to working standards like addressed in the CMMI (CMMI, 2006). It is indirectly mentioned in Carmel et al. (2005). Client organizations that have low capability levels often have an over reliance on back-office capabilities (Sabherwal, 2003), not understanding that the project output is dependent on the input.

The final risk taken in consideration is hierarchical differences. This factor, that loosely translated resembles the *power distance* dimension addressed in Hofstede's (1996) cultural dimensions, can trouble the cooperation between back-office and front-office team (Kotlarsky, 2005; Dibbern et al., 2006).

Table 28 Cultural risks

Risk	Description	Source(s)
Native language understanding	Measurement of the extent by which there were unforeseen efforts necessary due to native language understanding difficulties.	(Khsetri, 2007) (Bhat et al., 2006) (Casey et al., 2006) (Prikladnicki et al., 2003) (Kobitzsch et al., 2001) (Favela et al., 2001) (Carmel et al., 2001) (Kshetri, 2007)
Different Communication customs	Measurement of the extent by which there were unforeseen efforts necessary due to different communication customs	(Carmel et al., 2005) (Casey et al., 2006) (Kotlarsky, 2005)
Misalignment regarding precision of deliverables	Measurement of the extent by which there were unforeseen efforts necessary due to misalignment ideas about required preciseness of project deliverables	(Carmel et al., 2005)
Hierarchical Approachability	Measurement of the extent by which there were unforeseen efforts necessary due to hierarchical approachability difficulties.	(Hofstede, 1996) (Kotlarsky, 2005) (Dibbern et al., 2006)

4.4.6 Out of scope collaboration risk

The risk factors identified above are a subset of the risk factors that exist in total. Our intent is to get as much surface risk variables in the survey with much explaining power. However, our research efforts are focused on operational / tactical risk factors. Strategic factors like for example vendor choice (Carmel et al., 2005; Willcocks et al., 2006), geopolitical risks (Beulen et al., 2005; Carmel, 1999; Carmel et al., 2005; Feiman et al., 2005), job loss (Aspray et al., 2006; Stack and Downing, 2005), currency risks (Carmel et al., 2005) or infrastructural risks (Beulen et al., 2005;

Lewin et al., 2006) were taken of out scope. The specific project itself and the project success were taken as the central point of measurement.

Another out of scope topic to be discussed briefly is related to project governance. The governance of the project can be considered to be an operational / tactical factor, but is still left out for the large part. Although touched upon by risk factors like *clear role responsibilities* and the *use of liaisons*, we deliberately did not include different specific project roles in the research. Such an inclusion would indicate that besides different roles that were present in the project, this would also indicate in which part of in the project they were active, for how long, what their interactions were with other stakeholders etc. This would require much more questions than desired to acquire a high response rate. We are confident enough that the risk factors taken in consideration have enough distinctive power.

4.4 Collaboration risk variables wrap up

- One risk variable category was left out of scope: basic incontrollable collaboration risk

Three risk variable categories were discussed:

- *Basic control* refers here to basic controllable collaboration risk factors: Control in the collaborative environment
- *Residual control* refers here to residual controllable collaboration risk factors: Compliance to the collaborative environment
- *Expectations* refers here to residual incontrollable collaboration risks; the exceeding collaboration complexity expectations.

4.5 Controlling Project Factors

Some basic project characteristics that might have an effect on project success were asked. Before the actual risk part of the survey starts, a number of questions are related to obtain these controlling factors. They can roughly be distinguished in:

- Time zone differences
- Project size
- Distribution of work
- Perspective
- Development method used

Time zone differences

The first question is related to time zone differences. Some authors addressed this to be an important factor, since it frames the overlap in working hours to a suboptimal collaborative window (Beulen et al., 2005; Herbsleb et al., 1999; Kobitzsch et al., 2001).

Table 29 Time zone differences

Factor	Description
Time zone differences	Measurement of the time zone difference between front-office and back-office (hours)

Project size

The mere size of the project can also be an important variable that affects project success. It is mentioned in Barki et al. (1993), Carmel et al. (2005) Gopal et al. (2002) that the size of the work (project) has a negative impact on aspects of success. There are more possibilities were things may go wrong. There are many ways of indicating project size. A few of them were included in the survey (see table 30).

Table 30 Project size

Characteristic	Description
Functional points	Measurement of the project application size in functional points
Billed-man hours	Measurement of the total amount of billed man-hours in the project
Modules	Measurement of the number of new modules developed
Budget	Measurement of the used project budget? (in K-euro's)
Duration	Measurement of the duration of the project (in months)
Members front-office	Measurement of the amount of team members in the front-office
Members back-office	Measurement of the amount of team members in the back-office?

Distribution of work

The back-office and front-office team have different responsibilities of work. One of the most common divisions of onshore/offshore distribution in application development is performing the requirements and integration as well as parts of the design and testing onshore. The coding and the other parts of the design and testing are often done offshore (Carmel et al., 2005). The distribution of work is taken into account. A rough distinction is made in the five phases of software development presented earlier. Preliminary design and detailed design are taken together in "Design".

Table 31 Distribution of work

Characteristic	Description
Requirements work	Measurement of the percentage of requirements work done in the back-office?
Design work	Measurement of the percentage of design work done in the back-office?
Coding work	Measurement of the percentage of coding work done in the back-office?
Testing work	Measurement of the percentage of testing work done in the back-office?
Deployment work	Measurement of the percentage of deployment work done in the back-office?

Perspective

A characteristic on the respondent's level is the perspective by which the survey was filled out. Also the perspective of the project was asked for.

Table 32 Perspective

Characteristic	Description	Answering options
Country	Country/region origin of filling in survey	{Project manager, Project team lead, Delivery manager, Other}
Role	Perspective role of filling in survey	{India, Netherlands, Western Europe(not Netherlands), Other}
Type of project	Question indicating the type of project	{Rightshore, Direct Offshore}

Development method

The final project characteristic that was asked was the type of development method used. Agile development methods like RUP and DSDM use an iterative approach of developing (Sakthivel, 2005) in which the developed system goes to several trials with users and modifications before completion. A waterfall approach follows the five phases of software development rigidly. A combination of the two types of developing (for different parts of the application) can be used according to the experts we interviewed.

Table 33 Development method

Characteristic	Description	Answering options
Type of development method used	Question of the type of development method that was used.	{Agile, Waterfall, Both agile and waterfall, non existent/ad hoc, don't know}

4.6 Risks in the Rightshore Assessment Study

The initial trigger of this study was the Rightshore Assessment Study; an offshore readiness assessment of application and organization potential to offshore outsource applications. Placing the location of the RAS scope in the research framework leads is revealed in figure 10. The scope of our research is shown in figure 11, represented by the blue overlays. RAS fills the gap of basic incontrollable risks. The RAS-assessment is performed prior to offshore outsourcing and in cooperation with the outsourcing organization. This enables an in depth analysis of the organization and its applications. The inherent complexity (basic incontrollable risk), left out of scope in our research is within scope of RAS.

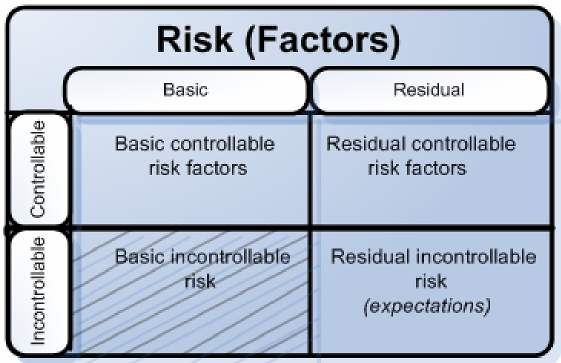


Figure 10: Risk (factors) in RAS

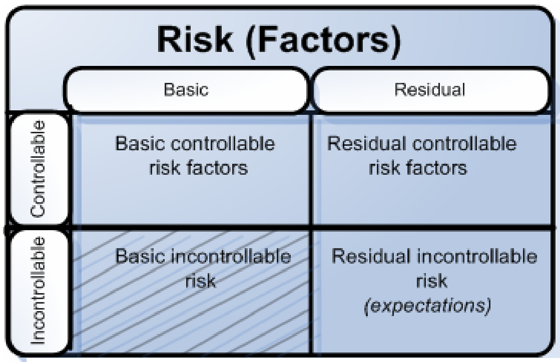


Figure 11: Risk (factors) in this research

RAS is developed in practice, based on consulting experiences. A large number of application assessments have already been performed at client IT organizations. There are no track records of projects that followed from a RAS-assessment. This means it is for the moment impossible to make an informed decision of prioritization in mitigation actions that the outsourcing IT organization can take. The constructs of RAS resemble to some extent the constructs of this research. We deliberately choose however to start from a scientific literature point of view, which is why the constructs do not align fully. The results of different risk factor impacts on project success can therefore only loosely be translated into weighing factors in the RAS-model.

A mapping was made to fill in research variables in the RAS subdimensions (See appendix 12.3). Based on relative occurrence and impact of the risk factors, the RAS assessment can be adapted to the offshore application development derivative: **RAS'** by assigning these new weights. The strategy, in which these new weighing factors are derived, is explained in Chapter 6: Research Method.

4.7 Operationalized risk framework

Figure 12 reveals the operationalized risk categories that follow from this chapter of research constructs. All risk variable categories represent a distinctive origin and are believed to have an effect on project success.

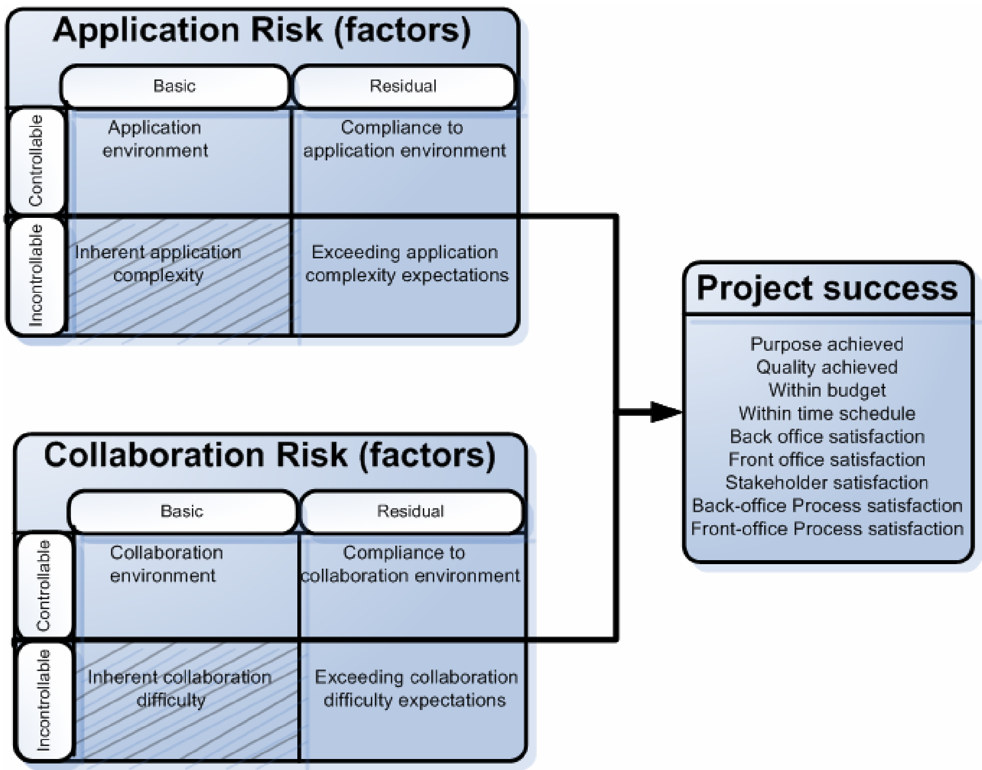


Figure 12: Operationalized risk categories

5 Research framework and hypotheses:

At this point the sub questions regarding scope and research constructs have been answered. This brief chapter places the subdivided hypotheses in the conceptual model.

5.1 RQ1: Categorized risk impact on project success

1. Is there an impact of controllable and incontrollable application and collaboration risk factor occurrence on project success in offshore outsourcing application development?

Our hypothesis is that there is a significant impact of all risk variable categories on project success. Roughly said, these hypotheses test the rationale behind the Rightshore Assessment Study which is based on capability (basic and residual control) and complexity (residual expectations). Since the first research question consists of a number of different elements, they are split up in six sub hypotheses.

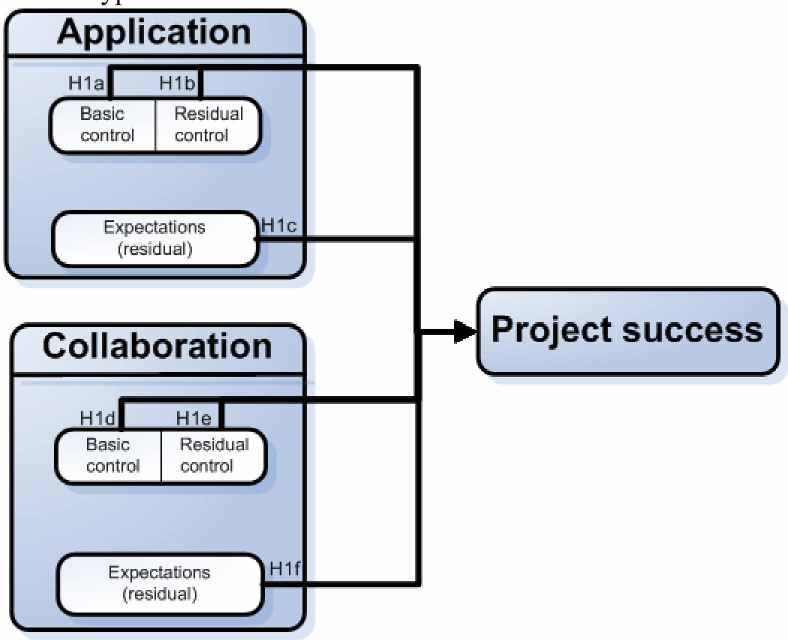


Figure 13: Research question 1 hypotheses

***H1*: There is a significant impact of controllable and incontrollable application and collaboration risk factor occurrence on project success.**

H1a: There is a significant impact of basic controllable application risk factor occurrence on project success.

H1b: There is a significant impact of residual controllable application risk factor occurrence on project success.

H1c: There is a significant impact of residual incontrollable application risk occurrence on project success.

H1d: There is a significant impact of basic controllable collaboration risk factor occurrence on project success.

H1e: There is a significant impact of residual controllable collaboration risk factor occurrence on project success.

H1f: There is a significant impact of residual incontrollable collaboration risk occurrence on project success.

5.2 RQ2: Risk dependencies

2. Does controllable risk have an impact on residual incontrollable risk occurrence?

The second explorative research question described here, tests the effect that a controlled environment has on residual incontrollable risk occurrence. The incontrollable risks were only measured by their residual effects which were reflected in their deviation from expectations. It is hypothesized that a stable and controlled environment has a positive (stabilizing) effect on exceeding expectations from the incontrollable risks. This would thus indicate the potential of controllable risk factors as indirect risk mitigation options to reduce the negative impact of incontrollable risks on project success.

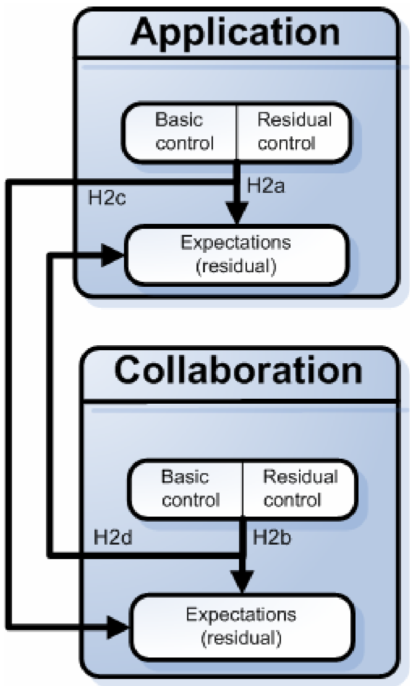


Figure 14: Research question 2 hypotheses

H2: **Controllable risk factor occurrence has a positive impact on residual incontrollable risk occurrence.**

H2a: Controllable application risk factor occurrence has a positive impact on incontrollable application risk occurrence

H2b: Controllable collaboration risk factor occurrence has a positive impact on incontrollable collaboration risk occurrence

H2c: Controllable application risk factor occurrence has a positive impact on incontrollable collaboration risk occurrence

H2d: Controllable collaboration risk factor occurrence has a positive impact on incontrollable application risk occurrence.

5.3 RQ3: Offshore readiness assessment improvement

3. How can an offshore readiness assessment be improved to include the results of the previous questions?

The third and final research question provides a description of how the results of the previous two research questions can be integrated in the Rightshore Assessment Study. This is not an empirical question and there is no predetermined hypothesis that is tested here.

6 Research Method

This chapter describes the different steps taken in this research from the start until the description of the results in the next chapter. In fact most of the research constructs have already been described in chapter four. Because the centric measurement instrument is an online survey, first, a description of the rationale behind survey research is given.

6.1 Rationale behind survey research

The reason to conduct survey research is twofold: First, like described in the scientific relevance section, there is little empirical evidence with regard to (delicacy in) offshore outsourcing risk factors. A survey approach is well suited to lead to this kind of contribution. Second, the relatively narrow scope: distributed offshore application development project provides the means to ask many specific characteristic questions to respondents. A survey tool is suited for systematic questioning or observation of data; mostly on a large amount of characteristics (Baarda and Goede, 2001).

Next to the rationale behind survey research itself, another issue is rationale behind a project-focused approach. Our research takes specific projects as input in the survey to come up with risk occurrences and impacts. This was done after consideration of other alternatives like asking for risk factor perceptions and vignette studies. The reason for the project-centric approach was chosen because asking for specific project attributes or risk occurrence in a project is more accurate than asking for risk perceptions that can easily be distorted (Pfleeger, 2000). More-over, Pfleeger (2000) also proposes to base software and system related risk distribution upon historical data, not just on expert judgment. This approach was taken on in this study. A vignette study alternative was also considered, but rejected due to a lower added value than the project-centric approach. A combination of the two would make the survey to large and potentially suppress response rates.

6.2 Research steps

This research consisted of seven main (partially overlapping) steps.

- Literature research: Derive risk factors
- Expert interviews: Validate and reflect upon found risk factors
- Survey construction: The integration of the risk factors in a methodologically solid survey
- Approaching respondents: Contacting respondents to fill out the survey
- Gathering data: Gathering data and reporting back to respondents
- Data preparation: Controlling and recoding of data to prepare it for analysis.
- Translation of outcomes: Translation of outcomes to the Rightshore Assessment Study

6.2.1 Literature review

Being the primary trigger for this research, the constructs behind the Rightshore Assessment Study were the first main reason for canalizing our literature research efforts. The scope of this assessment is both Application Development and Application Maintenance. Based on preliminary studies of literature (Carmel et al., 2005; Beulen et al., 2005; Erickson et al., 2006) revealing clear differences in project oriented outsourcing (development) and continuing service provision (maintenance), we scoped searching efforts to application development outsourcing.

Consulted and referenced literature sources in this thesis come almost without an exception from validated and referenced sources in journal publications, conference proceedings or edited books (92%). We decided to also include some studies more related to practice, like Gartner and Capgemini's own internal resources for example.

The strategy for finding relevant sources was twofold. First a broad literature search was conducted in scientific literature search engines. Queries like "Global Software Development", "Offshore outsourcing Application Development" and all possible combinations were entered. Reviewing abstracts, lead to the choice of further studying the paper. Next, based on this first wave of papers, a snowball approach in which relevant paper lead to other relevant papers was followed.

An important element of the literature research was to come up with a generic framework to place risk variables in. The final choice in this thesis was based on a more high-level generic approach, splitting up risks in controllable – incontrollable and basic – residual categories. Although the risk factors themselves did not change, the way in which they were presented did vary between this thesis and the survey. In the survey they were presented according to the taxonomy based risk identification (Carr et al., 1993) and social capital (Willcocks et al., 2006). It was however a deliberate choice to present the risk variables according to content-structure in the actual survey, since this is more in line with respondents experience. More-over, by presenting risks in terms of "controllable" and "incontrollable", could lead to biased filling out of the survey.

The end-deliverable of this step was a preliminary list of risk variables categorized based on the taxonomy based risk identification (Carr et al., 1993) and the three dimensions of social capital (Willcocks et al., 2006). This list was the primary input for the expert interviews.

6.2.2 Expert interviews

After obtaining our first list of risk factors, a number of eleven semi-structured interviews were conducted. All interviews, except for the one through a conference call, were recorded with a voice recorder and worked out afterwards.

Four out of the eleven interviews were held with Project Managers from the Netherlands. They all had experience in managing one or more distributed application development projects. Two interviews were conducted with project managers from India which were also experienced in managing such projects. The first interview was held over a conference call; the other was held face to face, since this person was coincidentally in the Netherlands at the time. Another two interviews were held with two Dutch risk managers and two Dutch sourcing consultants from Capgemini that advice organizations with regard to potential sourcing arrangements. The final interview was held with the Dutch Rightshore director who is responsible for all Rightshore projects and has knowledge on risks in many current and performed projects. The reason for multiple stakeholder groups is to obtain a wide perspective of opinions with regard to distributed application development risks.

A few days before the meeting, an agenda was send to the experts. Also the list with risk variables and a short description of the research set-up was sent to give the expert some basic information to make the meeting more effective. The agenda had an approximate similar set-up over all respondents:

- Getting acquainted
- Short overview of goal and intended (practical) deliverables of research

- Risk factors according to expert
  - What are according to the expert the most important risk variables with regard to distributed collaboration?
  - What are according to the expert the most important risk variables with regard to the newly developed application?
- Risk variables in literature
  - Reflection upon risk variables found in literature
  - Mapping of risk variables from practice with risk variables found in literature
- Risk variables in context of survey research
  - Evaluation of approximate equal level of abstraction (granularity)
  - Evaluation of common understanding (possible biases)

The latter point (common understanding) is one of the most important aspects of good question design (Fowler, 1995). Respondents should understand a question in the same consistent way, according to the researcher's expectation of its meaning.

After every interview, the results were worked out in minutes of meeting. If the expert had supplements to the list of risk factors, this was cross-checked in a focused literature search to find similar supporting evidence of their claims. The list of risk factors was thus iteratively improved and used as input in following expert interviews. The resulting list has in fact already been introduced in chapter four. This list of risk (factors) combined with controlling project and respondent's information was used as the input for the survey construction.

6.2.3 Survey operationalization

Choice of survey tool

The first task before the actual implementation of questions in the survey was to choose a platform for the survey. The number of risk f variables was quite large which excluded suitability of distributing a survey through an excel file. The potential group of respondents was not that high, because the specific project type of interest (application development) is a very narrow topic. This calls for even more attention for a platform with a low participation threshold. The online survey seemed the best option.

After a brief evaluation of constructing one ourselves, we decided that the benefit of a significantly faster solution; acquiring a survey tool was the better choice. Some tools were considered; Netq from Netquestionairres was found the most suitable tool. Netquestionairres supported the type of questions that were required, was inexpensive to purchase and most importantly made very easy translations of data, including labels to SPSS and MS Excel outputs.

Answering categories

The choice for risk variable inclusion was thus made, but the way in which they were asked was not. After meetings with consultants and a scientific supervisor, it was agreed that the risk factor *occurrence* could best be asked in terms of statements. Similar approaches in similar studies have been performed (Lee, 2001; Loh and Venkatraman 1995; Barki et al., 1993). Respondents could agree to a certain extent with that statement based on a five-point Likert scale ranging from:

- Completely agree
- Mostly agree
- Partially agree
- Mostly disagree
- Completely disagree

All risk factors were thus translated in a statement to which respondents could respond.

With regard to the residual incontrollable risks, that were measured based on exceeding expectations, an extra point of consideration was taken. Not all questions were posed in the same direction. A question that could be asked was: *“ [...] was more complex than expected”*. The opposite of exceeding expectations (more complex) presents some methodological problems. This opposite would be: “*[...] was less complex than expected*”. However, the purpose of the question is solely to check exceeding expectations. Complexity according to expectations would not fully be agreed upon when the statement was posed as: “less complex”. We solved this problem by posing the inverse statements as: *“[...] was* ***according to or*** *less complex than expected”*.

The measurement of seven out of nine project success items was done according to the same five point Likert scales. Two items: time schedule and budget overrun were measured in a different way. Respondents were asked by which percentage the original budget and time schedule was exceeded. To avoid confusion, two examples were given of filling out. Also a mouse-over event was placed to foresee in an adaptation of time schedule/budget during the project based on new functionality requests of the client organization:

> “Note: When the original time schedule/budget was enlarged due to *increased functionality requests of the client organization*, please answer this question with respect to this adapted time schedule / budget”.

This only applied for new functionality requests, because in that case the original dedicated resources would not be accurate by definition. Conversely, this did not apply for schedule and budget overruns when the assignment remained the same, because these are the overruns that are caused by project risks.

Heuristics of survey construction

The design of the survey in statements was not the only consideration taken into account. Heuristics for coming up with good question design is described in Fowler (1995). Some important points were considered.

- *Objective of question should be clear and the answer needs to meet the objectives of the question* (Fowler, 1995).
    a. All respondents were given an introduction text in which the objective of the whole research was explained with a hyperlink to more background information.
    b. The modules of the survey were given an introduction text.
    c. The objective of the project controlling questions was to be able to control for project size, distribution of work and perspective. The answers met the objective of this purpose. The objective of the risk variable questions was to measure the occurrence of specific risk variables in the project. This was made clear to the respondents and the answers met the questions objective.

- *The question should be understood in a consistent way for all respondents and the researcher* (Fowler, 1995).
    a. As a part of the expert interviews, one element was to evaluate the risks on common understanding. Not only was asked for the expert’s opinion on the meaning of the risk factor, but also potential problems for understanding of other respondents was asked. As a result, some questions were specified to who they applied to (front-office

or back-office team). Other questions were given a mouse-over event to specify a concept used in the question.

- *Questions should be administered in a consistent way* (Fowler, 1995).
    a. The survey was distributed in the same language to all respondents (English)
    b. All questions were the same for all respondents.

- *Consistent communication of the kind of answer that is expected* (Fowler, 1995).
    a. The first section of the survey containing project controlling questions used specific triggers to make clear exactly what kind of answer was expected to the respondents.
    b. The second section of the survey, containing risk factors was implicitly clear. Likert scales with radio buttons were used. There was no cause for confusion here.

- *Respondents should be able to answer the questions* (Fowler, 1995).
    a. An initial check in the expert interviews was conducted to check if project managers would be able to fill out the questions of the survey. A deliberate choice was made not to make the risk variables to specific or technical. However, some questions related specifically to the front-office or back-office team. Respondents were informed in the introduction text how to deal with not being able to answer a question.
    b. In the first part of the survey this was even integrated with the question itself. In the second part of the survey every question had the answering category: “Don’t know / does not apply”. For the sake of the research it was not important whether the answer was unknown or not applicable.

- *Respondents should be willing to give correct and valid answers* (Fowler, 1995).
    a. Since this research had a project centric approach and the primary respondent was the project manager, this was a big issue. This problem was solved (partially) by stressing on multiple places the complete anonymity of the respondents filling out the survey. The actual anonymity was in fact guaranteed. There was no real way of tracking down responses to individuals or projects.
    b. The introduction text contained text fragments where the goal of the research was related to the big importance of filling in the survey as trustworthy as possible, again emphasizing anonymity. Also it was deliberately stressed that we were actually looking for risk variables in projects.

- *Ask for one question at a time* (Fowler, 1995).
    a. This point was initially not considered enough. It is very tempting to ask for two things at a time, but the problem is that it is not clear what the respondent is answering on if these two things might be mutually exclusive. Some changes were made to ensure that every question was related to one (aspect of a) risk factor, some deliberate exceptions excluded.

Respondents could change their inputs as long as they did not submit the final filled out form. They were also able to browse back and forth across survey modules.

Implementation and validation of the survey

After construction of the initial survey, it was send to supervisors. Based on their input some changes in colors and attractive appeal was made. The length of some questions was reduced by simplifying them and making use of new mouse-over events with further explaining concepts.

The survey was finally sent to two project managers that provided feedback. This did not result in major changes.

The final survey consisted of 19 questions related to demographic information and/or characteristics of the project and 81 risk factor variables. 9 questions related to the measurement of project success. The total amount of questions was 109.

6.2.4 Approaching respondents

When the choice was made for survey research, it was obvious that offshore application development project managers should in any case be part of the respondent's profile. A "Project team lead" and "delivery manager" that both have a similar broad overview of the project were added to this profile. Initially, sourcing consultants would also be a part of the respondents profile, but after the decision to scope efforts solely to completed offshore application development projects, this decision was revised.

The list of potential respondents was created in the following way:

- Direct: Based on the input of consultants at the Technology Advisory Services practice, a list of respondents with relevant experience was created.
- Indirect: Based on the expert interviews, leads to other potential respondents were checked.
- Respondent groups: Some sponsors were willing to approach large groups of potential respondents on our behalf.

Potential respondents were then approached in two ways in the second week of August 2007:

- Direct: An email was sent to all 63 potential respondents that were known by name and email address. In the email, a brief description of the research was given. Also benefits for the respondents were made clear. All respondents were promised the end-results of the research. Also a link to preliminary results after filling in the survey was described as triggering prospect. The email contained the hyperlink to the survey and a username and password. All potential respondents were approached individually, not in bulk. Primary reason was to tailor the email based on the name of the lead that led us to them and approach people individually by name to increase the chance of filling in the survey.
- Indirect: The sponsors of our research were sent the email described above. These sponsors forwarded our email to other potential respondents. It is unknown how many people were contacted in total. Even if known, this would give a distorted effect. Potential respondents approached indirectly, were approached predominantly in bulk and without accurate pre selection of their fit with the respondents profile. The required profile of the respondent was the first capitalized message in the introducing email.

A reminder email was sent in the last week of August 2007, requesting potential respondents again to fill out the survey, if they did not do that yet.

6.2.5 Gathering data

Data on projects was collected in the period between mid August 2007 and the first week of September 2007. During that period, 46 respondents filled out the survey completely which resembles a 44% response rate. Two respondents were removed due to an extensive amount of missing values. The average time of filling in the survey took approximately 20 minutes.

The data was exported on a regular basis and preliminary results were published on www.capgemini-resultaten.nl. This domain name was requested and approved by Capgemini

Utrecht to be a frame-forwarding site to a site with preliminary results. The link to this site was given only after completely filling out the survey to prevent biased responses.

Data until September 8, 2007 was taken in consideration. After this date, the survey was kept online for a couple of weeks, but the extra data coming from this extended period was not considered in the data analysis.

6.2.6 Data preparation

When the survey closed down, some revision was done to the data to prepare it for further analysis. Also to guarantee statistical legitimacy, some considerations were made with regard to the use of statistical techniques

Recoding
First, all data on the 81 risk variables were recoded in the same direction such that a value of 1 represented low risk occurrence and 5 a high risk occurrence. Seven of the nine items, supposedly elements of project success, were measured on the same 5-point Likert scale as the risk factors. Two items determining project success were measured by the percentage of budget and time schedule overrun. We recoded them by their own minimum and maximum values based on input of the 44 respondents. The lowest overrun; which was in fact under planned budget and time, was recoded as 1 and the highest overrun was recoded to 5. The intermediating values were recoded between 1 and 5. Cronbachs alpha was used to test the nine resulting items for internal consistency. It is generally assumed by Nunnally in (Bahli et al., 2005; Nidumolu, 1996) that a value of 0.7 is sufficient for internal consistency

Project characteristic effects
The project characteristic variables were checked for a relation on project success.

- size of the project
- distribution of work
- perspective of filling in the survey
- type of project
- development method used

The outcomes of these checks are dealt with in the next chapter. Data were also controlled for collinearity by removing one item in an extremely highly related correlation (correlation coefficient greater than 0.9). This would indicate that both risk factors measure the same thing. A control for outliers was also conducted.

The two main statistical techniques that were used to derive *impact* scores are factor analysis and linear regression. Main considerations with choosing and applying them are derived from Kachigan (1991). A factor analysis of risk factors in each of the categories was performed prior to testing the hypothesized relations with project success. The reason for a factor analysis is twofold: First, because the average number of risk factors in a specific category is quite large, it is difficult to distinguish which factors are related to each other and which are not. A factor analysis can capture this relatedness and at the same time distinguish between different (non related) clusters of risk factors within each category. Second, with a factor analysis, residual factor loadings (correlations with the factor) can be saved and used as input in a linear regression model to predict project success. The assumptions for both should be met in order to proceed with statistical analysis.

Factor analysis assumptions
The factor analysis requires variables from the interval or ratio level. However ordinal variables are also allowed when believed they do not seriously distort the underlying metric scaling (Garson, 2007). Normally distributed input variables are not a strict demand for factor analysis, which would be almost impossible when using Likert scales. Two tests that will be taken in consideration are the Kayser-Meyer-Olkin and Bartlett test of sphericity. The former indicates multicollinearity between input variables of the factor analysis. In general it is considered that a value bigger than 0.5 is required to interpret the factor solution (Garson, 2007). The latter: Bartlett's test of sphericitiy tests the null hypothesis that the input variables originate from an identity matrix. In other words, this test indicates if the variables are sufficiently correlated at all to meaningfully rearrange them in a (smaller) factor solution. Bartlett's test should result in a significance smaller than 0.05 ($p<0.05$). In this research, we performed a principal components (PCA), varimax rotated factor analysis. The few missing values were replaced with mean values. An Oblique rotation (oblimin) in which the factors are not orthogonally (uncorrelated by definition), was also considered and performed. This did not lead to dissimilar clustering of variables in factors, indicating robustness of the varimax factor solution. It was left out of the results to prevent redundancy of presented information. The varimax rotation was chosen for a more conceptual purpose of really "polarizing" risk factors due to the orthogonal character of the rotation.

Linear regression assumptions
A linear regression model is stricter than factor analysis with regard to underlying assumptions. Input variables should be interval or ratio (without concessions) and approximately distributed normally. After performing factor analysis, the residual factor loadings of all cases were saved as variables were automatically transformed in an interval scale. To test the resulting factors for normality, the Shapiro-Wilk test was used (Shapiro and Wilk, 1965). This test; often used in small samples sizes tests the null hypothesis that the data are distributed normally. The p-value should be larger than 0.05 ($p>0.05$).

6.2.7 Translation of outcomes to RAS weighing factors

The translation of outcomes is actually a methodological step that was taken after the results of the hypotheses 1-2 were known. The primary goal of this translation is to be able to feedback the results of our research to every day practice at Capgemini and come to a RAS derivative (RAS') based on the results. The translation was made rather loosely, according to three steps.

1. The risk variables representing parts of RAS were mapped in its twelve constructs.
2. A calculation was made based on the results of the Spearman correlations of risk variable with project success and was multiplied with the occurrence of the risk variable. Spearman Rho was used instead of Pearson due to the ordinal character of the risk variables (Urdan, 2001)
3. Finally the average was taken per RAS construct; This was recoded to a scale 0 to 1 to standardize the weights.

6.3 Reliability and Validity:

Before discussing the results of this study, some remarks are made with regard to reliability and validity of the research. A reliable research is a research that has similar results when performed again. Validity has two aspects. Internal validity of a research indicates that the research constructs measure what they are supposed to measure. External validity refers to the question whether the results of the research can be generalized beyond the boundaries of the research. This is addressed in the discussion chapter (Chapter 9).

6.3.1 Reliability

There are a number of levels in which the reliability of the research is guaranteed. First, almost all risk variables in the research have been derived from scientific literature, some based on case-study findings, some based on empirical evidence. Second, based on the expert interviews, these risk variables were cross-validated and mapped on the context of offshore application development. This was done by interviewing people from multiple perspectives. This gives the risk variables their second reliability foundation. Thirdly, surveys were made anonymous and preliminary results were only made available after filling in the survey. Based on a bottom-up approach of using real-life project experience for risks instead of perceptions on risk, another aspect of reliability is achieved. The "don't know" option in the survey prevented respondents from filling in information they did not possess. In studying data, efforts were put to prevent misguided results. Statistical assumptions were taken in consideration and different factor rotations were performed to control for solid solutions.

6.3.2 Internal validity

Also efforts were put in internal validity. First the questions were asked as straightforward as possible. All possible problems in common understanding of the questions were checked in the expert interviews. Here, the intent of all questions was mapped with that of potential respondents. If questions contained specific terms that required a definition, this was given through a mouse-over event in the survey. The existence of this mouse-over event was clearly described in the introduction of the survey. To conclude, questions were aimed at only one specific risk factor. Asking two questions in one; thereby not knowing what is actually being answered, was prevented. By taking on three different approaches, a concept described as data triangulation was performed (Jick, 1979). Data triangulation is defined as: the combination of methodologies in the study of the same phenomenon by Denzin N. in (Jick, 1979) and intends to boost the validity of the concepts under examination. By using a literature study to derive risk variables, expert interviews to validate them and guarantee common understanding and quantitative survey research to test, an important form of *between method* triangulation was realized (Jick, 1979). By creating factors based on own internal relatedness and checking internal consistency for project success, *within-method* triangulation was performed.

7 Results and interpretation

This chapter presents the results of analyses on the survey data. The paragraphs follow the order of the research questions, but first some data preparation issues are discussed.

7.1 Data preparation

Data considerations

A number of descriptive tests were run to check the data for strange values. Two respondents were removed due to a very large number of missing values on variables (more than 50%). Data on the remaining 44 projects and respondents was used for statistical analyses. On the individual variable level with regard to project characteristics (number of people in front-office and back-office team), some outliers were left out of consideration due to extreme values that distort the average distribution.

Project success internal consistency

The central aim of this research is to relate risk variables to project success. The nine items that determine project success were put in an internal consistency analysis by examining Cronbachs Alpha. The result was a sufficient value of 0.863. The average of the nine items was thus taken as a measurement for project success.

Collinearity reduction

Like addressed in the previous chapter, collinearity was reduced by checking the cross-tabulation of correlations between variables. When two variables had a Spearman r > 0.9, one of the variables (in the risk factor 2 column) was removed from analysis as illustrated in table 34 below.

Table 34 Collinearity

Category	Risk factor 1	Risk factor 2	r (n=44)
Basic controllable application	Application architecture standardization	Technical architecture standardization	0.904
Residual controllable application	Sufficient use testing tool support	Consistent use testing tool support	0.932
Residual incontrollable application	Functional detailedness of req. larger than expected	Functional precedentedness of req larger than expected	0.904

7.1 Data preparation findings

- 2 respondents removed leading to n=44
- No outliers found on a generic scale
- Process success internal consistency was sufficient and taken as dependent variable
- Collinearity removed in three places

7.2 Basic descriptives

Before testing the hypotheses, some basic descriptive information about survey responses and data considerations is given in this paragraph, starting with basic project and respondent characteristics.

7.2.1 Project and respondent characteristics

Table 35 illustrates basic information of respondents and characteristics of the project.

Table 35 Project and respondent characteristics

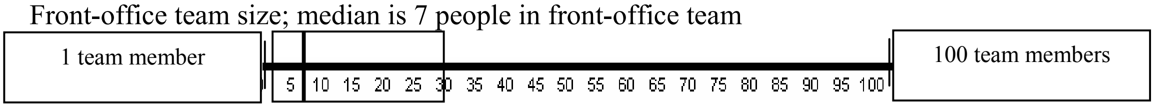
Variable	Value	Number
Perspective of filling in survey	- Netherlands	26
	- India	17
	- Other	1
	Total	44
Project role	- Project manager	19
	- Project team lead	12
	- Delivery manager	9
	- Other	4
	Total	44
Type of project	- Rightshore	38
	- Direct offshore	6
	Total	44
Development method used	- Agile (DSMS | RUP)	12
	- Waterfall	15
	- Combination of agile and waterfall	9
	- Non-existent / ad-hoc	2
	- missing values	6
	Total	44
Time zone differences	- Between 3.5 and 4.5 hours	37
	- Between 4.5 and 9 hours	5
	- Between 9 and 11.5 hours	2
	Total	44
Project success	Average exceeding budget percentage	36%
	Average exceeding time schedule percentage	39%
	Average purpose achieved (1=good, 5 = bad)	1.90
	Average quality achieved (1=good, 5 = bad)	2.10
	Average stakeholder satisfaction (1=good, 5 = bad)	2.37
	Average front-office satisfaction (1=good, 5 = bad)	2.49
	Average back-office satisfaction (1=good, 5 = bad)	2.31
	Average front-office process satisfaction (1=good, 5 = bad)	2.78
	Average back-office process satisfaction (1=good, 5 = bad)	2.64

The perspective of filling in the survey was mostly Indian or Dutch which can also be seen in the time-zone differences which are mostly between 3.5 and 4.5 hours. Moreover it can be seen from the project success indicators that the purpose of the application and quality is mostly achieved, but that the satisfaction with the result and process falls a bit behind.

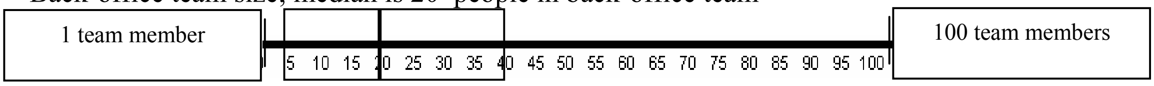
Team size and duration of project

The figures below illustrate the distribution of the number of team members and the duration of the projects. They are illustrated by boxplots. The yellow squares represent the range of distribution of people from the 25th percentile until the 75th percentile. The thick line inside this square represents the median, which marks the 50th percentile. The small lines perpendicular on the thick line from the front-office to the back-office marks the total range of the distribution (the 0th and 100th percentile of the distribution). The purpose of these figures is to visualize the distribution of the amount of people that worked on the projects and the distribution of the duration of the projects. These boxplots exclude three outliers of extremely large projects. This would distort the meaningfulness of the figures.

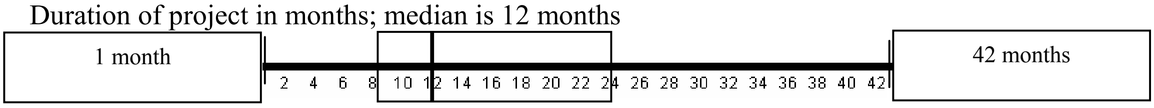
Front-office team size; median is 7 people in front-office team



Back-office team size; median is 20 people in back-office team



Duration of project in months; median is 12 months

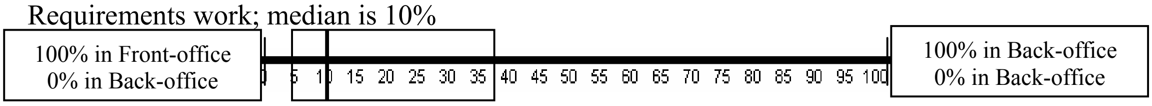


The boxplots indicate that the team size varies greatly between projects and is in general skewed to the left. Most projects taken in consideration are thus smaller than would be expected compared to examining mean values.

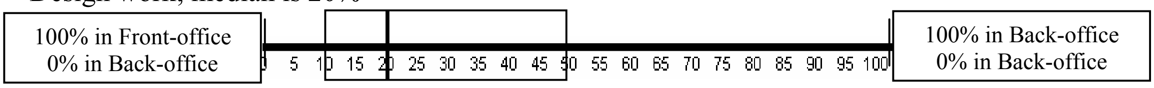
Distribution of work

The figures below illustrate the distribution of the distribution of work in the projects. They are also illustrated by boxplots. The purpose of these figures is to visualize the different distributions of work between front-office and back-office in different stages of application development.

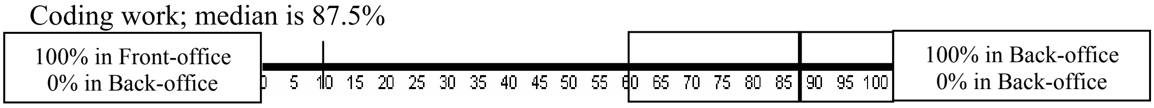
Requirements work; median is 10%



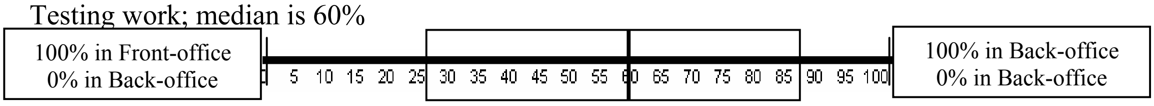
Design work; median is 20%



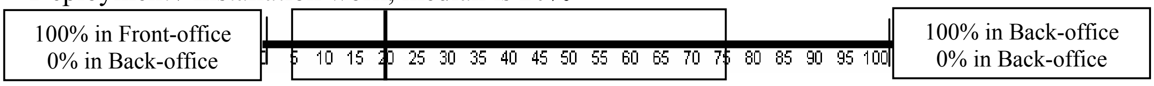
Coding work; median is 87.5%



Testing work; median is 60%



Deployment / Installation work; median is 20%



The boxplots indicate that requirements and design work is done mostly in the front-office team. The actual coding is done for the large part in the back-office. Distribution of testing and deployment work is very diverse in the examined projects. In general testing work is done more in the back-office and deployment work more in the front-office.

7.2.2 Project characteristic effects on project success

A number of tests were run to control for possible effects on project success based on project characteristics.

Project size

Based on a first overview of input on the project size indicators, it appeared that a reliable control mechanism could not be found for all of them, due to strange outlier values. Three of them that did were tested for an effect on project success; the number of members in front and back-office and the duration of the project (in months). A Pearson correlation was calculated for the indicators with project success and presented in table 36. Pearson was more suitable here, since the input variables were both variables measured or recoded at the interval level (Urdan, 2001). No significant relations were found. There was no effect of project size on project success in the sample.

Table 36 Effect of project size with project success

Variable	Correlation with Project success		p-value
Number of members in front-office	0.06	(n=44)	0.72
Number of members in back-office	0.06	(n=44)	0.68
Duration of project (months)	-0.0	(n=41)	0.98

** = significant at 0.05 level, ** = significant at 0.01 level*

Distribution of work

Next the distribution of work was tested for an effect on project success. Pearson correlations were calculated for the five variables with project success. No effect was found for distribution of requirements, design and deployment work. However a significant effect ($p<0.00$) *was* found for distribution of coding and testing work. More coding and testing work in the back-office is positively related to project success.

Table 37 Effect of distribution of work with project success

Variable	Correlation with Project success		p-value
Percentage of requirements work done in back-office	0.20	(n=44)	0.20
Percentage of design work done in back-office	0.20	(n=44)	0.20
Percentage of coding work done in the back-office	0.40	(n=44)	0.01**
Percentage of testing work done in the back-office	0.49	(n=44)	0.00**
Percentage of deployment work done in the back-office	0.00	(n=44)	0.97

** = significant at 0.05 level, ** = significant at 0.01 level*

Perspective and development method used
There were three perspective related questions in the survey: perspective country, project role, and type of project. The remaining samples sizes of different project roles, type of project and perspective country were too small (n<30) to derive statistical conclusions. The type of development method used presented the same problem. For sake of interest, a brief investigation was performed over these different characteristics to compare their means on project success. No indications for an effect were found (mean differences were negligible).

7.2.3 Top and bottom rated risk occurrence * impact

Based on the raw results of the survey data, we were able to derive a list of risk variables, sorted by occurrence * impact in the projects. The value of the risk factor occurrence was simply derived from the averages of the risk factors. The impact was derived by the Spearman Rho correlations of each individual risk variable with project success. Figure 15 below shows the pattern of all 81 risk variables sorted to occurrence * impact.

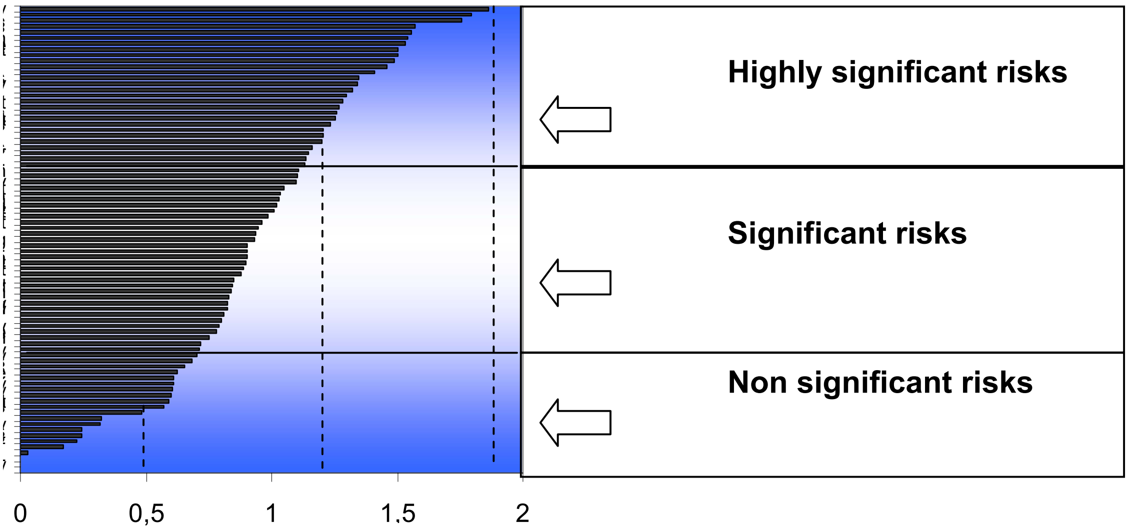


Figure 15: risk pattern occurrence * impact

For sake of space, the list is limited to the top twenty of risk variables and the bottom twenty-one risk variables. They are represented in figure 16-19 below. The order (top-down) resembles their occurrence * impact with the highest occurrence | impact on top.

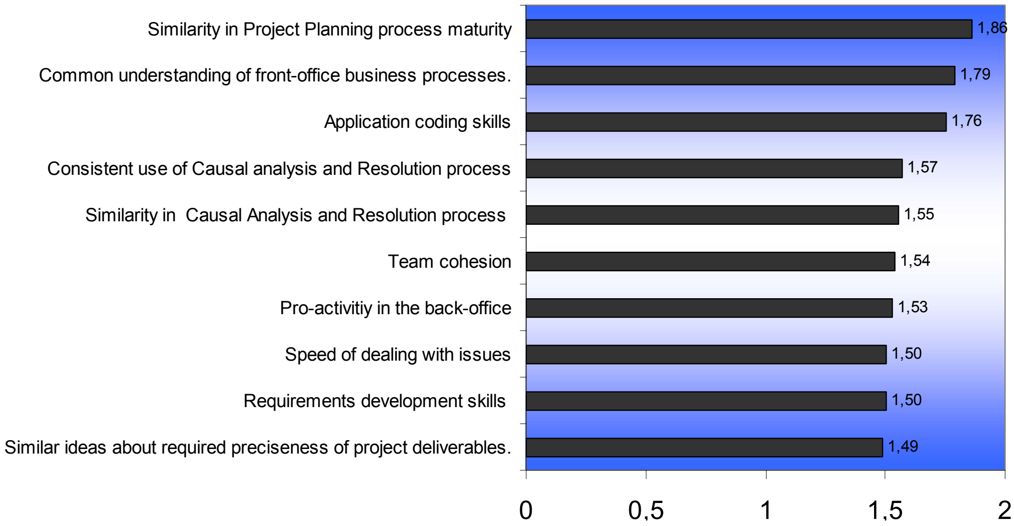


Figure 16: top-10 occurrence * impact

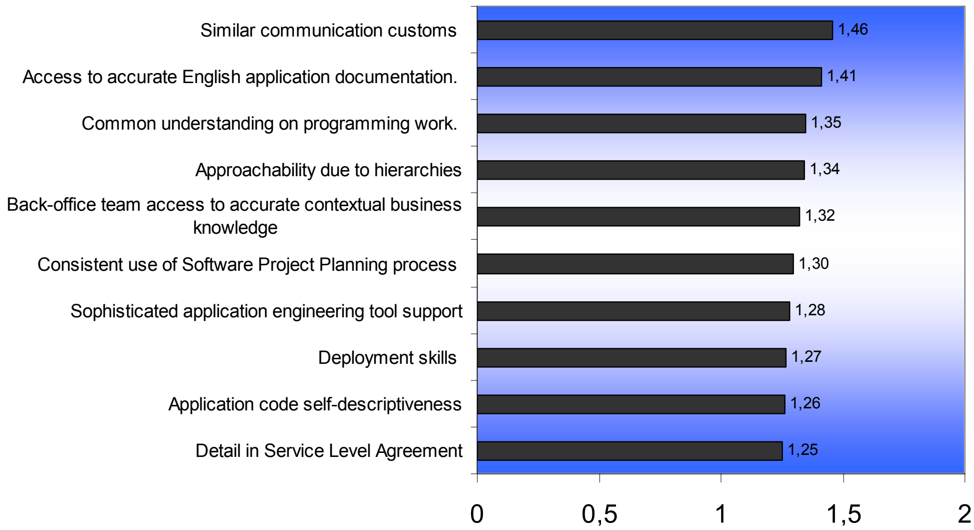


Figure 17: Risks 10-20 occurrence * impact

The individual top-rated risks shown in the two tables above represent risk from all of the defined risk categories. Most highly rated risks based on impact * occurrence are a lack of Project Planning process similarity, followed by a lack of common understanding of business processes. This set of risk factors gives a first impression of the most relevant risk variables. Quite some risk variables relate to the collaboration risks here, like a lack of access to application documentation, contextual business knowledge, approachability issues, dissimilar communication customs, a lack of pro-activity in the back-office, team cohesion and a low speed of dealing with issues. A meaningful interpretation or generalized statements can not be made yet at this point though.

The bottom rated risks are given in figure 18 and 19 below. They represent the lowest impact * occurrence of the 81 risk variables.

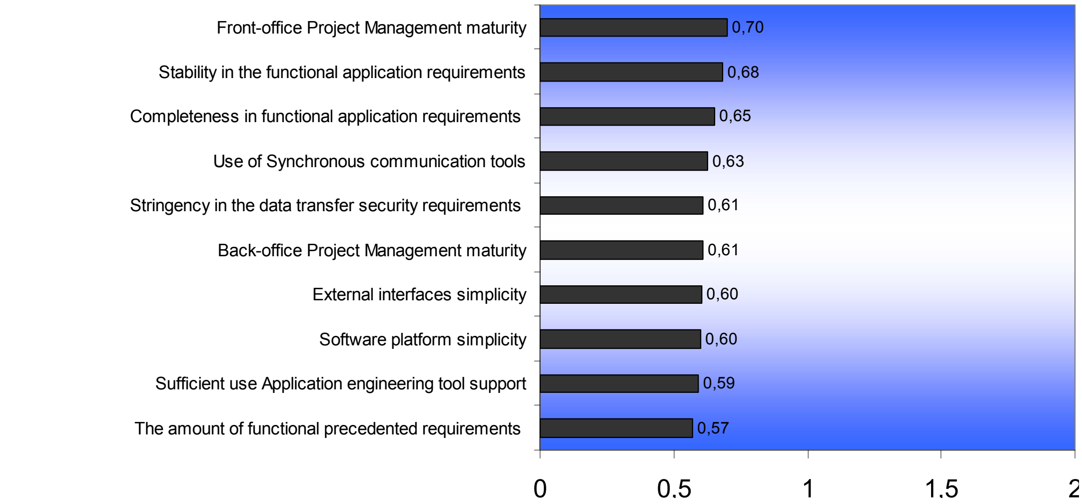


Figure 18: Risks 60-70 occurrence * impact

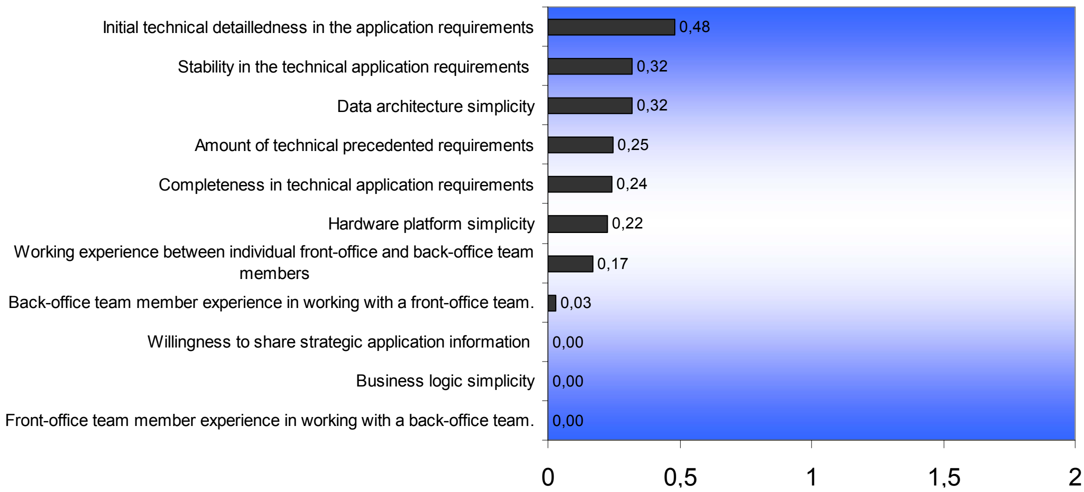


Figure 19: Risks 70-81 occurrence * impact

These bottom rated risk variables did not lead to a significant impact on project success. The most noticeable risks here are all variables related to experience, although it should be noted that there did not seem to be much experience in working in a distributed environment in general, which might distort the effect of "experience" risk variables. Moreover, there is a big presence of application risk variables in these bottom regions like for example hardware platform, data architecture, business logic, external interfaces, software platform and multiple requirements complexity variables. Again conclusions can not be drawn here, but an overall pattern seems to emerge here.

To conclude this section a generic scatterplot is shown (figure 20) of all risk combined (x-axis) and project success (y-axis). There is a significant (Pearson) correlation of 0.666 *($p < 0.00$)*. Note that this not indicates that risks in general should be added up and merged together, since all projects should have their own distinctive set of risk occurrences. An unsuccessful project has on average multiple risk occurrences. Risk occurrences are thus related to each other on average. Generally speaking, riskfull project are riskfull at multiple levels.

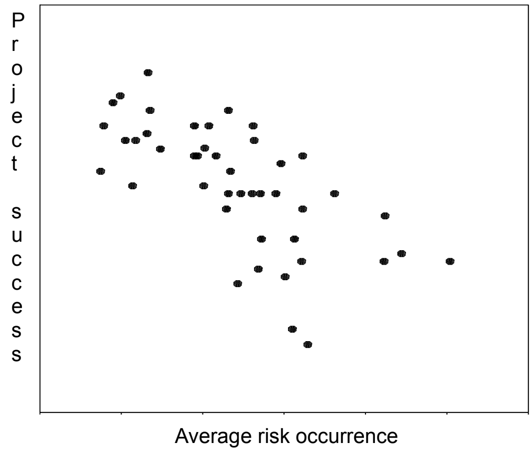


Figure 20: Average risk occurrence and project success

Basic descriptives findings

- Requirements and deployment work is predominantly done in front-office, coding in the back office. Design and testing in between.
- Average (based on median, not mean) project setting: 7 team members in front-office, 20 in back-office for duration of 12 months.
- More coding and testing work done in back-office = more project success.
- Project size does not influence project success
- Highest individual occurrence * impact of risk variables
    - Different process maturity Project Planning
    - Lack of common understanding business processes
    - Lack of Application coding skills
    - Lack of team cohesion
- Lowest individual occurrence * impact of risk variables
    - Lack of experience (front-office | back-office)
    - Lack of Business rule simplicity
    - Lack of willingness to share strategic application information
    - Lack of hardware platform simplicity
    - Lack of completeness in technical requirements
- Riskfull project are riskfull at multiple levels

7.3 RQ1: Categorized risk impact on project success

The basic descriptive information above provides a premature feeling with the dataset. As discussed, some more delicacy is required to accurately reflect upon risk factor impacts. In order to explain project success, linear regression models were used to test the first main research question:

> ***H1*: There is a significant impact of controllable and incontrollable application and collaboration risk factor occurrence on project success.**

Risk variables from six originating pools; categories have been distinguished. These are:

- Basic controllable application risk factors
- Residual controllable application risk factors
- Residual incontrollable application risk
- Basic controllable collaboration risk factors
- Residual controllable collaboration risk factors
- Residual incontrollable collaboration risk

A Principal Component Factor analysis of risk factors in each of the categories was performed prior to testing the hypothesized relations with project success.

Below are the results of every sub hypothesis in isolation. The number of extracted factors depended mostly on the scree plot and ability to give a meaningful interpretation of the factors. Residual factor loadings were saved as new variables. The rotated factor solution is shown for every sub hypothesis, revealing only variables with loadings higher than 0.5. In the few cases were variables loaded higher than 0.5 on more than one factor; the variable was categorized based on the highest factor loading. Risk variables that did not load on any of the factors with a value higher than 0.5 are not represented in the tables. An attempt was made to give a meaningful interpretation to the factors in the factor solution and name them accordingly. The residual factor loadings were regressed on project success. Shapiro-Wilk test in H1 proved not significant for all factor solutions but one: "Use of communication tools". This indicates that the other factors are approximately distributed normally and can be used as input for a regression analysis. The output of the regression analyses are given in this chapter. Note that p-values are rounded to two digits. The SPSS outputs of the regression models are given in Appendix 12.1.1. Those of the rotated factor solutions can be found in Appendix 12.2.1.

7.3.1 H1a: basic controllable application risk factors

H1a: There is a significant impact of basic controllable application risk occurrence on Project success. (Supported)

Keyser – Meyer – Olkin of the factor solution equalled 0.809 which is assumed high enough for distinctive power of the factor solution. Bartlett's test of sphericity was significant ($p < 0.00$) which indicates that the variables are sufficiently correlated to each other to be loaded on an individual factor. The rotated factor solution with all seventeen basic controllable application risk factors as input resulted in findings, denoted in table 38. Constructed factors were named: *Process similarity* and *architecture standardization / simplicity of retained application.* The residual loadings (scores) were used as predictors for the regression analysis.

Table 38 Factor solution: Basic controllable application risk factors

Aggregated Component	Risk factors	Factor loading
Process similarity (Factor 1)	- Requirements Management similarity	0.850
	- Requirements Development similarity	0.828
	- Configuration Management similarity	0.807
	- Development method similarity	0.769
	- Causal analysis and Resolution similarity	0.679
	- Monitoring and Control similarity	0.661
	- Measurement and Analysis similarity	0.654
	- Project Planning similarity	0.600
	- Service level agreement detailedness	0.546
Architecture standardization / Simplicity retained application (Factor 2)	- Self descriptiveness existing application	0.856
	- Structuredness existing application	0.837
	- Application architecture standardization	0.789
	- Familiarity with existing application	0.634
	- Data architecture standardization	0.511

Using the constructed factors in table 38 to explain project success in a linear regression proved significant ($p<0.01$). Process similarity ($p=0.01$) and architecture standardization/simplicity of the retained application ($p=0.01$) both have a significant impact on project success.

Table 39 Regression model, basic controllable application risk factors to project success

R	Value	Factor	Beta	p-value
R	0.505	Constant		0.00**
R Square	0.255	Process similarity	0.35	0.01*
Adjusted R Square	0.219	Architecture standardization / simplicity retained application	0.37	0.01**
p-value	0.00**			

** = significant at 0.05 level, ** = significant at 0.01 level*

7.3.2 H1b: residual controllable application risk factors

H1b: There is a significant impact of residual controllable application risk occurrence on project success (Supported)

Keyser – Meyer – Olkin of the factor solution equaled 0.777 which is assumed high enough for distinctive power of the factor solution. Bartlett's test of sphericity was significant ($p < 0.00$) which indicated that the variables are sufficiently correlated to each other to be loaded on an individual factor. The rotated factor solution with all ten residual controllable application risk factors as input lead to findings, denoted in table 40. All compliance variables scored high on the first factor; which was therefore named: *process compliance*. The second factor was named: *consistent and sufficient use of tool support.* The residual loadings (scores) were used as predictors for the regression analysis.

Table 40 Factor solution: Residual controllable application risk factors

Aggregated Component	Risk factors	Factor loading
Process compliance	- Requirements Management consistent use	0.905
(Factor 1)	- Configuration Management consistent use	0.895
	- Requirements Development consistent use	0.867
	- Monitoring and Control consistent use	0.854
	- Measurement and Analysis consistent use	0.843
	- Project Planning consistent use	0.732
	- Causal analysis and Resolution consistent use	0.705
	- Development method consistent use	0.533
Consistent / sufficient use of tool support	- Application engineering tool support consistent use	0.891
(Factor 2)	- Testing tool support sufficient / consistent use[4]	0.773
	- Application engineering tool support sufficient use	0.716

Using the constructed factors from table 40 to explain project success in a linear regression proved significant ($p<0.01$). Process compliance ($p=0.01$) and a consistent / sufficient use of tool support ($p=0.03$) both have a significant impact on project success.

Table 41 Regression model, residual controllable application risk factors to project success

R	Value	Factor	Beta	p-value
R	0.473	Constant		0.00**
R Square	0.233	Process compliance	0.36	0.01*
Adjusted R Square	0.185	Consistent/sufficient use tool support	0.31	0.03*
p-value	0.01**			

** = significant at 0.05 level, ** = significant at 0.01 level*

7.3.3 H1c: residual incontrollable application risk

H1c: There is a significant impact of residual incontrollable application risk occurrence on project success. (Supported)

Keyser – Meyer – Olkin of the factor solution equaled 0.737 which is assumed high enough for distinctive power of the factor solution. Bartlett's test of sphericity was not significant which indicates that the variables are insufficiently correlated to each other to be loaded on an individual factor. However, due to the meaningful interpretation / sense making of the three factors and high factor loadings, we are confident enough to use the residual factor loadings for the regression on project success. The rotated factor solution with all twenty-two residual incontrollable application risk factors as input resulted in the solution, represented in table 42: The factor solution falls apart in three clear-cut groups. *Application complexity*, representing the extent that all aspects of the application proved more complex than expected and *Requirements complexity* representing all exceeding complexity with regard to requirements. Finally *Human skills* are distinguished, representing risk variables on failed expectations of counterpart human skills. The residual loadings were used in a linear regression analysis.

[4] These in fact represent two variables; they measured the same thing according to their correlation > 0.9

Table 42 Factor solution: Residual incontrollable application risk

Aggregated Component	Risk factors	Factor loading
Application Complexity (Factor 1)	- Technical architecture complexity	0.871
	- Internal interface complexity	0.827
	- Module diversity	0.823
	- Data architecture complexity	0.812
	- Hardware platform complexity	0.732
	- Data transfer security stringency	0.707
	- External interface complexity	0.703
	- Business logic complexity	0.681
	- Software platform complexity	0.596
Requirements complexity (Factor 2)	- Technical detailedness in requirements	0.887
	- Technical precedented requirements	0.861
	- Functional detailedness in requirements	0.834
	- Functional unwritten requirements	0.826
	- Technical unwritten requirements	0.746
	- Functional stability in requirements	0.741
Human skills (Factor 3)	- Application coding skills	0.815
	- Architecture design skills	0.754
	- Requirements development skills	0.683
	- Deployment skills	0.635
	- Software testing skills	0.632

Using the constructed factors from table 42 to explain project success in a linear regression proved significant ($p<0.00$). Exceeding human skills ($p=0.00$) has a significant impact on project success. Requirements complexity and application complexity are not significant at the 0.05 level. The former is significant at the 0.1 level.

Table 43 Regression model, residual incontrollable application risk factors to project success

R	Value	Factor	Beta	p-value
R	0.615	Constant		0.00**
R Square	0.378	Application complexity	0.19	0.14
Adjusted R Square	0.331	Requirements complexity	0.23	0.07
p-value	0.00**	Human skills	0.54	0.00**

** = significant at 0.05 level, ** = significant at 0.01 level*

7.3.4 H1d: basic controllable collaboration risk factors

H1d: There is a significant impact of basic controllable collaboration risk occurrence on project success. (Supported)

Keyser – Meyer – Olkin of the factor solution equaled 0.665; high enough for distinctive power of the factor solution. Bartlett's test of sphericity is significant ($p < 0.00$). The variables are sufficiently correlated to each other to be loaded on an individual factor. The rotated factor solution with all 11 basic controllable collaboration risk factors as input lead to the factor solution denoted shown in table 44: The factor solution resulted in three factors: *Collaborative means, Project Management* maturity and *Experience.* Residual factor loadings were used as input for a linear regression analysis.

Table 44 Factor solution: basic controllable collaboration risk factors

Aggregated Component	Risk factors	Factor loading
Collaborative means (Factor 1)	- Access to contextual business knowledge	0.906
	- Access to application documentation	0.840
	- Working with liaisons	0.558
	- High level synchronous communication tools availability	0.503
Project Management maturity (Factor 2)	- Front-office Project Management maturity	0.850
	- Front-office / back-office differences in PM method	0.837
	- Back-office Project Management maturity	0.635
Experience (Factor 3)	- Back-office experience working with front-office team	0.826
	- Individual working experience with each other	0.707
	- Front-office experience working with back-office team	0.677

Using the constructed factors from table 44 to explain project success in a linear regression proved significant ($p<0.00$). Collaborative means ($p=0.00$) has a significant impact on project success. It appears that the Experience Factor ($p=0.47$) and Project Management maturity ($p=0.41$) did not have an impact on project success.

Table 45 Regression model, basic controllable collaboration risk factors to project success

R	Value	Factor	Beta	p-value
R	0.465	Constant		0.00**
R Square	0.216	Collaborative means	0.44	0.00**
Adjusted R Square	0.157	Project Mgt. maturity / alignment	0.12	0.41
p-value	0.02*	Experience	-0.10	0.47

** = significant at 0.05 level, ** = significant at 0.01 level*

7.3.5 H1e: Residual controllable collaboration risk factors

H1e: There is a significant impact of residual controllable collaboration risk occurrence on project success. (Supported)

Keyser – Meyer – Olkin of the factor solution equaled 0.490 which is just beneath the threshold of 0.5 to determine distinctive power of the factor solution. However, the two factors did seem to come from a profoundly different origin. Moreover Bartlett's test of sphericity was significant ($p = 0.02$). We are again confident enough to use the residual factor loadings for regression analysis. The rotated factor solution with the three residual controllable collaboration risk factors as input lead to the following factor solution in table 46: Two factors from which interpretation is rather forthcoming were formed: *Project Management compliance* and *synchronous communication tools use.*

Table 46 Factor solution: residual controllable collaboration risk factors

Aggregated Component	Risk factors	Factor loading
Project Management compliance (Factor 1)	- Front-office compliance to Project Management method	0.874
	- Back-office compliance to Project Management method	0.829
Synchronous communication tools use (Factor 2)	- High level synchronous communication tools use	0.985

Using the constructed factors from table 46 to explain project success in a linear regression proved significant ($p<0.05$). Project Management compliance ($p=0.02$) has a significant impact on project success. The "Synchronous communication tools use" factor was based on only one variable and did not meet the assumption for approximate normal distribution (Shapiro-Wilk <0.05). It was thus not used for the regression analysis.

Table 47 Regression model, residual controllable collaboration risk factors to project success

R	Value	Factor	Beta	Sig.
R	0.339	Constant		0.00**
R Square	0.115	Project Management compliance	0.34	0.02*
Adjusted R Square	0.094			
Sig.	0.02*			

** = significant at 0.05 level, ** = significant at 0.01 level*

7.3.6 H1f: Residual incontrollable collaboration risk

H1f: There is a significant impact of residual incontrollable collaboration risk occurrence on project success. (Supported)

Keyser – Meyer – Olkin of the factor solution equaled 0.726 which is assumed high enough for distinctive power of the factor solution. Bartlett's test of sphericity is significant (p < 0.00) which indicated that the variables are sufficiently correlated to each other to be loaded on an individual factor. The rotated factor solution with all 16 residual incontrollable collaboration risk variables lead to the results in table 48. Three rather forthcoming factors were formed: *Scope of work clarity, Collaboration efficiency* and *Language and customs issues.* Residual factor loadings were used as input for a linear regression analysis.

Table 48 Factor solution: residual incontrollable collaboration risk

Aggregated Component	Risk factors	Factor loading
Scope of work clarity	- Clarity of contextual business knowledge	0.812
(Factor 1)	- Clarity of business processes in application	0.796
	- Application objective clarity	0.788
	- Hierarchical approachability	0.694
	- Required precision of deliverables clarity	0.627
	- Role responsibilities clarity	0.613
	- Pro-activity in back-office team	0.521
	- Common understanding programming work	0.537
Collaboration efficiency	- Willingness to accommodate other side objectives	0.728
(Factor 2)	- Consistency in team member objectives	0.724
	- Team cohesion	0.722
	- Speed of dealing with issues	0.715
	- Collaboration skills	0.709
Language and customs issues	- Different communication customs	0.763
(Factor 3)	- Native language understanding problems	0.739

Using the constructed factors from table 48 to explain project success in a linear regression proved significant (p<0.00). Scope of work clarity (p=0.00), Collaboration efficiency (p=0.00) and Language and customs issues (p=0.02) have a significant impact on project success.

Table 49 Regression model, incontrollable collaboration risk factors to project success

R	Value	Factor	Beta	p-value
R	0.683	Constant		0.00**
R Square	0.467	Scope of work clarity	0.39	0.00**
Adjusted R Square	0.427	Collaboration efficiency	0.48	0.00**
p-value	0.00**	Language and customs issues	0.29	0.02*

** = significant at 0.05 level, ** = significant at 0.01 level*

7.3.7 H1 Combined results

All constructed factor solutions were significantly able to predict project success. To sum up this section, an overview of relative impacts is given below (figure 21). These impacts can not be added together, since factor solutions were constructed in stages and explained variances might thus overlap between the six factor solutions, but their relative impact is valid in this context. The orange bars represent aggregated risks; the black bars represent aggregated risk factors. The next hypotheses attempt to explain the risks by using the risk factors.

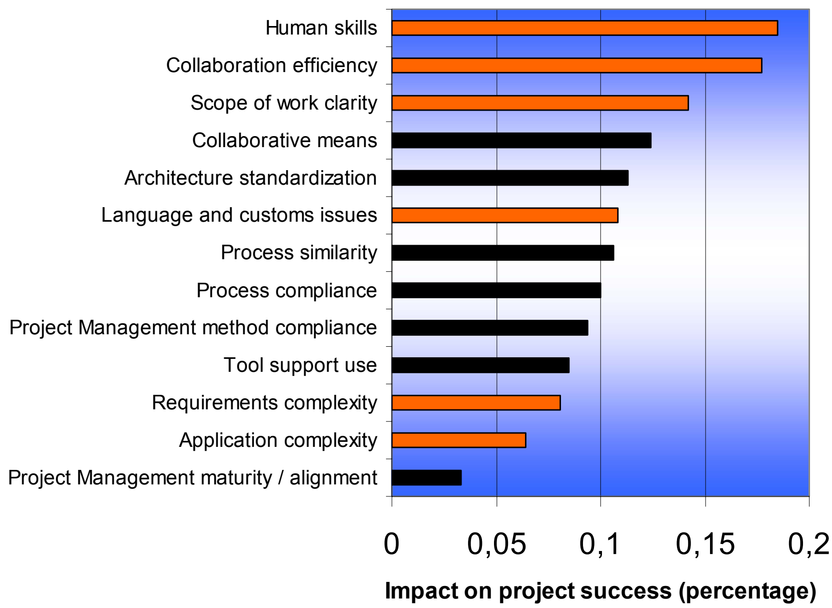


Figure 21: H1 Combined results

7.4 RQ2: Risk dependencies

The second hypothesis is about the effect of controllable risks factors on incontrollable risks; the effect that a controlled environment has on exceeding expectations. Again, factor analyses in combination with linear regression were chosen as statistical techniques. The SPSS outputs of these rotated factor solutions can be found in Appendix 12.2.2. Appendix 12.1.2 contains the complete SPSS output of the regression models.

Factors are formed here one aggregation level higher then in H1. Basic and residual controllable risk variables were merged together as input for a varimax rotated principal components factor solution. The reason for this higher level of aggregation is to explore the *total* set of controllable risk factors for similarities in a new factor solution. It might well be that similarity (basic) and compliance (residual) factors load high on the same factor.

The factor loadings are put in a linear regression model to explore the effect that the orthogonal aggregated variables have on residual incontrollable risks (stability in expectations). The Shapiro-Wilk test for approximate normal distribution of input variables proved non-significant for all factors ($p>0.05$), indicating that the distribution of the factor loadings is approximately normal. One that did not: "Process similarity" was excluded in the linear regression model.

7.4.1 Factor solutions

Two factor solutions were performed twice, distinguishing between controllable application and controllable collaboration risk factors.

Controllable application risk factors

The first factor solution resulted in a sufficient value of 0.612 for Keyser – Meyer – Olkin. Bartlett's test of sphericity was not significant, but a meaningful interpretation could still be given to the factors. The best interpretable factor solution results in five factors: *Process compliance*, *process similarity*, *simplicity of the retained application*, *level of tool support* and *consistent / sufficient use of tool support.*

Table 50 Factor solution: Controllable application risk

Aggregated Component	Risk factors	Factor loading
Process compliance (Factor 1)	- Monitoring and Control consistent use	0.873
	- Configuration Management consistent use	0.843
	- Requirements Management consistent use	0.843
	- Requirements Development consistent use	0.828
	- Measurement and Analysis consistent use	0.787
	- Project Planning consistent use	0.728
	- Service level agreement detailedness	0.690
	- Causal analysis and Resolution consistent use	0.613
	- Project Planning similarity	0.593
	- Monitoring and Control similarity	0.587
Process similarity (Factor 2)	- Requirements Management similarity	0.834
	- Requirements Development similarity	0.804
	- Configuration Management similarity	0.778
	- Development method maturity similarity	0.666
Simplicity of retained application (Factor 3)	- Structuredness existing application	0.842
	- Self descriptiveness existing application	0.803
	- Familiarity with existing application	0.783
	- Application architecture standards	0.610
	- Measurement and Analysis process similarity	0.510
Level of tool support (Factor 4)	- Application Engineering tool support level	0.758
	- Testing tool support level	0.715
	- Data architecture standardization	0.635
Consistent / sufficient use of tool support (Factor 5)	- Application engineering tool support sufficient use	0.772
	- Application engineering tool support consistent use	0.701
	- Testing tool support sufficient / consistent use	0.572

Controllable collaboration risk factors

Keyser – Meyer - Olkin of the factor solution was 0.654, which is assumed high enough. Bartlett's test of sphericity was significant ($p<0.00$). Results are presented in table 51. Five factors could be named and interpreted. The first factor is *"relief of collaboration efforts"*. The variables that load high on this factor are related to making the collaboration efforts easier. The second factor relates to *front-office Project Management maturity / compliance*. The third factor relates to *Experience* of front and back-office in working together. The fourth factor is about *availability and use of communication tools*. The fifth factor represents one isolated variable, the *use of liaisons* in the project team.

Table 51 Factor solution: Controllable collaboration risk

Aggregated Component	Risk factors	Factor loading
Relief of collaboration efforts (Factor 1)	- Access to application documentation	0.833
	- Access to contextual business knowledge	0.734
	- Back-office compliance to Project Management method	0.663
	- Collaboration team training	0.620
Front office Project Management maturity/ Compliance (Factor 2)	- Front-office Project Management maturity	0.854
	- Front-office / back-office differences in PM method	0.758
	- Front-office compliance to Project Management method	0.732
Experience (Factor 3)	- Back-office experience working with front-office team	0.772
	- Front-office experience working with back-office team	0.772
	- Individual working experience with each other	0.665
Communication tools (Factor 4)	- High level synchronous communication tools availability	0.880
	- High level synchronous communication tools use	0.878
Liaisons (Factor 5	- Working with liaisons	0.887

7.4.2 H2a controllable application risk -> incontrollable application risk

H2a: There is an impact of controllable application risk factors on incontrollable application risk occurrence. (Partly supported)

The first sets of factors in table 50 were used as independent variables to explain the exceeding expectations measured in the residual incontrollable application risks (see table 42) by performing three linear regression analyses. Table (52) below shows the Beta's and p-values of these analyses. There was some supporting evidence for controllable application risk factors on "human skills" expectations ($p<0.01$) and an individual impact of the "process compliance" factor to "application complexity". Significant effects are marked in bold.

Table 52 Regression of controllable risk factors on incontrollable application risks (H2a)

Aggregated Component	Application complexity		Requirements complexity		Human skills	
	Beta	p-value	Beta	p-value	Beta	p-value
Process compliance	**0.30**	0.04*	-0.07	0.63	0.17	0.20
Simplicity retained application	-0.07	0.49	0.22	0.16	**0.33**	0.02*
Level of tool support	0.24	0.10	0.01	0.96	**0.30**	0.03*
Consistent/Sufficient use tool support	0.23	0.13	0.08	0.60	**0.27**	0.05*
Regression model in total		0.094		0.508		0.005**

** = significant at 0.05 level, ** = significant at 0.01 level*

7.4.3 H2b controllable collaboration risk -> incontrollable collaboration risk

H2b: There is an impact of controllable collaboration risk factors on incontrollable collaboration risk occurrence. (Mostly supported)

The second factor solution of table 51, representing aggregations of controllable collaboration risks was used to predict exceeding expectations of incontrollable collaboration risks (see table 48). The beta-coefficients and p-value of the regression models are given in the table 53 below. The significant effects are marked in bold. The strongest impact found is that of the “relief of collaboration efforts” to reduce the risk of an “unclear scope of work”.

Table 53 Controllable collaboration risk factors on incontrollable collaboration risks (H2b)

Aggregated Component	Scope of work clarity		Collaboration efficiency		Language/Customs	
	Beta	p-value	Beta	p-value	Beta	p-value
Relief of collaboration efforts	**0.60**	0.00**	**0.28**	0.03*	0.27	0.08
Front office PM maturity/compliance	0.19	0.10	**0.33**	0.01**	-0.17	0.26
Experience	-0.06	0.62	0.06	0.62	-0.11	0.46
Communication tools	**0.23**	0.05*	**0.44**	0.00**	0.02	0.89
Liaisons	0.22	0.06	0.14	0.29	0.20	0.20
Regression model in total		0.005**		0.001**		0.252

** = significant at 0.05 level, ** = significant at 0.01 level*

7.4.4 H2c controllable application risk -> incontrollable collaboration risk

H2c: There is an impact of controllable application risk factors on incontrollable collaboration risk occurrence. (Mostly supported)

The third subquestion referred to an effect between controllable application risks and incontrollable collaboration risk. The controllable application risk factors of the factor solution in table 50 are used as independent variables to explain the factors of the incontrollable collaboration risk (see table 48). The beta-coefficient and p-values of the regression model are given in the table below (54). The “level of (sophistication in) tool support” leads to less exceeding expectations in “scope of work clarity”. “Process compliance” leads to a decreased risk of “collaboration efficiency”. Significant effects are marked in bold.

Table 54 Controllable application risk factors on incontrollable collaboration risks (H2c)

Aggregated Component	Scope of work clarity		Collaboration efficiency		Language/Customs	
	Beta	p-value	Beta	p-value	Beta	p-value
Process compliance	0.27	0.06	**0.40**	0.01**	-0.05	0.78
Simplicity retained application	0.20	0.17	0.21	0.16	0.18	0.26
Level of tool support	**0.29**	0.03*	-0.11	0.96	0.12	0.44
Consistent/Sufficient use tool support	-0.16	0.27	0.14	0.60	0.10	0.54
Regression model in total		0.038*		0.041*		0.791

** = significant at 0.05 level, ** = significant at 0.01 level*

7.4.5 H2d controllable collaboration risk -> incontrollable application risk

H2d: There is an impact of controllable collaboration risk factors on incontrollable application risk occurrence. (Partly supported)

The final three linear regression models were constructed based on the controllable collaboration risks of table 51 as independent variable to predict exceeding application risk expectations; the incontrollable application risk denoted in table 42. The beta-coefficients and p-values of the regression model are given in the table below (55). Only an effect was shown of "relief of collaboration efforts" to reduce misaligned "human skills" expectations, which is marked in bold.

Table 55 Controllable collaboration risk factors on incontrollable application risks (H2d)

Aggregated Component	Application complexity		Requirements complexity		Human skills	
	Beta	p-value	Beta	p-value	Beta	p-value
Relief of collaboration efforts	-0.08	0.63	0.12	0.46	**0.44**	0.00**
Front office PM maturity/compliance	0.03	0.87	-0.09	0.55	0.16	0.28
Experience	0.14	0.37	0.06	0.71	-0.05	0.71
Communication tools	0.10	0.53	-0.16	0.29	0.11	0.44
Liaisons	-0.22	0.16	0.18	0.24	-0.03	0.83
Regression model in total		0.617		0.619		0.067

** = significant at 0.05 level, ** = significant at 0.01 level*

To sum up: Evidence for H2 was partially found. Some of the controllable factors were able to explain project success, some where not. An overall picture of the results of H2 combined with a part of H1 is given in the next paragraph.

7.5 Combined results of H1 and H2

The combined results of relations found in H2 are illustrated in the model below. The impacts from the residual incontrollable risk factors in H1 are also included in the figure to give a comprehensive overview. The arrow denotes a significant relation, with the exception of requirements complexity and application complexity, which appeared to have a modest, but non significant impact. It should be noted that the effect of the basic and residual factor solutions directly to project success from RQ1 is not integrated in this model. This might well explain another portion of explained variance of project success, not captured in this model though.

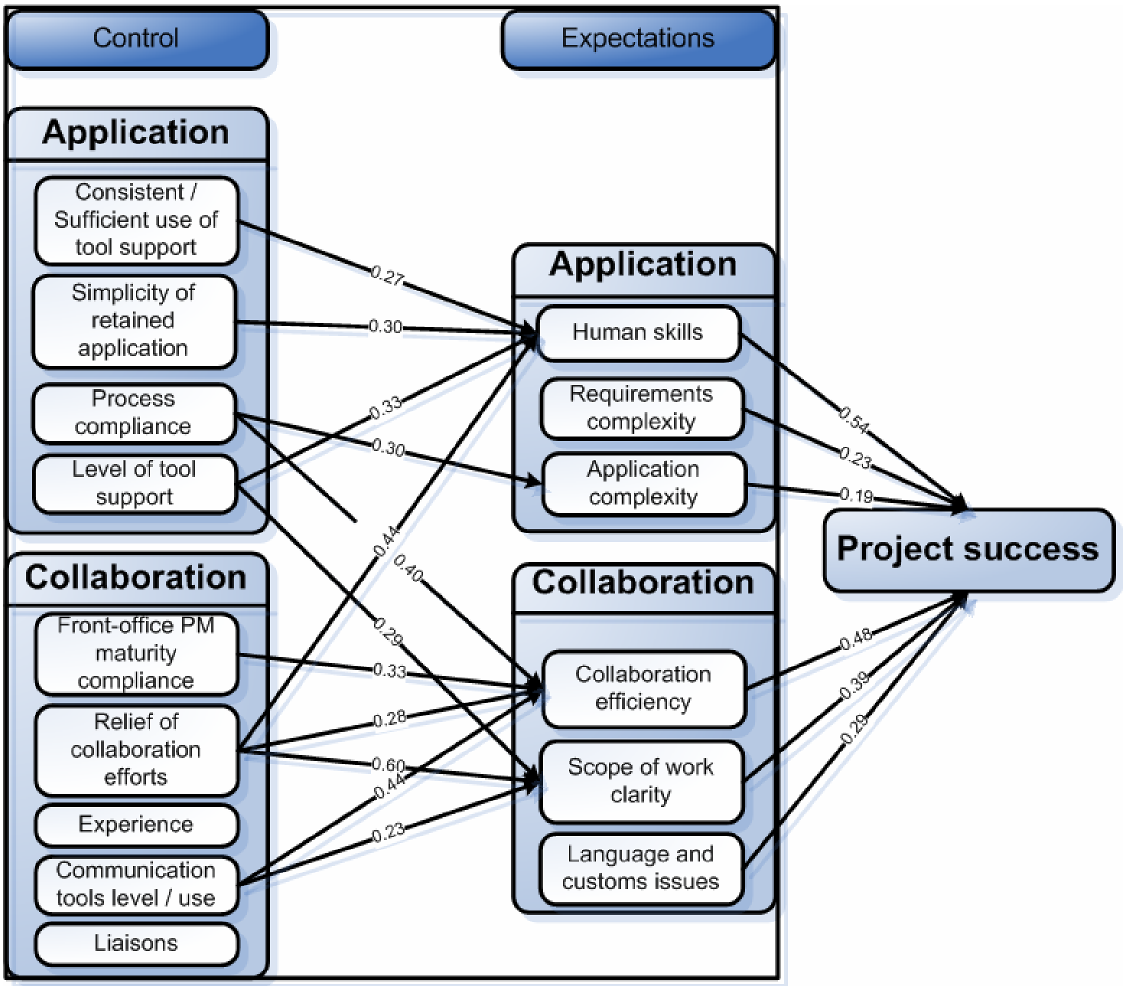


Figure 22: H1 + H2 Combined results

7.6 RQ3: Offshore readiness assessment improvement

Three steps were taken to improve the offshore readiness assessment. The mapping of risk (factors) on the RAS constructs is illustrated in appendix 12.3. Some constructs could not be mapped onto RAS. 40 out 81 risk variables were mapped.

Appendix 12.3 also contains the Spearman correlations multiplied with occurrence of the risk variable in projects. New weighing factors were derived. This was done by adding all averaged value for Complexity and Capability separately and recode relative contributions to the same scales of respectively 7 and 5 points. There were no risk variables for “Support” or “People” that could be properly aligned, so the weight of support and people was kept to 1.

The end results of RAS’ Application development derivative are illustrated below in table 56.

Table 56 RAS’ weights

RAS construct	Original weight	New weight (correlation-based)
Architecture	1	0.90
Business logic	1	1.15
Application complexity	1	1.45
Interface complexity	1	0.90
Infrastructure complexity	1	0.52
Interaction complexity	1	1.08
Support complexity	1	1
Knowledge Management	1	1.25
Methods	1	1.02
Requirements	1	0.86
Test	1	0.87
People	1	1

8 Conclusion

The goal of this study was to bring more delicacy in knowledge about risk presence and impact in offshore outsourced application development and to give risk mitigation options for an IT organization looking for sourcing alternatives. These goals have primarily been met in answering the first two research questions as will become clear. This concluding chapter is built up by remarking on the results of the three research questions followed by an overall reflection to the main research aim. First, the basic findings on survey data are being discussed.

Most importantly, we were able to use one solid dependent variable (See section 7.1.2) in this research: project success. It appeared that the operational project success items were sufficiently high related to each other to be taken as one variable; a critical step, since most resulting evidence of this research is about the eventual project success. A brief overview of their respective occurrences shows that the offshore application development project meets the intended purpose of the client IT organization in general. Averages of dissatisfaction with the process of developing the application are highest, indicating that the actual day –to day operations in the project are suboptimal. In the end, the client will get its desired application, but at an average of 36% more budget and 37% time schedule overrun. These overruns could well be the main causes that give dissatisfaction with the process the highest occurrence.

The small sample size prohibited measuring of statistically solid project characteristic effects based on perspective or used development method. However, a brief investigation of mean differences for project success was negligible, strongly indicating that neither the perspective of filling in the survey nor the used development method influences project success.

An effect was found though based on distribution of work with regard to coding and testing work in section 7.2.2. More coding and testing work done in the back-office is thus positively related to project success. An explanation for this can be threefold:

1. The more work that is done in one location, the less intensive interaction over distance is required.
2. The back-office team is just better in coding and testing than the front-office team
3. The more responsibility laid on the back-office team, the more commitment and willingness is brought to bear

Especially for coding work, which is already done for the large part in the back-office (see the boxplots in section 7.2.1), the first reason presumably gives the best explanation. Straightforward coding work that is done completely in the back-office implicitly entails less interaction between a front-office and back-office team. All possible problems can be solved on the spot.

On the individual risk factor level (figure 16 and 17), some supporting evidence can be found for the third explanation. A lack of pro-activity and willingness to accommodate other side objectives are among the highest impact*occurrence of the risk variables. Finding parallels and similarity in the other highest impacts is difficult. To analyze a few of them, it can be seen that starting the project prepared, can pay of. The same process maturity of Project Planning, a detailed service level agreement and back-office access to (contextual) business knowledge are among the top influential risk variables for project success. In general a conclusion that can be drawn is that the biggest risks follow from the distributed (human side) character of the project, like a lack of team cohesion, low speed of dealing with issues, misaligned ideas about preciseness of deliverables, different communication customs, hierarchical approachability difficulties, lack of contextual knowledge to name a few. To sum up, it does not seem to be the unexpected technical complexity issues that *mostly* determine project success, but the human collaboration factors of dealing with this uncertainty.

1. *What is the impact of controllable and incontrollable application and collaboration risk factor occurrence on project success?*

To validate different origins of risk, six subhypotheses were tested. The extensive amount of eighty variables was brought back to a comprehensible set of Factors for each of the six categories. H1 and all of its six subhypotheses proved significant, indicating that the risk variables in the different risk categories were representing meaningful types of risk. The individual factor impacts lead to more meaningful conclusions that can be drawn.

H1a

The basic controllable application risk factor solution lead to two factors: "Process similarity" and "architecture standardization / simplicity of the retained application". Both factors had an impact on project success. An IT organization could thus benefit from elevating CMMI processes to the level of the outsourcing vendor prior to offshore outsourcing or standardizing their application / data architecture. This can be an expensive undertaking. Looking at the individual risk variable impacts, priority should be given to maturing the Software Project Planning process. Service level Agreement detailedness could not really be placed in the factor solution, but its isolated impact (see figure 17) is quite large. The outsourcing IT organization should consider this to prevent misaligned expectations.

H1b

The residual controllable application risk factor solution also leads to two factors: "Process compliance" and "consistent / sufficient use of tool support". There are few researchers that address the actual compliance to the process in a project, but it appeared to have a significant impact on project success. The application engineering and testing tool support should be used consistently and sufficiently by team members. In general, having mature processes and tool support in place is not enough; a lack of compliance in the project has a negative effect on the success of the project. Project Managers should be aware of this and attempt to stress the importance to team members to comply consistently with the defined processes and tool support options in place.

H1c

The residual incontrollable application risk factor solution leads to three factors: "Application complexity", "requirements complexity" and "human skills". Note that these were all measured by the extent of exceeding expectations. The large part of the explained variance on project success was caused by the "human skills" factor. The results indicate that human skills expectations are at the very least a good predictor for project success. It seems that a lack of skills in the counterpart team is a good "excuse" for failure. However, loosely translated as team pressure and an indirect (but very powerful) indicator for project success, another side of risk can be enlightened. The technical unexpected complexities like application and requirements complexity have a modest impact on project success. In general it can be concluded that the impact of the latter is not as big as could be expected based on the extensive amount of literature that address requirements engineering difficulties as a big risk. It is difficult to speculate on possible reasons. The most plausible cause is the early stage identification and action potential on requirements complexity. Roughly said, half of the projects considered, used the waterfall approach of development. Problems in requirements might occur here, but sufficient time is left to keep up with development work. Also process satisfaction measurements (two items of project success) might not be affected that much by these early stage problems. In the other half of the project using agile development, the requirements complexity is part of the small iteration process. Potential problems can be solved fast.

H1d

The basic controllable risks factor solution lead to three factors: "Collaborative means", "project management maturity" and "experience". The collaborative means are most important for project success. Access to (contextual) documentation, working with liaisons and the availability of high level synchronous communication tools are rather inexpensive things to consider for an outsourcing organization and show to have large effect. One of the most peculiar findings is the total lack of impact evidence of "experience" on project success. Also on the individual risk factor level (see figure 17), there is no influence to be found. One cause for this might be that a lack of working experience had by far the highest occurrence in the projects considered. In other words, most projects were performed without much previous working experience of team members in a distributed team. The discriminative power of the skewed variable is thereby somewhat delimited.

H1e

The residual controllable collaboration risk factor solution lead to two factors: "Project Management method compliance" and "use of synchronous communication tools". It seems that not project management maturity but rather compliance to whatever is in place, leads to project success. This is backed up by the high impact of the "project planning process" discussed earlier. The use of high level synchronous collaboration tools could unfortunately not be used as input for the regression analysis, due to the failure to meet required data conditions.

H1f

Like concluded before, the most influencing risks are the incontrollable collaboration risks. This category lead to three factors: "Scope of work clarity", "collaboration efficiency" and "language and customs issues". This category consists of risk measured in exceeding expectations, similar to the incontrollable application risks. However, as is shown, exceeding expectations of these human factors have a profoundly bigger impact on project success than the incontrollable application risks. "Scope of work clarity" consists mainly of high-loading variables like a lack of common understanding or clarity of what the scope of work entails. A lack of contextual business knowledge, lack of business processes knowledge, role responsibilities and a lack of understanding of the end-objective of the application are among the most important variables in this factor. The two other factors refer mostly to the offshore and distributed character of the project. "Language and customs issues" provide difficulties in obtaining common understanding on project work. The outsourcing IT organization should not underestimate these "soft" factors, because they have a rather large impact on project success. The single most influencing factor is "collaboration efficiency". "Collaboration efficiency" consists of variables mostly related to enablers for more efficient communication. It is arguable that the risks in this factor are difficult to control, since they seem to touch the core of working in a distributed environment with variables like, low speed of dealing with issues, and lack of team cohesion for example.

Translating the findings of H1 to the Rightshore Assessment Study, two main conclusions can be drawn. First, it has been shown that every defined category has a significant effect on project success as a whole. Similar processes, methods and advanced tools have an impact on project success, providing the foundation for the organizational *capability* pillar of the RAS assessment. Supporting evidence for exceeding application complexities was also found, indicating the relevance of the remaining application *complexity* pillar of RAS.
The other conclusion is that the biggest impact on project success seems to come though from the human side of collaboration issues, not yet integrated in RAS. At this point in the concluding

remarks, it is unknown *how* risks relating to the latter can be controlled. It might well be that these collaboration issues can be controlled by unexpected (technical) mitigation actions. Knowing the origin and impact of risks is one part of the puzzle, creating a foundation for risk control and mitigation priority is another. This was the main topic of research question 2 from which conclusions will be drawn now.

2. *Does controllable risk have an impact on residual incontrollable risk occurrence?*

An initial answer to this question is yes and no. Yes, there are certainly controllable risk factors that impact incontrollable risk occurrence. No: These effects are restricted to specific relation. There can be no generalization of all controllable risk factors that impact incontrollable risk occurrence.

The basic and residual controllable risks were grouped together for application and collaboration factors and another factor analysis was performed. The main reason was to further remove redundancies on risk factors that are controllable and just take the "controllable" concept as one for application and collaboration. Two factor solutions were thus created. The controllable application risk factor solution leads to five distinctive factors:

- "Process compliance"
- "Process similarity"
- "Simplicity of the retained application"
- "Level of tool support"
- "Consistent / sufficient use of tool support"

The naming of the first two factors was difficult. Generally speaking, the first factor consisted of compliance to CMMI processes, but there were also some "similarity of processes" variables. The other factors could be interpreted quite meaningful and forthcoming. The controllable collaboration risk factor solution also leads to five factors:

- "Relief of collaboration efforts"
- "Front-office project management"
- "Experience,"
- "Communication tools"
- "Liaisons"

It proved difficult naming the first factor again. The highest loading variables in this factor are access to application documentation and access to contextual business knowledge, but also collaboration training is involved. The overall resemblance is that all variables should lead to a relief of collaboration efforts. The other factors could be given a meaningful interpretation rather forthcoming. The two factor solutions were used as input for a regression analysis with the incontrollable risk factor solutions developed in the first part of the study.

H2a

Evidence was found for a reduction of "application complexity", based on "process compliance". Complying with CMMI processes in place, leads to less exceeding application complexity expectations. Moreover, an effect was found of the "use of tool support", "simplicity of the retained application" and the "consistent/sufficient use of tool support" on exceeding expectations on "human skills". Having and using mature engineering and testing tool support seems to simplify work and reduce the necessity for human skills of individual team members in the development project.

H2b
A controllable collaboration environment has a large impact on exceeding collaboration expectations. "Relief of collaboration efforts" (e.g. documentation, training) has a large impact on "scope of work clarity" and a more modest effect on "collaboration efficiency". Using a mature project management method and complying with it in the front-office has an effect on "collaboration efficiency". Since "collaboration efficiency" on its turn has a large impact on project success, it could be wise to invest in mature project management methods and more importantly: complying with them. The availability and use of "communication tools" has an impact on "scope of work clarity" and "collaboration efficiency". Investing in advanced video-conferencing tools and integrating them in projects seems to have profound positive implications for the project. It creates more willingness to accommodate each others objectives, creates transparency, increases speed of dealing with issues and clarifies the scope of work. Finally, the use of "liaisons" was not found to have a significant impact on any of the aggregate factors, but a modest (non-significant) impact was found on exceeding expectations with regard to native language and different communication customs. Using liaisons in the team might thus still be a good strategy to reduce these kinds of risks.

H2c
The relation between application risk factors and collaboration risk was also researched. The effect of controllable application risk factors on incontrollable collaboration risk resulted in two main findings. First the compliance to CMMI processes has a rather large positive effect on "collaboration efficiency". It seems that complying with the identified processes removes the unnecessary efforts in coordinating between front-office and back-office. All team members work transparently according to formalized working procedures which makes the collaboration presumably more to the point instead of coordinating overhead. Also the existence of high level engineering and testing tool support seems to clarify the scope of work. Potential problems in clarity of work are apparently reduced, since team members are enabled to use specific and detailed tools that make transparent, potential differences in common understanding of the application.

H2d
The final interaction researched was between controllable collaboration risk factors and incontrollable application risk. Only one effect was revealed between "relief of collaboration efforts" and exceeding "human skills" expectations. A relief of collaboration efforts reduces the collaborative dependencies between front-office and back office team and reduces the pressure placed on human skills to cope with this complexity in the development project.

Overall conclusion H1 and H2
To review: What is the impact of operational risks in offshore application development projects and how can they be mitigated, Most importantly it seems to be to:

Make life as simple as possible for all team members

Collaborate efficiently: It seems that "collaboration efficiency" (speed of dealing with issues, collaboration skills, commitment, consistency in team member objectives, team cohesion) surpasses expectations in the project and has on of the largest combined impact on project success. Focusing on the factors that should get the highest priority, this can be done best in making sure to work according to mature project management and process standards and comply with them. Also, attention should be given to relieving collaboration efforts by providing the necessary contextual application and business documentation to team members. They don't have all required (contextual) business knowledge. The availability and use of sophisticated communication tools has shown to be almost equally effective.

Clarify the scope of work: Efficient collaboration is not enough. Make sure that the scope of work is clear for all team members. Most effective actions to be taken are to make sure there is sufficient documentation available for reference. Moreover, the level of engineering / testing tool support is very important to force the team to get the scope of work right. Also: make use of high level synchronous communication tools. They provide the communication rich means to get the scope of work clear.

Reduce pressure on human skills: The final aspect is more a consequence than a cause. Is it no wise to dedicate valuable human resources to inefficient or coordinating overhead communication. Using high level engineering or testing tool support, documentation availability for the back-office and sufficient team training reduces this pressure on human skills.

Be aware of language and customs differences: Team members in a distributed team should not underestimate the impact of language and communication customs issues. They have shown to have a profound impact on project success. No *significant* mitigation actions could be found, but the use of liaisons in the team as well a relief of collaboration efforts (documentation, collaborative training) seemed to have a modest positive effect. This backs up the relevance of obtaining a common understanding on what is supposed to be developed. Having the right documentations reduces necessity for communicating this information through a medium, exposed to misunderstandings of language and customs.

Also in the technical application environment, mitigation actions can thus be found. It has been shown that process and project management compliance has a positive effect as well as the level of sophistication in application engineering tool support. The practical feasibility of risk prioritization should always be combined with common sense of the order of prosecution risk mitigation actions as well as the financial picture of these actions. The latter two aspects were left out of scope in this thesis.

3. *How can an offshore readiness assessment be improved to include the results of the previous questions?*

The final research question completes the feedback loop to the main trigger this project started with. A rough attempt was made to adapt the weighing factors of the offshore readiness assessment (RAS) by mapping risk variables to the constructs of RAS and assigning new weighing factors to the subdimensions based on the relative impacts found in the first two research questions. The weighing factors in the RAS' derivative that was constructed provide the means to more accurately assess an IT organization preparing for offshore outsourcing.

The approach used was based on the spearman Rho correlations and occurrence found in the basic descriptive information on risk variables in figure 16 and 17. It should be noted though that the new RAS' derivative is solely suited for future application development projects. The current focus of RAS is both application development as well as maintenance. Most important conclusion to be drawn based on the new weighing factors is that there is an underestimation of application complexity, interaction complexity, knowledge management, Methods and business logic complexity in the original RAS. These five elements have the resemblance of having to do with "exchanging information". The business logic complexity is difficult to be transferred over distance; the knowledge management provides the means for dealing with the exchange of this information. Methods can provide the common ground of working and dealing with application complexity.

Another contribution that can be brought to RAS; more specifically, just after a RAS assessment is the evidence on impact of controllable risk factors on incontrollable risks expressed in the second research question. This provides the outsourcing IT organization with knowledge to make an informed decision with regard to risk prioritization. If the RAS assessment at an IT-organization would result in a number of risk areas, an important next step would be to start the discussion where to focus resources and efforts on to make the potential future application development project a success. Using the risk mitigation information revealed by answering the second research question can provide the foundation for this type of risk control. Thereby an IT organization can evaluate and balance their options to increase the chance of a successful project.

9 Discussion

This final chapter identifies a number of limitating considerations that were taken in this research, discusses the external validity of the research and addresses a number of future research options.

Limitations

There are some limitations in this research. The most important limitating consideration made was the choice of respondents. The project-centric approach decision was made to use actual real-life risk occurrences instead of perceived risks, but an important side-effect of this choice was the isolated project manager perspective. An alternative would be to make multiple stakeholders participate per project. However, the project centric-approach is a very delicate matter. The advantage or even the necessity of guaranteeing strict anonymity of filling in the survey was the most important aspect of this study. Two options remained. First, we could simply distribute the survey over all people that had experience in and offshore outsourced application development project. This was rejected since there would be a very high chance of the same projects filled out by more team members. When the projects are anonymous, a similar project filled out by ten team members would completely distort the comparative power if another project was filled out by only two team members. Another option was to assign a project to the team members in it. This would indicate that (1) the anonymity would be gone and (2) the average team member would only have information about a subset of the questions, thereby overestimating or underestimating certain aspects. The Project manager was chosen for his generic and comparable overview. It seems that the data of project managers lead to enough discriminative power to draw some meaningful conclusions.

On a methodological level, like briefly addressed in the thesis, the RAS-constructs have a different scope than the research constructs. Where RAS focuses on an existing application, our research was focused on new functionality in new applications. Moreover RAS aims also at the inherent basic complexity, while our research constructs were focused on the residual effects measured in exceeding expectations based on this inherent complexity. A translation should thus still be made to the inherent complexity and the exceeding complexity expectations. Therefore the translation to the new weighing factors, although scoped to new development should be interpreted with caution. Not in the very least due to the little number of variables that could be mapped on every RAS subdimension.

On the statistical level, some limitations were present. The most obvious limitation is the relative small sample size of 44. More projects would increase the reliability with which statements can be made. At the individual risk variable level, strange distributions like for example in the experience variables were noticed. These could well have had a (modest) distorting effect.
Some KMO-Bartlett's tests did not meet the required values to interpret a "meaningful" factor solution, but fortunately these indicators are not strictly required to perform a factor analysis. The factor-solutions were necessary for data reduction and recoding in an interval scale with approximate normal distribution to meet the requirements of linear regression, but in this data reduction some unique variable variance was lost.

External validity

This study was performed from an IT consulting firm, with predominantly projects performed from the same firm. More-over the majority of projects entailed a front-office in the Netherlands and a back-office in India. The sample is therefore no full accurate representation of all offshore application development projects which pressures the external validity of this work. However, the risk variables that were integrated in the survey were, a few exceptions excluded, did not refer to specific nationalities. The questions were generic and stressed the distributedness in general of the project. There is thus no reason that the impacts would be very different in another setting. For example: different communication customs would be different with a Chinese back-office, but the essence of the risk would not change.

The results of this research are thus a bit colored; focusing on an Indian back-office, but the generic patterns collaboration efficiency, scope of work clarity would probably be generic, as will be the mitigation actions that might affect them. In short: the generic effects are presumably externally valid; the specific risk variable centric impacts are not.

Future research
This work can be extended and built upon in many ways. Five options are briefly discussed.

From a practical perspective, the most interesting follow-up study would be to delve into to relation of inherent risk towards exceeding complexity. Like addressed before, the actual complexity of a new or existing application was not researched, because the project manager would not be a suitable respondent. An interesting research though, to complete the 2*2 categorization of risks would be to examine the relation between a highly complex application (legacy, loads of interfaces etcetera) and exceeding complexity expectations. This would further support the RAS assessment and create even more delicacy and risk weighing opportunities.

Another research would be to build up the RAS assessment very differently. The focus of RAS is both application development and application maintenance. A proposed improvement research would be to make more RAS derivatives and work with layered scenarios of RAS. An assessment base of questions and risk weights could be customized for every scenario of outsourcing (like for example development, maintenance or infrastructure management). This would be in line with for example Beulen et al. (2006) who distinguish between application development outsourcing and infrastructure management as having different risks that apply to the alternatives. Ideally, all offshore projects from every kind of offshore outsourcing should be iteratively kept track of and included in the knowledge base. The outcomes of a project or continuing outsourcing service can then be calibrated into this knowledge base and further improve existing knowledge and better prepare an IT organization for offshore outsourcing.

Another research option, more related to this one is a follow up research using a policy capture (vignette study) technique. The basis behind such a research would be to "control" or polarize the occurrence of risk factors in a short descriptive text (the vignette). Controlling risk occurrence can be deliberately simulated by manipulating vignette descriptions. Because a foundation for risk impact and mitigation options has been laid here, it would be interesting to enlarge knowledge in the measured effects. The best or worst predictors for project success can be isolated in a policy capture design where experts (sourcing consultants or project managers) can fill out perceived effects on (different aspects of) project success based on the vignette descriptions. This would give extra supporting evidence for results of this research and might demystify the lack of project success impact of for example experience and requirements complexity.

An option would also be to perform a follow-up research and go in depth with regard to every risk variable. The approach of this thesis was mostly explorative and focused on broad range of different risks. Every defined risk category could be researched in isolation and depth to get an even better grasp of potential underlying concepts and mechanisms that determine success of failure of an offshore outsourced application development project.

The final and perhaps most interesting future research option could be to really make a distinction between offshore and onshore projects where one can look at similarities and dissimilarities in risk profile. This research has provided a useful approach to split project risks in application risk and collaboration risk. Such a distinction enables the split of collaborative elements that give the offshore project its unique character. My research focuses exclusively though on offshore distributed projects, so no more than indications on different risk profiles could be given. A structured comparison is difficult, but would lead to a unique contribution to the scientific and practical outsourcing field.

10 References

Amberg, M., & Wiener, M. (2005). "Lessons learned in IT Offshoring", *Proceedings of ISOneWorld 2005,* Nevada, USA, 30 March - 01 April 2005.

Amberg, M., Schröder, M.,& Wiener, M. (2005). "Competence-Based IT Outsourcing – An Evaluation of Models for Identifying and Analyzing Core Competences", *Paper presented at the 11th Americas Conference on Information Systems,* Omaha, USA, 11-14 August 2005.

Aron, R., & Singh, J.V. (2005). "Getting Offshoring Right", *Harvard Business Review, 83*(12), 135-143.

Aspray, W., Mayadas, F. &, Vardi, M.Y. (2006). "Globalization and offshoring of software. A report of the ACM job migration task force", Association for Computing Machinery. Retrieved: June 03, 2007 from: http://www.acm.org/globalizationreport.

Aubert, B.A., Patry, M., & Rivard, S. (1998). "Assessing the Risk of IT Outsourcing", *Proceedings of the 31th Annual Hawaii International Conference on System Sciences (HICSS'98),* Volume 6, p685.

Baarde, D.B., & Goede M.P.M (2001) "Basisboek Methoden en Technieken", Groningen, Stenfert Kroese

Bahli, B., & Rivard, S. (2001). "An assessment of information technology outsourcing risk", *Proceedings of the 22nd International Conference on Information Systems (ICIS),* New Orleans, USA, 16-19 December 2001, 575-580

Bahli, B., & Rivard, S. (2003). "The information technology outsourcing risk: a transaction cost and agency theory-based perspective", *Journal of Iinformation Technology 18*(3), 211-221.

Bahli, B.,& Rivard, S. (2005). "Validating measures of information technology outsourcing risk factors", *Omega: the international journal of management science, 33*(2), 175-187.

Balaji, S., & Ahuja, M. (2005). "Critical Team-Level Success Factors of Offshore Outsourced Projects: A Knowledge Integration Perspective", *Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS'05),* Volume 1, 52.2.

Balaji, S., & Brown, S.A. (2005b). "Strategic IS Sourcing and Dynamic Capabilities: Bridging the Gap", *Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS'05),* Volume 8, 260b.

Balaji, S., Ahuja, M., & Ranganathan, C. (2006). "Offshore Software Projects: Assessing the Effect of Knowledge Transfer Requirements and ISD capability", *Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS'06),* Volume 8, 199a.

Barki, H., Rivard, S., & Talbot, J. (1993). "Towards an Assessment of Software Development Risk", *Journal of Management Information Systems, 10*(2), 203-225.

Beulen, E. (2007). "The management of global sourcing partnerships: implications for the capabilities and skills of the IS function", *paper presented at the first Information Systems Workshop on Global Sourcing Services*, Val d'Isère, France, 13-15 March 2007.

Beulen, E., Van Fenema, P., & Currie, W. (2005). "From application outsourcing to infrastructure management: extending the offshore outsourcing service portfolio", *European Management Journal*, *23*(2), 133-144.

Bhat, J.M., Gupta, M., & Murthy, S.N. (2006). "Overcoming Requirements Engineering Challenges: Lessons from Offshore Outsourcing", *IEEE Software, 23*(5), 38-44.

Bhattacharya, S., Behare, R.S., & Gundersen, D.E. (2003). "Business risk perspectives on information systems outsourcing", *International Journal of Accounting Information Systems, 4*(1), 75-93.

Boehm, B.H. (1991). "Software risk management: principles and practices", *IEEE software, 8*(1), 32-42.

Boehm., B. (2000). "Safe and Simple Software Cost Analysis", *IEEE Software* 17(5), 14-17.

Boehm., B., Abts, C., Brown, A.W., Chulani, S., Clark, B.K., Horowitz, E., Madachy, R., Reifer, D., & Steece, B. (2000b). "*Software Cost Estimation with Cocomo II,* Prentice Hall, New York.

Business Informatics (2007). Utrecht University Master program website. Retrieved: June 18, 2007 from: http://businessinformatics.nl

Capgemini (2006). "European CIO survey – views on IT delivery 2006", *consulting services internal publication.* Utrecht

Capgemini (2007). "Corporate website" Retrieved multiple times April – July 2007 from: http://www.capgemini.com

Capgemini (2007b). "Capgemini RightshoreTM website" Retrieved: June 18, 2007 from: http://www.capgemini.com/collaboration/rightshore/

Carmel, E. (1999). *Global Software Teams.* Upper Saddle River, NJ: Prentice Hall.

Carmel, E., & Agarwal, R. (2001). Tactical Approaches for Alleviating Distance in Global Software Development", *IEEE Software, 18*(2), 22-29.

Carmel, E., & Agarwal, R. (2002). "The Maturation of Offshore Sourcing of Information Technology Work", *MIS Quarterly Executive*, *1*(2), 65-77.

Carmel, E., & Tjia, P. (2005). "*Offshoring Information Technology: Sourcing and Outsourcing to a Global Workforce",* Cambridge: Cambridge University Press.

Carr M. J., Konda S.L., Monarch I., Ulrich F.C., & Walker C.F. (1993). "Taxonomy-Based Risk Identification" *Technical Report CMU/SEI–93-TR-6, Software Engineering Institute*, Carnegie Mellon University.

Casey, V., & Richardson, I. (2006). "Project Management within Virtual Software teams", *Proceedings of the IEEE international conference on Global Software Engineering (ICGSE'06),* 33-42.

Center for Systems and Software Engineering; CSE (2007). "Introduction text Cocomo II" Retrieved August 27, 2007 from: http://sunset.usc.edu/research/COCOMOII

Chandrasekaran, N., & Ensing, G. (2004). "ODC: A Global IT Services Delivery Model", *Communication of the ACM, 47*(5), 47-49.

CMMI product team (2006). "CMMI for Development version 1.2", Software Engineering Institute, Carnegie Mellon University. Retrieved June 26, 2007 from: http://www.sei.cmu.edu/publications/documents/06.reports/06tr008.html

Conchuir, E.O., Holmstrom, H., Agerfalk, P.J., & Fitzgerald, B. (2006). "Exploring the assumed benefits of global software development", *Proceedings of the IEEE International conference on Global Software Engineering (ICGSE'06),* 159-168.

Cramton, C.D. (2001). "The mutual knowledge problem and its consequences for dispersed collaboration". *Organization Science, 21*(3), 346-371.

Currie, W.L. (2003). "A knowledge-based risk assessment framework for evaluating web-enabled application outsourcing projects", *International Journal of Project Management, 21*(3), 207-217.

Cusick, J., & Prasad, A. (2006). "A Practical Management and Engineering Approach to Offshore Collaboration", *IEEE Software, 23*(5), 20-29.

Damian, D. &, Moitra, D. (2006). "Global software development: how far have we come?", *IEEE Software*, *23*(5), 17-19.

Damian, D., & Chisan, J. (2006). "An Empirical Study of the Complex Relationships between Requirements Engineering Processes and Other Processes that lead to Payoffs in Productivity, Quality and Risk Management", *IEEE Transactions on Software Engineering 32*(7), 433-453.

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, 35*(4), 6-102.

Dibbern, J., Winkler, J., & Heinzl, A. (2006) "Offshoring of application services in the banking industry – A transaction cost analysis", Working paper in Information Systems 1, Retrieved: May 21, 2007 from: http://wifo1.bwl.uni-mannheim.de/fileadmin/files/publications/Working_Paper_16-2006.pdf

Dibbern, J., Winkler, J., & Heinzl, A. (2007). "Explaining Variations in Client Extra Costs between Software Projects Offshored to India", Working paper in Information Systems 1, Retrieved: May 21, 2007 from: http://wifo1.bwl.uni-mannheim.de/fileadmin/files/publications/Working_Paper_3-2007.pdf

Dixon, K., & Panteli, N. (2007). "The Strength of Virtuality in Teams: Social Capital built on "Weak Ties", *Proceedings of the 40th Annual Hawaii International Conference on System Sciences (HICSS'07),* 43b

Djavanshir, G.R. (2005). "Surveying the risks and benefits of IT outsourcing", *IEEE Computer Science 7*(6), 32-37.

Erber, G., & Ahmed, A.S. (2005). "Offshore outsourcing – A global shift in the Present IT Industry", *Intereconomics, 40*(2), 100-112.

Erickson, J.M., & Ranganathan, C. (2006). "Project Management Capabilities: Key to Application Development Offshore Outsourcing", *Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS'06),* Volume 8, 199b

Evaristo, J.R., Scudder, R., Desouza, K.C., & Sato, O. (2004). "A dimensional analysis of geographically distributed project teams: a case study"*, Journal of Engineering Technology and Management, 21*(3), 175-189.

Favela, J., & Pena-Mora, F. (2001). "An experience in Collaborative Software Engineering Education", *IEEE Software, 18*(2), 47-53.

Feeny, D., & Willcocks, L. (1998). "Core IS Capabilities For Exploiting Information Technology", *Sloan Management Review, 39*(3), 9-21.

Feiman, J. (2005). "Globally distributed application development: Concepts, Tools and Practices", *Gartner Dataquest*, G00136330.

Fowler, F.J. (1995). "*Improving Survey Questions: Design and Evaluation",* Thousand Oaks, CA, SAGE

Friedman, T.L. (2005). "*The world is flat: a brief history of the globalized world in the 21st century".* Londen, Penguin Books.

Garson, D.G. (2007) Factor Analysis, Statistical syllabus, North Carolina State University: Retrieved September 4 from:http://www2.chass.ncsu.edu/garson/pa765/factor.htm

Gonzales, R., Gasco, J., & Llopis, J. (2004). "A study of information systems outsourcing risks", *Paper presented at the 13th European Conference on Information Systems (ECIS),* Turku, Finland, 14-16 June, 2004

Gopal, A., Mukhopadhyay, T., & Krishnan, S. (2002) "The role of Software Processes and Communication in Offshore Software Development", *Communications of the ACM, 45*(4), 193-200.

Harmsen, F., Brand, M., Hillegersberg, J.& Aydin, M.N. (2007). "Agile methods for offshore information systems development", *paper presented at the first Information Systems Workshop on Global Sourcing Services*, Val d'Isère, France, 13-15 March 2007.

Herbsleb, J.D., & Grinter E. (1999). "Splitting the Organization and Integrating the Code: Conway's Law revisited", *Proceedings of the 21st international conference on Sofware Engineering (ICSE'99),* 85-95.

Herbsleb, J.D., & Mockus, A. (2003). "An Empirical Study of Speed and Communication in Globally Distributed Software Development", *IEEE Transactions on Software Engineering, 29*(6), 481-494.

Herbsleb, J.D., & Moitra D. (2001). "Global Software Development", *IEEE Software 18(2)* 16-20.

Hirschheim, R. A. & Lacity, M. C. (2000). "The Myths and Realities of Information Technology Insourcing", *Communications of the ACM*, *43*(2), 99-107.

Hofstede, G. (1996). *Culture and organizations: software of the mind*. New York, McGraw-Hill.

Institute of Electrical and Electronics Engineers (1990). *IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries*. New York, NY.

Jashapara, A. (2004). "*Knowledge Management: An integrated approach"*, Essex, Prentice Hall.

Jiang, J., & Klein, G. (2000). "Software development risks to project effectiveness", *The Journal of Systems and Software, 52*(1), 3-10.

Kachigan, S. (1991). "*Multivariate Statistical Analysis: A Conceptual Introduction (2nd edition)",* New York, Radius Press

Keil, M., Cule, P.E., Lyytinen K., & Schmidt, R.C. (1998). "A framework for identifying software project risks", *Communications of the ACM 41*(11), 76-83.

Khan, N., Currie, W.L., Weerakkody, V., & Desai, B. (2003) "Evaluating Offshore IT Outsourcing in India: Supplier and Customer Scenarios", *Proceedings of the 36th Annual Hawaii International Conference on System Sciences (HICSS'03),* Volume 8, 239.1.

Kobitzsch, W., Romback, D., & Feldmann, R. (2001). "Outsourcing in India", *IEEE Software, 18*(2), 78-86.

Kotlarsky, J. (2005). "Management of globally distributed Component-Based Software Development Projects", *Ph.D. dissertation,* Erasmus University Rotterdam.

Kraut, R.E., & Streeter, L.A. (1995). "Coordination in software development", *Communications of the ACM 38*(3), 69-81.

Kshetri, N. (2007). "Institutional factors affecting offshore business process and information technology outsourcing", *Journal of International Management, 13*(1), 38-56.

Kuni, R., & Bhushan, N. (2006). "IT Application Assessment Model for Global Software Development", *Proceedings of the IEEE international conference on Global Software Engineering (ICGSE'06),* 92-100.

Layman, L., Williams, L., Damian, D., & Bures, H. (2006). “Essential Communication practices for Extreme Programming in a global software development team”, *Information and Software Technology, 48*, 781-794.

Lee, J. (2001). “The impact of knowledge sharing, organizational capability and partnership quality on IS outsourcing success”, *Information & Management, 38*(5), 323-335.

Levina, N., & Ross, J.W. (2003). "From the Vendor's Perspective: Exploring the Value Proposition in IT Outsourcing" *MIS Quarterly 27*(3), 331-364.

Lewin, A.Y., & Peeters, C. (2006). “Offshoring Work: Business Hype or the Onset of Fundamental Transformation”, *Long Range Planning, 39*(3), 221-239.

Loh, L., & Venkatraman, N. (1992). “Determinants of information technology outsourcing”, *Journal of Management Information Systems, 9*(1), 7-24.

Loh, L., & Venkatraman, N. (1995). “An empirical study of Information Technology Outsourcing: Benefits, Risks, and Performance Implications”, *Proceedings of the 16th International Conference on Information Systems (ICIS95)*, 277-288.

Looff, de (1996). *“A model for information systems outsourcing decision making”,* Idea Group Publishing, Hershey, USA.

McCarthy, J.C., Martorelli, W., Moore, S., Agosta, L., & Ross, C.F. (2004). “Offshore Outsourcing: The Complete Guide”, *Forrester Research Collection.*

Mirani, R. (2007). “Procedural coordination and offshored software tasks: Lessons from two case studies”, *Information & Management, 44*(2), 216-230.

Na, K., Simpson, J.T., Li, X., Singh, T., & Kim, K. (2007). ”Software development risk and project performance measurement: Evidence in Korea”, *The Journal of Systems and Software, 80*(4), 596-605.

Nahapiet, J., & Ghoshal, S. (1998). “Social Capital, Intellectual Capital, and the Organizational Advantage”, *The Academy of Management Review, 23*(2), 242-266.

Nidumolu, S.R. (1996). “Standardization, Requirements uncertainty and software project performance”, *Information & Management 31*(3), 135-150.

Palvia, P.C. (1995). “A dialectic view of information systems outsourcing: Pros and cons”, *Information & Management, 29*(5), 265-275.

Penrose, E.T. (1959). “*The theory of the growth of the firm*” John Wiley and Sons, New York.

Pfleeger, S.L. (2000). “Risky Business: What we have yet to learn about risk management”, *The journal of Systems and Software 53*(3), 265-273.

Pries-Heje, J.P., Baskerville, R., & Hansen, G.L. (2005). ”Strategy Models for Enabling Offshore Outsourcing: Russian Short-Cycle-Time Software Development”, *Information Technology for Development, 11*(1), 5-30.

Prikladnicki, R., Audy, J.L.N., & Evaristo, R. (2003). "Global Software Development in Practice Lessons Learned", *Software Process: Improvement and Practice, 8*(4), 267-281.

Prikladnicki, R., Audy, J.L.N., & Evaristo, R. (2004). "An empirical study on Global Software Development: Offshore insourcing of IT Projects", *Proceedings of the International Workshop on Global Software Development, International Conference on Software Engineering (ICSE 2004),* 53-58.

Prikladnicki, R., Audy, J.L.N., & Evaristo, R. (2006). "A Reference Model for Global Software Development: Findings from a Case Study", *Proceedings of the IEEE international conference on Global Software Engineering (ICGSE'06),* 18-28.

Ramasubbu, N. Krishnan, M.S., & Kompalli, P. (2005). "Leveraging Global Resources: A Process Maturity Framework for Managing Distributed Development", *IEEE Software, 22*(3), 80-86.

Sabherwal, R. (2003). "The evolution of coordination in outsourced software development projects: a comparison of client and vendor perspectives", *Information and Organization, 13*(3), 153-202.

Sakthivel, S. (2005). "Virtual workgroups in offshore systems development", *Information and Software Technology, 47*(5), 305-318.

Shami, N.S., Bos, N., Wright, Z., Hoch, S., Kuan, K. Y., Olsen, J., & Olsen G. (2004). "An experimental simulation of multi-site software development", *Proceedings of the 2004 conference of the Centre for Advanced Studies on Collaborative research",* 255-266.

Shapiro, S.S. & Wilk, M.B. (1965). "An analysis of variance test for normality (complete samples)", *Biometrika*, *52*(3/4), 591-611.

Singh, S., & Zack, M.H. (2006). "Information Technology Outsourcing: Reducing costs or knowledge?", *Paper presented at the International Conference on Organizational Learning, Knowledge and Capabilities(OLKC'06},* University of Warwick, UK, 20-22 March 2006.

Smith, M.A., Mitra, S., & Narasimhan, S. (1996). "Offshore outsourcing of software development and maintenance: A framework for issues", *Information & Management, 31*(3), 165-175.

Stack, M., & Downing, R. (2005). "Another look at offshoring: Which jobs are at risk and why?",*Business Horizons, 48*(6), 513-523.

Stark, J., Arlt, M., & Walker, D.H.T. (2006). "Outsourcing Decisions & Models – Some Practical Considerations for Large Organizations", *Proceedings of the IEEE international conference on Global Software Engineering (ICGSE'06),* 12-17.

Steenbeek, W., Wijngaert, L., Brand, M., Brinkkemper, S., & Harmsen, F. (2005). "Sourcing decision making: Elicitating consultancy knowledge using policy capturing", *Paper presented at the 13th European Conference on Information Systems: Information Systems in a Rapidly Changing Economy,* Regensburg, Germany, 26-28 May 2005.

Teece, D.J., Pisano, G., & Shuen, A. (1997). "Dynamic Capabilities and Strategic Management", *Strategic Management Journal*, *18*(7), 509-533.

Tiwana, A. (2004). "Beyond the Black Box: Knowledge Overlaps in Software Outsourcing", *IEEE Software, 21*(5), 51-58.

Tramacere, G., & Marriott, I. (2005). "User Survey: Offshore outsourcing, Western Europe, 2004-2005", *Gartner Dataquest,* G00127878.

Tsotra, D., & Fitzgerald, G. (2007). "The role of culture in global IS/IT sourcing", *paper presented at the first Information Systems Workshop on Global Sourcing Services*, Val d'Isère, France, 13-15 March 2007.

Urdan, T.C. (2001). "*Statistics in plain English*", Mahwah, New Jersey, Lawrence Erlbaum Association.

Weerd, I., Brinkkemper, S. (2007). "Meta-modeling for situational analysis and design methods" *To appear in the Handbook of Research on Modern Systems Analysis and Design Technologies and Applications*, Idea Group Publishing, Hershey, PA, USA.

Wernerfelt, B. (1984). "A Resource-based view of the firm", *Strategic Management Journal, 5*(2), 171-180.

Wijk, C. (2000). "*Toetsende statistiek: basistechnieken",* Bussum, Coutinho,

Willcocks, L.P., & Lacity, M.C. (2006). *"Global Sourcing of Business & IT Services",* Hampshire: Palgrave Macmillan Ltd.

Williamson, O.E. (1981). "The Economics of Organizations: The Transaction Cost Approach", *The American Journal of Sociology, 87*(3), 548-577.

Yang, C., & Huang, J.B. (2000). "A decision model for IS outsourcing", *International Journal of Information Management, 20*(3), 225-239.

Young, A. &, Potter, K. (2006). "User Survey Report: Strategies for IT Outsourcing, Worldwide", *Gartner Dataquest*, G00141315.

11 Appendix

11.1 SPSS output on regression analyses

11.1.1 H1 categorized risk impact on project success

H1a: Basic controllable application risk factors

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	,505[a]	,255	,219	,73765

a. Predictors: (Constant), Factor 2: Architecture standardization / simplicity existing app, Factor 1: Process similarity

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	7,655	2	3,827	7,034	,002[a]
	Residual	22,309	41	,544		
	Total	29,964	43			

a. Predictors: (Constant), Factor 2: Architecture standardization / simplicity existing app, Factor 1: Process similarity

b. Dependent Variable: Project success

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2,523	,111		22,688	,000
	Factor 1: Process similarity	,289	,112	,346	2,571	,014
	Factor 2: Architecture standardization / simplicity existing app	,307	,112	,368	2,731	,009

a. Dependent Variable: Project success

H1b: Residual controllable application risk factors

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	,473[a]	,223	,185	,75340

a. Predictors: (Constant), Consistent/Sufficient use of tool support, Process compliance

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	6,692	2	3,346	5,895	,006[a]
	Residual	23,272	41	,568		
	Total	29,964	43			

a. Predictors: (Constant), Consistent/Sufficient use of tool support, Process compliance

b. Dependent Variable: Project success

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients		
		B	Std. Error	Beta	t	Sig.
1	(Constant)	2,523	,114		22,214	,000
	Process compliance	,301	,115	,360	2,618	,012
	Consistent/Sufficient use of tool support	,255	,115	,306	2,222	,032

a. Dependent Variable: Project success

H1c: Residual incontrollable application risk

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	,615[a]	,378	,331	,68273

a. Predictors: (Constant), Human skills, Requirements complexity, Application complexity

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	11,319	3	3,773	8,095	,000[a]
	Residual	18,645	40	,466		
	Total	29,964	43			

a. Predictors: (Constant), Human skills, Requirements complexity, Application complexity

b. Dependent Variable: Project success

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients		
		B	Std. Error	Beta	t	Sig.
1	(Constant)	2,523	,103		24,513	,000
	Application complexity	,157	,104	,188	1,508	,139
	Requirements complexity	,194	,104	,233	1,866	,069
	Human skills	,448	,104	,537	4,304	,000

a. Dependent Variable: Project success

H1d: Basic controllable collaboration risk factors

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	,465[a]	,216	,157	,76637

a. Predictors: (Constant), Experience, Project management maturity and alignment, Collaborative means

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	6,471	3	2,157	3,673	,020[a]
	Residual	23,493	40	,587		
	Total	29,964	43			

a. Predictors: (Constant), Experience, Project management maturity and alignment, Collaborative means

b. Dependent Variable: Project success

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2,523	,116		21,838	,000
	Collaborative means	,365	,117	,438	3,127	,003
	Project management maturity and alignment	9,777E-02	,117	,117	,837	,408
	Experience	-8,61E-02	,117	-,103	-,737	,466

a. Dependent Variable: Project success

H1e: Residual controllable collaboration risk factors

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	,339[a]	,115	,094	,79452

a. Predictors: (Constant), Project Management method compliance

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	3,451	1	3,451	5,467	,024[a]
	Residual	26,513	42	,631		
	Total	29,964	43			

a. Predictors: (Constant), Project Management method compliance

b. Dependent Variable: Project success

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2,523	,120		21,064	,000
	Project Management method compliance	,283	,121	,339	2,338	,024

a. Dependent Variable: Project success

H1f: Residual incontrollable collaboration risk

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	,683[a]	,467	,427	,63188

a. Predictors: (Constant), Language and Custom issues, Collaboration effciency, Scope of work clarity

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	13,993	3	4,664	11,682	,000[a]
	Residual	15,971	40	,399		
	Total	29,964	43			

a. Predictors: (Constant), Language and Custom issues, Collaboration effciency, Scope of work clarity

b. Dependent Variable: Project success

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2,523	,095		26,486	,000
	Scope of work clarity	,321	,096	,385	3,335	,002
	Collaboration effciency	,402	,096	,482	4,174	,000
	Language and Custom issues	,246	,096	,294	2,549	,015

a. Dependent Variable: Project success

11.1.2 H2 controllable risks impact on incontrollable risks

H2a: controllable application risk -> residual controllable application risk

Linear regression of Factors with Application complexity

Model Summary[b]

					Change Statistics				
Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	R Square Change	F Change	df1	df2	Sig. F Change
1	,460[a]	,212	,108	,94433176	,212	2,044	5	38	,094

a. Predictors: (Constant), application tool support consistent use, application tool support level, retained application simplicity, process level similarity, process consistent use

b. Dependent Variable: Application complexity

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	9,113	5	1,823	2,044	,094[a]
	Residual	33,887	38	,892		
	Total	43,000	43			

a. Predictors: (Constant), application tool support consistent use, application tool support level, retained application simplicity, process level similarity, process consistent use

b. Dependent Variable: Application complexity

Coefficients[a]

		Unstandardized Coefficients		Standardized Coefficients		
Model		B	Std. Error	Beta	t	Sig.
1	(Constant)	1,684E-16	,142		,000	1,000
	process consistent use	,303	,144	,303	2,106	,042
	process level similarity	-7,55E-02	,144	-,075	-,524	,603
	retained application simplicity	-7,01E-02	,144	-,070	-,486	,629
	application tool support level	-,244	,144	-,244	-1,693	,099
	application tool support consistent use	,223	,144	,223	1,551	,129

a. Dependent Variable: Application complexity

Linear regression of Factors with Requirements complexity

Model Summary[b]

					Change Statistics				
Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	R Square Change	F Change	df1	df2	Sig. F Change
1	,321[a]	,103	-,015	1,00744784	,103	,873	5	38	,508

a. Predictors: (Constant), application tool support consistent use, application tool support level, retained application simplicity, process level similarity, process consistent use

b. Dependent Variable: Requirements complexity

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	4,432	5	,886	,873	,508[a]
	Residual	38,568	38	1,015		
	Total	43,000	43			

a. Predictors: (Constant), application tool support consistent use, application tool support level, retained application simplicity, process level similarity, process consistent use

b. Dependent Variable: Requirements complexity

Coefficients[a]

Model		Unstandardized Coefficients B	Unstandardized Coefficients Std. Error	Standardized Coefficients Beta	t	Sig.
1	(Constant)	-6,56E-17	,152		,000	1,000
	process consistent use	-7,38E-02	,154	-,074	-,480	,634
	process level similarity	-,207	,154	-,207	-1,347	,186
	retained application simplicity	,220	,154	,220	1,429	,161
	application tool support level	7,802E-03	,154	,008	,051	,960
	application tool support consistent use	8,097E-02	,154	,081	,527	,601

a. Dependent Variable: Requirements complexity

Linear regression of Factors with human skills

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics: R Square Change	F Change	df1	df2	Sig. F Change
1	,587[a]	,345	,258	,86121094	,345	3,995	5	38	,005

a. Predictors: (Constant), application tool support consistent use, application tool support level, retained application simplicity, process level similarity, process consistent use

b. Dependent Variable: Human skills

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	14,816	5	2,963	3,995	,005[a]
	Residual	28,184	38	,742		
	Total	43,000	43			

a. Predictors: (Constant), application tool support consistent use, application tool support level, retained application simplicity, process level similarity, process consistent use

b. Dependent Variable: Human skills

Coefficients[a]

Model		Unstandardized Coefficients B	Unstandardized Coefficients Std. Error	Standardized Coefficients Beta	t	Sig.
1	(Constant)	-5,97E-17	,130		,000	1,000
	process consistent use	,172	,131	,172	1,309	,198
	process level similarity	,196	,131	,196	1,493	,144
	retained application simplicity	,334	,131	,334	2,545	,015
	application tool support level	,303	,131	,303	2,304	,027
	application tool support consistent use	,271	,131	,271	2,060	,046

a. Dependent Variable: Human skills

H2b controllable collaboration risk -> residual controllable collaboration risk

Linear regression of Factors with scope of work clarity

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics				
					R Square Change	F Change	df1	df2	Sig. F Change
1	,709[a]	,503	,438	,74980063	,503	7,697	5	38	,000

a. Predictors: (Constant), Liaisons, Communication tools, Experience, Front-office Project Management maturity / compliance, Relief of collaboration efforts

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	21,636	5	4,327	7,697	,000[a]
	Residual	21,364	38	,562		
	Total	43,000	43			

a. Predictors: (Constant), Liaisons, Communication tools, Experience, Front-office Project Management maturity / compliance, Relief of collaboration efforts

b. Dependent Variable: Scope of work clarity

Coefficients[a]

Model		Unstandardized Coefficients B	Unstandardized Coefficients Std. Error	Standardized Coefficients Beta	t	Sig.
1	(Constant)	-2,05E-17	,113		,000	1,000
	Relief of collaboration efforts	,602	,114	,602	5,263	,000
	Front-office Project Management maturity / compliance	,191	,114	,191	1,668	,104
	Experience	-5,69E-02	,114	-,057	-,497	,622
	Communication tools	-,232	,114	-,232	-2,031	,049
	Liaisons	-,218	,114	-,218	-1,904	,064

a. Dependent Variable: Scope of work clarity

Linear regression of Factors with collaboration efficiency

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics				
					R Square Change	F Change	df1	df2	Sig. F Change
1	,634[a]	,402	,324	,82237428	,402	5,116	5	38	,001

a. Predictors: (Constant), Liaisons, Communication tools, Experience, Front-office Project Management maturity / compliance, Relief of collaboration efforts

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	17,301	5	3,460	5,116	,001[a]
	Residual	25,699	38	,676		
	Total	43,000	43			

a. Predictors: (Constant), Liaisons, Communication tools, Experience, Front-office Project Management maturity / compliance, Relief of collaboration efforts

b. Dependent Variable: Collaboration effciency

Coefficients[a]

Model		Unstandardized Coefficients B	Unstandardized Coefficients Std. Error	Standardized Coefficients Beta	t	Sig.
1	(Constant)	1,527E-16	,124		,000	1,000
	Relief of collaboration efforts	,276	,125	,276	2,200	,034
	Front-office Project Management maturity / compliance	,333	,125	,333	2,654	,012
	Experience	6,279E-02	,125	,063	,501	,619
	Communication tools	,439	,125	,439	3,504	,001
	Liaisons	,136	,125	,136	1,083	,286

a. Dependent Variable: Collaboration effciency

Linear regression of Factors with language and customs issues

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics				
					R Square Change	F Change	df1	df2	Sig. F Change
1	,393[a]	,154	,043	,97837472	,154	1,384	5	38	,252

a. Predictors: (Constant), Liaisons, Communication tools, Experience, Front-office Project Management maturity / compliance, Relief of collaboration efforts

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	6,626	5	1,325	1,384	,252[a]
	Residual	36,374	38	,957		
	Total	43,000	43			

a. Predictors: (Constant), Liaisons, Communication tools, Experience, Front-office Project Management maturity / compliance, Relief of collaboration efforts

b. Dependent Variable: Language and Custom issues

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	-1,92E-16	,147		,000	1,000
	Relief of collaboration efforts	,271	,149	,271	1,814	,078
	Front-office Project Management maturity / compliance	-,172	,149	-,172	-1,156	,255
	Experience	-,110	,149	-,110	-,740	,464
	Communication tools	2,147E-02	,149	,021	,144	,886
	Liaisons	,196	,149	,196	1,314	,197

a. Dependent Variable: Language and Custom issues

H2c: Controllable application risk -> residual incontrollable collaboration risk

Linear regression of factors on scope of work clarity

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics				
					R Square Change	F Change	df1	df2	Sig. F Change
1	,508[a]	,258	,161	,91612003	,258	2,647	5	38	,038

a. Predictors: (Constant), application tool support consistent use, application tool support level, retained application simplicity, process level similarity, process consistent use

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	11,108	5	2,222	2,647	,038[a]
	Residual	31,892	38	,839		
	Total	43,000	43			

a. Predictors: (Constant), application tool support consistent use, application tool support level, retained application simplicity, process level similarity, process consistent use

b. Dependent Variable: Scope of work clarity

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	-6,30E-17	,138		,000	1,000
	process consistent use	,267	,140	,267	1,909	,064
	process level similarity	-,198	,140	-,198	-1,414	,165
	retained application simplicity	,198	,140	,198	1,421	,164
	application tool support level	,290	,140	,290	2,072	,045
	application tool support consistent use	-,158	,140	-,158	-1,131	,265

a. Dependent Variable: Scope of work clarity

Linear regression of factors on collaboration efficiency

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics				
					R Square Change	F Change	df1	df2	Sig. F Change
1	,505[a]	,255	,157	,91834747	,255	2,597	5	38	,041

a. Predictors: (Constant), application tool support consistent use, application tool support level, retained application simplicity, process level similarity, process consistent use

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	10,952	5	2,190	2,597	,041[a]
	Residual	32,048	38	,843		
	Total	43,000	43			

a. Predictors: (Constant), application tool support consistent use, application tool support level, retained application simplicity, process level similarity, process consistent use

b. Dependent Variable: Collaboration effciency

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2,207E-16	,138		,000	1,000
	process consistent use	,403	,140	,403	2,877	,007
	process level similarity	,121	,140	,121	,861	,395
	retained application simplicity	,213	,140	,213	1,519	,137
	application tool support level	-,111	,140	-,111	-,791	,434
	application tool support consistent use	,142	,140	,142	1,017	,315

a. Dependent Variable: Collaboration effciency

Linear regression of factors on language and communication customs

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics				
					R Square Change	F Change	df1	df2	Sig. F Change
1	,243[a]	,059	-,065	1,03184752	,059	,477	5	38	,791

a. Predictors: (Constant), application tool support consistent use, application tool support level, retained application simplicity, process level similarity, process consistent use

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	2,541	5	,508	,477	,791[a]
	Residual	40,459	38	1,065		
	Total	43,000	43			

a. Predictors: (Constant), application tool support consistent use, application tool support level, retained application simplicity, process level similarity, process consistent use

b. Dependent Variable: Language and Custom issues

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	-1,80E-16	,156		,000	1,000
	process consistent use	-4,48E-02	,157	-,045	-,284	,778
	process level similarity	-3,26E-02	,157	-,033	-,207	,837
	retained application simplicity	,179	,157	,179	1,137	,263
	application tool support level	,122	,157	,122	,776	,442
	application tool support consistent use	9,535E-02	,157	,095	,606	,548

a. Dependent Variable: Language and Custom issues

H2d: Controllable collaboration risk -> residual incontrollable application risk

Linear regression of factors on application complexity

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics				
					R Square Change	F Change	df1	df2	Sig. F Change
1	,293[a]	,086	-,034	1,01707176	,086	,714	5	38	,617

a. Predictors: (Constant), Liaisons, Communication tools, Experience, Front-office Project Management maturity / compliance, Relief of collaboration efforts

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	3,691	5	,738	,714	,617[a]
	Residual	39,309	38	1,034		
	Total	43,000	43			

a. Predictors: (Constant), Liaisons, Communication tools, Experience, Front-office Project Management maturity / compliance, Relief of collaboration efforts

b. Dependent Variable: Application complexity

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	1,164E-16	,153		,000	1,000
	Relief of collaboration efforts	-7,46E-02	,155	-,075	-,481	,633
	Front-office Project Management maturity / compliance	2,529E-02	,155	,025	,163	,871
	Experience	,140	,155	,140	,901	,373
	Communication tools	9,769E-02	,155	,098	,630	,533
	Liaisons	-,225	,155	-,225	-1,450	,155

a. Dependent Variable: Application complexity

Linear regression of factors on requirements complexity

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics				
					R Square Change	F Change	df1	df2	Sig. F Change
1	,295[a]	,087	-,033	1,01638900	,087	,725	5	38	,609

a. Predictors: (Constant), Liaisons, Communication tools, Experience, Front-office Project Management maturity / compliance, Relief of collaboration efforts

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	3,744	5	,749	,725	,609[a]
	Residual	39,256	38	1,033		
	Total	43,000	43			

a. Predictors: (Constant), Liaisons, Communication tools, Experience, Front-office Project Management maturity / compliance, Relief of collaboration efforts

b. Dependent Variable: Requirements complexity

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	-4,27E-17	,153		,000	1,000
	Relief of collaboration efforts	,115	,155	,115	,745	,461
	Front-office Project Management maturity / compliance	-9,35E-02	,155	-,094	-,603	,550
	Experience	5,745E-02	,155	,057	,371	,713
	Communication tools	-,167	,155	-,167	-1,080	,287
	Liaisons	,184	,155	,184	1,184	,244

a. Dependent Variable: Requirements complexity

Linear regression of factors on human skills

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics				
					R Square Change	F Change	df1	df2	Sig. F Change
1	,480[a]	,230	,129	,93333431	,230	2,272	5	38	,067

a. Predictors: (Constant), Liaisons, Communication tools, Experience, Front-office Project Management maturity / compliance, Relief of collaboration efforts

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	9,898	5	1,980	2,272	,067[a]
	Residual	33,102	38	,871		
	Total	43,000	43			

a. Predictors: (Constant), Liaisons, Communication tools, Experience, Front-office Project Management maturity / compliance, Relief of collaboration efforts

b. Dependent Variable: Human skills

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	-9,78E-17	,141		,000	1,000
	Relief of collaboration efforts	,436	,142	,436	3,061	,004
	Front-office Project Management maturity / compliance	,155	,142	,155	1,091	,282
	Experience	-5,27E-02	,142	-,053	-,370	,713
	Communication tools	,112	,142	,112	,790	,435
	Liaisons	-2,99E-02	,142	-,030	-,210	,835

a. Dependent Variable: Human skills

11.2 SPSS output on PCA Factor analyses

11.2.1 H1 factor solutions

Basic controllable application risks

Rotated Component Matrix[a]

	Component	
	1	2
Front-office and back-office process maturity for Requirements Management was similar.	,850	-1,46E-02
Front-office and back-office process maturity for Requirements Development was similar.	,828	-,128
Front-office and back-office process maturity for Configuration Management was similar.	,807	1,137E-02
Front-office and back-office maturity with regard to development method standards was similar.	,769	,225
Front-office and back-office process maturity for Causal Analysis and Resolution was similar.	,679	,562
Front-office and back-office process maturity for Project Monitoring and Control was similar.	,661	,320
Front-office and back-office process maturity for Measurement and Analysis was similar.	,654	,475
Front-office and back-office process maturity for Project Planning was similar.	,600	,505
The Service Level Agreement was highly detailed.	,546	,258
Testing tool support was highly advanced.	,465	,305
Application engineering tool support was highly advanced.	,446	,404
The application code of the existing application was self descriptive (with design rationale). If applicable	-,111	,856
The structure of the existing application was well structured. If applicable	3,028E-02	,837
Application architecture, was based on standards.	,312	,789
The programmers familiarity in the back-office with the existing application at the front-office was large. If applicable	,169	,634
Data architecture, was based on standards.	,441	,511

Extraction Method: Principal Component Analysis.
Rotation Method: Varimax with Kaiser Normalization.
a. Rotation converged in 3 iterations.

Residual controllable application risks

Rotated Component Matrix[a]

	Component	
	1	2
Requirements Management process standards were followed consistently in the project.	,905	7,936E-02
Configuration Management process standards were followed consistently in the project.	,895	9,990E-02
Requirements Development process standards were followed consistently in the project.	,867	,102
Project Monitoring and Control process standards were followed consistently in the project.	,854	,263
Measurement and Analysis process standards were followed consistently in the project.	,843	,163
Software Project Planning process standards were followed consistently in the project.	,732	,316
Causal analysis and Resolution process standards were followed consistently in the project.	,705	,416
All project members complied consistently to the used development method.	,533	9,511E-02
Application engineering tool support was used consistently.	,262	,891
Testing tool support was used sufficiently.	,145	,773
Application engineering tool support was used sufficiently.	6,832E-02	,716

Extraction Method: Principal Component Analysis.
Rotation Method: Varimax with Kaiser Normalization.

a. Rotation converged in 3 iterations.

Residual incontrollable application risks

Rotated Component Matrix[a]

	Component		
	1	2	3
The technical architecture was more complex than expected.	,871	,180	9,940E-02
The internal interfaces were more complex than expected.	,827	-5,86E-02	,102
The module diversity was larger than expected.	,823	,134	6,962E-02
The data architecture was more complex than expected.	,812	8,517E-02	-5,47E-02
The hardware platform was more complex than expected.	,732	-8,28E-02	,126
The data transfer security requirements in the application were more stringent than expected.	,707	,300	,102
The external interfaces were more complex than expected.	,703	7,068E-02	-9,81E-02
The business logic was more complex than expected.	,681	7,828E-02	-6,98E-02
The software platform was more complex than expected.	,596	,324	,268
The level of initial technical detailledness in the application requirements was smaller than expected.	-7,22E-02	,887	6,115E-02
The amount of technical precedented requirements (similarity in application requirements) was smaller than expected.	-5,24E-02	,861	-4,07E-03
The level of initial functional detailledness in the application requirements was smaller than expected.	,110	,834	3,359E-02
The amount of functional unwritten application requirements was higher than expected.	,108	,826	-5,82E-02
The amount of technical unwritten application requirements was higher than expected.	,127	,746	5,938E-03
Stability in the functional application requirements was smaller than expected.	,195	,741	9,021E-02
Stability in the technical application requirements was smaller than expected.	,258	,416	9,635E-02
Application coding skills in the other team were according to or above expectations in the project.	-9,78E-02	6,044E-02	,815
Architecture design skills in the other team were according to or above expectations in the project.	,117	,144	,754
Requirements development skills in the other team were according to or above expectations in the project.	4,870E-02	,132	,683
Deployment skills in the other team were according to or above expectations in the project.	,406	-8,21E-02	,635
Software testing skills in the other team were according to or above expectations in the project.	-4,56E-02	-,103	,632

Extraction Method: Principal Component Analysis.
Rotation Method: Varimax with Kaiser Normalization.

a. Rotation converged in 4 iterations.

Basic controllable collaboration risks

Rotated Component Matrix[a]

	Component		
	1	2	3
Back-office team members had access to accurate English contextual business knowledge to be captured in the application.	,906	,140	-2,19E-02
Team members had access to accurate English application documentation.	,840	-4,50E-02	9,163E-02
The project-team worked with liaisons.	,558	,134	6,590E-02
High-level Synchronous communication tools between front-office and back-office were available.	,503	7,387E-02	7,614E-02
Front-office Project Management was conducted according to a highly mature Project Management standard.	,127	,850	2,870E-04
There were no initial differences between the front-office and back-office method in conducting Project Management.	4,711E-02	,837	,183
Back-office Project Management was conducted according to a highly mature Project Management standard.	,530	,635	8,245E-02
All back-office team members had previous experience in working with a front-office team.	1,063E-02	-,158	,826
The previous working experience between individual front-office and back-office team members with each other was high.	,120	,165	,707
All front-office team members had previous experience in working with a back-office team.	3,779E-02	,508	,677
All team members were sufficiently trained to be aware of potential collaboration pitfalls.	,476	,224	,483

Extraction Method: Principal Component Analysis.
Rotation Method: Varimax with Kaiser Normalization.

a. Rotation converged in 5 iterations.

Residual controllable collaboration risks

Rotated Component Matrix[a]

	Component	
	1	2
Front-office team-members complied consistently to the used Project Management method.	,874	-,100
Back-office team-members complied consistently to the used Project Management method.	,829	,224
Synchronous communication tools were used whenever necessary in the project.	5,287E-02	,985

Extraction Method: Principal Component Analysis.
Rotation Method: Varimax with Kaiser Normalization.

a. Rotation converged in 3 iterations.

Residual incontrollable collaboration risks

Rotated Component Matrix[a]

	Component		
	1	2	3
There were unforeseen efforts necessary to obtain common understanding on contextual business knowledge.	,812	,182	,102
There were unforeseen efforts necessary to obtain common understanding of front-office business processes.	,796	,178	,196
There were unforeseen efforts necessary to clarify the client organization application objective to the back-office.	,788	8,394E-02	-,125
There were unforeseen project efforts due to hierarchical approachability difficulties of other side team members.	,694	,385	,129
There were unforeseen project efforts due to misaligned ideas about required preciseness of project deliverables.	,627	,376	,315
There were unforeseen efforts necessary to clarify team member role responsibilities.	,613	,150	,281
Back-office team members acted according to or more pro-actively in the project than was expected.	,521	,469	7,082E-02
The willingness to accommodate other side team member objectives was according to or higher than expected.	,322	,728	,232
team member objectives in the project was according to or above expectations	,142	,724	-5,89E-02
The teamness; sense of being part of one team between front-office and back-office was according to or higher than expected.	,401	,722	,235
The speed with which issues could be dealt with between front-office and back-office was according to or faster than expected.	8,852E-02	,715	,369
Collaboration skills in the other team were according to or above expectations in the project.	,134	,709	1,609E-03
There were unforeseen efforts necessary to obtain common understanding in the project due to different communication customs between front-office and back-office.	,336	,366	,763
There were unforeseen (native) language understanding difficulties between front-office and back-office.	,226	,190	,739
There were unforeseen efforts necessary to obtain common understanding on programming work.	,537	,241	,602
There was more reluctance in sharing strategic application information than was expected.	,241	,228	-,448

Extraction Method: Principal Component Analysis.
Rotation Method: Varimax with Kaiser Normalization.

a. Rotation converged in 5 iterations.

11.2.2 H2 factor solutions

<u>Controllable application risk factors</u>

Rotated Component Matrix[a]

	Component				
	1	2	3	4	5
Project Monitoring and Control process standards were followed consistently in the project.	,873	,133	1,303E-02	-4,81E-02	,318
Configuration Management process standards were followed consistently in the project.	,843	,241	-5,69E-02	,158	4,374E-02
Requirements Management process standards were followed consistently in the project.	,843	,291	-3,27E-02	7,033E-02	,109
Requirements Development process standards were followed consistently in the project.	,828	,148	-9,71E-02	,186	,165
Measurement and Analysis process standards were followed consistently in the project.	,787	,241	,116	,254	-2,05E-02
Software Project Planning process standards were followed consistently in the project.	,728	9,013E-03	,434	6,942E-02	,185
The Service Level Agreement was highly detailed.	,690	,187	,124	,273	-,267
Causal analysis and Resolution process standards were followed consistently in the project.	,613	,356	,354	,219	,182
Front-office and back-office process maturity for Project Planning was similar.	,593	,294	,417	,177	,224
Front-office and back-office process maturity for Project Monitoring and Control was similar.	,587	,337	,300	-9,72E-03	,259
All project members complied consistently to the used development method.	,452	,153	,220	,241	-2,84E-02
Front-office and back-office process maturity for Requirements Management was similar.	,239	,834	1,042E-02	,160	,249
Front-office and back-office process maturity for Requirements Development was similar.	,277	,804	-9,87E-02	6,313E-02	,209
Front-office and back-office process maturity for Configuration Management was similar.	,333	,778	,109	3,077E-02	-1,50E-02
Front-office and back-office maturity with regard to development method standards was similar.	,299	,666	,244	,357	-2,44E-02
The structure of the existing application was well structured. If applicable	8,098E-02	1,756E-03	,842	9,134E-02	,174
The application code of the existing application was self descriptive (with design rationale). If applicable	5,695E-02	-,197	,803	,128	,216
The programmers familiarity in the back-office with the existing application at the front-office was large. If applicable	-9,42E-02	,318	,783	5,989E-02	-4,10E-02
Application architecture, was based on standards.	,406	3,200E-02	,610	,463	-7,96E-02
Front-office and back-office process maturity for Measurement and Analysis was similar.	,410	,471	,510	8,551E-02	,173
Front-office and back-office process maturity for Causal Analysis and Resolution was similar.	,455	,413	,483	,349	,161
Application engineering tool support was highly advanced.	,325	3,649E-02	,101	,758	,246
Testing tool support was highly advanced.	-1,18E-02	,331	,121	,715	,234
Data architecture, was based on standards.	,421	8,582E-02	,274	,635	-9,33E-02
Application engineering tool support was used sufficiently.	,132	7,492E-03	,188	3,158E-02	,772
Application engineering tool support was used consistently.	,224	,394	,229	,185	,701
Testing tool support was used sufficiently.	5,230E-02	,408	9,619E-03	,330	,572

Extraction Method: Principal Component Analysis.
Rotation Method: Varimax with Kaiser Normalization.

a. Rotation converged in 7 iterations.

Controllable collaboration risk factors

Rotated Component Matrix[a]

	Component				
	1	2	3	4	5
Team members had access to accurate English application documentation.	,833	-3,25E-02	5,130E-02	,227	,108
Back-office team members had access to accurate English contextual business knowledge to be captured in the application.	,734	,127	-3,02E-02	,334	,392
Back-office team-members complied consistently to the used Project Management method.	,663	,311	-,232	3,865E-02	-,444
All team members were sufficiently trained to be aware of potential collaboration pitfalls.	,620	,127	,497	-4,49E-02	8,752E-02
Front-office Project Management was conducted according to a highly mature Project Management standard.	-5,65E-03	,854	,117	,112	,267
There were no initial differences between the front-office and back-office method in conducting Project Management.	,128	,758	,349	,106	-,114
Front-office team-members complied consistently to the used Project Management method.	,159	,732	-,290	-6,94E-02	-,237
Back-office Project Management was conducted according to a highly mature Project Management standard.	,478	,499	,211	,131	,332
All back-office team members had previous experience in working with a front-office team.	,111	-,237	,772	2,461E-02	-,219
All front-office team members had previous experience in working with a back-office team.	-2,73E-02	,349	,772	8,243E-02	,135
The previous working experience between individual front-office and back-office team members with each other was high.	4,863E-03	,107	,665	7,336E-02	,259
High-level Synchronous communication tools between front-office and back-office were available.	,147	2,256E-02	7,737E-02	,880	6,612E-02
Synchronous communication tools were used whenever necessary in the project.	,168	,106	5,046E-02	,878	-1,14E-02
The project-team worked with liaisons.	,217	1,021E-02	8,067E-02	3,227E-02	,887

Extraction Method: Principal Component Analysis.
Rotation Method: Varimax with Kaiser Normalization.

a. Rotation converged in 6 iterations.

11.3 Roughly translated RAS – Risk variable mapping

Complexity	Impact * occurrence	Capability	Impact * occurrence
Architecture	**0.71 (Average)**	**Knowledge Management**	**1.37 (Average)**
Data architecture simplicity	0.32	Access accurate application documentation	1.41
Technical architecture simplicity	0.79	Access contextual business knowledge	1.31
Data architecture standardization	0.75	**Methods**	**1.11 (Average)**
Application architecture standardization	0.84	Development method maturity similarity	0.98
Technical architecture standardization	0.85	Project Planning process similarity	1.86
Business	**0.90 (Average)**	Causal analysis and resolution similarity	1.55
Business logic complexity	0	Configuration management similarity	0.95
Data transfer security stringency	0.61	Project Monitoring and Control similarity	1.02
Common understanding front-office business processes	1.79	Measurement and Analysis similarity	1.13
Common understanding contextual business knowledge	1.21	Front-office project management maturity	0.7
Application	**1.14 (Average)**	Sophistication of application engineering tool support	1.28
Module diversity	0.82	Consistent use application engineering tool support	1.01
Common understanding programming work	1.35	Sufficient use application engineering tool support	0.59
Cohesion in existing application	1.11	**Requirements**	**0.94 (Average)**
Descriptiveness existing application	1.26	Similar process requirements development maturity	0.83
Interface	**0.71 (Average)**	Similar process requirements management maturity	1.2
Internal interface complexity	0.82	Consistent use requirements development process	0.78
External interface complexity	0.6	Consistent use requirements management process	0.93
Infrastructure	**0.41(Average)**	**Test**	**0.95 (Average)**
Hardware platform complexity	0.22	Sophistication of testing tool support	0.9
Software platform complexity	0.6	Consistent use testing tool support	1.05
Interaction	**0.85 (Average)**	Sufficient use testing tool support	0.89
Initial functional detailedness requirements	0.72		
Initial technical detailedness requirements	0.48		
Speed of dealing with issues	1.5		
High level synchronous communication tools	0.71		
Support	**1**	**People**	**1**

11.4 Process data model RAS