Improving offshore communication by choosing the right coordination strategy

Matthias Fabriek, Mischa van den Brand, Sjaak Brinkkemper, Frank Harmsen, and Remko Helms

Department of Information and Computing Sciences
Utrecht University
Technical Report UU-CS-2007-021
www.cs.uu.nl
ISSN: 0924-3275

ABSTRACT

This thesis researches communication and coordination in offshore custom software development (CSD) projects. Nowadays, a lot of organizations are offshoring their information technology to countries with low wages. Main reasons for offshoring are cost reduction, entering a large labor pool, increasing international opportunities, increasing the quality of service, and exploiting around the clock development. Research shows that almost all of these goals of offshoring are not or partly met. This thesis analyzes why certain offshore projects are unsuccessful and how they should be set up in order to meet the predetermined goals. The main question is: *"How should a distributed offshore software development project be coordinated, in order to increase the chance on a successful project?"*

Literature research in this thesis concludes that success of projects can be defined as a realized scope and quality that is above the expected scope and quality, and a project duration and costs that are below the total expected time and costs. All aspects of success are influenced by distances that are faced in an offshore project: geographical, time zone, cultural, organizational, and stakeholder distances influence negatively the communication and knowledge exchange in a project. Coordination by means of standards, planning, formal mutual adjustment, informal mutual adjustment, and team selection, can be introduced in a project to compensate the negative effects and increase the final performance.

The research that answered the main question consisted of analyzing 19 completed CSD projects that were all partly executed in The Netherlands. The other part of the projects was done by team members who were located in India, Romania, or Malaysia. Two third of these projects were executed at two business units of Capgemini. One third was executed by other organizations. The projects varied in successfulness, size, duration, type of customer, programming language, costs, type of contract, and organizational complexity.

Half of the data about the projects was collected by interviewing project managers face to face in The Netherlands, and interviewing most of the offshore project managers in India via an advanced videoconference tool. The other half was collected by questionnaires that were sent to the team members of nine cases. Five of those projects were used to perform a social network analysis which revealed the communication between the project members during the projects.

The results show that offshore custom software development projects mostly have more costs and take more time than was expected beforehand. Introducing coordination measures costs more time and money, but do have a positive effect on the results of the projects. Successful offshore projects consisted of a team that had already worked together in previous projects. Successful projects took less than six months and had low organizational complexity: a maximum of three organizations were involved. Successful projects had a budget that was flexible in the first part of the development and fixed in the last part.

According to the results all categories of coordination measures should be applied in an offshore project, but some are more important than others. Standards were the least important. A lack of good planning was a major cause of project failure. It resulted in unclear requirements, a lot of changes, and unclear or too tough budgeting. Informal mutual adjustment and formal mutual adjustment were mentioned as major causes of project success. Selection of experienced team members is critical, too.

The thesis concludes with an addition to the general software development method. In order to have successful offshore custom software development projects, a thorough preparation should be added. This preparation helps to analyze the projects situation and choose the best road to reach success. It starts with the evaluation of the desired end performance by all stakeholders. After analyzing the distances between the offshore and onshore stakeholders, for each activity should be determined which part can be offshored. In the end, all coordination measures have to be determined up front. Further research can fine tune the results, by giving statistical information about the cause of project successes or failures in quantitative research, and by making project success more operational.

CONTENTS

1 INTRODUCTION

This chapter introduces the subject of research. It will explain what we investigated during our thesis project. At the end of the chapter, the following points will be clearer to you:

1. The subject of this thesis
2. The added value of research in this area to society and science
3. The scope of this thesis
4. The environment in which we executed our research
5. The research questions that we answered

1.1 Subject of research

In our office, we type on Microsoft keyboards. If we turn them around, we see the sentence: "Made in China". When we look at the back of our IBM mouse, they say: "Made in China". Analyzing our Cisco telephones on our desks, we see: "Made in Malaysia". This is the same for our microwaves, our computers and so on. A lot of equipment around us is made by western companies in Asian countries. Places, where labor costs are much lower than in western regions.
After decades of offshore production experience, the manufacturing and distribution of physical goods goes fluently: efficiency is the consequence of years of experience and years of research about cooperating on a long distance.

1.1.1 The trend of offshoring

A lot of people are used to the idea that products are being manufactured overseas. Less people know that the computer program which we are currently working with is also the product of lots of people who worked in Asian countries. Software that is the result of intense cooperation between project managers, architects, analysts, developers, testers and other people who were located all around the world.
Due to the internet, it is now possible to share information with anyone around the world. This opens the way to locate digital services around the world. In the last few years, a lot of companies used this opportunity to offshore their information technology (IT) services to other countries. Currently, offshoring IT services as a company is becoming a necessary step in order to survive in a competing market. This development is described in the bestseller "The world is flat", of Thomas L. Friedman (2005): he describes the current trend of offshoring knowledge intensive services as a third phase in globalization, which started around the year 2000.

1.1.2 Reasons to offshore

Why do companies decide to locate their IT services in other parts of the world? Five main reasons are often mentioned in scientific literature (see Table 1.1 for an overview):

- **Cut the costs by paying lower wages:** The first reason to offshore IT services was to lower the IT costs. Wages in Asian countries such as India are less than 50% of their equivalent labor in European countries or the USA (Aspray, Mayadas & Vardi, 2006). During our research, we attended several business conferences (such as 'Transforming business through offshore outsourcing', organized by Rotterdam School of Management on November 28th, 2006) about offshoring, and cutting the costs was the most mentioned reason to offshore.

- **Entering a large labor pool:** Another important reason that is mentioned by a lot of researchers is the access to a large talented labor pool. Since there is a shortage of skilled IT labor in European countries, or countries in the USA, companies start to look for labor pools in other countries.
- **Increased international opportunities by entering new markets:** Since the economies of some non-western countries are evolving, new offshore markets are added to the already consisting western markets. Companies want to enter these new markets by acquiring non-western companies or establishing new business units in those countries. These increased international opportunities are the third important reason that is mentioned in literature.
- **Increased quality of service:** Working together with people in another country helps to increase the quality of the services or products that are delivered. One important reason for this quality increase is that different cultures have different views, which prevents having a bias in a team (Chan & Chung, 2004).
- **Reduce time-to-market by establishing around the clock development:** One extra reason that is also mentioned a lot by researchers is around the clock development. Reducing time-to-market by applying follow-the-sun-development seems interesting, but applying it asks so much discipline and coordination, that it is only realized by a few companies (Conchuir, Holmstrom, Agerfalk & Fitzgerald, 2006).

	Cut costs	Enter a large labor pool	International opportunities	Increased quality	Reduced time-to-market
Ravichandran & Ahmed (1993)	x	x	x	x	
Carmel (1999)	x	x	x	x	x
Herbsleb & Grinter (1999)		x	x		x
Carmel & Agarwal (2001)	x	x			
Cramton (2001)		x	x		
Krishna, Sahay & Walsham (2004)	x	x		x	
Gartner (2005)	x	x		x	
Holmstrom, Conchuir, Agerfalk & Fitzgerald (2006)	x	x			x
Aspray, Mayadas, Vardi (2006)	x				
Damian & Moitra (2006)	x	x	x		x
Farrell (2006)		x			

Table 1.1. An overview of reasons to offshore.

1.1.3 Realization of these promises

The promises that were mentioned in the previous section are the driving factors of the trend to offshore IT services. Companies want to lower their costs, have enough skilled employees, enter new markets, increase their quality and reduce time to market. The main reason for this is that all these aspects increase the chance on getting a better competitive position and growing profits, which in their turn rejoice the stakeholders of a company. This reasoning forces executive boards of companies to offshore IT services.
But what happens if one or all of the promises are not realized? According to Cramton (2001), developing software in an offshore setting leads to poor decision quality, poor productivity and poor relationships. Companies feel many threats at each of the mentioned advantages:

Overhead causes higher costs
While reducing the costs is the first reason to offshore, companies recognize that extra costs are involved. The amount of overhead costs and the increases in salary levels in targeted countries often lead to costs that are higher than was the case in onshore development (Kuni & Bhushan, 2006).

Labor forces are as quickly lost as acquired
A lot of companies entered the new labor pools in Asian countries. While looking for the best location, they were searching for language skills, IT skills, a good legal system, good technical facilities, and other aspects (Farrell, 2006). This comparison led most of the companies to India. Currently, the economy of this country is almost overheated because of the growing amount of offshored work (BBC News, 19 December 2006; Reuters India, 2006). Organizations want to hire and fire extra personnel in a fast way. The downside of this development is that people also leave the company quickly when they are offered a better paid job elsewhere.

New entrants in the western market increase competition
Where western IT companies saw Asian markets as a new market to enter, Asian IT companies are starting to see the Western countries as a new market, too. These 'pure players' are trying to increase their market share in western countries, by offering low prices for their services. This development increased competition enormously (Narayan, 2006).

Quality not as high as intended
Cooperation among different countries did not always lead to quality, too (Conchuir, Holmstrom, Agerfalk & Fitzgerald, 2006). Different organizations, cultures, locations, time-zones and stakeholders often lead to lower quality, as will be explained in the next chapter.

Development costs more time
As was already mentioned, follow-the-sun-development is almost impossible to establish. Even more, a lot of projects that are being executed offshore without follow-the-sun principles are taking more time than expected (Herbsleb, Mockus, Finholt & Grinter, 2001). This is perfectly formulated by Seshagiri (2006), CEO of Advanced Information Services: "The follow-the-sun model is essentially a quick-and-dirty strategy that converts a schedule problem into a quality disaster." Even more, Herbsleb et al. (2001) prove that cross-site work involves more people, too.

1.1.4 Conclusion

We can conclude that offshoring not only offers opportunities to gain extra profit, but also offers extra threats that influence the profit negatively. Despite these threats, offshoring is becoming a norm in the software industry (Damian & Moitra, 2006). How can the threats be reduced, in order to increase the chance on high profits? Rephrasing this question, provides an overview of our topic of research:

Main research question:
"How should a distributed offshore software development project be coordinated, in order to increase the chance on a successful project?"

As becomes clear in the research question, we focus in our project on distributed offshore software development. This is because this activity is most attractive to organizations to perform offshore (Gartner, 2005). Before explaining the different parts of the research question in detail in section 1.3, we will explain the social and scientific relevance of performing a research in this area.

1.2 Social and scientific relevance

Organizations are increasingly looking for ways to reduce the threats that were mentioned in the previous section, because they recognize that not all offshore projects end successfully. They are looking for the best ways to offshore their IT work. In our research we investigate what those best ways are. The answer will help organizations to set up an effective and efficient collaboration between a team in a western country and a team in another country.

1.2.1 Social relevance

Looking at social relevance, research in the area will help:

- **Customers:** To be able to profit from good services or software with low costs and in a reasonable time.
- **Western IT service providers:** To be able to set up efficient cooperation with their offshore colleagues and deliver better results to their customers.
- **Offshore IT service providers:** They are often part of a customer or a western IT service provider. Improvement will help to get more work offshored, when successes increase.

All three stakeholders are benefited with improved collaboration. This thesis adds value by analyzing how success or failure was caused in a project.

1.2.2 Scientific relevance

In the special IEEE Software special issue on Global Software Development, Damian & Moitra (2006) say, that "although it's true that a body of knowledge on global software development has been crafted over time, the art and science of organizing and managing globally distributed software development is still evolving." They also state, that "with the intensification of globalization and the resulting growth in globally dispersed software development, the need for understanding the engineering and management approaches necessary for successful global software development has only become more pronounced."
These citations underline the importance of researching the how-question of offshoring. Previous research has mostly focused on defining critical success factors that influence offshore success. This research will go deeper, by analyzing how the factors influence success or failure. More information about previous research can be found in the next chapter. From the literature research that we describe in that chapter, we confirmed that there is a lack of in-depth research about the influence of offshore locations on collaboration in teams. Finding an answer to the main question would therefore add value to the scientific community. This justifies our research.
As will become clear in the methodological chapter, we will use Social Network Analysis and Method Engineering to analyze offshore projects. These techniques are not yet applied to distributed software development. This thesis adds value to those techniques, by applying them to an offshore situation and using them to find bottlenecks that can be improved.

1.3 Scope

Before going deeper into the area of global software development, we will elaborate on the different parts of the research question. Which subjects are in scope of our research and which are out of scope?

Projects
In our research we focus on completed projects with a clear beginning and end. This helps us to define if the projects were successful once they have been completed. We do not look at offshore situations where the team is continuously working on for instance bug fixing. Sometimes, projects are called 'releases'. A release is when new functionalities are added to already existing software. When a release has a clear start and end, it is also in scope.

Software development
During our thesis, we focus on projects concerning software development. As we already mentioned, this activity is most attractive to organizations to perform offshore (Gartner, 2005). Software development uses methods that are similar to each other, while other types of projects often follow other procedures. In order to be able to compare different project outcomes in our research, we decided to focus on software development: comparing IT implementation with software development is much harder, because these different types use different project management approaches. Software development starts with idea generation and ends with realization and delivery of the final product. We focus on custom software development (CSD), and not on the customization of prepared IT packages. Capgemini has a lot of experience in custom application development. That explains their request to optimize offshore software development.
We will not focus on one major software development method, but will generate a general method with activities that are used by almost each CSD method.

Distributed development
The development projects that are in scope develop software on two locations (front-office and back-office, see Figure 1.1). Organizations in this area are called 'non pure players'. We are not looking at situations where a complete project is done offshore ('pure players'). The main reason for this decision is that Capgemini works with a front and back-office.
An example of a project that is in scope is a bank that outsources its software development activities to Capgemini, who decides to perform project activities partly in the Netherlands and India. An example that is out of scope is a bank that decides to outsource its software development activities completely to a service provider in India.

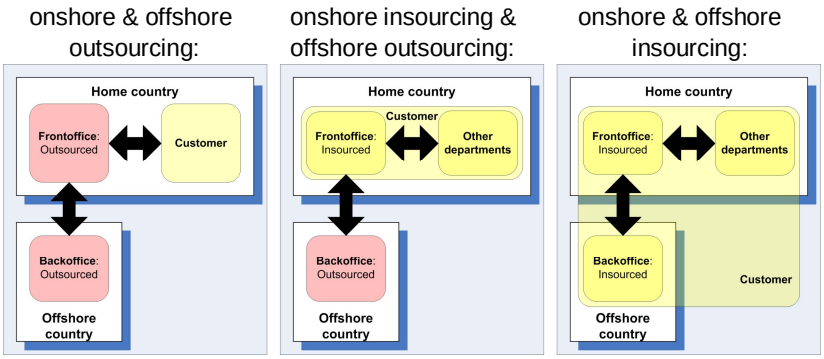


Figure 1.1. Overview of different cooperation types that are in scope

Offshore insourcing and outsourcing
Both of the locations where the software development takes place can be insourced or outsourced. Figure 1.1 illustrates, that insourcing or outsourcing can be done in three ways: an organization can decide to hire its own personnel in the offshore country (onshore and offshore insourcing), an organization can ask an offshore company for an offshore development team that works with their own developers onshore (onshore insourcing and offshore outsourcing), or an organization can ask a company in their own country to deliver a team that is located onshore and offshore (onshore and offshore outsourcing). This can be done with a full project, but also with one part of the project. All situations are in scope, as long as two different locations cooperate. The main reason for this choice is that Capgemini is working in this way: one part of the project teams is located in The Netherlands, and a second part is located in India. Throughout this thesis, the word 'customer' will be used for the stakeholder who initiates the project and who accepts the end result. This is also done by Procaccino, Verner, Overmyer, and Darter (2002), Agarwal and Rathod (2006), and others.

Offshore is the case when traveling between both locations costs so much time and money, that having regular face to face meetings becomes impossible. In the next chapter, this term will be defined more clearly. Our focus is on projects that are partly executed offshore.
This thesis is focusing on situations where the decision to offshore has already been made. The question about whether the project should be offshored will not be answered: we focus on the question about how to offshore. The main reason for this is that Capgemini already researched the 'if'-question, and wishes to get extra information about the 'how'-question.

Multiple outsourcing / Single outsourcing
Most of the companies today have multi-sourcing strategies: for different projects, they select different suppliers. It is unclear if a lot of organizations select multiple vendors for developing one specific application. Most probably, this will not often be done: it costs too much overhead investment and energy, when synchronizing multiple front-offices and/or multiple back-offices. Because of this reason, and because a multiple outsourcing situation is much more complex to analyze and research, we will focus on situations that involve a back (offshore) and front (onshore) office.

1.4 Research environment: Capgemini

This thesis was performed at Capgemini, in cooperation with Utrecht University. This section aims at giving an overview of Capgemini and its stakes in this research.

1.4.1 History of Capgemini

Capgemini is an information technology service provider, delivering services in the area of technology, consulting and outsourcing. This company was founded in 1967 by Serge Kampf in Grenoble, France. Since its founding, Capgemini acquired Sesa, Hoskyns, Volmac, Programator, United Research, Mac Group, Gruber Titze and Partners, Bossard, Ernst & Young Consulting, Transiciel, and other companies. Most of these acquisitions were done in order to increase market presence in certain countries or certain specializations. The last acquisition that is done by Capgemini is the USA-based organization Kanbay. This acquisition adds 15.000 people to Capgemini, mostly located in the USA and India (Capgemini, 2006).
Nowadays, Capgemini employs 60.000 people, who are working in more than 30 countries. Capgemini is currently led by Paul Hermelin, from the headquarters in Paris. The main services that Capgemini offers are:

- **Technology services:** Services that focus on technological improvements, such as application development, IT transformation, etcetera.
- **Consulting services:** These services focus on delivering general (non-IT) advice on strategic or tactical level.
- **Outsourcing services:** These departments are able to take care for a complete IT department of an organization. They also take care for the maintenance of applications that are developed in technology services.
- **Local professional services:** These services focus on a local market, and are offered under the brand name 'Sogeti' in stead of 'Capgemini'.

We performed our thesis at technology services, which are specialized in application development. More specifically, we performed our thesis in a department called "Technology Advisory Services", who is giving advice in the area of IT sourcing. They wish to increase their knowledge about offshore sourcing.

1.4.2 Offshore development

Application development is one of the core businesses of Capgemini. The last years, Capgemini started to offshore part of the IT services to other countries. Especially its presence in India has increased a lot: currently, 6000 employees are working in offices in Mumbai, Bangalore and Kolkata. The acquisition of Kanbay adds another 5000 employees to this number.
Capgemini registered a special trademark for its approach in offering IT services from all over the world: Rightshore™. Righshore is a concept of "offering the right resources, at the right location, at the right time". This means "leveraging networks of industrialized 'near-' and 'off'-shore centers to provide increased value at reduced total cost of ownership" (Capgemini, 2005).
Rightshore is becoming one of the core businesses of Capgemini, and should therefore be optimized. Our research is focusing on offshore development in cooperation with an onshore team. CSD projects that are done Rightshore at Capgemini are therefore in scope, as long as they are partly executed in an offshore country.

1.4.3 Stakeholders

In short, we will describe the stakeholders that we took into account, during the execution of our research:

- **Customers of Capgemini:** The main goal of improving offshore collaboration is that better services can be delivered to the customers of Capgemini. These services can be successful developed applications, but also advice to customers about how to set up their own offshore arrangements.
- **Rightshore managers and developers at Capgemini Holland and India:** These people are responsible for offering application development to the customers, and wish to improve the chance on success of offshore development.
- **Consultants at Capgemini Technology Advisory Services:** These people intend to give advice to customers about offshore (outsourcing) arrangements. Increasing the knowledge about how to set up such a project will add value to this practice.
- **Utrecht University:** Since this research is done as a thesis, Utrecht University is also involved. The people of the Master's program Business Informatics wish to see added value to their own research, in order to improve the quality of Business Informatics.

1.5 Values during our research

Before starting our research, we defined three important values for our research. These values indicate the climate in which our thesis should be done.

Independence
As is already stated in the previous section, our intent is to add value to both Capgemini and its customers. We will not try to look from the Capgemini perspective alone, but also from the customer's point of view. This stresses the need for an independent view, since a customer wants to have a successful project, with or without the help of Capgemini: a customer wants to get independent advice.

Trustworthiness
During our thesis, we analyzed both successful and unsuccessful projects. Information about both types of projects is sensitive to competitors. This requests a climate of trust. All information in this paper is published anonymously, in order to fulfill that need.

Besides organizations who need trust in order to provide delicate information about certain projects, people who are interviewed need it, too. Their opinions could be harmful to their positions in their organization, which also underlines the need for trust. All people are mentioned anonymously, too.

Applicable
Besides being independent and trustworthy, the results must be applicable to all stakeholders as mentioned in the previous section. Our intention was to have results that really matter, and not to come up with certain semi-applicable factors that disappear in certain drawers.

1.6 Research questions

As was already stated in the section Area of Research, our research question is:

Main research question:
"How should a distributed offshore software development project be coordinated, in order to increase the chance on a successful project?"

The different aspects of this question are already mentioned in the scope section. In popular language, we focus on the "holy grail of how offshoring should be done".

1.6.1 Sub questions

The main question can be split up into two questions. Question A investigates the relationship between offshore collaboration and performance. Question B investigates how these elements can be positively influenced: some coordination measures should be taken, in order to increase the chance on success. The purpose of splitting the main question into two parts is that the outcome of the research must be applicable to the real world. A lot of research that has been done in the area of offshoring focuses on answering question A, while companies are interested in question B.

- **Research question A:** Which threats or opportunities cause the success or failure of an offshore development project?
- **Research question B:** Which coordination measures should be introduced in a project, in order to mitigate these threats or exploit the opportunities?

1.6.2 Sub-sub questions A

Question A raises certain extra questions about offshoring, success and failure:

- **Research question A.1:** What is the difference between an onshore and an offshore CSD project?
- **Research question A.2:** When are offshore CSD projects successful?
- **Research question A.3:** Why is the success of offshore CSD projects different, compared to onshore CSD projects?

1.6.3 Sub-sub questions B

Question B can also be split up in multiple questions:

- **Research question B.1:** Which coordination measures exist in a CSD project?
- **Research question B.2:** Which coordination measures should be introduced, in order to positively influence the success of offshore CSD projects?

1.7 Approach

All questions will be answered by using two methods. In the next chapter, we will describe one of them: literature research. Science has answered lots of questions about how offshoring should be done. This knowledge can be used to answer parts of the questions. More information mostly leads to more questions. This is why we also apply a second method: empirical research by analyzing CSD successes or failures. The methodological chapter provides more information about how this research is set up. The findings from literature and the results from our research will help to draw conclusions and discuss each other's results.

1.8 Justification

As is explained in section 1.1, current promises of offshore CSD are often not met. This research shows some ways to change this situation. Section 1.2 stated already, that both customers and IT service providers will benefit of a positive change in the realization of the promises.
From a scientific point of view, this research helps in improving applications of the Social Network Analysis and the Method Engineering approaches, as was already explained in section 1.2.2. That section also explained that in depth research is needed about how collaboration in an offshore CSD project is influenced and how it can be improved. We believe that this research fulfills a part of the need for research as was mentioned in that section. The advantages of finding answers on the above mentioned questions made us decide to start with the research from literature.

2 LITERATURE RESEARCH

This chapter describes our journey trough scientific research. This chapter analyzes literature for initial answers on the following questions.

1. What is offshore CSD? *(research question A.1, section 2.1 & 2.2)*
2. When are offshore CSD projects successful? *(research question A.2, section 2.3)*
3. Why is the success of offshore CSD projects different? *(research question A.3, section 2.4)*
4. How can offshore CSD projects be managed? *(research question B.1, section 2.5)*
5. How can you manage the success of offshore CSD? *(research question B.2, section 2.6)*

Finding answers from literature on these questions will serve two goals: first, they help to answer the main questions as mentioned in the previous chapter. The links to those questions are depicted. Second, the answers help to setup the research as will be described in the next chapter.
In the end of this chapter, a research framework is presented that will be used in the rest of the thesis. The framework shows that distance in an offshore CSD project influences communication and coordination, who in their turn influence the success of a project. In order to facilitate the mapping of the knowledge that is presented throughout this chapter, a stripped version of the framework is already depicted in Figure 2.1, and will be extended throughout this chapter.

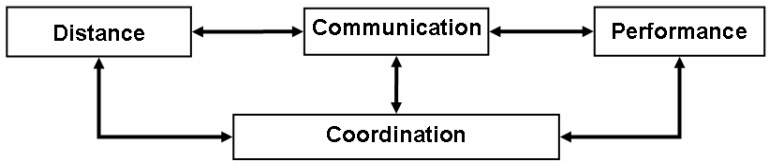


Figure 2.1. A stripped version of the research framework

2.1 Distributed software development

In the next section we will provide an overview of offshoring literature. The ancestor of 'offshore software development' is 'distributed software development'. In order to understand offshore development, it is important to also describe its ancestor.
Software development can be done at one or multiple locations. Using multiple locations is called distributed software development. According to Layman, Williams, Damian, and Bures (2006), distributed software development is the case when people from a development team are not physically collocated and cannot see or speak in person on a regular basis.
The expression 'not physically collocated' is further explained by Carmel (1999): when people of one team are split at more than 25 meter, the chance on regular communication is less than 10 percent. Such a situation can be called 'distributed', when applying the definition of Layman et al.
Other terms that mean exactly the same, are 'multi-site software development' (Chan & Chung, 2004) and 'cross-site software development' (Herbsleb et al., 2001). These terms also mean that software is developed at multiple locations.

Definition:

Distributed software development *is the development of software in a team where people are not physically collocated and cannot see or speak face-to-face on a regular basis.*

Unfortunately, in some research papers the term 'distributed software development' is also used for offshore software development (Taxén, 2006). The terms are not the same, since software can be developed at one offshore place (not distributed), and developing software in two different buildings in the same city (distributed) are not offshore.

2.1.1 Why would someone develop on multiple sites?

As anyone can imagine, developing the same application in two teams on two locations will be much harder than developing in one team on one location. Herbseb et al. (2001) proved that cross-site development costs more time and involves more people than would be needed on one location. Organizations choose to develop on multiple sites (not specifically offshore), because of the following reasons.

- **Knowledge is located at multiple sites:** Gertler (2003) states that tacit knowledge does not travel easily. This is one of the reasons why organizations have to develop software at different sites: if the required knowledge is not available at the department of the development team of the company, they have to locate part of the team in a location where that knowledge is available. Since much of the development knowledge is available at the IT department, another part of the team will still be located on the other site. Besides people, computer systems or documentation can also contain knowledge that is located at another location than an IT department.
- **People are located at multiple sites:** Chan & Chung (2004) state that another main reason to develop multi-site is that elsewhere a large pool of resources is available. The previous reason focused on knowledge as resources, while this reason is focusing on people.
- **Increase quality by introducing different views:** The third important reason to develop on multiple sites is that the quality increases when groups with different perspectives work on the same application (Chan & Chung, 2004).

2.1.2 How to develop in a distributed situation?

How can teams on multiple locations work efficiently together on the development of one application? Chan & Chung (2004) state that a coordination framework is needed that integrates the software development process and project management. Such a framework would help to get successful distributed development. The technical development process needs to be aligned with the overall control and coordination.
Sarker & Sahay (2003) also looked at how distributed teams should work together. They introduce the term ‘virtual team’ development. The teams on multiple sites need to feel part of one team, which is established on macro level by product related and social collaboration and on micro level by individual communicative actions.
Specific actions to improve multi-site development are proposed by Gertler (2003). These actions focus on the knowledge that is spread among the locations. This knowledge needs to be combined, in order to develop the desired application. Some examples are:

1. Introduce boundary spanners at each location
2. Introduce communities of practice
3. Establishing “learning regions”

All described solutions are focusing on distributed software development. Developing on two locations, where one is situated offshore, is more complicated, as will be described in the next section.

2.2 Offshore software development

What is offshore software development? In the previous section we defined the term ‘distributed development’, where a team is located at different locations. Offshore software development is a special version of distributed development, as is explained in this section.

2.2.1 Distributed teams versus offshored teams

Sakthivel (2005) gives a definition for a distributed team which he calls 'virtual workgroups': geographically dispersed people working independently with shared purpose across space, time and organizational boundaries and using technology to communicate and collaborate. Besides a different location, he mentions time and organizational boundaries. What is mentioned by these extra differences? The concept of virtual workgroups belongs to distributed software development as mentioned in the previous section.
This is also the case with Cramton (2001), who gives a definition for 'geographic dispersed teams': Groups of people with a common purpose, who carry out interdependent tasks across locations and time, using technology to communicate much more than they use face-to-face meetings. Besides different locations, which is an aspect of distributed development, Cramton also mentions time as characteristic of the team.
The definitions of virtual workgroups and geographic dispersed teams are in fact not an instance of all distributed teams, but of teams who are distributed over long distances. So far west or east that different time zones are involved.
These terms are also mentioned by Damain & Moitra (2006). They state that certain teams face time and distance while developing software. This is where we enter the area of offshore, nearshore, best shore, anyshore, rightshore, farshore, dualshore, offsource, nearsource, multishore (Carmel & Tjia, 2006).

2.2.2 Definition of offshoring

The most important aspect of all terms with which the previous section ended, is that software is developed on different places in the world. This is why Carmel (1999) calls it 'global software development'. According to Layman, Williams, Damian & Bures (2006), global software development means that the software development team is dispersed across national boundaries. Sakthivel (2005) calls it 'offshore system development': system development in another country. The main characteristic is that national boundaries are crossed.
Maznevski & Chudoba (2000) mention 'global virtual teams'. These teams have the following characteristics:

1. The people are identified as members of the team
2. All members they are responsible for the end product
3. The members communicate with each other
4. The members are located in different countries

The first three points are aspects that were also part of normal distributed teams. The fourth item is a differentiating aspect: again national boundaries are crossed.
Jarvenpaa & Leidner (1999) are not mentioning different countries. They define globally distributed teams as being spread across time, space and culture. Time and space differences were already mentioned in the previous section as being the main reason to introduce the new concept of offshoring and other related terms.
When part of the software is developed in the eastern part of the USA, and another part is developed in the western part of the USA, this is still in the same country, but in different time zones. This can be called global development. When software is partly developed in The Netherlands and partly in South Africa, activities take place in the same time zone, but in different countries. This is also an example of global software development.

In summary, we define offshore software development as:

Definition:

Offshore software development *is the development of software in a team where people are physically located in different countries or different time zones.*

The last part of the definition is the main reason why offshore software development is different from distributed software development. Meeting face-to-face costs a lot of time and money, since the team members are located far away from each other. This places a strong emphasis on finding the right ways to communicate (Sakthivel, 2005; Layman et al., 2006).
Besides being located at different places, the other characteristics of global virtual teams that were mentioned by Maznevski & Chudoba (2000) also play their role. This is confirmed by Bozarth, Handfield & Das (1998), who introduce the term 'international software development'. This means software development in an offshore location without having an intense relationship.

	Software development	Team
Distributed	distributed software development multi-site software development cross-site software development	virtual team virtual workgroup geographic dispersed team
Offshore	offshore software development global software development offshore system development international software development	global virtual team

Table 2.1. An overview of all terms that were defined in the previous sections

2.2.3 Outsourced versus insourced

In the previous sections, we cited a lot of terms that people give to the same kind of software development an overview of all terms is given in Table 2.1. Heeks, Krishna, Nicholson & Sahay (2001) mention another term: 'global software outsourcing'.
This term does look like the previous ones, but there is an important difference: this is an example of a way to perform global software development. There are two kinds of relationships that are possible to develop software offshore (Prikladnicki, Audi & Evaristo, 2003; Steenbeek, Wijngaert, van den Brand, Brinkkemper & Harmsen, 2005):

- **Offshore outsourcing:** This means contracting services with an external organization located in another country or time-zone.
- **Offshore insourcing:** Contracting with a wholly owned subsidiary also located in another country or time-zone.

2.2.4 Distances in offshoring

While finding a definition for offshoring, the geographical distance between the cooperating teams seems to be very important. When this distance becomes large in western or eastern direction, time differences also influence the collaboration: it becomes more difficult to communicate.
When teams are far away of each other, three other dimensions also influence the ease of collaboration. In Gumm (2006), Egan et al. (2006), and Holmstrom, Conchuir, Agerfalk & Fitzgerald (2006) all these dimensions are called 'distances', although they are not only geographical distances. They mention five kinds of distances that are also mentioned by other researchers. An overview of all five distances and sources that mention them is given in Table 2.2.

	Location	Time	Culture	Organization	Stakeholder
Bozarth et al. (1998)	x				
Jarvenpaa & Leidner (1999)	x	x	x		
Maznevski & Chudoba (2000)	x				
Cramton (2001)	x	x			
Carmel & Agarwal (2001)			x	x	
Prikladnicki et al. (2003)	x			x	
Sakthivel (2005)	x	x		x	
Damain & Moitra (2006)	x	x			
Layman et al.(2006)	x				
Gumm (2006)	x	x		x	x
Conchuir et al. (2006)	x	x	x		
Egan et al. (2006)		x	x		
Aranda et al. (2006)			x		
Holmstrom et al. (2006)	x	x	x		

Table 2.2. Overview of all distances that influence offshore development

Location
The influence of being located far away from each other is already discussed in the previous sections. This distance is the most mentioned characteristic of offshore software development projects, as can be seen in Table 2.2.

Time
Being located in different time zones influences which communication channels can be used. These can be synchronous or asynchronous (Sakthivel, 2005). This also influences the way in which the globally dispersed teams work together.
Egan et al. (2006) also mention that besides time-zone differences, there are also differences in the perception of time. The importance of time is different in different parts of the world. This influences the way in which software development takes place.

Culture
Working across different national cultures is mentioned as third. Different countries have different cultures. According to Aranda et al. (2006), cultural differences determine the cognitive styles of people. People are different in perception (sensitive or intuitive), input (visual or verbal oriented), processing (active or reflective), and understanding (sequential or global). Carmel (1999) mentions cultural differences with regard to revering hierarchy, individualism vs. collectivism, taking care of the business, risk avoidance, and long-term orientation. These aspects are also mentioned by Hofstede (1980).

Organization
When two different organizations are far away from each other, they have their own methods and company cultures. This is even the case when both organizations are departments of the same company (Gumm, 2006). Carmel & Agarwal (2001) mentioned this type of distance as separate from cultural distances. Nevertheless, some other authors that were mentioned in Table 2.2 focused only on cultural differences on organizational level and not on national level. Because both influence a project, they are mentioned as different distances in Table 2.2.

Stakeholder
The last distance that is mentioned, is ‘stakeholders’. Involving another location in software development, includes extra stakeholders, which are not well known to the rest of the stakeholders:

the department or organization on the offshore site, the government of the country where the other site is located, infrastructure providers, local competitors, international organizations, etcetera (Smith, Mitra & Narasimhan, 1996). This also influences the process in which people collaborate in a software development process (Gumm, 2006).

With this information, the framework from Figure 2.1 can be extended to Figure 2.2. These distances challenge successful collaboration, by negatively influencing communication, coordination, control and cooperation (Carmel & Agarwal, 2001; Casey & Richardson, 2006). The next sections explain how distances influence these factors.

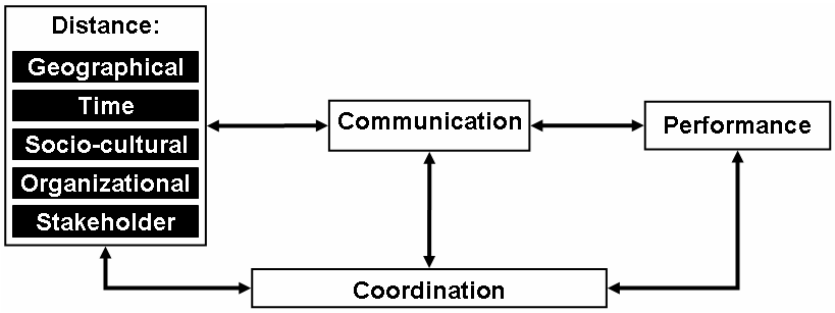


Figure 2.2. The distinguished distances added to the framework.

2.3 When are offshore CSD projects successful?

In our research question, we mention the term success. This thesis is looking for ways to improve the chance on success. What is meant by this term? When can offshore CSD projects be called 'successful'? This section gives an answer on that question.

2.3.1 When are projects successful?

In order to define project success, literature about software development project success needs to be analyzed. Jiang, Klein & Discenza (2002) researched conventional literature about project successes. In earlier days costs, time and savings were the most important success determinants. A project should not cost more than was determined in its budget. A project should be completed on time and the results of the work that was done during the project should lead to benefits that are higher than the costs of the project.
The focus on costs and time is mentioned by many scientific papers, although the last decades extra elements of success are added. According to those papers, successful projects are:

Definition:

*A **successful software development project** is a software development project where the delivered software meets the scope, has at least the expected quality, is completed on time and does not exceed its total budgeted costs.*

Scope

The most important aspect that determines if a software development project is successful, is that is delivered what was requested. The customer has specific requirements, and decides to develop an application that fulfills those needs. Both the customer who requests for software and the software development team agree that delivering the required product is the most important (Agarwal & Rathod, 2006).

Quality
The second most important aspect is quality. Besides the required functionality, a company also desires high enough quality (Agarwal & Rathod, 2006). This will not always be the highest quality that is possible, since the project must be completed with limited knowledge in limited time. Nevertheless, the customer has an expected quality of the result, and this must be reached. If it is not reached, the customer satisfaction will be very low (Procaccino, Verner, Overmyer & Darter, 2002).

Time
The third aspect of a successful project is that is must be completed on time. As was mentioned in the beginning of this section, time is recognized as a very important aspect. Nevertheless, Agarwal & Rathod (2006) state that it is less important than the first two aspects: a company prefers to get an application with the desired functionality a bit late, than an application on time that is not completely applicable. Sometimes, a deadline must be met. An example was the Y2k problem, where software had to be updated in order to function after the change of the century. Austin (2001) has proved that if deadlines are very important, this will cost quality of the system. A company has to decide which one is most important.

Costs
The last aspect of success is costs that are equal to or lower than was budgeted (Procaccino et al., 2002). A project that costs more than was budgeted at start of the project is unsuccessful. According to Agarwal & Rathod (2006), this aspect is the least important. This is not only mentioned by project managers, but also by researchers and customers.

These aspects of success can also be added to the research framework that is presented in this chapter. The result is the framework in Figure 2.3.

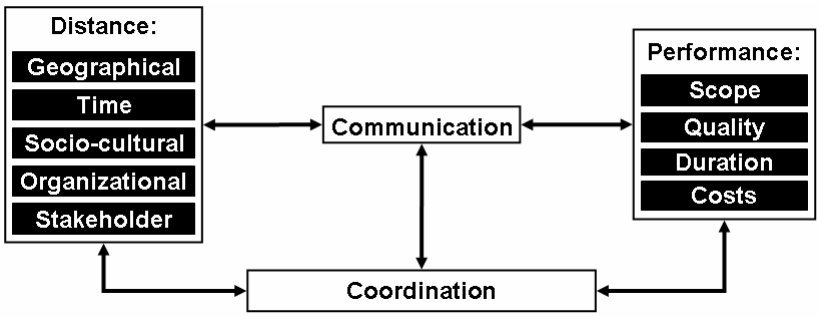


Figure 2.3. Aspects of performance added to the framework

2.3.2 When are projects unsuccessful?

The previously mentioned aspects determine the success of software development projects. How can the failure of a project be defined? Agarwal & Rathod (2006) state that a lot of research is done on project successes, but almost no research is done in project failures. They state that more research needs to be done in this area.
Someone who did analyze failed software development projects, was Linberg (1999). He defines project failure as a project that is cancelled or that does not meet budget, delivery objectives and/or business objectives. Delivery objectives and business objectives include scope, quality and time. Therefore, we can rephrase the project success definition to a project failure:

Definition:

*An **unsuccessful software development project** is a software development project that is cancelled, that delivers software that does not meet the scope, that does not meet the expected quality, that is not completed on time, or that exceeds its total budgeted costs.*

In order to distinguish the areas in which a project is unsuccessful, we introduce four different unsuccessful types of projects. In order to visualize the contribution of different success aspects to the overall project success, we developed formulas that summarize the definitions.

1. A **de-scoped project** is a project where the final result is *less* than the result that was expected by the customer.

This can be formulated as: $\frac{S^r}{S^p} < 1$, where S^r is the realized scope and S^p is the expected scope.

2. A **below-quality project** is a project where the final quality is *below* the expected quality.

This can be formulated as: $\frac{Q^r}{Q^p} < 1$, where Q^r is the realized quality and Q^p is the expected quality.

3. An **out-of-time project** is a project that takes *more* time than was planned.

This can be formulated as: $\frac{T^p}{T^r} < 1$, where T^r is the real time needed and T^p is the planned time.

4. An **out-of-budget project** is a project that costs *more* than was budgeted.

This can be formulated as: $\frac{C^p}{C^r} < 1$, where C^r are the real costs and T^p are the planned costs.

Note that the formulas of de-scoped and below-quality projects are the inverse of out-of-time and out-of-budget projects. This is because the scope and the quality of projects must not be *lower* than expected, while the other two should not be *higher* than expected.
Possible indicators for the above mentioned variable could be, for example: function points (scope), error rate (quality), deadlines (time), and money paid by a customer (costs). A project that is unsuccessful on all four areas of success is more unsuccessful than a project that is out-of-time, but successful in all other three areas. An overall unsuccessful project as defined above could be formulated as:

$$\frac{1}{4} * \left(\frac{S^r}{S^p} + \frac{Q^r}{Q^p} + \frac{T^p}{T^r} + \frac{C^p}{C^r} \right) < 1$$

In practice, some projects will have specific goals. For instance a project should really be completed on time. This is more important than the costs of the project. In such a case, the cost-factor should have a lower weight than the time-factor. When we include this in the formula, this is the result for an *unsuccessful* project (where W^X is the weight factor of X):

$$\frac{1}{W^S + W^Q + W^T + W^C} * \left(\frac{S^r}{S^p} * W^S + \frac{Q^r}{Q^p} * W^Q + \frac{T^p}{T^r} * W^T + \frac{C^p}{C^r} * W^C \right) < 1$$

In a *successful* project, the result is larger or equal to one:

$$\frac{1}{W^S + W^Q + W^T + W^C} * \left(\frac{S^r}{S^p} * W^S + \frac{Q^r}{Q^p} * W^Q + \frac{T^p}{T^r} * W^T + \frac{C^p}{C^r} * W^C \right) \geq 1$$

This formula is less strict than the definition that was given for a successful project in the previous section. In practice, almost all projects have one or more aspects that are not completely successful. Nevertheless, if the other aspects are very successful, the projects will still be called successful. This

is why in the formula some compensation is possible for small negative results, if the weights are not too heavy.

2.3.3 Three different stakeholder groups

Does everyone agree when a software development project is called successful or unsuccessful? Unfortunately, this is not always the case. The main reason is that there are different stakeholders that have their own view on the goals of a project and on the best applicable definition of success (Procaccino et al., 2002):

Software development team
According to Linberg (1999), software developers have their own view of success. Their job satisfaction of development teams is not depending on project success or failure: they want good communication and collaboration.
This is explained by Agarwal & Rathod (2006), who say that you can be internally or externally focused. Since most developers have almost no direct contact with a customer, this automatically turns the focus inwards: meeting budget and delivering on time becomes more important than delivering what the customer expects (scope and quality).

Project management
Agarwal and Rathod also show that project managers have the tendency to focus on budget and time. Procaccino et al. (2002) recognize the same trend: project managers are not always focused on delivering the expected quality and functionality, which is very important to a customer. Their work is approved when they complete projects on time and in budget, which turns their focus inward, too.

Customer
As will be clear from the other two perspectives and the definition of success, customers are not focused on being on time and budget in the first place. Customers wish to get software that meets the requirements, and to get a quality that makes them 'happy' (Agarwal & Rathod, 2006).

Three different perspectives in a project lead to different opinions about the success. As will become clear in the next section, offshore CSD makes the difference between the perspectives larger.

2.3.4 Perceiving success of offshore CSD projects

Offshore software development influences the project characteristics in terms of scale, scope, duration and difficulty (Smith, Mitra & Narasimhan, 1996). The different aspects of success are influenced by developing offshore.
One example is the impact on costs of the software development. Besides production costs, transaction costs are also involved in a CSD project. These costs involve the costs of sourcing, contracting and managing the project (Sakthivel, 2005). Since the estimation of these costs is more elaborate for offshore development projects, project managers are focusing more on costs: they tend to focus on internal success factors.
In offshore development projects, the developers are mostly located offshore, while the (business) architects and project managers are mostly located onshore. Since the developers are located offshore, they only communicate with the architects, analysts and managers of the onshore development team. The chance on having contact with the customer is much less than would be the case in an onshore distributed software development project. This turns the focus of the developer more internal than was the case in a non-offshore situation.
Not only the developers and the project managers are influenced when a project is partly executed offshore. Procaccino et al. (2002) state that project success is predicted and expected by the customer

and the end users, when they have a high level of confidence in a project manager and a development team. In an offshore development project it is harder for a customer to predict success, since the development team is far away in terms of location, time, culture, organization and perspective.
These impacts of offshore development on the perspectives of success stress the need for integrating all those views. The integration of the different stakeholders is stressed by Jiang, Klein & Discenza (2002): "Efforts should be made to establish partnerships with all stakeholders to provide a more complete control process for information systems development." They state that besides all mentioned aspects of success, the integration of all perspectives and the feedback of all stakeholders should be taken into account.

2.4 Why is the success of offshore CSD projects different?

In the previous section, we defined successful software development projects. This section zooms in on the success of offshore software development projects.
Linberg (1999) states, that developing offshore adds different temperaments to the development team. The increased diversity leads to a stronger team and a better quality. This advantage is already mentioned in the introduction chapter.
Are offshore software development projects more successful? The introduction chapter explained that this is not always the case: the goals (such as cost reduction) that are the business case for offshoring are often not or partly met.
In the definition of offshore software development, we mentioned that people hardly see or speak each other face to face. This was brought forward as the main reason to call such arrangements 'offshore' instead of 'distributed'. According to the definition of offshore software development that is given by Sakthivel (2005), this activity therefore always requires the use of communication technologies. Therefore, communication is different in an offshore CSD project, in comparison with an onshore distributed CSD project.
This touches the heart of the influence of offshoring on software development. As will be explained in the rest of this section, the distances in an offshore development lead to less successful results because communication and knowledge exchange are influenced negatively.

2.4.1 Offshore distances influence communication negatively

Communication is one of the main challenges in offshore CSD (Chan & Chung, 2004; Paasivaara & Lassenius, 2006; Layman et al., 2006). Before elaborating on the reason for this challenge, a definition of communication should be given.
Rosengren (2000) states that communication is sharing knowledge on individual, group or organizational level. In a more detailed definition, he defines communication as "Intersubjective, purposive interaction by means of doubly articulated human language based on symbols". This definition focuses on general communication between two or more subjects.
Gowda and Polzella (2006) give a definition that focuses on communication in a specific context: a project. Since our research is about CSD projects, this definition will be used for this thesis.

Definition:

Project communication *is the exchange of information about the task, resources and organizational issues (Gowda & Polzella, 2006).*

Carmel & Agarwal (2001) stress that the exchanged information should be complete and unambiguous for the communication to be successful. They state that this is a lot harder when a team is distributed over large distances (see Figure 2.4). This is confirmed by Aranda et al. (2006).

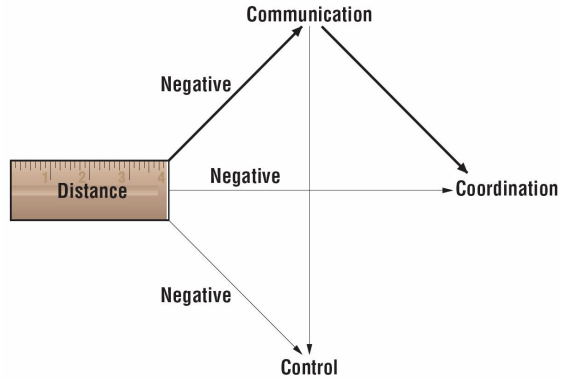


Figure 2.4. Impacts of distance (Carmel & Agarwal, 2001)

According to Egan et al. (2006), communication is not only negatively influenced by geographical distances, but also by the time and cultural distances.
Rosengren (2000) states that there can be multiple distances that influence communication: the more heterogeneity in age, gender, personality, education, occupation, social class and nationality, the more negatively is the influence on communication.
When a team is communicating on international level, this increases the difficulty to communicate with the extra distances (Rosengren, 2000). In one of the previous sections we explained that offshore development teams face distances in terms of location, time, culture, organization and stakeholders. These additional differences are the main cause for poor communication in an offshore CSD project. This poor communication is explained by the lack of three important communication aspects:

- **Contextual information:** According to Cramton (2001), it is very difficult to communicate and retain contextual information in a distributed project. When people do not meet face-to-face, they miss information about the context in which the project takes place on the other location.
- **Informal communication:** Layman et al. (2006) stress that informal communication is needed to exchange all knowledge. A lot of information is exchanged during lunch time, coffee breaks or other informal meetings in an organization. This communication is not available when teams are far away from each other and never meet informally.
- **Social relations:** Argote, McEvily, and Reagans (2003) mention that social relations are very important when knowledge must be exchanged. Social relations are much stronger when people meet regularly face-to-face, which is not the case in offshore CSD projects.

Layman et al (2006) state that less informal communication leads to a low level of trust and low awareness of the work and the process that is going on. In other words: all three aspects of communication negatively influence each other when one of them is not as high as expected. Communication is negatively influenced by developing offshore. All team members together can have the required knowledge to develop the required software, but this cannot happen when the knowledge is not easily shared. Hsieh (2006) says that overcoming cultural diversity is about knowing and not about knowledge. The collective ‘knowledgeability’ is subject of the next section.

2.4.2 Poor quality of communication leads to less knowledge exchange

Cramton (2001) defines the collective ‘knowledgeability’ as ‘mutual knowledge’. He defines it as “knowledge that the communicating parties share in common and know they share it”. Nowadays, mutual knowledge is very valuable to organizations, and is therefore considered as intellectual capital by Stewart (2001). Mutual knowledge can only be established via communication (see Figure 2.5).

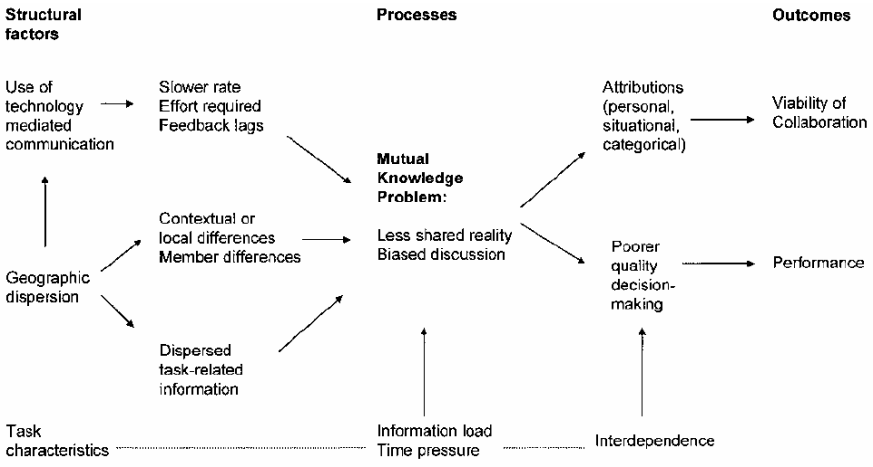


Figure 2.5. Likely impact of dispersion and mediated communication on mutual knowledge and on collaborative outcomes (Cramton, 2001)

At the start of an offshore CSD project, knowledge is situated at different locations. In order to share all knowledge, and hence create mutual knowledge, communication is needed. Since communication is poor, the creation of mutual knowledge is poor. Furthermore, because there is poor knowledge exchange, communication is also more difficult (Sole & Edmondson, 2002): People on one location do not know what people on the other location know.
This is visualized by Helms and Buijsrogge (2005). They introduced a technique to visualize and analyze knowledge management bottlenecks in an organization. In a specific knowledge area, they depict knowledge actors together with interconnecting knowledge flows. Each actor has a specific kind of expertise and experience. Each knowledge flow has a velocity (speed) and a viscosity (richness). In a network graph they show that some knowledge can be unreachable for some team members (for an example, see Figure 2.6). According to Helms & Buijsrogge (2006), 'sub-communities' can be distinguished in a team that cooperates, by using the top-down Girvan Newman algorithm. This algorithm starts with a group as a whole and splits it in smaller groups. The sub-communities share a lot of knowledge within their community, but less knowledge with other communities. Another type of communities are 'cliques', who are derived with the help of bottom-up clique analysis (Hanneman & Riddle, 2005): people who are closely connected are cliques, who are connected with each other or with single people on a higher level.

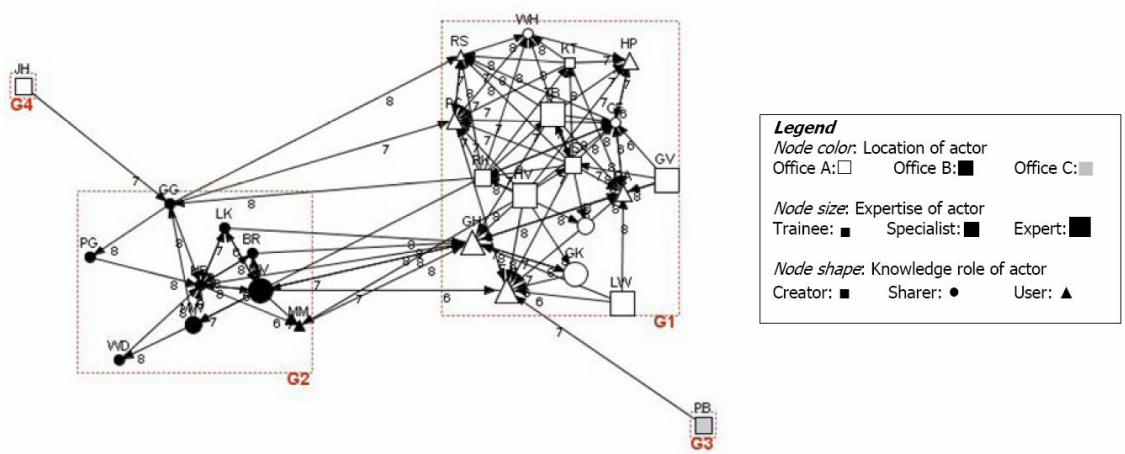


Figure 2.6. Example of a knowledge network (Helms & Buijsrogge, 2006)

Which part of the knowledge should be exchanged, in order to get successful CSD projects? According to Heeks et al. (2001), successful projects need synchronization of tacit knowledge,

informal information and cultural (social) issues. Sharing tacit knowledge is also mentioned by Gertler (2003) as a critical success factor for good collaboration.
Nonaka and Konno (1998) explain that in order to exchange tacit knowledge, someone needs to communicate it to another person. Offshore CSD projects often focus on sharing explicit information about technical aspects of the software. If people need to know process related information, they often tend to ask this to external people on their own location instead of team members on the other location (Ehrlich & Chang, 2006).
This is the bottleneck in offshore CSD projects: communication is poor, which negatively influences the possibility to share (tacit) knowledge. Bruegge, Dutoit and Wolf (2006) underline this statement by saying that informal communication is needed to disseminate implicit knowledge.

2.4.3 Less knowledge exchange leads to lower success

Successful knowledge exchange leads to successful organizations, according to Jashapara (2004). We explained that the increased differences in an offshore CSD project lead to poor communication, which leads to a poor realization of mutual knowledge. Cramton (2001) states that failing to establish mutual knowledge leads to poor productivity. In short, this means that poor communication in the beginning leads to poor team performance in the end. This is confirmed by Egan et al (2006), who not only state that team performance is influenced by communication, but also team satisfaction (see Figure 2.7).
Faraj and Sproull (2000) researched the relationship between the coordination of expertise and the performance of a project. They found that it is important to know about all the knowledge in the project: it is not important to know everything. It is important to know that knowledge is available and who has access to it.

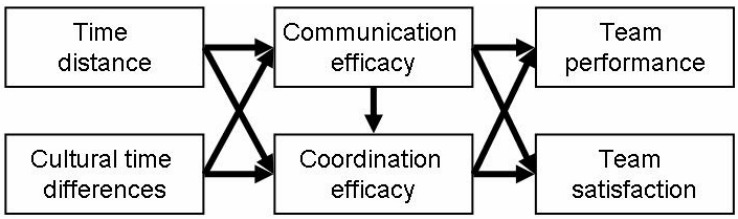


Figure 2.7. Proposed correlation model by Egan et al. (2006)

2.4.4 Lower perceived success influence communication negatively

The previous sections showed that communication influences mutual knowledge, and that mutual knowledge influences communication. We also showed that mutual knowledge influences the outcome (performance) of a project. In this section, we will elaborate on the influence of perceived project success on the quality of knowledge exchange.
In the previous section we already mentioned the influence of communication on team satisfaction. The attitude of the team has its influence on the quality of communication and knowledge exchange in a project. Another important attitudinal aspect that is influenced by success is trust. According to Jarvernpaa & Leidner (1999), trust is not very strong in a globally distributed team. If the team members expect a negative outcome of the software development project, this will negatively influence their actions and communication behaviors.
This is explained by Cramton (2001): if a (part of a) project tends to fail, people will attach attributes to their relations with the people on the other site. These attributes can be personal or situational (someone is inadequate to work with or he is just in the wrong organizational situation). The attributes are constructive (positive) or non-constructive (negative) to the relation between the members of a team. Since most offshore projects tend to become failures, this negatively influences the relationships

between the team members on both locations. This negatively influences the communication and the creation of mutual knowledge. In the end, this influences the success of an offshore software development project.
Previous sections explain the relations between distance, communication, and performance in the research framework. They also add knowledge exchange as an essential part of communication to the framework, as is depicted in Figure 2.8. The final element in the framework, coordination, is explained in next section.

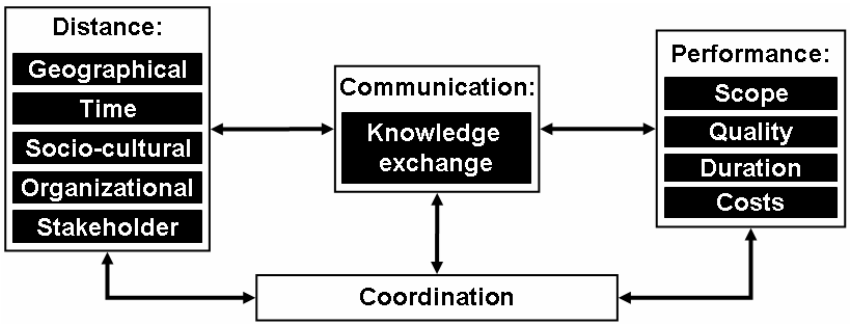


Figure 2.8. Knowledge exchange added to the framework

2.5 How to influence the success of offshore CSD projects?

How can the negative influence of all the different distances on the success of offshore development projects be reduced? An organization can choose to locate all team members on one location, but this is not an offshore scenario. If the team has to develop offshore, how can they improve their collaboration? This section explains that communication needs to be improved, and that in order to improve communication, several coordination measures have to be taken.

2.5.1 Focus on communication and coordination

Layman et al (2006) state, that communication plays a critical role in the success of global software development. In the previous section this was already mentioned by showing that communication is influenced by the offshore distances and that communication influences the success of the project. Communication is not only the challenge in offshore CSD projects, but it is also a benefit that can be used to overcome the problems. Communication can be used to overcome the distances (Paasivaara & Lassenius, 2006). Cusick & Prasad (2006) also confirm this statement, by saying that a focus on communication is one of the key success factors of successful offshore collaboration.
In Figure 2.4 and Figure 2.7, communication is a central process that is translating the negative impact of the offshore distances into negative performance. These figures also mention control and coordination. Carmel and Agarwal (2001) define control as adhering to goals, policies, standards, quality levels. They define coordination as orchestrating the integration of all activities. In essence, both of these activities are focused on managing the whole project. This is the reason why Egan et al (2006) only mention coordination. In the rest of my research, we will use Egan et al.'s definition, where coordination means all managing activities that influence the project and thus the communication.
According to Aranda et al. (2006), not only communication is challenged in an offshore situation, but also the control/coordination. If coordination is not sufficient, team performance and the final result will also not be sufficient (Egan et al., 2006).
Herbsleb and Grinter (1999), say that coordination is essential to align all efforts and knowledge in a project. Coordination is needed to streamline the communication in a project. Communication is negatively influenced by the distances. Therefore, project coordination should be adapted, in order to

get the quality of communication on the level as it should be. This turns out to be one of the main sources why offshore projects fail: project management is not adapting to the new offshore situation that is different from a distributed situation in the home country (Nicholson & Sahay, 2004).

2.5.2 Coordination measures

How should coordination be improved, in order to get successful offshore communication and thus successful projects? Chan and Chung (2004) state, that both process and project management mechanisms have to be analyzed.
Aron and Singh (2005) give the following solutions: pick the right processes, calculate both the operational and the structural risks and match organizational forms to the needs in an offshore environment. These activities are all examples of coordination.
Casey and Richardson (2006) mention the following coordination areas that need attention: organizational structure, risk management, infrastructure, process, conflict management, team structure, and team organization. These and other areas are also mentioned in other literature. Sabherwal (2003) categorizes them into four main categories:

- **Standards:** Standards include all methodologies, rules, dictionaries, procedures etcetera. Together with the plans from the next category, Kraut and Streeter (1995) call these formal impersonal procedures. They are focused on delivering the right product (scope) with the right quality.
- **Plans:** This category includes all schedules, milestones and other plans. They all are focused on delivering the product on time and in budget (see also the list of success aspects).
- **Formal mutual adjustment:** Kraut and Streeter (1995) call this category 'formal interpersonal procedures'. They are specifically focused on coordinating the formal communication. This category includes the creation of hierarchies, the planning of formal meetings, etc. Delegation is also an important aspect, according to Zhang et al. (2006). They state that delegation in the right way affects team performance over distance: people have certain responsibilities in the project or the process.
- **Informal mutual adjustment:** As was already mentioned in the previous sections, informal communication channels can also be managed. Explicit coordination measures are not sufficient to coordinate informal coordination (Herbsleb & Grinter, 1999). Therefore, some small measures can be taken to increase the chance on informal communication between (groups of) people.

Kraut and Streeter also mention two extra mechanisms: 'communication', which is looking at which communication facilities are available, and 'interpersonal network', which focuses on individual contact with people from outside the network. These two categories are mainly aspects of communication and not of coordination. Therefore, they are not mentioned by Sabherwal (2003) as coordination categories.
One category that is not mentioned by Sabherwal, which is important in teams that collaborate over distance, is team selection:

- **Team selection:** The knowledge and experience of all team members together influence the success of the project. This is mentioned by Zhang et al. (2006). They state that all knowledge and experience of the team members together influence the overall team maturity. Zhang et al. define team maturity on three dimensions: team technical competency (with regard to the task that has to be done), team motivation (with regard to the individual and the team goals), and distributed teamwork skills (the ability to cooperate in a distributed environment). The

selection of the team members at start of the project is therefore also a way to influence project success.

The five previously mentioned categories of coordination measures aim at improving communication and knowledge exchange. Carmel and Agarwal (2001) state, that in order to overcome cooperation problems in offshore teams, temporal distance and cultural distances have to be reduced. This can be done by working at night or selecting countries in the same time-zone to offshore, and by facilitating cultural exchanges. Organizational distances and stakeholder distances can also be reduced. Herbsleb and Grinter (1999) propose the introduction of team members ('boundary spanners') on both sides, who have more experience with the culture, stakeholders, organization and time on the other location. By choosing a set of coordination measures from the previously mentioned categories, the communication, knowledge exchange, and in the end the success of a project can be influenced.

2.5.3 Balance the success factors and coordination measures

In the introduction, we mentioned that the main reason to offshore was the reduction of software development costs. The expectation of cost reduction is mostly not realized (Kuni & Bhushan, 2006). The main reason for this is that not all costs are included in the cost estimation. The expected cost reduction is mostly based on lower wages in other countries. The above mentioned coordination measures should be added or changed in an offshore software development project. This costs extra time and money. Sakthivel (2005) mentions sourcing, contracting, project management and risk estimation costs as part of this increase.
In short, we state that:

1. Offshoring saves money
2. Offshoring requires communication quality
3. Communication quality requires coordination measures
4. Coordination measures cost money

The successes in terms of scope, quality, time and costs are dependent on the choice of coordination measures. A firm has to choose the right coordination measures, with regard to the desired success. If costs are important and time and quality are less important, less coordination measures are required. If time or quality is important, the offshore CSD project needs extra or changed coordination measures. This is also mentioned by Harmsen, Lubbers and Wijers (1995), who emphasize the relation between a situation in which a project is being done, and the success at the end of the project (see Figure 2.9). In their S^3 model they state that a scenario influences the final success of a project. If the scenario focuses on costs, the success will also be with regard to costs.

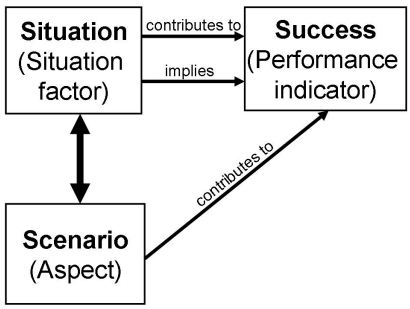


Figure 2.9. The S^3 model of Harmsen, Lubbers and Wijers (1995).

2.6 Framework of research

This section combines the scientific literature that was presented in the previous sections, and shows how the success of offshore CSD projects can be managed.
Layman et al. (2006) say that the heart of a global software development process is formed by communication and coordination. In the previous sections, we showed that offshore development affects communication. We concluded from literature that communication affects the mutual knowledge creation, and that this in the end influences the outcome of the project. We also explained the importance of coordination in this whole situation. How can the success of an offshore project be influenced? The steps that we explained in the previous sections provided an overview of how success of offshore CSD projects can be influenced. In Figure 2.4 we saw that Distance influences Communication and Coordination. Figure 2.5 showed that Communication and Knowledge exchange were connected with Performance. Figure 2.7 showed that two types of distance influenced Communication and Coordination. This influenced the team Performance and Satisfaction. Figure 2.9 also connected the scenario (Coordination) and situation (Distances & Communication) to the Performance of the project. When we combine all this information, we conclude that there are four main concepts that influence each other: distance, communication, coordination, and performance. For each of these aspects, we defined which types can be found in literature. Figure 2.10 depicts the concepts, their relations, and their categories. Below, a summary of all items in the figure are described:

- **Distances:** Offshore CSD project face not only geographical distances, but also time, socio-cultural, organizational and stakeholder distances.
- **Communication:** These distances influence communication between the team members of a project. Larger distances lead to less efficient communication, and shorter distances lead to better good communication. The efficiency of the communication has its impact on the knowledge exchange in the project.
- **Performance:** The amount of knowledge exchange influences the performance expectations and the final performance: the amount in which the requirements are met in the end product, the quality, if the product is delivered on time, and the total amount of costs. The performance influences the communication, since people accuse or praise each other.
- **Coordination:** The expectation of a specific performance due to the distances and the quality of communication leads to changes in the coordination measures. Coordination measures can decrease the distances in an offshore environment, improve the communication in the project, or change the customer's expectations of the end result (performance).

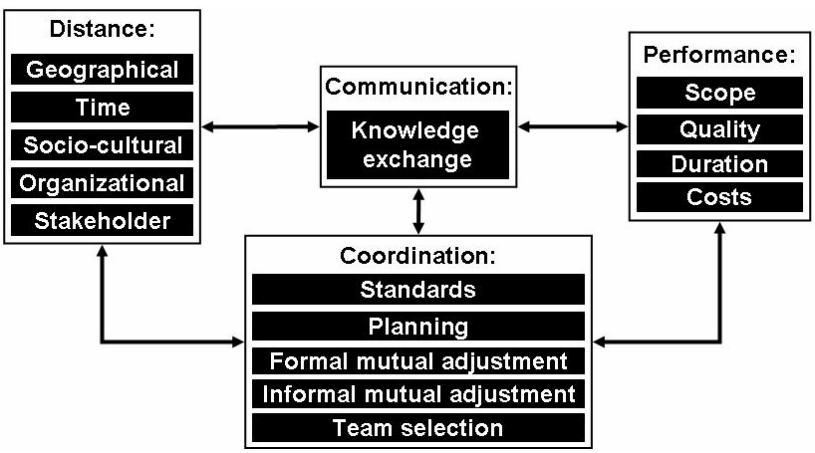


Figure 2.10. An overview of the research framework

2.7 Research focus

In our research, we will focus on the communication and the coordination aspects of a CSD project. The influence of the distances is already researched a lot, but the impact of coordination on communication and the impact of communication on knowledge exchange and performance are not yet researched in depth. Researchers such as Herbsleb et al. (2001), Prikladnicki, Audi and Evaristo (2003), Rottman and Lacity (2006), etc. came up with lists of possible solutions, but did not show if these solutions led to success or not. In this research, we will analyze communication and coordination in successful and unsuccessful offshore CSD projects, and show how they lead to success or failure.

2.8 Hypotheses

From the literature that is described in this chapter, we define answers on the sub-questions that were described in the previous chapter. Together, these answers are a hypothesis on the main question. Based on the described literature, and based on several interviews that we had with project managers from the back-office, managers from the customer side and people from the front-office, our hypotheses are:

Question A.1: What is the difference between an onshore and an offshore CSD project?
Hypothesis: ***In an offshore CSD project two or more teams are located in different countries or time zones. They cannot meet regularly face to face.***

Question A.2: When are offshore CSD projects successful?
Hypothesis: ***Offshore projects are successful when they are completed on time and within budget, with the expected scope and quality.***

Question A.3: Why is the success of offshore CSD projects different, compared to onshore CSD projects?
Hypothesis: ***This is different because geographical, time, cultural, organizational, and stakeholder distances negatively influence the coordination and the communication in a team. In the end, this influences the project success.***

Question B.1: Which coordination measures exist in a CSD project?
Hypothesis: ***Five categories kinds of coordination measures can be applied: application of standards, planning, formal mutual adjustment, informal mutual adjustment, and team selection.***

Question B.2: Which coordination measures should be introduced, in order to positively influence the success of offshore CSD projects?
Hypothesis: ***In an offshore CSD project all coordination categories should be applied. Special attention should be given to informal mutual adjustment, for example the introduction of an onsite coordinator and facilitating contact between the customer and the back-office.***

This chapter aimed at finding a hypothetical answer from literature on the research questions. The rest of this thesis describes our own research that in the end falsifies or confirms the hypotheses. Before we were able to do that, we had to find an answer to questions about how to analyze communication and coordination:

1. How to analyze communication in an offshore CSD project?
2. How to analyze coordination in an offshore CSD project?

These questions will be answered in the following methodological chapter.

3 RESEARCH METHOD

In order to answer the sub-questions of this thesis, offshore CSD cases have been researched. We interviewed the onshore project managers and most of the offshore project managers. In a couple of cases, the customers were interviewed, too. The interviews delivered information about the distances, communication, coordination, and success of the projects. The interviews also gave information about the team members that were involved in some projects. Those team members were asked to fill out a questionnaire about communication during the project. From this information, we were able to draw knowledge/communication network diagrams as were described in previous chapter, and get additional project data.
This chapter describes which cases were researched, how project distances and success were made operational, and how data was collected, processed, and analyzed. At the end of this chapter, the issue of validity of the research is discussed.
Figure 2.10 gave an overview of the relations between distances, communication, coordination and results in a CSD project. This was based on literature research, and answered mainly research sub-question A. Based on scientific literature we showed that there is a relation between those aspects. We did not yet find what the best way is to set up a CSD project. To find that way, which is the subject sub-question B, we conducted an empirical research in practice. We chose to research the communication, coordination and results of 19 completed projects in depth. It was not possible to choose a larger set of projects to perform statistical significant research, since there are currently not many completed offshore CSD projects about which organizations are willing to share information. According to Swanson & Elwood (2005), qualitative research should analyze more than one case. Therefore, we chose to perform a qualitative research that analyzes a couple of projects in depth.

3.1 Choice of research units

In order to be able to answer the research questions, the researcher and its supervisors from both the university and Capgemini selected 19 software development projects that were executed partly offshore. All projects were completed. We chose completed projects, because at the end of a project the result is visible. In such a case it is possible to say if the project was successful or not.
In all projects, a part of the team was located in The Netherlands. The other part of the team was located in another part of the world. Most of the projects had links with Capgemini, which made it possible to interview the project managers on both the Dutch and the Offshore sides. All offshore Capgemini projects were executed in India. This caused that 17 of the 19 cases were offshored in India. In order to have more valid results, we also selected two projects that were not executed in India. The other two projects were not conducted by Capgemini, and offshored to Romania and Malaysia.
To be able to research successful and unsuccessful projects, we selected a set of projects that varied on the level of successfulness. In order to have a representative set of projects, we chose a set of projects that varied in size, responsibility, duration, type of customer, programming language, costs, type of contract and organizational complexity. An overview of the deviation of these aspects is given in Table 3.1. Below, a short description of the dimensions is given:

- **Responsible organization:** We intended to analyze projects that were executed in different surroundings. Therefore, we chose five projects that were not led by Capgemini, eight projects that were done by the business unit “Outsourcing” and six projects that done at the business unit “Technology” (see also section 2.4 on page 7).

- **Size:** In this research, we focus on collaboration and communication in teams. The amount of people that were involved in the project influences the team. Therefore, large, medium and small teams were examined in this research.
- **Duration:** Some offshore projects are completed in a few months. Other projects take many years. In order to have a complete set of projects, both short and long lasting during projects were selected.
- **Type of contract:** Projects can have a fixed budget and a fixed end date. This is not always the case: some projects are executed without fixing, since a sound estimation can not be made. In our set, we included both types of projects.
- **Organizational complexity:** Sometimes, a software development project only consists of a back and a front-office. Mostly, this is not the case. In all Capgemini projects, the customer is a third party that is also involved in the project. Sometimes, other consultancy or software development companies are also involved in the projects. This increases complexity. The selected set contains both a small and a large amount of stakeholders.
- **Type of customer:** In order to have a wide representation of projects, we selected different areas where software was developed. Both governmental and non-governmental organizations are represented, financial, industrial and service organizations are included in our research.
- **Total costs:** We chose both cheap and expensive projects, in order to have a representative set.
- **Programming language:** A large part of the software that was developed was written in Java. To be able to draw general conclusions for all programming languages, we also chose software that was developed in other programming languages.
- **Back-office location:** As was already told in the beginning of this section, two projects had a back-office that was not located in India.

Responsible organization:	
Capgemini Outsourcing	6
Capgemini Technology	8
Non-Capgemini	5
Total	**19**

Size:	
Extra large (> 75 people)	5
Large (between 25 and 75 people)	3
Medium (between 10 and 25 people)	5
Small (< 10 people)	6
Total	**19**

Duration:	
Canceled	2
Shorter than 6 months	7
Between 6 and 12 months	4
Longer than one year	6
Total	**19**

Type of contract:	
Fixed price	5
Time & materials	9
Combination	5
Total	**19**

Organizational complexity:	
Two or three stakeholders	8
Four or five stakeholders	6
Six or more stakeholders	5
Total	**19**

Type of customer:	
Government	3
Product software company	4
Bank	4
Insurance	3
Professional services	2
Industry	1
Telecommunications	1
Retail	1
Total	**19**

Total costs:	
Less than €0,1 mln.	3
€0,1 mln. <> €1 mln.	4
€1 mln. <> €5 mln.	6
More than €5 mln.	4
Unknown	2
Total	**19**

Programming language:	
.NET	4
Java	10
Coolgen	1
Other	4
Total	**19**

Back-office location:	
India	17
Romania	1
Malaysia	1
Total	**19**

Table 3.1. An overview of the characteristics of the selected projects

3.2 Making the research framework operational

Before the interviews and the questionnaire are described, it is necessary to make two elements more operational. First, the amount of offshoring needs to be clear, in order to show how large the distances were within the project teams. Second, the final performance needs to be clear. All interview and questionnaire data is compared with the success of the projects. Therefore, an explanation of how success was determined is necessary. Next sections describe how both elements were made operational.

3.2.1 Activities and roles in the cases

In order to get an overview of the project team and their location, information is needed about the involvement of both countries in the project. This information can be collected by analyzing the activities of the projects, or by analyzing the roles of the team members onshore and offshore.

Activities in the projects
In order to analyze the activities in the projects, we made an overview of the general activities in a CSD project. These activities were based on scientific literature, generally applied software development methods, and a maturity model that defines the phases that a CSD project should include. The following sources were used to define the activities:

1. Waterfall (Davis, Bersoff & Comer , 1988)
2. CMMi (Kasse, 2004)
3. Dynamic Systems Development Method (DSDM: Schuh, 2005)
4. Rational Unified Process (RUP: Gibbs, 2007)
5. Gartner (2005)
6. Cusick & Prasad (2006)
7. Caprihan (2006)

These sources were selected because they have different perspectives (scientific, organizational, qualitative point of views). The three development methods (1, 3, and 4) were chosen because they are often applied in CSD projects, as becomes also visible in the case descriptions in the next chapter. Capgemini applies a variant of RUP ('Deliver'). That was another reason to include RUP.
We compared all main activities in these CSD software development methods in order to define a general CSD method. The cases that ware researched used different development methods. Thus, a general method was needed to be able to compare the different project situations with each other. An overview of the comparison is given in Table 3.2.

	Business & Requirements		Design		Build & Test			Deploy	Supportive	
1	System requirements	software requirements	Preliminary design	Detailed design	Code & debug	Test & Pre-operations		Operations & Maintenance		
2	Requirements & planning		Design		Implementation	Testing (unit test & verification test)		Packaging & Validation	General availability	
3	Feasibility & business study		Functional model	Design	Build			Implement		
4	Business modeling	Requirements	Analysis & design		Implementation	Test		Deployment	Project management	Environment
5	Analysis / requirements / business modeling		Design / architecture		Construction	Unit test	System test	Deployment	Configuration management	Change management
6	Concept		Analysis	Design	Construction				Quality assurance	
7	Requirements		Design		Build	Test		Deploy		

Table 3.2. Activities of a couple of CSD methods compared (the numbers refer to the CSD methods that were mentioned above)

Each development method had a **business & requirements** phase, in which the business was *analyzed* and the *functional and non-functional requirements* were written down. After this phase, the **design** was done. This was mostly done in two main activities: the *preliminary design* (also called 'functional design') and the *detailed design* (also called 'technical design'). Then, the software is **built and tested**. This is mostly done in two phases: firstly, different *units* of the software are built. When these parts are *implemented*, *tested*, and *debugged*, they are integrated into one system: the *system integration*. After integration, the system is also *tested* and *debugged*. At the end of a project, in the '**deployment**' phase, the system is *deployed* and the future users are *trained*.
Besides the main CSD activities, there are also some supportive activities that support the project during a couple or all phases. One 'supportive' activity is project management. Project management is essential, but not in the primary development process. A project manager leads the whole project. Real supportive activities are quality management, configuration management, and change management. An overview of all described activities is given in Figure 3.1. The non-supportive activities are often executed in the order as described. In agile methods such as RUP and DSDM there is an iterative approach. This means that a project can be split up and executed in parallel parts or that couple of activities can be repeated. Nevertheless, also in agile methods the order is still from business study, through design, build and test, to deployment or implementation (Schuh, 2005).

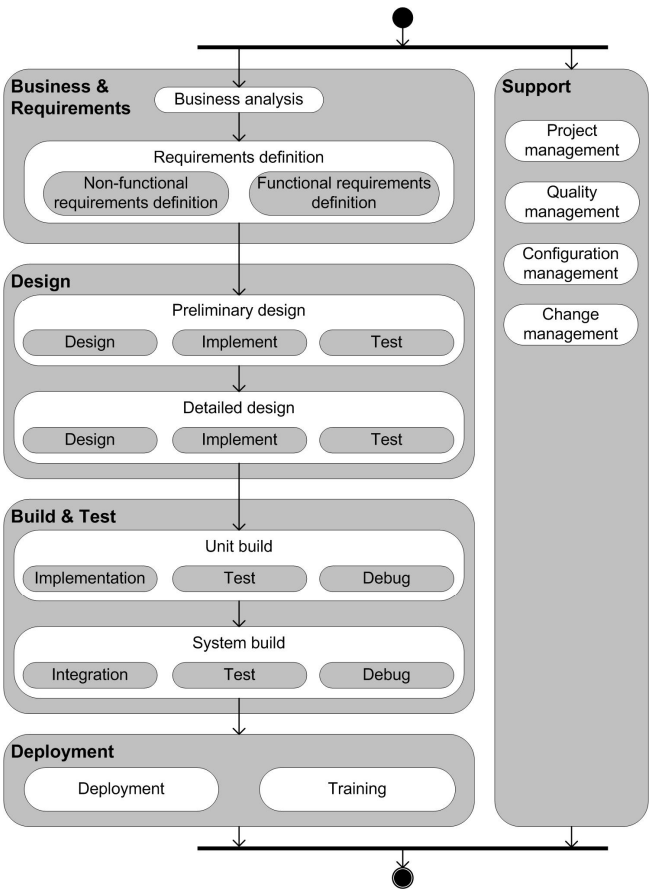


Figure 3.1. An overview of the general software development method

From the described activities in the general CSD method, we made a sheet that was given to the project managers during the interviews (Appendix 2). The project managers were asked to write down how many team members were involved in each activity.
The goal of this sheet was, to see which activities in the model were offshored and to what extent. This was shown by dividing the number of offshore team members by the total number of team members that was involved in a certain activity ('offshore percentage', sometimes called 'offshore

mix'). If for instance 10 people from India and 2 people from The Netherlands would work on the unit build, we conclude that this activity is 10/(2+10) = 83% offshored.
This information gave knowledge about the distances in the project activities, and also about the coordination of the project: the sheet tells about the choices that were made about responsibilities and cooperation for each activity.

Roles in the project teams
Besides the activities that were described previously, we also asked all project managers to describe the roles of the team members that were onshore and offshore. In order to get an overview of all roles that are important in CSD projects, three software development methods and one maturity model that defines important CSD roles were researched:

1. Open Unified Process (The Eclipse Foundation, 2007)
2. CMMi (Kasse, 2004)
3. DSDM (Dynamic Systems Development Method, 2007)
4. RUP (Gibbs, 2007)

These sources were chosen because they clearly define roles, and are applied in many CSD projects. The comparison of the roles in these four methods can be found in Table 3.3. On the left side we defined the roles and their names that we used in our interviews. We used these roles to make another sheet (see Appendix 3). The managers were asked to describe how many team members in both countries performed certain (main) roles. The same offshore percentages were calculated here as was described for the CSD activities.

	OUP	CMMi	DSDM	RUP
Client	Stakeholder	Senior management	Executive sponsor Visionary	IT manager Project manager PMO Project architect
User			Ambassador user Advisor user	Lead user representative Internal project champion
Project manager	Project manager	Middle management	Project manager	Project manager
Leader		Project leaders	Technical coordinator Team leader	Technical team lead
Developer	Developer	Practitioners	Developer	Developer
Analyst	Analyst	Process group	Business architect	Requirements analyst
Tester	Tester	Integration and system testing	Tester	Tester
Architect	Architect	Systems engineering	System integrator	Architect
Quality manager		Quality assurance	Quality manager	Quality assurance
Configuration manager		Configuration Management		Configuration manager
Facilitator		Measurement team	Facilitator Scribe	Toolsmith

Table 3.3. A comparison of roles in four CSD methods.

3.2.2 Performance of the cases

The previous chapter described how to measure the success of projects. It defined that it can be split up into success concerning scope, quality, time and costs. The success formula should be filled in to describe accurately the success of the project:

$$\frac{1}{W^S + W^Q + W^T + W^C} * \left(\frac{S^r}{S^p} * W^S + \frac{Q^r}{Q^p} * W^Q + \frac{T^p}{T^r} * W^T + \frac{C^p}{C^r} * W^C \right)$$

Unfortunately, it costs a lot of effort to know how much weight must be applied to the different aspects, and to know how large scope, quality, time and costs were at start and at end of the project. Possible indicators could be function points for scope, error rate for quality, deadlines for time, and money paid by a customer for costs. Unfortunately, all of these were hard to measure: only half of the projects used function points, not all errors were registered and communicated, and deadlines were mostly not on paper. Almost all projects experienced that the customer wanted to change some parts of the software during the project. This was possible, but the costs and time were recalculated from the current status. This made that there were many planned and realized costs and time plans. Information was scattered over multiple organizations, countries and people. It was not universally available in one collaboration tool.

Therefore, we asked the front-office project managers to indicate how successful the project was on a scale from 1 to 5 (1 was very unsuccessful, 2 was unsuccessful, 3 was successful nor unsuccessful, 4 was successful, and 5 was very successful). The answer for scope *S* would be an indication of the result when we would estimate and divide S^y by S^p and multiply it with the weight of scope. If scope success was very important, but not completely reached, it would get a low success score. The same was done for *Q*, *T*, and *C*. In 14 projects we also asked these questions to the customer and/or the back-office manager. For these projects, we used the average of the answers to indicate the success. If one of the aspects is below 3, this indicates that at least one of the project mangers has said that that aspect is not successful (he/she answered '1' or '2'). In that case, we call the aspect is *unsuccessful*. The formula for an *unsuccessful* project is a combination of all answers:

$$\frac{1}{4} * (S + Q + T + C) < 3$$

If one of the aspects is above 3 and below 4, at least one person has said that this aspect was not successful nor unsuccessful (answered '3'), or even unsuccessful (answered '1' or '2'). In such a case, we don't call this aspect unsuccessful, but also not successful. In the remaining of this thesis, 'unsuccessful' will be used for project with a score below 3 (projects that are unsuccessful), and 'not successful' for a score below 4 (they are not successful).

If one of the aspects is equal to or above 4, on average everyone agrees that such an aspect is successful. We call such an aspect successful. The formula with the answers for a *successful* project would be the following:

$$\frac{1}{4} * (S + Q + T + C) \geq 4$$

These formulas give a rough indication about the success of the projects. *S*, *Q*, *T*, and *C* give a rough indication about the success of the project on each item. The answers were used to select the successful and the unsuccessful projects on all aspects of success. As will be described later on, the team members of nine projects confirmed the successfulness that was determined by applying the method that was described in this section to their projects.

3.3 Description of research method

The purpose of our research is to find the coordination measures that improved communication and led to successful collaboration. To find these measures, we need information about coordination, communication, and success in the selected projects. We collected information about the projects in two ways: via semi-structured interviews and via a questionnaire.

3.3.1 Semi-structured Interviews

We interviewed 19 project managers of the front-office. According to Yin (2003), more than one source should be used in order to get valid qualitative results. To reach this redundancy, we also interviewed project managers that were involved from the offshore country. This was done in 15 cases. The rest was not allowed to talk to us, or were unreachable. Besides looking at the Indian and the Dutch side, we also planned to talk to the customers that requested the software. Unfortunately, access to these customers was often not allowed by the project managers. The consequence was that we interviewed the customers in 5 cases.
Each interview was semi-structured: all topics were well prepared, the questions were not always asked strictly in the same way (Baarda, Goede & van der Meer-Middelburg, 1996). The interview topics were the following:

1. **Project data:**
    a. A description of the deliverable
    b. A description of the customer
    c. A description of the reasons to offshore
    d. The planned start-date, planned end-date, real start-date and real end-date
    e. Total budgeted and real costs
    f. A description of the programming language
    g. Past offshore collaboration experience of the team members

2. **Distances** in the project team:
    a. Activities that were done onshore or offshore (Appendix 2)
    b. Roles that were located onshore or offshore (Appendix 3)
    c. A description of all involved stakeholders

3. **Coordination** in the project:
    a. Fixed budget and deadlines or time & materials
    b. A description of the used development method
    c. Traveling between the offshore and the onshore country
    d. Contact between back-office and the customer
    e. Onsite coordinator
    f. Availability of information to back- and front-office
    g. Reasons for success in the project
    h. Reasons for failure in the project

4. **Communication** between the project team members:
    a. A description of the communication lines in the project
    b. The availability of documents onshore and offshore
    c. Use of communication tools

5. **Performance** of the project:
    a. Successfulness with respect to scope, quality, time, and budget

The questions are described in Appendix 1. The answers were written down per topic in an Excel-sheet. In next sections, we provide a short description of the different categories in the interviews.

Category 1: Project data
Category 1 aimed at receiving an overview of the project. We asked to describe the deliverable, the customer, the reasons why the project was offshored, the programming language, the planned and realized deadlines, and the planned and realized costs. The managers were also asked if the team members had already worked together in previous projects. All answers helped to understand the project situation and the answers that followed in the other categories.

Category 2: Distances
To have an overview of the complexity of the project situation, information is needed about the amount of companies or departments that were involved in the project. In order to get an overview of the project team and their location, information is needed about the involvement of both countries in the project. This information was collected by analyzing the activities of the projects and by analyzing the roles of the team members onshore and offshore, as described in previous section.

Category 3: Coordination
Category 3 of the interview topics focused on coordination measures that were applied in the projects. The managers were asked to tell about traveling, onsite coordinators, contact between the back-office and the customer, the availability of information/documents to each country, and the development method that was used. We also asked if the project was fixed price and had a fixed date, or not. If this was not the case, the project would be based on 'time & materials', which means that people and material was paid on an hourly/daily basis. Besides these questions, we the project managers were asked to mention at least one point why the project was not a complete success. We also asked them to mention at least one point why the project was a success. This list indicates the strong and weak points in the project. Some weak points are the inverse of some positive points. Those points were connected with each other. Each strong or weak point was categorized as belonging to one of the five coordination categories that are mentioned in the research framework: standards, planning, formal communication enhancement, informal communication enhancement, and team selection. This would help to see which kind of coordination measures were used in successful projects, and which measures were the probable causes for a project failure. In the end, this list helps to determine how a project should be set up in order to be successful.

Category 4: Communication
Communication questions looked at the intended communication lines, the used communication tools and the documents that were available to different team members. Communication was mainly analyzed in the questionnaire, as becomes clear in the next section.

Category 5: Performance
The performance of the cases has been determined in the way as described in the previous section.

3.3.2 Questionnaire for social network analysis

Figure 2.10 showed that communication and knowledge exchange are a major key to success. In order to get a complete overview of communication in the project, we sent a questionnaire to all project team members. This questionnaire would show insight in the communication lines during the project, and also showed whether the team members had enough knowledge to perform their activities during the project. The questionnaire was a Microsoft Excel file that was sent to all project members of nine projects. The questionnaire can be found in Appendix 4. It consisted of three parts:

General data
The first part asked for general data that could be useful during the analysis phase. We asked the following information:

- We asked the team member's nationality.
- We asked if the team member had already been to The Netherlands or the offshore country before the project. This indicates if a team member already had experience at start of the project.
- We asked if the team member had been to the other country, during the project.
- We asked for the main and the secondary role during the project.
- We asked what the team member's professional level of expertise was at start and at the end of the project.

Communication
In order to get an overview of all communication lines in the project, we used the social network analysis that was described in the previous chapter.
We asked how often the person who filled in the questionnaire had contact with each of his or her team members. The answers were based on a six point scale with increasing contact: never (0), a couple of times (1), every month (2), every week (3), every day (4), more than once a day (5). This information would help to do a social network analysis, as described by Hanneman & Riddle (2005) and Helms & Buijsrogge (2005). By using their technique, we are able to show the communication network of each team.
We decided to ask 'how much did you on average have contact with …?' and not 'how much did you contact …?' The reason behind this was that with low response, we would still be able to draw a communication network diagram. If person A had contact with person B every month, we can state that person B also had contact with person A every month. If a couple of persons would not fill in the questionnaire, we would still be able to know how much they communicated with other team members, from their answers. This would not be possible if we used the second question formulation.
We also asked in the questionnaire which communication channels were used by the team member to have contact. The order was according to communication information richness (Carmel, 1999): desktop sharing, e-mail, chat, phone, internet phone, video conference, advanced video conference, and face to face communication. These ways of communication were chosen from the communication tools that were mentioned in the interviews. These answers tell something about the richness of the communication with each team member: if team members communicated every day over the phone for one hour, there is less information communicated than if they communicated every day face to face for one hour. This is because face to face communication is richer than communication by phone (Carmel, 1999).

Knowledge exchange
In order to get an overview of the knowledge exchange in the projects, we also asked if the team members received information from their colleagues in The Netherlands or the offshore country, or from the customer, to perform their activities. We asked these questions in the form of sentences that could be agreed on a certain level. The answers could be given on a five point Likert scale: from totally not agree, to totally agree. The answers indicate whether the project team members had enough knowledge to complete the project successfully.
We also asked if the project team member agreed that the project was a success. The answer options were the same scale as the sentences that were used at the knowledge exchange sentences. Answers would help to see if the project team members agreed with the project managers about the success of the project.

3.4 Data collection

This section describes how we collected the data. First we will tell about the interviews. Second, we mention important aspects about the questionnaire and how it was answered.

3.4.1 Interviews

The length of the interviews was about an hour for each interview. All customers were interviewed at their own location. The front-office project managers were also interviewed at their own location: for the interviews with Capgemini Outsourcing, we went to the location where Outsourcing was located, and the managers at Technology were interviewed at their own department. This was done in order to raise trust between the managers and the researchers, which helped to increase the quality and depth of the conversations.
During the interviews, a couple of the offshore project managers were for business in The Netherlands. We interviewed them at Capgemini, in private rooms. In order to speak to the other project managers, we used an advanced videoconference tool, called the "cPort" (see Figure 3.2). This is a high-bandwidth video communication system, with which it is possible to look each other into the eyes. The camera is placed behind a one-way mirroring glass. The communication partner is shown on a large display behind the mirror. This facility made it possible to look each other in the eyes and use gestures. According to Carmel (1999), high band-width videoconferencing has a high psychological interaction and therefore communication richness. Communication became almost face to face, which made it possible to ask delicate questions to the offshore project managers.
All 39 managers mentioned strong and weak point of the projects. Offshore cultures often block the communication of weak points. We experienced that due to the use of the cPort, the block was removed and the offshore managers also shared weak points.
During interviews, some project managers had difficulty with filling out the detailed activities of the worksheet in Appendix 2. Since some activities were nested, they were allowed to skip the most detailed level. These activities have the color white on the sheet (non-functional requirements, functional requirements; design, implement, test; integration, test, debug; integration, test, debug). The consequence was that we were not able to analyze at the most detailed level.
Not all project managers were able to fill out the worksheets. Seventeen front-office managers, six back-office managers and one customer filled out the activity worksheet. Seventeen front-office managers, eight back-office managers and no customer filled out the roles worksheet. Overall, we had at least one filled out activity and role worksheet for each case. When we had multiple worksheets available, we used the average of the numbers that were given for the analysis.

Figure 3.2. A picture that shows one of the cPort interviews with a project manager in India

3.4.2 Questionnaire

Nine project managers gave us the names of all the project members that contributed onshore and offshore. These names were used to prepare the questionnaire and send it to each team member by email. The team members filled out the questionnaire and sent it back. We experienced that a lot of team members missed the question 'How much did you have contact with …?' In order to get all data, we asked them to spend also some time on that part of the questionnaire. Most of the respondents did that. The response on the questionnaire can be found in Table 3.4. In this table, the total number of team members and the number of members that returned the questionnaire is shown. The table also shows the numbers of respondents in a percentage. It also shows the part of all communication lines that were available from the answers of the respondents. All employees who responded filled out all communication lines with their team members who did or did not answer the questionnaire. As we already explained in section 3.3.2, the formulation of the question helped to increase the total response rate: if person A had contact with person B, we can conclude that person B had also contact with person A. The communication between team members that did not respond was unavailable.

Project ID	Number of team members	Response	Response (%)	Communication lines filled (%)
6	5	5	100%	100%
12	12	11	92%	100%
9	5	3	60%	93%
2	23	15	65%	91%
5	11	7	64%	89%
4	10	5	50%	78%
3	9	4	44%	72%
14	96	30	31%	53%
10	41	11	25%	47%

Table 3.4. An overview of the response on the questionnaire.

3.5 Data processing and analysis decisions

After completing the interviews, we processed the data in order to be able to analyze and draw conclusions. This section shortly describes how we processed the data.

3.5.1 Interviews

We used a MP3 USB-stick to record the interviews and to be able to listen to the interviews afterwards. We used a Microsoft Excel file to write down all the answers on the questions. This helped to have a clear oversight with so many interviews and questions. The general data about the projects was used to give a description about the projects, as was given in Table 3.1.
All answers about the success of the projects were categorized as 'unsuccessful', 'successful' and 'not successful and not unsuccessful', according to the definitions in section 3.3.1. This was done for all different success aspects and for the combined overall success.
The answers on the worksheets were used to calculate the offshore percentage of roles and activities as was described in section 3.3.1. We took the average of two or three percentages when we had sheets of more than one manager in one project. On average, when two different managers had written down the amount of people who worked on a certain activity, the offshore percentages that were derived from these answers differed on average circa 14%. For the roles worksheets, this was 18%. Large differences between projects were therefore useful to draw valid conclusions from.
In order to compare all projects, we categorized the offshored percentages according to the three categories of success as we described. In this way, the average offshored percentage of activities in

successful projects can be compared with the average offshored percentage of activities in unsuccessful projects.
The reasons for success or failure were labeled and categorized according to the categories that were described. Some managers mentioned the same reasons. Those reasons received the same label.
The managers mentioned sometimes positive points that were not related to coordination, but to the results. These 'reasons' were:

- We met the cost savings.
- We delivered what was requested.
- We had good results.
- We delivered high quality.
- The results met the expectations.
- We delivered more than was requested.
- The result had reasonable quality.
- The customer was satisfied.
- We learned a lot.

We decided that those reasons were not correct, because they referred to the result in stead of the cause. "We were successful because we had good results" is a statement that is true, but that is not useful in order to look at coordination measures. By this decision, we removed 20 arguments that were not real arguments.
Some reasons that were given are vague, since they are caused by other coordination measures. An example is 'There was good interaction between both countries.' This is a cause of success, but does not answer the question: 'How should an offshore project be set up?' It is caused by for instance an informal organization. We called these type of reasons 'indirect reasons', since they are caused by other reasons that fit in one of the five coordination categories.
In total 104 useful positive arguments were given by the managers. This resulted in 78 positive coordination measures (26 measures were already mentioned by others). With regard to the negative points, we counted 116 in total. Out of these reasons, we identified 85 negative coordination measures (31 measures were already mentioned by others).
The positive and the negative points were connected to the successfulness of the projects. If a project was successful, it would be useful to look at its positive points. If a project was not successful, it would be useful to look at its negative points. We also analyzed how often each category of coordination measures was mentioned by successful and unsuccessful projects. This gave information about the focus on the coordination measures. If for instance the category of standards was mentioned a lot by successful projects but almost not by unsuccessful projects, we could conclude that those coordination measures were important to have a successful project.

3.5.2 Questionnaire

All questionnaires of a project were combined in a Microsoft Excel worksheet. The answers resulted in matrices about the communication between the team members. When two members gave an answer about each other, we took the average of those answers. The team members could give answers on a six point scale. On average, the answers of two team members differed 0.8 points. If team member A said that he had daily contact with team member B, it was possible that B said that he had weekly contact with A, or more than once a day. It almost never happened that member B indicated to have never or just a couple of times contact with member A. We therefore concluded that our method for filling out communication was valid if one of the two answers missed.
Helms and Buijsrogge (2005) stress that almost all communication lines must be available, in order to draw valid conclusions. The first five cases in Table 3.4 had almost more than 90% of the

communication lines available. We therefore decided that those cases were useful for our research. The other four cases could be used for the questions about knowledge exchange and the used communication tools, but could not be used to draw a communication network and further use for drawing conclusions.
From the matrices that were filled with the amounts of contact between all team members in a project, we made diagrams that were similar to the diagram that was described in our literature research (Figure 2.6). For this analysis we used the computer program Netdraw (Borgatti, 2002). The location of the actors and the length of the communication lines was determined by using the algorithm 'spring embedding' (Hanneman & Riddle, 2005). This algorithm was chosen because it is described in scientific literature and because it is generally available. The algorithm starts with a random graph, measures how correct the graph is according to the matrix, moves something, and measures again. If that new graph is better than the old one, it again moves another point in the graph. This type of 'iterative fitting' was done 1000 times. The result was that the people with tight connections were placed near each other in the graph. We made graphs for people that had contact at least once a month, and graphs with data about contact that took place at least once a day.
As was already described in the literature research, sub-communities can be distinguished in communication networks. We applied one of the community algorithms on the diagrams about people that had contact at least once a month. In order to do this, we used the clique-analysis (Hanneman & Riddle, 2005). People who were closely connected to each other were one clique. Some people in one clique are also member of another clique. The people whose cliques overlap with a lot of other cliques are central in a project team. To depict the cliques of the projects, a dendogram was used. It shows which cliques are found on which levels. For this analysis, we used the computer program Ucinet (Borgatti, Everett & Freeman, 2002).
In order to be able to analyze the communication networks, we defined definitions of important roles in the networks. The definitions are based on the roles that are defined by Hanneman and Riddle (2005), but adapted to an onshore/offshore situation:

- A **liaison** is someone who has daily contact with the customer.
- A **pendant** is someone who only has daily contact with only one person.
- An **onshore representative** is someone at the front-office who has daily contact with someone of the back-office.
- An **offshore representative** is someone at the back-office who has daily contact with someone of the front-office.
- An **onshore coordinator** is someone at the front-office who has daily contact with all or at lease two members of the back-office.
- An **offshore coordinator** is someone at the back-office who has daily contact with all or at least two members of the front-office.

We used these definitions to analyze successful and unsuccessful networks, and to derive rules that indicate what is positive or negative for a project team.
Before analyzing, we defined how we could detect positive or negative communication and knowledge exchange in a project team. A positive situation of communication and knowledge exchange can be recognized as short communication lines (via a few or no team members) and a lot of contact with other team members. A negative situation of communication and knowledge exchange can be recognized as having long communication lines (via a lot of team members) and having not much contact with other team members.
With these definitions we identified bottlenecks in the failed projects and best practices in the successful projects. Bottlenecks are situations where we can learn from. Best practices help to learn what improves project team communication.

3.6 Reliability and validity

Baarda, de Goede, and Teunissen (1997) state that in qualitative research, extra attention should be given to reliability, internal validity and external validity. Reliability means that the method, data collection, data processing, and the analysis are technically sound. Internal validity means that the results are a correct representation of the projects that were analyzed. External validity means that the conclusions can be applied to other (new) projects that were not researched. In the next sections, we explain how and in which way these three aspects are applied in our research.

3.6.1 Reliability

In order to make our research method applicable to the situation of offshore CSD projects, we started our research with introductory interviews. Two conversations with Indian project managers and two conversations with Dutch project managers helped to get an overview of the CSD field. This helped us to prepare the interviews, the worksheets, and the questionnaire in a way that is understandable to all project managers and project members in the CSD teams.
We chose to interview project managers, who had a complete overview of the projects. They had information with regard to the software that was produced, but also information about the project status, the results, and the customer. Since we also chose completed projects, they were willing to give all information about the projects, even if it was not positive.
Before interviewing the project managers, we reviewed the questions with several consultants at Capgemini and several researchers at Utrecht University. This helped to optimize the questions. First, we did the interviews with front-office managers at Capgemini. This learned how to ask the questions and which questions required extra attention, before we started the interviews with the customers and the back-office managers. Due to this, we were able to understand the project situation and we were able to understand their situation better. This increased the quality of the interviews with the project manager at the customer and the back-office site.
We used semi-structured interviews, in order to get all information about the projects. Unstructured interviews of nineteen projects are difficult to compare and draw valid conclusions. Therefore we chose to put structure in the interviews.
During the interviews, we wrote the answers down on paper. We also recorded the interviews. This helped to be able to re-listen and re-read the answers, when something was unclear. We directly wrote down the answers on the questions after the interview. This reduced the possibility that we forgot some important information.
In order to get reliable answers in the interviews with the project managers who were offshore, we interviewed most of them face to face or via the cPort. Only one offshore project manager who was located in Romania was interviewed by telephone, since she did not have access to a cPort. The use of the cPort helped to create a feeling of trust between us and the offshore project managers. This increased the honesty in answers and therefore the reliability of the answers.
The onshore project managers were interviewed at their own company. Mostly, we interviewed them at their own departments. This helped to make them feel comfortable and get more honest and reliable answers.
The questionnaire was reviewed by two Dutch and two Indian employees of Capgemini. They gave feedback about concepts and questions that were not clear in the questionnaire. They also gave advice which questions were useful to add to the list. This review was very valuable, since it improved the questionnaire.
The research was set up to be qualitative. Therefore, we cannot tell about significance of the results with regard to the sample size. In order to get reliable and valid results, we therefore used multiple ways of triangulation. This will be explained in the next section.

3.6.2 Internal validity

The major road to internal validity was to use triangulation in two different ways. Triangulation means that more than way is used to get information about the same subject (Baarda et al., 1997). We used two types of triangulation:

Methodological triangulation
As was described in this chapter, we use both interviews and questionnaires to get information about the offshore CSD projects. Both ways indicate how successful the projects were, what the strong and the weak points were, and how team members communicated in the projects. Conclusions from one method should match with the conclusions of the other method, in order to have valid results. If a manager in an interview states one thing, but the questionnaire shows the opposite, it is not a valid conclusion.

Data triangulation
Besides methodological triangulation, we also applied data triangulation. At most projects, we interviewed the project mangers from the onshore and the offshore country. They have different perspectives on the projects. This was not the case in five projects. The main reason was that the organizations did not give access to the back-office. In one project we interviewed an extra person from the front-office, but this was not the case for the remaining four projects. This reduces the validity of the results of those four cases, since the information was collected from one person. We also intended to interview the customers of the projects. Unfortunately, this was only possible in five cases. This reduces the validity of the fourteen other cases, too.
Besides the interviews, data triangulation was also applied to the questionnaire. This was done by using the average of the amount of communication between two people when we had both people answering the questionnaire. By this, we had an extra check on 73% of all answers.
In order to check the validity of the answers of the people that were interviewed, it could also be useful to check the absolute data that was available about all projects as an extra source. Unfortunately, this was not possible, due to the following reasons: As different organizations (Capgemini and non-Capgemini) were involved to get high validity, this led to a high diversity of systems that all registered different data. Data from offshore countries was not always available. In some projects, the offshore teams worked more hours than were mentioned in the systems. The data in The Netherlands was sometimes unreliable, too: Some projects did not register the time that was spent. Some team members did not register the amount of hours that were spent at the end of a project. Some projects consisted of multiple projects with a lot of changes, which resulted in a high amount of data that needed to be cumulated in order to get the total statistics of the project. All of these reasons made it impossible to get comparable quantitative data about the projects in a few months. The described methodological and data triangulation were therefore the only ways to reach internal validity. These ways of triangulation helped to bypass biases and incorrect answers, and get a complete and valid overview of the projects.

3.6.3 External validity

Our intention is to draw conclusions about how offshore CSD projects should be performed in the future. Therefore, it is necessary that the conclusions can also be applied to other projects. We selected a large amount of cases, in order to have results that are applicable to all offshore CSD projects. We researched project situations that really happened, which increases the external validity of the project.
The projects came from different types of industries. Therefore, we state that the results are applicable to different types of industries. We also analyzed projects in different settings: projects at two

different business units of Capgemini and projects that were done without Capgemini. Therefore, we state that the results will also be applicable to other organizations than Capgemini. We also tried to select projects with different kinds of size, amount of involved stakeholders, costs, program languages, and type of contracts. This all contributes to external validity, since we can learn different things from different project settings.
We also tried to include other countries than India. Malaysia and Romania are included, but we did not manage to find other countries. Since most CSD projects that are offshored are located in India (Gartner, 2005), the results are applicable to most CSD projects. Nevertheless, it should be kept in mind that the final results are not based on a representative set of offshore countries.
We used literature to create general roles and activities that can be applied to all CSD projects. This also helped to draw conclusions for activities and roles in CSD projects that use methods that are different from the methods that were used in the projects that we researched (mostly waterfall and RUP).

4 RESULTS & ANALYSIS

This chapter presents the results from the interviews and the questionnaire. It also analyzes the results, which leads to observations. In the next chapter we will draw conclusions based on these observations. As we stated in the previous chapters, we want to be able to see which coordination measures lead to successful offshore CSD projects. This research focuses on distances, communication, coordination, and performance, as key attributes for a successful project.
First, the interviews provided information about the success of the projects that were analyzed. Furthermore, they provided general data about the projects and information about the distances between the team members in project activities. The interviews further more provided information about coordination measures that were applied to those projects. Finally, the questionnaires gave information about the communication in the development teams.
All project information will be related to the performance of the projects, since the performance indicates if the project was successful. Therefore, the next section shows this performance. The proceeding sections show the results with regard to distances, coordination, and communication.

4.1 Performance

Table 4.1 shows the successfulness of all projects in terms of scope, quality, time and costs. The IDs of the projects are used in the whole thesis. According to the definition of successful projects that was provided in the previous chapters, all unsuccessful aspects that score below 3 are marked black. The successful projects that score equal to or above 4 are marked white. In each success category, there are projects that are not successful or unsuccessful. We marked them gray.

ID	Type of company	Scope S	Quality Q	Time T	Costs C	Overall success (S+Q+T+C)/4
1	Insurance	4,0	3,5	1,5	2,5	**2,9**
2	Bank	2,0	3,0	2,5	2,5	**2,5**
3	Professional services	5,0	4,5	4,5	3,5	**4,4**
4	Bank	3,5	5,0	4,5	3,5	**4,1**
5	Insurance	4,0	4,0	2,5	3,5	**3,5**
6	Retail	4,0	3,3	3,3	3,3	**3,5**
7	Bank	4,0	4,0	4,0	3,5	**3,9**
8	Product software company	2,7	2,7	3,3	4,3	**3,3**
9	Government	5,0	4,0	2,5	4,5	**4,0**
10	Bank	5,0	4,0	2,0	3,0	**3,5**
11	Government	4,0	3,5	3,0	3,0	**3,4**
12	Industry	4,0	4,5	4,0	3,0	**3,9**
13	Government	4,3	3,0	4,7	4,0	**4,0**
14	Telecommunications	3,5	3,5	4,0	5,0	**4,0**
15	Professional services	4,0	3,0	2,0	2,0	**2,8**
16	Product software company	5,0	5,0	4,0	4,0	**4,5**
17	Insurance	5,0	3,0	2,0	2,0	**3,0**
18	Product software company	4,5	5,0	4,5	5,0	**4,8**
19	Product software company	5,0	4,0	5,0	3,0	**4,3**

Legend:	
Cell-color black	Unsuccessful project
Cell-color gray	Neither successful nor unsuccessful project
Cell-color white	Successful project

Table 4.1. Success of the projects that were researched

To get a clearer overview of the amount of successful projects in each category, we depicted the amounts as percentages in Figure 4.1. From this picture, we make the following observations:

Observations

- A lot of offshore projects were successful with regard to scope.
- There were not many below-quality projects.
- Many projects were not successful in the areas of time and costs.

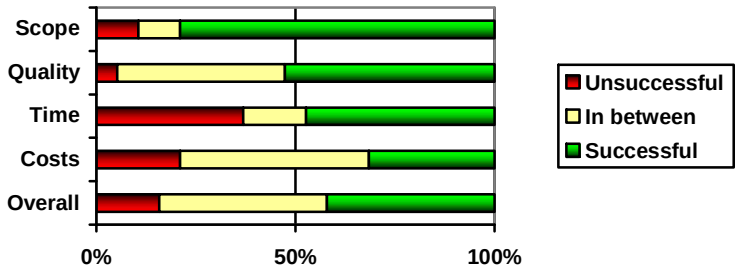


Figure 4.1. The amount of success on each aspect of success (N = 19)

4.2 Distances

This section provides an overview of the distances in the projects. First, general data about the projects and the project teams is described and analyzed. Second, results about activities and roles that were offshored are described.

4.2.1 General data

Table 4.2 gives an overview of the general data about the projects that were researched. As was already mentioned in the previous chapter, we chose a set that varied in team size, organizational complexity, project duration and costs. The table also shows if the team members already worked together with the client and each other in previous projects.

ID	Team size	Organizational complexity	Project duration	Total costs	Previous cooperation?
1	L	L	C (TTT)	€€€	No
2	M	M	C (TT)	€€€	No
3	S	S	T	€€	Yes
4	M	S	T	Unknown	No
5	S	L	TT	€€	Yes
6	S	M	T	€	Yes
7	XL	L	TTT	€€€€	No
8	M	S	TT	€€€	No
9	S	M	T	€	Yes
10	L	L	TTT	€€€	No
11	XL	L	TTT	€€€€	No
12	S	S	T	€	Yes
13	XL	M	TTT	€€€€	No
14	XL	M	T	€€	Yes
15	L	M	TTT	€€€	No
16	S	S	TTT	Unknown	Yes
17	M	S	TT	€€€	No
18	M	S	T	€€	Yes
19	XL	S	TT	€€€€	Yes

Legend:			
Team size (front office and back office):		**Organizational complexity:**	
XL:	Very large: more than 75 people	S:	2 or 3 organizations involved
L:	Large: between 25 and 75 people	M:	4 or 5 organizations involved
M:	Medium: between 10 and 25 people	L:	6 or more organizations involved
S:	Small: less than 10 people		
Costs:		**Project duration:**	
€:	Less than € 100.000	C(X):	Cancelled (duration)
€€:	Between € 100.000 and € 1.000.000	T:	Shorter than 6 months
€€€:	Between € 1.000.000 and € 5.000.000	TT:	Between 6 and 12 months
€€€€:	More than € 5.000.000	TTT:	Longer than one year

Table 4.2. Overview of the projects

The successfulness of the projects can be combined with the project data in Table 4.2. We did this for the overall success. This is shown in
Figure 4.2. From these graphs we make the following observations:

Observations

- The majority of successful project were completed within half a year.
- Most successful projects had not many organizations involved in the projects.
- In most successful cases the team members had experience with each other and the client.
- In most unsuccessful cases the team members had no experience with each other and the client.
- Many successful projects consisted of team members that had already worked together in previous projects. Some of these successful projects were executed at product software companies that collaborated with a back-office for about 10 years. Most team members knew each other and knew the customer.

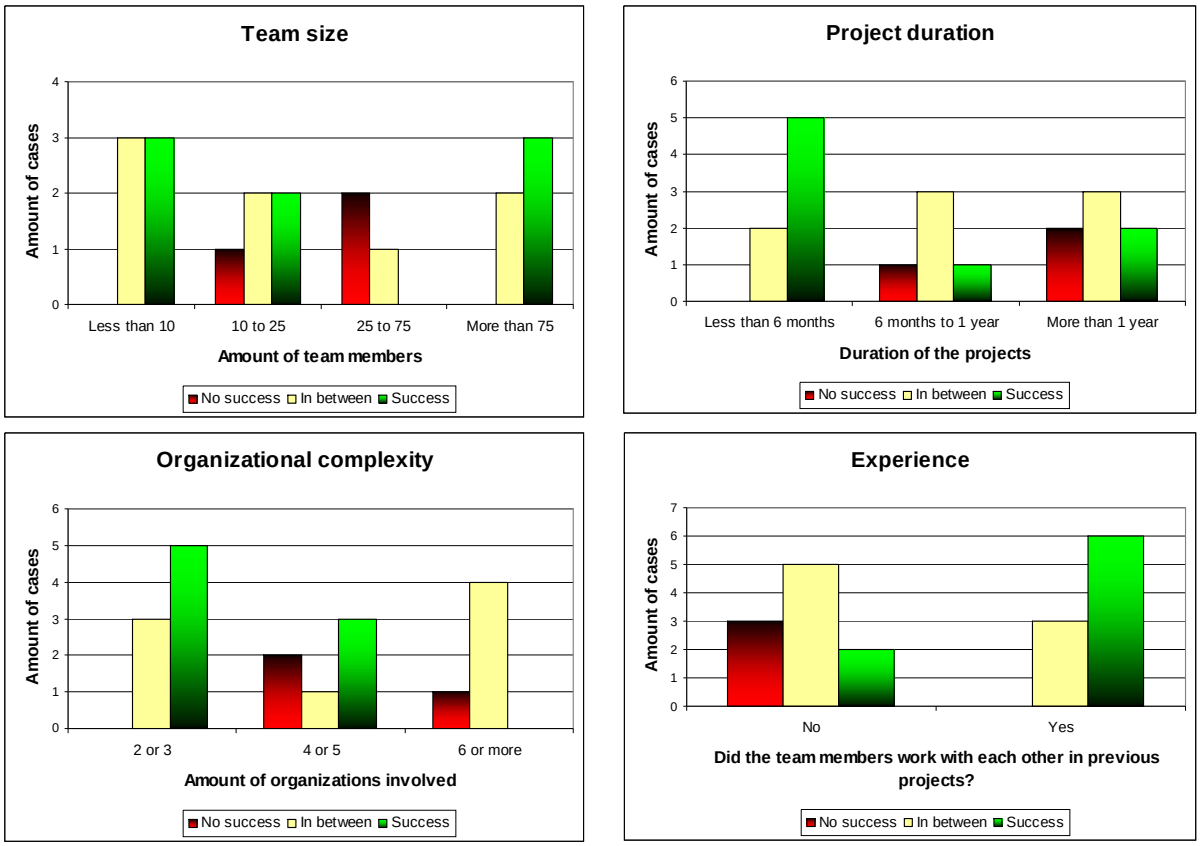


Figure 4.2. The overall success of the projects combined with project characteristics

4.2.2 Distances in activities

The offshore percentages of each project were calculated and related to the success of the projects. The result for overall success is shown in Figure 4.3. In the previous chapter, we said that the calculated offshore percentages of different managers in one project differed circa 14%. Therefore, we only made observations if the differences were larger than this amount. From the figure, we make the following observations:

Observations

- The most common offshored activities were the detailed design, the unit build, and the system integration.
- The main peek in the offshore activities was at the detailed design in successful projects, and at the unit build in unsuccessful projects. The main part of offshoring started earlier in successful than in unsuccessful projects.
- Successful projects offshored a large part of the detailed design, unsuccessful projects kept this activity mainly onshore.

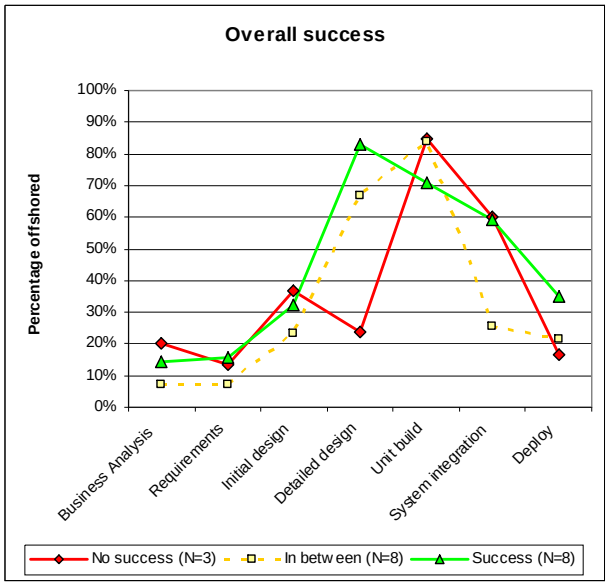


Figure 4.3. An overview of the offshored activities in successful and unsuccessful projects

The same graph can also be created for each aspect of success, as was defined in Table 4.1. The offshore percentages were combined for each success category, and depicted in the same way. This is shown in Figure 4.4. From these figures, we observe the following:

Observations

- Projects that were successful on scope, offshored less of each activity than unsuccessful projects.
- Projects that were successful on quality offshored a part of the business analysis and kept some parts of the detailed design and the unit build onshore.
- More than 30% of the unit build was done onshore, in successful quality-projects. Less than 10% was done in not successful projects.
- The detailed design was less offshored in successful projects with regard to scope and quality, but more offshored in successful projects with regard to time and costs.
- The projects that were successful in the area of costs, offshored almost all detailed design.
- The projects that were successful with regard to costs did a large part of the system integration and deployment & maintenance offshore.

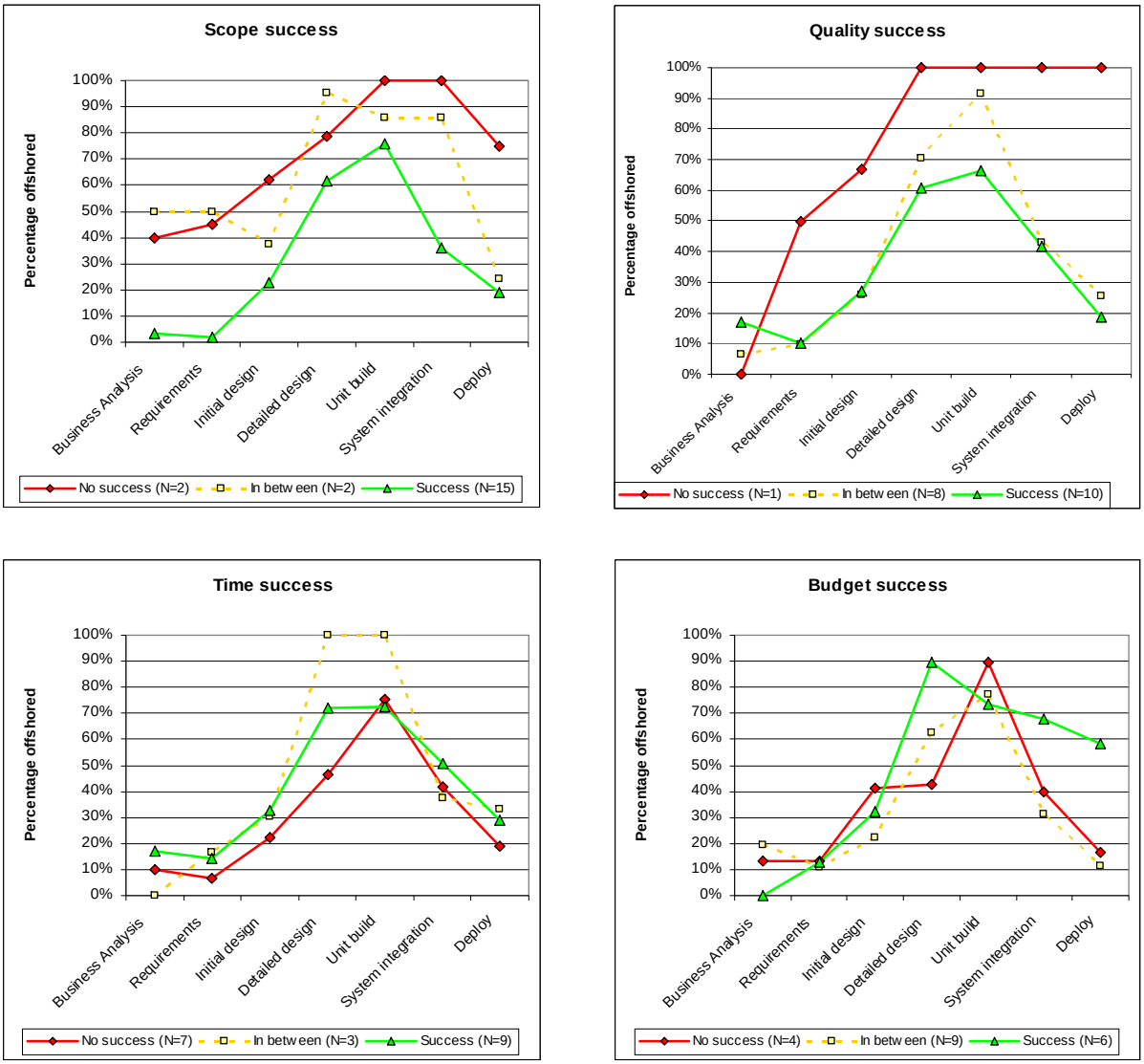


Figure 4.4. The offshored activities in projects that were successful on different aspects of success.

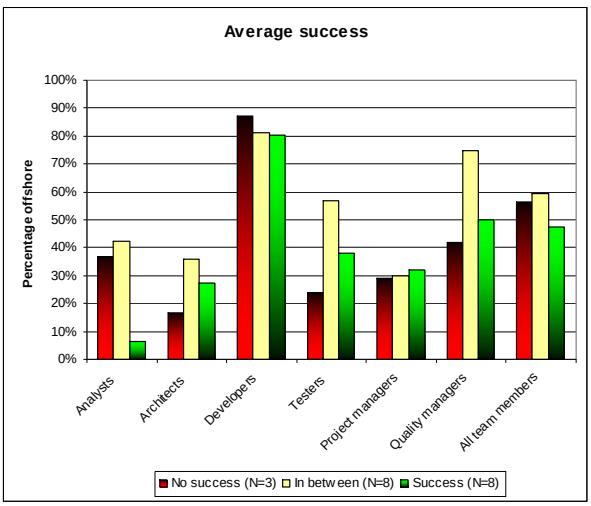


Figure 4.5. An overview of the offshored team roles.

4.2.3 Distances in roles

The offshore percentages that were used for the activities in each CSD project, were also calculated for the team roles that were available in each project. We added the analyst leads to the analysts, because not all project managers filled out how many analysts were leads. We did the same for architects, developers, and testers. Not all project managers knew if there was a configuration manager or a facilitator. Therefore, we decided to leave those roles out of the results.
An overview of the roles that were offshore can be found in Figure 4.5. The previous chapter mentioned that the calculated offshore percentages of two managers in one project differed on average 18%. Therefore, we only look at the large differences. We observe the following:

Observations

- The role of the developer was mostly offshored, both in successful and unsuccessful projects.
- Projects that were successful had almost all of their analysts onshore, while not successful projects had offshored a large part of them.
- The projects that were neither successful nor unsuccessful offshored more roles than the other projects.

When the previously described figure is split up into different success aspects, we get Figure 4.6. From this figure, we make the following observations:

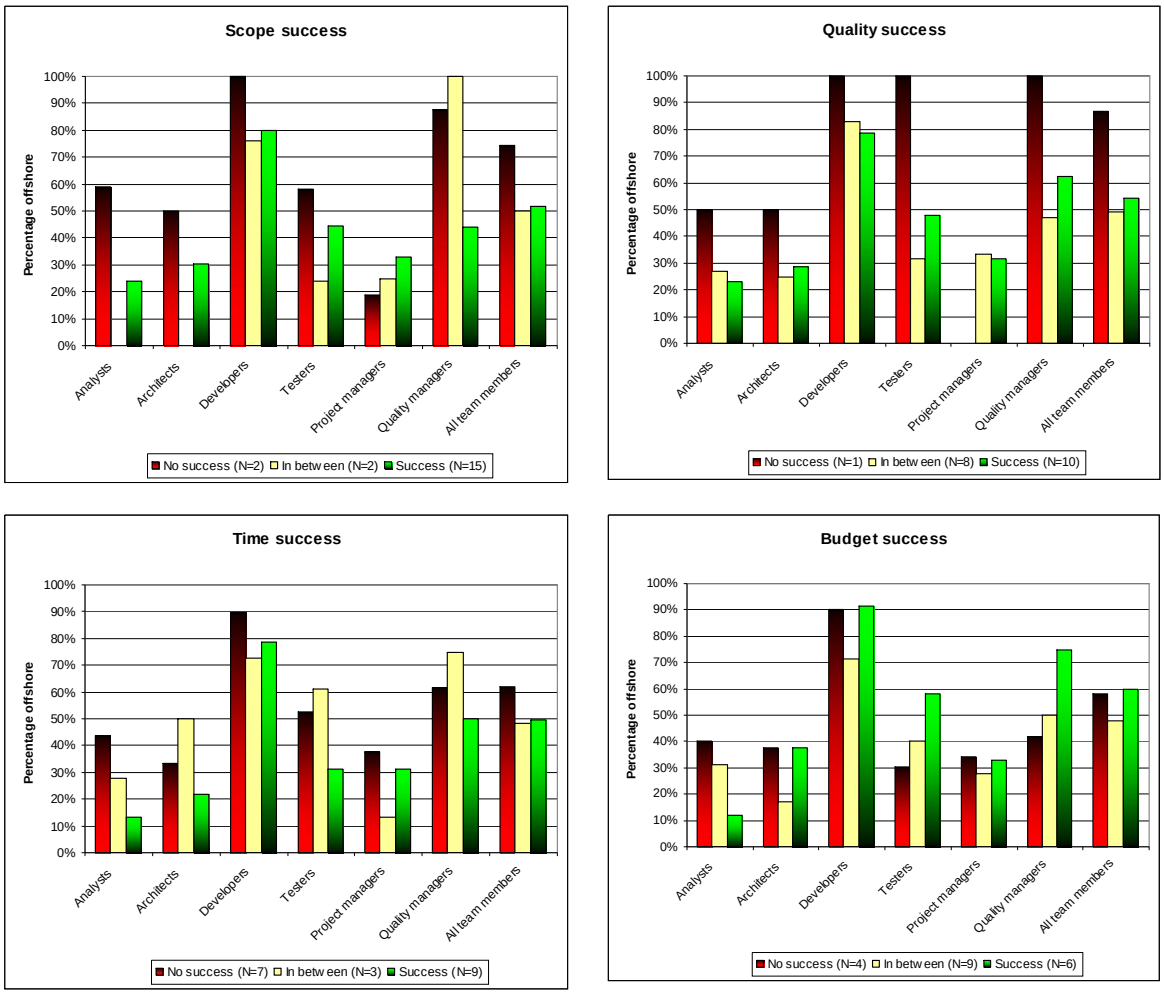


Figure 4.6. The offshored roles in projects that were successful on different aspects of success.

Observations:

- The successful projects with regard to time had for every role more team members onshore than the unsuccessful projects.
- Almost all roles in the successful projects on scope, quality, and time, had less team members offshore than the unsuccessful projects. This is not the case for successes with regard to costs.
- Developers were completely offshored in unsuccessful cases in the area of scope and quality. In successful cases, some developers were still located onshore.

4.3 Coordination

Summaries of the answers on the questions with regard to some coordination measures are given in Table 4.3. This table had ordered the projects according to the overall success of the projects.

ID	Type of budget[i]	Development method	# of docs unavailable to BO	# of docs unavailable to FO	Travel from BO to FO	Travel from FO to BO	BO/ customer contact?	Onsite coor-dinator?
2	F	Adapted RUP	0	0	21%	25%	No	No
15	T	No specific method	0	0	13%	5%	Yes	Yes
1	T	RUP-Waterfall combi	3	1	0%	10%	Yes	No
17	F	Adapted DSDM	1	1	7%	13%	Yes	No
8	T	RUP-Waterfall combi	0	1	15%	100%	Yes	Yes
11	T	Adapted RUP	2	0	10%	1%	Yes	Yes
5	TF	Waterfall	3	0	67%	25%	No	No
6	TF	Adapted waterfall	0	1	100%	33%	Yes	No
10	T	Adapted RUP	0	0	11%	15%	No	No
7	F	Adapted RUP	2	0	10%	53%	Yes	Yes
12	TF	No specific method	0	2	0%	20%	Yes	No
9	TF	No specific method	1	1	0%	100%	No	No
13	T	Adapted RUP	2	0	7%	2%	Yes	Yes
14	TF	Adapted waterfall	0	0	54%	9%	Yes	Yes
4	F	RUP-Waterfall combi	0	1	11%	50%	Yes	No
19	TF	Adapted waterfall	0	0	13%	4%	Yes	Yes
3	TF	Waterfall	1	1	67%	17%	No	No
16	TF	Adapted waterfall	0	0	17%	33%	Yes	Yes
18	TF	No specific method	1	2	57%	33%	No	Yes

Legend:	
T	Time & Materials
F	Fixed price & date
BO	Back-office
FO	Front-office
# of docs	One unit is one category of documents, such as the requirements
%	Percentage of all team members in one country (onshore or offshore) that traveled to the other country (onshore or offshore) during a project
Cell-color black	Unsuccessful project
Cell-color gray	Neither successful nor unsuccessful project
Cell-color white	Successful project

Table 4.3. An overview of coordination in the projects.

From Table 4.3, we make the following observations:

Observations:

- More than half of the projects had all documents available to the back-office and the front-office. The project managers on the onshore site told that the test results of offshore tests were unavailable to them. The offshore project managers missed the business analysis, the requirements, the functional designs, and formal documents or manuals that often were in Dutch.
- 17 of the 19 projects used adapted versions of development methods. Four of them even developed their own method.
- The development method RUP was applied in two of the three unsuccessful cases and only two times in all eight successful cases. The adaptation of RUP was possibly harder than the adaptation of the waterfall method.
- In most successful cases, the budget was based on a combination of time & material and fixed price and date. From the interviews, we learned that this was sometimes done according to the 'chinese roof' construction: paid per hour with a limit. In other cases this was done by splitting the project into phases with each a different budget construction: the business & analysis and design phase was done on basis of time & materials, while the build & test phase was fixed price and date.
- In most cases, the back-office had contact with the customer.
- In most cases, an onsite coordinator was present.
- In almost all cases, traveling between the front- and the back-office took place.

These observations were derived from closed questions about coordination aspects. We also asked all project managers in an open question because of what coordination measures their projects were successful or unsuccessful, as was described in the previous chapter.
The complete list of reasons is available in Appendix 5. The main reasons of failure that were given by the unsuccessful projects are important to pay attention to. They learn how you should not set up a project. The main reasons for success that were given by the most successful projects are also important to learn from. The reasons for success of the four most successful projects and the reasons for failure of the three unsuccessful projects are depicted in Table 4.4.
In both the table and the appendix, the reasons are categorized according to the coordination categories that were mentioned in the previous chapters: standards, planning, formal communication adjustment, informal communication adjustment, and team selection. The indirect reasons are also mentioned, in order to give a complete overview of all categories.

From Table 4.4 we make the following observations:

Observations

- Standards were not the main reason for success or failure in the offshore projects.
- Unsuccessful projects faced unclear requirements and a lot of changes, while successful projects had clear deadlines and a clear planning.
- A cause for failure that was mentioned multiple times was that the back-office was involved in a project too early or too late.
- In case of the unsuccessful projects, a system was applied in which all creative work was executed onshore, while the easy industrial labor was done offshore: everything was for example thrown over the wall, there was Dutch documentation, and traveling was not possible.
- In successful projects, functionalities of the software and responsibilities of the team members were clearly specified.

- In unsuccessful projects, information was kept at the front-office and not communicated to the back-office. In the successful projects this was not the case: collaboration tools and an informal organization improved sharing knowledge about the requirements and their backgrounds.
- Incapable team members and project managers led to failure of projects, while flexible and hard working team members led to success.
- Due to the coordination measures in successful teams, there was a lot of communication and sharing of knowledge.
- Due to the coordination measures in unsuccessful teams, the mindset of both the back- and the front-office were negatively influenced.

Coordination category:	Reasons for success mentioned by the 4 most successful projects:	Reasons for failure mentioned by the 3 unsuccessful projects:
Standards	None	None
Planning	• There were clear deadlines. • There was a good planning.	• There was a fixed price and a fixed date. • Due to a lot of uncertainties, planning was impossible. • The customer did not know all his requirements. • There were a lot of changes in the requirements. • Extra changes that the customer requested cost too much time. • The back-office was involved in the project very late. • The back-office started too early.
Formal mutual adjustment	• The functionalities were clearly specified. • The responsibilities of the back-office were predefined. • The back-office worked with good documentation. • Management of both countries met a couple of times.	• The use cases were in Dutch and had to be translated. • All requirements were 'thrown over the wall'. • The kickoff was onsite and not offshore. • There were visa problems which prevented traveling.
Informal mutual adjustment	• The organization was informal. • The background of the requirements was made clear to the back-office. • The team members used a collaboration tool. • The team members used an instant messaging tool.	• The front-office did not give the information that was needed. • There was not one common repository.
Team selection	• The team members were very flexible. • The team members worked very hard.	• The project manager in the front-office had not enough experience at start. • The size of the team was too large. • Team members did not give priority to the project. • People reacted late on mails and document requests.
Indirect	• The communication was straightforward. • There was good cooperation. • The team members got a positive image of the back-office country. • There was good interaction between both countries. • The relationships were based on trust. • There was a proactive back-office. • The team members shared knowledge in the right way.	• There was a 'we/they' feeling. • Team members had a negative mindset about the offshore country. • The back-office had no proactive attitude. • The front-office was an obstacle between the back-office and the customer. • There was not enough (tele)communication.

Table 4.4. A short overview of the mentioned reasons of success or failure

In order to see which coordination categories are most effective in successful projects, we analyzed all reasons for success and failure that were given by all project managers. We looked which categories of coordination measures were mentioned as reason for success or failure. Mostly, an argument for failure is the opposite of an argument of success. If for instance planning is mentioned as reason for failure, this would mean that there was no proper planning.

All reasons that were based on coordination measures were categorized, and counted. The result was compared to the success of the projects, as can be seen in Figure 4.7. The reasons for success that were given by the projects that had no success, were probably not good enough to cause a project success. In the area of failure, the categories that were mentioned by the failed projects are important to analyze, since these reasons did probably cause the failure of the projects.

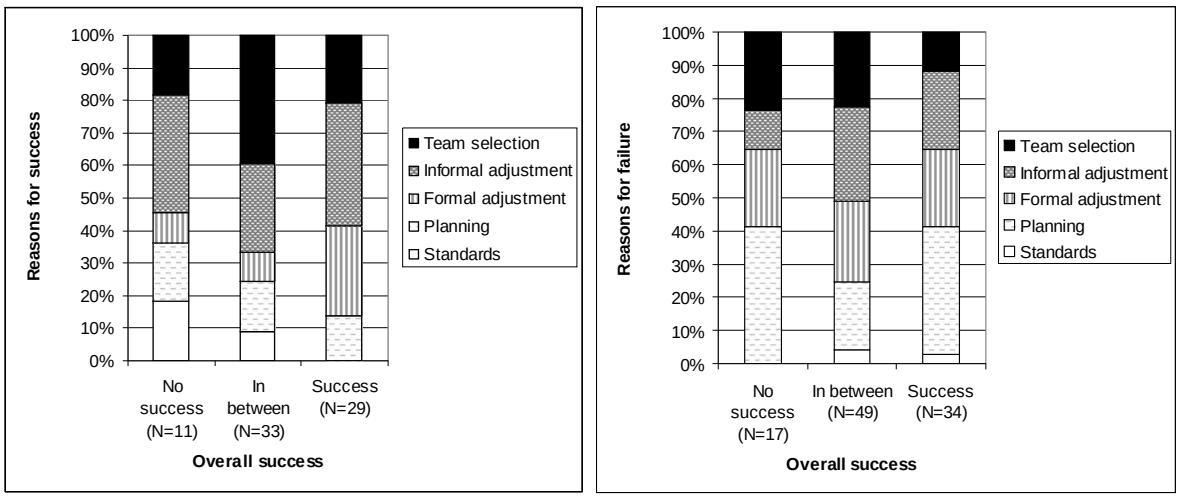


Figure 4.7. The amount of coordination measures that were mentioned in each category

We also split the success in the different aspects. An overview of those results can be found in Appendix 6. From the figure and the appendix, we make the following observations:

Observations

- Standards were not mentioned as a major cause of success by the successful cases.
- Formal mutual adjustment was mentioned as a major cause of success by successful projects, but not by unsuccessful projects.
- Almost 40% of all reasons for success consist of informal mutual adjustment.
- Improper planning is mentioned as the major cause of failure of CSD projects by both the successful and the unsuccessful projects. Remarkably, planning is not often mentioned as a major cause of success.

4.4 Communication

The questionnaire was used to collect information about the communication and the knowledge exchange in nine projects and to draw communication networks that can be analyzed in order to find causes for success or failure. Five cases had enough response to draw the networks. Before entering that analysis, we analyze the answers on the general questions of the questionnaires of all nine cases that were described in the previous chapter.

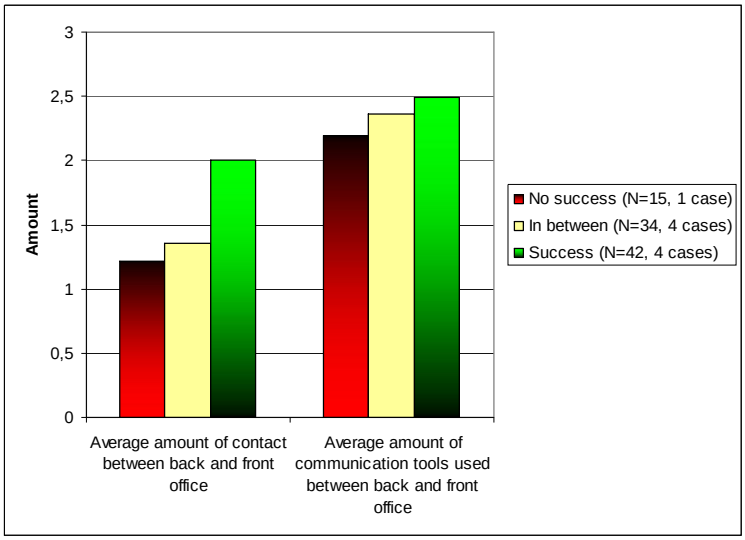
4.4.1 Interaction between back- and front-office

All team members who answered the questionnaires gave information about how often they communicated with other team members. They also answered the question about which communication tools they used.
We calculated the average of communication between offshore and onshore team members in each project. This average indicates the amount of knowledge transfer that took place between both

locations. We also calculated the average amount of communication methods that were used when team members from the front-office and the back-office had contact.
The calculated averages were combined with the overall success of those cases. The results are shown in Figure 4.8. The y-ax depicts the scores that were mentioned for these questions in the previous chapter. The success is determined for these projects in the same way as was done in all other results. From this figure, we make the following observations:

Observations

- In successful projects, all back-office and front-office team members had on average each month (= score 2) contact with each other. In unsuccessful projects, they communicated only a couple of times (= score 1) during the whole project.
- If back-office and front-office team members had contact, they communicated in successful teams on average via more communication ways than in unsuccessful teams.



Legend:	
In case of the average amount of contact: 0 = Never 1 = A couple of times 2 = Every month 3 = Every week	In case of the average amount of communication tools: 0 = 0 1 = 1 2 = 2 3 = 3

Figure 4.8. Average amount of contact and communication ways used by front-office and back-office

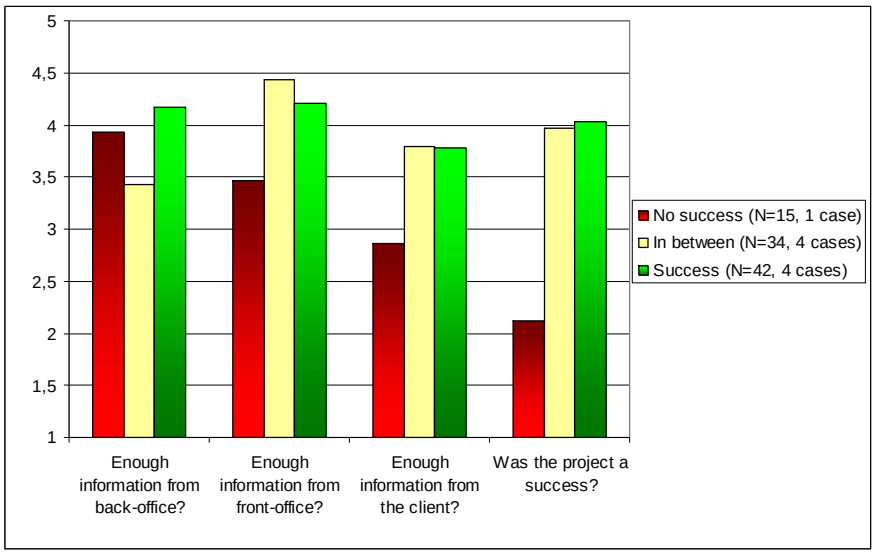
4.4.2 Information exchange

The team members also had to react on sentences about information transfer. They had to tell if they had enough information from the back-office, the front-office, or the customer. We combined all the answers of a case and related those to the successfulness of that case. The results are visible in Figure 4.9. In the same way as project success was defined in the previous chapter, answers above 4 mean in this figure that the team members agreed. A score below 3 means that at least a couple of team members disagreed. From the figure, we make the following observations:

Observations

- The team members of the not successful team agreed that the project was not successful. The team members of the successful team agreed that those projects were successful.

- The team members of the unsuccessful project indicated that they did not get enough information from the customer. They also mentioned that they did not get a lot of information from the front-office.
- The team members of the successful projects also indicated that they did not get enough information from the customer.



Legend:
1 = Totally not agree
2 = Not agree
3 = Somewhere in between
4 = Agree
5 = Totally agree

Figure 4.9. Information transfer related to the overall success

4.4.3 Social Network Analysis

The network diagrams and dendograms of the five projects with a high response rate that were described in previous chapter can be found in appendix 7 (in order to keep this chapter readable, the large amount of figures is not placed in this chapter). The roles that were defined in the previous chapter were used in the identification of bottlenecks in the failed projects and best practices in the successful projects. Bottlenecks are situations where we can learn from. Best practices help to learn what situations improve project team communication.

Unsuccessful case 2
Case 2 was unsuccessful on all aspects of success. It was a large project with a lot of team members. This could have impacted the project negatively, as we observe in the diagrams:

Positive:
- The customer had daily contact with project mangers onshore and offshore.
- The customer had daily contact with the onshore representatives.
- There was an offshore coordinator: Developer 2.

Negative:
- Offshore Developers 6 and 7 had no daily contact with the offshore representatives (Architect 3, Developer 2, Developer 3, and Developer 4). This probably caused long communication lines.

- Offshore team member Support 2 is a pendant, daily communicating with Developer 9, who was mentioned before as someone connected to a long line.
- A lot of offshore team members communicated daily with Project manager 5. He or she was not communicating daily with the onshore team members, even not with the onshore project manger.
- Looking at the dendogram, we see two clearly separated teams: onshore and offshore. There was not one combined team.

Not successful nor successful case 5
Case 5 was successful in the areas of scope and quality, but not in the areas of time and costs. The reasons become clear from the negative points. We observe the following:

Positive:
- The onshore core team (Analyst 2 and Project manager 2, as can be seen in the dendogram) and the offshore core team (Developer 2 and Developer 3) had a lot of contact within the teams.

Negative:
- There was no daily contact between the front-office and the back-office.
- Developer 1 and Support 2 in the front-office communicated not much with all team members.
- Project manager 1 was a pendant: he or she communicated mainly with analyst 2.
- There were no onshore/offshore representatives/coordinators.
- Looking at the dendogram, we see that the main team of the project was in The Netherlands. The offshore team members were not really part of that team. Support 2 and Project manager 1 were also not part of it.

Not successful nor unsuccessful case 6
Case 6 was not completely successful on quality, time, and costs. The scope was successful. In the diagrams, we see the following:

Positive:
- The project manager was a liaison, an onshore representative, and an onshore coordinator. He had daily contact with everyone.
- There were short communication lines: a maximum of two lines was needed to get knowledge from one person to another team member.

Negative:
- The project manager did a lot of communication between both countries. This makes the project vulnerable if he/she leaves the project halfway. Although it is out of scope, it is interesting to mention that this did happen after the project: the project manager left, the communication with the back-office became difficult, and the offshore team member was not a team member of future projects for this customer.
- The analyst is a pendant: he or she communicates daily with the project manager. Contact with the developers was only once a month.
- Looking at the dendogram, we see that the offshore team member was not part of the main team, which was located in The Netherlands.

Successful case 9
Case 9 was successful in the areas of scope, quality, and costs, but unsuccessful with regard to time. In the network diagrams and the dendogram we observe the following situations:

Positive:
- The liaison (Architect) is also the onshore representative.
- All but one team members had daily contact with the offshore representative.

- The onshore architect was part of the offshore main team, which can be seen in the dendogram.

Negative:
- One developer (Developer 2) had no daily contact with the rest of the team.

Successful case 12
Although case 12 had a lot of team members, it was successful in the areas of scope, quality, and time. The project was not successful in the area of costs. In the network diagrams and the dendogram we observe the following situations:

Positive:
- There was an onshore coordinator (Developer 1).
- All offshore developers had daily contact with the onshore coordinator.
- All offshore team members had daily contact with a couple of offshore representatives.
- There were multiple onshore and offshore representatives.
- All but one onshore team member had daily contact with onshore representatives.
- In the dendogram we see that both onshore and offshore project team members were closely connected in one team.

Negative:
- The customer had no direct contact with the onshore representatives.
- Developer 1 did a lot of communication between both countries. This makes the project vulnerable if he/she leaves the project halfway.
- Analyst 2 had no daily contact with onshore representatives. This brings long communication lines.
- There was no contact between the project manager in India and in The Netherlands. This prevents efficient people planning and project set up.

General observations
Based on all mentioned sub-observations, we define the following main observations/rules:

Observations
- In a successful case, the offshore and onshore team members are all part of one team. In unsuccessful projects, we see that there are two teams or that the group of offshore team members is not well integrated with the group of onshore team members.
- In a successful case, the team member who communicated with the customer also communicated with the back-office.
- In a successful case, the project members are representatives or have daily contact with representatives. In an unsuccessful case, there are a couple of team members who don't.
- In a successful case, there is an onshore or offshore coordinator.
- In unsuccessful cases, project managers do not communicate daily with each other.
- In an unsuccessful case, there is no daily contact between the front-office and the back-office.
- In an unsuccessful case, there were people who did not communicate a lot.
- A project that has one coordinator onshore or offshore is vulnerable for situations where that person would leave the project.

Bottlenecks & Best practices
From these observations, we define the following bottlenecks in projects, using the definitions that were defined:

Bottleneck 1: There are no onshore or offshore representatives.
Bottleneck 2: There is no onsite or offshore coordinator.
Bottleneck 3: There is no liaison.
Bottleneck 4: The liaison is not an onshore representative.
Bottleneck 5: Someone at the back-office has no daily contact with an offshore representative.
Bottleneck 6: Someone at the front-office has no daily contact with an onshore representative.
Bottleneck 7: A team member is a pendant.
Bottleneck 8: A team member has no daily contact with other team members.
Bottleneck 9: There is only one onshore or offshore representative.

These bottlenecks can be used to detect problematic situations in future network diagrams. The bottlenecks can also be rephrased as best practices that indicate how a successful project team should communicate:

Best practice 1: There are onshore or offshore representatives.
Best practice 2: There is an onsite or offshore coordinator.
Best practice 3: There is a liaison.
Best practice 4: The liaison is an onshore representative.
Best practice 5: Someone at the back-office has daily contact with an offshore representative.
Best practice 6: Someone at the front-office has daily contact with an onshore representative.
Best practice 7: No team member is a pendant.
Best practice 8: A team member has daily contact with some other team members.
Best practice 9: There are more than one onshore or offshore representatives.

5 CONCLUSIONS & DISCUSSION

In the introduction of this thesis, the following question was raised: *"How should a distributed offshore software development project be coordinated, in order to increase the chance on a successful project?"* This question was split up into five sub-questions, which were answered in hypotheses after having analyzed scientific literature. This chapter answers the questions, based on the research and results that were described in the previous two chapters. We draw conclusions and discuss if the results are valid. At the end of this chapter, opportunities for further research are described.

5.1 Conclusions

The main research question was split up in two parts. The first part (A) focused on the relationship between offshoring and success in an offshore CSD project. The second part (B) focused on ways to positively influence the success of such a project.

5.1.1 A: Offshoring and performance

Question A was: *Which threats or opportunities cause the success or failure of an offshore development project?* This question was split up into three sub questions:

A.1: What is the difference between an onshore and an offshore CSD project?

Hypothesis: In an offshore CSD project two teams are located in different countries or time zones. They cannot meet regularly face to face.

The literature showed that offshore CSD projects are different in comparison with onshore projects, because five kinds of distances exist: geographical, time, cultural, organizational, and stakeholder distances influence collaboration in offshore project teams. The results showed that all types of distances influence the success: all were mentioned in the interviews as causes for failure. When the distances were decreased by coordination measures such as cultural training or communication tools, this was mentioned as a cause for success. All types of distance therefore play an important role in offshore CSD projects. We accept the hypothesis, with the addition that offshore CSD projects also face cultural, organizational, and stakeholder distances.

A.2: When are offshore CSD projects successful?

Hypothesis: Offshore projects are successful when they are completed on time and within budget, with the right scope and quality.

This hypothesis can be accepted, since the answer was based on literature research. The results also showed that projects were successful when they had the right scope and quality, and were completed on time and within budget. Asking the onshore and offshore managers how successful the projects were on each aspect, gave a good overview of the project results. This was confirmed by the questionnaires.
With regard to scope, most projects were successful. The results show that many projects were not successful in the areas of time and costs. We therefore conclude that these aspects of success are most under pressure in offshore CSD projects. A customer will not accept software that does not have the functionality that was agreed upon. Therefore, it is possible that this conclusion is also applicable to onshore CSD projects. Since this type of projects was out of scope, we do not have data about this type of projects and cannot draw any conclusions in that area.
The majority of the investigated successful CSD projects lasted less than six months, and had not many organizations involved in the projects. Complex offshore projects in terms of planning and

organization were mostly unsuccessful. This is a conclusion that is also applicable to onshore projects. Nevertheless, if a project is done offshore and a project manager wants it to be successful, it would help to consider a less complex organization or splitting the project into different sub-projects supported by several business cases. Splitting the project would result in shorter planning periods, which helps to finish on time.
The results also showed that projects are mostly successful when the team members know each other from previous projects. They know how to collaborate and who the customer is. The team members already learned a lot, and this resulted in successful projects. Some product software companies, that were among the most successful projects, had almost ten years of offshore collaboration experience. The team members knew what the other country's knowledge, expertise, and experience level was. We therefore conclude that successful projects consist of team members that have shared previous experiences.

A.3: Why is the success of offshore CSD projects different, compared to onshore CSD projects?

Hypothesis: This is different because geographical, time, cultural, organizational, and stakeholder distances negatively influence the coordination and the communication in a team. In the end, this influences the project success.

Based on the conclusions that were mentioned in the previous questions, we accept this hypothesis. The results show how the different parts of success were influenced by the distances in offshore projects. Projects that were successful with regard to scope and quality executed a lot of activities onshore. Small parts of the projects were offshored to other countries. Unsuccessful projects with regard to scope and quality offshored a lot of their activities. Based on these results, we conclude that the distances negatively influence scope and quality. This is not the case with projects that were completed within budget. As was already mentioned in the introduction of this thesis, projects are often offshored to keep the costs low. This goal is confirmed by our results: successful projects with regard to costs have offshored more parts of activities than unsuccessful projects.
The main reason for the different success in offshore CSD projects is the interaction between the different success aspects: the desired result of low costs often results in lower quality and scope success, and the desired right scope and quality often result in too high costs. This interaction is also visible in the causes for success and failure. We see that in a lot of unsuccessful projects a limited amount of money was available for extra coordination measures. This caused success in the area of costs, but failure in the areas of scope and quality.

Conclusion A: Which threats or opportunities cause the success or failure of an offshore development project?
Based on the literature research and the results, we found that distances in offshore CSD projects influence the results negatively. Offshoring means a threat to scope and quality of a project, but also means an opportunity to lower costs. The threat consists of geographical, time, cultural, organizational, and stakeholder distances that affect a project highly when it is a long and complex project, and when the team members do not know each other and the customer at start of the project. Another threat is that the project team members focus on reaching costs benefits, which can cause projects to be de-scoped and below quality.

5.1.2 B: Coordination and communication

Question B was: *Which coordination measures should be introduced in a project, in order to mitigate these threats or exploit the opportunities?* The literature research in this thesis explained the influence of coordination and communication on the success of projects. What coordination measures can be

applied, and which measures increase the chance on a successful result? These are the two sub-questions that have to be answered.

B.1: Which coordination measures exist in a CSD project?

Hypothesis: Five categories kinds of coordination measures can be applied: application of standards, planning, formal mutual adjustment, informal mutual adjustment, and team selection.

The hypothesis mentioned five categories of coordination measures. In the interviews, measures from each category were mentioned. All categories of coordination measures can therefore be applied to influence the collaboration and the result of a project. We therefore accept the hypothesis. In the previous answers on question A, another type of coordination measure was mentioned: the choice which projects will be offshored. If for instance a project has many stakeholders involved, the project is complex. Offshoring would make the project even more complex, which would reduce the chance on success. The choice if a project should be offshored is therefore also an important coordination measure in a CSD project. Since the scope of this thesis is on projects where the decision to offshore is already made, we will not elaborate on this subject. Nevertheless, if a project will be offshored, a project manager can still decide which parts of the project will be offshored. The results show that almost no software development activity is offshored completely. The onshore-offshore mix has to be determined. This is a powerful coordination measure as can be seen in the results: in unsuccessful projects the main offshored activity (unit build) is almost completely offshored, while in successful projects 30% of the activity was still kept onshore. The successful offshoring of activities started earlier in projects than unsuccessful offshoring. We conclude that determining the onshore-offshore mix for each activity is another coordination category that influences the success of a project.

B.2: Which coordination measures should be introduced, in order to positively influence the success of offshore CSD projects?

Hypothesis: In an offshore CSD project all coordination categories should be applied. Special attention should be given to informal mutual adjustment, for example the introduction of an onsite coordinator and facilitating contact between the customer and the back-office.

The results show that the hypothesis is partly true. First of all, as was mentioned in the previous answer, determining the right onshore-offshore mix is also influencing the project. To have a successful result on all four aspects of success, a project should find a balance between keeping activities onshore and offshore. The results showed that successful projects with regard to scope, quality, and time, had more team members onshore than the unsuccessful projects. In order to have good results, team members should be available both onshore and offshore. The results of this thesis also show that in successful projects, the main offshored activity was the detailed design, whereas unsuccessful projects offshored mainly the unit build. We conclude that if the back-office is earlier responsible for CSD activities, this improves the results of the project.
In the second place, not only informal mutual adjustment, but also planning and formal mutual adjustment need extra attention. The results show that informal mutual adjustment is a major cause of success. Nevertheless, unsuccessful projects did not mention this category as the main reason for failure: planning was the main reason for failure that was mentioned. Furthermore, formal mutual adjustment was mentioned as a major reason for success by the successful projects, but not by the unsuccessful projects. We conclude that good informal mutual adjustment and formal mutual adjustment are major causes for project success, and that bad planning is a major cause for project failure. These three coordination categories therefore need special attention in an offshore CSD project.

Thirdly, standards are not mentioned by successful projects as a cause of success and by unsuccessful projects as a major cause of failure. Although it is necessary that the same method and concepts are used, this category does not need a lot of extra attention.
In the fourth place, the results show that the introduction of an onsite coordinator and direct contact between the customer and the back-office not necessarily lead to project success. A project with experienced team members is more worth than a project with an onsite coordinator but without experienced team members. Some successful projects had contact between the back-office and the front-office, but did not have good communication facilities. We therefore conclude that a successful project has coordination measures in all categories and does not focus on one category. Which kinds of coordination measures are useful in offshore CSD projects?

- **Standards:** Almost all offshore CSD projects used adapted versions of development methods. The waterfall method was applied in more successful cases than RUP or DSDM. We therefore conclude that a simple and adapted method is better than a complex and non-modified method. This is possibly because a non complex method is easier to communicate and understand when the team that uses it is spread over multiple locations.
- **Planning:** Unsuccessful projects mainly complained about unclear requirements. In an offshore CSD project, almost all requirements should be clear at the end of the Business & Requirements phase. Changes cost not only extra time and money: the results show that they also decrease team morale. Another important aspect of planning that needs attention is the timing of when the offshore team starts to be involved into a project. If this is too early, the team members have no tasks, although they cost money. Such a project will be unsuccessful on costs. If the offshore team is involved too late, time pressure prevents clear communication of the desired scope and quality, which decreases the chance on success on those aspects. A third important aspect of planning is budgeting. Successful projects had both flexible costs (time & materials) and fixed costs. Flexible costs should be applied in the Business & Requirements and the Design phases. Fixed costs can be applied to the Build & Test and the Deployment & Training phases (see Figure 3.1).
- **Formal mutual adjustment:** Successful projects had clearly divided responsibilities. The offshore team knew what was expected by the front-office. In unsuccessful projects, this was only done on high level, which resulted in a situation where the front-office did the creative work, and the back-office was the factory. In order to be successful, a project should involve both the onshore and the offshore team members in all main activities with clear responsibility descriptions.
- **Informal mutual adjustment:** In most cases the back-office had contact with the customer, an onsite coordinator was present, and a lot of traveling took place. In the interviews, the managers indicated that the offshore country did not have a complete view of all requirements and expectations. In order to have a better understanding of the desired scope and quality, the front-office should make the non-technical documents (the business analysis, requirements, functional designs, and other formal documents) in English available to the back-office. Offshore countries should also communicate all documents to the front-office.
- **Team selection:** Most successful cases mentioned capable, flexible, well educated, well experienced, hard working team members as a major reason for success. This coordination category influences all other four categories. If a project had coordination measures on all categories, but not very experienced people, it was often not a success. If a project had experienced team members but not a lot of coordination measures, it still was a success. We therefore conclude that in offshore CSD projects more attention should be given to team selection than would be the case in onshore CSD projects. At least a couple of experienced team members both onshore and offshore increase the chance on success.

Question A.2 already mentioned that time and costs need extra attention in order to have successful CSD projects. It furthermore mentioned that successful offshore CSD projects are shorter than six months, have no high organizational complexity, and consist of team members that share previous experiences. These are main project characteristics that also influence the success of projects. A project manager can split a project into small parts, reduce the organizational complexity, or select people with a lot of experience, as was also mentioned at team selection.
The results showed that successful coordination measures resulted in a lot of communication and sharing of knowledge, because the back-office and the front-office cooperated closely together. In successful projects each member has at least once a month contact with all other team members. Successful projects had a back-office who received enough knowledge from the customer via the front-office.
The social network analysis helped to define rules that can be applied to detect best practices and bottlenecks in offshore CSD teams. The results show that the successful projects had more best practices and less bottlenecks than the unsuccessful projects. The best practices indicate how a project should communicate, in order to be successful:

- There should be onshore or offshore representatives.
- There should be an onsite or offshore coordinator.
- There should be a liaison.
- The liaison should be an onshore representative.
- Someone at the back-office should have daily contact with an offshore representative.
- Someone at the front-office should have daily contact with an onshore representative.
- No team member should be a pendant.
- A team member should have daily contact with some other team members.
- There should be more than one onshore or offshore representatives.

According to the dendograms, the offshore and onshore team members in successful projects were all part of one team. In unsuccessful projects, we see that there were two teams or that the group of offshore team members was not well integrated with the group of onshore team members. We conclude that the best practices contribute to the creation of one integrated team, while the bottlenecks contribute to the creation of two teams, offshore and onshore.

Conclusion B: Which coordination measures should be introduced in a project, in order to mitigate these threats or exploit the opportunities?
The previous section mentioned a lot of ways to coordinate successfully offshore CSD projects. In question A we concluded that a main threat in offshore CSD is the focus on cost reduction. The answers on question B focus mainly on reaching scope, quality, and time success. All of these measures cost money. A threat could therefore be that the focus on coordination measures negatively influences the result of a project in the area of costs. This is often the case, as became clear in the results where we saw that many projects were out of budget.
Which success area or coordination measure category should get attention in order to be successful? The researched cases had different characteristics and followed different ways to reach success. To determine which way a project should go to be successful, good preparation is required. Preparation in which the desired performance is determined, the distances between the offshore and the onshore team members are depicted, the amount of offshoring of each activity is determined, and the extra coordination measures are chosen. These main activities help to prevent project failure and to increase the chance on project success. If we add these activities to the general CSD method that was described in Figure 3.1, we get Figure 5.1. The first steps help to setup a project situation in which offshore collaboration can take place. These steps are:

- **End performance evaluation:** Determine with all stakeholders what the most important aspect of success is in this project, and determine the importance of the other aspects.
- **Distances measurement:** Analyze how large the distances are between the locations, time zones, cultures, organizations, and stakeholders.
- **Onshore-offshore mixing:** Discuss and decide which parts of the project will be done offshore and which parts will be done onshore.
- **Coordination measure choice:** Discuss what coordination measures in each coordination category should be applied in order to face the distances in the projects and reach the desired end performance.

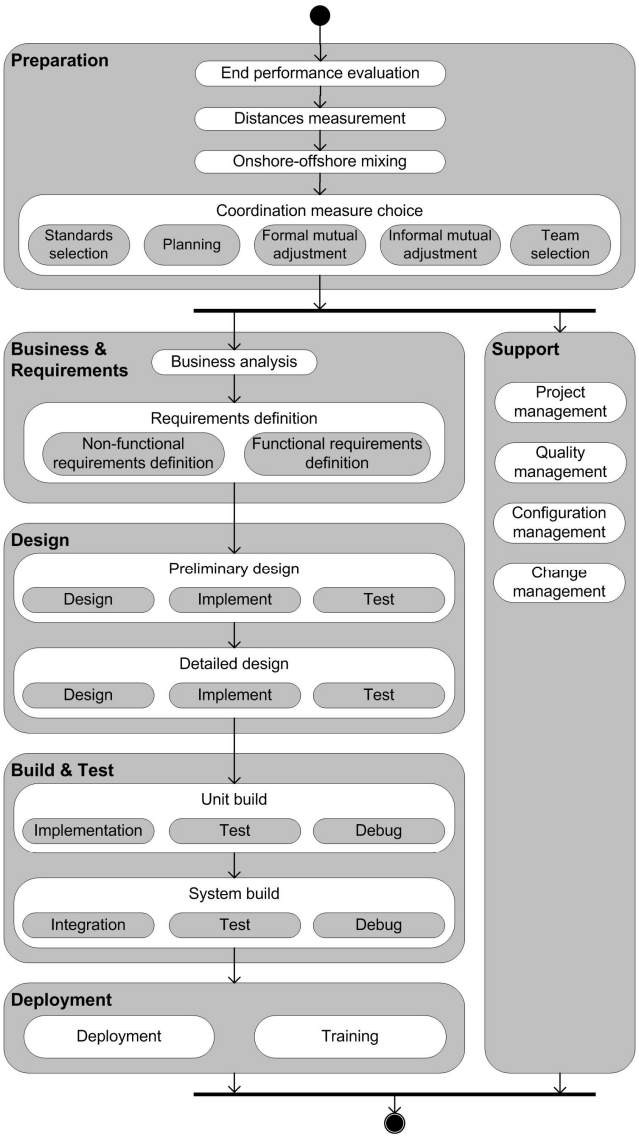


Figure 5.1. The expanded general software development method

5.2 Discussion

At start of this thesis, we asked ourselves “How should a distributed offshore software development project be coordinated, in order to increase the chance on a successful project?” The previous section answered that question by answering the sub questions. Our hypotheses were correct, but required some additions. The additions indicate the added value of the case research in this thesis to the

literature research that was described in chapter 2. The conclusions show that there is not one best way, but multiple ways to create successful offshore collaboration. The most important thing turned out to be the recognition of distances between both countries, and taking responsibility to reduce those distances by introducing certain coordination measures.
It is important to discuss the described results with regard to the scope of this thesis. The scope was set towards CSD projects, and not to other offshore IT projects. The conclusions can therefore not be applied directly to offshore non-CSD projects. Furthermore, the scope was set to projects that contain an onshore and an offshore team. Offshore projects where all development is done in the offshore country could apply the coordination categories that were described in this thesis, but the importance of each category and the influence on communication and performance can be different in comparison to the results that are described in this thesis.
In chapter 1, we discussed the justification of this research. The next section combines that discussion with the end results: are they applicable to science and society?

5.2.1 Applicability to society

In the conclusions it became clear that there is not one way to create a successful offshore CSD project. Multiple actions can lead to the same success. This thesis resulted in a description of those ways. It also described which aspects are important to pay attention to in a project. The results should be applicable to future projects that are performed by team members that are located offshore and onshore. This thesis therefore analyzed cases that came from real life. Because there were many results, we described an addition to the general software development method as a main pathway. The addition of a preparation phase can help project managers to pay attention to the aspects that were mentioned in this thesis. In order to have a clear overview of all results, we also made a sheet that can be found in Appendix 8. This sheet depicts the most important conclusions and results of this thesis for a project manager.
Besides using the results for the setup of future offshore CSD projects, the described methods can also be used by organizations to evaluate their projects. Organizations can use them to determine successfulness, distances, communication, and coordination in their projects. For each aspect we used a tool to measure and depict a project situation. The tools can be used to collect data about projects, and to indicate what the strong or weak points are in a project. Data of the projects in this thesis can be used as a benchmark for those projects. In this way, the results help current projects to improve.

5.2.2 Applicability to science

In order to get an overview of all coordination measures and their influence on communication, this thesis used a couple of research methods and a couple of data sources. These methodological and data triangulation led to all information that was described in this thesis. Unfortunately, not all data can be described. There was enough data to describe all cases in depth, but this was not the aim of this research. The aim was to analyze the similarities of successful and unsuccessful offshore projects. A lot of scientific researchers focused on the influence of offshoring in a team, and use an approach where a couple of cases are analyzed. This thesis went a couple of steps deeper, by analyzing a lot of cases and comparing them to each other. This was done by creating and using a research framework that was based on research from scientific literature.

5.3 Future research

This thesis was qualitative and not quantitative. We described that successful and unsuccessful projects had certain characteristics. The characteristics were mentioned by team members as a cause of success. This does not prove that those characteristics really led to the success. The research therefore opens ways for further research. By applying quantitative research on many offshore projects, statistically significant results can show which project characteristics are influencing the projects. Research in that area can also show how all coordination measures contribute to the final success of an offshore CSD project.
The conclusions pointed out that some results could also be applied to successful and unsuccessful onshore CSD projects. Future research can improve the results of this thesis by also researching onshore CSD projects and comparing them to the offshore CSD projects. This would help to identify which coordination measures and which communication best practices exclusively increase the chance on success of offshore CSD projects, and which ones increase the success of all CSD projects.
Another part of this thesis that opens doors to future research is the definition of success. In the research method, the formula of success was downsized in order to determine the success. Future research can determine how the weight of different success factors can be determined. Future research can also determine how scope, quality, time, and costs can be measured before and after a project in a consistent way.
The results sometimes showed unclear differences of the group that was not successful and not unsuccessful. This was caused by the small amount of cases that were researched. In order to have a useful benchmark tool for organizations, some parts of this thesis can be researched in future with more cases. If more cases are added, the differences in characteristics of successful and unsuccessful projects become clearer.
The application of the social network analysis helped to get an overview of offshore CSD teams. As becomes clear in literature about social network analysis, a lot of statistical analysis tools are available. We used a couple of them, but not much. The main reason was that we did not have a lot of data about the projects: information about one communication line between actors was applied, in order to get a higher response rate. Information about the expertise of all actors was not complete, since not all actors answered the questionnaires. Future research can focus on social network analysis, and perform elaborate research on a couple of offshore CSD projects. This would help to show the knowledge flows from the customer to the back-office. This would also help to validate the applicability of the bottlenecks that were defined in this thesis.

REFERENCES

Agarwal, N. & Rathod, U. (2006). Defining 'success' for software projects: An exploratory revelation. *International journal of project management*, 24, pp. 358-370.

Aranda, G.N., Vizcaíno, A., Cechich, A. & Piattini, M. (2006). Technology selection to improve global collaboration. *IEEE International conference on global software engineering*, 2006.

Argote, L., McEvily, B. & Reagans, R. (2003). Managing knowledge in organizations: an integrative framework and review of emerging themes. *Managing science*, 49(4), pp. 571-582.

Aron, R. & Singh, J.V. (2005). Getting offshoring right. *Harvard Business Review*, December 2005.

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. Derived on 16 November 2006, from http://www.acm.org/globalizationreport.

Austin, R.D. (2001). The effects of time pressure on quality in software development: an agency model. *Information systems research*, 12(2), pp. 195-207.

Baarda, D.B., Goede, M.P.M. de & Meer-Middelburg, A.G.E. van der (1996). *Basisboek open interviewen.* Houten, Educatieve Partners.

Baarda, D.B., Goede, M.P.M. de & Teunissen, J. (1997). *Basisboek kwalitatief onderzoek.* Houten, Educatieve Partners.

Baarda, D.B. & Goede, M.P.M. de (2001). *Basisboek methoden en technieken.* Groningen, Wolters-Noordhoff.

BBC News (2006). *Indian economy sees strong growth.* Derived on 19 December 2006, from http://news.bbc.co.uk/2/hi/business/6192583.stm.

Borgatti, S.P. (2002). NetDraw: Graph visualization software. Harvard, Analytic Technologies.

Borgatti, S.P., Everett, M.G. & Freeman, L.C. (2002). Ucinet 6 for Windows: Software for social network analysis. Harvard, Analytic Technologies.

Bozarth, C., Handfield, R. & Das, A. (1998). Stages of global sourcing strategy evolution: an exploratory study. *Journal of operations management*, 16, pp. 241-255.

Bruegge, B., Dutoit, A.H. & Wolf, T. (2006). Sysiphus: enabling informal collaboration in global software development. *IEEE International conference on global software engineering*, 2006.

Capgemini (2005). *Rightshore™ overview.* Utrecht, Capgemini.

Capgemini (2006). *Corporate website.* Derived multiple times between November 2006 and June 2007, at www.capgemini.nl.

Caprihan, G. (2006). Managing software performance in the globally distributed software development paradigma. *IEEE International conference on global software engineering*, 2006.

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

Carmel, E. & Agarwal (2001). Tactical approaches for alleviating distance in global software development. *IEEE Software*, March/April 2001.

Carmel, E. & Tjia, P. (2006). *Offshoring information technology: sourcing and outsourcing to a global workforce.* Cambridge University Press.Casey, V. & Richardson, I. (2006). Project management within virtual software teams. *IEEE International conference on global software engineering*, 2006.

Casey, V. & Richardson, I. (2006). Project management within virtual software teams. *IEEE International conference on global software engineering*, 2006.

Chan, K.C.C. & Chung, L.M.L. (2004). Integrating process and project management for multi-site software development. *Annals of software engineering*, 14, pp. 115-143.

Conchuir, E.O., Holmstrom, H., Agerfalk, P.J. & Fitzgerald, B. (2006). Exploring the assumed benefits of global software development. *IEEE International conference on global software engineering*, 2006.

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

Cusick, J. & Prasad, A. (2006). A practical management and engineering approach to offshore collaboration. *IEEE Software*, September/October 2006.

Damian, D. & Moitra, D. (2006). Global software development: how far have we come? *IEEE Software*, September/October 2006.

Davis, Bersoff & Comer (1988). A strategy for comparing alternative software development life cycle models. *IEEE Transactions on software engineering*, 14(10).

Dynamic Systems Development Method (2007). *DSDM roles.* Derived from http://www.dsdm.org/tour/roles.asp on 12 January 2007.

Eclipse Foundation, the (2007). *OpenUp – Eclipse Process Framework.* Derived from http://www.eclipse.org/epf on 12 January 2007.

Egan, R.W., Tremaine, M., Fjermestad, J., Milewski, A. & O'Sullivan, P. (2006). Cultural differences in temporal perceptions and its application to running efficient global software teams. *IEEE International conference on global software engineering*, 2006.

Faraj, S. & sproull, L. (2000). Coordinating expertise in software development teams. *Management science*, 46(12), pp. 1554-1568.

Farrell, D. (2006). Smarter offshoring. *Harvard business review*, June 2006.

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

Gartner (2005). *User survey: Offshore outsourcing, Western Europe, 2004-2005.* Gartner Dataquest.

Gartner (2006). *User survey report: strategies for IT outsourcing, worldwide, 2005.* Gartner Dataquest.

Gertler, M.S. (2003). Tacit knowledge and the economic geography of context, or the undefinable tacitness of being (there). *Journal of economic geography*, 3, pp. 75-99.

Gibbs, R.D. (2007). *Project Management with the IBM Rational Unified Process: lessons from the trenches.* Lebanon, IBM Press.

Gowda, R.G. & Polzella, D.J. (2006). Comparison of selected survey instruments for software team communication research. *IEEE International conference on global software engineering*, 2006.

Gumm, D.C. (2006). Distribution dimensions in software development projects: a taxonomy. *IEEE Software*, September/October 2006.

Hanneman, R.A. & Riddle, M. (2005). *Introduction to social network methods.* Riverside: University of California (published in digital form at http://faculty.ucr.edu/~hanneman/).

Harmsen, F., Lubbers, I. & Wijers, G. (1995). Success-driven selection of fragments for situational methods – The S^3 model. *Proceedings of the second international workshop on requirements engineering: foundations of software quality*, pp. 104-115.

Heeks, R., Krishna, S., Nicholson, B. & Sahay, S. (2001). Synching or sinking: global software outsourcing relationships. *IEEE Software*, March/April 2001.

Helms, R. & Buijsrogge, K. (2005). Knowledge network analysis: a technique to analyze knowledge management bottlenecks in organizations. *Proceedings of the 16th international workshop on database and expert systems applications.*

Helms, R. & Buijsrogge, K. (2006). Application of knowledge network analysis to identify knowledge sharing bottlenecks at an engineering firm. *Proceedings of the 14th European conference on information systems.*

Herbsleb, J.D. & Grinter, R.E. (1999). Splitting the organization and integrating the code: Conway's law revisited. *Proceedings of the 21st international conference on software engineering*, pp. 85-95.

Herbsleb, J.D., Mockus, A., Finholt, T. & Grinter, R.E. (2001). An empirical study of global software development: distance and speed. *Proceedings of the 23rd international conference on software engineering.*

Hofstede, G. (1980). Culture's consequences: international differences in work-related values. Beverly Hills: Sage Publications.

Holmstrom, H., Conchuir, E.O., Agerfalk, P.J. & Fitzgerald, B. (2006). Global software development challenges: a case study on temporal, geographical and socio-cultural distance. *IEEE International conference on global software engineering*, 2006.

Hsieh, Y (2006). Culture and shared understanding in distributed requirements engineering. *IEEE International conference on global software engineering*, 2006.

Jarvenpaa, S.L. & Leidner, D.E. (1999). Communication and trust in global virtual teams. *Organization science*, 10(6), pp. 791-815.

Jashapara, A. (2004). *Knowledge management – an integrated approach.* Essex, Pearson Education.

Jiang, J.J., Klein, G. & Discenza, R. (2002). Perceptions of software success: provider and user views of system metrics. *Journal of systems and software*, 63(1), pp. 17-27.

Kasse, T. (2004). *Practical insight into CMMI.* London, Artech House.

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

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

Kuni, R. & Bhushan, N. (2006). IT application assessment model for global software development. *IEEE International conference on global software engineering*, 2006.

Layman, L., Williams, L., Damian, D., & Bures, H. (2006). Essential communication practices for extreme programming in a software development team. *Information and software technology*, 48, pp. 781-794.

Linberg, K.R. (1999). Software developer perceptions about software project failure: a case study. *The journal of systems and software*, 49, pp. 177-192.

Maznevski, M.L. & Chudoba, K.M. (2000). Bridging space over time: global virtual team dynamics and effectiveness. *Organization science*, 11(5), pp. 473-492.

Narayan, T. (2006). Indian offshore firms are game-changers in Europe. *Indian management*, august 2006.

Nicholson, B. & Sahay, S. (2004). Embedded knowledge and offshore software development. *Information and organization*, 14, pp. 329-365.

Nonaka & Konno (1998). The Concept of "Ba": Building a Foundation for Knowledge Creation. *California management review*, 40(3), pp. 40-54.

Paasivaara, M. & Lassenius, C. (2006). Could global software development benefit from agile methods? *IEEE International conference on global software engineering*, 2006.

Prikladnicki, R., Audi, J.L.N. & Evaristo, R. (2003). Global software development in practice: Lessons learned. *Software process improvement and practice*, 8, pp. 267-281.

Prikladnicki, R., Audi, J.L.N. & Evaristo, R. (2006). A reference model for global software development: findings from a case study. *IEEE International conference on global software engineering*, 2006.

Procaccino, J.D., Verner, J.M., Overmyer, S.P. & Darter, M.E. (2002). Case study: factors for early prediction of software development success. *Information and software technology,* 44, pp. 53-62.

Ravichandran, R. & Ahmed, N.U. (1993). Offshore systems development. *Information & management*, 24, pp. 33-40.

Reuters India (2006). *India says needs pre-emptive measures on inflation.* Derived on 19 December 2006, from http://in.today.reuters.com/news/newsArticle.aspx?type=businessNews &storyID=2006-12-19T141410Z_01_NOOTR_RTRJONC_0_India-280769-2.xml.

Rosengren, K.E. (2000). *Communication – an introduction.* London, SAGE Publications.

Rottman, J.W. & Lacity, M.C. (2006). Proven practices for effectively offshoring IT work. *MIT Sloan management review*, Spring 2006.

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

Sakthivel, S. (2005). Virtual workgroups in offshore systems development. *Information and software technology*, 47, pp. 305-318.

Sarker, S. & Sahay, S. (2003). Understanding virtual team development: an interpretive study. *Journal of the association for information systems*, 4, pp. 1-38.

Schuh, P. (2005). *Integrating agile development in the real world.* Hingham, Charles River Media.

Seshagiri, G. (2006). Counterpoint: 'GSD: Not a business necessity, but a march of a folly.' *IEEE Software*, September/October 2006.

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

Sole, D. & Edmondson, A. (2002). Situated knowledge and learning in dispersed teams. *British journal of management*, 13, S17-S34.

Steenbeek, W., Wijngaert, L., Brand, M., Brinkkemper, S. & Harmsen, F. (2005). Sourcing decision making: elicitating consultancy knowledge using policy capturing. *13th European conference on information systems: information systems in a rapidly changing economy*, May 2005.

Stewart, T.A. (2001). *The wealth of knowledge: intellectual capital and the twenty-first century organization.* New York, Doubleday.

Swanson, R.A. & Elwood, F.H. (2005). *Research in Organizations: Foundations and Methods of Inquiry.* San Francisco, Berrett-Koehler.

Taxén, L. (2006). An integration centric approach for the coordination of distributed software development projects. *Information and software technology*, 48, pp. 767-780.

Yin, R. K. (2003). *Case study research, design and methods, 3rd ed.* Newbury Park: Sage Publications.

Zhang, S., Tremaine, M., Fjermestad, J., Milewski, A. & O'Sullivan, P. (2006). Delegation in virtual team: the moderating effects of team maturity and team distance. *IEEE International conference on global software engineering*, 2006.

APPENDIX 1: INTERVIEW QUESTIONS

1. **Project data:**
    a. Please describe the software that was made, and why it was made.
    b. Please describe who gave the assignment to make the software.
    c. Why was a part of this project offshored?
    d. What were the planned start-date, the planned end-date, the real start-date and the real end-date of the project?
    e. What were the total budgeted and the real costs?
    f. Which programming language was used during the project?
    g. Did the team members already work together in previous projects?

2. **Distances** in the project team:
    a. Which activities were done onshore and which were done offshore? Please indicate on the sheet (Appendix 2) how many people were involved in each activity.
    b. Which roles were located onshore and offshore? Please indicate on the sheet (Appendix 3) how many people had each role.
    c. Could you please describe all involved stakeholders?

3. **Coordination** in the project:
    a. Were the project price and the end-date fixed?
    b. Please describe the used development method.
    c. Was there traveling between the offshore and the onshore country? Who traveled between the countries?
    d. Were there any formal and informal meetings where onshore and offshore team members met, during the project?
    e. Was there an onsite coordinator?
    f. Which types of documents were unavailable to the front-office, and which ones were unavailable to the back-office?
    g. Please give at least one reason why the project was successful.
    h. Please give at least one reason why the project was not successful.

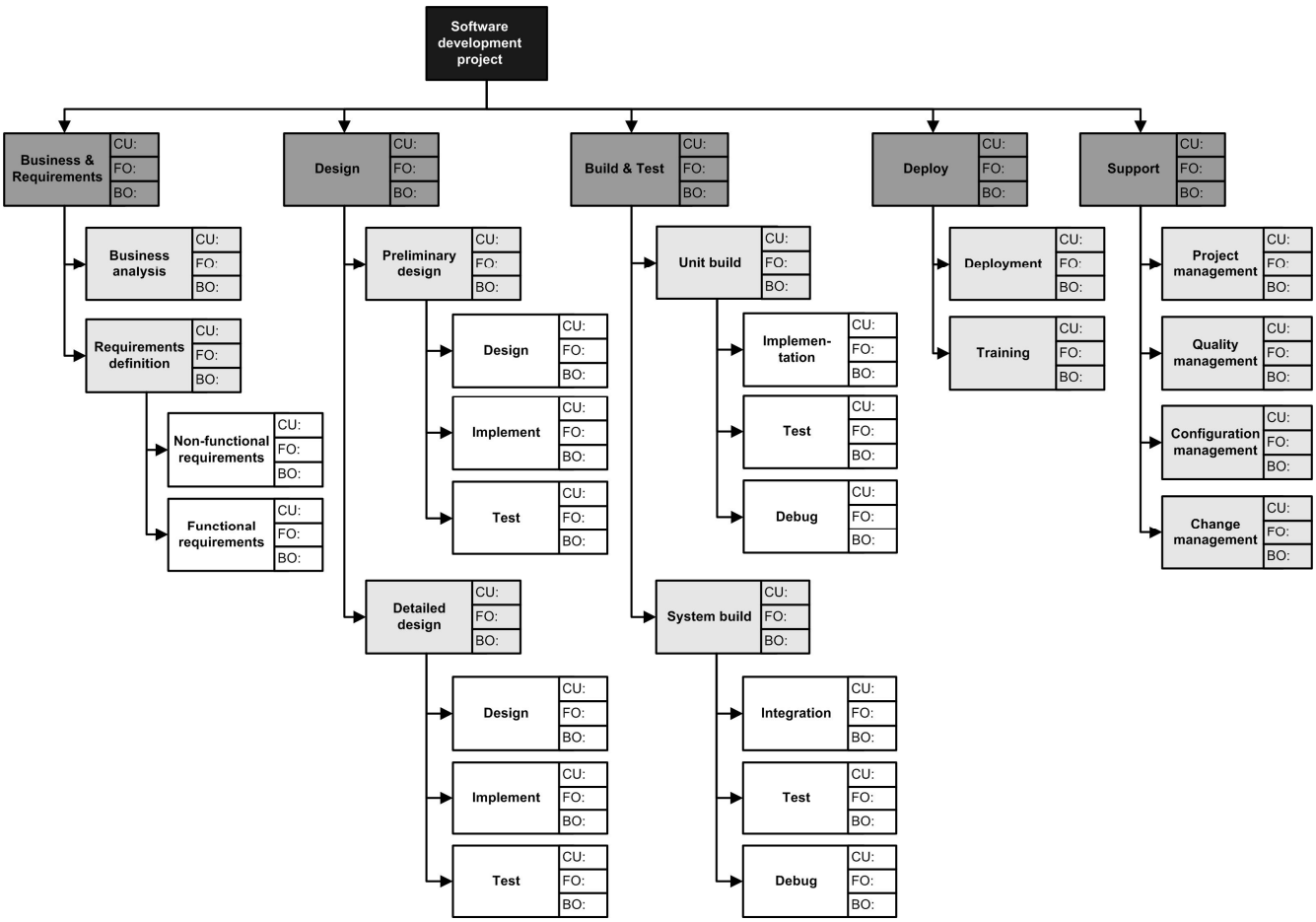
4. **Communication** between the project team members:
    a. Who were allowed and able to communicate with each other, during the project?
    b. Which documents were not available to the offshore or the onshore country?
    c. Which communication tools were used during the project?

5. **Performance** of the project:
    a. Please indicate how successful the project was on the areas of scope, quality, time, and budget. Please indicate this on a scale from 1 to 5, where:
        - 1 is very unsuccessful,
        - 2 is unsuccessful,
        - 3 is successful nor unsuccessful,
        - 4 is successful,
        - 5 is very successful.

APPENDIX 2: WORKSHEET CSD ACTIVITIES

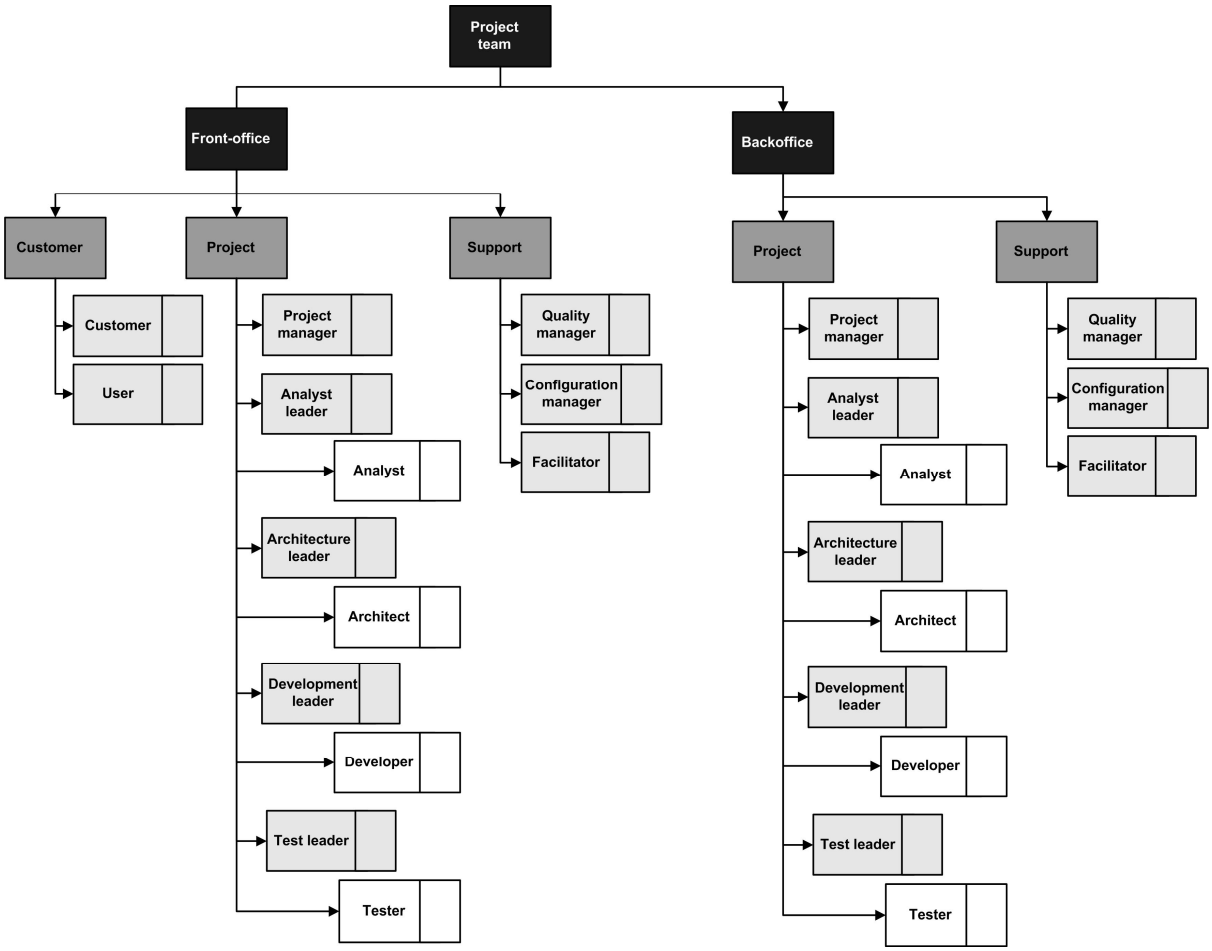
Please write in the boxes next to the activities how many team members of the back-office (BO), the front-office (FO) or the customer (CL) performed the following activities. Leaving boxes empty means that there was no-one who contributed to that activity.

Legend:
CU = Customer
FO = Front-office (onshore)
BO = Back-office (offshore)



APPENDIX 3: WORKSHEET CSD ROLES

Please write in the boxes next to the roles how many team members had that role. Leaving boxes empty means that there was no-one with that role.

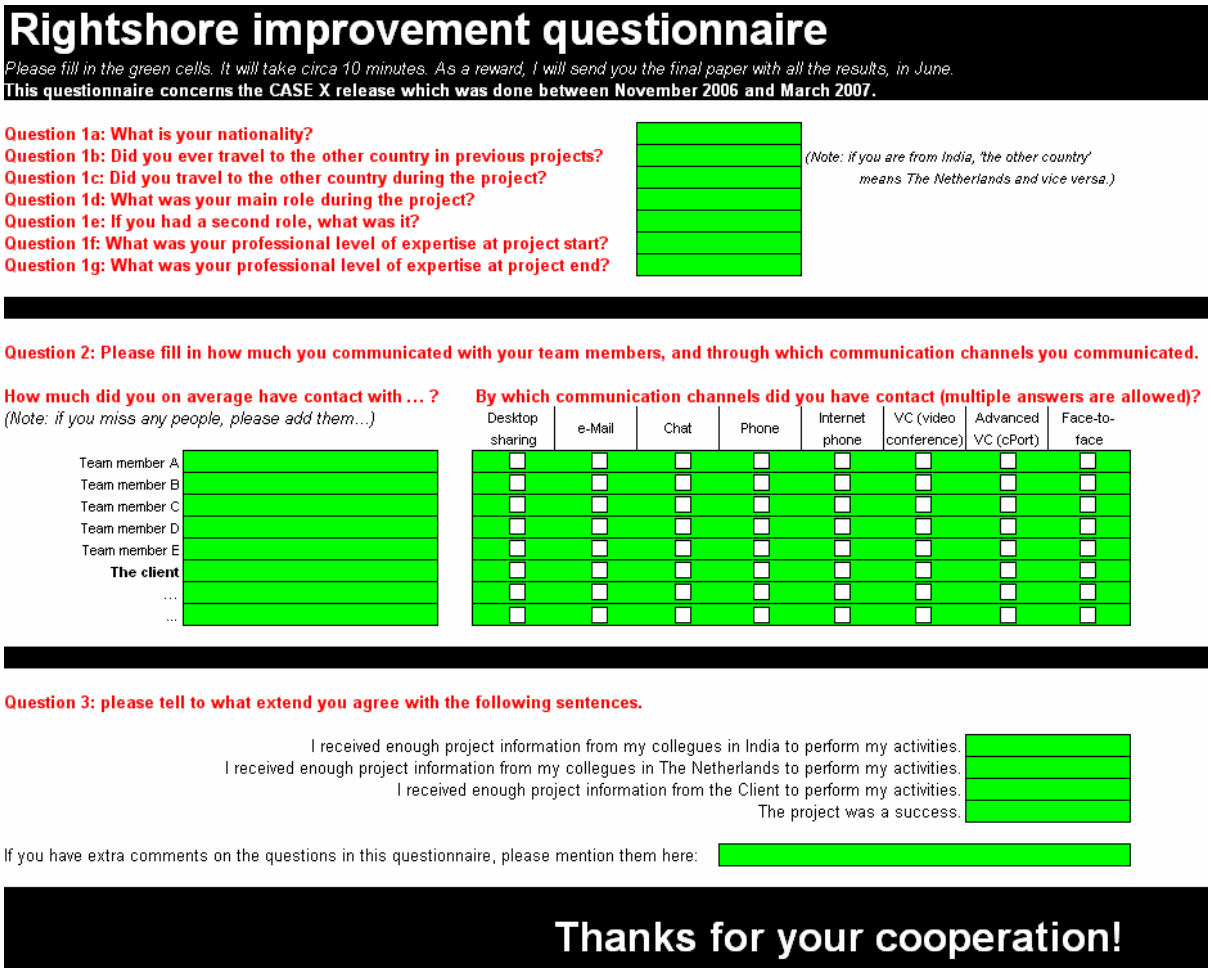


APPENDIX 4: QUESTIONNAIRE

This is an overview of the questionnaire that was sent to the team members of a couple of projects. Each questionnaire was a Microsoft Excel file. The names of all team members were written down in stead of 'Team member X', which is now shown because of privacy reasons. The team members were asked to fill in the green boxes. All green cells were drop-down boxes, except the checkboxes and the cell for extra comments. The options in these drop-down boxes were:

- **Question 1a:** India, The Netherlands
- **Question 1b and 1c:** Yes, No
- **Question 1d and 1e:** Analyst, Analyst leader, Architect, Architect leader, Developer, Development leader, Tester, Test leader, User, Project manager, Quality manager, Configuration manager, Facilitator
- **Question 1f and 1g:** Junior, Medior, Senior
- **Question 2 (how much did you have contact with...?)**: Never, A couple of times, Every month, Every week, Every day, More than once a day
- **Questions 3**: Totally not agree, Not agree, Somewhere in between, Agree, Totally agree

Below the communication channels team members could check each checkbox on if they used that channel with another team member.



Rightshore improvement questionnaire

Please fill in the green cells. It will take circa 10 minutes. As a reward, I will send you the final paper with all the results, in June.

This questionnaire concerns the CASE X release which was done between November 2006 and March 2007.

Question 1a: What is your nationality?
Question 1b: Did you ever travel to the other country in previous projects?
Question 1c: Did you travel to the other country during the project?
Question 1d: What was your main role during the project?
Question 1e: If you had a second role, what was it?
Question 1f: What was your professional level of expertise at project start?
Question 1g: What was your professional level of expertise at project end?

(Note: if you are from India, 'the other country' means The Netherlands and vice versa.)

Question 2: Please fill in how much you communicated with your team members, and through which communication channels you communicated.

How much did you on average have contact with ... ?
(Note: if you miss any people, please add them...)

By which communication channels did you have contact (multiple answers are allowed)?

	Desktop sharing	e-Mail	Chat	Phone	Internet phone	VC (video conference)	Advanced VC (cPort)	Face-to-face
Team member A								
Team member B								
Team member C								
Team member D								
Team member E								
The client								
...								
...								

Question 3: please tell to what extend you agree with the following sentences.

I received enough project information from my collegues in India to perform my activities.
I received enough project information from my collegues in The Netherlands to perform my activities.
I received enough project information from the Client to perform my activities.
The project was a success.

If you have extra comments on the questions in this questionnaire, please mention them here:

Thanks for your cooperation!

A filled in questionnaire is shown on the next page. This helps to understand the questionnaire better.

Rightshore improvement questionnaire

Please fill in the green cells. It will take circa 10 minutes. As a reward, I will send you the final paper with all the results, in June.
This questionnaire concerns the CASE X release which was done between November 2006 and March 2007.

Question 1a: What is your nationality?	The Netherlands	
Question 1b: Did you ever travel to the other country in previous projects?	No	*(Note: if you are from India, 'the other country' means The Netherlands and vice versa.)*
Question 1c: Did you travel to the other country during the project?	No	
Question 1d: What was your main role during the project?	5. Developer	
Question 1e: If you had a second role, what was it?	7. Tester	
Question 1f: What was your professional level of expertise at project start?	3. Senior	
Question 1g: What was your professional level of expertise at project end?	3. Senior	

Question 2: Please fill in how much you communicated with your team members, and through which communication channels you communicated.

How much did you on average have contact with … ? *(Note: if you miss any people, please add them…)*	**By which communication channels did you have contact (multiple answers are allowed)?**							
	Desktop sharing	e-Mail	Chat	Phone	Internet phone	VC (video conference)	Advanced VC (cPort)	Face-to-face
Team member A: 4. Every day	☐	☑	☑	☑	☑	☑	☐	☑
Team member B: 4. Every day	☐	☑	☑	☑	☐	☐	☐	☑
Team member C:	☐	☐	☐	☐	☐	☐	☐	☐
Team member D: 4. Every day	☐	☑	☐	☑	☐	☐	☐	☑
Team member E: 3. Every week	☐	☑	☑	☑	☐	☐	☐	☐
The client: 4. Every day	☐	☑	☑	☑	☐	☐	☐	☑
…	☐	☐	☐	☐	☐	☐	☐	☐
…	☐	☐	☐	☐	☐	☐	☐	☐

Question 3: please tell to what extend you agree with the following sentences.

I received enough project information from my collegues in India to perform my activities.	2. Not agree
I received enough project information from my collegues in The Netherlands to perform my activities.	4. Agree
I received enough project information from the Client to perform my activities.	4. Agree
The project was a success.	2. Not agree

If you have extra comments on the questions in this questionnaire, please mention them here:

Thanks for your cooperation!

APPENDIX 5: REASONS FOR SUCCESS AND FAILURE

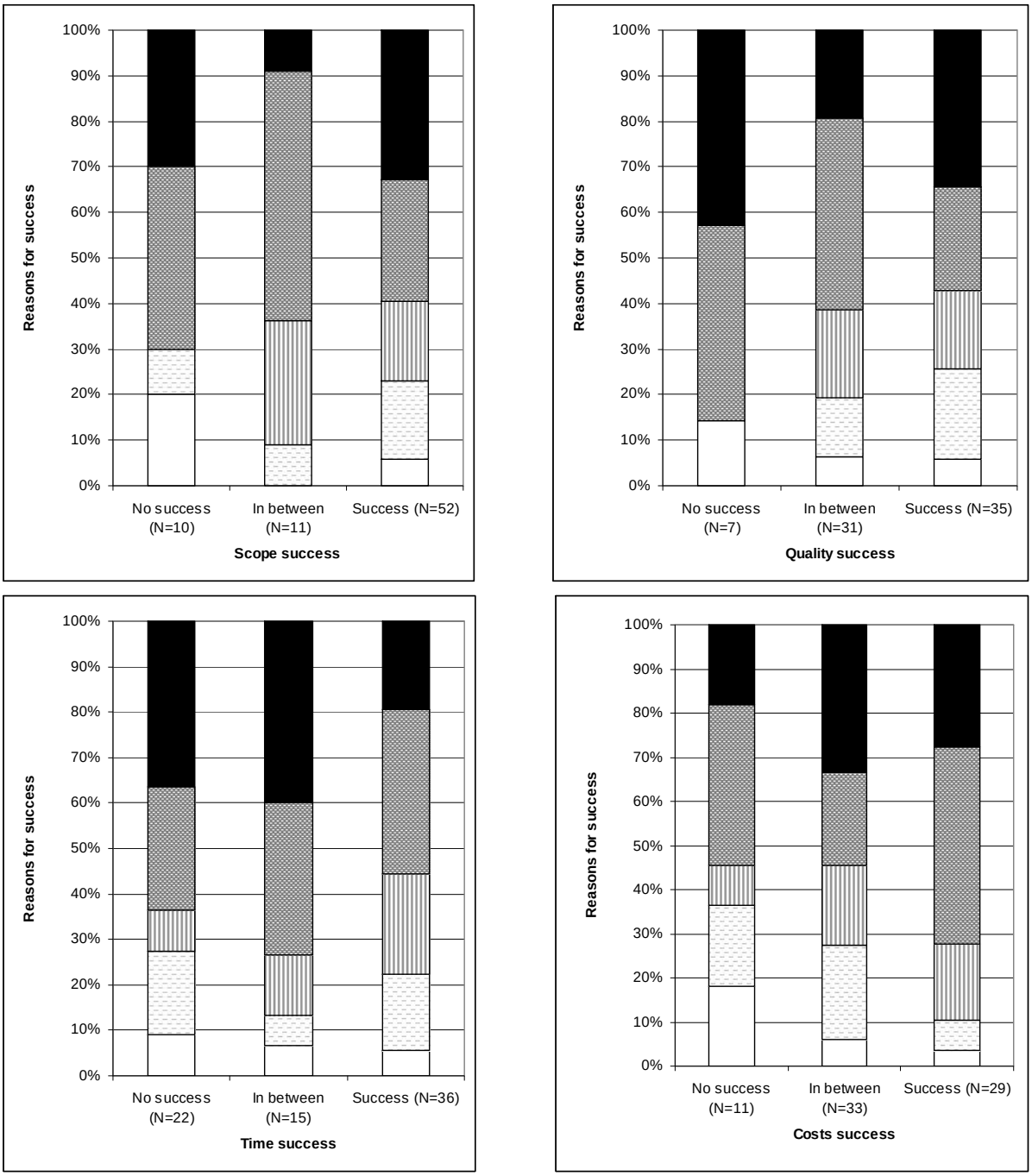
Below, the complete list of reasons for success and failure can be found.

	Reasons for success	Reasons for failure
Standards	• The project was very industrialized. • There was a clear flow of documents. • Standards were implemented. • The team members used an automatic report scripting tool. • The team members used a uniform issue tracking tool.	• Team members had different ideas of certain concepts, such as 'use case'. • There was not enough standardization. • There were no shared agendas between onsite and offshore.
Planning	• A part of the program was completely made in the front-office. • Extra flexible workforce from back-office helped to overcome the problems. • The project was split up in parts. • There was a good planning. • There was a good planning in front. • There were clear deadlines. • The team members used parallel executable use cases. • The team members used round-the-clock development. • The team members used use performance reports.	• Ad hoc issues were not done in the front-office but send to the back-office. • Bugs in the software were sent one by one to the back-office (in stead of bundled). • Changes were not clearly communicated to the back-office. • Due to a lot of uncertainties, planning was impossible. • Extra changes that the customer requested cost too much time. • Idle time in the offshore country was not efficiently used. • Some other projects overlapped with this project. • Some problems on both sides caused delays. • The back-office started too early. • The back-office was involved in the project very late. • The budget was too low to set up the project properly. • The customer did not know all his requirements. • The customer wanted a planning that was not longer than an onshore project. • The progress tracking tool was not useful. • There was a fixed price and a fixed date. • There was a large time pressure. • There was not enough preparation time for a proper planning. • There was sometimes too much to do, and sometimes not enough work. • There were a lot of changes in the requirements.
Formal mutual adjustment	• Management of both countries met a couple of times. • The back-office worked with good documentation. • The functionalities were clearly specified. • The project was tightly managed. • The responsibilities of the back-office were predefined. • The structure of the team was set-up together with the back-office team. • There was a general meeting for everyone at start of the project. • There was good security handling in the back-office. • The team communicated in English. • Extra project controls were introduced.	• All requirements were 'thrown over the wall'. • Everyone had to be screened before they could start, which caused inflexibility. • Functional design was done onsite and not offshore. • Functional documents were in Dutch. • Only programs with low customer interaction were done offshore. • Responsibilities were not clearly divided. • Some documents were in Dutch. • The back-office was not involved a lot in the project. • The customer did not want contact with the back-office. • The kickoff was onsite and not offshore. • The offshore country was not involved in non-technical tasks. • The requirements were unclear to the back-office. • The use cases were in Dutch and had to be translated. • There was no contact between the offshore team members and the customer. • There was not much documentation at start of the project. • There were visa problems which prevented traveling. • Translating the documents cost too much time.
Informal mutual adjustment	• An onsite coordinator was introduced. • Data, resources, etcetera were available to both countries. • People traveled from the back-office to the front-office. • People traveled from the front-office to the back-office. • People traveled to and from the back-office. • The background of the requirements was made clear to the back-office. • The customer changed according to the new situation.	• Communication with the back-office had no priority. • Offshore team members did not understand the customer's context. • Only one person communicated with the back-office. • Some documents at the back-office were not available to the front-office. • Some documents were not for everyone available. • The customer did not adapt to the new situation. • The cultural differences were large. • The front-office did not give the information that was needed. • The offshore team members did never say 'no'.

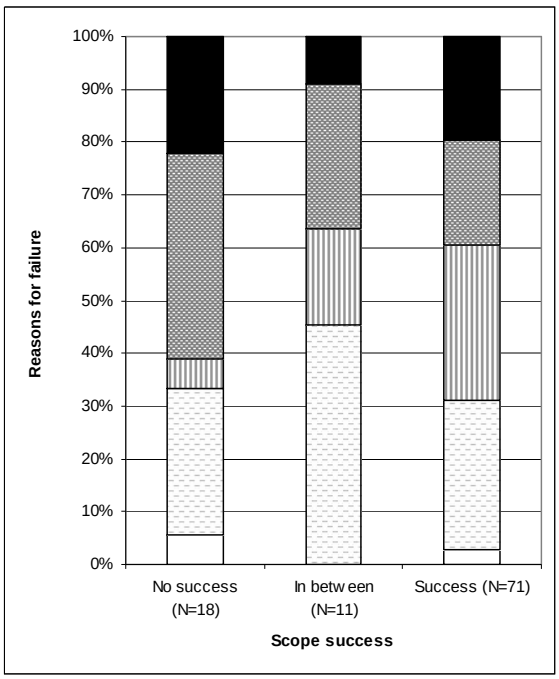
	• The database was located in the offshore country. • The front-office gave good support to the back-office. • The organization was informal. • There was a lot of sharing of cultures. • There was open communication between the customer and the back-office. • There were multiple ways of telecommunication. • The team used a collaboration tool. • The team used a videoconference tool. • The team used an advanced video conferencing tool. • The team used an instant messaging tool.	• The project manager did not go to the offshore country. • There was no advanced telecommunication facility available. • There was no cultural training. • There was no onsite coordinator. • There was no traveling from the back-office to the front-office. • There was not much traveling. • There was not one common repository.
Team selection	• The developers in back-office were also the unit testers. • The team members were high educated and skilled. • The team members were very disciplined. • The team members were very experienced. • The team members were very flexible. • The team members were very kind. • The team members were very professional. • The team members worked very hard. • A nearshore and an offshore development center were combined in one team. • The team finally learned to improve the situation. • The team consisted of a good mix of junior and senior resources. • There were not many people in the front-office. • The right people were at the right place.	• An extra involved organization had all functional knowledge. • An onsite project manager did not clearly communicate about the project status. • Many seniors offshore did nothing in comparison with juniors. • Nonfunctional activities were not well done in the back-office. • One of the extra involved stakeholders was not very mature. • People did not have past offshore experience. • People from the back-office had not much experience. • People reacted late on mails and document requests. • People were not prepared enough. • Team members did not give priority to the project. • Test leader onshore was a Chinese, while the offshore members were from India. • The offshore team members had no good education. • The onsite team members were not very professional. • The project manager in the front-office had not enough experience at start. • The size of the team was too large. • There was a lot of job hopping in the back-office. • There was no flexibility in requirements possible in the team. • There were too many stakeholders involved in the project.
Indirect reasons	• All members felt part of one team. • All members had respect for everyone. • Everyone had the same goals. • The communication was straightforward. • The humor of people from India and Holland was the same. • The relationships were based on trust. • There was a high motivation in the back-office team. • There was a lot of informal and face-to-face communication. • There was a proactive back-office. • There was good cooperation. • There was good coordination. • There was good interaction between both countries. • There were good personal contacts which helped problem handling. • The team got a positive image of the back-office country. • The team members shared knowledge in the right way.	• Communication cost too much time. • Not all issues were communicated from the back-office to the front-office. • Team members had a negative mindset about the offshore country. • The back-office had no proactive attitude. • The front-office was an obstacle between the back-office and the customer. • The organizations differed a lot. • The relationship with the customer changed. • There was a 'we/they' feeling. • There was no open communication about problems or issues. • There was not enough (tele)communication. • There was not much trust between all stakeholders. • Two different stakeholders did not work together in the offshore country.

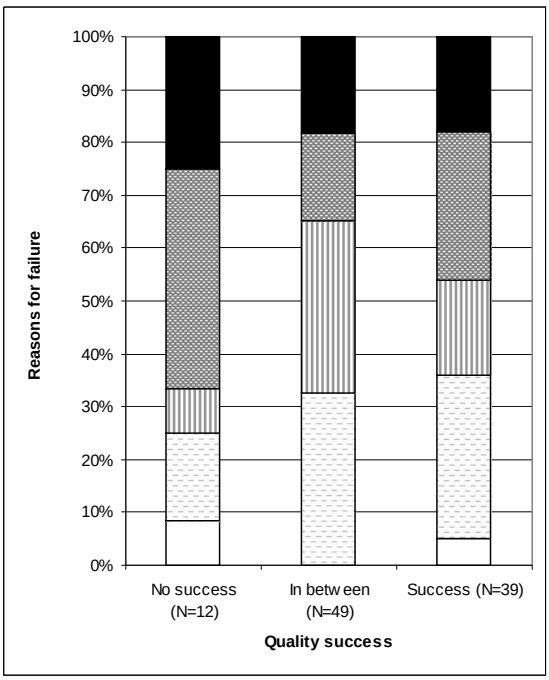
APPENDIX 6: COORDINATION MEASURE CATEGORIES AS REASON FOR SUCCESS

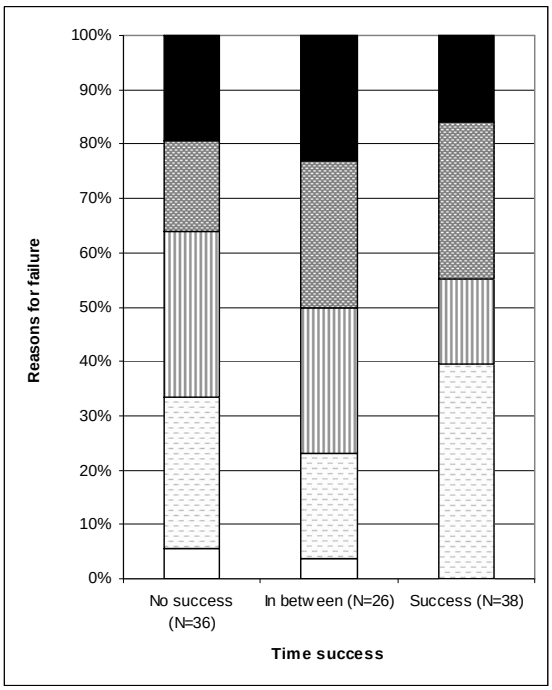
The graphs below depict which types of coordination measures were given as reasons for success by successful projects, unsuccessful projects, and projects that were not successful or unsuccessful:

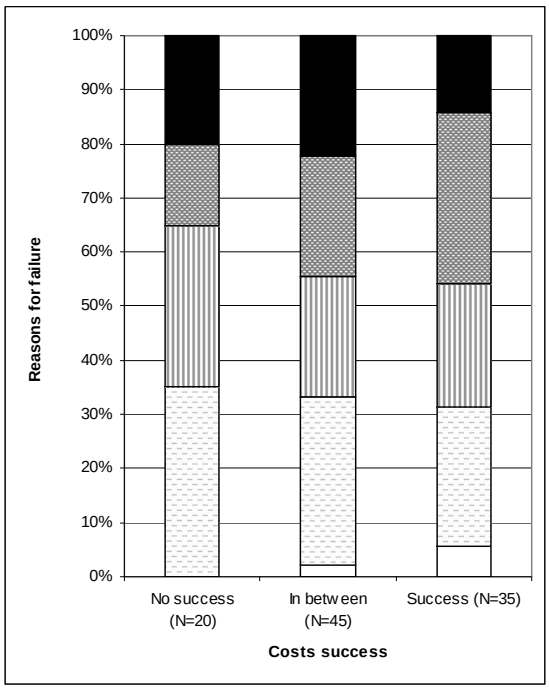


The graphs below depict which types of coordination measures were given as reasons for failure by successful projects, unsuccessful projects, and projects that were not successful or unsuccessful:

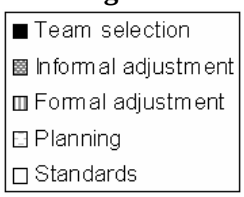








Legend:

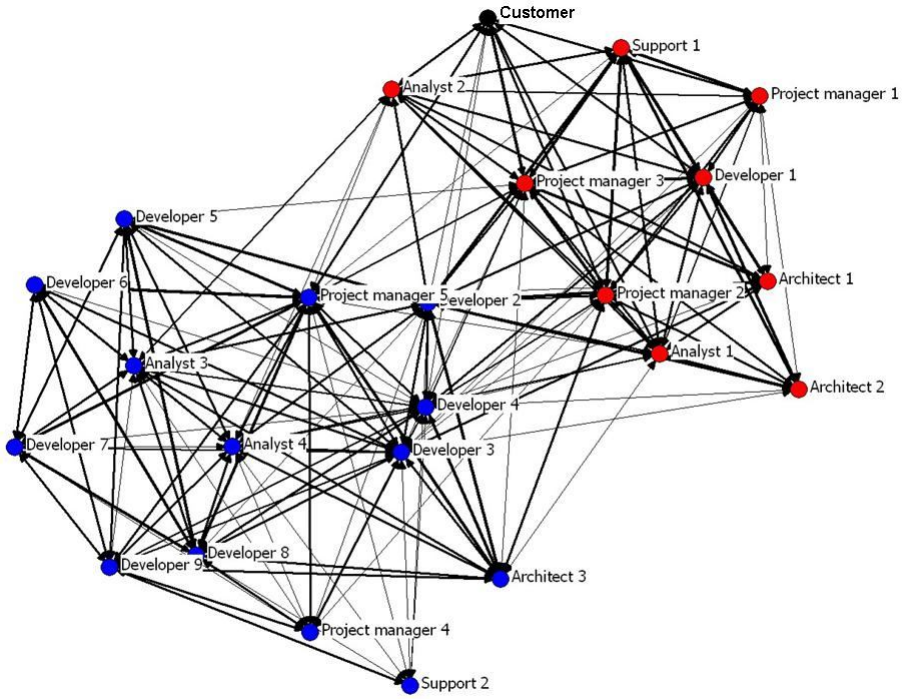


APPENDIX 7: NETWORK DIAGRAMS OF FIVE CASES

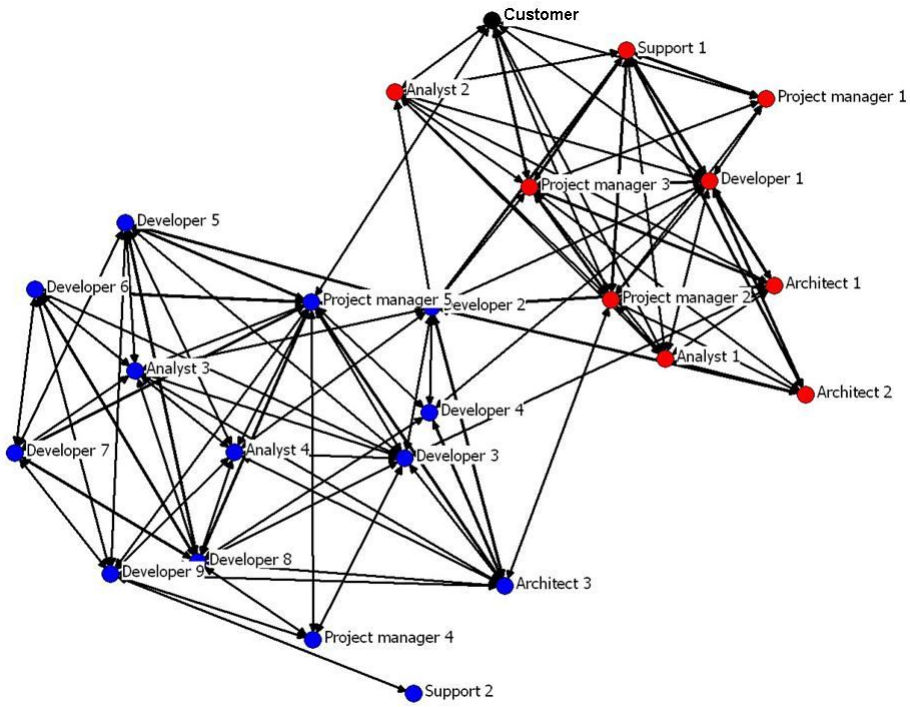
This appendix shows the network diagrams that were created by applying the Spring Embedded algorithm to the five cases that were researched in depth. Of each case, we present two diagrams that depict the contact that took place at leas once a month and at least once a day (Legend: blue color = offshore, red color = onshore, line thickness = amount of contact).

Case 2

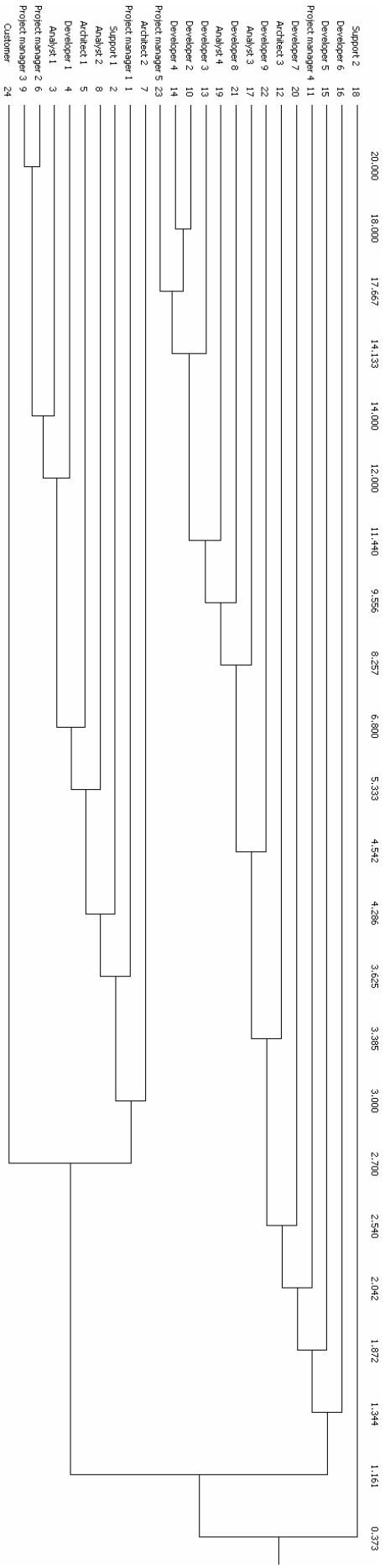
At least once a month contact:



At least once a day contact:

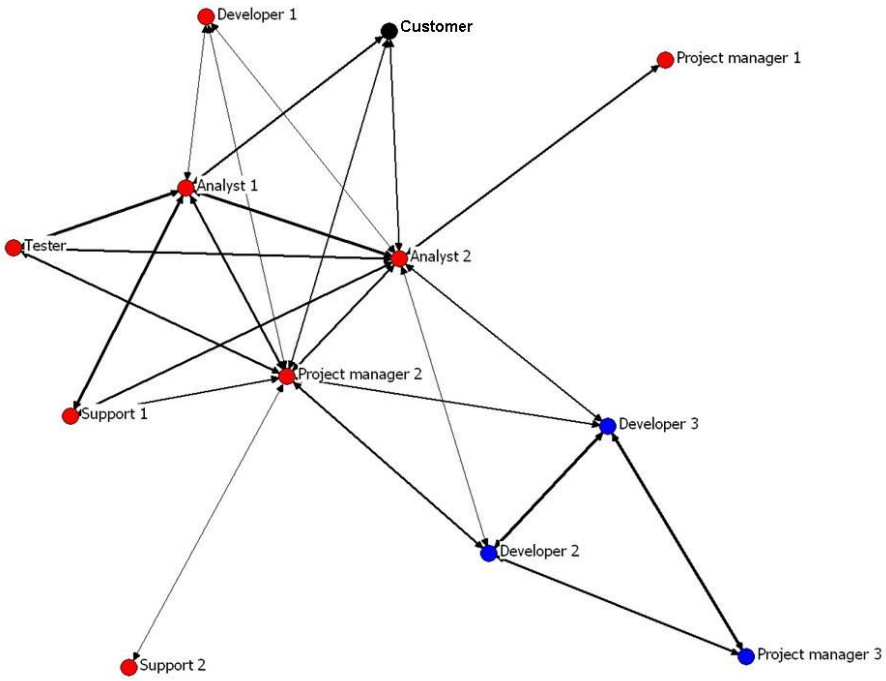


Clique analysis of the team members that had at least once a month contact:

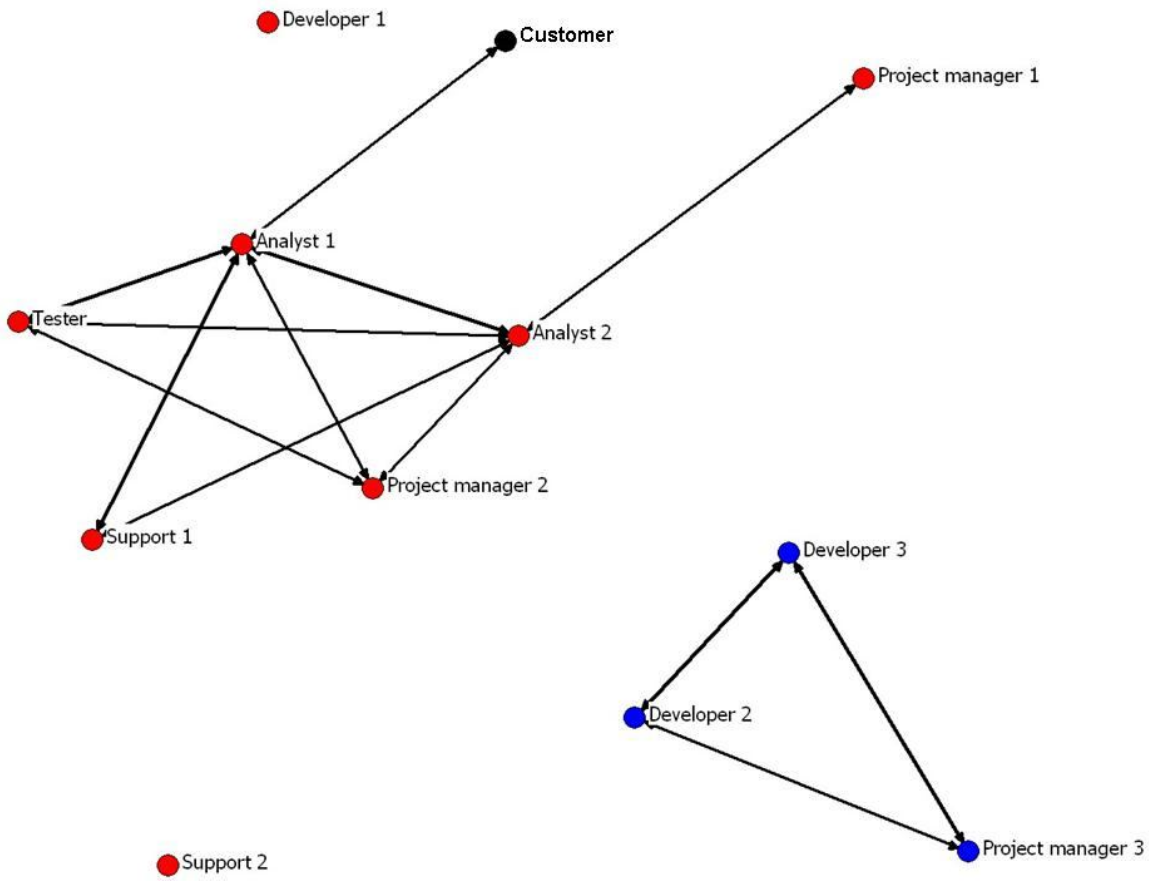


Case 5

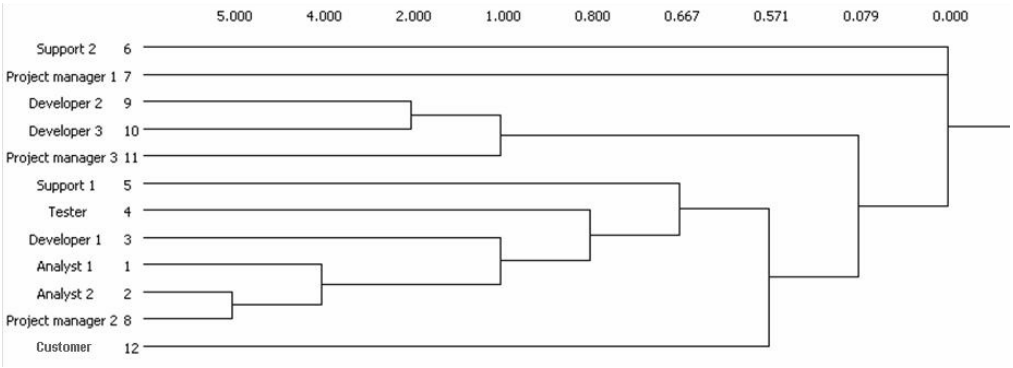
At least once a month contact:



At least once a day contact:

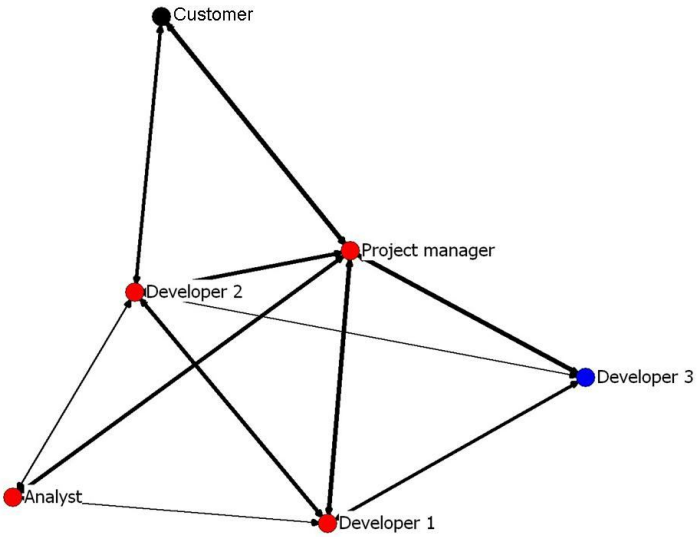


Clique analysis of the team members that had at least once a month contact:

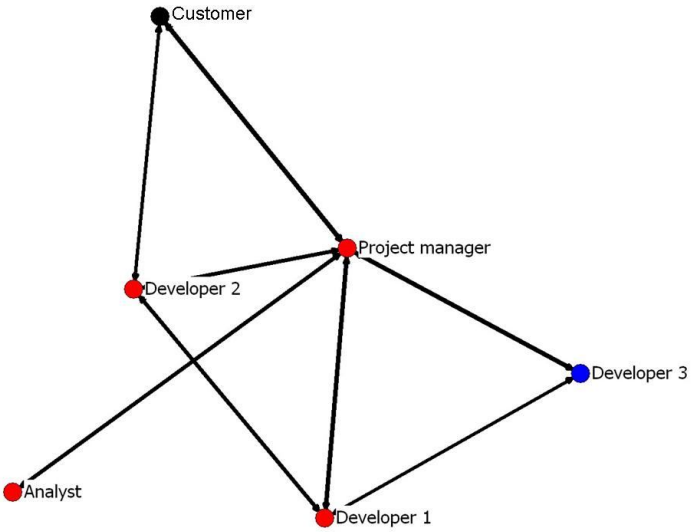


Case 6

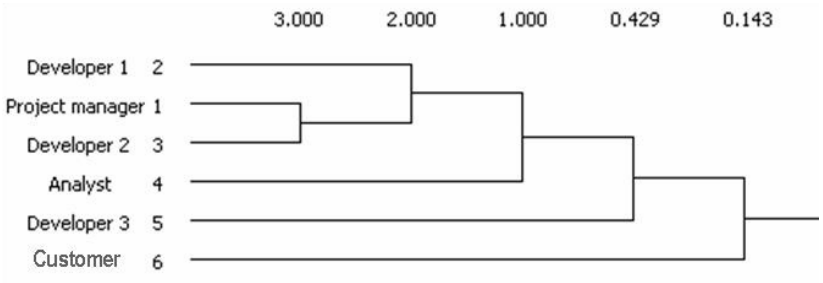
At least once a month:



At lease once a day:

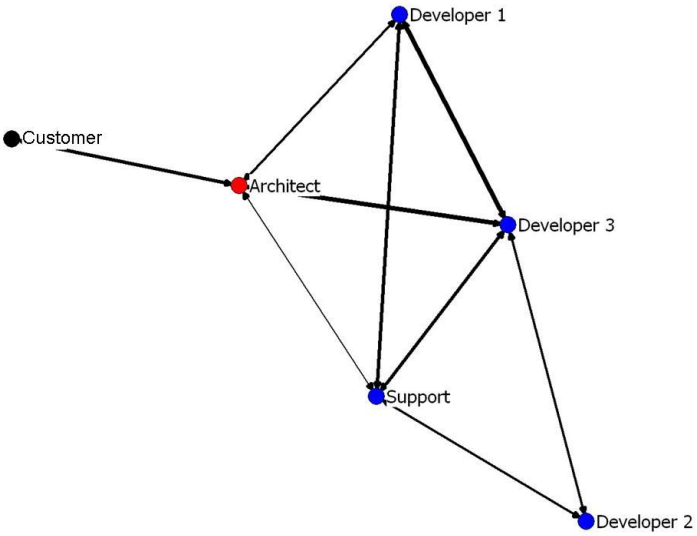


Clique analysis of the team members that had at least once a month contact:

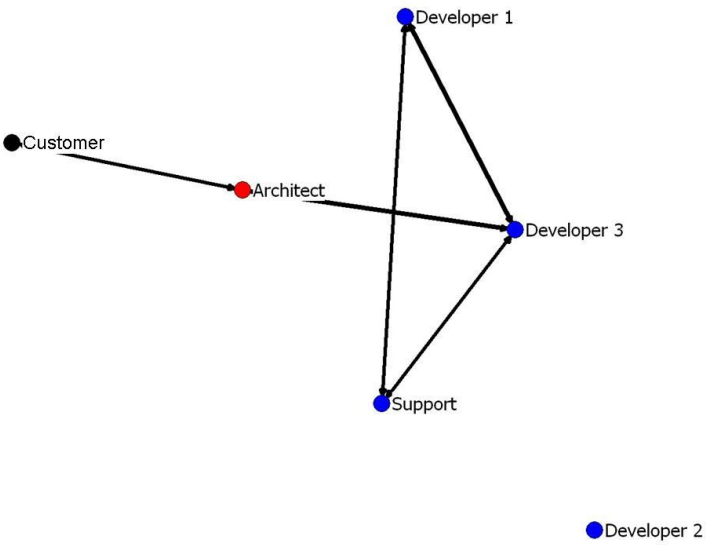


Case 9

At least once a month:

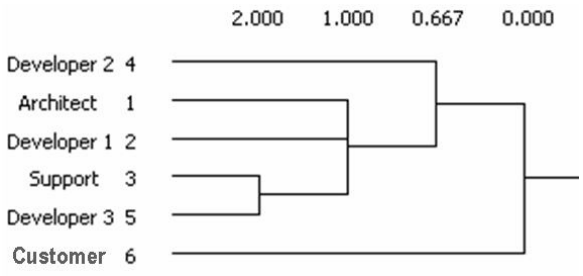


At least once a day:



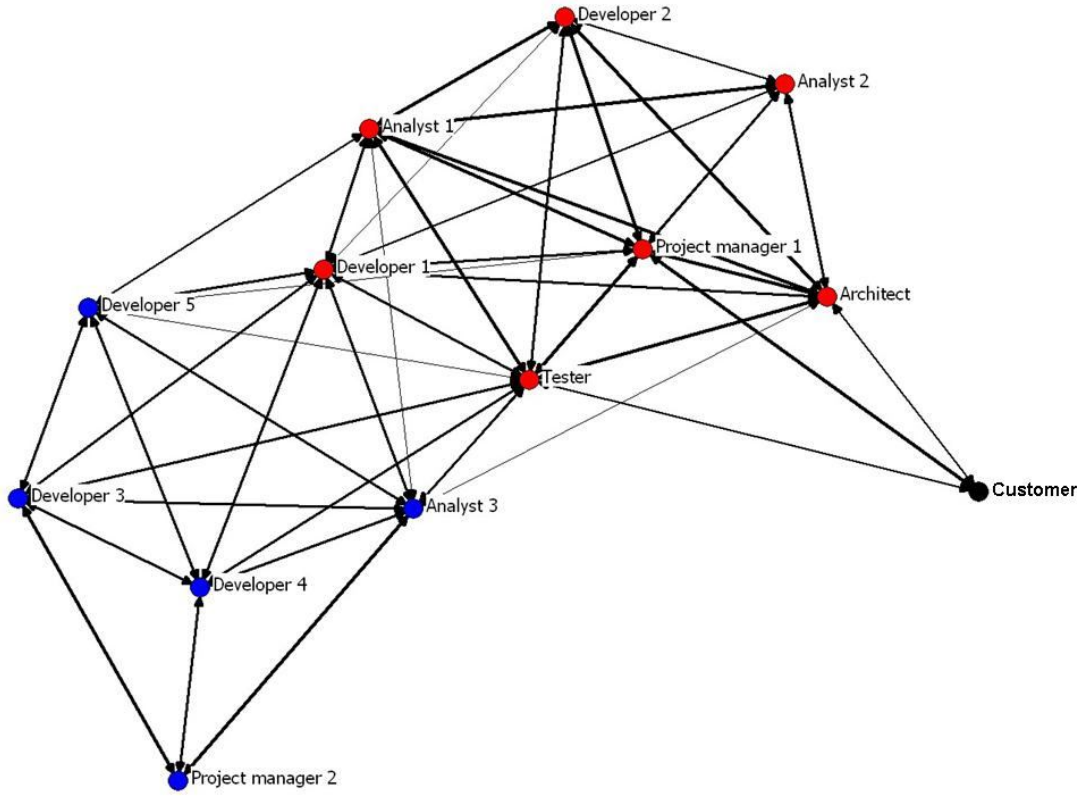
Developer 2

Clique analysis of the team members that had at least once a month contact:

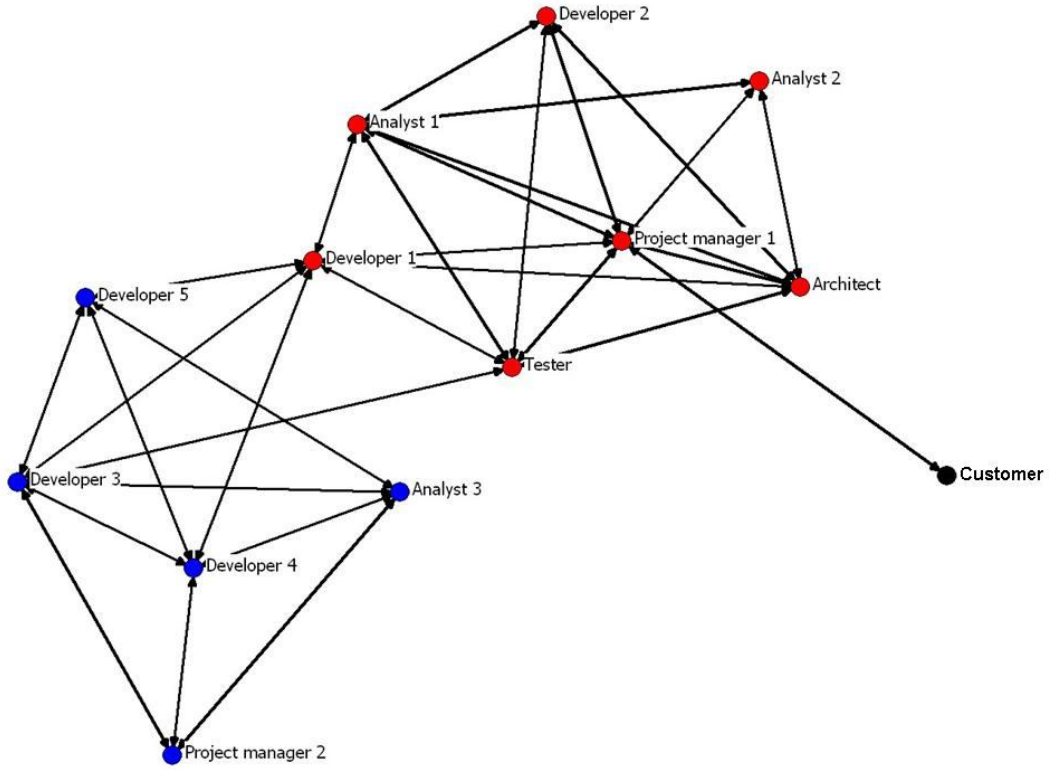


Case 12

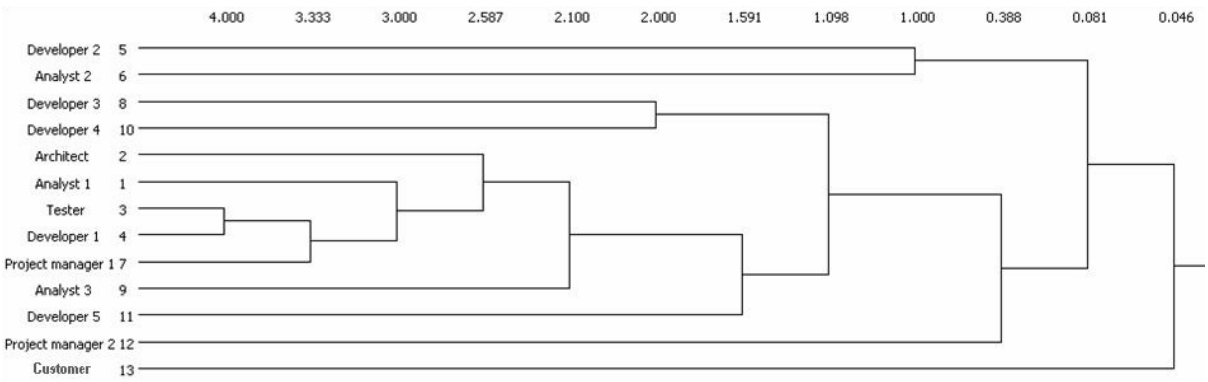
At lease once a month (size = operational expertise level):



At least once a day (size = operational expertise level):



Clique analysis of the team members that had at least once a month contact:



APPENDIX 8: HOW TO COORDINATE AN OFFSHORE CSD PROJECT SUCCESSFULLY?

This is an overview of how an offshore project should be prepared by project managers:

Four main activities are important, before an offshore project starts:

1. **End performance evaluation:** Determine with all stakeholders what the most important aspect of success is in this project, and determine the importance of the other aspects.
2. **Distances measurement:** Analyze how large the distances are between the locations, time zones, cultures, organizations, and stakeholders.
3. **Onshore-offshore mixing:** Discuss and decide which parts of the project will be done offshore and which parts will be done onshore.
4. **Coordination measure choice:** Discuss what coordination measures in each of the following coordination categories should be applied in order to face the distances in the projects and reach the desired performance:

Standards:

- Use a simple and adaptable method instead of a complex that is not adapted to an offshore situation.

Planning:

- Make sure that almost all requirements are clear at the end of the Business & Requirements. If this is not the case: don't offshore.
- Do not setup the offshore team too early (costs a lot), and not too late (not time for communicating functionality & quality, no team building).
- Do the first half of a project (business & requirements, design) with a time & materials contract, and the second half (build & test, deployment) with a fixed-price contract.

Formal mutual adjustment (to enhance formal communication):

- Clearly divide and communicate responsibilities.
- The back-office should be involved early in the software development cycle.
- Don't offshore too much! In successful cases, each CSD role is present on both locations. Offshoring often means not only cost reduction: also scope and quality reduction.

Informal mutual adjustment (to enhance informal communication):

- Make all non-technical documents in English available to the back-office.

Team selection:

- At least some experienced team members should be available onshore and offshore.
- If the team members already worked together before, or worked with the same customer, this is a large success factor.
- The following best practices are useful to facilitate successful communication in a project team:
    - There should be two or more team members onshore and offshore who communicate daily with the other country (onshore or offshore representatives).
    - There should be at least one person who communicates with multiple team members offshore and onshore (onshore or offshore coordinator).
    - There should be a team member who communicates daily with the customer and with at least one offshore team member.
    - Everyone at the back-office should have daily contact with an offshore representative.
    - Everyone at the front-office should have daily contact with an onshore representative.
    - A team member should have contact with at least two other team members each day.

Note: Successful projects are shorter than six months, have no high organizational complexity, consist of team members that have experience, and have stable requirements. If a project lasts long, is complex, has inexperienced team members, or has requirements that are often changed, it can be useful to reconsider offshoring.