TAMK University of Applied Sciences, Master's Degree
Degree Program in Information System Competence
Sanna Noronen

PROFESSIONAL MASTER THESIS

Making the Best of a Merger

A Case Study of Product Data Management Application Development Team

Supervisor M.Sc. Pekka Pöyry
Tampere 12/2009

Tampereen ammattikorkeakoulu, ylempi amk-tutkinto
Tietojärjestelmäosaamisen koulutusohjelma

Tekijä Sanna Noronen
Työn nimi Fuusion hyötyjen maksimoiminen – tapaustutkimus tuotetietohallintajärjestelmän kehitystiimistä
Sivumäärä 89
Valmistumisaika 12/2009
Työn ohjaaja FM Pekka Pöyry

TIIVISTELMÄ

Fuusioimalla kaksi yritystä saadaan molemmista käyttöön parhaat palat. Kuitenkin pääseminen tavoitetilaan, jossa on todella vain yksi yritys ja sillä yhdet prosessit ja tietojärjestelmät, vaatii paljon kovaa työtä; muutoksen hallintaa ja vaikeita päätöksiä. Tämän tutkimuksen tavoitteena oli löytää tapoja, joilla helpottaa tuotetietohallintajärjestelmän kehitysorganisaatiossa meneillään olevaa muutosta yrityksessä, joka on perustettu fuusioimalla huhtikuussa 2007. Vaikka organisaatio on ollut olemassa jo yli kaksi vuotta, yhteistyössä ja päivittäisten tehtävien hoitamisessa on vielä useita kipeitä kohtia.

Tämän opinnäytetyön tavoitteena oli luoda uusien toimintatapojen ohjeistus, jonka avulla mahdollistetaan tehokas tiimityö tässä uudessa monella paikkakunnalla toimivassa organisaatiossa. Ohjeistus luotiin enemmän johtamisen kuin teknisestä näkökulmasta ja se on suunnattu koko organisaation henkilöstölle, käytettäväksi sekä linjaorganisaatiossa että projekteissa, joissa järjestelmäkehitystä tehdään.

Tämä tutkimus suoritettiin tapaustutkimuksena, käyttäen myös benchmarking-menetelmää. Työn teoreettisen osion tarkoituksena oli osoittaa miten ohjelmistokehitys eroaa muista toimialoista ja miten erot vaikuttavat ohjelmistokehityksessä käytettäviin prosesseihin ja projektinhallintaan. Teorian pohjalta laadittiin myös ehdotus siitä, millainen prosessi ja mitä projektinhallinnan tapoja tutkittavaan organisaatioon kannattaisi valita.

Tutkimusmateriaali kerättiin teemahaastattelemalla kolmea henkilöä, jotka työskentelevät tutkimuksen kohteena olevan organisaation yhdessä toimipisteessä, ja yhtä henkilöä vertauksen kohteena olevassa yrityksessä. Haastatteluaineistot analysoitiin temaattisesti ja analyysin tuloksten avulla tunnistettiin ne alueet, joilla parannuksia eniten tarvittiin. Ottamalla askel eteenpäin perinteisestä tapaustutkimuksesta kehitettiin haastatteluista saatujen tietojen ja teorian pohjalta ehdotus uusista toimintatavoista ongelmallisilla alueilla.

Tutkimuksen kohteena olevassa organisaatiossa seuraava askel olisi ehdotettujen parannuksien käyttöönottaminen ja seuranta jonkin ajan jälkeen sen selvittämiseksi olivatko muutokset hyödyllisiä. Jatkotutkimuksena olisi mielenkiintoista selvittää miten muissa toimipisteissä ja maissa työskentelevät henkilöt ovat kokeneet fuusion ja millaisia parannuksia he ehdottavat. Vaikka ehdotetut parannukset on luotu tietyssä tilanteessa olevalle tietylle organisaatiolle, voidaan toimintatapoja käyttää missä tahansa organisaatiossa, joka sijaitsee usealla paikkakunnalla ja jossa tehdään monimutkaisen ja suuren järjestelmän kehitystä.

Avainsanat Ohjelmistotuotanto, projektinhallinta, fuusiot

TAMK University of Applied Sciences, Master's Degree
Degree Program in Information System Competence

Writer	Sanna Noronen
Thesis	Making the Best of a Merger - Case Study of Product Data Management System Development Team
Pages	89
Graduation Time	12/2009
Thesis Supervisor	M.Sc Pekka Pöyry

ABSTRACT

Merger is a way to get the best of two existing companies by combining them. However, reaching the target state of one united company with one set of processes and applications requires a lot of hard work; change management and difficult decisions. The goal of this research was to find ways to ease the ongoing change of the product data management application development organization in a company that has been created by a merger in April 2007. Even though the organization has existed for more than two years, there are still several sore points in the cooperation and daily work.

The target of this thesis was to create guidelines for new ways of working to enable efficient team work in the new multi-site organization. The guidelines were created more from the management than technical point of view and they are targeted for the whole personnel of the organization, to be used both in the line organization and in the projects in which the application development work is done.

This research was carried out as a case study with benchmarking. The purpose of the theoretical part of this work was to show how software development is different from other lines of business and how this affects the processes and project management in application development. Based on the theory also a proposal for selecting the process and team structure for the studied organization was created.

The research material was collected with semi-structured interviews of three persons working in one location of the target organization and one person in the benchmarked company. The interview materials were analyzed using the thematic analysis and results of the analysis were used to recognize the areas were improvements were mostly needed. By taking a step forward from a traditional case study, a proposal for new ways of working was created for the problematic areas based on the interview materials and theory.

In the target organization the next step would be to implement the proposed improvements and do follow-up after some time to see if the changes are beneficial. As further research it would be interesting to find out how persons working in other locations and in other countries have experienced the merger and what kind of improvements they would propose. Even if the proposed improvements were created with the specific organization and situation in mind, the ways of working can be applied in any organization that is located on several sites and is developing a complicated and large application.

Keywords Software production, project management, mergers

Table of Contents

1 Introduction

Major changes in the company level are very frequent nowadays. Merger is a way to create a larger company rapidly by combining two existing companies. Even if a merger is seen as a fast way to develop the operation of the company and grow the business opportunities, it is not an easy way to do this. Merger is a huge change for the employees of both old companies as there are two different histories and somehow these should be combined so that the future is one common story of the new company. To achieve this common ways of working should be agreed and implemented throughout the whole organization.

The company that is the target of the research in this thesis has gone through a merger in April 2007. Because of the merger the company is going through changes that affect the whole organization from the highest management level to the single persons and also the work in all levels from the general guidelines to the smallest details. Many decisions are needed just to reach an understanding for the target situation. Finding the best and most effective organization structure and agreeing the common ways of working is not easy inside one company, but when two companies with different histories and values are combined, the work is much more difficult.

After a merger the general idea is that costs can be cut down and personnel reduced by simplifying the IT (information technology) environment. However, there are usually at least two applications for one purpose and also two separate development teams for these applications and thus the simplification is not as easy as it may sound. This is also the case in the PDM (product data management) organization of the merged company with the difference that the number of both the applications and their development teams is five. This organization, which is a part of the IT organization, is the specific target of the research in this thesis.

Teams that form the PDM organization have already been working together in several release projects. I worked as one of the project managers in the first projects, until the summer 2008. As the target in the projects was to integrate the legacy applications, mainly the old roles and responsibilities and old ways of working were still used.

However, the target is to implement a new PDM application for the whole company and this application will replace all the legacy systems. The new application has been selected and the first project with it is already ongoing. From organization and management point of view target is to implement new roles and responsibilities and ways of working with the new application.

Managing and organizing the work of the team that is developing a large and complicated application is not an easy task. There is a huge amount of different kind of processes and instructions available to help in running a successful development project and organizing the work in the line organization. Even though software development and engineering can be considered a young field, one is easily lost in the jungle of the books and materials. The question is not how to create new ways of working but how to select the correct ones offered by many experts in the field. The selection cannot be done by using numbers and calculations, but discussions and cooperation are needed to find and decide with which ways of working to start with.

This research is an attempt to find the correct practical instructions in the situation the PDM organization is in. To understand the situation from the personnel point of view, the research was conducted as qualitative one, a case study with bench marking. The research materials were collected with semi-structured interviews, which were recorded and transcribed. As agreed with the supervisor from the commissioning company and so that the interviewees are not recognizable, the name of the company is not included in this work. I refer to the company as the ICT company.

First objective of this research is to understand the special nature of software and the effect it has on the processes and project management of software development and software engineering principles. This part is based on the available literature and it is used as guidance in the analysis of the interview materials. In the second part I present the analysis of the interviews: the experiences of the interviewed persons and their proposals for improvements. Third part is already a step forward from the traditional case study: the summary of the proposals for the ways of working in the areas that are identified as the most problematic ones. In the last chapter I give the conclusions and proposals for further studies.

2 Theoretical Framework of the Research

The very basic software related notions form the theoretical framework for this research: software development process, software project management and software engineering. All of these are needed to understand what we are talking about when we discuss software development and the ways of working in that area. These notions are also quite stable contrary to the current situation in many companies: even though there are many and frequent changes in the organizations, there are always some rules in the background that can be taken as guidelines when struggling with the new situations.

Software as such is different from other developed and producted artifacts and thus also the processes, engineering and project management must be understood from software point of view. In the following chapters I try to point out the special characteristics of software development that need to be taken into account when selecting the correct process and project management practices.

2.1 Software Development Process

In general a process is a series of steps that produce the designed end result. Acting according to a process can take time, space, resources and/or knowledge. In a process the similar actions and outputs are repeated when examined from a certain point of view. Processes are modelled and developed so that the quality, effectiveness and productivity of the target area of the processes can be improved.

Software development process describes the different phases, inputs, actions and outputs in software development. According to Pohjonen (2002, 21) the concept of software process is essential for the systematic development of information systems. Software process covers the development from the first idea to the ramp down, that is the whole life cycle of one system. Pohjonen (2002, 21) points out that also processes modular and dynamic, not stable as such.

Tsui and Karam (2007, 31-32) share this view. They say that there is a continuous development in the field of processes but still no one has proposed to abandon processes altogether. To begin with software development and support processes were invented to coordinate and manage complex projects involving many people. Still some kind of a process is needed to ensure a successful software development project as there are several sequenced and overlapping tasks that are performed by several persons. Process also defines the tasks of persons in different roles. In addition to the development tasks there are for example plans to be created and decisions to be made.

Like any other process, software process model defines the following:

- a set of tasks that need to be performed,
- the input and output from each task,
- the preconditions and postconditions for each task and
- the sequence and flow of these tasks.

These are needed to "provide guidance for systematically coordinating and controlling the tasks that must be performed in order to achieve the end product and the project objectives". (Tsui and Karam, 2007, 74.)

The software process models can be divided to traditional and modern ones, starting from waterfall model and ending up currently in the agile models. The main characteristics of traditional models are careful planning and documenting of tasks, milestones, specifications and architecture. The development is seen as a straight forward lifecycle that starts from requirement specification and ends up in maintenance. Agile models rely on communication and the phasing is not strictly defined. Most important is functioning software and immediate response to changes. In table 1 the characteristics of plan-driven and agile methods have been compared.

Table 1: Home-ground for agile and plan-driven methods (Abrahamsson, Salo, Ronkainen & Warsta 2002, 16)[1]

Home-ground area	Plan-driven methods	Agile methods
Developers	Plan-oriented; adequate skills; access to external knowledge	Agile, knowledgeable, collected and collaborative
Customers	Access to knowledgeable, collaborative, representative, and empowered customers	Dedicated, knowledgeable, collocated, collaborative, representative, and empowered
Requirements	Knowable early; largely stable	Largely emergent; rapid change
Architecture	Designed for current and foreseeable requirements	Designed for current requirements
Refactoring	Expensive	Inexpensive
Size	Larger teams and products	Smaller teams and products
Primary objective	High assurance	Rapid value

Also Tsui and Karam (2007, 132) have compared the characteristics of agile and traditional software processes with similar result. Based on the comparison they give a suggestion on which process to select for different kind of projects. According to their views traditional process

- is better suited to larger projects,
- can be used for mission-critical systems,
- defines many roles, which can be appropriate for most kind of people; doesn't require tight team playing; almost any personality will work, as long as the team members can follow rules,
- is better suited for larger companies with possibly geographically remote sites and more formal cultures and
- is less suited to cope with changes, assumes a relatively stable environment where requirements don't change much.

[1] Original source: Boehm B. 2002. Get Ready For The Agile Methods, With Care. Computer 35(1): 64-69.

Venkula (2005, 96-9) makes a distinction between a mechanical process and a flowing process. In mechanical process the end result can always be anticipated, the resources and methods can be strictly defined before hand with plans, procedures are not related to persons in any way and can be automated. The flowing process is the opposite: the objective is shaped during the development and can be something else than was anticipated, plans are for loose guidance and the procedures are related to the know-how of the persons and cannot be automated. Industrial manufacturing process is a typical example of the mechanical process. Flowing process is for example problem solving, creating something new and also development of organizations.

Based on Venkula's ideas software process should be understood partly as a flowing one. In software development something new is always created and in many cases it is difficult to know the end result exactly. From this point of view the processes should be used as instructions that give some guidelines for working, but do not restrict the creativity. For the people working in software development the flowing process means that they need to be ready to face uncertainty and unpredictability. Still parts of the more mechanical features are also needed in the processes, as software development is not only being creative but also some instructions need to be followed strictly.

2.2 Software Project Management

There are at least two organizations, IPMA (International Project Management Association) and PMI (Project Management Institute), which are generally and internationally understood as official organizations in the field of project management. Their task is to promote project management to business and organizations and also to certify project managers. Both IPMA and PMI have their own definition for project, and these are used widely when the word project needs to be officially defined. (International Project Management Association 2009, Project Management Institute 2009.)

However, in real life the word project has nowadays several different, even contradictory, meanings. Project can refer to a one time task were several parties are participating. It can also mean a temporary organization or a problem that is scheduled

to be solved. In some cases project is a unique assignement regarding specific requirements like objectives, time, cost and quality. Still there is one characteristic that is common for all different definitions: according to all of them project has a clearly defined beginning and end – projects do not last forever. (Artto, Martinsuo & Kujala 2006, 24-5.)

Even if project can be defined in different ways and examined from different points of view, a project needs always to be managed. Project management is organizing and managing the resources in such a way that the project can be ended with the achieved goals and objectives in the planned schedule with the planned budget. Resources include for example money, personnel, materials, energy, space and salaries. In addition to resources for example communication, quality and risks are also included in project management.

Lehtimäki (2006), who has a long experience in software project management, sees project management as quite a simple task, like shown below.

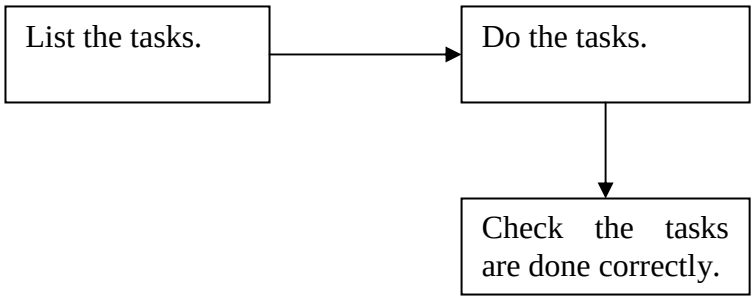

Figure 1. Simplistic project management (Lehtimäki 2006, 2)

Even though project management can be seen as this simple, there are quite many things that need to be taken into account when working in a project mode. Even more so, when the project is set up for software development. Even if the general definition applies also to software development projects, there are some characteristics that make these projects different from other projects, for example road building. Stepanek (2005, 7-8) has created the following diagram of the characteristics that are unique to software development and affect the software development projects. The uniqueness does not necessarily apply to all the single characteristics as such but software development is unique in the sense that it encompasses these all.

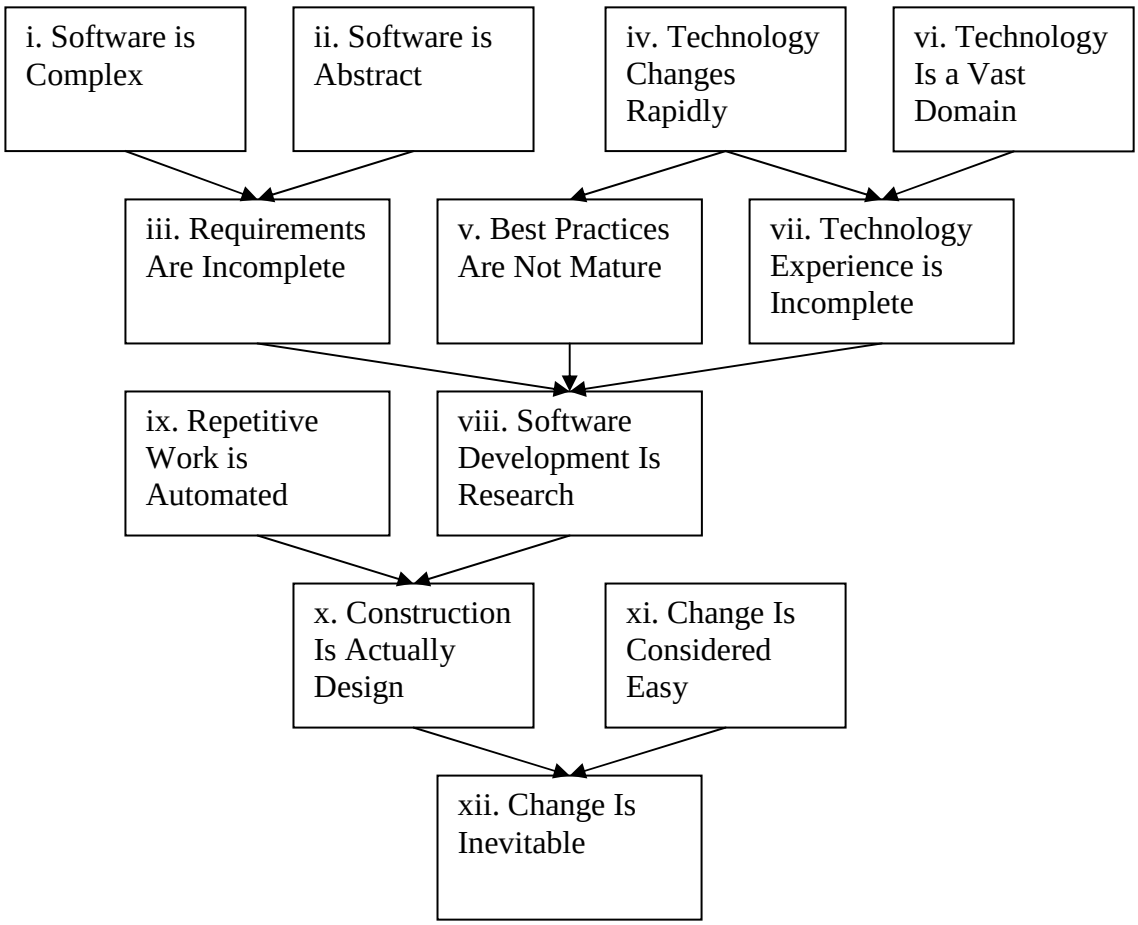

Figure 2. Characteristics of software development (Stepanek 2005, 8)

Stepanek (2005, 23, 48-49) has analyzed the project management tasks in relation to the software development characteristics described above and come to the conclusion that especially the following areas are effected by the uniqueness of software development: the management of scope, time, cost, quality and risk. He has come to the conclusion that there are the following ten hidden assumptions in the general project management guides that are not valid for software development:

- Scope can be completely defined.
- Scope definition can be done before the project starts.
- Software development consists of distinctly different activities.
- Software development activities can be sequenced.
- Team members can be individually allocated to activities.
- The size of the project team does not affect the development process.
- There is always a way to produce meaningful estimates.

- Acceptably accurate estimates can be obtained.
- One developer is equivalent to another.
- Metrics are sufficient to assess the quality of software.

Also Forselius, Dekkers, Karvinen and Kosonen (2008, 22) have found similar issues that affect the middle- and large-sized ICT projects. They say that “the objectives, completion criteria, costs and anticipated schedule are often prematurely set before there is adequate knowledge about the development scope and quality requirements. It becomes only too clear at the end of an ICT program just how far apart were the customer’s expectations and the suppliers understanding of the overall scope.”

As a solution to overcome the misunderstandings Stepanek (2005, 97) proposes the following agile practices:

- continuous development,
- on-demand programming,
- SWAT teams,
- subteam encapsulation,
- feature trade-off,
- triage and
- scoping studies.

For a major new system he sees the combination of subteam encapsulation, feature trade-off, triage and scoping studies as the best solution. Subteam encapsulation means that the large team needed for the system development is divided into smaller subteams. There are contact persons in each subteam for the communication. There are mediators that are not leaders but serve as conduits for information. The responsibilities of the teams are defined as well and as strictly as possible so as not to overlap. (Stepanek 2005, 104-105, 110.)

Feature trade-off is simply allowing the customer to change their mind about features they need: they can get new features inculded into the scope but only if they let go of others included earlier. Triage is a term borrowed from medical world and in software development it stands for prioritizing the features needed. Thus the developers always know what is the most critical thing to do. If the customer is not able to divide the

features into groups of “must-do”, “should-do” and “could-do”, that is already a sign of underlying problems. (Stepanek 2005, 106-107.)

Scoping study is kind of a miniproject to clarify requirements and can already include some development to understand the expected level of productivity. It is best to organize the scoping study as a separate project and its outcome should be

- use cases or other requirement specifications,
- screen mock-ups,
- a high-level design,
- working code for a few key features,
- an acceptance test plan,
- a break-down of the work, with estimates and
- an iteration plan. (Stepanek 2005, 108.)

Lehtimäki’s (2008, 156-7) proposal for tackling the problem of a giant project is similar with Stepanek’s ideas. He says that if there is no other way to prevent a too long project to be started then it is best to divide it to several shorter projects. This way of working can be thought of as kind of a triage: the customer can get the most important features into use as the result of the first project. Also the risk is much smaller as it is divided into several projects. If the project is longer there is more time for the risks to appear.

In the very practical level most important is to choose some kind of practices for managing a software project and stick to them. According to Stellman and Greene (2006, 6) every project is much better of with the selected practice, no matter which one it is, than with no practise at all. This must be remembered when a project start to slip from the planned schedule and the project manager is tempted to use all the possible effort for creating code and leave out for example reviews and testing. Instead of making the project go faster, forgetting the defined practices will only make the project last longer.

2.3 Software Engineering

Software engineering is a young field that has its origin in computer science and programming. The term software engineering was first introduced in 1968 after a realization that software can only be produced successfully if there is a discipline guiding the everyday work. (Tsui & Karam 2007, 56-7.) One definition for software engineering is “the application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software, and the study of these approaches; that is, the application of engineering to software” (Guide to …, 2009).

Even though the most common way to refer to building of software is development, engineering describes the complexity of the tasks better. Software, and especially its outcome, has impact on the business and because of this it needs not just development, but engineering. Thus engineering refers to more holistic understanding of what is needed for building software: not only the knowledge of technical concepts or programming, but also some business understanding. Software engineering is also about having a sensible and smart approach to principles and standards which have been already realized and worked upon, to give not the only the quickest but the most efficient outcomes for the desired results.

What makes software engineering different from other fields of engineering, is again the characteristics of software. According to Pollice (2005) the main difference between software and physical devices is that software is intangible and designed to be changed. In addition there are only few, if any, laws that can be universally applied to software. Electronic engineers know that there are laws of physics that they can follow, but such are not available for software engineers. Even though a particular program such as an operating system may be copied and released to millions of customers, it is only making a copy, not building another identical product. Thus software is not mass produced, unlike for example cars. One more big difference is the requirements and specifications: for software they can change in the very late phase of the development but not in bridge building, for example.

Several sets of principles have been created for software engineering. Tsui and Karam (2007, 63-8) have presented principles created by Alan Davis[2], Walker Royce[3] and Anthony Wasserman[4]. Below are the most relevant ones for this thesis from their sets.

- Use an appropriate process model.
- Good management is more important than good technology.
- People are the key to success.
- Follow with care (the decisions made about tools, processes, methodology etc.).
- Establish the process for object quality control and project progress assessment that includes the assessment of all the intermediate artifacts.
- Lifecycle and process: most of the large, complex software projects have not survived without some defined process.

There is also Software Engineering Code of Ethics and Professional Practice which describes eight principles related to the behavior of and decisions made by professional software engineers. The idea behind the principles is that "software engineers must commit themselves to making software engineering a beneficial and respected profession". (Association for … 1999.) Each of the eight principles is divided to several sub-principles. The most relevant of those for this thesis are:

- Public - software engineers shall act consistently with the public interest.
    - Accept full responsibility for their own work.
- Client and employer - software engineers shall act in a manner that is in the best interests of their client and employer consistent with the public interest.
    - Identify, document, collect evidence and report to the client or the employer promptly if, in their opinion, a project is likely to fail, to prove too expensive, to violate intellectual property law, or otherwise to be problematic.
- Product - software engineers shall ensure that their products and related modifications meet the highest professional standards possible.

---

[2] Original source: Davis, A. M. 1994 (November). Fifteen Principles of Software Engineering. IEEE Software 94-101.

[3] Original source: Royce, W. 1998. Software Project Management a Unified Framework. Reading, MA: Addison-Wesley.

[4] Original source: Wasserman, A.I. 1996 (November). Toward a Discipline of Software Engineering. IEEE Software: 23-33.

  - Ensure an appropriate method is used for any project on which they work or propose to work. Ensure that specifications for software on which they work have been well documented, satisfy the users' requirements and have the appropriate approvals. Ensure adequate testing, debugging, and review of software and related documents on which they work. Ensure adequate documentation, including significant problems discovered and solutions adopted, for any project on which they work.
- Judgment - software engineers shall maintain integrity and independence in their professional judgment.
  - Only endorse documents either prepared under their supervision or within their areas of competence and with which they are in agreement.
- Management - software engineering managers and leaders shall subscribe to and promote an ethical approach to the management of software development and maintenance.
  - Ensure good management for any project on which they work, including effective procedures for promotion of quality and reduction of risk.
- Profession - software engineers shall advance the integrity and reputation of the profession consistent with the public interest.
  - Take responsibility for detecting, correcting, and reporting errors in software and associated documents on which they work.
- Colleagues - software engineers shall be fair to and supportive of their colleagues.
  - In situations outside of their own areas of competence, call upon the opinions of other professionals who have competence in that area.
- Self - software engineers shall participate in lifelong learning regarding the practice of their profession and shall promote an ethical approach to the practice of the profession.
  - Improve their understanding of the software and related documents on which they work and of the environment in which they will be used.

2.4 Developing Software with the Help of Processes, Project Management and Engineering Disciplines

The organization doing software development faces many difficult decisions when it wants to or has to change the ways of working it has been using. To help with the decision there are lots of different processes and ways to manage projects available. The engineering principles could be used as starting point when finding a way through the jungle of all the options. It is also good to remember the basics: process describes the tasks and roles and responsibilities, and project management is for planning and making decisions. Managing a project is not enough, but the process is also needed to understand the tasks and persons to be managed.

In addition to the options available, there are several factors that effect the decision what kind of process and project management procedures to use for developing software. As software is different from other artifacts it is not possible to have one right solution even for one specific case, as it is not only the needed application and its features that count but also for example the organization that will use the application, the team that will do the developing and the resources that are available. These are only to mention few. The situation needs to be considered very carefully, but once a decision is made it is best to stick to chosen way of working for long enough to see if the decision was the right and to plan the changes needed based on the feedback.

3 Information and Communication Technology Company in the Change

I formulated the first proposal for the subject of this research in autumn 2007. The main idea was to create a study of the first release that integrated the different PDM systems in the merged company. However, the work was very intensive and the release was ready before I was able to start thinking about the study. During the year 2008 there were several ideas about the exact target of the study but in all ideas there was always something in common: how to find new and common ways of working for the PDM development team.

In the end of year 2008 I decided to interview my colleagues to gather material for the research before starting my maternity leave. I had selected the themes for the interviews based on my own experiences as a project manager in the PDM development releases. In autumn 2009 it was decided with my supervisor that the main focus will be in developing ways of working for multi-site organization, both line organization and project work included. Also the change management is emphasized: how to make sure that the constant changes can be managed and especially how to do it in practice.

3.1 Target and Objectives of the Research

The efficiency of an organization is usually evaluated by its results. In the PDM application development organization that was studied for this thesis the results are the modifications of the legacy PDM systems. From the result point of view the organization is quite efficient: since the merger there have been several successful releases and the legacy systems have been integrated to support the business needs of a merged company.

What is interesting in a situation like this, is the people and ways of working that are behind the successful releases. I have worked as IT project manager in the releases and based on my own experiences it is exactly the persons that have made the releases

possible, not the common ways of working. Quite contrary, the common ways of working are still missing in the PDM organization. How is it then possible to create something new and cooperate with new colleagues when there is lot of changes happening all the time? And on the other hand, how do the persons themselves feel about the situation and what kind of proposals do they have for future ways of working?

To answer these questions my aim is to describe the situation and circumstances in the PDM organization of the ICT company about 20 months after the merger. Both my own experiences and the experiences of the three interviewees are used for the description. Based on the description I will create guidelines for new ways of working to enable efficient team work in this multi-site organization. The guidelines are created more from the management than technical point of view and especially for this organization in its current situation. The target is not to develop a comprehensive guidebook that can be used in any organization but to concentrate on the issues that could help the persons in the PDM organization to make the work more effective and also more enjoyable.

The research questions for the thesis can be summarized in the following way:

1) How have the persons working with PDM application development experienced the merger related changes happening in their organization?
2) What are the ways of working that enable efficient teamwork in a multi-site organization?

3.2 Description of the Target of the Research

Two information and communication technology companies merged into one in April 2007. As always after a merger, there were two or in some cases several different kind of business processes and applications used for one purpose. Since the merger took place, one of the targets both in business side and in IT department has been to achieve a situation where there is only one process and one application for one purpose. This of course is a huge change to the personnel of the new company, both to the ones using the processes and the applications and to the ones developing them.

In PDM area the situation was quite challenging at the time of the merger: there were five different applications that were used for managing the product data and each of these had its own development team and the teams had their own ways of working in the IT organization. There were about 50 persons working in the organization. The first target in the PDM area was to integrate the legacy applications as much as needed to be able to send harmonized data to company's ERP (Enterprise Resource Planning) system and to find common ways of working for all the development teams.

PDM application is one of the critical applications in the company and because of this it must be as reliable as possible. In the ICT company the term PDM is used in the broad sense and thus it can be understood as PLM (product lifecycle management), which is usually seen as the next step from PDM. The criticality of the PDM system can be understood for example from the PLM definition created by Grieves (2006, 39):

> PLM is an integrated, information-driven approach comprised of people, processes/practices, and technology to all aspects of a product's life, from its design through manufacture, deployment and maintenance – culminating in the product's removal from service and final disposal. By trading product information for wasted time, energy, and material across the entire organization and into the supply chain, PLM drives the next generation of lean thinking.

In the ICT company PDM application is used for creating and maintaining the master data of all the products. In addition to the data needed in the factories there is also some other data that can be classified as master data. According to Saaksvuori and Immonen (2005, 49) the product structure is the heart of the PLM system and it provides the foundation for some of the basic functions of the PLM system. The objects in product structure are data elements and they have different dependencies in relation to each other, see figure 3 below.

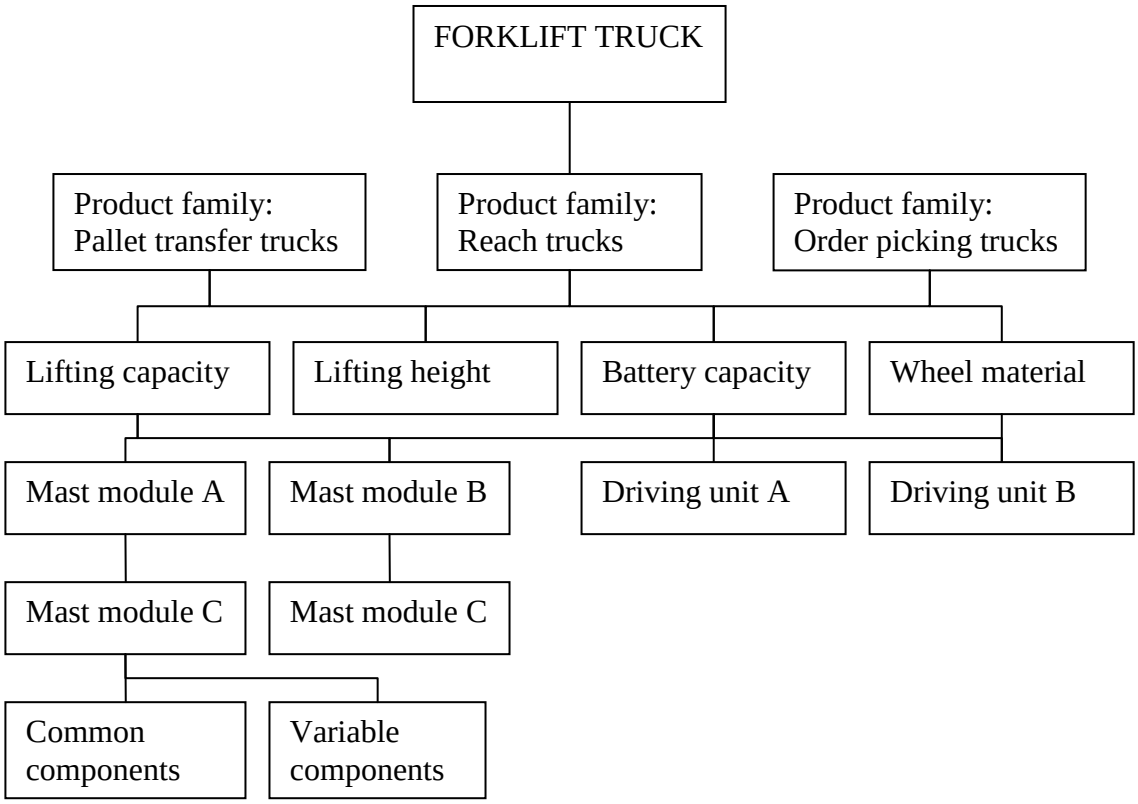

Figure 3. Example of the generic product structure (product model) of a customizable product. (Saaksvuori & Immonen 2005, 55)

Most of the data managed in the PDM application is needed in other applications used in the ICT company and some also in other companies, for example by subcontractors. According to Saaksvuori and Immonen (2004, 13-4) the task of PLM is to provide necessary conditions for connecting separate information data systems, processes and automation islets. PLM should also be the starting point for integrated totalities by commanding a wide variety of information systems. Like seen in the figure 4 below, the PLM system is located in the middle of a large and quite complicated environment.

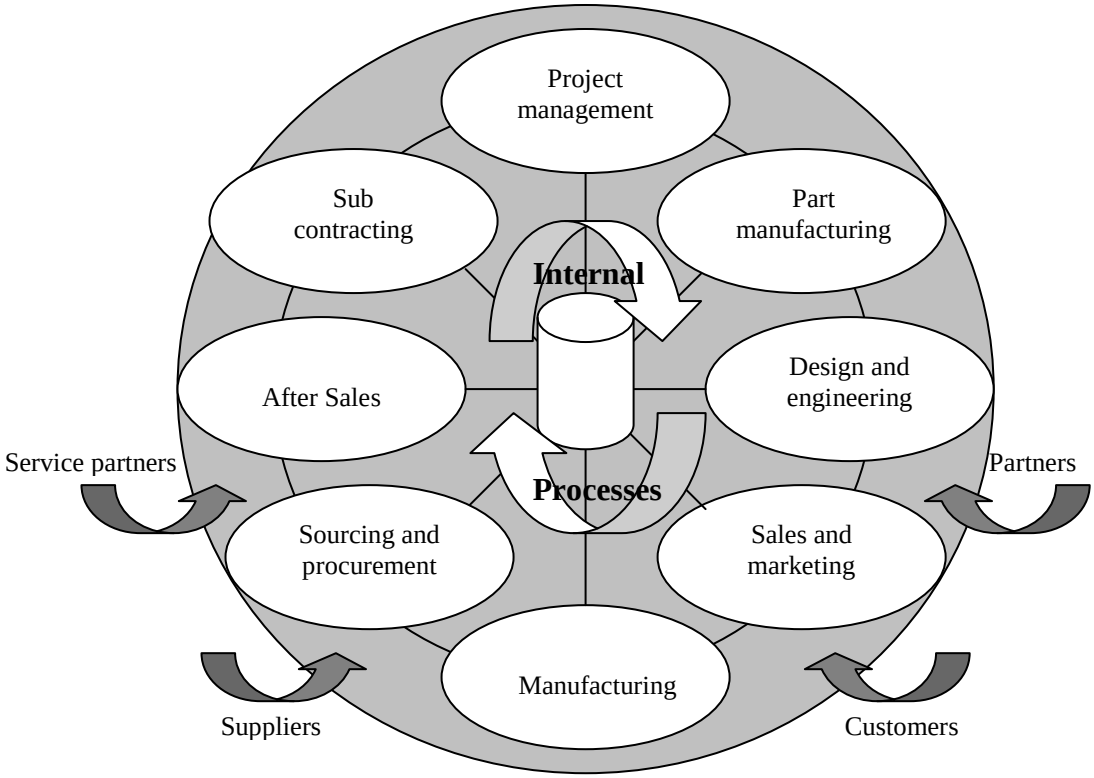

Figure 4. PLM system's relation to processes involved in the creation, recording, updating, distribution, utilization and retrieval of information. (Saaksvuori & Immonen 2005, 15)

During the two and half years after the merger the five PDM applications have been integrated but they are all still used. The development teams of the applications have already worked together in several projects and most of the team members are in the same line organization, but still the ways of working are different. One of the main reasons behind this is that it is easy to continue with the old ways as also the application people work with is still the same. Erkkilä (2001, 184) also points out that the change and the work pressure give stress and in a stressful situation it is only human to subconsciously act according to old model. Also the fact that the persons belonging to the teams are located in different sites and in different countries does not make the cooperation and agreeing the new ways of working and taking them into use any easier.

The ultimate target in the PDM area is to introduce a new PDM application that will gradually replace all the currently used systems. In IT organization the objective is also to deploy new roles and responsibilities and ways of working to be used with the new application. New roles and responsibilities are needed not only because of the merger

but also because the new application will be an enterprise level software that will be mainly configured according to the needs of the business. Configuration will be done by another company and the persons in the IT organization of the ICT company will have less to do and different tasks compared to the earlier situation. This is a huge change to the current situation as all the legacy systems have been strongly modified, and for example one of them has a user interface that has been built in-house.

According to Tsui and Karam (2007, 55) established enterprise software products are bought and outsourcing is used to reduce the risk of software project failure. However, even if the risk of technical failure is reduced, there are the following issues that need to be taken into account when planning the project:

- executive commitments and leadership,
- thorough planning of both business and technical processes,
- skilled and experienced consultants,
- relentless management focus and monitoring the project and
- willingness to change and make adjustments when required (Tsui and Karam, 2007, 55).

These are all issues that need to be thought of when planning the new organization and the ways of working for it in the ICT company.

The development of the legacy PDM systems has been done in projects but there is a top level project called program which incorporates the projects. The situation regarding the number of applications and processes has been similar in all areas in the ICT company and thus in the program level there are also other programs that act as umbrellas for projects of other areas, like ERP. To clarify how program and project management is related to other development management activities in a company, Forselius et al. (2008, 13) have created a five level picture (see figure 5.).

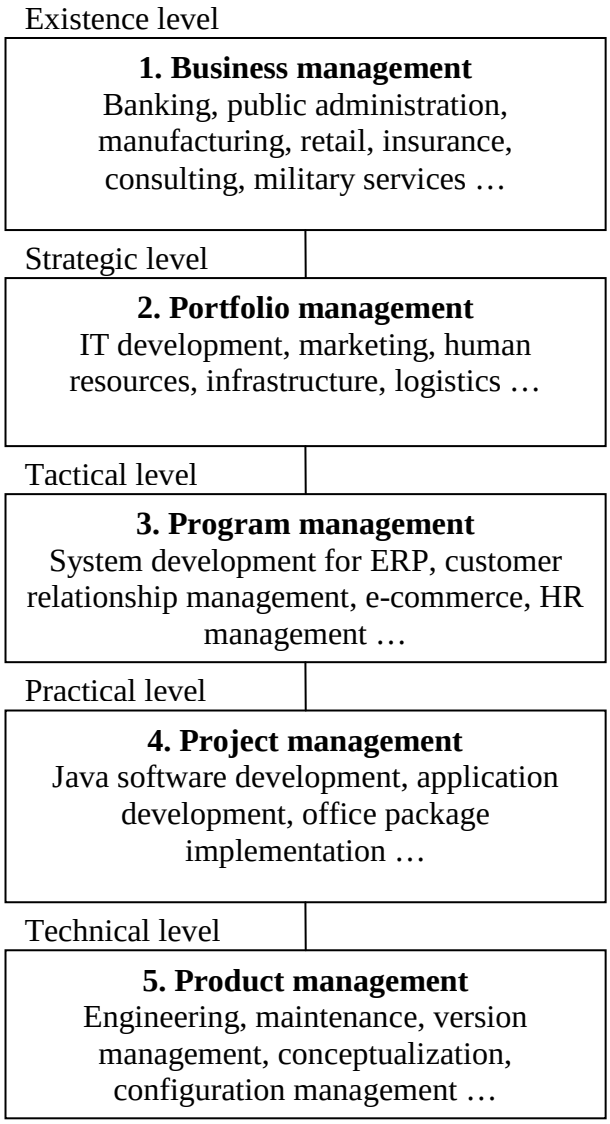

Figure 5. Main levels of business development-related management (Forselius et al. 2008, 13)

In this thesis the main focus is on the practical level, project management. Also the program management, which is in the tactical level, is touched, as the interviewees often refer to the program in their speech. Forselius et al. (2008, 14) remind that even though the persons working in projects and the project managers see mainly the management of the project and possibly the program, they need to remember that their work has always a connection to the higher levels and to the development portfolio.

The connection to surrounding organizations is probably something that is easily forgotten in the everyday work, even though the connection might explain the decisions

that feel strange in the project level. Because of its nature as an integrating system PDM application and the way it is developed is often affected by decisions made in other organizations. This is something the development team needs to understand and take into account when planning and doing their own work. So in addition to all the changes brought by the merger the people in the PDM organization face also many changes that are directly related to the development of the application.

3.3 Research Materials

In this research I will use the experiences of the persons who are affected by a change in their organization and the ways of working, and the meanings they have given to the incidents related to the change. To be able to do this I collected the following materials:

- interviews of the persons affected by the merger in the PDM application development team,
- my own experiences and
- interview of a person who works in a similar organization in another company.

I interviewed the persons working in the merged company on 8th, 9th and 10th December 2008 and the person of the benchmarked organization on 25th November 2008. I organized all the interviews so that they took place during the working hours in the company premises. The person of the benchmarked organization was interviewed in his own office. I recorded all the interviews. The duration of the interviews varied from one hour to two hours, depending on the interviewees' willingness to tell about the issues I had selected as themes for the interviews.

Two of the three persons working in the merged company had the same line manager at the time of the interviews. The interviewees, two men and one woman, worked as systems analysts and system designer. These were their new titles but their tasks varied quite a lot as they were all still working with their old responsibilities while learning the new ones. They also have quite different educational background. The person in the benchmarked organization worked as system designer during the project the interview focused on.

According to Anttila (2005, 195-6) interview is the correct way to collect information when there is a need for opinions, attitudes, experiences and observations. Interviews can be classified based on the distance between the interviewer and interviewee that is formed with the type of the interview: an interview can be a strictly structured with clearly defined questions and even alternatives for answers, or on the other hand a very open discussion with no prepared questions or guidelines. To be able to get to the bottom of the way the interviewees felt about the situation I needed more than just short answers to prepared questions. On the other hand, I wanted to guide the interview to the selected directions.

Semi-structured interview is selected when the interviewees are familiar with the subject. The themes for the interview can be created based on the experiences of the researcher. With the themes and the planned questions the interviewer focuses the discussion to the selected areas of interest, but s/he is still free to include new questions based on the answers. The interview gives the respondents a possibility to talk about the subject in detail and in depth with only some guidance from the interviewer. (Anttila 2005, 197, Livesey) For this thesis I selected the semi-structured interview for two reasons: I was able to create the themes and questions for the interviews based on my own experiences and the interviewees were all very familiar with the themes. However, as they all work in different areas, the semi-structured interview gave me the opportunity to explore the relevant areas in more detail with each person. All the themes were discussed with all respondents but the discussion was focused on the areas the respondent was most familiar with.

Semi-structured interviews have also some limitations. The result can be affected by the skills of the interviewer, especially the ability to create new questions during the interview. To avoid this problem a good interviewer needs to be a good listener and able to build positive rapport so that s/he can clarify and understand the respondent's point of view as well as possible. The interviews are difficult to repeat and thus the reliability can be questioned. Generalisations are difficult to do because of the personal nature of the data and it can be hard to decide what is relevant. (Anttila 2005, 200, Livesey)

I have used semi-structured interviews in my previous research and have thus some experience in doing them. As I have also experience in the field of the study it was quite

easy for me to follow the speech of the interviewees and understand the background for their comments. However, sometimes I felt like I was doing assumptions in my mind and I tried to clarify these with detailed questions. The aim of this research is not to create a general description of working in the field of application development, but to describe the unique circumstances in the certain point of time in the selected organization. Selecting the relevant parts of the interviews for the study was quite cumbersome, but using the theoretical part as the background I was able to combine the parts of each interview and to form the most meaningful areas.

When starting the interviews I stressed the fact that the names of the interviewees will not be included in the thesis and even the name of the company will be left out. However, as several of my colleagues knew about the thesis I also mentioned that it might be possible for some of them to figure out who has been interviewed. I also promised to give the interviewees a possibility to read the work before it is published so that they can ask me to modify the text if they feel they can be recognized from it. With this I hoped to get as open discussion and opinions as possible.

The interviews gave the respondents a possibility to go through their experiences of the past year. Even though the situation was quite familiar to all of us, the opportunity to discuss it deeply gave both me and the interviewees new insight to where we were at that point of time. Using the gathered material to view and criticize the current situation and then rethinking the whole situation in order to find a new framework can be called reflection-in-action. The reflections give new ideas on how to solve the problems with new points of view. (Anttila 2005, 90.)

The interviews were based on four themes with more detailed questions (see the list of all questions in appendix 1 and 2):

- outsourcing,
- roles and responsibilities,
- project management and
- change management.

The themes were selected based on discussions with several colleagues and my own experiences about the areas that needed development. At the time of the interviews the

idea about the target of the research was a bit different: I was supposed to study how to organize the work in the situation where tasks previously done in-house will be outsourced. Because of this some of the themes and questions for the interviews are not as on the point as they could be, especially with the person from the benchmarked company. Still I was able to use parts of the interview in this thesis.

3.4 Research Strategy and Analysis Methods

Based on the target of the research and the materials used, the first part of this research can be defined as a case study. Usually case studies focus more on systems of action than an individual or group of individuals (Tellis 1997b). This case describes how the group of people working in one location of the ICT company and doing PDM software development experienced the situation in December 2008.

Based on Tellis (1997b) case studies cover one or two issues that are fundamental to understanding the system being examined. In this research the experiences of one group are used to create recommendations for the whole organisation. Enlarging the study to cover several sites even in several countries would have given more depth to the description, but this was not possible for practical reasons. However, the persons forming the group were already familiar with multi-site work before the merger, and this gives the possibility to find the factors that have changed as a result of the merger in the multi-site work. The persons also work in different positions and because of this their points of view are different, giving more comprehensive picture of the situation.

According to Tellis (1997a) single cases are used to represent a unique case and they are also ideal for revelatory cases where an observer may have access to a phenomenon that was previously inaccessible. The group of people I have studied have been working together for some years and they have also worked in multi-site organisation before the merger. It is the merger that creates the uniqueness of the case.

Case study is more about explaining than interpretation. The aim is not to generalize the results of the study but to give a description of an interesting case. This description can be used to recognize relevant factors, processes and interaction which can be then used as materials for further studies. (Anttila 2005, 286-8) In this thesis I have already

continued further from describing the case and used the results of the case study to recognize the areas and ways of working where changes are needed in order to achieve effectiveness in the multi-site team work.

I have also used benchmarking to find out how managing the multi-site work has been handled in another ICT company. Benchmarking is a method used to find the best practices and solutions that can be adapted to solve the problems in own organization. The way benchmarking has been used in this thesis can be called functional: the objective was to find the best practices from the team that works in the same field. (Karjalainen 2002.)

Even though the other company had not experienced a merger, there were enough similarities in their situation so that the learnings from one of their projects could be used as a part of this research: the target of the project was to develop PDM application and the project personnel was located on several sites in different countries. The project was not a perfect success as such and thus the whole project cannot be used an example to learn from, but the experiences gained were important for creating the suggestions in this research.

There are at least three different ways to do a case study: either to start with theory and an analytic strategy that will lead to conclusions, to develop a case description, which will be a framework for organizing the case study or to do pattern matching (Tellis 1997b). In this thesis I developed a case description and used that for organizing the whole study.

According to Tellis (1997a) “the analysis of case study is one of the least developed aspects of the case study methodology. The researcher needs to rely on experience and the literature to present the evidence in various ways, using various interpretations.” I started the analysis by transcribing the interviews in January 2009. Transcripts can be either naturalized or denaturalized depending on the purpose of the interview (Oliver, Serovich & Mason 2005). The transcripts in this research are denaturalized in the sense that in addition to the words said I did not include any other voices made by the interviewees or any pauses or alike. However, the transcriptions are natural in the sense that I wrote down the exact wordings without standardizing the language used. My main

target was to find out the meanings and perceptions created and shared during the interview.

For analysis method I selected thematic analysis. Thematic analysis can be either theory or material based. When used with semi-structured interviews the material based thematic analysis can reveal new themes in addition to the ones which were selected for the interview. To allow this to happen, the interviews must be analyzed without any prejudices. (Saaranen-Kauppinen & Puusniekka 2006). According to Aronson (1994), the thematic analysis based on the material consists of the following steps:

- classify the main patterns that emerge from the material,
- identify all data that relate to the already classified patterns,
- combine and catalogue related patterns into sub-themes,
- build a valid argument for choosing the themes by reading the related literature and
- formulate theme statements to develop a story line.

I adapted this model a little and used it in the following way to analyze the interviews. I started with the interviews of the ICT company persons. I read through all the interviews several times, then text by text selected all the relevant parts and combined the related texts under one title. In other words I classified the main patterns of each interview separately. Next step was to go through the patterns of each interview to combine the related ones under one title. Once I had all the titles and the quotations from all the interviews in one document, I read this text again several times so that I was able to catalogue the themes under bigger headings, sub-themes.

As I also had the benchmarking interview material, I moved on to that one at this point to see if I could find comments for and against the opinions that had come up in the interviews of the ICT company persons. At this point I had a document that included the most important themes in the interviews, and I was able to trace back all the quotations the themes were built upon.

In addition to the main themes I had used to guide the interviews, I identified the following themes from the material:

- cultural differences,

- management, both in line organisation and in projects,
- building teams and
- working in a multi-site organization.

Next step was to go through the literature and build the final statements that combine the valid parts from the interviews and the theory. When writing the statements I had to translate the quotations from the interviews to English. Because of this the quotations are not word-for-word exact. I gave the interviewees a possibility to read through the quotations so that they could give me comments if they feel that I have not understood them correctly or that they are too easily recognizable. However, I did not receive any comments.

I did not try to create a story about the statements, but I built an analysis chapter that consists of the main themes: the change, line and project organization, leadership and management and managing the information flows. Under these themes I have placed the issues that are mainly related to each of the themes. It was difficult to find a correct place for some of the issues as all the main themes are quite closely related: for example managing the change is partly about managing the information flows and leadership and management is done in line and project organization. However, I tried to find solutions that make the result as understandable and coherent as possible.

Once I had the case study results I created the final result for this research: the suggestions for new ways of working. The suggestions are based both on the theoretical part of this research and the experiences of the interviewees. The experiences were used to recognize the areas where improvement is needed but also some of the suggestions come from the interviewees. In the last chapter I analyze the success of this research and give some ideas for further studies in this area.

4 Coping with the Changes and Proposing Improvements for the Future

The change has touched all the persons from both merging companies in many ways. As the time has passed it has become obvious that building one new company takes a lot of time, probably more than any of us working in the company was able to predict and some of us are ready to admit even now. Different kinds of events have been organized to advance the adaptation to the changes, but still there are many issues that need to be solved before it is possible to talk about one company with one set of ways of working, even in one quite small organization.

In this chapter my target is to describe how the interviewees have experienced the changes, the change management and working in the new PDM organization so far. The main point is to tell what kind of issues they have noticed in the current ways of working and what kind of proposals they have for improving the situation. I have built the chapter by selecting one area to each of the subchapters. The areas are a combination of the themes I had chosen for the interviews and the issues the interviewees brought up during the interviews.

4.1 The Change

4.1.1 Getting Tired but Not Willing to Give in

Even though the merger happened already 20 months before the interviews, the interviewees feel that the change, combining the two organizations, is only half done, if even that. They are getting tired of trying to get some clarity to the whole situation and fear that the same confusion might continue with the new tool if the change management is not handled properly.

> *In a way I feel that I have just given up. I can't be bothered anymore. If you have to use a lot of energy to try to, then you just are too tired to continue asking, well, I've tried some months now, but it was some time ago and it didn't change anything. Then you'll just get tired of it all.*

> *You could think that if this same continues with our new tool, or do we even have any reason to suspect that this would not continue? So this same continues even if we have the new tool and common ways of working and so on, I mean does it really change?*

Interviewees' feeling that there is still a lot to do to really have the two companies combined is supported by Erkkilä (2001, 180-1). She says that integrating two large companies takes in the active level from 12 to 18 months and in the mental level much longer, even several years.

According to Erkkilä (2001, 196) the saying the devil is in the details applies very well to planning and implementation of mergers. The issues that are faced during a merger and require leadership cannot be considered as exceptional compared to any other situation, but it is the number of issues, the speed at which the decisions must be made, the outbursts of feelings and the different company cultures that make the mergers exceptionally challenging situation to handle.

In the studied organization the development of the current tools has been successful so far only due to the people who feel that they need to take care of their old responsibilities if they see that no one else is doing it. The prevailing characteristic of working is hero culture; individuals are doing the results, but common quality practices are missing.

> *And people are doing like, ok, we have done these before and this should now be taken care of, and then everyone is trying to do her/his best, and then we have achieved a tolerable result. If we worked with the attitude that ok, this has not been agreed with me, this is not my business, we would not have any releases. The outcome would not be useful. Of course people need to take care of, but at which point the situation should be checked …*

The reason for continuing to do the old tasks is that the interviewees feel that the business would suffer if they did not do this. And even if they are tired of the situation, they feel that they would be ashamed if no one did their old tasks and this would lead to problems.

> *But then again, my understanding is that if we would do some things better also the business would get more benefit. This is partly why I don't want to let things just go, if I hear something. … And then we don't need to be ashamed of what we are doing.*

Mattila (2008, 69) says that a change situation easily doubles the work load of an expert. A way to manage the situation and to handle the stress there must clear priorities and on the other hand limits. However, following the priorities and limits is more easily said than done. Mattila (2008, 69-70) proposes using a table where tasks are defined as ‘important but not urgent’, ‘important and urgent’, ’not important but urgent’ and ’not important and not urgent’. In a change situation it is especially important to cut down the number of tasks that are not important but urgent. Knowing the priorities of the organization and also own tasks is the prerequisite to success in these tasks.

In the current situation people are nominated to new roles, but they still work with their old responsibilities and struggle to cope with combining the old tasks and trying to learn the new ones. This can also be seen in the way decisions are done: even if there is the official new organization and the decisions should be done according to agreed rules, some decisions are done quietly by the old managers who still work together in the background, in a ghost organization.

> *When the organization is turned upside down, the old organization continues to work like it used to. Then there is a new person from another organization who is left in a kind of dead position in the sense that the old organization is working around her/him, and then the new person who has responsibility cannot affect anything done in the organization. Then s/he can be used as a scapegoat when needed.*

Erkkilä (2001, 64) points out that if there is discussion about values and agreed targets, but these do not actualize in the real life, meaning that the talk and acts do not match, there is insecurity in the organization. Trust and appreciation for the management diminishes, there is less commitment and it is slower to implement any changes or objectives. In the worst case there is political game going on in the organization.

It seems that the persons interviewed have the qualities that are needed in a successful organization: they want to make sure that the needed results are achieved even if it means that they do tasks that are not directly assigned to them and they understand that the main reason for doing anything in the IT organization is the success of the business. However, in a change situation these persons are easily stuck with their old tasks and responsibilities and they are overloaded when also trying to learn the new tasks.

4.1.2 Different Cultures Require Different Kind of Change Management

What makes the merger and the change even more challenging is the different cultural backgrounds of the companies that were merged. This can be seen both in the high level discussions and decisions and in the everyday work. The interviewees have noticed it for example in what kind of leadership the colleagues from the other company are used to: the way of working has been more hierarchical and top-down, with less independence and responsibilities for the individuals than the interviewees are used to.

> *Part of it naturally comes from the nature, but in my opinion also part comes from the cultural difference. I mean if you have worked in the way that the manager tells you, and now s/he doesn't, then it just doesn't change. And then the others have had the way of working where the manager has never told you exactly but you have just somehow discussed, then you can't just leave things as they are.*

According to Erkkilä (2001, 49) the hierarchy and the different ways how the responsibilities are defined and divided in an organization is strongly affected by the national culture. She continues by saying that it is important to understand that it is not about whose ways of working are the correct ones but about the differences in the ways of working.

Even if the time difference that needs to be taken into account when the persons from the merged companies cooperate is not big, it also affects the cooperation. Some kind of rules would be needed for finding the most suitable times for meetings. People would need to be flexible, and the flexibility is needed in all the teams and in all team members.

> *Well, we tried to discuss this in a small group and the result was quite sad. There are very few hours when we all are working. … And of course we can be flexible, and probably we can start some times, this was a discussion, we can start at nine their time. But still the meetings are short, and the discussion is really needed.*

This also shows that people are resisting the change. Even if it is only the lunch times that might change, people want to continue doing everything in the same way they are used to. It is not enough to have overall discussions about the change management but

people need help in agreeing the changes in everyday issues, the ones that affect their working days and habits immediately and in very practical level. It is easy to say that everyone must be flexible but really to agree how to do it in practice is not easy if people feel that change is not making their life better.

Erkkilä (2001, 189) however points out that resisting change is not only a bad thing. If there is no questioning, doubt and different views from the personnel, it is most probably a sign of mistrust towards the management. Change resistance is a normal part of change, learning and gaining the trust. In an open management culture change resistance is hoped for as it is a good way to hear the real feelings and views of the personnel. Mattila (2008, 55) agrees with these ideas by saying that taking the change resistance into account helps the change to succeed: if it is not possible for the personnel to tell when and why they are dissatisfied, the actions to fix the problems are started too late. Listening to the criticism is especially important in an expert organization. Persons in expert roles are usually critical of the management as they have ability to evaluate and criticize the business management in their own organization (Mattila 2008, 16).

4.1.3 How to Manage the Change?

The general feeling is that the change management done so far has not been sufficient. Lot of changes have happened and are still happening both in the line organization and in the program, but the discussion and agreements in the detailed level have not taken place. This can be seen both in the ways of working and in the roles and responsibilities. It is expected that the persons just start cooperating successfully without any practical help and guidance from the management.

> *So two, quite big companies, were combined, with totally different cultural backgrounds. We have had some culture type of discussions, like "what did each of the countries do in the war" and. But the real discussion, like how have you worked before and how have we worked and how could this be, like, we could write down what each of us is expecting. That hasn't been done at all. New fine combined teams are just created where there are persons from both companies and then the manager is from either one, and there is no discussion, like.*
>
> *The starting point is that no change management has been done in the program and still everything has changed a lot. And the result is that*

> *everyone has just found whatever place and there has been very little comments for these.*

Erkkilä (2001, 45) has remarked that organizational culture can also be the common ways of working that guide the everyday work and have been formed unconsciously over time. Organizational culture can also be referred to as "the way we do the things around here" (Harris, Moran & Moran 2004, 83). The unspoken agreements and assumptions are also the basis for the trust in an organization, not the written documents and information shared in briefings. It is even said that only with the trust born with the unspoken agreements makes it possible for big companies to exist. (Mattila 2008, 19)

Based on my own experiences also the line managers and program team leaders expected to have more support from their own managers to cope with the change. It is quite normal in situations where two big companies are combined that the higher management starts already planning the new strategies and next moves even if there is still a lot to do to clarify the new organizational structures and ways of working in the level that is required for the everyday work to run smoothly. It would be important for the higher management to participate also this work. (Erkkilä 2001, 183.)

The interviewees have noticed that even though there are people working in the same organization, the common goal is still missing and sometimes people feel left alone with the open issues the new situation has brought. Everyone is still doing the old tasks and without sharing the current knowledge and understanding each team member has, it is not possible to really start cooperating.

> *Like, now we have an organization where everyone is working in her/his own area. The information should be such, information should move well in the modular, make it such that everyone would know as well as possible how we can move forward.*

It seems that from the interviewees point of view discussions are the key both to the common understanding and to the change. All the team members, both team leaders and team members, need to discuss to have a common language and common understanding of the notion of PDM and all the related concepts. Only when this is clarified, it is possible get started with creating the one organization everyone can feel oneself comfortable in.

> *I've thought that one option might be to start having these weekly meetings on IT level. Just really to start going through things. … we have quite*

> *different understanding even about some very basic issues. And it will mean that the beginning will be a pure conflict when we really start working together. And because of that I would like to say that for the concept, for reaching a consensus and making it work, for that the change management is quite crucial.*
>
> *But yes, I would like doing things together and discussing. Now it's all about yourself, how much you can manage, have strength to, know how to, are capable of and can be bothered with to take care of things and clarify them.*

In addition to the discussion, the interviewees have also thought of other ways for reaching a common understanding. It is also important to agree the ways of working and to have the feeling that we are doing this together and working towards the same target. Even the word force is used by one of the interviewees to stress the fact that it is not even enough to agree something together but also to do the follow up, to make sure that the change is really happening and to the right direction.

> *Mostly, well, really to force people to work in the agreed way and with common concepts.*
>
> *Well, we should like have a drive to do it and then these, to have a common understanding with the responsible persons that we pull together. In this way we would have more drive in advancing things.*

The change management must be done in practical level and work must continue as long as needed. The wanted changes do not happen by themselves, and it is not enough to just tell a person the name of her/his role, for example.

> *I don't know, I mean it feels however you think about it, well I see that this will not start working just like that, this just doesn't get started. Like, either it goes so that the other side will just twiddle their thumbs and the others are doing the tasks, or then there will be a huge war, or we can maybe have some persons changing sides so to say, but the separate sides remain and the fences remain. Or then we can maybe have some little areas that are ok, but other areas just go to hell.*

Grönroos (2004, 87) has found several reasons why the tacit knowledge, the knowledge that is beneath the surface, hidden and not clearly visible, cannot be transferred from one person to another. As one of them he has listed the lack of time. This is also what the interviewee refers to: it is only possible for a person to take on new responsibilities if there is someone guiding her/him. The official information can be found in the documentation, if such exists, but a large part of the information needed to handle the

tasks well is not written down on paper or cannot be explained to others in simplistic words.

> *The only chance there is, in my opinion, is to give the guy some time. But with guidance. It takes the time from two persons for some time, and. But otherwise we will not have any substitute persons and if someone falls ill, then we just don't have any.*

It has been understood that it is not possible to change everything at once and have successful cooperation in all the areas immediately. The problems need to be tackled one by one. A decision is needed where to start and then it is important to continue one step at a time, according to a plan.

> *And we should start from something a bit easier. Like, you can't solve everything at once, but start somewhere.*

This idea is supported by Pohjonen (2002, 23). He says that even if the effort to develop ways of working is a good thing, it can go terribly wrong. Trying to make a huge leap either from technological or philosophical point of view is especially dangerous as it is impossible to go from level N to level N+2 at one go. So the target is to develop each area with small enough steps. Otherwise it is possible that the benefits gained in one area are lost because of problems in other area.

Tsui and Karam (2007, 65) share Pohjonen's view and continue that it is important to follow with care the decisions made about tools, processes and methodology. Nothing should be adopted just because someone else is doing or using it and experiments are needed before making a major commitment.

4.2 Line and Project Organization

4.2.1 Roles and Responsibilities

The current situation of roles and responsibilities is quite unclear in the studied organization. The basic feeling the interviewees have is that there is still much to do: either it seems that the roles and responsibilities have not been defined at all or if there

are some nominations, for some reason the persons are not acting according to their new roles.

> *In the program we don't have any roles and responsibilities in the expert level. That's what this is all about. We have been kind of asking that where are the descriptions and there are none. Maybe that is the reason why it is like this, like.*

It has also been noticed that persons can be changed from one role to another in the program. These changes have not been handled properly, and the result is that the work that has already been done to define the roles and responsibilities is thrown away.

> *Well this is, I have now come to the other end, that program can change persons, and they have done it, occasionally. And then there are no roles in the end.*

The interviewees understand that they need to be active themselves to find their new place in the organization, but they also feel that there is a limit: you can only do so much on your own without any support and help from your managers. This causes problems in the whole organization or team as it is not only the person who is nominated to the new role who does not know what it means, but also the persons around her/him might have wrong expectations. This can lead to a situation where some tasks are not taken care of by anyone and on the other hand there are two persons working with the same tasks without knowing about each other.

> *Well, ok, I have this new title, but no one has, I probably should have actively found out myself what it means. - - It's like, what are all the things that are your own responsibility.*

> *Well, maybe the effect is that the persons who don't necessarily know what the role means. When they take it, understand that the nominated person is not acting according to her/his role, or the person see that the area is not working correctly and thinks that the nominated person should take care of it. Then the person starts doing something as s/he gets tired of watching the situation from aside. So the person can start doing more than s/he should, take more responsibilities.*

Based on McConnell's (2002, 283-84, 302) views as important as defining the roles and responsibilities clearly, is to make sure that the team members have understood both their own roles and responsibilities and also those of others. Team members also need to be ready to take tasks outside their own official responsibilities when needed. This is the only way a team can work effectively as there are no conflicts within the team and all the tasks are taken care of. Other characteristics that can be found in well functioning

teams are supervision of individuals' performance and feedback, effective communication and fact-based decision making.

The interviewees have a clear understanding what the roles and responsibilities should be like: clear enough so that there are no conflicts but not too detailed, as creating very detailed descriptions might be quite difficult. In a team of experts it might even be possible to work without any descriptions, but the definitions are needed for the situations when changes occur in the team as it is not possible for the new team member to understand what is expected from her/him if there is no documented role descriptions available.

> *… so specific that there are no conflicts. If the expert is not happy with the responsibilities, then they must be made clearer. Like how I see it, well ok, it's about personalities, if we have three persons working together they can find their roles and they don't need to be told what they are. They can manage it.*

> *It depends a lot on the area. If you know very little, then it would be better to know, to have it written down quite specifically, but the detailing takes quite a lot of time. I would start from writing the titles of the novel, kind of, and then tell how is doing this chapter and who is doing that. And then after a month or two we would check together if we have a better understanding. If it is, good. Then we'll just continue. And if someone starts doing too many things, which I quite often do, then s/he is guided to the right direction.*

When the roles and responsibilities are defined and the change management to take the new roles into use has been handled well, the people in the teams can easily work together with very little leading from their managers.

> *In the model that I know from another company, you have people who are responsible and the only task managers have is to make sure that people have it or that they have the possibility to work. When it in fact means roles and responsibilities and then the management needs to interfere only when, like when they need to, but in practice they don't need to do it at all.*

In the situation where the roles and responsibilities are defined clearly and everyone is aware of them, the whole team knows what each person is doing and what can be expected from the others. However, quite the opposite has been noticed in the PDM organization and there is even some personal experiences of situations, when the interviewee does not know what s/he can decide without her/his decisions being overruled afterwards.

> *Yes, in the way, because it's all about the fact that if a person has an expe... or role and responsibility, you can't just walk over the person. If someone*

> *does, then you can ask what is this role and responsibility, if you are not responsible after all.*
>
> *I have already learned to say that if I wouldn't do this, but in many cases I am not the person who decides. I continue like that, because I feel miserable if I say no, and then someone walks all over in a minute.*

Mattila (2008, 66) points out that it is good to give the coordination responsibilities also to the experts. This is a way to increase commitment and share the know-how in the organization.

However, reaching the best solution for roles and responsibilities and for the composition of the teams is not just about the decision what needs to be done and doing it, but also about the way people feel about the change. There is a feeling that the new organization with new roles can work with the existing people, but only if these persons are willing to accept the change and understand what it means in practise and work accordingly.

> *If we can define what each of us is doing, properly, and we have time and are willing to, on the other hand also willing to, think about what each of us is doing, then. - - It doesn't necessarily mean that we would need to change people but we need to understand what each of us is doing, then also this organization might work quite well.*

When using the line organization as a starting point for the project organization, there is always some continuity. However, when building a project team for the first time, selecting the correct persons is mentioned as one of the most important tasks for the project manager (Tsui and Karam, 2007, 64). But when building up line organization teams in an existing organization, it is not possible to select the persons but it is more important to know the strengths and development needs of the persons available.

According to the interviewees the roles and responsibilities should be defined in the line organization, and then the same definitions could then be used in the program. Line organization is seen as the best home for the definitions, as there are so many changes in the program and the definitions should be quite stable. Roles and responsibilities are related to the development discussions and those are also handled in the line organization.

> *Well, at least it's clear that it cannot be a release, as releases come and go, and there shouldn't be fixed roles in the releases. It could be the program with one condition: if the program creates the long term roles and also*

> *holds on to them. But there are so many changes in the program that it is difficult and typically it is the line organization where the development discussions and such take place, which support the role nominations. So the line should be the place. So if the clear roles can be found in the line organization and the line can create them, then. The person would then have the same role automatically in the program.*

This idea is supported by McConnell (2002, 292-93) who points out that it is a good idea to build a project team with continuity in mind, as this is the only way to guarantee effectiveness of several projects. It is easier to start a project and it is less probable that there are changes in the project organization when you already have a good team in place.

In additions to defining the roles and responsbilities in the IT organization, it is also important to have the definitions existing with the business organization. Having different understanding of who is responsible for what can lead to situations where no one is responsible and important decisions are missing or they come too late. This can even lead to delays in the projects, if there is for example overlapping testing going on in the same database and the proper coordination of the activities is missing.

> *This is it, from IT point of view the activities on the business side are quite invisible to us. Like when they are doing and what. And then accordingly it is sometimes unclear in the IT organization who is doing, how is doing and where, when is doing. Some coordination would be needed to make this work.*

There are similar experiences in the benchmarked company. The suggested cure is to decide the roles and responsibilities and ways of working, and then use these definitions unless a need for change is noticed. If something needs to be changed, then the change must be managed so that it is clear to everyone working in the project what is changed and who is affected. And also the follow up is important to make sure the change has really happened in the everyday work, not just on paper.

> *Like in the future the organization should be thought of quite carefully, and these roles and responsibilities and who reports to whom. And then really stick to the plan. Or if it is noticed that changes are needed, then it has to be done so that everyone is definitely aware of the change that is happening. And then works with that. So that everyone would work according to the specific model.*

4.2.2 Working with External Persons

Outsourcing is a practice where another company is hired to make software development. Having external persons in an in-house project led by an internal person is not outsourcing but staff augmentation. In outsourcing IT is the customer as it contracts for a service or a specific body of work and the supplier is responsible for the delivery as agreed. (Tayntor 2007, 359.) It seems that not only in practice but also in literature the term outsourcing is very often used instead of staff augmentation or out tasking.

In the studied case there are also external persons, both developers and testers, working with the PDM systems. They are augmented staff. There are the same roles and responsibilities related open issues with the externals as there are in the internal teams: it is not clear for all the internal persons what the external persons are supposed to do, and also vice versa. This causes misunderstandings and even problems, as it is possible that everyone thinks that someone else is doing a specific task when in reality there is no someone else and the task is not done at all.

> *I have been disappointed in cases, where we have had an external person in a project and the way s/he does it, is like this: I ask what have you tested, where does the information go and the answer is "I don't know". Like I feel that any of the internal persons I know couldn't do testing like that, like "I've been sending the data". "Where does it go?" "I don't know".*

When the internal persons find out that there are tasks no one has taken care of, it is quite laborous to start clarifying the case and doing what has not been done so far. The internal persons also feel that the tasks they are doing are invisible as in the planning documents it says that the external person is responsible for that specific area. The clarifications are not planned anywhere, but still it takes time of the internal persons and this time is away from the tasks that have been planned for them.

> *These certain data loadings are done after data refreshes in the test databases, and then I receive the question if the loadings have been done and what has been done. And for me to find out the status, as I haven't, if I had done it myself I would have probably informed when everything has been sent. But for me to clarify it now, either by checking from the database or by asking, well it's like, not quite as much, but almost as much work for me as if I had done it myself. And now I've like done nothing.*

According to McConnell (2002, 491) outsourcing usually brings both benefits and disadvantages with it: he says that outsourcing leads to loosing control and downsizes

company's own development capacity but on the other hand brings either faster development or smaller costs or both. From resource management point of view the biggest risk is that outsourcing leads to a situation where there is not enough expertise inside the company doing outsourcing. Then the company is at the mercy of the partner: partner can dictate what to do and not to do and there is no one to question them. Technical internal expertise is still needed to answer to the questions of the partner and to do testing.

The interviewees know that they are not aware of the content of the contracts that are created with the externals. This makes the daily work difficult as they feel that there are cases no one is taking care of, and no one makes the decision on who should be responsible for these cases. Still just leaving these cases aside is not an easy option, even if the interviewees knew that they would be allowed to do it.

> *Of course we can buy development from the consultants, but I feel that it, if it is the purpose for the external consultant to work in this specific area daily or continuously, it hasn't been working very well. I don't know if their, how it has been agreed or how it should be if we can't participate everything ourselves. Like these large areas that are left totally, like you don't have to take it, no one needs to take a stand on who is responsible for these. And still in most cases in the real life someone should say, or I feel it's really difficult, when someone asks me and then I would just leave it, like this is not my business, I'm not even trying to find out. Probably I'm allowed to do it, but it's difficult.*

The general understanding seems to be, especially in the higher management, that having an external person in the team is having an extra pair of hands. But software development is not so straight forward that a new person could just jump in and start doing what is needed. And even the experienced external persons often need help from the internal persons, if they are mainly doing the actual system development and still there are things that need to be clarified with the business persons. This is often not noticed by the persons doing the planning. However, it is the internal persons supporting the external persons who feel that they are not using their time efficiently if they are answering to many questions from the externals and the time used for this is not planned as an official task.

> *Well. I have experienced several times with these externals, some of them, that it's quite a burden, that they say yes, yes, we will do it and then there are quite many questions. And it's not, it's like one step, which is nowhere, even though you have said that it's not visible and it's not counted as work.*

> *It's not marked at all, well ok, give support half a day in the week for these persons.*

This is supported by McConnell (2002, 494) who says that when outsourcing software development most important task for the internal persons is the requirements specification. Only internal persons know how the company operates and what the current processes are. Careful requirements specification is the absolute condition for a fixed price contract: without clear requirements the offers are either really high or the partner giving the offer does not understand the risks included in the project. In the studied case the internal IT persons do not necessarily know themselves the processes but they are the ones to contact the correct persons and do the clarifications needed.

Even if McConnell says that the requirement specification is the most important task for internal persons when using outsourcing, also the supervision of the external persons is quite important. The motives of the external persons might be different from the internal persons' ones and this is something that needs to be taken into account when working with them and doing the follow up of their tasks.

> *And then the situation that we have, that this person is an external and in the fear of loosing his job he is doing the tasks of this other external person, as he is handing them over to him.*

In the benchmarked company it was noticed that all the internal project members need to know what has been agreed with the external company, especially about the responsibilities. They learned it after the whole management team of the project was changed and then there was no one in the project who would have known the content of the agreements in the level that is needed in the everyday work.

> *In practice all the four internal key persons left. - - Then the new product manager started and tried to get things going, but he was quite lost as he came from another area. - - It definitely should be so that every internal who is participating the project knows about the contracts, how they have been done and what are the responsibilities. It was quite unclear in our project.*

There are some critical factors that will make the outsourcing relationship to be a success:

- a clear understanding of each partner's goals,
- shared accountability,
- flexibility and

- the WIT (whatever it takes) factor (Tayntor 2007, 374-75).

In short, it is vital for both the customer and the supplier to understand the target in the same way and work together to achieve it. Compromises are always needed along the way, but this is not a problem as long as both parties are willing make them. As well as the supplier has the "whatever it takes" attitude for the actual development work, the customer needs to have the same for setting up the procedures and practicalities needed. This way the end users requirements can be met in good cooperation, not only as contractual partners. This is important for the future as the new PDM application development will be done using real outsourcing, not only staff augmentation.

4.2.3 Building the Teams

The unclarities in the roles and responsibilities have effect also on the team structures. It is difficult for the interviewees to understand whether the new organization is reflecting only the new application and related roles, or should the old system also be somehow still taken into account. And even if the organization was built only with the new application in mind, it is not possible to leave all the old tasks immediately but the change needs to be planned carefully. There has to be enough support from the managers so that the starting point for the restructuring is clear to everyone: how much time people are using for the old tasks and how much for the new ones.

> *Is the team created so that we, the persons who have similar tasks, are in the same box, do we continue roughly in the same way we have worked so far? Or do we think that this is now, like everyone has a new job description? And then, it should be quite clear also with the line manager that, like agreed, that if there are changes, what is expected from me.*

In Mattila's (2008, 82) opinion it is important for the persons to understand how the change affects her/him. In addition to knowing what is expected by the managers and the team it helps when one can clarify the following things: what am I expecting of the change and how can I make my objectives to come true, and is it only important for me to cope with the situation or do I see possibilities to develop myself and succeed.

The interviewees have noticed that it is not only the knowledge the people in the organization have that matters when building a team, but also the personalities of the team members affect the cooperation in the team. Having the same understanding of how to do the tasks and trusting the team members is sometimes more important than the knowhow, which can always be acquired. There is even an idea that the teams could be built so that persons who like to work together would be in one team.

> *And I don't know if we could, but it seems a good idea if we could look at the personal chemistry. Like we have clear evidence of cases where all the persons just can't work together and then there is a war. - - I would like to be in a situation where we can have the persons who like to work together in one team. And I would like to see where it leads. - - And maybe it is the attitude towards work that matters, not directly your knowledge or what you do, but the way you do it.*

When working in a team where the collegues are alike you, they can also more easily notice if you have issues you need help with. This applies not only to the exact tasks but also to the overall situation, for example with work load.

> *It's not, like, sometimes the colleagues can, colleagues who are similar enough can notice what is your situation.*

According to Mattila (2008, 16) a safe working community also helps in a difficult change situation. Team spirit and solidarity grow whom the feeling of reciprocal dependency. The trust that we feel towards our colleagues is born with the good spirit in working community and with the solidarity that is demonstrated in tough situations. Venkula (2005, 141) extends the understanding of trust by saying that the main point about trust is not knowing exactly what someone will do but trusting that s/he will do is fair and that s/he takes care of her/his responsibilities.

One of the interviewees knows that also psychologists can be asked to help in creating new teams. Again it is the personalities than the skills of the team members that is the most important factor when building the teams.

> *But I've heard that some managers in another company, they use psychologists when they start building a new team or there are substantive changes in a team. - - The psychologist creates an analysis of the team dynamics. And the idea is to find the possible problems and also some guidance for leading the team. And what kind of, how the roles should be like, and such.*

According to Pohjonen (2002, 52) there has to be task oriented, self-oriented and interaction oriented persons in a team to make the group functional. The psychologists could help in defining the personalities and this information could then be used to form the best possible groups from this point of view.

From system development point of view it is important to know the comprehensive objective of the team to be able to build an effective team. On high level the objective can be related either to problem solving, being creative or tactical performance. Problem solving is needed in fixing use phase software incidents, team needs to be creative when building a new product and tactical performance is crucial when developing a new software release. Based on this information you can build the team so that the most important characteristic is emphasized. (McConnell 2002, 300-1) In the studied case the characteristics needed are being creative and tactical performance. Even if the persons already exist, this is good to remember when organizing the work of the team.

The interviewees also have a proposal that the teams could be built from down to top, meaning that the people in the organization could define the team structures themselves. It is something that the interviewees have never done before, but this is quite natural as it is very seldom when you are in a situation that the whole organization needs to be renewed or restructured.

There are different team models that can be applied when building a team. As the models can only affect the internal activities of a team, the team can choose the model itself. Example of this kind of team is “hidden team”: the management is not interested in following the work of the team in a detailed level, they only need to know that the team is working to achieve the agreed objective. Other models are more or less bureaucratic with agreed roles and responsibilities and a leader, from SWAT (Special Weapons and Tactics) groups to sport teams. The purpose of the models is to offer different kind of ideas that can be combined and utilized when building a team, not to give exact instructions for building a specific team. (McConnell 2002, 304-13.)

Combining features of the business team, characteristics team and professional sports team would be most beneficial when building a team that works with out-of-the-box system in a larger program. A leader that communicates with and reports to the program

management is needed (business team), inside the team there needs to be several separate teams and each of them is responsible for a specific area (characteristics team) and the leader of the whole team is in a supporting role from the technical development point of view, her/his main task is to make sure there are no obstacles for the effective work of the other persons in the team (professional sports team). (McConnell 304-13.)

The target of starting from the bottom is that people would really discuss together and find the best solution also for building the organization in the same way they usually find solutions for the technical and system related issues. Still there is some doupt in the minds of the interviewees whether the structure of the teams could really be defined by the team members themselves, but they would like to try it.

> *I have said in unofficial discussions that I would like to have a situation where we can build the teams ourselves, to have in the same team the persons I want to work with. And I would really like to see what would happen.*

> *To kind of build the cooperation from bottom up. - - That way I think we could break down the fences. - - This is something that just came into my mind and when you think more about it, it's quite a good idea, but how to do it in practice, that's another story.*

McConnell (2005, 293-94) states that people want to work in almost any kind of a project when they like the persons they work with and they are even less likely to leave a company. The understanding that experts from different companies working together leads to good results not only from technical but also from organizational culture point of view is supported by Erkkilä (2001, 64, 186). She says that the sooner the people start really working together with concrete issues the better real merger and the creation of new common culture gets started. Working together also diminishes the stress and uncertainty.

In the program it is important to have expert teams in addition to the teams that form the hierarhcical organization. The hierarchical organization consists of area specific teams, like concepting, IT and training. To have an efficient organization, there has to be also the experts from different areas working together. For example when creating a new solution that requires changes both in the concept and system, and probably also training for the end users, the solution should be created by a team where the best experts for the solution in question from the hierarchical concept, IT and training teams

are members. The cooperation of the experts is also important with issues where comments are needed: there does not need to be specific team meetings but the experts need to know their counterparts so that they can contact them easily.

> *There are the teams, like bubbles. But this is only the hierarchical organization. Now the task of the team leaders is to create subteams of the experts that are in the bubbles. - - You draw lines from one bubble to another, and the line stops at the persons in the bubbles. Then these lines are different teams.*

Venkula (2005, 80-1) says that in an unstable environment the decision making should be as fast as possible and that this can achieved by creating real, flexible cells. These cells are then allowed to and able to make independent decisions. Still the most common way to organize work in a change situation is to add more hierarchy into the organization and thus stiffen the work still more.

4.3 Leadership and Management

4.3.1 Leading the people

According to Harris et al. (2004, 134) the successful leader acts as a fellow worker with their subordinates. S/he understands that cooperation in sharing ideas and insights is better than competition, it is even the key to the organization's survival, problem solving and growth. The leader needs to facilitate teamwork and ensure professional synergy in her/his organization.

Form interviewees' point of view most important in the leadership seems to be the right attitude: you are interested in your subordinates work and tasks and if you do not know enough, you are willing to learn. However, the line manager of the experts does not need to know, and possibly even cannot know all the details her/his subordinates are working with, but s/he needs to have general understanding so that s/he is able to discuss the work with the subordinates and also help when needed.

> *I would like to be in an organization where the manager knows at least something, I'm not saying that s/he should know about everything. But like this situation, that the manager knew almost nothing for a year, and even*

> *now very little, of course s/he is now trying a bit more. But the idea, that "I thought that I don't need to know what you really do", it just doesn't.*

Finding the right balance is important: to know when to help and give guidance but also to know when to trust the subordinates and let them make their own decisions. Learning by doing applies to also to software development, and for the learning to happen, there has to be room also for some mistakes. Lehtimäki (2006, 191) proposes that the spirit in the organization would be such that making one mistake is accepted if you learn from it and do not repeat the same mistake. Gathering information about mistakes is only positive if also the fix is included. This way it is possible to plan how to avoid the mistakes in the future.

> *But in the first place the person should like independently ... on the other hand the experience that you are allowed to. On the other hand if can take care of, can solve some problem alone, then it is easier to motivate the person. Than to guide always. Like if there is always someone guiding you, where does it lead? You can't make any decisions by yourself.*

In addition to understanding the content of the work in high level, it is important for the line manager to know what her/his subordinates are capable of doing and also the real work load they have. Of course the allocations are planned and hours are reported, but these do not always tell the truth well enough for the manager to know if modifications in the plans are needed. Especially in a situation where there is a new task and it needs to be decided who will start doing it, the manager needs to know who has the needed know-how to do it but also if s/he has time for it.

> *If more resources are needed in a specific area, then you have to think the possibilities, in my opinion. If you know someone who definitely can do the task, if you give it to her/him, and s/he is already with a lot of burden, then it's possible that s/he isn't anymore able to do the other tasks, which s/he was doing properly before.*

When the manager knows her/his subordinates, s/he also notices if someone needs extra help in adjusting to the new situation and finding her/his place in it. One of the interviewees has the feeling that the organization could be more effective if all the persons were working actively in the right place.

> *But then again, I feel that the internal persons either in the line organization or in the program need to be activated. We have at least, I feel, one person who could be more active. - - And like, I don't know who should do it.*

When leading people it is not just about knowledge and know-how, it is also about personalities. Somehow the managers would need to know their team members well enough to know what kind of leading each of them needs. This is important especially in a situation when a person is taking on new responsibilities, as it is not only the manager who guides the person with the new tasks. Also the colleague giving guidance needs to be motivated and possibly instructed what kind of and how much guidance is needed.

> *If the person only walks behind you, then that is all s/he does. This is my opinion. It is then difficult to motivate yourself to have the person included in what you are doing, in everything you do.*

Lehtimäki (2006, 114) says that leading is situational: the leader needs to know the persons well enough to know whether to give a very high level instructions like "please go and create this plan" or to guide the person in more detailed level like "Please create me a plan about testing this interface. Contact these persons and go through the interface specification with them. Agree the details and make sure the persons ask for approval for the plan from their managers." Lehtimäki points out that it is very important to know what kind of leading is needed in different situations.

Knowing the subordinates is also important from the work load point of view. There are persons who are willing to take on new responsibilities and do not want to say "no" even though they will then have too much work load. In a situation like this also the old tasks can suffer. However, the interviewee acknowledges that it is probably difficult to find managers who know their subordinates this well. At least a quite close connection would be needed to have this kind of understanding.

> *It can also sometimes go, regress. If you are given too many responsibilities, it can lead to a situation where the person can't do it anymore. And then it's, because of that it is good to have someone around who notices how this person works, and then sees it when s/he is going to positive, when to negative direction. But do such persons exist, not necessarily.*

Even though reporting is usually seen as a way to control the subordinates, it is also a way to get better understanding of what they are working with and how much time they are using for each task. This is also how the interviewees see reporting. In the current situation the best report of one of the interviewee's work is the fact that the application works. Still the interviewee has the feeling that if her/his manager knew her/his situation well enough, s/he could be more effective in her/his work.

> *At the moment I don't report my tasks properly to anyone. It can be noticed, what I'm doing, it can be noticed in the fact that the application I work with works well enough. At least I haven't been lazy. I could be more efficient if needed, though.*

The experience the interviewees have about leadership in the program and projects is not very flattering. There is even the feeling that leading people is not done at all, but only the tasks are being managed. The interviewees know that leading an expert organization is not an easy task. They see that the nominations for the manager roles is one of the reasons for the current situation: one of the experts is nominated as a leader and then s/he is not able to handle to whole area but focuses on her/his own expertise area also in the leadership.

> *Yes, yes, but that's exactly it, project work is leading the people and that is not working now. There is lack of know-how. Like it's not easy to lead an expert organization. That is not mastered at the moment.*

In the literature there are even ideas that the only task a manager has is to create and manage the culture in the organization. S/he needs to give the personnel a context with which the issues are explained and can be understood. However, the part of organizational culture which can be seen and is commonly acknowledged is only the top of the iceberg. Below this is the agreed part which has not been documented. The lowest and largest part is the subconscious area of the culture which can only be understood by examining the way people act and work in the organization. This is also the part of the culture which guides the work in the organization and is more stable than the official guidelines that can be changed easily (Mattila 2008, 21, 25) Thus to be able to manage the culture the manager really needs to know the organization, not just the official part but the real life, even in detailed level when it comes to ways of working.

Reporting is seen as an area where there is need for more transparency. With current way of working the reports are only the tip of the iceberg, but it is not possible to find the iceberg anywhere.

> *You can only see what everyone wants to lie, that is what you can see as you can't dig into it. You can't open it anymore, the document. - - It should be, if you want to find out, it should be transparent so that you can go and check the documentation of the group, you can see what is really happening in there. - - Some kind of consistent way of working, so that it would be transparent.*

Lehtimäki (2006, 108) sees that there are two kinds of project managers. There are the ones, who want to share as little information as possible, both out from the project team and into it. This way they feel that they can protect their team, solve all the problems themselves and be heroes. But there are also project managers who understand that heroes die young and that it is best to share as much information as possible to have transparency in the work. The open environment gives the whole project team a possibility to discuss all the problems and solve them.

Even if the manager does not need to participate in all the discussions and meetings, s/he needs to be the enabler for the cooperation.

> *But how the teams work together, it doesn't necessarily mean that there always has to be someone, all the team leaders or managers participating. Of course if it is needed, if there are persons who need the support, then one of the managers can be there leading the meeting.*

There is even fear that the way things are managed at the moment will lead to a situation where the business loses important data. In the personal level the lack of leadership leads to decreasing motivation.

> *And I'm especially bored with the way these projects are now being handled. I'm not at all interested in doing the task, but I'll do it anyway.*

> *But I would have expected something from the management of the project or middle level or somewhere. Like these, it makes me so tired, if I have tried to check things, I've been asking myself if this is now ok and then I'm answered that "yes, yes, you don't need to do anything". And then after a month I'm facing the fact that it's not ok.*

Pitkänen (2006, 15) says that in change situations the basic condition is that managing tasks is in control. However, leading people is quite often more difficult than managing tasks. In large organizations there are several issues that cause conflicts and if there is not enough room for interaction then the negative forces start affecting the cooperation.

What makes the studied case even more challenging is that the persons in the IT organization are working in the middle of two changes: there is the change caused by the merger and then there is the change that is always part of the work of people who develop systems. Like Pohjonen (2002, 15) says, system development is always a change process in the certain environment that is started by the development team. This double change can be seen in the expectations of the interviewees: they are the ones

who are changing the work of others with the modifications of the systems, but to be able to do this properly they need help and support from the management for the change they are going through themselves.

The interviewees' understanding is that leadership in the projects is about coordinating and making sure that the right people are working together. It is not about doing the actual tasks yourself, but finding the capable persons to do them and making sure the whole area is working well. To keep the team members motivated the manager should give the credit to the person it belongs to, and not take it her/himself for the single tasks. The manager can take the credit when all the tasks have been done in time and the project has reached its target.

> *The task of the team leaders is not to lead the work but to make sure that there are the right persons working together. No escalation is needed if the teams work.*

> *The person can take the credit for having been able to use the best possible persons for doing the task. That's the position of the manager. - - In my opinion it is not the manager, who does the work, buts/he knows who is the best person to do it and uses her/him. If the manager knows how to, can coordinate the tasks properly, then s/he is doing a good job. And s/he can take the credit for it. But in my opinion the manager can't take the credit for the work. This is how I see it. And sometimes it can be even more demanding to coordinate the tasks.*

4.3.2 The Importance of Meetings

In the projects the information needed for management is mainly shared in meetings. Of course there are also discussions of individuals, but a correctly set up meeting is a good way to get an overall understanding of the situation. It is not important only for the team leader or the manager to get the latest understanding of what is going on, but it is also necessary for the team members to hear what others are doing and what kind of issues they have faced. This is especially important in projects where the team members are working in different sites and they cannot have any "corridoor discussions". In some of the projects the meetings have not been invited.

> *There is so much, like we don't have that many common [meetings], if you think about the program work, in the latest release we didn't have any common, like for the whole IT team. Like then everyone was working separately. In the previous [release] we had IT team's, I can't remember*

> *further. And now we tried to have also in the release we are now working with also some common [meetings].*

Lehtimäki (2006, 59) has quite strong opinions about teams that are situated in several locations: he says that there are no virtual teams. If the team is a virtual one, it does not really exist. In his opinion the only way to achieve enough communication and the right team spirit is that the whole team is situated in one location and even sits in one room. My own experience has proven otherwise: it is quite possible to have virtual teams, but the communication needs are different and the leaders need to understand this. For example using chat tools is a good way to cooperate, but there is also more need for meetings, where all the relevant persons are participating.

A meeting is only as effective as the chairman of the meeting. The interviewees' opinion is that the person who invites the team meeting should also be the chairman and the secretary in the meeting. It does not mean that the chairman should know and understand all the details without any explanations, but s/he needs to have a plan for how to go through the issues on the agenda and to have a real leading position. The chairman should know what are the issues that need to be discussed, but of course there has to be some room for the urgent cases the participants want to bring up.

> *But like in the meetings, thinking about them, the person, who invites them, should have some kind of leading position, and also an idea how to go through the meeting. Of course then people have to, like when I've been, and have to ask many things, but still.*

> *Like there's always one person, who invites the meeting, who's the driving force. Then it's better that the driving person is the one who takes care of documenting and such. And then it should be in her/his interest to have the case handled. Then if someone else is interested in the case there can be a conflict if the discussions from the meeting are not documented.*

This is supported by Lehtimäki (2006, 56-7) who says the chairman of the meeting needs to plan the structure of the meeting so that the meeting has an objective and an agenda. All the preparations need to done carefully before hand so that no time is lost when the meeting should already start, for example all the materials need to be ready. Even if there has to be room for discussions in the meetings so that all the needed opinions and information are heard, all the issues that require detailed clarifications should be handled separately.

There are different kinds of meetings for different purposes. And also each meeting type needs to have its own practices to be effective. The interviewees recognized at least three types of meetings: status meetings, ad hoc meetings which are usually needed to clarify an urgent issue and meetings where issues are discussed and the target is to create a common understanding of a certain subject. The last type is almost like workshop.

> *Like when we are talking about status or a typical project meeting then there should be the continuous follow-up. And that is the status.*
>
> *And then there are these ad hoc meetings, in which it is not about status but something to be taken care of. Like these are a bit, in my opinion, then there should be the person who's responsible for the area.*
>
> *Then there are meetings where the target is to have a common understanding of something. And then the result is in participants' minds, not on paper, and writing it is not useful as no one would read it, because the discussion continues in the next meeting. It's kind of - - discussing high level things, the result of one meeting, if it was documented so that everyone would understand it, it would be several pages.*

In addition to the meetings which have a clear target, the meetings are also needed for more unofficial discussions when people are working on several sites. This applies also to situations when the team is working with the same application. Of course team membes can organize these meetings themselves but especially when starting the multi-site work it is better if the team leader makes sure there are meetings like this.

> *We are on four different sites. Like then it's not, that we have the one and a half hour weekly meeting, that is the time when we together, it's not that we would want to report to our manager or the team leader, but it's the time when we discuss many things together.*

In each meeting it is important that some kind of minutes is written of the content of the meeting. In status meeting the minutes can be a list of issues which are discussed in each meeting and the status is updated for each issue in every meeting. In ad hoc meetings more background information might be needed, but recording the decisions is very important as these meetings are usually organized to understand and decide what is the best way to start solving a problem. In the workshop kind of meetings it is not necessary to record all that is discussed, but the target is usually to create documentation of the subject that has been discussed.

> *The meeting minutes are created, whether it's a power point or whatever, they are created during the meeting. And it's not modified afterwards. - - So*

> *that the content is decided during the meeting and then no one should have anything to say about it.*
>
> *In the status meetings it's quite clear that when you write the meeting minutes, then there is something about every meeting. … There is always something written down and you can follow, and then old things can be moved aside later.*
>
> *Then at that point we decided that now it is time to write documentation, meaning that someone needs to do the hard work, like write, to use a lot of time to create the document and then we'll review, so that we can finalize it, the common understanding.*

Still it is not just about having the minutes written, they must be written by a person who knows what s/he is writing about. Otherwise the minutes can be even totally useless.

> *This way of writing the minutes, there are secretaries who don't understand anything what they are writing about, it really doesn't work. - - We need to try to have transparency in what we are doing and it must be possible to trace back everything. You can go through one thing and see who is doing what.*

4.3.3 Making Decisions

Decision making is almost like a form of art: you need to know when you can make a decision both from your role point of view and also from the information point of view. It is not enough to have the role that gives you the right to make decisions but you also need to have enough information so that you know you are making the decision based on relevant facts. And after the decision is made, you need to do follow up to make sure the decision is respected and to know the decision was the right one.

In some cases it is not enough to decide something and do follow up but there has to be support for the persons who are affected by the decision. Also communicating the decision and making everyone understand what it means in practise is part of the decision making task, when it is done succesfully.

> *Decision are done, like, sometimes they are done, there are decisions, they, I feel that ok, now you made a decis..., you like accepted this. But then when you need a kind of a stamping, that yes, this is how this actually works, then there are these fine, even from high level, that now we work like this in this area and if takes this many hours, then you do like this. But then again, to*

> *have more than just the one email, where there is the one paragraph, to have the help, then it feels like there is none, no one is interested anymore.*

In the program experts have not been listened to or their views and comments have not been even asked before significant decisions have been made. Because of this the decisions have not been accepted by the experts or even by larger audiences, as important aspects have not been taken into account and the experts have not been committed to work according to the decisions.

> *No preparations by the experts, but everything is done, the management prepares. And it means that the result is not acceptable. Like, usually, very narrow view of the whole case, there is no wider view which is usually needed so that the work can continue without conflicts. Now there is the problem that many views are not taken into account at all and then there is not much commitment. Especially in an expert organization, like we have.*

It is also the message from the benchmarked company that the information from all the relevant experts is needed before decisions can be made.

> *Like it was interesting that the core team was built up and the persons who were invited in the beginning, there were some persons clearly missing. Like for example there was no one who knows the application, in the beginning. There was only the infrastructure expert, like an internal person, and then someone got an idea that a person who knows the application needs to be invited, then I was also invited. - - They noticed that there are so many things that need to be asked from me. Then they probably decided that I also need to participate in the meetings regularly.*

Experts also need to know that they can make the decisions their roles entitles them to make and that no one will come and change the decision without discussions. On the other hand the management should not make any decisions without comments from the experts.

> *Like having the control and the support from the management for the single experts in the cases where they have the role, their own role. In the program there is always the danger that the program management makes decisions without asking or ignoring the expert role, and then the whole thing is useless, it doesn't have any value. Like there is also the problem that we need to have the commitment also from the program.*

Lehtimäki (2006, 117) points out that the team needs to have freedom to do tasks and make decisions it is been created for but on the other hand the leader needs to know when to participate the work her/himself. Delegating is about giving both the task and the authority needed for task related decisions to someone. Still the final responsibility about the task stays with the person who delegated the task.

Making decisions is the task of the management both in the line organization and in the program, but doing it without proper preparations only makes things worse: the management does not have the trust of the experts and the decisions might be wrong.

> *At least I can't trust this person, or persons like him, at all. In some cases they don't ask for help and they make their decisions like that.*

Trust is also a central part of the work community's capability to face a change situation. Building trust in organization is a long process, it can take years. However, in a change situation it can be easily broken and the effects of single incidents should not be underestimated. (Mattila 2008, 15, 17-8.)

In the program the milestone decisions are made by a group of people, the steering group of the program, and the decision is based on the evidence shown to this group. The interviewees have noticed that it is quite common that the information given to the steering group does not match the reality: even though a certain milestone is granted, in practise there are still many open issues to be clarified before the IT people can really start their work. From the steering group's point of view all the clarifications should have been done before the milestone is granted and thus for them it seems that the IT persons who continue the work are not able to keep the planned schedule even if they could start the work in time.

> *And then about decisions, maybe the fact that if we have these fine, that now it's really frozen, these, there was the PM2[project milestone 2] and it's like frozen now. And now there is this huge amount of mails sent back and forth asking is this or is this not, and what is this, there are these if sentences in the requirements and others, what does this all now mean. Like really having a decision that this is now frozen, that is needed.*

The ideal situation would of course be such that the milestone decisions are based on truth and facts, and the situation is not sugared before it is presented to the steering group. But how to achieve this, that is something the interviewees do not know.

> *Like not just having the beautiful world that is presented to the steering group and then the other world that the steering group knows nothing about, the world that is something totally different. I would like to see these coming together. But for that one either I don't have the great solution, how to change it so.*

When there are several actions happening simultaneously in the same area, decisions are needed from the person who is coordinating the whole area. An overall view is needed to know what can be done, when and where. This applies especially to testing.

> *There was a message from a tester that he will make changes in the test database. But was it really ok for him to do it? Like when the other project moves from one test database to another, is it approved in that sense? No one said anything. And I don't, I didn't want to, then I didn't say anything. Let's hope that it's ok.*

4.4 Managing Information Flows

4.4.1 Communication

Interviewees have noticed several ways how the communication could be improved in the program. In the team level it would be important to have available information that is targeted to the specific team as usually each team needs to examine the issues from its own point of view. The general communication packages are not enough for this purpose as the information is too high level and it can be too difficult to find the relevant parts from the team's point of view.

> *Like when I think the latest release that was done in project mode, it was upsetting experience. Like there was nothing. Well, there was some, like these common messages, for the whole program and the interest groups, but nothing for the group that was working on the specific area.*

The details can also be missing in other cases, for example with decisions. Usually it is not enough for experts to hear or read that this has now been decided, they also want to and need to know the reasoning behind the decision. Sometimes the additional information is needed to understand the decision but in some cases it can also be needed to be able to take actions needed related to the decision. It is also important to know who are behind the decisions, as this gives the expert a possibility to see the decision in right light.

> *Like decisions, you notice all the time that there are, things are happening, like yesterday I heard that there is a decision, we thought that the database is not needed in the portal. - - There was no documentation about it. Like the documentation, on the other hand there is too much information but the short, like we had a meeting, these people, this was decided or we like*

> *agreed this. - - The documents you could come back to later on, there are so few of those.*

Lehtimäki's (2006, 56) way of working sounds like one that would solve the problems in the studied organization: he shares so much information and with such details that the persons in his team would make the same decisions he has done.

Managing the information flows and handling communication can be easier with right tools. Nowadays there are many solutions that can be used for storing information so that it is easy for the reader to see if there is something new information for her/him available. These tools can also be used so that the information is stored in levels: you can start from the general level and dig into the details as much as needed.

> *But there should be solution for having all the information available so that you can also see when it is available. And trying to have the transparency there. Like if something changes a bit, you can see it easily.*

Interviewees would like to be able to dig into the details also with the info session materials. Currently this is not possible as the material in the info sessions is in very general level, which is good thinking about the wide audience of the sessions, but there should be the possibility for the experts to find out more about the issues one is interested in.

> *And then the communication for the end users, well, now it's like, there is the info session, something very general is told as it's a huge material in a very short time, and then you can't verify it with any other documentation, go through the details. - - You can only see that something has been done, but you don't know any specifics and you don't know whom to contact. You can't get into the details, and this is something you should be able to do.*

In the daily work communication is seen as everyone's task. When someone has new information, s/he would give this information to the person who is responsible for the area the information is about. However, for this to work the roles and responsibilities need to be defined so that it is clear who needs to know what.

> *It would be best, if everyone would communicate when they learn something new, they would know who is responsible for it and would tell her/him. Then the responsible person would communicate it to the others.*

McConnell (2002, 276) stresses the importance of communication in relation to team work. He even says that all groups are not teams even if thought so by the persons in the group, as it is characteristic for a team that there is good interaction between all its

members, and interaction is quite impossible without proper communication. The team size affects the communication a great deal, as the number of communication channels in a team does not increase linearly according to the number of team members, but according to the square of the number. Thus in a project of 10 persons there are 45 communication channels if all the persons communicate with each other. In a situation like this the communication can be effective only if it is made formal enough and there is enough coordination in the team. Usually formality and coordination mean creating sub-teams. (McConnell 2002, 311-12) Tsui and Kram (2007, 36) agree with McConnell by saying that "organizational structures need to be put in place to reduce the complexity and increase the chance of correct communications".

The interviewees understand the importance of communication, but they have experienced that everyone does not. They feel that communicating is not seen as something you need time for, and as the communication is not taken into account when planning the allocations, it is then difficult to use time for it even if you know yourself that it is important. Especially in projects where there are people from different teams and organizations, a large part of the cooperation is communicating.

> *Is it always my job to think about them, like when you do this, then you communicate it to these persons and then you do this. Like probably part of it is my job, but then again I'm not given any time to do it. As it's not work. Only testing and sending data is work.*

Pohjonen (2002, 51) says that according to studies developers use 50% of their time for communication and only 30% for independent productive work. It sounds odd that in the studies communication is not seen as productive work, even though communication would seem the best way to acquire information needed for successful software development. However, it seems that also in the studied organization it is not understood that communication takes time and people who have communication tasks need big enough allocations for them.

Even though communication is seen as essential part of cooperation with business people, it is understood that having a full time person handling the communication is quite costly.

> *There should be one person from business side who knows everything that is happening on their side and we should also have one. It would be good to have it like that related to the timetables. But of course the complexity and on both sides, it's a challenge that … not necessarily, it's almost a full time*

> *job. If it was done properly. If you start doing it like that it costs a lot of money but is the benefit big enough, to have a full time person doing it. Not necessarily.*

Still there would be benefits in having persons on both sides, IT and business, who would take care of the communication. This applies especially to testing as there is often many projects testing in same database at the same time, and it is easy for one project to make a little change which then affects the work of the other projects and can even damage their schedule.

> *Well, communicating with business has always been a challenge in the IT. We have never had a proper interface, a single point of contact so that we could agree all the timetables so that they suit everyone. And taking into account that we are working with a large database and the test databases are needed for debugging. If someone is then testing there, it does not look good. The UAT (user acceptance testing) can be ruined.*

Both Pohjonen (2002, 50) and McConnell (2002, 276) stress the importance of communication. According to Pohjonen (2002, 51) a developer needs to be humble enough and ready to admit her/his ignorance. However, he also says that the social skills, attitude and characteristics of developer effect the communication between the developer and the user. Thus it is important the IT persons who are in contact with the users understand how to act in such situations and have the right qualities for this kind of work.

In the benchmarked company the communication channels of the project had not been defined clearly. Because of this the intreviewed person was forced to discuss same issues with several persons, and he ended up being the contact person for the external company's persons even though this was not the intention. This shows that if the communication and communication channels are not planned carefully time is lost in doing the same work twice or even several times and people find them themselves doing tasks that were supposed to be handled by someone else.

> *Communication was usually handled by the grapevine. There was mails going back and forth. Core meeting, or this core team, the meeting was each Friday if I remember correctly, or Monday, and there the main points were discussed, but there was no clear model about who reports to whom. Like at least I had the feeling that I was discussing with our configuration manager and with the product manager and between them. And then I told them what the external company had told me. And then we ended up in the situation where the external persons contacted me directly.*

This is not uncommon, as communication is a key factor also when outsourcing the software development or even parts of it. Several matters can complicate the cooperation: language, culture, time difference, network connections. Language for example can be a problem even if everyone uses English as the accents can be so strong that it is very difficult to understand one another. However, effective communication is the only way to follow up the work of the partner carefully enough to be able to notice any misunderstandings or deviations from the project plan early enough. (McConnell 2002, 493, 496; Tayntor 2007, 378-9.) According to Tayntor (2007, 379) it is important to discuss also these issues openly with the partner so that the methods to reduce the risks can be agreed together. The main methods are quite basic project management tasks: regular status reporting and milestones. In addition to these Tayntor stresses the importance of frequent face-to-face meetings to enhance communication.

4.4.2 Documentation

The documents that are related to software development process can be divided to three groups: product documentation, project management documentation and the documents that are related to the quality system. Product documentation includes the technical documents that are for the developers and the different user manuals and training materials for the end users. Project management documentation is for organizing, managing and follow-up of the project. They are one time use only: after the project they are archived. For example different instructions, document templates and meeting minutes are quality system related documentation. This material tells about the quality of the work done. (Pohjonen 2002, 79.)

Current situation with the legacy PDM system technical documentation in the studied organization is not very good. There is some documentation, but parts of it is outdated and some documentation is missing totally.

> *We have some system documentation and some concept documentation, but they are not very detailed. And in practice they have not been developed, the documentation has not been modified since this quite old project.*

However, this is quite normal for a system that has been developed for quite many years. Legacy system can be even be defined as a system for which there is no proper

documentation available and it is not possible to understand how the system works if the original developers and administrators are not available (Pohjonen 2002, 19).

The main reason for the deficiencies in the documentation is that there is not enough time for documenting in the projects. The interviewee also feels that the people working in the projects are not given motivation to create and update documents. Also it is not always clear how to document something. Overall it seems that the documentation is not seen as very important in the projects.

> *Partly it's because the persons have not been motivated to create these documents. And other thing, which is probably more important, is that we are doing so fast that there is no time to create documents. And then there are persons, like me, who don't know how to document, then some support would be needed to decide how to document something.*

It is good to remember that based on statistics only one third of the application development is developing a totally new system. All the rest is maintaining old systems: modifying, fixing and further development. (Pohjonen 2002, 18.) From this point of view it is very important to create and update documentation in all software development projects. The maintenance is difficult and time is lost if no proper documentation is created when the new application is configured for the first time or in the later projects.

The deficiencies in the documentation slow down or even prevent the changes in the roles and responsibilities: as there is not enough acceptable documentation of the legacy systems available it is not possible to give the still needed development tasks to an external company. This leads to a situation where the people are stuck with their old tasks and cannot start using time to learn the new ones.

> *We should improve our current documentation quite a lot, to start with. So that we could do it. We could have them doing some of basic support tasks, we could have them there, but in my opinion we could not have them doing anything more demanding.*

It was also noticed in the benchmarked company that the development of a system that is not properly documented is not a project you can give to an external company.

> *And you can say that it is our fault that the technical documentation is not in the level, that anyone who does not know the system could just start doing the changes. Like that is one, what I see in a project like this, before you can give the development to an external company.*

Outdated documentation can also have serious effects on the everyday work of the business: if people who have the not-documented critical knowhow about the legacy systems decide to leave, it is not possible to guarantee even the continuity of the operation of the systems and especially development of these systems.

> *If we think that two persons, who are working with the same system, leave, then the situation can't be such that everything stops. In any case most of our data goes through this one system, and then, I don't know, how much documentation we have for the system and is it transparent, can it be found.*

The current situation could be used to find out what have been the ways of working: the documentation that has been created tells something about the level of current practices. The documentation reveals at least the following: Which documents have been seen so important that they have been created, who has created them and at which point. Once this is acknowledged it is easier to start defining the new ways of working for the new application.

> *We should do some kind of auditing to find out what we have and do it in relation to the changes of legacy systems. Like how to do documentation reviews. To check the existing documentation and how it is related to the new documentation, and to have some kind of audit what documentation can be found, how it can be found and what is missing. And in that way to see on which level we are working at the moment.*

In addition to the application related documentation, there are also other kinds of documents created in projects. Plans are the most relevant ones, and for the IT people especially the plans that tell something about the coming changes. The sooner the IT people hear about the planned concept modifications or new features, the easier it is to start the cooperation with the business side with the case in question and to plan the actions needed in IT team.

> *The plans for the concept area should be available, like what are they doing now. So that it can be easily checked if there is something relevant. And then in own status meeting to say that there is this kind of case, should we do something about it. And then we could agree together, for example in tool team, that ok, we need to ask about this and who's going to do it.*

Like communication also documentation needs to be as open as possible and include all the needed details and background information. It is not enough to write down the end result but also the reasoning needs to be somehow included. This is important especially in the situation when the existing solution is changed: if it is not known why the current solution is as it is, it is not possible to evaluate all the effects the changes will have.

> *If you start going through the product structure model yourself and if you ask why it is like this, you can't find the answer in the documentation. There is the data model, what information is in it, where it goes and so forth, but like the clear understanding of what the data model means, why it is like it is, that is not included.*

Tsui and Karam (2007, 31) point out that documenting the work performed is important even if there is only one designer doing the work. This it to make sure that the reasons for the decisions made can be retraced if needed.

4.4.3 Cooperation and Reviews

Reviews are often thought of as something that takes time and must be done to achieve the planned milestones, but the benefits are not understood. According to Tsui and Karam (2007, 256) inspections and reviews are part of quality control. "They require the involvement of more than one person in addition to the original creator of the product. They are typically labor-intensive but have been shown to be an extremely effective way of finding errors."

One reason for not understanding the importance of reviews might be that also the advantages of having good cooperation between different teams during the whole project cannot be seen if people are working in their own teams and there is not enough discussions between groups, especially during the first phases of the project.

> *It's not noticed at all that the mistake in the beginning, that is what costs. Like it has not been taken into account that the whole chain needs to be thought of all along. It doesn't mean that when we use the waterfall model that then we just, that the people are also divided into different phases of the waterfall.*

According to Pohjonen (2002, 16) it is good to remember that it is not a good idea, or even possible, to automate everything. Because of this some parts of the work are always manual, and there is a relationship between the manual and automated work. These are interfaced with each other and also to the environment where the system is used. The notion of system includes also the environment where it is used and the organizational, social and human dimensions. This is something to be taken into account also when planning the system development: it is not possible to sequence the

work very strictly as there are so many different areas and options to be thought of. One team is not able to do this, but the teams need to cooperate from the very beginning of the project.

In practice there is a clear link between the cooperation of the teams and the reviews: if all the needed people have been participating the work from the beginning, the reviews are more like check points for making sure that nothing essential was missed during the work. However, if one team creates a solution and it is handed over to the next team through a review, the review takes a lot of time and it is much more than just a check point. And usually the clarifications continue after the review as it is not possible to go through everything in one meeting, and usually there is a lot of pressure time-wise to get an approval for the reviewed documents.

> *And now if we compare this situation to the good old times, what we did earlier was that we had the experts who said whether the proposal is good or not. And typically the same experts had participated already in the preparation one way or another, meaning that we had established ways of working and then the review took a reasonable time. Then we did not need, there was no such problems that we should have reserved a month. But now, in a way, as this situation is totally different, now we really should reserve. It's a change that should be made, but it has not been done.*

Stepanek (2005, 131) says that the most obvious reason for the software development project to fail is the conceptual gap between the technical and non-technical members of the project team. Both the communication of the requirements from the customers to the developers and communication of the repercussions of the same requirements from developers to customers is the very common source of problems. "The key to software development success is frequent, on-going communication between the developers, the customer and the project manager through-out the project, with regular opportunities to confirm understanding and give feedback " (Stepanek 2005, 132).

One of the interviewees also feels that there might be hidden agendas in creating the documents by one group and organizing the review so that there must be a final decision immediately available: by doing like this you can easily get the decision you want as no one is able to give comments, let alone make a proposal of a better solution.

> *Now it's usually so that the documents are delivered the day before and then there is the review and then everything should be ready. No, this is typical way of doing it, what I've seen before, typical way of getting the decision you want, if you know that someone would oppose, you make last minute*

> *changes just before the review and then, as there are no counterproposals, then this is approved.*

The solution for problems like this is to have the reviews organized and followed up in a defined way. And it is not enough to have the review meeting, even more important is to have the documents available some time before the meeting so that people can read them. In addition to making sure that the documents are available, people need to have time allocated for getting familiar with the documents and for participating the reviews.

> *There has to be the rules, like if a review is planned then also the documents need to be ready and someone has to record it. Like there has to be the follow up, someone from the management team does the follow up. … That's why there is the management, to make sure that the program works.*

In the benchmarked company the idea was to have internal persons doing reviews but when the external company was not able to create all the needed solutions in time, the internal persons started doing them also. Then there was not enough time for reviewing anymore. Still it was noticed that it would have been important to have someone reviewing the solutions, in their case especially from the technical point of view as the idea was not to change the concept.

> *The way of working was supposed to be such that the external company creates and delivers the new solutions and we [internal persons] review the solutions. We do all kinds of technical reviews, code inspections and everything and then give approvals. - - What happened was that we and our own external persons, we were all full time employed with creating the solutions. And then we did not have any time for it [reviews]. - - But there really has to be, no matter who is creating the new solutions, there has to be an internal person who has the tools and allocations for technical reviews.*

Reviews are one way to make sure that the work done is of high quality. Quality can be defined in two different ways: a product can be said to have high quality when it

- conforms to specifications and
- serves it purpose.

It is possible that a product conforms perfectly to specifications but it does not serve its purpose at all. This is an extreme example, but shows that two activities are needed to make sure the quality is in place: verification and validation. Verification is the act of checking that a product conforms to its requirements and specifications. Validation, on the other hand, is the act of checking that a finished product meets users' requirements and specifications. (Tsui and Karam, 2007, 257.)

5 New Ways of Working for PDM Application Development

It is old-fashioned to say that the continuous chaos must be tolerated as it is temporary. The need for change is constant, and the cure is not to bite your teeth together and cope with it but to find the recipe for controlled renewal. (Mattila 2008, 87.) I have listed below some ingredients for the recipe as instructions for the whole PDM line organization and also the program in which the PDM application development work is done. The instructions are based both on the theory part of this work and interviews. The interviews showed the areas were improvements are most urgently needed and both from the interviews and the theory I found the solution proposals.

Based on the literature it seems that a traditional process would suit the coming PDM development projects better than a very agile one. This is because the PDM application is critical system in the company, the development projects are large with people working on several sites and the architecture needs to take the foreseeable requirements into account. However, the requirements are at least partly emergent and this should be managed with the right project management practices: subteam encapsulation, feature trade-off, triage and scoping studies.

The more specific instructions below are both for personnel and managers. It depends on the job description of the person whether an instruction is something that the person is responsible for doing her/himself or if it something that s/he needs to know about but the execution is someone else's responsibility. For example the instruction "Make sure there is enough time for adjusting to changes, for example teaching new tasks for a person in a new role" includes a task both for the person teaching the new tasks and for her/his manager: the person needs to tell what her/his situation allocation wise is and the manager needs to make the adjustments needed in the allocation, if any. Because of this I have not created separate lists for different roles, but the idea is that everyone reads the instructions carefully and thinks how s/he can apply it in her/his own work.

Some of the instructions could be placed in several categories, but I have placed each of them into one category only. I have tried to find the category that is the most relevant one for the instruction in question. For example the instruction "Communicate all the

important changes in as many ways as possible: do not think that having information somewhere in the intranet is enough" could also be about communication. However, in practice it is the changes that need to be communicated most effectively as lack of information increases the uncertainty and number of rumors which most often are not true.

5.1 Change Management

Change management is an area where there seems to be quite much need for improvements. It is important to understand that because of the different histories in the two companies also the change management needs to be handled differently. In short the recipe for successful change management is to decide to make a change, plan the execution in detailed level, communicate the change and follow up.

1. Discuss the different ways of working openly and take the history into account when planning and agreeing the new, common ways of working.
2. Organize meetings and workshops where persons from both organizations can discuss both work related and other issues.
    a. In work related issues it is important that there is enough sharing of ideas and thoughts so that common understanding of notion of PDM can be built.
    b. Discussing other issues brings people closer each other and helps understanding the differences in all areas of life.
3. Listen to all the opinions, do not quiet down the loud ones and try to encourage the more quiet ones: the criticism is important for recognizing the areas where most urgent improvements are needed.
4. Plan and agree also the details in the practical level that are important to the persons when doing their everyday work.
5. List carefully all the tasks, both old and new ones, and prioritize them. Follow the prioritization even if it is difficult at first.
6. Agree the common target and ways how to reach it.
7. Follow up all the introduced changes carefully: make sure that changes are really happening in practice, not only on paper.

8. Make sure there is enough time for adjusting to changes, for example teaching new tasks for a person in a new role.
9. Communicate all the important changes in as many ways as possible: do not think that having information somewhere in the intranet is enough.
10. Follow up the communication about the changes: check that all the persons have the information and if they do not have it, find out why and use this information to handle the communication more effectively next time.

5.2 Roles and Responsibilities

In the studied organization the issue with the descriptions of the roles and responsibilities is the lack of them. There are some old descriptions but for some reason the creation of new descriptions has not yet taken place.

1. Define roles and responsibilities in the line organization and use these in the projects.
2. Define also the roles and responsibilities of the external persons (augmented staff).
3. Define the interfaces with contact persons to other organizations, for example the ones in the business side.
4. Delegate and make sure the delegation is respected: the person who is handling the task needs to have the power to make the decisions needed without anyone overruling them without discussions.

5.3 Working with External Persons: Augmented Staff or Outsourcing

With external persons it is most important to know what has been agreed and to follow up that the work done complies with the contract. Of course also some flexibility is needed on both sides, otherwise it is impossible to adjust to the changes that always occur in software development.

1. Make sure all the relevant persons (at least all the person who are in contact with the augmented staff or the persons of the outsourced company) know what has been agreed in the contracts.
2. Make sure there are internal persons allocated for specification creation when externals are doing the actual technical system development.
3. External persons supervision needs to be agreed and handled, taking into account that their motives might be different from the internal persons' ones.
4. Make sure everyone knows the target and is ready to work to achieve it and make compromises if needed along the way.

5.4 Building the Teams

In the studied organization there are two kinds of teams: teams in the line organization and teams in the program. As most of the work is done in the program, also the instructions below are for the program organization. Of course the more general instructions, number 1 and 3, can be used in both organizations.

1. Build trust in the new team members remembering that trust is not about knowing exactly what someone will do but trusting that what s/he will do is fair.
2. Team structure could be the following when working with out-of-the-box system in a larger program:
    c. a leader that communicates with and reports to the program management,
    d. inside the team there needs to be several separate teams and each of them is responsible for a specific area and
    e. the leader of the whole team is in a supporting role from the technical development point of view, her/his main task is to make sure there are no obstacles for the effective work of the other persons in the team.
3. Build the teams so that there are members from both old companies and make sure the team has real task they need to work with and achieve results.
4. In addition to the hierarchical teams make sure there are teams that have members from all the relevant hierarchical teams and are able to solve specific problems.
5. Team leaders need to be able to lead in the first place, the team members are the specialists that do the actual work.

5.5 Leading the People

Instructions about leading people show that leadership requires skill to recognize issues behind the surface and sensitivity to understand the whole situation of the subordinates or team members. Again these instructions are both for the managers in the line organization and in the program, but especially the number 1 and 2 are for the line organization managers.

1. Be truly interested in the tasks of your subordinates and get to know the environment they work in.
2. Learn to know your subordinates well enough to have an overall understanding of their situation.
3. Give enough freedom but be supportive when needed.
4. Remember that making mistakes is human and learning from the mistakes is what matters.
5. Share as much information as possible and listen to your team members.
6. Make sure your team has everything that is needed for cooperation both inside the team and with other teams.
7. Give the credit to the persons it belongs to: the ones who have done the work.

5.6 Meetings

Even though participating and organizing meetings is nothing new for the persons from either of the companies, in a hectic situation it is easy to forget even the basic ways of working for organizing a successful meeting. Also the fact that persons are working on several sites sets some requirements for the meeting practices. There are easily too many meetings but the number of them does not help if the ways of working are not efficient.

1. Make sure there is the right number of meetings and the right kind of meetings:
    a. status meetings to update the understanding of the current situation of all the team members,

   b. meetings that are for discussing and solving issues and
   c. workshop kind of meetings for creating understanding of a new issue.
2. A meeting needs to have
   a. a chairman who makes sure all the preparations have been done and leads the meeting,
   b. an agenda,
   c. meeting minutes created during the meeting by a person who understand the contents of the meeting in detailed enough level,
   d. time for discussion so that all relevant opinions are heard and
   e. decisions that issues requiring clarifications are handled separately.

5.7 Decision Making

Making decisions is easy, but making them so that persons affected accept the decision is a bit more difficult in the sense that it requires more work. However, when decisions are made following the instructions given below, also the quality of the work in general improves: the reasoning for the decisions can be tracked down later on when it might be difficult to understand the decisions without knowing or remembering the environment and circumstances of the time.

1. Before making a decision
   a. make sure all the information needed from the experts is available and
   b. try to make sure the information given is truthful and correct.
2. In areas where a lot is happening at the same time, like testing, the decisions are needed from the person who has the overall coordination responsibility and a comprehensive understanding of the situation.
3. Make sure the reasoning for decisions made is documented and available for persons who need it in their work.
4. Follow up the decisions made.

5.8 Documentation

Documentation is most often the part of the work that is not done when time is scarce. Depending on the development process selected documentation has different kind of values. In the studied case it seems that there is a need for documentation both for the development and maintenance phase. During the development the people working on different sites need to be able to share knowledge also in documented format. This applies both for the people working with processes and the IT organization people. In maintenance phase the documentation is needed for the persons responsible for the maintenance as they are not the ones doing the development.

1. Define what kind of documentation is needed for different purposes.
2. Make sure there is enough time for creating the documentation.
3. Review the current situation to understand what documentation has been created and why and use this in planning the documentation tasks of the new system.
4. Document all the planned changes openly and distribute the documentation to all the relevant persons.

5.9 Communication

Communication is one of the most important tasks in software development and it concerns everyone. From the requirement specifications to testing arrangements there is a need for information sharing to make sure people understand each other correctly and that the work is proceeding as planned. Even though documentation is used to convey information, there is always a need for discussions. Still, communication is not only about spoken information sharing but there are also documents that are created only for communication purposes.

1. Team level communication is needed to have detailed enough information available.
2. Use the tools available for information sharing.
3. Details of issues included in general information packages must be available.
4. Take communication into account when planning the allocations.

5. Role and responsibility definitions are needed for successful communication: to know who needs what kind of information.
6. Communication is the key in understanding the requirements correctly.
7. Face-to-face meetings are needed for starting effective communication, especially when there are new people working together.

5.10 Reviews

Reviews are about communication and documentation. It could be said that the main idea of a review is to make sure that there has been enough communication to create the documents with quality and to have correct information in the documents. The same instructions that were presented for organizing a meeting apply also for reviews, below are a couple of more points to be taken into account.

1. Good cooperation is a prerequisite for efficient reviews: if all needed persons participate in the creation work, the review is mainly a checkpoint for quality control, not for revealing the work for the first time.
2. Reserve enough time for reviews.
3. Make sure the documentation to be reviewed is ready in time; if not, postpone the review.
4. Create the minutes of the review meeting.
5. Follow up the action points recorded in the meeting.
6. Do not give up the reviews even if there is time pressure.

6 Conclusions

The starting point for my research was a problematic situation in the PDM organization where I have worked as a project and planning manager. The organization was created as a result of a merger of two large ICT companies by bringing together the persons who worked in PDM area in the companies that were combined. Still about 20 months after the merger there were several sore points in the ways of working in this organization. The target of this research was to recognize the areas where improvements were needed the most and to create a proposal how to tackle the problems.

In the theoretical part of this research I wanted to clarify what are the characteristics of software that make the development of applications and management of application development projects different from the same activities in other areas. To find out the situation in the PDM organization I interviewed three persons working there. I used the conclusions based on the theory as a point of view for the research material analysis. With the results from the analyses and the theoretical part of this work I listed the recommendations that could be used as guidance in future to improve the ways of working in the selected areas.

My original idea was to conduct an action research where the new ways of working would have been first developed and then tested in practice. This of course would have given a possibility to evaluate whether the instructions created are of any help. However, to really find out what is behind all the complaints I had heard since the merger took place required one research. What made the case very interesting was the fact that the complaints were heard from the persons working in the legacy application projects while the projects reached their targets.

What is wrong if the objectives are fulfilled but people are not happy about how it happens? And could the work be more effective if people were happy with the way the work is organized? Timo Kaisla (Mikä on systeemityössä IN?, 2007), who has experience in project management services, says that even a consultant is not able to help with large and unique programs. Management and project managers know how to organize a normal project but not how to manage a change that affects the whole

personnel. And the project management guides like PMBOK (Project Management Body of Knowledge) describe what should be done but not how to do it.

With the first look it might seem that the instructions listed are nothing new and some of them are even self-evidences. Who would even think about organizing a meeting without an agenda or intent to write the minutes of the meeting? Still in a situation where nothing is stable and the flow of significant changes is continuous it seems necessary to create a list of these quite simple things. As the target of this work was not to create new ways of working altogether but to find the ones that would help the studied organization function more efficiently, I feel that the research has been successful.

On the other hand, even if the instructions are simple and repetition of old advices, there are also more details than in the project management and application development manuals. For example it is a fact that communication could always be handled better, and usually the instruction is to communicate more. In this work I have pointed out what kind of communication is needed by the persons working in the PDM line organization and in the program that were the target of the research. Even if the merger makes the situation quite unique, looking back at the organizations and programs in which I have worked before, I feel that most of the instructions are valid in any project that focuses on application development. Of course the need for the instructions depends on the individual case, but the advices given can be applied in any large application development program.

My current personal situation, being on maternal leave, gave me a new perspective as I was able to go through the interview materials without being part of the organization at the time. Of course my experiences affected the creation of the whole work, but stepping aside and looking at the case from a distance helped me to analyze the materials and create the results with not too many feelings attached to the work. Even if this is not required in qualitative research I felt that from this role I was better able to create the constructions so that match the ones of the participants and also make the constructions understandable to others. This is a requirement in qualitative research like also thinking how the subject and nature of the research have affected the information the persons participating in the study have given. (Saaranen-Kauppinen & Puusniekka

2006) I feel that the interviewed persons did not have a need to sugar their views or leave out anything as their names are not included in this work and I gave them a possibility to read through the material before it is published.

The interviews helped me to understand better the reasons that had led to the situation. This started already when I was doing the interviews. Even though I had worked with all the interviewed persons for several years and thought I knew them and their ideas quite well, the interviews gave me a possibility to have a discussion with them. There is usually no time for discussion like these during the work days and still the comments and development ideas from the interviewees would be very beneficial for any project manager or team leader.

There are several directions for further studies in this area, and even with this specific case. Like Harris et al. (2004, 65) say organizational culture clash is a major problem in integrating different companies. They continue that making the deal is easy, but making it work is difficult. This is exactly the case in the studied organization even if there is already some proof of making it work. Taking the cultural differences into account it would have been interesting to find out how the persons in other sites and countries have experienced the situation and what the instructions based on their proposals would be like. Now the instructions are only from one cultural point of view. Still when comparing them to the theoretical background I feel that the results are applicable worldwide. But of course the people from other cultures might miss other kind of improvements.

Research is always only scratching the surface and it is not possible to describe the phenomenon exactly like it is in reality. To continue further with the results presented in this research and scratch a bit more, the next step would be to take the instructions into use and to do an evaluation after some months. This could of course be done by my colleagues or by me once I am again working in the organization. The instructions could also be extended with still more precise guidelines like templates for meeting of minutes and proposals of tools that can be used for sharing information. Also concentrating more on one area, like management in line organization after a merger, would be an interesting subject for further research.

References

Printed

Artto, Karlos, Martinsuo, Miia & Kujala Jaakko 2006. Projektiliiketoiminta. Helsinki: WSOY.

Erkkilä, Kaija 2001. Haltuunoton ja yhdistämisen haasteet: integraatio yrityskaupassa. Helsinki: WSOY.

Forselius, Pekka, Dekkers, Carol, Karvinen, Matti & Kosonen, Matti 2008. Program Management Toolkit for Software and Systems Development. Helsinki: Talentum Media Oy.

Grieves, Michael 2006. Product lifecycle management: driving the next generation of lean thinking. New York: McGraw-Hill.

Grönroos, Mauri 2004. The dynamics of knowledge and networks. Tampere: Transatlanta.

Harris, Philip, Moran, Robert & Moran, Sarah 2004. Managing cultural differences: global leadership strategies for the 21st century. 6th ed. Burlington, MA: Elsevier Butterworth Heinemann.

Lehtimäki, Timo 2006. Ohjelmistoprojektit käytännössä. Helsinki: Readme.fi

Mattila, Pekka 2008. Otollinen tilaisuus: miten tarttua muutokseen. Helsinki: Talentum.

McConnell, Steve 2002. Ohjelmistotuotannon hallinta. Translation. Toikkanen, Tarmo & Arola, Jussi. Helsinki: IT Press.

Pohjonen, Risto 2002. Tietojärjestelmien kehittäminen. 2nd ed. Jyväskylä: Docento Finland Oy.

Saaksvuori, Antti & Immonen, Anselmi 2004. Product Lifecycle Management. 2nd ed. Berlin: Springer.

Stellman, Andrew & Greene, Jennifer 2005. Applied Software Project Management. Sebastopol: O'Reilly Media, Inc.

Stepanek, George 2005. Software Project Secrets: Why Software Projects Fail. Berkeley: Apress.

Tayntor, Christine 2007. Six Sigma Software Development. 2nd ed. Boca Raton: Auerbach Publications.

Tsui, Frank & Karam, Orlando 2007. Essentials of Software Engineering. Massachusetts: Jones and Bartlett Publishers.

Venkula, Jaana 2005. Epävarmuudesta ja varmuudesta: johdantoa epävarmuuden kohtaamiseen. Helsinki: Kirjapaja Oy.

Electronic

Abrahamsson, Pekka, Salo, Outi, Ronkainen, Jussi & Warsta, Juhani 2002. Agile software development methods: Review and analysis. [online] [referred to 07.04.2009]. http://www.vtt.fi/inf/pdf/publications/2002/P478.pdf [Published in print: VTT Publications 478, Espoo: VTT]

Abran, Alain, Moore, James, Bourque, Pierre & Dupuis, Robert 2004. Guide to the Software Engineering Body of Knowledge. Washington: IEEE Computer Society Press. [online] http://www.computer.org/portal/web/swebok/htmlformat

Aronson, Jodi 1994. A Pragmatic View of Thematic Analysis. [online][referred to 21.09.2009] http://www.nova.edu/ssss/QR/BackIssues/QR2-1/aronson.html [Published in print: The Qualitative Report, No 1, 1994.]

Association for Computing Machinery, Inc. and the Institute for Electrical and Electronics Engineers, Inc. 1999. Software Engineering Code of Ethics and Professional Practice. [online][referred to 18.08.2009]. http://www.acm.org/about/se-code#full

International Project Management Association. [www page]. [referred to 24.10.2009]. Available: http://www.ipma.ch/Pages/default.aspx

Karjalainen, Asko 2002. Mitä benchmarking-arviointi on? [online][referred to 30.09.2009]. http://www.oulu.fi/w5w/benchmarking/bm.RTF

Livesey, Chris 2009. Sociological Research Skills, Research Methods. [online][referred to 16.09.2009]. http://www.sociology.org.uk/methfi.pdf

Mikä on systeemityössä IN? [online][referred to 20.11.2009]. http://eijataina.wordpress.com/2007/09/07/mika-on-systeemityossa-in/

Oliver, Daniel, Serovich, Julianne & Mason, Tina 2005. Constraints and Opportunities with Interview Transcription: Towards Reflection in Qualitative Research. [online][referred to 21.09.2009]. http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1400594

Pollice, Gary. 2005. Teaching software development vs. software engineering. [online][referred to 16.08.2009]. http://www.ibm.com/developerworks/rational/library/dec05/pollice/index.html

Project Management Institute. [www page]. [referred to 24.10.2009]. Available: http://www.pmi.org/Pages/default.aspx

Saaranen-Kauppinen, Anita & Puusniekka, Anna 2006. KvaliMOTV - Menetelmäopetuksen tietovaranto. Tampere : Yhteiskuntatieteellinen tietoarkisto [ylläpitäjä ja tuottaja]. [online][referred to 21.09.2009]. http://www.fsd.uta.fi/menetelmaopetus/kvali/L7_3_4.html

Tellis, Winston 1997. Application of a Case Study Methodology. [online][referred to 16.09.2009] http://www.nova.edu/ssss/QR/QR3-3/tellis2.html [Published in print: The Qualitative Report, No 3, 1997].

Tellis, Winston 1997. Introduction to Case Study. [online][referred to 16.09.2009] http://www.nova.edu/ssss/QR/QR3-2/tellis1.html [Published in print: The Qualitative Report, No 2, 1997.]

Non-printed

System developer. Interview 25.11.2009. Benchmarked company.
System developer. Interview 8.12.2009. ICT company.
System analyst. Interview 9.12.2009. ICT company.
System analyst. Interview 10.12.2009. ICT company.

Appendices

Appendix 1: Themes and questions of the interviews in ICT company

Outsourcing:

- What was good in the cooperation with partners?
- What was not so good?
- Why?
- How should cooperation be managed in the future?

Roles and responsibilities

- What do you think about the roles and responsibilities at the moment?
- What kind of changes are needed in the roles and responsibilities?
- What kind of changes have you already noticed in the roles and responsibilities?

What do you think about the following in relation to project management?

- meeting practices
- decision making
- reporting
- communication
- document management

Change management

- How has the change been managed so far?

Appendix 2: Themes and questions of the interview in the benchmarked company

Outsourcing

- Why were the partners included in the work?
- How was the partner selected?
- What did the partner do?
- Creating contracts: Where all the internal persons informed about the content of the contract with the partner?
- How was the communication and cooperation with the partner managed?

Roles and responsibilities

- What kind of roles and responsibilities there were

    - in house
    - with the partner
- How were the roles and responsibilities defined?
- How was the work of the internal persons changed?
- How well did the changing of the roles and responsibilities succeed?

Project management, how were the following agreed

- meeting practices
- making decisions
- reporting
- communication
- document management