HELSINKI UNIVERSITY OF TECHNOLOGY
Department of Computer Science and Engineering
Laboratory of Software Business and Engineering

Jouni Hartikainen

Adoption of Agile Software Development from the Perspective of Organizational Change Theories

Master's Thesis

Espoo, May 9, 2008

Supervisor: Professor Reijo Sulonen

Instructors: Kristian Rautiainen, Lic. (Tech)
Jouni Virtaharju, M.Sc. (Tech)

HELSINKI UNIVERSITY OF TECHNOLOGY
Department of Computer Science and Engineering

ABSTRACT OF MASTER'S THESIS

Author
Jouni Hartikainen

Date
9 May 2008

Pages
90

Title of thesis
Adoption of Agile Software Development from the Perspective of Organizational Change Theories

Professorship
Software Business and Engineering

Professorship Code
T-76

Supervisor
Reijo Sulonen Dr.Sc. (Tech)

Instructors
Kristian Rautiainen Lic. (Tech), Jouni Virtaharju M.Sc. (Tech)

The goal of this master's thesis was to find out how organizations could more easily adopt Agile software development methods and Agile thinking behind them. The main focus was on finding out the challenges faced during the transition and producing suggestions about how these challenges could be tackled. The situation was examined from the perspective of an external change agent.

The thesis consists of two parts: a theoretical part examining various organizational and social psychological theories and their connections to Agile software development, and a qualitative research part based on the agile adoption experiences of the three case companies. By combining the theories to the empirical data, the suggestions for implementing Agile adoption were constructed.

Most found challenges during Agile adoption were connected to the fundamental concepts of software development, organizational issues and changing environment of individual employees. In order to avoid these challenges the Agile adoption should be people-driven, goal oriented and incremental. The change agent supporting the adoption should act as catalyst to enable organizational discourse. His main goal should be to help his clients help themselves by providing them domain knowledge about Agile methods and teaching them how continuously reflect and improve their organization.

In addition to the suggestions, the study showed that various organizational and social psychological theories can be successfully applied in the context of software development. This finding encourages further research in the area.

Keywords
Agile, agile methods, agile adoption, change theories, consultation, change agent, consultant

TEKNILLINEN KORKEAKOULU Tietotekniikan osasto	DIPLOMITYÖN TIIVISTELMÄ
Tekijä Jouni Hartikainen	Päiväys 9.5.2008
	Sivumäärä 90
Työn nimi Ketterien ohjelmistokehitysmenetelmien käyttöönotto organisaation muutosteorioiden näkökulmasta	
Professuuri Ohjelmistoliiketoiminta ja -tuotanto	Koodi T-76
Työn valvoja Professori Reijo Sulonen	
Työn ohjaajat TkL Kristian Rautiainen, DI Jouni Virtaharju	

Tämän diplomityön tavoitteena on tutkia kuinka organisaatiot voisivat helpommin ottaa käyttöön ketteriä ohjelmistonkehitysmenetelmiä sekä omaksua ajattelua niiden taustalla. Työssä keskitytään tunnistamaan niitä haasteita, joita ketterien menetelmien käyttöönotossa esiintyy, sekä tarjoamaan ratkaisuehdotuksia niihin. Tilannetta tarkastellaan ulkoisen muutosagentin näkökulmasta.

Tutkimus koostuu kahdesta osasta: teoreettisesta osasta, jossa tarkastellaan useita organisaatio ja sosiaalipsykologisia teorioita sekä niiden yhteyksiä ketteriin ohjelmistonkehitysmenetelmiin, sekä kvalitatiivisesta tutkimuksesta, joka perustuu kolmen yrityksen kokemuksiin ketterien menetelmien käyttöönotosta. Yhdistämällä ensimmäisen osan teoriat empiiriseen aineistoon tutkimus tarjoaa joukon ehdotuksia siihen, kuinka ketteriä menetelmiä tulisi ottaa käyttöön organisaatioissa.

Suurin osa tunnistetuista haasteista ketterien menetelmien käyttöönoton yhteydessä liittyivät perustavanlaatuisiin käsityksiin ohjelmistotuotannosta, organisatorisiin seikkoihin sekä yksittäisten työntekijöiden muuttuvaan ympäristöön. Jotta näitä haasteita voitaisiin välttää, tulisi ketterien menetelmien käyttöönoton olla ihmisvetoista, päämäärätietoista ja vähittäistä. Käyttöönottoa tukevan muutosagentin tulisi luoda ja ylläpitää organisatorista keskustelua. Hänen päätehtävänään tulisi olla auttaa muuttuvaa organisaatiota auttamaan itseään. Tähän hänen tulisi pyrkiä tarjoamalla tietämystään ketteristä menetelmistä sekä opettamalla asiakasorganisaatiota kehittymään jatkossa itsenäisesti.

Ratkaisuehdotusten lisäksi tutkimus osoitti, että organisaatiotieteiden ja sosiaalipsykologian teorioita voidaan onnistuneesti soveltaa ohjelmistokehityksen alalla. Tämä rohkaisee jatkotutkimuksiin aiheen piirissä.

Avainsanat
Ketterät menetelmät, ketterien menetelmien käyttöönotto, muutosteoriat, konsultointi, muutosagentti, konsultti

Acknowledgements

When I was starting my thesis project in autumn 2007 conducting a study in the magnitude of a master's thesis seemed an overwhelming task. Also, at first, the subject for the thesis was not much more than a vague idea of something about Agile software development and consultancy. However, with the guidance and support from my colleagues at Reaktor Innovations my goal became gradually clearer and simultaneously my personal fascination about the subject grew exponentially.

When searching for an instructor to my work I contacted Jouni Virtaharju from the Department of Industrial Engineering and Management as I had already received excellent guidance from him with my previous studies. To my delight he accepted my proposition. Later, also Reijo Sulonen and Kristian Rautiainen came along helping me to finalize the study.

Naturally I am very grateful to all you who helped me during the thesis process.

My colleagues Hannu, Panu & Timo: You gave me the guidance and support in crystallizing the focus of the study as well as motivated me during the journey. Your feedback was invaluable and kept me on track! Thank you for that!

Jouni: I truly owe you a big one. You know how hard it was at some points, but I could still always count on your support. The long discussions about relevant theories and their applications were extremely helpful and I enjoyed them much. Thank you for showing me the way!

Reijo and Kristian: You helped me to get my project to its goal. You also provided lots of valuable feedback and helped me to make this thesis better. Thank you!

All Reaktorians: It has been a privilege to work with you! I have had dozens of discussions with you about my thesis and all those helped me go forward. I am also very grateful about the support you showed me when I shared my findings and ideas!

I would also like to thank the representatives of all case companies. You made this study possible by giving me your time and resources.

I wish to also thank my family and friends who have always helped and supported me. Especially grateful I am to you, dear Nora: You gave me the courage and support I needed to finalize the thesis as well as always brought light to those dark days.

Jouni Hartikainen,

Espoo, 9 May 2008

LIST OF CONTENTS

1 INTRODUCTION

We live in the age of information. It is the vast amount of information and our current ability to transfer it with such an enormous efficiency that also makes our world more complicated and fast-paced than ever before.

Current information society combined with the global economy sets an ever changing playground for today's businesses to arise, live and fall in. As the context in which the companies exist is transforming, it has become imperative for them to constantly change as well in order to survive. Often the ability to adapt to new situations faster than competitors might be the question between big profits and great losses.

This thesis is about change as well. In particular, it is about change in organizations. It is about how organizations can be changed and developed more efficiently and with less pain that is so often associated with change.

In addition to change, the thesis is also about information technology and computer science. These create the surrounding context for the study as well as provide means to more closely focus on one particular kind of organizational change IT companies are currently increasingly willing to undertake.

The aforementioned change concerns the way how software is developed and how software related companies should be run. The change is as much about fundamentally transforming the thinking and principles driving the software development as actually starting to use new practices and tools to do the job. The new ways of thinking are entitled Agile principles and ways of doing are called Agile methods.

The third major contributor to the content of this study is a company called Reaktor Innovations. Reaktor is a consultancy company that specializes in Agile and among other things helps its clients to change their organizations and adopt new Agile ways of thinking and working. As Reaktor is in the role of a client for the study and the employer of the author, its context is used as a basis for suggestions and findings of the study.

1.1 Motivation

There are three motives behind the study. The first motivation for the thesis is the limited perspective from which Agile themes mostly are approached. Commonly Agile is seen from technical viewpoints with some commercial tones, but true interdisciplinary perspective incorporating psychological, philosophical, organization theoretical, commercial as well as technical views is somewhat missing. Consequently, this study draws its theoretical backgrounds mainly from organizational theories and behavioral sciences leaving the technical perspective with lesser significance.

The second driver for the study comes from its connection to Reaktor Innovations. Recently Reaktor has been increasingly asked to coach and help different kinds of organizations in their objectives of transitioning towards more Agile operations. To be better at this, we at Reaktor want to understand how the transition is affecting the organizations, how it could be managed and how we as external consultants could facilitate it more efficiently.

The last reason for the study is personated to the author. I personally believe that many of the challenges with which the IT business is struggling originate from poor understanding of how different the computer science and software development fields are from more matured engineering practices. I strongly believe, that by concentrating more energy on people instead of technology great achievements could be harvested and improvements made both in the business sense as well as in employee well-being.

1.2 Goals

The main goal of the study is to find out how organizations could more easily transform into Agile thinking and Agile methods. As the study is conducted in the context of a consultancy company, the main focus is set on how an external change agent could best help organizations in their pursue of adopting Agile.

The secondary goal of the study is to widen the commonly used theoretical background of Agile research more towards organization theories and behavior sciences. With this the study aims to provide Agile change agents with new insights

into their field of work. Hopefully these, maybe not so commonly referred theories, offer them new food for thought and benefit them and ultimately their clients.

The following research questions arise from the goals mentioned above:

1. Why is transformation from the traditional software development methods to Agile ones so difficult?

2. How should an Agile transformation be done in order to avoid the difficulties?

3. How could an external consultant best help his client organization during the Agile transformation?

1.3 Research Methods

This study is based on Case Study research method (Patton 2001). Three cases were constructed based on interviews in the case companies. These cases were later on analyzed from the perspective of selected theoretical frameworks.

With such setting the study is purely a qualitative one. By using qualitative research methods the study can penetrate more deeply into the subject and better increase the depth of gathered understanding (Patton 2001). At the same time the generalizability is decreased, however (Patton 2001). For these reasons, the study aims on understanding the case situations and providing suggestions based on them.

As Case Studies are always context sensitive, investigating the general applicability of Agile methods in different situations is knowingly left out of the study's scope. The applicability of Agile methods is left out of study's scope also in case contexts. Practically this means that the study does not evaluate the case companies from the perspective of whether Agile methods should be used in their situation or not, nor does it try to prove that solutions and suggestions found during the study would necessarily result in successes in them. With this definition the focus of the study is more on what should be done when the decision of adopting Agile methods has already been made.

If one is interested in the applicability of Agile methods in different situations, for example Boehm and Turner (Boehm 2003, Boehm, Turner 2003, Boehm, Turner 2003) provide discussion about it.

1.4 Structure of the Study

This thesis is divided into seven chapters. After this introductory chapter, a quick description of Agile software development is presented in Chapter 2. Following it, in Chapter 3, a description of study's context is shown. Used theoretical frameworks are presented in Chapter 4 and collected research material and utilized research methods are described in Chapter 5. Chapter 6 contains the results of the study and finally Chapter 6.3 presents the conclusions and suggestions for further research.

2 AGILE SOFTWARE DEVELOPMENT

Traditionally in business world software has been developed using similar methods than in other, more mature, engineering disciplines such as construction. A comparison to bridge building is commonly used. When one starts to build a bridge, one first assesses the environment and requirements for the bridge: how wide is the river to be crossed, how much traffic the bridge should handle, against what kind of weather it should hold out etc. After the initial analysis, planning containing schedules, budgets, resources, materials etc. is done. When all plans are ready, the bridge is built following them. The construction starts from the poles dug into the bottom of the river and ends to the deck of the bridge with railing, lighting and such. When the bridge is ready, it is opened to the public. Using these methods most of our bridge building projects are pretty successful and very rarely half-done bridge is dismantled or just abandoned standing there unfinished. Unfortunately, this is not too uncommon in the software world at all.

Even when using exactly the same methodologies and principles than construction engineers, great many of our software projects fail completely or at least miss their budgets and/or schedules. For example in their review of different software project success studies Moløkken et al. conclude that "it seems as if most projects (60-80%) are completed over budget and/or schedule" (Moløkken-Østvold, Jørgensen 2003). They also claim that an average cost overrun for a software project exceeding its budget is around 30-40%.

To tackle to problems of such plan-driven linear processes, many other software development models have also been suggested. For example as early as in 1970's Mills (Mills 1976) suggested that software should be implemented incrementally. Later Boehm suggested that development should be iterative and risk-driven (Boehm 1988). Even if such incremental and iterative development methodologies have a long history (Larman, Basili 2003), they have not been too widely used in software business until recently, when so called Agile methods have gained more ground.

Arguably the most famous definition for Agile methods is the Agile Manifesto[1] published by a group of software practitioners in 2001. The manifesto has four distinctive points (explanations adapted from (Abrahamsson et al. 2002)):

1. Individuals and interactions over processes and tools

    - People, communication, human roles and self-organizing teams should be more important than formal contracts, forced processes and specified development tools.

2. Working software over comprehensive documentation

    - The main measurement of progress is working software of which new releases are produced with steady and predetermined pace. Effort is put on keeping the software as simple and elegant as possible to avoid problems of extra complexity and need for superfluous documentation.

3. Customer collaboration over contract negotiation

    - Good and healthy relationship with the customer is held vital for the success of the project and it should be maintained. Contracts are needed, but the spirit of the negotiations should be of achievement not of strict competition.

4. Responding to change over following a plan

    - The collaborative group of development (including all relevant stakeholders) should be prepared to, be responsible for and be empowered to make all the necessary changes needed during the project's execution in order to steer it towards its goal.

In addition to Agile Manifesto, many other descriptions of Agile have been used. For example Abrahamsson et al. (Abrahamsson et al. 2002) describe an Agile method being a method that is 1) incremental (small software releases with rapid cycles), 2) cooperative (customer and developers working constantly together with close

[1] agilemanifesto.org and www.agilealliance.org, (26.10.2007)

communication), 3) straightforward (the method itself is easy to learn and to modify, well documented) and 4) adaptive (able to make last moment changes).

The above mentioned descriptions present Agile software development from quite abstract perspective concentrating mainly on values behind Agile thinking. However, Agile can be examined also at more concrete level where these values are manifested in practical means about how software should be developed. At this level Agile methodologies such as Scrum (Schwaber 2004), Extreme Programming (Beck 2000), Crystal Clear (Cockburn 2004) and DSDM (Stapleton 1997) provide good means about how to turn the Agile values into practice.

Because of such variance about what Agile actually is, companies adopting it might have hard time figuring out what they should adopt or how agile they should be (Boehm 2002). Some companies might aim for adopting only certain specific practices from Agile methodologies such as test-driven development or continuous integration, whereas others might want to get the "whole thing" from Agile values to hands-on practices and extend Agile to even outside software development field (e.g. (Baker 2007, Steindl 2005)).

Naturally the more about Agile world is tried to absorb, the bigger the needed change in the organization usually is. When Agile is adopted in its fullest, it might not only require changing the ways of working, but also changing the ways of thinking (e.g. (Seffernick 2007)). Commonly during such changes also organizational structures need to be changed (e.g. (Cloke 2007)). Additionally, such fundamental concepts as what is productive work and how it is measured, how to decide when something is done and who is responsible for what, are frequently under change (Cohn, Ford 2003). These essential concepts demonstrate well how deep an Agile change is – and should be – when Agile is fully applied.

In this thesis the main focus is on this kind of deeper change where a company is striving for adopting Agile in a wider scale and also at the levels of values and principles. In the text the term "Agile transformation" is used for this kind of adoption.

3 CONTEXT OF THE STUDY

The context of the study is described in this chapter. The context is here understood in a quite wide meaning involving all different pieces affecting the study's viewpoint and goals as well as selection of used theories. To keep the various areas of context clear and to point out all relevant relations between its different pieces, the context is presented by introducing all general concepts first and after that deepening them step-by-step.

We start opening up the context by defining the three main concepts. As the study is about how to implement an organizational change, the change itself is the first central concept. It also gets most of our attention during the study.

The second main concept is the group of existing theories that are used to understand change. The selection of theories and the viewpoints they provide connect them very closely to the concept of change.

In addition to change and theories, the environment in which the study is conducted greatly affects the selection of used focus and goals. The environment also affects the selection of used theories. With the environment we also define much about how we actually see the change and about which aspects of it we are interested in. As such, the environment becomes our third main concept. The three main concepts and their relations are presented in Figure 1.

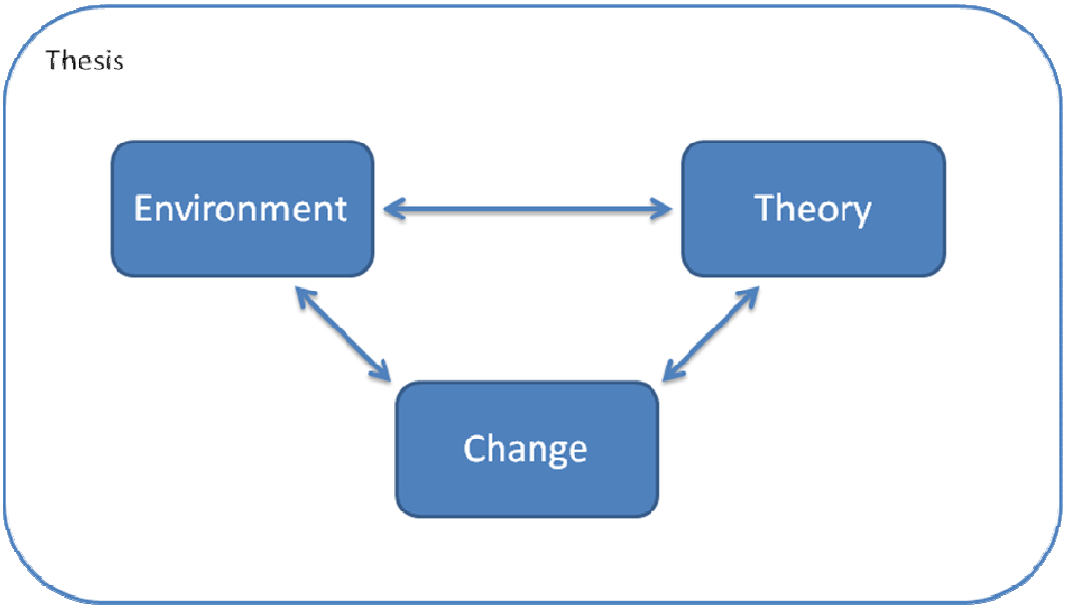

Figure 1: Main concepts of the thesis

To get better understanding of the more detailed levels of the context, we need to open up the main concepts and study how they form. This is done in the following sections.

3.1 Environment

The main environmental aspect affecting the study is author's working relationship to Reaktor Innovations. Practically this means that the main goals for the study are set from the standpoint of Reaktor to guide the work towards finding out information that is relevant to its business. This relation also means that most of the practical propositions and findings are based on Reaktor's context and thus cannot necessarily be well generalized to other situations.

Lots of knowledge about Agile and how it is seen also derives from Reaktor's consultant's understanding of it. This naturally steers the study to find solutions to how to implement Agile thinking and methods as they are understood in Reaktor.

Used case studies were also gathered using connections of Reaktor's employees. Naturally our own consultants were also interviewed.

Figure 2 shows the environmental parts and their connections.

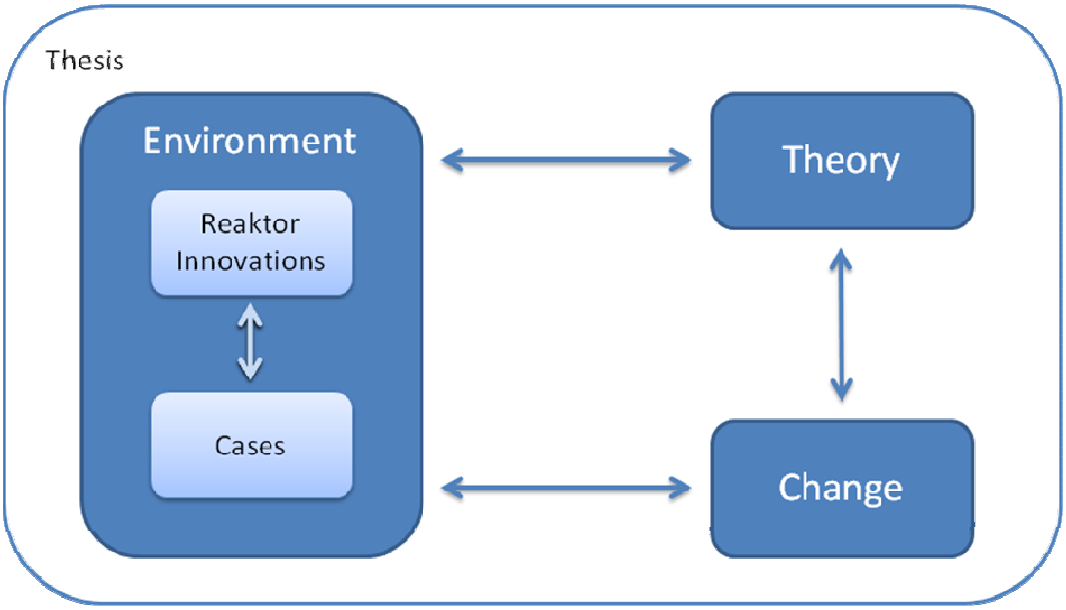

Figure 2: Environmental concepts opened

3.2 Change

Change as a concept is the essence of the study. To better understand different characteristics of change, a model by Buchanan & Boddy is used (Buchanan, Boddy 1992). Their model states, that in order for one to understand organizational change, four different parts of the phenomenon must be studied. These parts are content, control, program and context of the change. In addition to these features, the concept of change agent must be defined as it has an integral relation to the study's point of view.

By content part of change Buchanan & Boddy mean the subject matter of change. With organizational change in business world, this normally stands for the business of the company or work at hand. For example in case of changing a company's payroll processes, the content of change would be its payroll system, how it is operated and what kind of knowledge is needed to calculate salaries and other rewards. In case of this study, the content of change is naturally Agile software development. The selection of Agile as main content derives directly from the environment of the study and from Reaktor's perspective, as its change management consulting is mainly focused on Agile transformation.

According to Buchanan & Boddy, the control aspect of change is about "traditional project management" incorporating all practical project management tools (critical-

path analysis, work breakdown structures, timetabling, budgeting etc.) and the thinking behind them. Even though this view might be very much needed during a change project, for this study the view is considered mainly as providing different kinds of tools for change agent and is thus not much covered. There is also a multitude of different writings about the subject (e.g. (Project Management Institute 2004, Turner 1999)) that provide further information about it already.

The process view of change is about how to actually implement it. It contains all perspectives of human interaction (communication, motivation, consultation, management of resistance, etc.) and organizational considerations of the change. As the main goal of the study was to find out how transition to Agile could be done, this exact field becomes the main focus of the thesis. Also all used theories were selected to describe this area and support change agent's work in it.

By change context Buchanan & Boddy mean the environment in which the change is taking place. In this study the main change contexts are the three case companies. However, as the study contains material also from Reaktor's consultants, the study does not base purely on these cases.

In addition to four aspects of change already presented, the concept of change agent is relevant to the study. Here the change agent is seen as an external consultant that is hired to help his/her client organization in its change endeavor. This definition comes directly from the study's relation to Reaktor. As practically all change consultation in Reaktor is done in an environment where Reaktor's consultant is acting as an external change agent in the client organization, this is a very natural choice. It is noteworthy, however, that in all case studies the change agent was not a Reaktor's consultant.

In Figure 3 the various concepts of change have been added to the context of the study.

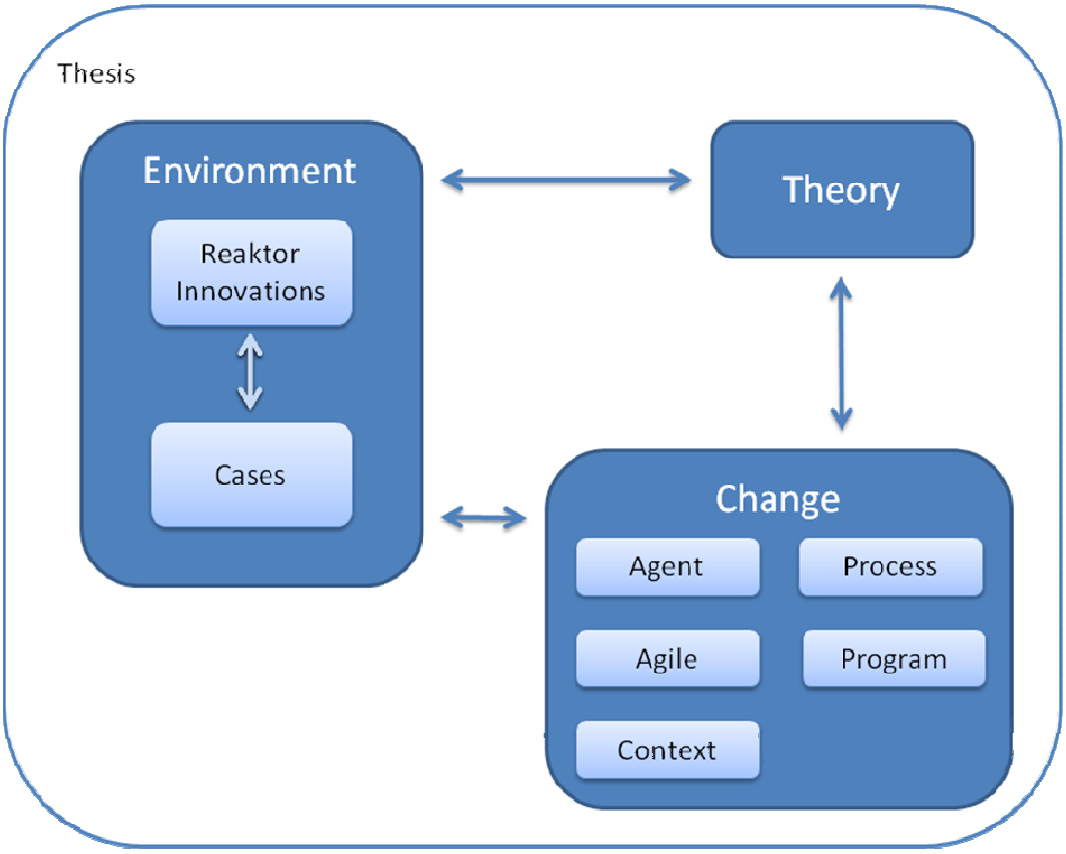

Figure 3: Change concepts opened

3.3 Theory

The theoretical perspective of the thesis provides a background and lenses through which the change is observed. The set of used theories were selected using four criteria: 1) Selected theories should match the objectives of the study and be compatible with the selected change content (Agile), 2) they should be able to produce practical results when combined with empirical data, 3) they should form a wide enough palette to help Reaktor's consultants to widen their theoretical understanding of organizational change, and finally 4) they should focus on process aspect of change as it is the main focus of the study.

The selected theories set on different levels of abstraction. At the highest level the theories of episodic and continuous change from Weick & Quinn are used (Weick, Quinn 1999). These theories provide two different pictures about change and in that way help us understand how it can be seen at the different levels of the organization. At more concrete level Kurt Lewin's theories (Lewin 1951) provide a stable framework with which the human change process can be understood. With the

additions of Schein (Schein 1999, Schein 2007), these theories provide a complete picture how an organization could be changed and the change could be stabilized for future. Lewin's theories also provide a central core on how change is seen in the episodic context.

Schein's process consultation theory (Schein 1999) is used at the most concrete level of the theories. Process consultation focuses more on change agent's perspective and provides guidance about how the agent should see himself and his relationship to his client. In this way it complements Lewin's theories.

The fourth used theoretical framework providing quite different perspective about change is social constructivism and language of change (Juuti, Lindström 1995, Juuti, Rannikko & Saarikoski 2004, Palmer, Hardy 2000). As social constructivism is applicable to various levels of change, it cuts through all other three theories and supports them. Used theories are presented in greater detail in Chapter 4.

Selected theories and their relationship to other aspects of the study are presented in Figure 4.

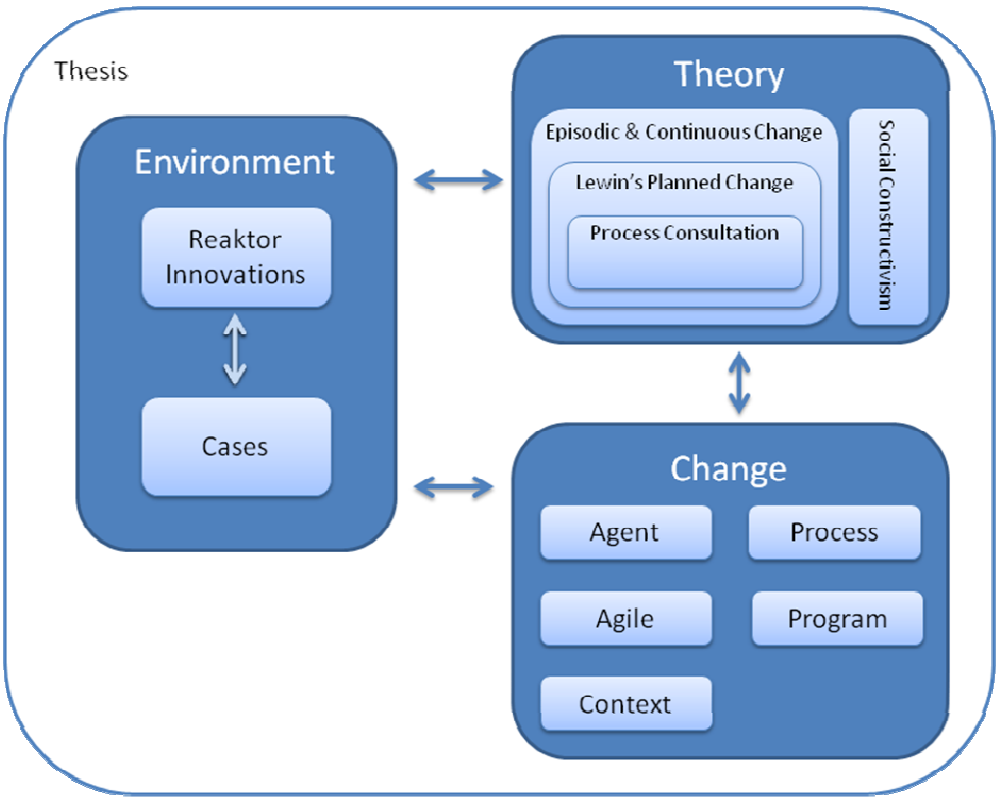

Figure 4: Theoretical perspective opened

4 THEORETICAL BACKGROUND

In this chapter the various theoretical frameworks are presented and connected to Agile transformation.

4.1 Episodic vs. Continuous Change

Weick & Quinn (Weick, Quinn 1999) describe two perspectives of change in organizations, that can also be used to characterize Agile transformation. Based on the tempo of change, they call the views episodic and continuous change.

From the episodic view the change is infrequent, discontinuous and intentional. It is planned in advance, executed according to plans and lasts for certain period of time after which the situation is stabilized again. In this way the change can be seen as a transformation between the steady initial and end states. (Weick, Quinn 1999) For example after Agile transformation, people could refer to times "before Agile" and "after Agile" as if the change would have been somehow instantaneous.

According to episodic view, the change is also top-driven and happens when an organization is driven out of its normal stable equilibrium into a more unstable state during which new ways of thinking can emerge. The shift into the unstable state is usually triggered by some distinct event. Such events can either be internal (e.g. personnel change in key position or out bursting dissatisfaction to working environment) or external (e.g. technological or legislative change, bad customer feedback). (Weick, Quinn 1999)

From the process perspective episodic change is seen as a linear three phased process. The first phase, unfreezing, moves the organization from its equilibrium to the unstable state in which the change can be realized during the second, implementation phase. After the change has occurred, the equilibrium is again returned during refreezing phase. (Weick, Quinn 1999) Practically such process could for example mean bringing up current problems into common knowledge, reorganizing work and processes to solve the problems and the continuing development using the new ways of working.

In addition to giving change a linear nature, episodic perspective describes organizational change from macro level. From this perspective, higher level

organizational issues such as organizational structures, business processes, division of operations into business units etc. dominate the view. Change is seen achievable by restructuring or reorganizing these in a planned and controlled way. (Weick, Quinn 1999) In the Agile transformation context this could for example mean changing organization structure to better support individual empowered teams or creating new reward systems to support new software development processes.

A change agent's role from the episodic perspective is the one of the prime mover of the change. The agent's responsibility is to keep wheels rolling and focus on removing obstacles of change. The agent also builds commitment and sets goals for the organization as well as uses language to provide alternative schemas for the situation to endorse the change. (Weick, Quinn 1999)

In contrast to episodic view, the change from continuous perspective is ongoing, evolving and cumulative. According to this view, change is not planned beforehand – more like it "just happens". Even considerable changes take place gradually via small continuous adjustments that by themselves are aimed towards improving work processes in a smaller scale. Gradual change and improvement is so habitual that it is hardly noticed. Change becomes organization's way of working. (Weick, Quinn 1999)

When organizational change is examined from the continuous change perspective, the micro level phenomena start to dominate the view. With this view, constant improvements achieved by everyday learning and feedback are seen as the key drivers of the change. The change is seen as a pattern of endless modifications in work processes that cumulatively improve the whole organization. The focus of the change is on local and close issues and it is driven by lower levels of the organization. (Weick, Quinn 1999)

Process-wise the change from continuous perspective can be described as freezing, rebalancing and unfreezing. During freezing the current situation is made visible and observable, so that it can be reinterpreted, relabeled and resequenced in rebalancing phase. Finally unfreezing resumes the improvisation and continuous improvement. (Weick, Quinn 1999) In Agile consultation context this could for example mean telling other consultants a story about some customer case. When telling his story,

the teller freezes the things he has learned from it by codifying his experiences into an explicit language. Now the codified experience can be interpreted and analyzed by the group, and it can be used to learn further (rebalancing phase). After learning normal action goes on, but with slightly changed understanding about the situation (unfreezing).

In Agile transformation context this kind of perspective focuses on issues at the team and individual levels. For example software development methods are not seen as static, but instead they are just best guesses of the optimal ways of working at the time and should be under constant inspection and improvement. Same principles can also be seen in the self-organizing ways of dividing and managing work in Agile teams.

In continuous change environment the change agent does not need to worry that much about getting the change started. The agent can thus concentrate on helping to build common understanding of issues at hand and providing more visibility throughout the organization. This can be done with effective usage of language and dialogue. In this way the agent does not actually drive the change, but gives it a more uniform direction. The role of the change agent from continuous perspective is very much of a facilitator or a coach that helps others to better achieve their goals and at the same time transfers information inside the organization. (Weick, Quinn 1999)

Even if episodic and continuous perspectives paint very different pictures of change, one cannot clearly categorize changes into episodic or continuous ones. Rather, the two perspectives act like windows through which the change can be observed. Even when looking at the same change, the perceived images are very different. However, neither is more “precise” or “correct” than the other. (Weick, Quinn 1999)

Understanding the two different perspectives of change and their relationship not as exclusive, but complementary, is very important to external change agents. By seeing how change can be perceived at the different levels or from the different perspectives, they are better able to conceptualize and frame what they see and hear during Agile transformation. With this understanding the change agents can also more easily concentrate on the right things at the right level of abstraction and also at the right level of the organization.

4.2 Lewin's Change Theories

Kurt Lewin has for a reason been praised and admired for his achievements in social sciences. Even now, well over 50 years after his time, his theories provide valuable perspectives on human behavior and organizations. In this thesis Lewin's organizational change theories (Burnes 2004, Lewin 1951) are used as a framework for understanding how individuals and organizations perceive change and how this can be used to help them in adopting Agile methods. Edgar Schein has also used Lewin's theories widely and provides relevant additions and more concrete bindings to practical implementations of Lewin's work (Schein 2007). These views are also presented in this section as they relate quite inseparably to Lewin's work.

Lewin's theory of planned change consists of four parts that sometimes are treated separately, but according to Burnes (Burnes 2004), should rather be seen as parts of a whole strengthening and supporting each other. The four parts described here are Field Theory, Group Dynamics, Action Research and 3-Step Model of Change.

4.2.1 Field Theory

Field Theory states that at each situation the behavior of individuals and groups is a function of forces from group's environment that are in equilibrium. Consequently, all changes in behavior are effects of changes in these environmental forces. If one was able to discover and understand these forces, it would not only be possible to understand why groups behave as they do, but also change their behavior by appropriately weakening the forces resisting the change and strengthening the ones supporting the change. Generally Lewin thought that change would be quite gradual and slow, but under certain circumstances (e.g. personal, organizational or societal crisis) the forces acting on groups could shift radically and the existing status quo could disappear quickly thus leaving room for new patterns of activity to emerge and new equilibrium to form. (Burnes 2004)

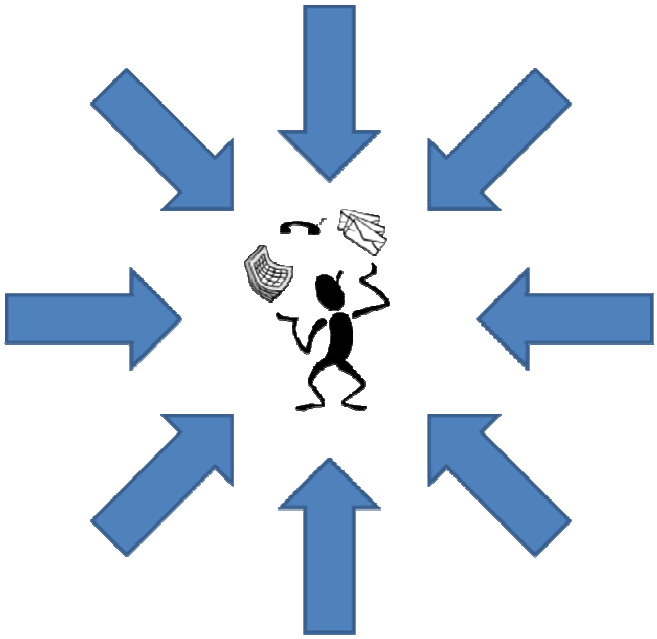

Figure 5: Field Theory – behavior is a function of environmental forces

Field Theory can explain for example how a reward system based on traditional software development thinking might resist Agile transformation. If the old reward system for example focuses on honoring individual achievements such as the amount of features implemented per developer, it creates a very strong force against Agile principles that emphasize playing well in a team and succeeding together. So, in order to help the change, this resisting force should be weakened. This could be achieved for example by changing the reward system to promote co-operation and achievement as a team for example by measuring how well the whole team can reach its targets or how well it is able to estimate its progress.

4.2.2 Group Dynamics

By Group Dynamics Lewin wants to emphasize the group behavior over individual. Group Dynamics is based on the ideas of Field Theory and its main premise is that in each group there are norms, roles, processes and culture that create strong forces that influence all group members. Lewin states that these forces are so powerful that if an individual leaves the group, experiences some change outside it and then returns back, the achieved change will be diminished by the group norms, if it does not fit to them. (Burnes 2004)

According to Lewin, the main implication of Group Dynamics is that one should not try to change individuals, but groups. The change should also be directed towards

group level issues such as norms, roles, interactions and socialization processes, because changing these would be the easiest way to achieve the desired behavior of the individuals. (Burnes 2004)

In Agile transformation context Group Dynamics emphasizes the role of teams in defining the behavior of individuals. It also advices change agents to for example introduce Agile methods or tools at the team level, not at the individual. Practically for example training should be organized in a way that the whole team could participate and thus team's norms and culture would be exposed to change during the training.

4.2.3 Action Research

Action Research is encapsulated into the famous quote of Lewin: "One cannot understand an organization without trying to change it" (Burnes 2004). Based on this, Action Research challenges a very common change management process, which maintains that a change agent should at first perform an objective diagnosis of the system being changed and after that, based on this diagnosis, create a set of interventions, that, when implemented, would ultimately lead to a desired change. However, with his statement, Lewin argues that there is no difference between diagnosis and intervention and they should be seen as one (Burnes 2004). Schein also supports this by indicating that everything a change agent does is an intervention that changes the observed system (Schein 1999). The result of all of this is that a change agent cannot get an objective picture of the client system, but the view he gets is always distorted and biased by his own presence and actions in the system.

Based on this, Action Research formed into a research and change management method that willingly involves both the members of the studied organization as well as the change agent himself to collaboratively work on how to better understand the situation and how to change it. During Action Research common understanding is created by implementing some changes and then feeding back the results of these attempts to the organization. Through this feedback loop the change process becomes a cyclical one, which starts by planning, continues with implementing the plans and ends in analyzing the results of the actions. Understanding created during the analysis then leads to new plans and cycle starts again. Important with Action Research is, that it emphasizes not the planned action, but the understanding and

learning created by *inspecting the results of the action*. Only through understanding the reasons that lead to witnessed results, true knowledge about the situation is gathered. Figure 6 presents the phases of Action Research. (Burnes 2004)

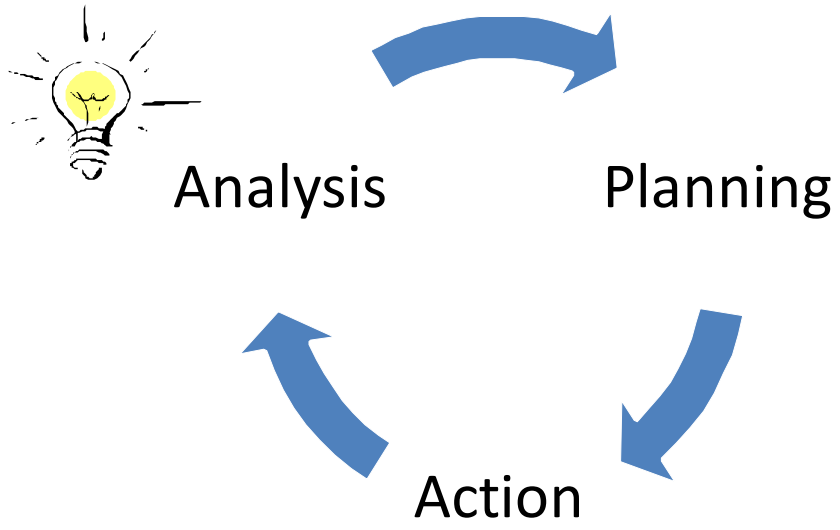

Figure 6: Action Research cycle

The cyclical characteristics of Action Research make it a good match to Agile transformation. As Agile methods are also inherently cyclical, the same nature of Action Research can be effectively used by incorporating change to the Agile cycles. For example a reflection session supported by the change agent could be organized after each development cycle. By analyzing the reflection results, the team gets better understanding about themselves and their work. At the same time also the change agent acquires more knowledge and can thus better support the team.

4.2.4 3-Step Model of Change

Last part of Lewin's theory package is undoubtedly also the most famous one. The 3-Step Model of change has been cited widely and influenced many authors throughout the field as it simply, yet effectively, describes how change can be achieved in human behavior (Burnes 2004). 3-Step Model states that any human change can be seen to consist of three phases: 1) Unfreezing during which motivation for change is created and old ways of doing are unlearned, 2) changing during which new ways are

learned and 3) refreezing during which new achieved state is established and slipping back to old habits is prevented.

The unfreezing step is based on Field Theory and on an assumption that there exists equilibrium of change supported by complex driving and restraining forces. Consequently, in order for change to happen, this equilibrium needs to be destabilized (unfrozen). Schein describes three processes through which the unfreezing can be achieved and willingness to change created. (Schein 2004, Schein 2007)

Schein states that "all forms of learning and change start with some form of dissatisfaction or frustration generated by data that disconfirms our expectations and hopes" (Schein 2007). Consequently, in order to start change, something disconfirming must be presented to create this dissatisfaction or frustration. Disconfirming information can be almost anything that for example shows that some of the organization's goals are not going to be met or some of its processes are not accomplished as they are supposed to be. This kind of information could for example be declining sales, increasing customer complaints, growing employee turnover etc. Whatever the reason behind the disconfirmation, it becomes the main driving force for the change. (Schein 2004)

Disconfirming information is not enough to create change, however, as we could ignore it, dismiss it as irrelevant to our goals, blame somebody else for it or just deny its validity. For example, declining sales could be rationalized to be an effect of minor recession and thus there would be no need to worry. So, in order for disconfirmation to have effect on us, the disconfirming information must be related to some important goals and must be cognitively undeniable. If we accept the information as valid and relevant to our goals, anxiety and guilt starts to build up and we feel some need for change. (Schein 2004, Schein 2007)

Even if we start to feel the need for change, the anxiety and guilt driving it could still be suppressed. This can happen if the needed change implies some threat to the changing group or person. This resisting force is called learning anxiety. It is caused by our fears of loosing our effectiveness, our self-esteem and maybe even our identity during the change. Learning anxiety is the primary resisting force against the

change and thus dealing with it properly is the key for successful change. (Schein 2004, Schein 2007)

Learning anxiety can be tackled by creating sufficient psychological safety that overcomes it. According to Schein "the true artistry of change management lies in the various kinds of tactics that change agents employ to create psychological safety". The psychological safety could be created e.g. by using support groups, having enough leadership and management support, proceeding through change step-by-step etc. (Schein 2007)

The previously described process can be illustrated using an example. Let us imagine a situation where a software company is starting to use Agile methods in its development work. Previously all employees have had their personal rooms, but new team rooms created in an open space area are now introduced to improve communication inside the teams. Some employees take the change happily, but others are more reluctant.

To help the change in this situation, some disconfirming information about staying in the old rooms needs to be created. This could for example be achieved by telling that all new projects will primarily be handed to the teams working in the open space area or that teams moving together will be awarded by a bonus which they can freely use to decorate and equip their new workspace.

When such information arrives, a reluctant engineer might start to think that if he does not move to a new workspace, he might be left with less important or challenging assignments that will not be so interesting and rewarding to work with. As he does not want to end up with such projects, the anxiety about staying away from the team rooms starts to build up and he starts to feel the need for change.

The engineer might not be ready for the change yet, however. Even if he sees the benefits of moving, he might still be held back by his learning anxiety. The employee could for example fear that his skills might not be on par with the others and that this will come up when working closely with them. He could also believe that he cannot work as efficiently in the open space, because there would be more interruptions. This again feels threatening, because the engineer wants to be seen as a good, disciplined, efficient and competent employee.

To help the engineer overcome his learning anxiety, some psychological safety needs to be created. The project manager responsible for the team could create it for example by stressing that the whole point of being in the same space is to utilize different skills of the team members. He could also emphasize that Agile methods require constant communication and thus being in the open space area actually makes the engineer more efficient, not less. The project manager could also propose that some part of the open space could be reserved for individual work, so that whenever somebody needs more space, it could be arranged.

Combining the concepts mentioned above, we have valid and accepted disconfirming information which creates anxiety and guilt that drive the change. At the same time learning anxiety tries to resist it, however. The learning anxiety can be reduced by creating enough psychological safety. Thus achieving change crystallizes in *balancing between creating big enough disconfirmation and providing sufficient psychological safety.* This interplay is presented in Figure 7. (Schein 2007)

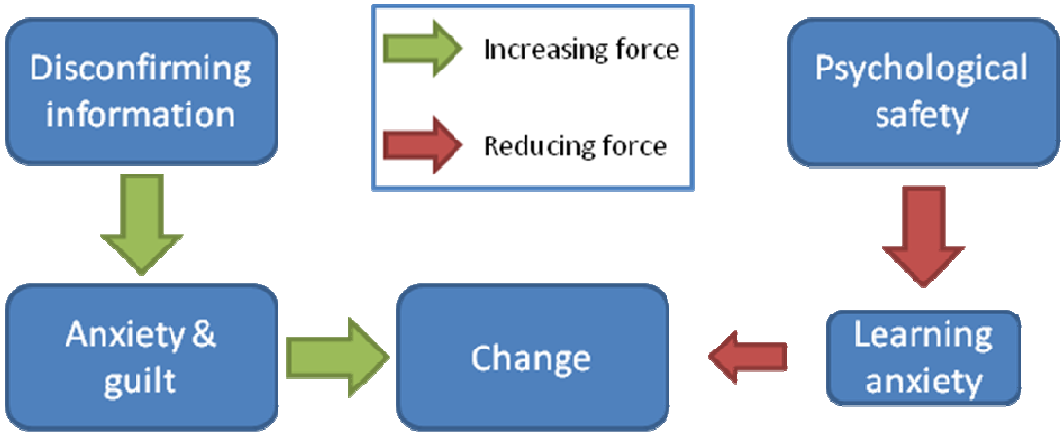

Figure 7: Forces increasing and reducing change

Using the aforementioned process the unfreezing part of 3-Step Model can be managed successfully. During the second phase of the model the actual change is taking place. Lewin points out that it is extremely hard to guide, predict and identify the outcomes of this phase and one should thus use more iterative trial and error based methods to search for all available options. Not surprisingly this is exactly the nature of Action Research.

As with unfreezing, Schein provides some more insight into the process of learning in the second phase of the 3-Step Model (Schein 2007). He states that learning

constitutes of taking in new information that changes our understanding of the situation. This can happen in three ways:

1. Semantic redefinition

    - We learn that words can mean something different than we assumed. Common example in Agile world is the definition of done. Quite often there is no mutual understanding about when some task is considered as done. However, by learning what others mean by it, the concept gets redefined and generates new understanding.

2. Cognitive broadening

    - We learn that a given concept can be much more broadly interpreted than we assumed. For example encountering a problem or an error in software is traditionally considered negative and feelings of guilt and blame are commonly connected to it. However, by broadening the concept we can admit that everybody makes mistakes and the same error can be seen as an opportunity to learn and to spread knowledge around how to avoid similar mistakes in the future. In this way the originally negative concept has got much wider meaning as a possibility to improve and change things.

3. New standards of evaluation

    - We learn that our evaluation criteria are not absolute and if we use different criteria, the judgments based on the criteria will change accordingly. For example traditionally software projects have been measured by their schedules, budgets and the degree of which the planned features of the software were implemented. If we change the criteria to consist of how much value the end customer gets or how many new versions of the software can be released in a year, the same projects measured earlier will show up in a different light. By understanding that our evaluation criteria are not carved in stone, we learn to see the situation in different ways.

Schein also presents two modes of learning, in which the three previously mentioned mechanisms work: 1) imitation and identification using positive or negative role models and 2) learning through trial and error.

Learning through imitation and identification happens when a learner is willing to learn and has opened himself for new information. According to Schein, the most basic way of getting new information is to be in a conversation process with someone else with different kinds of interpretations of some concept that the learner also has. By understanding these new interpretations cognitive restructuring happens and the learner is able to "see" or "hear" things from a new perspective. This process relates closely to socialization and role models, because new interpretations of the situation are often most influential, when they come from a party the learner is psychologically identified with. Learning using identification is a slippery road, however, because even though the role models themselves might be successful, the methods they are using might not fit to the learner's context, and thus will not stick for long. (Schein 2007)

If there is no good role model available or the learner has no idea where the answer or solution might be, the learner turns to learning via trial and error. Schein calls this process scanning. During scanning the learner searches new information by reading, travelling, having conversations, educating himself and so on. He might also just get a spontaneous insight about a solution. Whatever way the new information and cognitive broadening happens, after that, a period of trial and error follows, during which the new understanding is either reinforced or a new learning loop is created via disconfirmation. Schein points out, however, that if appropriate role models do exist at any time during the scanning, they will likely be used and learning via identification occurs instead. (Schein 2007)

After learning has taken place in the second phase of the 3-Step Model, the gained changes are stabilized and new equilibrium is created in the refreezing phase. Schein argues that the new learned behavior must fit to some degree to the rest of the person's behavior and personality or it will create more discomfort, which often leads to unlearning. The same applies also at the group level and for this reason both Lewin and Schein stress that groups should be changed together. For example organizing a training session about continuous integration does not help much if the

host organization does not allow developers to use CI tools or maintains an attitude that frequent need for integration is a sign of a bad design and just leads to more bugs in the system.

In Agile transition context refreezing means strengthening such organizational issues that support the new achieved state, and weakening the ones that would support old ways of doing. For example new laptop computers could be bought to replace old desktops to help development teams self-organize more efficiently. Also by for example not requiring a specific set of documentation regardless of project's nature, the team is supported to produce only such documents that will be useful and create value.

4.3 Process Consultation

By describing Edgar Schein's process consultation philosophy (Schein 1999) this section presents one concrete method on how Lewin's change theories can be applied in practice. Most of the attention is paid on Schein's views of what consultation fundamentally is and how it should be done in order for it to be most beneficial both for the consultant as well as his client. In Agile transformation context process consultation is relevant theory as very often an external change agent is in the role of a consultant supporting client's change, not primarily driving it himself.

Schein describes process consultation as being "a philosophy about and attitude toward the process of *helping* individuals, groups, organizations and communities" (Schein 1999). This highlights the main assumptions behind the process consultation: "One can only help a human system to help itself." By this Schein means that no matter how highly trained or competent any consultant could be, he can never know as much about client's situation than the client himself. Consequently, direct advices given in early stages of consultation are deemed to be based on beliefs, assumptions and stereotypes of the consultant rather than his real understanding of the situation at hand. These kinds of advices are often irrelevant, inaccurate, not executable or even rude in client's eyes.

To avoid this, the consultant must understand that he should not act as an expert in any other area than in how to be helpful. He should also strive to create a good and effective helping relationship with his client through which he is able to closely co-

operate with him (the client) during the consultation. To build and foster this kind of helping relationship is the ultimate goal of process consultation.

To be successfully able to build a helping relationship with a client a consultant needs to focus on two main issues: status balance between him and the client, and his own role as a consultant.

4.3.1 Status Balance between Consultant and Client

The status difference between the consultant and the client is normally not in balance in the beginning of the consultation process. This is due to the fact that the client has admitted that he has some kind of a problem and needs help with it. For example the client might admit that his software projects are often late or do not deliver the software that would satisfy its end users. By admitting his problems and assuming that the consultant is able to help with them, the client has made himself dependent on the consultant and empowered the consultant to take a leading role. (Schein 1999)

The problem with this situation, according to Schein, is that as long as the client feels vulnerable and not at equal level with the consultant, he is not willing to reveal all deepest parts and full complexity of his problems fearing that he might loose his face or the consultant might decline his views. In order to get these views exposed, the consultant must reestablish client's status and reduce the dependency felt by him. If this cannot be done, there is a great possibility that the client will not reveal all data, rejects suggestions or becomes defensive in other ways.

The consultant is able to elevate the client back to the same status level than himself using a method called active inquiry. Active inquiry is a dialogue process where the consultant steers the action while simultaneously maintaining supportive and listening stance in order to help the client regain his self-esteem. At the same time active inquiry acts as a tool to gather information about the situation and to involve the client in the process of diagnosis.

Active inquiry's idea is to let the client freely tell his views about the situation without having to confront the consultant's views until the needed mutual trust has been established between them. In this way the client never gets into the situation where he considers himself still being at the lower level than the consultant, but feels

anyway the urge to comment on consultant's propositions as they seem unreasonable from his point of view. At the same time active inquiry assures that the consultant is able to hear the client's version of the story before it is biased by his own actions.

To avoid confrontations and such biasing, active inquiry is started with pure inquiry. With pure inquiry the consultant encourages the client to tell his story *in his own words*. The types of questions the consultant could use are for example "Tell me what is going on?" or "What happened?" It is important that the consultant does not expose his own views yet. The consultant should also avoid directing the client's story with questions like "How the project manager reacted to this?" or "Could you have used Java here instead of Perl?" During the pure inquiry consultant should just support the client to tell what the situation is without any further analysis.

Once the client's story starts to slow down and further encouragements do not reveal any new information, the consultant might start using exploratory diagnostic inquiry. In this form of inquiry the consultant starts to more actively guide the dialogue towards the issues the client did not mention during the pure inquiry. This can be achieved by stating deliberate questions about the story and supporting the client to elaborate on them. If the client has for example spoken about change resistance in some parts of the organization, the consultant might ask why the client thought this resistance emerged or how he thinks it could be decreased. With these kinds of questions the consultant directs client's thinking. It is important to note, however, that the consultant is still not presenting his own views and all questions are based on the story already told. By acting this way the consultant has taken control over the process of the dialogue, but the content is still left for the client.

Finally during confrontive inquiry the consultant gives his own ideas to the client and starts to control the content of the dialogue as well. By presenting his own ideas the consultant forces the client to reframe his thinking and starts to push the client into the consultant's conceptual world. Confrontive inquiry could be for example a question "Do you think you could have used shorter iterations with this project?" or "Why did not you modify the project plan even if it was not up-to-date?" Also by providing new ideas ("Maybe we should organize a coding Dojo to clarify the meaning of test-driven development"), concepts ("Only the product owner should be able to change the priority of the user stories") and hypotheses ("I think the

marketing is not informed well enough about our progress") the consultant enables the client to widen his views about the matter at hand and thus may help him find new solutions to the problem. As such interventions are quite strong, the consultant should be very careful with them and be certain that the relationship between him and his client is already strong enough so that the client will not be offended.

During active inquiry the consultant affects the client in various ways; with pure inquiry maybe less, but with confrontive inquiry very much. In the spirit of Action Research the consultant should pay attention on the effects of his actions, because those reveal a great deal of information about how the client system functions. For example if the consultant interviews an engineer that becomes very defensive when talking about pair programming, could that mean that the engineer does not think pair programming is effective? Or could it be that he is concerned about his skills that would be revealed while working closely together? Or is he just uncomfortable being physically so close to his colleague? Deciphering this kind of information is crucial for the consultant to really understand how the client system works.

4.3.2 Consultant's Role

A consultant can have various different roles during the consultation process. According to Schein, it is very important to understand all these roles and to be able to successfully switch between them as needed. Even if changing roles can be done quickly, Schein maintains that a consultant can effectively play only one role at a time. The consultant must also be aware in which role he currently is, and recognize the consequences of it as well as understand the basic assumptions he then relies on.

As mentioned earlier, there is a status imbalance between a consultant and a client in the beginning of consultation. The client empowers the consultant and thus implicitly offers the consultant a role of an expert. As this happens quite automatically, an expert role is a very common role for consultants.

Normally during expert consultation a consultant is invited to organization, briefed about the situation and asked for help. After that, the consultant starts gathering information and studying the situation in order to finally provide the client with some kind of results report with suggestions on how to proceed. It is then up to the client to implement the suggestions. An example of this kind of consultation is a program

manager who wants to know how to re-organize his people during Agile transformation in order to help team building. A manager might go for some IT consultation company and hire consultants to assess the situation and to provide proposals how to do the reorganizing.

The expert model of consultation is based on an assumption that the client wants to purchase some information or advice from a consultant he is not able to provide for himself. (In our example the program manager could not figure out how the new organization chart should look like.) The client is assumed to know what information he is looking for, and that the consultant is able to provide this information. Schein argues that in order for pure expert consultation to be successful, the following requirements must be met:

1. Client has to have correctly diagnosed his needs
2. Client has to have communicated his needs correctly to the consultant
3. Client has to have accurately assessed the consultant's capabilities of providing needed information or service
4. Client has to have properly thought through the consequences of having a consultant gathering information, or implementing the changes the consultant might suggest
5. There has to exist an external reality that the consultant can objectively study and gain knowledge about that will be useful to the client (In the example the consultant should be able to carry out an objective study about how the organization is currently organized, what kind of interaction exists in the organization and how all this should be changed to help team building)

Given the extent of these requirements, it is not a surprise that quite often not enough of them are fulfilled and thus the expert mode of consultation leads into problems. Very often these problems lead to a client not implementing the suggestions provided by the consultants and consultation in that sense fails.

It is also noteworthy, that by using expert consultation model the client gives a great amount of power to the consultant and becomes dependant on his suggestions and

information. This dependency and discomfort might well bring up conscious or unconscious resistance against the consultant in later stages of the process. By having so much power the consultant is also very tempted to sell whatever he has used to or is good at – even unconsciously. In extreme cases the consultant might even misuse his power in order to sell certain products or services he for example gets provisions from. IT world is famous about these kinds of "selling consultants" who can make any client's problem look exactly like one that the product he is offering can solve.

When the client does not exactly know what kind of information or service he needs, but still senses that something is wrong, another consultation model called doctor–patient model can be used. In this model a client hires consultants to "check them over" – to discover some hidden problems and to find a cure for them. For example the previously mentioned program manager might have noticed some signs of change resistance during the transformation and is wondering why such happens and how to decrease it. In order to find out what is going on, he might hire consultants to study why some workers are reluctant to adopt new methods. He also expects the consultants to be able to offer some solutions to problems they might find out.

In doctor–patient model consultants can use different kinds of survey methods and analysis tools to try and find out what could be the main problems behind observed phenomena. For example different kinds of interviews, psychological tests, inquiries, group works etc. could be used. Once the problem has been found, the process continues like in the expert model: find proper suggestions and offer them to the client for implementation.

The doctor–patient model is very attractive to consultants as it provides them with even more power than the expert model. Not only is the consultant now empowered to provide solutions, but he is now authorized to formalize also the problem in the first place! For these reasons many consultants sincerely believe that the doctor–patient is the only "proper" way of doing consultation, because in this way they can give as much of their knowledge to the client as possible and thus believe that the client gets best return for his money.

However, as with the expert model, the doctor–patient model has its inherent problems that very often lead to confrontations between the consultant and the client

as well as to rejection of consultant's advices. The biggest difficulty, according to Schein, is the assumption that a consultant could on his own create an accurate and objective diagnosis of the client organization. As both Lewin and Schein state, the consultant always affects the client system during this kind of diagnosis. For example if an external consultant enters a research and development department of some organization and starts delivering out questionnaires about how employees do their job, there is an immediate affect: Engineers start wondering why the consultant came. Do we have a problem? Is there going to be some re-organizing? Maybe layoffs? Should I try to please the data collector and hide problems or should I try to exaggerate them so that they would be noticed?

Another problem with doctor–patient model is the lack of communication between the consultant and the client. During the diagnosis phase the client just waits passively, and when the consultant comes back with his suggestions, they live in completely different kinds of realities about in which state the organization is. This difference in realities might make consultant's suggestions seem totally irrelevant for the client and he is thus inclined to deny the diagnosis and proposals based on it.

The last major problem with the doctor–patient model is that even if the consultant could convince the client about his diagnosis and suggestions, the client might not be able to implement the recommended changes. According to Schein, this might be actually the most common problem in organizational contexts. For an external consultant it might be quite obvious what should be done in the organization, but for example organization culture, its structures or organizational politics may prevent the recommendations being fulfilled. Unfortunately the consultant might find out these issues only after making the recommendations and seeing those being discarded based on them.

As a summary, the following statements should hold in order for doctor–patient model to work properly:

1. Client has to have properly identified which person, group or department is sick and needs therapy

2. Target group has to be motivated and willing to reveal accurate information

3. Client has to accept the diagnosis and recommendations that the consultant arrives at

4. Client has to understand the consequences of doing the diagnosis

5. Client has to be able to make the recommended changes

Schein's last role for a consultant is his process consultation model. In this role the consultant's main responsibility becomes to create a helping relationship to the client. The basic assumption is that the client does not know what his problem actually is or what kind of help he should look for. However, most clients are seen having a positive attitude towards improvements as long as they are helped to identify what to improve and how to improve it. Thus the consultant should help them identify the problem and find solutions for those problems. During the process consultation the problem is owned by the client as the consultant should understand that only the client knows what ultimately works in his organization. By this way the consultant's main goal is to pass on his own skills of diagnosis and intervention in order to help the client improve his organization on his own. With this stance the process consultation differs greatly from the expert and doctor–patient models as it is partly *preventive* not only remedial.

In Agile transformation context process consultation means that the consultant should approach the client with an open mind and focus mainly on building good relationship, not pointing out obvious deficiencies for example in the client's software development process. The consultant should also keep in mind that fundamentally he does not even know if the Agile thinking and methods can help the client at all and thus he should not push them as easy cures for all problems. Instead of providing Agile methods as solutions, he should concentrate on figuring out together with the client the most fundamental reasons why the client is having the problems he is encountering. Only after understanding these reasons the applicability of Agile methods can even be assessed.

The three different models of consultation are summarized in Table 1.

Table 1: Three models of consultation

	Expert Model	Doctor–Patient Model	Process Consultation Model
Initial state	Client has identified the problem, but cannot solve it	Client has not identified the problem, but has a feeling that something is wrong	Client has not identified the problem, but has a feeling that something is wrong
Consultant's role	Provide solutions for the problem	Identify the problem and provide solutions for it	Create a helping relationship to the client and support him in finding and solving the problem
Client's role	Wait for consultant's suggestions	Wait for consultant's suggestions	Participate in identifying the problem and finding solutions
Power distribution	Consultant has lots of power	Consultant has even more power	Power is shared between the consultant and the client
Status balance	Client is dependent on consultant's advice	Client is dependent on consultant's advice	Client is equal to consultant

4.4 Social Constructivism and Language of Change

Traditionally our Western understanding of reality has been based on the scientific world view that was formed during the Renaissance in 16th and 17th centuries. According to this view there is a reality out there that can be observed, theorized and measured objectively. The basic principle is that if one is able to formalize rules that could be reproducibly and empirically proven, these rules describe the way how things are – the reality. (Juuti, Lindström 1995)

The same view of reality adopted by natural sciences was and still is very strong also in organizational and change literature. Extreme examples of organizational forms based on this concept are the ones inspired by Taylorism in the beginning of the last century. These organizations emphasized for example scientific measurement of work processes, tight optimization, clear division of work and strong specialization. (Juuti, Lindström 1995)

However, starting from 1940s more and more critique has been presented against using scientific understanding of reality as a basis for human sciences (Juuti, Lindström 1995, Susman, Evered 1978). Basic theme in this critique is that the

human world is different from the natural, physical world and therefore must be studied differently (Patton 2001, Susman, Evered 1978). From this basis social constructivism emerged to challenge the classical view of single existing objective reality. Social constructivism states that there is no such reality, but a reality is constructed subjectively and socially via activities such as communication and interaction. As reality is constructed by individuals and groups independently, there are not only one, but multiple differing realities that exist simultaneously. This leads to a situation in which one cannot state that some reality would somehow describe the "truth" better than some other, or that some reality would indeed be "right" or "wrong" – they are just different (Juuti, Lindström 1995).

In the world of socially constructed realities language becomes an important concept, because it is the practical tool used to create understanding and realities (Juuti, Rannikko & Saarikoski 2004). Narratives, dialogs, texts, stories and such become the building blocks of realities. For example a wooden structure, that is normally called a chair, becomes a chair only after language is used to label and "create" it. The structure exists before that, but it is not a chair before the association is made (Patton 2001). In this way language is used to specify what being a chair means, and this again creates a new reality in which chairs exist.

As languages themselves are merely social constructions, they reflect the assumptions and worldviews of the individuals and groups who created them. This in turn means that no language can fully capture or represent reality, but details are always lost when the codification of perceived reality with language is done. Because language is the tool to create realities, it also becomes the tool of power. The one who controls the voices and languages of an organization is able to affect how different realities surface and what information they contain. (Patton 2001)

What does all this mean in the context of Agile transformation and external change agent? The first implication is that no external change agent, consultant or coach can ever find out objectively what is really going on in his client organization. All he can do is try to identify various existing realities and by comparing them elicit new information about the situation. The change agent must also understand that his own view of the situation is just another reality among others and by itself it is no more valid or exact than all the others are.

The change agent must also understand that people will have and act on different realities. There might exist for example one “official” reality in a company that is created with strong voices of the upper management. However, at the different levels of the organization other voices get mixed up with this strong narrative. These more silent local voices might not reach throughout the organization, but they still create local realities that affect the organization. In order to hear these more quiet voices and understand the realities they create, the change agent must come close enough to be able to sense them. The change agent should also understand that by observing different realities, comparing them and understanding the reasons behind them, more information about organizations can be gathered. To be able to do this, the change agent must tune himself to identify various realities and regard everything, especially his own views, with certain amount of criticism.

Through social constructivism the change itself can also be seen as a linguistic construction which is created by the voices in the organization and the voice of the change agent together. For example by using language that describes change as remote, top-management driven and slow, the change agent creates very different change than by speaking about our effort, small scale goals and continuous improvement. It is the responsibility of the change agent to try to understand what kind of language best supports the Agile transformation and then actively and consistently use it as a tool.

5 RESEARCH MATERIAL AND METHODS

In this chapter the material used during the study is presented. Also the methods that were used to analyze the material are described.

5.1 Material

The main research material consisted of 21 individual interviews conducted during October, November and December 2007. The interviews formed two groups. The first group consists of interviews performed with five consultants from Reaktor Innovations. These interviews were directed to gather more general data about Agile transformation as Reaktor's consultants see it. As such they did not concentrate on any specific customer or case. The remaining 16 interviews addressed the three customer cases that are presented in Section 5.2.

All interviews were carried out as semi-structured individual interviews. The main themes and some questions were planned beforehand, but most of the exact questions asked in the situation were constructed on the fly. This helped fine-tuning the structure according to interviewee's position and knowledge. The preplanned structures for the two interview groups differed slightly to accommodate the different perspectives of the interviews (general vs. case specific). All interviews were recorded for later analysis. Altogether some 20 hours of interviews was gathered.

5.2 Cases

In this section the three case companies are presented.

5.2.1 BigSoft

By Finnish standards BigSoft is a large software company having hundreds of employees located mainly in Finland, but also in subsidiaries all over the world. The company's business is based on its software products, but it is currently heading more and more towards service oriented approaches. Software in BigSoft was earlier developed using traditional processes and methodologies, but two years ago a big bang transition towards Agile methods was conducted. Some principles and methods of Agile had been already used before that, however, but the transition finally formalized the used methods and spread them throughout the company's research

and development department where all interviews were done. Currently Agile methods are used in all software development projects in research and development organization and experiences have been good. At the moment the main focus of working methods development is set on how to widen Agile principles to other organizational units as well and how to improve co-operation between different units in general.

Four interviews were conducted with the people from BigSoft. The interviewees consisted of a Vice President of Research & Development, a Program Manager, a Systems Architect and a Software Quality Engineer.

5.2.2 Multitalent

Multitalent is an old state-owned consolidated company that is active also outside Finland. The company is divided in three main business units that are assisted by various support organizations. Geographically the company consists of more than 20 subsidiary companies in different countries mainly in Northern Europe.

In Multitalent's business segment software has mainly been in supportive role, but recently it has become an important business asset as new software products and especially software services have been developed. Agile methods in Multitalent were first prototyped some two years ago and have since spread gradually. Most of the projects in the organization are still done using traditional methods, however, but plans and intentions to further widen the use of Agile methods are ongoing.

Nine interviews were done within the organization concentrating on projects using Agile methods. The interviewed people varied from project managers and developers to general management and included employees from all three business units as well as from the supporting ITC organization.

5.2.3 Telco

Telco is a big Finnish telecommunications company having thousands of employees and hundreds of thousands of customers in Finland. Software development is not Telco's core business, but acts as a support function in company's services. Like Multitalent, Telco is also divided into separate business units and ITC organization.

The case used here is about a big, important and software intensive development project that influences Telco's main business units by enabling them to provide new and more advanced administration services for their end users via Internet. The project is the first bigger scale Agile project in Telco. Agile principles were chosen, because the project was seen as very schedule and business critical. Also both business units wanted to use methods that allowed for greater flexibility and shorter time-to-market than the ones they have used earlier.

Three interviews were done with Telco's employees. All individuals represented manager level from both business units as well as the ITC organization working daily with the aforementioned development project.

5.3 Data Analysis Methods

Research was based on the analysis of the information gathered via the interviews in case companies as well as in Reaktor Innovations. At first the interviews were listened through and a mind map of the main themes was constructed for each of them. This was done shortly after each interview. After most of the interviews were done, all individual mind maps were laid out and data analysis phase began.

In the beginning of the analysis, the data in individual mind maps was codified and categorized as suggested by Patton (Patton 2001). At first the main goal was just to get an overview of the data and no reduction or interpretation was done yet. After the data in each mind map was codified and categorized, the interpretation phase began.

During the interpretation the main focus was at first on identifying various challenges that were encountered during Agile transformation. Data that suggested such challenges was copied from the individual mind maps to another specific map that collected the findings together. After all findings were collected to this map, it was reduced and categorized further to remove overlapping.

After identifying challenges the analysis shifted more towards finding solutions as well. As the case companies were mostly still in the middle of their Agile transformation, there was not much direct solutions available in the data. Thus the identification of solutions was mainly based on the synthesis of found challenges and existing theory. Theories were also used in the categorization of the data. They also

guided the analysis focus and determined its abstraction level so that the analysis results could be combined with the theory. Interviews were also revisited constantly during the analysis to sharpen the focus and to find out more subtle meanings in the data. Also more theory was accessed and searched during the analysis.

As the analysis of solutions proceeded, a new mind map representing them was created. This mind map was further reduced and restructured to clarify and separate individual solutions.

Based on the mind maps of challenges and solutions, a text file of topics was generated and this file was used as a basis for the writing work. During the writing process this file evolved constantly as the text searched its form.

6 RESULTS

The results of the study are presented in this chapter. At first the identified challenges from the cases are presented. Later, solutions and suggestions about how these challenges could be tackled are described. The ways how an external change agent could help organizations during their Agile transformation are incorporated to these suggestions. In addition, the last section of this chapter presents the results of a reflection workshop held with Reaktor's consultants.

6.1 Challenges of Agile Transformation

To better help his clients transform into Agile methods the change agent must understand what kind of challenges organizations face during the process. In this section challenges met by the case companies are presented and reasons behind them are analyzed.

6.1.1 Hearts and Minds

When interviewing people from BigSoft it became clear that they shared some kind of common understanding about software development and this understanding seemed to be in line with Agile principles. Employees talked about trying it out, refining the software constantly, emphasizing communication and praising team's ability to handle it by themselves. Some of these concepts were also used at Multitalent, but the used language varied much more and also the language of processes, control, checkpoints and rules was common. This difference and variance in language hints that there might be quite different basic understandings among employees about how software should be developed.

A Reaktor's consultant had also noted the existence of this kind of mental image about software development. He describes:

> *"People live in the illusion that they can manage these uncertainties [of the projects] by using detailed specification work and also manage the risk. There is this kind of contradiction living there and to be able to work using Agile methods it would require pretty radical attitude change here."*

Another consultant described a case in which a kind of a cycle of increased control was forming. At first there was some schedule disappointments in the project and it

was concluded, based on the understanding of software development, that it must be because people were not concentrating on right things and planning was not done properly. So, to improve the situation, some new control mechanisms were introduced. They did not help, however, and yet again more control was added to the process thinking that it will help.

What we see in these cases are the effects of fundamental concepts of how software should be developed and how software development world functions. These concepts can be observed in the used language as well as in the actions of how organizations cope with setbacks. As Agile methods are based on quite different fundamental concepts than the traditional methodologies, some kind of change of these concepts is required in order to successfully implement them. This change of fundamental concepts seems to create challenges to organizations.

The main challenge here is about transforming from rational-linear understanding of software development towards a chaotic-collaborative view. This shift is very much analogous to the shift from the scientific worldview of natural sciences to the socially constructed view of social constructivism. In rational-linear thinking the software world is seen as something that we can and must control through detailed planning, resource allocation and supervision or otherwise the situation gets out of hands. This leads us to traditional waterfall processes and hierarchical and bureaucratic organizations supporting them.

However, when one adopts a chaotic-collaborative view, one admits, that the software world is fundamentally uncontrollable, ambiguous, multidimensional and socially constructed. With these axioms the traditional processes do not make sense anymore and Agile methods look more reasonable answer.

For example in the last case of ever increasing control at no point during the cycle did it come to people's minds that the process should probably be controlled less, not more. It was just something that would be completely irrational considering the used mental model of software development. The only way out of this kind of cycle is to change the underlying assumptions about how software world functions and armed with this new knowledge finally reason that maybe less control and more preparing for change is better than trying to prevent the change.

Organizations might also believe that they have actually successfully applied Agile methods, but when inspected more carefully one might still find out that deeper, often unconscious, assumptions are still based on the traditional thinking. In these kinds of situations the practical work might be based purely on rules of some Agile methods (such as Scrum) and not that much on true motivation of regarding Agile as "the right way" to develop software. Without deeper changes in attitudes organizations might also have challenges in adapting Agile methods to their own needs as the adaptation would then be driven by traditional understanding.

6.1.2 Organizational Context

In many interviews it was brought out that some organizational issues such as bureaucracy or excess hierarchy are hindering possibilities to work in Agile ways. For example in Telco one interviewed manager said:

> *"The whole organizational model doesn't support innovative and effective organizational culture. This organizational model has been driven us.. So that the gang here is a bit tired like 'I don't bother doing this at all. I have to order this with some form from the production and it won't be accepted anyway.'"*

> *"I'd say that the individuals are good and willing to change things, but the working methods that have just been given from top.. the organization just forces into specific working methods."*

Both quotes reflect well the frustration the manager is feeling about the current organizational form which he thinks is not supporting the innovative and effective ways of working he would like to promote.

The same kind of challenge was met at Multitalent where the development teams had set up their own tools and servers to support their work. At first they had tried to get the same services using the "official" way, but after having been turned down and discouraged by the support organization they had thought that there was no point on fighting the windmills and had done the job by themselves.

Problems of high hierarchy and bureaucracy are described also by a Reaktor's consultant:

"I'd say that Agile models shouldn't necessarily be brought to other than professional organizations.. and I think that one shouldn't develop software in other than professional organizations. If the structure is such.. if it becomes such a hierarchical tree so that if you are not after three or four years a manager, you haven't progressed on your career. Not in that kind of organizations.. you can't develop software using Agile methods in such organizations, because you want to have the brightest minds at the development level."

The need for experienced and skilled people when using Agile methods was acknowledged by other interviewees as well. For example in Telco one manager said that he'd rather take "five good guys than 100 mediocre" into his Agile team.

Hand in hand with the organizational hierarchy goes the division of authority and responsibility. Practically in all interviews the importance of sharing responsibility in organization when using Agile methods was noticed. For example in BigSoft the interviewees described that with Agile methods the responsibility has become more attached to teams, not to individuals. Also an interviewee from Multitalent described that with Agile methods the decision making has been transferred to lower levels of the organization and responsibility has been shared to wider group of employees.

A manager from Telco described the division of responsibility and authority in the following way:

"Giving responsibility.. We should, our managers should give, not spurious responsibility, but give responsibility and at the same time authority. Maybe it is one of our challenges that people have responsibility, but not necessarily enough authority. It takes juice out of the people. You are responsible for something, but still you have no possibilities to do the things the way you would like to."

The question of responsibility and authority might also go the other way round as individuals might not necessary be willing to take personal responsibility. A Reaktor's consultant described:

"It's only committees and boards that make decisions. No single person can be attached, in good or bad.. in a way that someone has made a decision. And then to find out who has done some decision, and with whom one needs to talk about something.. It is impossible to track down and when you want that something happens, it is practically impossible."

This kind of committee mentality was also noted in Telco. Exact reasons behind it are not clear, however. One manager said that it is because of Telco's organization culture. Another interviewee said that people might be afraid of taking responsibility or just frustrated that even if they would take some responsibility they would not be handed with enough authority to do the job in a way they would like to.

The division of responsibility is mirrored also on how software developers[2] are seen in the organizations. A consultant working at Multitalent said:

> *"I think the budgets go somehow quarterly.. what you get for some project. Then they might put more people to the team.. or take away. For example we had two projects. The other project got more people and my project lost one.. We shouldn't have lost the guy because of the amount of work to do. It was just so that the other project had more money for the end of the year and we had less. It didn't make any sense, but it still had to be done. Now probably when the next comes, we get more people. … It has nothing to do with how much work you have. It just comes.. like out of the blue.. 'Now there is not enough money.' And you can't do anything. … In terms of team building, it's quite crazy."*

Another consultant said:

> *"They might be doing for example four different products in four projects. After each project ends, they dismantle all teams and when the next release of the same product is done, they take a completely different bunch of guys doing it. It would make much more sense if the same crew continued working with the product. But they have thought so that resources are resources and that everybody gets it done equally fast."*

Yet another Reaktor's consultant told that he has heard the following many times from various managers:

> *"Anybody can code Java. You can get it from India, if you want."*

To combine the presented material, it seems that three different kinds of organizational challenges can be identified. The first one is about excessive bureaucracy and hierarchy that seem to be slowing down the Agile rhythm of making things happen. This is seen as a problem especially among the people who are

[2] Developers here mean all employees actually creating software, such as coders, testers, database specialists and so on.

working in Agile teams or are in some other way very close to them. From their perspective the organization is seen as massive and slow and hindering their possibilities of achieving the goals they have committed to. For example by having to set up their own servers and services for development, the teams in Multitalent have surely got the image that their time was wasted because the host organization was not responsive and agile enough for their requests. Also the manager at Telco must have felt quite frustrated when there were great pressures on achieving his goals while at the same time the organizational rules and lack of authority prevented him from doing the things in a way he would have liked to.

Second identified challenge is about how responsibility and authority should be divided in an organization when Agile methods are applied. It seemed that in many cases the Agile teams did not have enough decision making power to be really empowered in the sense required by Agile principles. Also if the organization has a culture of promoting more experienced people to more managerial jobs, this might lead into situations where there is not enough expertise at the hands-on level of development to assure that Agile methods could be used efficiently. All in all, if the host organization is based on traditional top-driven model of authority and responsibility as well as on command and control management, quite radical changes to these are needed during Agile transformation.

The third identified organizational challenge concerns the way how software developers are often seen from the management perspective. The management seems to often regard developers as mechanistic and uniform work force that can be distributed and reorganized rapidly to accommodate new situations and needs in the organization. From Agile point of view this leads to great challenges as it hinders team building and self-organizing that are at the very core of Agile principles. If people cannot see any sensible reasons behind arbitrary moving employees around, they might also get frustrated, insecure and unmotivated that lead to further challenges in the form of change resistance.

6.1.3 Software Development as IT Projects

In two case organizations, Multitalent and Telco, there was a clear division of responsibilities between different organizational units about who initiates an IT

project and who decides how it is done. People from BigSoft also told that they had a similar kind of separation a couple of years ago if only in lesser degree.

Both in Telco and Multitalent there are business units that use IT projects in order to answer market demand, improve existing products and research for new product possibilities. These units are very goal oriented and concentrate on getting something useful ready for the customers. In both companies the business units do not run the IT projects independently, however, but there are separate IT support organizations that participate. The role of these organizations is for example to assure that reasonable technical decisions are made, the technologies and software products used in the projects are in-line with each other and that the technical quality of the projects exceeds the set criteria. In Telco the division is done for example in a way that the business units can decide what is implemented and when, but the IT organization decides how the software will be done and who ultimately is going to do it (the actual implementation work is outsourced to subcontractors that are selected by the IT organization).

Partitioning responsibility in this way seems logical from the standpoint of traditional software development methodology. However, when an organization starts to adopt Agile methods, this separation of concerns becomes a challenge. For example in Telco from the perspective of business units the IT organization was seen as restricting, slow and inhibiting their efforts of creating more end-user value for their customers. The manager from a business unit in Telco said:

> *"Currently we are designing our services in detail by ourselves in business unit, because we don't want guys from IT organization tampering with it. It is just completely unnecessary overhead. We just don't see any benefit in having another party there that on paper needs to be participating.. and then we need to explain them from the beginning that we want to do this in this way, because we have this another thing coming and so on.. … We would like to work very close to the guys implementing the software so that we could quickly guide them like 'not that way, but this way."*

On the other hand the situation is seen differently when looked from the IT organizations. For example some interviewees from the IT organization of Multitalent saw business units as unable to decide what they want, slow on their

decisions, constantly changing their minds and having unreasonable expectations about how fast software can be implemented.

One reason that might lead to these kinds of different views is how different departments are measured and awarded in an organization. Business units for example in Telco are measured by how much and how effectively they can produce products and services to company's customers. On the other hand the IT organization is measured by costs and schedules. This for example drives them to pursue for fixed prize and fixed scope projects. However, these kinds of projects are not necessarily needed or wanted by the business units. A manager from Telco describes:

> *"Business units are steered by what your profit is, how much good you get from clients to us, how much you sell, how much you introduce new products. IT organization is on the other hand guided by what is the smallest amount of money with which this can be realized. And in a way these ways don't meet always. IT is directed like 'decrease costs, decrease costs, decrease costs' and business is directed like 'invest, invest, invest'."*

The situation between business units and IT organization can also be seen at more general level. From this point of view the challenge is about the chasms between different parts of the organization and these seem to just become deeper if some parts of the organization start adopting Agile, but others do not. For example in BigSoft the research and development department has now fully adopted Agile principles in its operation, but most of the other parts of the organization have not. This has created various challenges. Vice president of research and development describes:

> *"For example marketing told us that at first we were late with the releases, but now we are too early and shouldn't release so often, because they can't react quickly enough. They expect that we plan for a year or so."*

This kind of lack of synchronization and co-operation through inter-department boundaries might create challenges both for those who are implementing Agile methods as well as for those who do not intend to do it or are slower in doing it.

In addition to crossing organizational boundaries, the different mental models about IT projects in them often create challenges during Agile transformation. A manager from Telco describes:

"In Telco we have this traditional understanding that a basis for a project is that we know what we want, we recognize risks, we have specifications and we have schedule and cost estimates. … You need to have these conditions fixed before the work can start.. before the permission comes.. funding and such."

Clearly this kind of traditional conceptual model might create challenges during Agile transformation, because Agile projects will not fit to the same mold than the traditional ones. For example an Agile project cannot precisely answer to questions such as how much a project will cost or when it will be ready. Also without exact budgets and schedules the IT organization might not be able to put the project out to tender for subcontractors and thus cannot select them using their normal processes.

The described fundamental understanding of what is actually a project is reflected also on top management's assumptions about what kind of information should be available from the project organization. In Telco for example managers assumed that they should be able to get the exact project measurements (cost, schedule, scope) at any time they wanted and this created lots of extra pressure for the project people, because with Agile methods this kind of exact data far into the future is not readily available. Exactly the same challenge was also noticed in BigSoft when they started using Agile methods. According to interviews, they needed to change their concept of project estimation and measurement at the company level to mitigate the change.

To conclude, there are many aspects about IT projects that might set challenges during Agile transformation. The ones met with the case companies include the partition of responsibilities between various departments during the project's lifetime and the fundamental concept of what an IT project is and how it is measured.

6.1.4 Challenges at Individual Level

Even though previously mentioned challenges are on quite high level and abstract, Agile transformation influences employees also at individual, more concrete, level. The most significant challenge found in the case companies at individual level is based on how Agile transformation changes the roles of employees. According to interviews old strictly defined roles and titles are fading away and new effort and ability based roles emerge. Most notable changes are taking place at the lower levels of the organization. The roles of architects, developers and testers get mixed and responsibilities blend together in Agile teams leaving each individual looking for his

new role independently and without formal support from the organization. An architect from BigSoft describes:

> *"Especially in teams where there was a strong project manager, who has told that 'now we do this in this way'.. people should now start to take responsibility on their own. If you have never in your life taken any responsibility, it is hard to start it just by starting to use Scrum."*

The same situation applies also at the project manager level. For example in BigSoft people thought that there is no place for controlling and governing project manager when Agile methods are used. Instead, facilitating and removing obstacles that slow the team down should become project manager's main responsibilities. BigSoft's architect has a story about a project manager:

> *"We got a project manager who was very traditional. He was like 'let's make a plan, let's create metrics for the plan and then with the metrics I can monitor if the plan is progressing.' He left after about year and a half. He left, because he thought that there was no role for him anymore. … And it is so! In Scrum and Agile there is no such role. There are roles for coordinators and facilitators, but he was neither of those. He was so traditional project manager."*

The situations where roles and responsibilities get mixed are favorable for learning anxiety to rise as there is lots of uncertainty in air: What should I be doing? Am I skilled enough to do this? Is this my responsibility? To whom should I report to? What other team members think about me and my skills? The increased learning anxiety then creates resistance of change, as Schein points out.

Role changes are not the only factor that might create change resistance. An interviewee from BigSoft told that with traditional methods it was common in a team to have separate database specialists, web interface specialists and integration specialists who were very good at their specific area, but could not do much outside of it. However, after the transition to Agile methods, people at BigSoft have tried to get rid of such specific skill sets and currently an ideal Agile team is seen as a one having people with abilities to do various tasks depending on team's needs. According to interviews also such skills as communication and teamwork have now become much more appreciated than before.

A manager from Telco describes the problems of having too many specialists in this way:

> *"This guy can do PowerPoint specifications, this guy can write HTML, then we need a graphics guy, someone who knows Tapestry, then we need some integration guy, a database guy etc. Suddenly we have ten persons doing the same feature and they are not necessarily able to do the next feature."*

Based on this data it seems that Agile methods require more skilled people than the traditional ones and also the relative importance of required skills changes. This naturally becomes a challenge if there are not enough of such employees available. Also higher and changing competence requirements create more learning anxiety and due to this more resistance against the change might appear.

In addition to shifting roles and skill requirements, many interviewees stated that Agile methods have created more visibility to their projects. It was stated that for example when working together in a team room, the competences of each team member have become more visible for everybody. A consultant at Multitalent described:

> *"Maybe it's a challenge of a big organization, but not necessarily everybody is up to their job. … The problem is that when the team is formed and tasks are given.. it is revealed that there are people that actually can't do the job they were supposed to be doing. It's not actually problem in Agile.. It just exposes it."*

This kind of increased visibility to one's work is not necessarily always considered as a positive thing. If for example a person is concerned about his abilities, he might not want others to find out at which level of proficiency he is. This again might create learning anxiety and resistance against such practices that promote added visibility such as daily Scrum meetings.

According to interviews, the same pattern of increased visibility repeats itself in many other situations as well. Especially managers noted that all challenges and problems have come out more quickly with Agile methods. This was seen on one hand as a benefit as the challenges could now be tackled, but on the other hand it was also seen more painful for the organization and slowing the change down.

The mechanisms behind increased visibility can be explained using social constructivism and how Agile challenges participant's thinking. When Agile is introduced to an organization, it creates new language and discourse that can be used to codify knowledge about the prevailing situation. Through this codification more information can be translated from implicit to explicit. New explicit information in turn may challenge existing knowledge and perceptions and might thus create anxiety. One example of this kind of language introduced by Agile is the term for team's speed, "velocity". By defining velocity team's effectiveness can be codified into explicit knowledge that can be traced in time. New realities are created and these might not necessarily fit well to old ones and thus anxiety might rise. If for example some team has perceived itself as a steady performer, but the new knowledge challenges this view, they might need to reposition their conceptions. In the same way all other claims about Agile creating more visibility can be explained. The basic premise is that Agile brings in methods and language that can be used to more effectively codify already existing situation to create more explicit information.

When drawn together, the most important challenges at the individual level of the organization are changes in employee's roles and responsibilities, changes in competence requirements and increased visibility through new language in various contexts.

6.1.5 Summary

As four previous subsections present, various challenges emerge when an organization goes through Agile transformation. The identified challenges settled on different areas of organizational context and their appearance varied greatly between case companies. However, still some categorization can be applied to them and it seems that three different categories emerge from the research material.

The first category is about fundaments of software development including concepts about what software is, how software development can be managed, how software projects should be seen, etc. The next two categories follow closely Weick & Quinn's presentation of change in its macro and micro level phenomena. At macro level challenges form around concepts of how authority should be divided, how organizations should be laid out and how bureaucracy and hierarchy could be handled. On the other hand, at micro level, issues such as what are employee roles

and responsibilities, what kind of competencies they should have and how they are treated from organization's viewpoint form the core of encountered challenges.

The summary of challenges and their categorization is presented in Table 2.

Table 2: Identified challenges of Agile transformation

Challenge Category	Identified Challenge
Fundaments of software development	Traditional fundamental understanding about software development
	Traditional concept of an IT project
Macro level issues	Excessive bureaucracy and hierarchy
	Top-driven division of authority and responsibility, command & control management
	Distribution of responsibilities between departments, inter-department co-operation
Micro level issues	Changes in employee competence requirements
	Changes in employee roles and responsibilities
	View of software developers as resources
	Anxiety by increased visibility challenging current perceptions

6.2 Solutions

This section answers to the last two research questions about how Agile transformation should be done in order to tackle the challenges identified earlier and how an external change agent could help during the transformation process. At first, a consultation process that can be used to help organizations during Agile transformation is presented. After that, three more specific aspects of Agile transformation are inspected and their connections to theory are presented. The three aspects describe the change from different perspectives. They are consistent with the division of challenges presented in Subsection 6.1.5.

6.2.1 Agile Transformation Process

This subsection describes a consultation process that can be utilized to help organizations during Agile transformation. The process is based on the synthesis of challenges identified with the case companies and theoretical frameworks used to analyze them. The suggested process is presented in Figure 8.

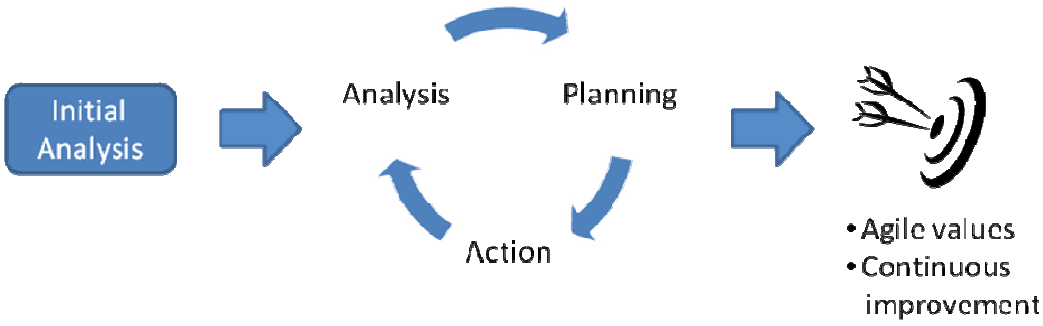

Figure 8: Suggested consultation process for Agile transformation consultants

As can be seen from the figure, the process consists of two phases that together help organizations achieve the goals of Agile transformation. The following paragraphs describe more closely the different parts of the process.

According to Schein's process consultation, the ultimate mission of any consultant should be to help his clients help themselves (Schein 1999). In Agile transformation context this means that the main goal of the consultation should be to guide the client organizations to a state in which they would no more need help from external consultants, but instead could independently monitor, improve and adapt their operations and organization to meet the requirements of the changing environment while still maintaining Agile ways of working.

To achieve this, the consultation process used by the change agents should aim for helping clients absorb two things (presented at the right in the process figure): 1) solid enough understanding about Agile values and 2) ability and culture of self-reflection and continuous improvement. Adaptation of both of these is required for the client organization to continue on Agile path on its own. If Agile values are understood and accepted, but the ability to reflect one's own actions and improve on them is missing, the Agile practices will gradually decay as they will not keep up with the changes of their surroundings. On the other hand even if there would be a strong culture of continuous improvement and self-reflection, but the proper

understanding of fundamental Agile values is lacking, the organization would be developing itself blindly and would soon diverge from practices based on Agile.

Even if the main goals of the consultation process are quite clear, the change agent should not rush into practical doing without first familiarizing himself to the circumstances at more general level. Practically this means that the change agent should, along the principles of process consultation, at first try to understand what is going on in the client system, why consultants are needed and what the client actually wants to achieve. It might for example well be that the client himself does not know very well what his goals are or how to achieve them, and thus hurrying into action might just create more harm than good. In some cases it might not even be feasible to start Agile transformation and the sooner the change agent is able to tell this to his client the more resources are saved and left available for other uses. This kind of initial analysis work is presented as the fist step of the process in Figure 8.

During the initial analysis the change agent must remember that the client is ultimately the owner of all issues at hand and thus he should also be the main responsible for them. The agent should thus strive to create a relationship with the client in which the client sees himself as the main driver of the change and the agent is only supporting his work. As Schein points out, through this kind of participation the developed results are much more likely to be truly helpful and also fitting to the client's circumstances.

After the initial analysis of the situation has been done and mutual understanding between the change agent and his client has been created, more concrete change work can begin. Even though during the early phases of the consultation process the change agent relied mostly on the process consultation principles, he should now start widening his theoretical perspective.

As Lewin presents, even deep and fundamental changes are possible through the use of Action Research and 3-Step Model of change and these should be the main theoretical frameworks around which the actual change work is tied during the cyclic part of the consultation process. In practice, the theories can be applied simultaneously at different levels. The sequential framework of 3-Step Model should be used to outline the situation at the higher level while change and learning at more

concrete level should be performed iteratively in small steps as suggested by Action Research.

Regardless of the conceptual level at which the change agent is working, he should support Lewin's theories by using social constructivism to help him understand the variety and natures of organizational realities encountered during Agile transformation. Practically this means that the use of change agent's own as well as interpretation of other's language should be his main tools used during the consultation process.

Language can not only be used to understand existing realities, but to actually create new ones (Juuti, Lindström 1995). During Action Research cycles the change agent should thus actively use it to create and strengthen realities that support Agile transformation. For example by spreading success stories about previous Agile projects the change agent can strengthen the reality in which the Agile methods are regarded as good and successful ways of working. In similar ways providing new vocabularies, for example for task estimation, the existing reality can be extended and new learning opportunities emerged.

To be able to use language in such effective ways, the consultation process should last long enough. Also the natures of Action Research as well as process consultation indicate that the change agent must invest enough time to be with the client to witness many enough feedback loops on one hand and to be able to create a trusted relationship on the other. From the viewpoints of all of these theories it seems doubtful that short interventions would provide very much help for the client as in these cases the change agent does not have enough time to understand the situation and be in interaction with the organization.

Needed long-term commitment combined with the fact that the client should own the change process leads to a conclusion that some kind of a dedicated change group inside the client organization should be used to effectively manage the change. By having this kind of group working inside the organization the ownership of the change is more apparent and the constant caretaking of the change process can also be more easily facilitated. It is also easier for other employees to identify the change to a specific change management group that has been particularly appointed for the

task. A specific change group can also more easily create better visibility to change and act as a central hub of information.

BigSoft has good experiences of using such internal change groups. During their transformation two groups that were primarily responsible for driving the change were established. High level decisions and guidance was in the hands of a steering group, which had meetings less frequently than the operational group, which was responsible for putting things in action at the concrete level. This operational group had daily face-to-face meetings during which the status was checked, decisions made and new action points given. With this kind of active and short cycled participation all encountered problems could be tackled quickly and still viewpoints of different stakeholders could be taken into consideration. Daily meetings were also effective in reducing bureaucracy around the change process. In this way a clear symbolic sign was sent that things can also be done without it.

From the theoretical point of view these kinds of change management groups are responsible for driving Lewin's 3-Step Model at the organizational level. They should have enough authority to create credible disconfirming information and at the same time support change by creating psychological safety by providing relevant information, listening people's concerns and acting on to remove those concerns. With its two separate change management groups, BigSoft was able to achieve exactly this: The steering group, which consisted of senior management, brought in the needed top management support and authority while the operational group was closer to employees and was thus able to listen to their worries and create needed psychological support.

As social constructivism suggest, involving people from different parts of the organization is important for effective change and this should also be visible in the composition of people in the change management groups as well. By having a good mixture of people in the group, the group is better able to understand different organizational realities and can thus function more effectively. This was also taken into consideration in BigSoft as the operational group consisted of people from different areas of the organization.

In conclusion from the process perspective, Agile transformation should be started with an investigative process consultation that aims to shed light on the client's current situation and goals. Practical change work should be done iteratively and learning as well as understanding through trying should be encouraged. The client must be engaged and own the change, and some kind of internal change management organization should be used keep the wheels rolling. Throughout the whole process the change agent should apply social constructivism to better understand the situation and use language as his main tool. He should also never forget his main goal: To help his clients help themselves.

6.2.2 Changing Fundaments of Software Development

As presented earlier, already existing fundamental concepts of software development and software projects often create challenges during Agile transformation as they might not support well Agile values and methods. To be able to change the fundamental concepts, the change agents can use Lewin's theories. Especially the 3-Step Model of change provides a good framework to understand how change can be accomplished.

The first step in Lewin's change model is to unfreeze the current situation. At the level of fundaments this can be achieved by first bringing them visible and then challenging them to create needed change motivation. Bringing fundamental concepts visible is not easy, however, because they are often so firmly established that people do not even realize their existence or how they affect their actions.

To help people understand how they see software development and how their understanding affects their behavior and decisions, the change agent can use the principles of Action Research. By engaging the client organization to continuous self-reflection and feedback, all participants can gather more understanding about the forces that affect the organization. Then, by analyzing these forces and their origins, more fundamental concepts can be found and brought to daylight. Through Action Research the change agents could for example bring forward different issues associated with project planning, requirements elicitation and change management. By investigating why these activities are done the way they are, common understanding about the reasons behind can be developed. This understanding can then be used to identify and bring out the basic concepts.

When eliciting fundamental concepts it is extremely important that the consultant is very well aware in which role he is at any given time. As the clients might not yet explicitly recognize their own assumptions, there are great risks that the consultant reflects his own expectations or beliefs to the client organization and thus biases clients' reality as well as his own. It is for example very easy to create mental generalizations for organizations such as "all governmental organizations are very bureaucratic and slow" or that "these guys here do not know anything about software development". If further actions are executed based on realities distorted by such assumptions, they might lead to unexpected and unwanted results. To avoid such misinterpretations the process consultation rule of "Access Your Ignorance" (Schein 1999) is truly a helpful one.

After some visibility to fundamental concepts has been achieved, the change agent should concentrate on creating suitable change motivation based on them. This can be achieved for example by bringing up challenges or problems they are creating. It is important, however, that the change agent mainly facilitates the discovery of such issues as the clients seem to regard findings more reliable and significant if they have identified those by themselves or at least in close co-operation with the change agent (Schein 1999). Again the change agent should also be careful not to project his own attitudes to client organization during the process.

A good example of how better visibility can create change motivation comes from BigSoft. There people started to use Scrum product backlogs to better understand and organize their work and through the communication needed to create those, many previously unidentified problems came up. These new problems then created motivation for improvement and provided a common shared reality about what can be achieved by changing existing working methods.

After unfreezing through elicitation of fundamental concepts and creation of change motivation, the change agent should help the actual change take place. As Lewin points out, it is very difficult to predict to which direction learning occurs once the change motivation has been created. However, Schein still gives the change agents tools to manage learning by presenting that all learning happens either via identification to some promising role model or, if lacking them, via trial and error method called scanning. In the context of Agile transformation both methods have

their strengths and weaknesses from the point of view of the change agent. As Schein states scanning produces more probably solutions that fit to learner's situation, but on the other hand the direction of learning is highly unpredictable as individuals gather information on their own. Identification again might lead to situations where used role model does not work in learner's context, but at the same time learning is much more controlled as the change agent has at least some kinds of possibilities to affect the selection of role models the learners are exposed to.

As it is important that the organization under change is moving towards common goals, it seems, that role models should be the main method for the change agents in controlling learning during Agile transformation. However, to reduce the probabilities of unfitting role models, the feedback loop of Action Research should be utilized to bring in also forms of trial and error to the learning process.

Practically role models can be created using language to strengthen realities where some specific ways of thinking or acting can be connected to existing needs of learning. If for example it seems that clients could benefit from enhanced communication between their Agile teams, the change agent could provide them with a war story about another organization that successfully used such methods as Scrum of Scrums or inter-team co-location. With such use of language the change agent at the same time provides more information to the clients as well as gives them an opportunity to identify themselves to other teams. The agent also projects the learning away from himself thus putting him to a more facilitative role than just being a teacher.

One role model, that was seen helpful in case organizations when trying to understand Agile principles, was the use of Lean Thinking (Womack, Jones 2003). This usefulness is probably because Lean uses the language of production to explain its concepts. For many, this language might be more familiar and concrete than the language of software development and thus easier to get grasp on. This again helps people more easily create a mental model of Agile as "Lean in IT" and thus supports understanding the basic principles behind it.

Another method that was utilized in case organizations was the use of Agile practices as an intermediating tool to learn also the principles behind the practices. The

language brought in by new practices extends the vocabulary of participants and thus enables creation of new realities through cognitive broadening. This then helps the change agent to better explain more abstract principles as he can now work in an already wider and more suitable reality.

New practices quite often also produce quickly some visible results and benefits that can be effectively used to create more motivation and buy-in for the change. However, the change agent should not get lured by quick gains of introducing new tools or methods, but instead remember that they should primarily be only means to help his clients better understand the more fundamental concepts of Agile. This is important, because only through understanding these issues, the clients can achieve the state where they can continuously improve their practices and keep learning – thus rendering the change agent's work completed.

Lewin's change model ends to refreezing, but the last phase needs to be seen from Agile perspective to fully understand what it means during Agile transformation. Lewin's basic idea about refreezing is to strengthen the new learned things and prevent slipping back to old ways. He suggests that this can be accomplished by for example creating structures that support new ways or prevent the usage of old ways. A company could for example change its software tools in a way that only new processes are supported. In the context of fundamental concepts, the main goal for refreezing becomes attaching new fundaments in such a stable way that organization does not loose them even after the external change agent leaves.

The change agents must be careful with refreezing, however, because generally in Agile transformation there is no specific end state to which the *working methods* are to be attached, but instead *the wanted end state is a state of continuous change and improvement of used methods*. Thus in Agile transformation in general the refreezing does not mean that change should end – it means quite the opposite, actually. If refreezing is achieved properly, the client organization freezes to a state where the fundamental concepts will not change, but at the methods level it keeps on reflecting, changing and improving all the time.

To conclude, changing the fundamental concepts of software development is at the same time necessary and difficult. The change agent must use delicate and careful

methods to first help his clients bring up the current fundaments. After that, the process of simultaneously challenging them and providing directions for learning using roles models should take place. Action Research provides methods to do this, but during the process, the change agent must be very cautious not to let his own beliefs or assumptions distort the existing realities. Finally the change agent must understand that refreezing in Agile transformation does not mean the end of improving working methods, but the freezing should happen mainly at the level of adopted values. These themes are summarized in Table 3.

Table 3: Goals and methods of changing the fundamental concepts of software development in different phases of the process

Phase	Goals	Methods
Unfreezing	Bring hidden fundaments visible	Action Research: Self reflection, feedback, analysis of the current situation
	Create needed motivation for change	Challenge current situation, bring up challenges caused by used fundaments
Change	Unlearn current fundaments	Use language to provide new role models for learning (e.g. war stories, Lean, Agile practices), use Action Research to enable trial and error learning
	Adopt new fundaments	
Refreezing	Stabilize new fundaments	Attach new fundaments explicitly to working methods to prevent slipping back, encourage reflection and constant improvisation based on the new fundaments
	Let working methods continue improving	

6.2.3 Change at Macro Level

Challenges of Agile transformation at macro level tie themselves around high level organizational issues. Issues such as bureaucracy, hierarchy and division of authority and responsibility are the main themes identified in case organizations. In organizations these are commonly regarded so abstract and slow to change, that people get easily frustrated even when thinking about trying to change them. Through better understanding of current situation, strong motivation to change and organization wide co-operation things can be changed, however.

From the perspective of social constructivism the main issue with these kinds of challenges is the lack of common reality in the organization and the lack of organizational communication that could be used to create it. For example the excessive use of email instead of face-to-face communication in modern organizations certainly affects the ways people are able to create common understanding about the matters at hand. With differing realities each party sees the change and the world around it from their own perspective and this tends to bias their actions towards what they see as the best option to go for. This might naturally lead to local optimizations and contradictions between different parts of the organization that in turn might slow down the change and decrease the value gained from it.

However, it is worth noting that these kinds of contradictions do not (necessarily) mean that different parties are trying to intentionally drive their own benefits or improve their situation in the organization. Their motives might, and very commonly are, genuine and they pursue for improving the organization as a whole. The problem just is that their understanding about what would be optimal for the organization might differ quite radically from other's views, because they do not share the same basic assumptions and realities on which the vision of future state could be build.

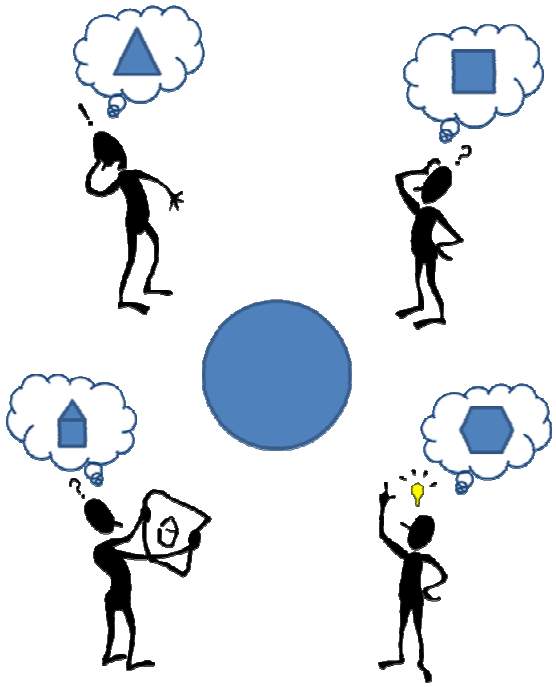

Figure 9: Differing realities in organizations may lead to local optimizations

To help building common shared understanding and better co-operation in organizations, the role of an external change agent at macro level should be to get people from different parts of the organization together, provide them with a common language and facilitate the process of creating mutual reality while at the same time bringing in subject knowledge about Agile principles and methods. While doing this, the consultant must be careful not to project his own interpretations of the situation to the client organization as it could bias the emerging common reality. The change agent must also understand that "different parts of organization" here does not only mean different organizational units, but any groups or parties having significantly different views of reality. For example people working in different roles in the same unit might have completely different views about the change and its goals, and the change agent should try to understand these and try to bring them together. As the Program Manager from BigSoft put it:

> *"You have to forget all organizational boundaries. Both horizontal and vertical."*

Involving people from different areas of organization also helps to establish Agile not only as a method of developing software, but as a more general way of thinking about how software development and functions linked to it should be seen and managed as a whole. This helps to lower organizational boundaries and avoid creation of confrontation between the parts of organization using Agile methods and the parts not using them. Here the already mentioned Lean thinking can be utilized as well. In the spirit of process consultation the change agent should still always remember that he is not there primarily to tell how the client should do their business or organize their software development, but rather help them learn it by themselves.

As with all human interaction, also in organizations there will always be some form of politics and power games happening all the time and these need to be taken into consideration during Agile transformation. Even if people would not actively and knowingly take part to such actions they might well do it unknowingly. For example there might come situations where Agile transformation leads to such organizational changes that employees might start to be worried about possible layoffs and in such situations it is just natural that they try to protect their own positions and

organizational units in order to keep their jobs. The change agent helping with Agile transformation should understand such situations and be prepared for them.

One way to prepare for organizational politics is to ensure support from high enough management that would help keep the change on track regardless of what is happening at the lower levels. The top management support is vital for Agile transformation also for other reasons than controlling politics. For example redistribution of authority and responsibility to lower levels of the organization might be impossible to obtain without support and mandate from the top management. Top management can also create much needed visibility for change and provide common language and realities for example through outspoken visions of the future. From the point of view of Lewin's theories the top management is commonly also the source of needed disconfirming information, which can be used to build motivation for change. All in all, it seems that Agile transformation without proper top management support will be extremely hard to execute.

In conclusion, the challenges at the macro level seem to base on the lack of shared reality between different parties in the organizations. Thus, to help his client, the change agent should facilitate the social process of constructing common reality by bringing people together, providing them with a common language and bringing in the needed subject knowledge about Agile. In addition to facilitation, the change agent must understand the consequences of organizational politics and assure the essential support from the top management.

6.2.4 Change at Micro Level

At micro level the Agile transformation is about how individuals and teams perceive the change and how the rest of the organization and external change agents could help them during the process. Again Lewin's change theories provide the foundation on which the action should be based here.

As Schein said, the true artistry of change is in the ways how change agents manage disconfirming information, learning anxiety and psychological safety. It seems that with Agile transformation the main reasons behind increased anxiety at individual level lie around the issues of changing roles and responsibilities. Also changing requirements about authority and competence produce their own shares of anxiety.

Consequently, these should be in the main focus of the change agent. As these changes take place, the change agent should be there to help involved understand the reasons behind the changes and support them in their search for new reality in which they could again feel comfortable in.

In practice, the required support and psychological safety can be achieved in many various ways and these are highly dependant on the situation and people involved. In any case the language should be the change agent's main tool also at the micro level. Deliberate use of language and constant deciphering of various realities behind it should always be used to provide better understanding of the situation at hand. A good example about how language affects the change at micro level comes from BigSoft.

During Agile transformation BigSoft had challenges with the concept of a task and when it should be regarded as done. On one hand the developers considered the task ready when it was implemented and integrated while on the other hand testers did not regard it done until they had approved that it worked as specified. Yet still the product managers thought that it was not done until proper documentation was also available, translations to various languages were done and so on. These kinds of different realities about the definition of done produced various challenges to for example task estimation and project tracking. These could not be solved until a common concept of what done actually meant was established. By paying attention to the used language, the change agent is able to help observe these kinds of situations and can help overcome them.

When implementing Action Research at the team level, the change agent must not concentrate too much on trying to find out the most efficient working methods for the team. Instead of purely assessing how different methods work and how they could be improved, the change agent should encourage the team to figure out *why the tried methods functioned as they did* and *why they caused the reactions they did*, because through understanding the reasons behind, more knowledge about the situation can be acquired.

At micro level the challenges of politics and power games must also be taken into consideration. They should be seen in smaller perspective, however, and focus

should be set on understanding why single individuals act as they do. In this context it is very important that the change agent remembers Schein's statement that everything a consultant does is an intervention that has some, possibly unexpected, consequences. With Agile transformation this is especially relevant as Agile methods tend to bring up hidden challenges and problems in the organization. If the change agent is not careful with his actions, he might introduce methods that bring out issues the individuals might not be ready to confront. If for example prevailing organization culture tends to promote search and punishment of guilty behind the problems, the acceptance of Agile methods among those with authority and responsibility might not be too wide as they are afraid of such consequences.

To avoid these kinds of conflicts the change agent should constantly be aware about possible outcomes of his actions. He should for example find out various power relationships in the organization in order to be able to avoid humiliating superiors in front of their subordinates. Also finding out who has been involved in or responsible for various projects, programs and agendas is vital information for the change agent. The change agent should also actively try to change the climate of the organization towards the state where finding a problem is regarded as a good thing and seen more as an improvement possibility than an opportunity for blame and punishment. This kind of attitude is required also to get good results from Action Research as honest and open feedback is vital in that context.

At individual level the consequences of Lewin's Group Dynamics should also be taken into consideration. Group Dynamics state that one should not try to change the individuals of the group separately, but the group as a whole. This is because if one changes only parts of the group, the unchanged group structures and culture might negate the changes once the individuals return to the group. From this standpoint it would for example be more beneficial to send a whole team to a same training even though the training might not fit perfectly for every individual of the group. With Group Dynamics also the bias for continuous learning through doing together is emphasized and common experimenting supported by an external coach should be better way of changing the group norms than letting each member learn new ways individually. Changing by doing also helps the organization keep its focus on productive work and ensures that new adapted methods will be in line with it.

To conclude, the main responsibility of the change agent at the micro level is to understand why teams and individuals perceive the change as they do. This understanding can be developed using Action Research and constant questioning of agent's own knowledge about the situation. The agent must also be aware of the consequences of his actions and try to gather enough information to be able to see them better. Again the language should be the agent's most important way to understand and influence the situation.

6.2.5 Summary

As shown by four previous subsections, the selected theoretical frameworks provide varying suggestions about how Agile principles and methods should be implemented. In this subsection all these are drawn together to show what different theories give to change agents and how they contribute to management of Agile transformation process. Also ten practical principles for change agents are presented to further summarize the suggestions from the point of view of the change agent. By following these principles the change agents can utilize the potential of the presented theories in their work.

According to previous subsections, the main goals of the organization during Agile transformation should be to learn Agile principles and adapt a culture of continuous improvement to use them. This learning should be supported by understanding and changing the fundamental concepts of software development. Organization-wide communication is of essence in the process and it should be utilized to create the required shared realities about the current situation. Also organizational boundaries must be boldly crossed and different parties in the organization participated to the change. Authority and responsibility needs to be re-divided in order to empower lower levels of the organization. Change process should be managed by an internal change management group that is fully engaged in promoting change and that owns it. If this group does not already have top management in it, the support from them needs to be assured in some other ways. The effects of Agile transformation must be understood also at the individual level and support required by for example changing employee roles or competence requirements needs to be arranged.

An external change agent can help during Agile transformation by facilitating organizational discourse and by providing needed domain knowledge about Agile

principles and methods. Change agent's main goal should always be to help his client help himself. He can achieve this by creating a good helping relationship to his client and by understanding that his primary role in the process should not be the one of an expert or a doctor, but the one of a process consultant. Consultant's main tool during Agile transformation should be the language he uses and listens to. It should be used to understand and influence the situations at hand. The summary of found solutions and their connections to identified challenges is presented in Table 4.

Table 4: Summary of identified challenges and solutions for them

Challenge Category	Identified Challenge	Solutions
Fundaments of software development	Traditional fundamental understanding about software development	Bring fundaments visible, create change motivation by challenging them, let client learn new fundaments using Action Research, stabilize new fundaments.
	Traditional concept of an IT project	
Macro level issues	Excessive bureaucracy and hierarchy	Create common understanding throughout the organization by participating different groups and facilitating the creation of shared realities between them. Support this by providing common language and bringing in the needed subject knowledge about Agile values.
	Top-driven division of authority and responsibility, command & control management	
	Distribution of responsibilities between departments, inter-department co-operation	
Micro level issues	Changes in employee competence requirements	Using Action Research and Process Consultation try to understand the different realities of individuals and groups as well as the reasons behind their actions. With this understanding help individuals and groups change by creating needed psychological safety and providing required domain knowledge of Agile.
	Changes in employee roles and responsibilities	
	View of software developers as resources	
	Anxiety by increased visibility challenging current perceptions	

When looking Agile transformation from the perspective of an external consultant the solutions presented during previous subsections can be encapsulated into ten practical principles or advices. These ten principles are presented in Table 5. Some of the principles are derived from the principles of process consultation presented by

Schein (Schein 1999). If this is the case, a reference to Schein is present in the description of the principle.

Table 5: Principles of change agent during Agile transformation

Principle	Explanation
Remember your primary mission	A consultant should always keep in mind that he is there to help his client help himself. Only through understanding the situation by himself, can the client continue on his own after the consultant leaves.
Know yourself	"The most important thing to understand in any relationship is *what is going on inside the head, especially one's own head*" (Schein 1999). The change agent must distinguish between what he knows, what he assumes he knows and what he truly does not know. He must also be aware in which role he is at any certain time. (Schein 1999)
Focus on Agile principles and continuous improvement	To be able to continue on Agile path by himself, the client must learn Agile principles and how they can be applied to continuously improve the organization. The change agent must help his client achieve this. Agile practices can be used as an intermediating tool, but implementing them should not be the essence of Agile transformation.
Spend time with your client	The change agent can truly understand the situation only by being part of it and having enough time to experience and interact with it.
Involve people to the change	The change agent must facilitate the creation of common realities in the organization by bringing people together. In this way a shared direction for change can be established and local optimizations as well as boundaries between Agile adopters and other parts of the organization can be avoided.
Let client own the change	As only the client must live with the consequences of Agile transformation, he must own the change and be responsible for it. Also only the client himself knows what ultimately works in his situation and the change agent must not lead him to believe that the agent knows it as well. (Schein 1999)
See people as individuals	Even big organizations are comprised of individuals having unique competences, wills and concerns that need to be taken into account during Agile transformation. The change agent must appreciate this and understand its consequences.
Use language to understand and create realities	The change agent must understand how language reflects underlying impressions of reality and use this knowledge to tune into those realities. He must also recognize how language can be used to create and alter existing realities and take benefit of this potential.

Principle	Explanation
Engage top management	The change agent must ensure the top management support for Agile transformation as it is needed to succeed. If support is not available, the change agent must inform his client promptly about greatly increased risks.
Understand that everything you do is an intervention	The change agent must understand that everything he does affects the client system and thus he must own and be responsible for all of his actions. He must also assess the consequences of his actions to be sure that they are in line with his goals of helping his client. (Schein 1999)

6.3 Reflecting the Study with Reaktor's Consultants

In order to get a better understanding about the validity of the study's results, a workshop with Reaktor's Agile consultants was arranged in March 2008. The workshop had two goals: 1) present the study to a group of Reaktor's consultants to inform them about the used theories and results and 2) collect feedback from the consultants about the results and their applicability, relevance and validity in normal Agile transformation consultancy. Four consultants attended the workshop that lasted altogether for about two hours.

The main message got from the workshop was that the results were seen relevant and potentially useful for the consultants in their work. Especially the ideas of process consultation were seen helpful in stressing the importance of consultant's self-awareness and delicate nature of the relationship between the consultant and the client. Consensus in the group was also that too often Agile transformation is seen as a production or technical problem and not enough time and effort are invested in understanding the organizational perspectives of the change.

Also the idea of having a dedicated change management group inside the client organization got support from the consultants. They mentioned that quite often clients actually assume that the consultants want to take the responsibility of driving the change and the clients themselves want to "step back" and let the consultants handle the situation. Some speculations about the reasons for this surfaced ranging from the fear of failing to the feelings of not wanting to disturb consultants' work.

Also some critique and open questions were presented during the workshop. The main challenge left completely unanswered by this study is how Agile transformation

should and could be sold. As the results of this study suggest, the concrete steps of the Agile transformation are always highly dependant on the situation at hand. This leads to a hard dilemma during the selling phase as there is practically nothing concrete or certain to sell. Consultants can sell neither pure training nor consulting, because they do not yet know what the situation requires. On the other hand it is imaginably very hard for the buyer to buy something as vague as "we will help you best we can" or "we assure you that with our knowledge you succeed".

Another challenge connected to selling somewhat fuzzy Agile transformation is the question about what actually should be sold. Should Agile transformation be considered as only taking Agile into production? Or is this more of a general organizational change program that just happens to lean on Agile values in order to improve the efficiency of the organization? Depending on the selected stance the selling should probably be done to different parties in the client organization or at least using different approach and language. As with the previous problem with selling, this study does not provide any clear answers to this either.

Yet another partly unanswered theme was about how consultants can justify their actions when they do not want to give direct answers or solutions to the clients, but instead want them to try and find those by themselves. Many participants of the workshop for example found it hard to defend the consultant's fees when confronted by client's questions like "Why don't you just tell us how we should do this?" or "What good can you provide us by just participating some meetings and sitting in the team room?"

Of course, from the process consultation viewpoint, the consultants might have been acting right when not taking too much of an expert role, but rationalizing this to a client that is paying an hourly fee certainly is a challenge. Based on the results of this study the best advice for the consultants would be to try to get the client more involved to the process in order for him to also see the benefits of learning to improve on his own versus doing what others suggest.

In conclusion, the arranged workshop proved very useful and set the results of this study to more concrete light. Also some weaknesses of connecting the results and real world business cases became more apparent. All in all the results seemed to

enjoy support from at least a small group of practitioners actually doing the consultation around Agile transformation. That, of course, is a promising sign.

7 CONCLUSIONS

This chapter concentrates on theoretical and practical implications of the study. Also the research questions are re-visited and discussion about answers to them is presented. After that, the results are compared to other similar studies. The validity of the results is also discussed. Finally some suggestions for further study are made.

7.1 Theoretical Implications

Even though the study did not concentrate on creating new theories, it still produces valuable insights about how used theoretical frameworks can be combined and how they support and interact with each other. This interaction is represented in Figure 10 that contains the relationships of the main used theories.

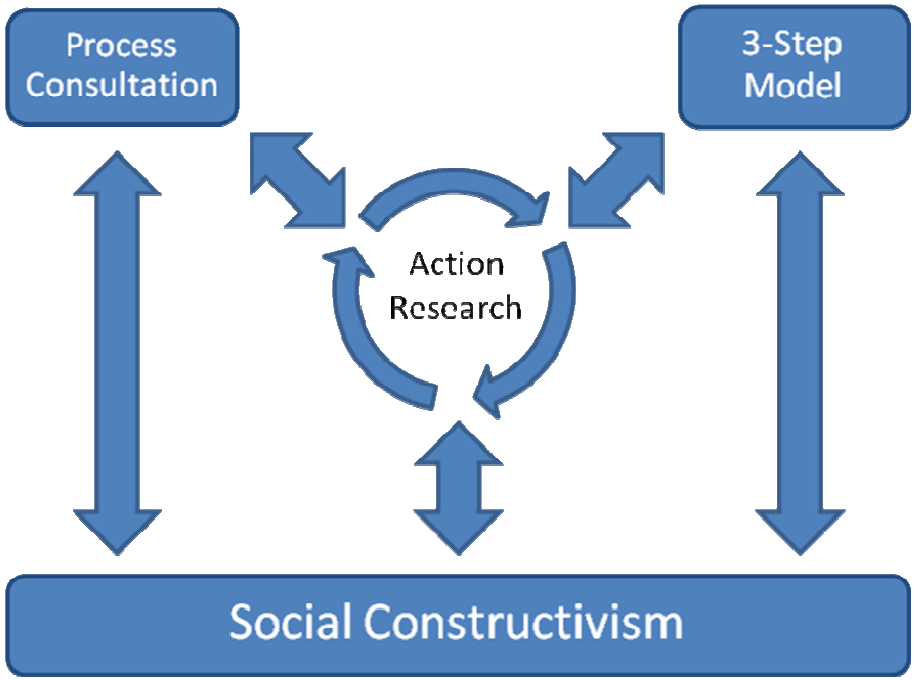

Figure 10: Interaction of used theoretical frameworks

As can be seen from the figure, Action Research gets the central place guiding the change agent's work during Agile transformation. In practice, the main contributions of Action Research are its iterative nature and emphasis on understanding why things in the organization are done in a way they are and why the organization responds to changes in a way it does.

Action Research is well supported by process consultation and Lewin's 3-Step Model of Change, however. These theories contribute to different areas of change, too. Process consultation mainly concentrates on guiding how a consultant should see himself and his relationship to his client. It also defines the main goals for the consultant and guides his actions towards producing as much help to his client as possible. In this way it gives more practical guidance compared to Action Research alone and thus complements it with a strong perspective of a change agent.

3-Step Model of Change on the other hand brings in a framework through which the change itself can be understood. It provides insights about what change is, how it is perceived and how one can control it. This clearly supports Action Research as one of the key elements during it is to understand why the systems behave as they do. Practically 3-Step Model provides the change agents with wisdom to start looking behind the immediate actions of individuals and to begin searching for underlying reasons for their actions. Schein's additions to 3-Step Model also give the change agents a practical framework with which they can understand how change can be achieved.

Even if the three mentioned theories comprise the core of the change agent's work, the social constructivism still provides invaluable views that can help applying the other theories. As social constructivism contributes to such fundamental levels of our understanding, it augments all other three theories. By stating that we do not live in a singular reality, it opens the world of interpretations and better understanding of different views in the organization. However, the most important contribution of it is still the way how language can be seen as a tool to both understand and influence the organizations and their members. By using language in this status, it gives practical means to apply other theories.

In addition to presenting how the used theoretical frameworks interact with each other, the study shows, that these theories can altogether be used in IT and Agile context and their applications provide reasonable results and suggestions for action. This itself is a valuable result and strengthens the faith that there probably is a great variety of other similar theories out there just waiting to be applied in the context of Agile. In this sense the study opens a window, if only a small one, towards the theoretical field that could well benefit Agile practitioners.

7.2 Practical Implications

The most central practical implications of the study are the suggestions for Agile transformation presented in Section 6.2. With these, change agents can utilize the potential of the used theoretical frameworks in their concrete work. The suggestions also provide examples about how different theories can be applied to the context of Agile. This again might help Agile practitioners further develop their working methods.

Also the identification of challenges in the case companies can provide practical benefits both for the companies themselves as well as to Agile consultants working in similar situations. By codifying such challenges into explicit form more understanding about the nature of Agile transformation could be created. Hopefully this new understanding helps all parties during the process. However, one must of course remember Lewin's point about the difficulties of creating objective diagnosis here, and understand that the created image of case companies as well as identified challenges reflect the reality in which the author lived during the study. As such, interpretations of same themes would surely be different from other perspectives.

The third practical implication of the study is that it brings visible the complexity and variety of organizational change context and how differently it can be seen. Each of the used theories provides its own perspective and paints different picture of the change. Understanding this variety and its implications to practice can help people working in the field to better see the change.

7.3 Research Questions

The research questions defined in the beginning of the study were:

1. Why is transformation from the traditional software development methods to Agile ones so difficult?

2. How should an Agile transformation be done in order to avoid the difficulties?

3. How could an external consultant best help his client organization during the Agile transformation?

The first research question is answered in Section 6.1 where different challenges encountered during Agile transformation are presented. Found challenges could be divided into three different categories: Issues concerning the fundamental concepts of software development, issues at macro level of the organization and issues at micro level of the organization. Summary of identified challenges is presented in Table 2 on page 55.

Second and third research questions are answered in Section 6.2 where suggestions for solutions are presented. Based on this section, Agile transformation should be done iteratively and it should be driven by an internal change management group that is fully committed to facilitate the change and owns it. Focus of the change management group should be on creating common shared understanding via organization-wide involvement and intense communication. Main goals for the change should be learning Agile principles and using them in continuous improvement of the organization.

An external consultant can help in Agile transformation by acting as a facilitator and coach during the process. His ultimate goal should always be to help his clients help themselves. Practically he should enable his clients see realities they are unable to see by themselves and provide them with insight with which the clients can better understand these realities. At the same time he should bring in domain knowledge about Agile principles and practices. Consultant's main tool during this process is language. It should be used to understand and influence the situations at hand.

7.4 Comparison to Other Research

Similar results that were found during the study have been witnessed by others as well. For example Dybå found in his quantitative study about software process improvement that strong business orientation and employee participation are related to successful process improvements (Dybå 2005). Dybå also stresses the importance of feeding back data gathered during improvement process to enable more effective learning. Dybå states:

> *"Our results suggest that the most effective use of data for organizational learning and SPI is to feed the data back in some form to the organization's members. Such feedback regarding actual performance not only motivates*

change in the behavior of individuals, groups, and organizations; it can also guide change in a specific direction"

With his stance Dybå is well in line with the propositions of Action Research discussed also in this study. It is interesting, however, that Dybå did not find any significant relation between upper management support and process improvement success. This is quite a strong contradiction to the results of this study as the top management support and commitment was held vital for Agile transformation to succeed. Dybå presents some explanations to this, however, and those might actually explain the differences in the results also between this and his study. The first of his explanations is the mixed and obscure definition of the top management support that might bias the results. Secondly he states, that the relevance of the top management support might actually be masked by other variables in the study. Thirdly he suggests that the management support is closely connected to business orientation and it might be shown in the results in this way.

In addition to Dybå, also Stelzer and Mellis have found similar results (Stelzer, Mellis 1998). In their study of success factors of organizational change in software process improvement they compared experience reports from 56 software organizations and found out that management support as well as employee involvement were the most cited reasons for success in the case descriptions. They also stress the importance of providing enhanced understanding about the current software processes and business activities to everybody involved in the change (Stelzer, Mellis 1998). Their findings also support the results of this study by pointing up that the change and improvement initiatives cannot be separated from other parts of the organization's business.

7.5 Limitations

As already was noticed in the previous subsection this study has its limitations. In this subsection they are inspected at more general level.

As the study is based on only three cases, the first limitation is of course the limited generalizability of the results. Even though the found challenges and suggestions might be relevant to the contexts of case companies and Reaktor Innovations, they might not fit well to other situations.

Also the selection of interviewed persons might limit the validity of the results. The selection was now made by the case company representatives after they were told about the subject of the study. Knowledge about the subject might have effected their selections and thus might have also biased the results.

It is also questionable how deep and extensive picture about an organization is achievable through mere interviews. As Lewin suggests with Action Research, more knowledge about the situation would surface through longer interaction with the client system and with the utilization of feedback cycles (Schein 2007). These were not possible with the resources available to the study, however.

It is also very probable that the subjective views and assumptions of the author have biased the study during the analysis phase. However, as social constructivism suggests, it is not only hard, but impossible, for any scientist, or any other person as a matter of fact, to describe an objective truth about some phenomena he has observed (Patton 2001). With this stance the study should be seen to represent the reality in which the author lived throughout its progress. This, of course, again limits the generalizability of the gathered results.

It is also possible that even if the suggestions presented in the study were valid, the Agile consultants would not be able to implement them. This could for example be because of lack of concrete bindings of the suggestions or due to the other factors such as already existing contracts, limited resources available or just unwillingness to accept the suggestions as valid and useful.

7.6 Further Research

This study showed that theories from organization development and behavior sciences can be used to analyze Agile transformation. However, the proofs about practical usefulness of such analysis were left missing. Even if some practical suggestions based on theory could be made, there is no direct evidence that these suggestions could benefit Agile practitioners or companies adopting Agile principles.

To gather such information, it would be interesting for example to perform a study among Agile consultants. During the study consultants' opinions and views about

suggested methods could be examined. Should the study last long enough, also some empirical evidence about the applicability of suggested solutions could be collected.

Another way to extend the themes of this study would be to concentrate more on the perspectives of the client organization. It would be for example very interesting to more deeply investigate how clients should act during Agile transformation and how they could assess the value of it in general. This kind of research would also help the consultants, as they could better understand their client's context.

More value might be generated also by extending the used theoretical basis. There is a multitude of for example organization development theories out there which could provide yet more understanding to this area. Also psychology and sociology could provide more useful frameworks. For example cultural anthropology could offer very interesting perspectives to behavior of organizations during such fundamental change processes as Agile transformation.

Instead of just finding more theories, also more practical methods of the same sciences could be utilized. For example in the area of work psychology there are many practical tools such as work conference (Lehtonen 2004) and future dialogue (Eriksson, Arnkil & Rautava 2006) that could well fit into the toolboxes of the Agile consultants.

REFERENCES

Abrahamsson, P., Salo, O., Ronkainen, J. & Warsta, J. 2002, *Agile Software Development Methods: Review and Analysis,* 1st edn, VTT, Espoo, Finland.

Baker, S.W. 2007, "Agile Principles as a Leadership Value System: How Agile Memes Survive and Thrive in a Corporate IT Culture" in *AGILE 2007*, ed. J.C. Thomas, 1st edn, IEEE Computer Society, Washington, DC, USA, pp. 415-420.

Beck, K. 2000, *Extreme Programming Explained,* 1st edn, Addison-Wesley, Upper Saddle River, New Jersey, USA.

Boehm, B. 1988, "A Spiral Model of Software Development and Enhancement", *Computer,* vol. 21, no. 5, pp. 61-72.

Boehm, B. & Turner, R. 2003, *Balancing Agility and Discipline: A Guide for the Perplexed,* 1st edn, Addison Wesley Professional.

Boehm, B. 2003, "Observations on balancing discipline and agility", *Agile Development Conference, 2003. ADC 2003. Proceedings of the*, ed. R. Turner, IEEE Computer Society, Washington, DC, USA, pp. 32.

Boehm, B. 2002, "Get ready for agile methods, with care", *Computer,* vol. 35, no. 1, pp. 64-69.

Boehm, B. & Turner, R. 2003, "Using risk to balance agile and plan-driven methods", *Computer,* vol. 36, no. 6, pp. 57-66.

Buchanan, D. & Boddy, D. 1992, *The Expertise of the Change Agent: Public Performance and Backstage Activity,* 1st edn, Prentice Hall, Hertfordshire.

Burnes, B. 2004, "Kurt Lewin and the Planned Approach to Change: A Re-appraisal", *Journal of Management Studies,* vol. 41, no. 6, pp. 977-1002.

Cloke, G. 2007, "GET YOUR AGILE FREAK ON! Agile Adoption at Yahoo! Music" in *AGILE 2007* IEEE Computer Society, Washington, DC, USA, pp. 240-248.

Cockburn, A. 2004, *Crystal Clear: A Human-Powered Methodology for Small Teams,* 1st edn, Addison Wesley.

Cohn, M. & Ford, D. 2003, "Introducing an Agile Process to an Organization", *Computer,* vol. 36, no. 6, pp. 74-78.

Dybå, T. 2005, "An empirical investigation of the key factors for success in software process improvement", *Transactions on Software Engineering,* vol. 31, no. 5, pp. 410-424.

Eriksson, E., Arnkil, T.E. & Rautava, M. 2006, *Ennakointidialogeja huolten vyöhykkeellä,* 1st edn, Stakes, Helsinki, Finland.

Juuti, P. & Lindström, K. 1995, *Postmoderni ajattelu ja organisaation syvällinen muutos,* 1st edn, Työterveyslaitos, Johtamistaidon Opisto, Helsinki, Finland.

Juuti, P., Rannikko, H. & Saarikoski, V. 2004, *Muutospuhe,* 1st edn, Kustannusosakeyhtiö Otava, Keuruu, Finland.

Larman, C. & Basili, V.R. 2003, "Iterative and Incremental Development: A Brief History", *Computer,* vol. 36, no. 6, pp. 47-56.

Lehtonen, J. (ed) 2004, *Työkonferenssi Suomessa*, 1st edn, Työturvallisuuskeskus, Helsinki, Finland.

Lewin, K. 1951, *Field Theory of Social Science,* Harper & Row, New York.

Mills, H.D. 1976, "Software Development", *Transactions on Software Engineering,* vol. 2, no. 4, pp. 265-273.

Moløkken-Østvold, K.J. & Jørgensen, M. 2003, "A Review of Surveys on Software Effort Estimation", *IEEE International Symposium on Empirical Software Engineering (ISESE 2003)*IEEE Computer Society, Rome, Italy, pp. 223.

Palmer, I. & Hardy, C. 2000, *Thinking about Management,* 1st edn, Sage Publications, Inc., Thousand Oaks, California, USA.

Patton, M.Q. 2001, *Qualitative Research & Evaluation Methods,* 3rd edn, Sage Publications, Inc., Thousand Oaks, California, USA.

Project Management Institute 2004, *A Guide to the Project Management Body of Knowledge,* 3rd edn, Project Management Institute, Newtown Square, Pennsylvania, USA.

Schein, E.H. 2007, 10/22/2007-last update*, Kurt Lewin's Change Theory in the Field and in the Classroom: Notes Toward a Model of Managed Learning* [Homepage of Society for Organizational Learning], [Online]. Available: http://www.solonline.org/res/wp/10006.html [2007, 10/22] .

Schein, E.H. 2004, *Organization Culture and Leadership,* 3rd edn, Jossey-Bass, San Francisco, CA, USA.

Schein, E.H. 1999, *Process Consultation Revisited - Building the Helping Relationship,* 1st edn, Addison-Wesley, USA.

Schwaber, K. 2004, *Agile Project Management with SCRUM,* 1st edn, Microsoft Press, Redmond, Washington, USA.

Seffernick, T.R. 2007, "Enabling Agile in a Large Organization Our Journey Down the Yellow Brick Road" in *AGILE 2007*, 1st edn, IEEE Computer Society, Washington, DC, USA, pp. 200-206.

Stapleton, J. 1997, *DSDM: The Method in Practice,* 1st edn, Addison Wesley.

Steindl, C. 2005, "From Agile Aoftware Development to Agile Businesses", *Software Engineering and Advanced Applications, 2005. 31st EUROMICRO Conference on*IEEE Computer Society, Washington, DC, USA, pp. 258.

Stelzer, D. & Mellis, W. 1998, "Success Factors of Organizational Change in Software Process Improvement
", *Software Process: Improvement and Practice,* vol. 4, no. 4, pp. 227-250.

Susman, G.I. & Evered, R.D. 1978, "An Assessment of the Scientific Merits of Action Research", *Administrative Science Quarterly,* vol. 23, no. 4, pp. 582-603.

Turner, R.J. 1999, *The Handbook of Project-Based Management,* 2nd edn, McGraw-Hill Companies, Maidenhead, Berkshire, England.

Weick, K.E. & Quinn, R.E. 1999, "Organizational Change and Development", *Annual Review of Psychology,* vol. 50, pp. 361-386.

Womack, J.P. & Jones, D.T. 2003, *Lean Thinking,* 2nd edn, Simon & Schuster, New York, USA.