Jani Antikainen

RELEVANCE OF SCOPE MANAGEMENT AND ORGANIZATIONAL CHANGE MANAGEMENT IN IT DEPLOYMENT PROJECTS

Master's Thesis in
Information Systems
27.7.2010

University of Jyväskylä

Department of Computer Science and Information Systems

Jyväskylä

ABSTRACT

This research investigates the relevance of project scope management and organizational change management in the context of IT deployment projects in which new IT artifacts are introduced in organizations. Scope management refers to the management of project content, whereas organizational change management refers to communicating, justifying and leading the change delivered by the project to the organization receiving the new IT artifacts. Deployment refers to the process of "taking the artifacts into use", involving both emergent and planned changes. This research provides new insights for both aspiring and current deployment project managers who need to understand better the characteristics of deployment projects, the motivations of project managers, and the most important success factors of deployment projects as those differ from software development projects and from other types of projects.

This research answers the following research question: Should project managers focus more on scope management processes than change management processes to reach deployment success? The author has identified these two critical processes through his extensive empirical experience in IT deployment projects. A conceptual literature study has then been used to deepen the understanding of the critical concepts. Finally, theme interviewing of IT deployment experts has been deployed to obtain relevant data in order to ensure that any personal biases of the researcher do not confound the results.

According to this research, IT deployment project professionals (1) value organizational change management more than scope management and (2) aim at reaching product success, not project success. The project success factors and criteria generally found to be of utmost importance to scope management do not seem to apply to IT deployment projects. Generalization of the findings should be done with care through further research.

KEYWORDS: scope, organizational change, deployment, project management

Dedicated to the memory of a great humorist and science fiction novelist, who has made our lives brighter and more full of smile. One characteristic of him, which is very much shared with yours truly and summarizes the writing process of this thesis to the point, is begging to be quoted:

"I love deadlines. I Like the whooshing sound they make as they fly by"

- *Douglas Adams (1952-2001)*

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

This chapter provides background on the field that this research is targeting. It explains the motivation and relevance of the research and presents the key research questions. Research methodology as well as the key findings and the structure of the research are also presented.

1.1 Background

IT projects and failures are far too often mentioned in the same context. IT projects do not have a good track record in project success according to many studies. Some of the reasons for failures can be masked behind the nature of IT projects: they are often complex in nature, full of uncertainties whether they are about e.g. requirements, technology, resources or quality (Vogelezang, 2007). Whatever the reasons for failures are, they are coming increasingly expensive and organizations try to understand the success factors but also the factors for failures.

Complexity of the projects is rather increasing than reducing. At the same time project managers, ultimately responsible for the project outcome, are struggling with fewer resources. This is due to IT costs generally getting reduced, and project managers having less time for themselves to concentrate on the most critical knowledge areas of their craft to make the project successful. This requirement of "faster and cheaper" makes the project manager think where to put his limited hours; time management, scope management, human resource management or something else, and how should one balance these areas. These are the practical problems this research is addressing.

Many researches indicate that *scope management* were the critical area to concentrate and they also claim that scope management is the single most important knowledge area/process, which dictates the project success or failure

(Agarwal & Rathod, 2006); (Boonzaaier & Van Loggerenberg, 2006); (Brown, 2004); (Carton, Adam, & Sammon, 2008); (Dekkers & Forselius, 2007); (Jugdev & Müller, 2005); (Kassab, Daneva, & Ormandjueva, 2007); (Lieberman, 2001); (Man & Leung, 2008); (Rozenes, Vitner, & Spraggerett, 2006); (Smith, 2002); (Syed Hussain, Rashid, Ahmad, & Syed Hussain, 2007) & (Vogelezang, 2007). To an extent the author agrees, but on the other hand, does not. The author's intuition and his solid (12+ years) experience with different types of IT projects raise their voice and say: "This does not necessarily apply to all project types and this kind of generalization might even make people draw conclusions resulting in not success but failure". Things need to be in the right context, otherwise generalizations, like the ones mentioned, can do more harm than good.

Having said that, the author does not downplay the importance of efficient and rigorous scope management – on the contrary. Scope management is very relevant in IT projects (whether they are projects creating IT artifacts (Orlikowski & Iacono, 2006) or projects deploying IT artifacts). Ultimately, the success of an IT project is a careful balancing of all project process areas (e.g. time management, integration management, human resource management, scope management, etc.) and is very context sensitive as all projects are unique in nature.

Via a literature study, it was soon discovered that very little literature exist on one specific IT project "type": *deployment*. And much of what existed tended to confuse deployment with projects containing software development, process development etc. The non-existence of literature was even recognized by some authors in the field of software development, like Caper Jones (2010, p. 12), with comments such as: "A major gap in the literature is that of best practices for installing or deploying large applications" and "Installation and deployment of

large software packages are common, but very poorly studied and poorly reported in the software literature".

Although Jones' comments are important and recognize a fact, with which the author can only but agree, they consider only a limited area of deployment. They generally miss (among many other studies/literature entries) the human aspect in the organizations receiving the change introduced by the deployment. Having gained experience from several very large scale (10 000+ stakeholders) deployment projects and programs, the author felt that the human aspect should be studied in context of IT deployment projects. Also the generalization of scope management's importance should be questioned. It was also important to understand, what the project managers aim to achieve – *project success* via traditional scope, time and budget success or were they aiming to achieve *product success*, even if not encouraged by their own IT organization? Do the IT organizations and the business organizations aim for the same target or not – several studies show that the linkage between business strategy and IT projects is weak and IT projects develop a life of their own. For example Steward (2008, p. 206) summarizes as follows: "[…]IT investments are often accompanied by poor vision and implementation approaches, insufficient planning and coordination and are rarely linked to business strategies". This clearly indicates that the dialogue between the IT organization and business should be stronger in the projects and among the people involved in the change (as projects are about change).

The above summarizes the purpose of this research: questioning the generalizations of scope managements' importance in deployment projects, investigating the human change factor – trying to understand these and the motives of experienced IT deployment project managers and how all of these are linked together.

1.2 Short introduction of key concepts

To minimize misunderstandings, the key concepts of this research are introduced briefly with syntheses they are given in the literature study part later in this thesis:

Deployment - *A project phase or a project responsible for introducing and taking into use the beneficial change (technology, operational model, process, …) for the receiving organization and the end users, which it receives from either the other project phases or as an input to a deployment project. Deployment is responsible for ensuring the needed change communication, training of the user population and approved handover of the beneficial change to the validated and ready operations function(s)*

Organizational change management - *Management of meaning, or with attempts to establish the credibility and legitimacy of particular definitions of problems and solutions with others, and to gain consent and compliance –* another, more commonly understood definition is used aside from the above mentioned in the interviews: *process of setting, managing and meeting the expectations of stakeholders, managing awareness and preparedness to ensure project success*

Scope management – *The cyclic process that extends the project lifecycle, which controls that all the work required, and only the work required, to complete the project successfully is contained in the project execution and no uncontrolled changes are permitted.*

1.3 Motivation and objective of the study

There are many motivating factors, which encouraged the author to conduct this research. Few of them are mentioned below:

- Deployment of IT artifacts is seldom trivial or easy, especially in case of undertakings that influence thousands of stakeholders and introduce

complex IT artifacts resulting in significant change in organizations' work practices (e.g. tools, processes, ways of working, roles and responsibilities). This is rarely identified in scientific or practioner literature on IT.

- The general confusion in literature about what *deployment* as a project phase or a separate project is about, what it might include and how important it actually is, and how little literature exists on the topic.

- The lack of recognizing the organizational and human aspect in IT projects, which is well summarized by Office of Government Commerce (OGC) in their ITIL Service Management framework (2007a, p. 165): "Within IT departments Project Managers often focus on the technical activities rather than the changes required for the organization or individuals".

- The author's notion from practicing project management in complex environments, backed up by but a few scientific sources. It can be highlighted by the following statements by Gareis and Huemann (2008, p. 771) :"[…][P]rojects and programs can be applied for managing change in organizations, the research community has been slow to respond", and: "[…][B]ut to date there has been very little interaction between the project management and [organizational] change management communities[…]". Henceforth, if the research mentions "change management" it refers to *organizational change management* (also in some literature considered as *human change management*), if not otherwise stated.

- Lastly, the author has had an intuition built over the professional years that the commonly used project failure/success reasons and highlighting of *scope management* as the ultimate entity to facilitate project success do not apply as such to IT deployment projects .

The author recognizes that entwining so many different complex constructs and concepts and their dependencies is usually out of the scope of M.Sc. thesis researches. Yet, the author does want to face the challenge and is confident that his years of experience in the industry allow him to handle and process the complex big picture. All the themes presented in this research are familiar and well known to the author and thus constructing the big picture view has become possible.

The scope of this research is to investigate two phenomena which the author recognizes from personal experience as very crucial for IT deployment projects (as well as for other project types): scope management and organizational change management. It should be mentioned that to limit the scope of the research, the concepts and constructs presented are researched only to the level where sufficient general knowledge of the themes is achieved and can be used for the empirical part of the study.

The context of this research is IT deployment projects and the focus is on the period in time equivalent to the project lifecycle. Lifecycles such as product lifecycle and IT service lifecycle are additionally introduced in the literature study to make a distinction between different lifecycles, but the focus of the empirical part is solely on project lifecycle. Again, this is due to the need to limit the scope of the work. If a IT deployment project was put into the context of a product lifecycle, then the motivations of the deployment project, the relevance of process areas and the success factors might be totally different from those concluded in this thesis. As an example, the relevance of release management – which covers a succession of several projects (some of which can be also parallel in execution) and drives to create releases where the IT artifacts created and further deployed can be managed in more controlled manner by splitting the "ultimate goal" into smaller pieces introduced over time– could be playing a major role as such or as a key success facilitator for scope management and

release management. This would be due to a single project's scope being smaller and more manageable. Also, the IT artifacts (i.e. the change) deployed would produce smaller one-time changes, which might dramatically help winning the change resistance always associated with introduction of a change.

The main objective of this research can be summarized as follows: Investigating which one, scope management (commonly acknowledged as main success factor for projects) or change management is more important for conducting a successful IT deployment project and how IT deployment experts perceive and conceptualize "a successful project" . All the themes presented are linked to this main red thread of the research. It should be mentioned that both scope management and organizational change management are needed to successfully conclude the deployment of IT artifacts. As both are important, this research aims to investigate which one should be weighted more (e.g. allocated more resources and attention) for successfully concluding the deployment project.

1.4   Research questions

There are three main research questions (Qs) which the author intends to answer. These questions have emerged from the author's extensive experience in IT artifacts creation and deployment and especially from the notions of the author, where he has felt that his personal empiric experience is in contradiction with scientific and popular beliefs. Hence the intuitional "This is not right"- feelings have been formalized to research questions, studied via literature and further investigated and enriched with further data from an empirical study of five IT deployment project experts. The key questions forming the standing stone of this research are:

Q1: Which one is more relevant in context of IT deployment projects for projects to succeed– *scope management* or *organizational change management*?

Q2: What do IT deployment project professionals consider most important from their own experience: *project success* or *product success*?

Q3: Can the studies and reports which refer to generalized IT project success/failure factors associated with scope management and/or the triple constraints of time, cost and scope be applied as such in context of IT deployment projects?

A wide range of research questions could be derived from the potent of this research and the findings presented, but to prevent the research from exploding in scope the focus is held on the three main questions.

For example, the research material would allow a research question such as: "Do IT and business organizations consider same goals (project success or product success) for deployment projects or is there a disparity, which suggests non-alignment of business strategy and IT implementing it?" The answer would be: "Several literature entries suggest such a possibility exists and the interviews committed relay the message that there is a disparity. In many cases the IT aims for stressing project success criteria, where the business wants to see product success criteria fulfilled. This results in different project goals being sought after, lack of alignment in project expectations, and change management not being efficiently made possible (as business is not involved enough in the change)".

1.5   Research methodology

The present research consists of two main parts. Firstly, a conceptual research based on literature review is conducted to understand the main concepts and constructs of the research and to synthesize the definitions and the conceptual models needed later. The second part is an empirical qualitative research based on semi-structured interview: theme interview. The research approach and plan is visualized in FIGURE 1. Research approach
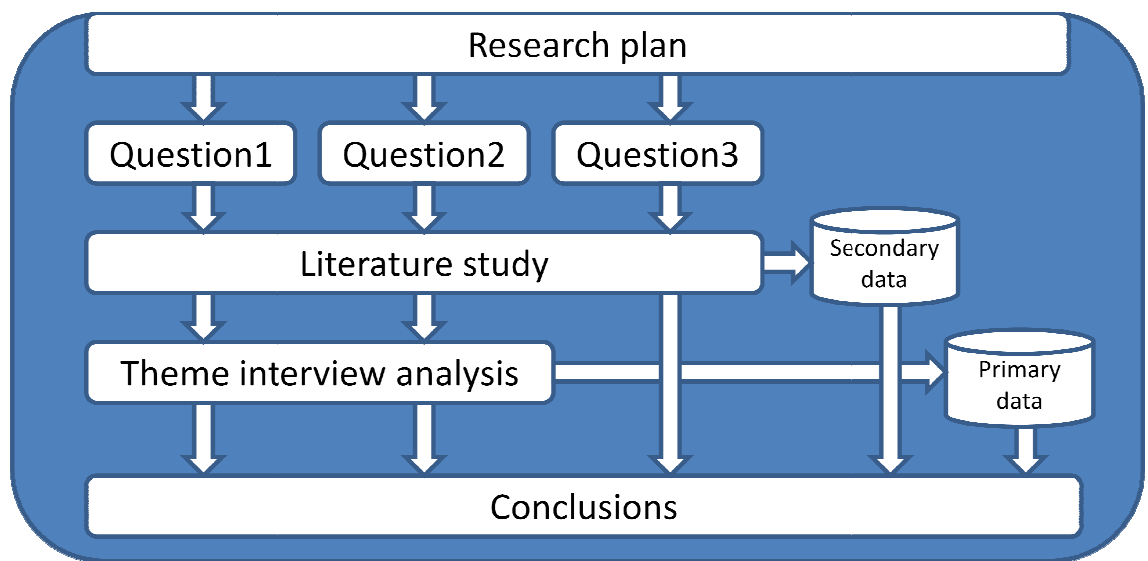
FIGURE 1. Research approach

FIGURE 1. Research approach explains that the research has started with a research plan (including initial literature study). The research questions have then arisen from the research plan. The conceptual models and constructions needed for the research questions are produced from the literature study part. Questions 1 & 2 are concluded almost as such from the interviewee answers in the Theme interview analysis part (Chapter 7), whereas Question 3 is answered in Chapter 8: Conclusions, limitations and future research.

The figure also shows the data collection approach in the research: the secondary data (the data already available as known/published and easily

available) is gathered from literature reviews and it works as input to the theme interviews; the primary data is the data gathered and analyzed from the interviews. Both the primary and the secondary data contribute to the conclusions.

Qualitative research conducted and the methodology used is further elaborated in Section 6.1. Possible limitations and reliability of the research are covered in Sections 6.3 and 8.6.

1.6   Key findings

Only the findings central to the presented research questions are briefly introduced here. All other, significant, findings are available in Chapter 7: Interview study analysis and Chapter 8: Conclusions, limitations and future research. The following findings should be seriously considered when either researching or practicing project management involving deployment activities and especially when practicing project management in IT deployment projects:

- For successful completing of an IT deployment project , *organizational change management* is considered more important than *scope management*. Although the reasoning is extensive, the main focus is on implementing the created change in the receiving organization, and thus organizational change management is seen as contributing more. The role of scope management is also recognized as very important, but less important than organizational change management. It is also noted that project success needs both, there is no black and white, but the focus should be on organizational change management.
- IT deployment professionals (of wide and considerable experience) consider the *product success* more important than the *project success.* This means that they are willing to sacrifice traditional triple constraints of time, scope and budget (on which their personal performance is often

metered in IT organizations) for the good of achieving product success. Very simplified, product success in context of IT deployment is considered as successful "taking into use and further usage" of the introduced change (i.e. IT artifacts: be it processes, systems, application etc.) which were to be deployed to the receiving organization(s) (usually business organization(s)).

- Much caution should be exercised when interpreting generalized IT project success and failure studies, statistics or critical success factors (CSF). This is due to a finding that in deployments the project managers consider project success only secondary and thus all the metrics associated with project success (time, budget, scope etc.) do not apply to them or the project as being the primus motors. For example, studies showing scope management as the single most important success factor in IT projects are very unlikely to apply to IT deployment projects.

1.7 Significance of the research

This research has been initiated from the author's keen interest on the topics covered and the author's perceived importance of these topics. The author has been lead by his intuition that the findings of the study are also very relevant to the scientific community, to the industry community of project management practioners as well as management trying to understand deployment projects and their unique nature.

This research contributes to rethinking the dominant classical models in IT project management by introducing a wide view on a very little researched topic, deployment and deployment projects. It also challenges over generalizing studies and statistics in project success/failure and the criteria contributing to the success of a deployment project. The perspective presented is challenging, as it weaves together many concepts and constructs and thus challenges

academics and practioners to integrate IT deployment, scope management, organizational change management, product/project success and several other themes introduced in the research.

As this research is limited in its data scope, namely the number of interviews, some care must be practiced when utilizing the findings of the research. Generalization should only be applied with caution and generally only to similar contexts as the research is presenting. Yet, the research presents a view and a new way of thinking not yet seen in any previous studies and offers many open windows for further research as elaborated in Section 8.8: Further research possibilities

1.8   Structure of the research

For being able to draw the "big picture" of the researched topic, the author is presenting the key concepts and constructs in literature study from Chapter 2 to Chapter 5. All the concepts and models presented later in this research, especially the themes investigated in the empirical part, can be grounded back to the entities in the literature part.

As the knowledge background is established, the approach and justification for the empirical study is explained in Chapter 6. Once the approach is elaborated and the interviewees are presented, Chapter 7 shows the analysis of the interviews separated into themes that investigate related phenomena or questions and it also reflects the author's personal opinions and findings in comparison with those in the expert interviews.

Chapter 8 draws the conclusions as answers to the presented research questions and provides other conclusions that are related to the research questions and can help further to understand the answers to the research questions. It also closes this research with suggestions for future research.

2 IT DEPLOYMENT

For the empiric part of this thesis it is relevant for both the readers and the interviewees to define *deployment* in context of IT projects. This chapter will start with inspecting different interlinked lifecycles (project, product and IT service (consists of one or several applications)), in which deployment needs to be defined. Furthermore, deployment will be inspected from the viewpoint of programs and projects governed under program(s). Lastly, derived from lifecycles and program/project -dimensions, deployment will be conceptualized as a *deployment project* in IT.

2.1   Project, product and IT service lifecycles

It is identified that there is no universal *lifecycle* for projects. Based on numerous definitions of projects and project lifecycles the simplest possible lifecycle is "start-end", this being based on the temporary character of a project, e.g. as per Slack, Chambers and Johnston (2007, p. 527): "[…] a set of activities with a defined start point and a defined end state […]". The author has witnessed this lifecycle-approach being used also, but the more common practice, especially in projects of significance and complexity, is to divide the project lifecycle into phases. Although the Project Management Institute does not provide any clear suggestions for universal project phases, it still recognizes the following lifecycle thinking: "Project managers or the organization can divide projects into phases to provide better management control with appropriate links to the ongoing operations[…]" (2004, p. 19).

2.1.1   Project lifecycle

It is common that project phases are sequential in nature (although overlaps do happen) and the deliverables from one phase to the next are validated prior to moving onward in the project lifecycle, i.e. the next phase (Project Management

Institute (PMI), 2004) & (Bonnal, Gourc, & Lacoste, 2002). Phases tend to take their naming from the activities which take place within the phase, e.g. requirements, design, build, test, startup and turnover (Project Management Institute (PMI), 2004); design, build, acceptance test, roll-out and handover (Van Bon, 2005) or definition, specification, design, construction, testing, installation and maintenance (Brandon, 2006). Variations are numerous, and it is of importance to note that the above mentioned examples of lifecycles (i.e. sum of the phases) all consider the project delivering an artifact(s) at the end of the lifecycle. Many lifecycles do even consider the official handover of the project's deliverables to the receiving body (e.g. operations, organization, customer) (Project Management Institute (PMI), 2004); (Van Bon, 2005); (Brandon, 2006); (Khosrowpour, 1999); (Office of Government Commerce (OGC), 2007a) & (Association for Project Management, 2000). With regard to the nature of a "deliverable" it can be a physical product, an intangible service or, for example, an organizational change. In context of this thesis "deliverables" are of class IT artifacts (whether referred to as a "change" or a "deliverable").

According to Turner (2006, p. 1) projects are to deliver "beneficial change". Whatever the nature of the beneficial change is, it needs to be taken into use, otherwise providing no value. An old proverb summarizes this well: "A fool with a tool is still a fool". In context of this thesis this "taking into use" is referred as *deployment* and it will be further discussed in Section 2.3. For the purpose of the project lifecycle, an essential part is to consider deployment as a specific project lifecycle phase, which in context of IT projects is well summarized in the ITIL-guideline (IT Infrastructure Library) as: "The Activity responsible for movement of new or changed hardware, software, documentation, process, etc. to the Live Environment" (Office of Government Commerce (OGC), 2007a, p. 232).

2.1.2 Product lifecycle

For clarification and avoiding confusion, the author chooses to introduce the *product lifecycle* to distinguish it from the project lifecycle. The Project lifecycle goes through a series of phases to create a product (be it a physical product or an intangible service). In context of IT projects, these are considered as IT artifacts. Still, several projects might follow to enhance the product in its lifecycle from an idea to retirement, i.e. the initial project creating the product may well be closed, but the product still continues its lifecycle. FIGURE 2 (Project Management Institute (PMI), 2004, p. 24) illustrates this relation between projects and product lifecycle from a business plan, through a product idea to the product's eventual divestment, with the project lifecycle covering only part of the product lifecycle:
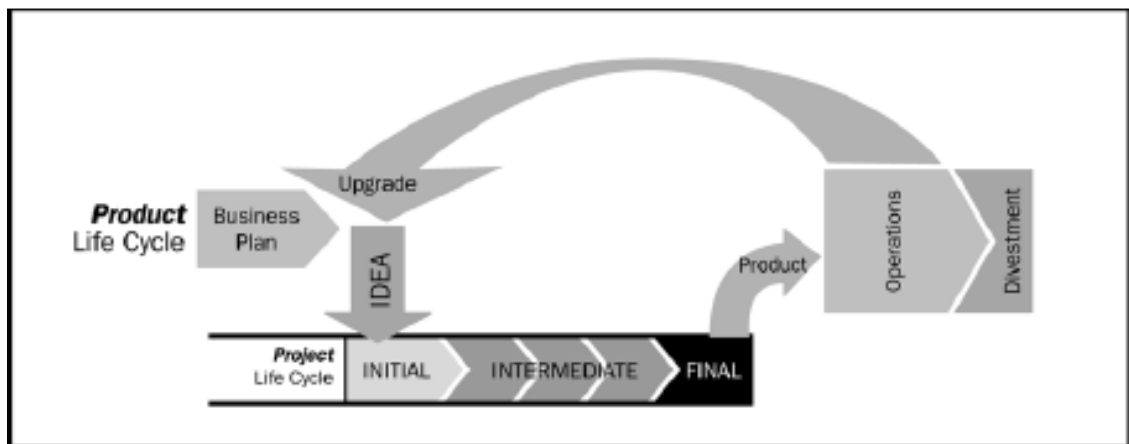
FIGURE 2. PMI's Relationship between the project and the product lifecycles
(Project Management Institute (PMI), 2004, p. 24)

2.1.3 IT service and application lifecycles

The ITIL framework has been recognized as a widely used good practice for *service management*. It works as a body of knowledge for achieving the ISO/IEC20000 standard audited and certified for organizations (Office of Government Commerce (OGC), 2007b). An IT *service* as defined by the ITIL is:

"A means of delivering value to [c]ustomers by facilitating [o]utcomes [c]ustomers want to achieve without the ownership of specific [c]osts and [r]isks" (Office of Government Commerce (OGC), 2007a, p. 244). Furthermore, the ITIL defines *application* as: "Software that provides [f]unctions that are required by an IT [s]ervice. Each [a]pplication may be part of more than one IT [s]ervice. An [a]pplication runs on one or more [s]ervers or [c]lients" (Office of Government Commerce (OGC), 2007a, p. 225).

The ITIL *service management lifecycle* is presented in FIGURE 3 (Office of Government Commerce (OGC), 2007b, p. 19) for promoting the fact that although an application might retire from a service, the service still might have other applications, and thus the service lifecycle continues on where a single application's lifecycle ends:
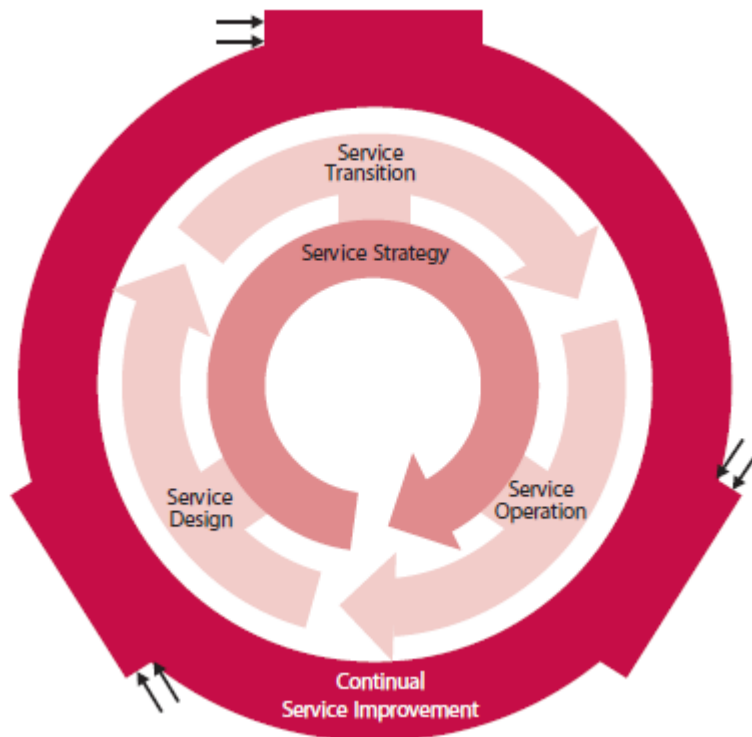
FIGURE 3. The ITIL service lifecycle (Office of Government Commerce (OGC), 2007b, p. 19)

The figure is explained by OGC (2007b, p. 19) as follows:

"The Service Lifecycle uses a hub and spoke design, with Service Strategy at the hub, Service Design, Transition and Operation as the revolving lifecycle stages, and anchored by Continual Service Improvement. Each part of the lifecycle exerts influence on the other and relies on the other for inputs and feedback. In this way, a constant set of checks and balances throughout the Service Lifecycle ensures that as business demand changes with business need, the services can adapt and respond effectively to them."

The relevance and interlinking of a project(s) and the ITIL service lifecycle is best shown in management of an individual application through its lifecycle. The ITIL framework underlines the usage of project management practices in many places, even with "business as usual" operations: "[…]formal [p]roject [m]anagement can be used to improve control and manage costs/resources" (Office of Government Commerce, 2007c).

Similarities can be seen in the application management lifecycle FIGURE 4 (Office of Government Commerce, 2007c, p. 130) and the examples mentioned for project lifecycle in 2.1.
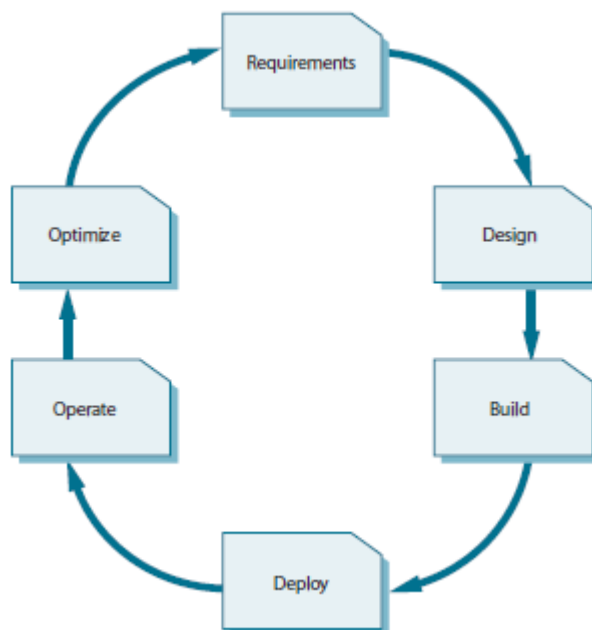
FIGURE 4. Application management lifecycle (Office of Government Commerce, 2007c, p. 130)

Requirement, design, build and deploy can be considered as phases in a single project's lifecycle or separate projects. Operate and optimize can be mapped to a product lifecycle and naturally to a service lifecycle and they present the ongoing operations and improvement of once delivered project 'beneficial change' as put by Turner (2006). For the purposes of this thesis the focus is put on the *deploy* lifecycle phase. The OGC explains that in *deployment* the operational model is incorporated into the existing IT environment and applications are getting installed on top of the operational model and testing is also taking place (2007c). Van Bon (2005) identifies activities such as planning the deployment, organizing deployment team, approving the deployment, distributing the application and piloting the roll-outs being part of this phase. More possible activities and what deployment phase or project is executing is explained in Section 2.3.

2.2 Projects within programs

For the purposes of Section 2.3 and understanding how deployment projects fit into larger IT endeavors, the concept of IT *program*s is elaborated. Many, even contradicting, definitions for a program do exist, but it is not necessary for the scope of this thesis to synthesize them. The Project Management Institute (2004, p. 16) sees program as: "[...]a group of related projects managed in a coordinated way to obtain benefits and control not available from managing them individually". While investigating roll-out projects, Bower and Walker (2007, p. 47) agree with the PMI's definition and describe a rollout program as: "[...] a group of related projects that are rolled out in a sequential fashion across multiple location and/or in the implementation of cumulative features" and further maintain that "Each project in a rollout program would have its own budget and schedule and possibly different project managers[...]".

For the purposes of this thesis, an IT program (or a *programme*) is defined as an *IT endeavor, which is of a size and complexity that renders managing the planned*

change more difficult as a single project and in which the holistic benefit of the projects in program mode, driving for the same common goal via a shared change target, is larger than the sum of the benefits of individual projects. There are various conceptual models of programs, where programs are more intended for *portfolio management* purposes or for driving an organization's business strategy implemented via project or resource allocation via projects (Association for Project Management, 2000). These models are ignored as being irrelevant for the scope of the thesis and thus only mentioned here. A relevant conceptual model of a program is well described also by Milošević, Martinelli and Waddel (2007, p. 457) in their principles of program management: "When an increase in project complexity and cross-discipline interdependencies becomes difficult to manage as a single project, redefine and reorganize the project into a program of multiple interdependent projects". This enhanced with Maylor's, Brady's, Cooke-Davies' and Hodgson's (2006, p. 671) finding on a program's nature: "[...program]refers to a collection of projects with a common objective [...], are the key elements the program definition presented earlier is based on, and which is of relevance in the context of this thesis.

2.3   Deployment and deployment project

Deployment was seen as a possible project lifecycle phase in Section 2.1 and further in that Section it was hinted that deployment could actually be a separate project itself – either as a standalone project or as a part of an IT program. In the scope of this thesis, to especially investigate deployment project and the context of program is of no greater relevance – a program was introduced as a possible lifecycle "holder" for a deployment project.

As with definition of a *program* in Section 2.2, the purpose is not to synthesize a holistic definition for deployment or deployment project. Deployment is examined to the level of getting understanding of the concept of deployment for the purposes of the empirical part of this thesis.

2.3.1 Confusing definitions of deployment, roll-out and implementation

Deployment, roll-out and *implementation* are all used very miscellaneously in the scientific and practioners' literature. There is a significant synonymic and homonymic character in the use of them, which at best is very confusing. For example, implementation is used as synonym for deployment in Shield's (2001, p. 24) ERP implementation literature: "[...] involve rollout strategies (which location gets implemented first and what is implemented there [...]". Similar term usage is identified in Stenzel's collection of best practices :"System [t]est & [r]oll [o]ut – [t]esting, training and implementation of new system" (2007, p. 103). Here *roll-out* and *implementation* both refer to deployment (as defined in the context of this thesis). On the other hand, *implementation* is referred to: "[m]ake, [b]uild and [t]est" in project lifecycle (Association for Project Management, 2000, p. 46). One more possible angle of attack to *deployment* is to consider the whole project lifecycle as deployment, although it would include specific application configuration and development activities (which do not fall into scope of deployment in the context of this thesis) – example being a rapid mail service deployment – a case study by Nangle (2003, p. 2): "[…] customization of the email web interface[…] and "[…]development of management and maintenance software for the email system[…].

Roll out (or *rollout*) is the most consistent, what comes to use of terminology. It is usually a synonym for deployment, although some wide scope definitions and uses do exist. Stenzel (2007, p. 112) explains production rollout as: "System is introduced to the people who will use it and they are trained for its use". Stadtler and Kilger (2005), in their *Supply Chain Management* book– although they mix implementation for deployment– give the following definition: "To ensure a smooth implementation of the SCM concepts in the *execution and deployment phase* […] implementation of the solution or business release is called roll-out". Brandon (2006, p. 62) covers many deployment activities when

writing about roll-out: "The *installation step* involves product roll-out, end user training, producing lessons learned documentation, and defining procedures for handling operations, user support and configuration management". Enforcing the actual synonymic nature of roll-out with deployment is promoted by Goodpasture (2002, p. 92): "Roll-out […] [d]eploy and apply the deliverables to the users". To stamp the definition for good, the ITIL (Office of Government Commerce, 2007c, p. 243) has the following definition for rollout: "Most often used to refer to complex or phased [d]eployments or to [d]eployments to multiple locations".

Considering the above examples, a definition for *deployment* is needed. The leading idea is that deployment does not create new functionalities, nor does it execute any major configuration or customization of the change solution. Example is shown in FIGURE 4, where deployment is clearly defined as an activity separate from *requirements, design* and *build*-activities. A border is also drawn between deployment and the *operate* and further *optimize* lifecycle phases. This indicates that deployment receives the change solution (IT artifact(s)) from the build-phase, takes it into use in the receiving organization and hands it over to the *operate*-phase. According to generalizations, deployment does not create new functionalities nor generic processes or roles. It can have the responsibility for localizing processes, roles and ways of working to multiple organizations in large scale deployments. However, any work considering creation of functionalities, global processes etc. is considered to happen in some other project phase or project (or a project part of a program). This is reflected in the ITIL definition of deployment (Office of Government Commerce, 2007d, p. 239): "The [a]ctivity responsible for movement of new or changed hardware, software, documentation, [p]rocess, etc. to the [l]ive [e]nvironment". It could happen, and usually happens that deployment notices a need for customization or other technical change, which is considered needed. It is not in scope of a deployment project to execute those changes, but to

inform the parties responsible for changes (like create-project) about the need. This could be visualized as a loopback from Deploy to Requirements (bypassing Operate and Optimize), in FIGURE 4. The deploy phase (or project) would then iterate some of the IT artifacts of previous phases (or projects), until the IT artifact changes are considered to be ready for deployment. Only after, perhaps, several iterations of this loop, the Deployment would handover to Operate. This fact is not considered in the simplified model in FIGURE 4 of ITIL framework.

Van Bon (2005, p. 190) considers deployment as a process which: "[…] creates or modifies an ICT solution, made up of one or more technical aspects, ensuring that technical and support capabilities are in place that enable the solution to become fully functional". Although Van Bon extends the scope of deployment to ICT solution creation area, he still highlights an important aspect of deployment, which is ensuring the existence of operational capabilities, once the change is introduced and taken into the end user use.

For context of this thesis and its scope *deployment* is defined as "*A project phase or a project responsible for introducing and taking into use the beneficial change (technology, operational model, process, …) for the receiving organization and the end users, which it receives from either the other project phases or as an input to a deployment project. Deployment is responsible for ensuring the needed change communication, training of the user population and approved handover of the beneficial change to the validated and ready operations function(s)*".

2.3.2 Deployment project and its activities

As mentioned in Subsection 2.3.1, deployment can be a project phase or a separate project. the rest of the Chapter will focus on justification of the project approach, i.e. why in many cases it is better to execute deployment activities as a separate project and on projectification of deployment, i.e. what possible

activities deployments consist of. A brief overview of deployment strategies is also introduced.

2.3.2.1 Justifying project approach for deployment activities

Justifications of a separate project are in many cases self evident: the deployment phases of IT endeavors can be very complex and, in their own right, be better organized as a single project responsible of them, while other projects concentrate on the other "phases" of the holistic delivery (e.g. requirements, design and build) (Office of Government Commerce (OGC), 2007a). Van Bon is stating (2005, p. 190) that: "Deployment [...] normally implemented as a single project or in number of consecutive stages or subprojects. Each deployment is essentially a change program or change project[...]". Speaking also for the deployment project approach is the British Office of Government Commerce (2002, p. 86) in its ICT infrastructure management best practices: "The deployment process is usually organized as a programme or project [...]." On very complex and high user count occasions, it is even reasonable to consider the deployment activities as being organized as a program with several deployment projects. This approach is suggested by Bower and Walker (2007, p. 49) in their phased rollout projects study: "[...] if each [deployment] stage has its own separately approved budget and schedule milestones, then it should be considered a separate project within a rollout program". An example from the ITIL framework of deployment of several components a service consist of is presented in FIGURE 5 (Office of Government Commerce (OGC), 2007a, p. 90)
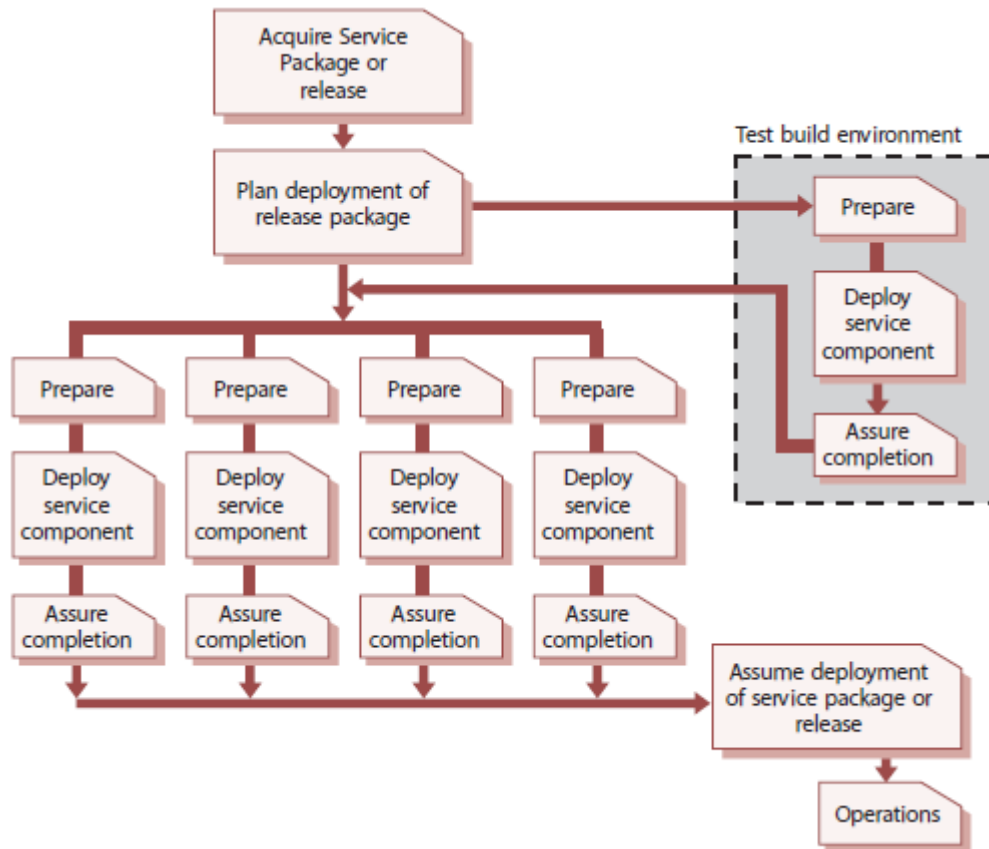
FIGURE 5. Example of coordination of service component is service deployment (Office of Government Commerce (OGC), 2007a, p. 90)

The example shows that a deployment project in an IT service context needs to consider a full range of elements which the service consists of – be they hardware, software, processes, documentation, knowledge, etc. These components, with their relationships and interdependencies, require very careful planning and deployment execution. Quoting the ITIL framework about the provided example in FIGURE 5: "Significant deployments will be complex projects in their own right". (Office of Government Commerce (OGC), 2007a, p. 90). To close and summarize the project approach as a best practice from the ITIL framework (2007a, p. 79): "It is best practice to manage several releases and deployments as a programme, with each significant deployment run as a project".

2.3.2.2 Deployment project activities

Goals of deployment have already been covered, but activities within deployment (projects) are of relevance also. Many have already been mentioned including testing, acceptance, training delivery, documentation of the change, communications, ensuring receiving operations capabilities, delivering the change itself and so on. Unfortunately, very few scientific or practioner works address the "fundamentals" of deployment. Many consider deployment as installing an application and that is it. Although well-known authors like Capers Jones do recognize the importance of deployment: "Make no mistake: installation, deployment , and training users of large software applications is not a trivial undertaking" (2010, p. 14), they still lack the understanding, what deployment fundamentally is – it is very seldom only installing an application and providing end user training.

Training still is an important part of the deployment and a part of the change process itself, like Stenzel (2007, p. 111) puts it: "[...]training always anticipates and supports the roll out into production, so that people who depend on the system are not left to cope with the change entirely on their own".

Involving the users is also recognized as important throughout the deployment – examples exist, such as Stenzel (2007, p. 129): "Business people in the pilot group smooth off the rough edges [...] [t]hese people sell the benefits of the system to the rest of the enterprise and often become the ones to train their co-workers[...]". The ITIL agrees with its framework's (Office of Government Commerce, 2007c) view: "The organization must first examine the processes that the tool is seeking to address and also ensure the staff are 'bought in' to the new processes and the way of working[...].

Although there is no definite and universal set of activities deployments should involve, as projects are always unique by nature (Maylor, Brady, Cooke-Davies,

& Hodgson, 2006), many still quite commonly do apply. It is not of relevance in the context of this thesis to investigate all possible deployment activities, but to demonstrate some possible activities by proving examples. Additional activities to those already listed above could include data sourcing, creating test cases, user acceptance testing, obtaining auditability certification, creating organization and role specific materials, job design, establishing measurement systems and data conversions (Watson & Kwak, 2004). The ITIL framework has solid understanding of deployment and thus an example from it is presented in FIGURE 6 (Office of Government Commerce (OGC), 2007a, p. 105):
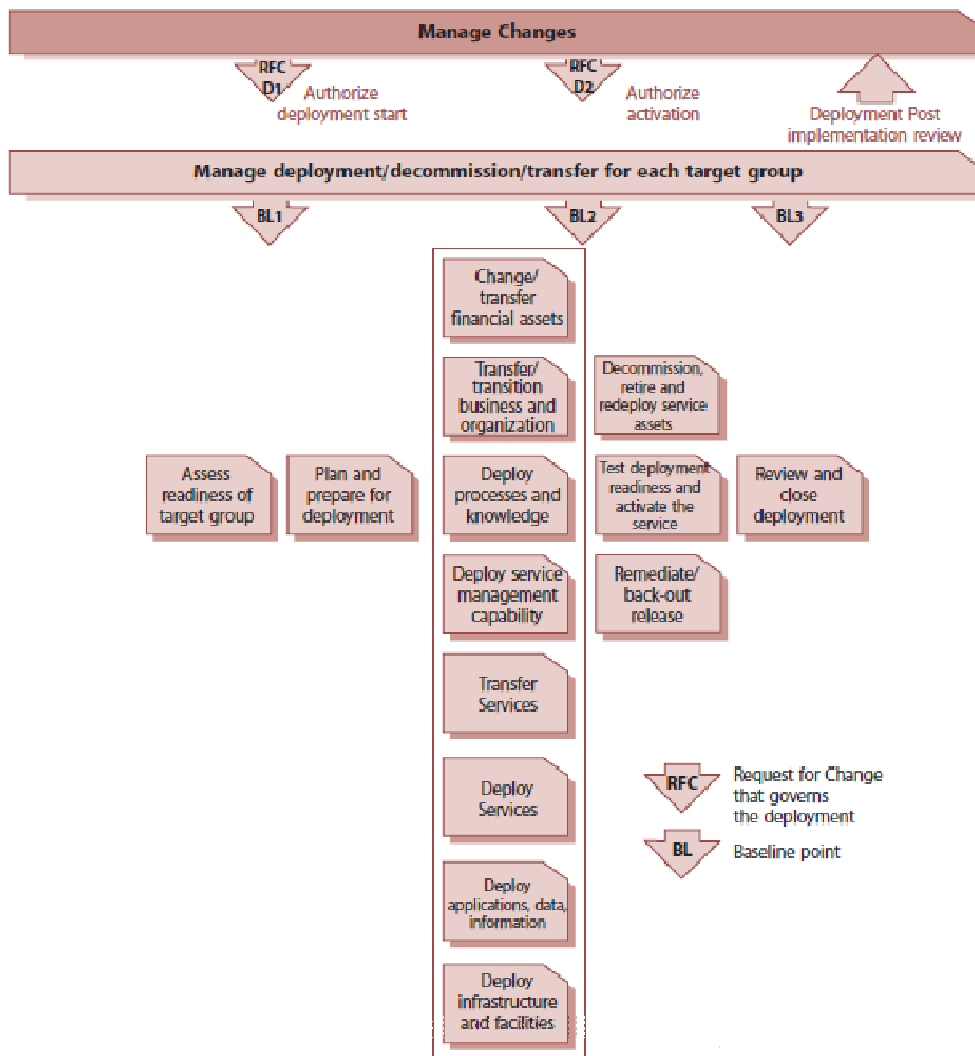
FIGURE 6. Example of deployment activities as per the ITIL framework (Office of Government Commerce (OGC), 2007a, p. 105)

The ITIL example shows the high level deployment activities and also presents the governance processes which direct the deployment activities. Example being the "Manage Changes", which is an embedded process to address all change requests affecting the deployment and authorization processes approving/rejecting the change requests.

With closing the topic, few example activities are still listed, which might prove very relevant later in the empiric part. The examples are under *communication* and *organizational change management*. Although there is research on the theme of organizational change management and the role of change in system implementations (like ERP and CRM implementations), there is a need to address more the relevance of efficient stakeholder communication and change management, and that is one of the reasons why this thesis exists. This is not to be confused with change management, which is by nature *change control*, e.g. controlled changes to developed application functionalities or activities within the project scope's WBS etc. Indications of importance of change management and communications can be identified from the ITIL framework in several notions, such as: "Communicating the proposed changes, the expected benefits and how the change affects the organization and staff"(Office of Government Commerce (OGC), 2007a, p. 95) and "Communicate change in organization, roles and responsibilities [...] [e]nsure that people adopt to and adopt new practices[...]" (2007a, p. 107).

2.3.2.3 Deployment strategies

The author identifies that there are several existing strategies and frameworks for optimizing between time to delivery and risk of the deployment. Usually these strategies are divided into three categories: *big bang*, where all users are introduced to the change at the same time; *phased approach*, where users are introduced to the change in smaller groups either sequentially or parallel or a combination of those (and variations), for example, running a small *pilot* for a

limited end user group and then introducing the change to the rest of the user population. (Office of Government Commerce (OGC), 2007a) & (Jeston & Nelis, 2006). An example by Buttle (2009, p. 90) is on large scale CRM deployments: "[...]a roll-out program is implemented. In larger companies this often is a phased roll-out [...] system might be rolled out first to 'champions', those identified earlier as buying in both emotionally and rationally".

All approaches have their advantages and disadvantages. Some fit some deployments well, whereas others are in some cases impossible to implement. For example, phased deployment is almost impossible for replacing an ERP system with a new, if the applications are not designed to co-exist – deployment has to be done with the big bang approach. The strategies were mentioned as acknowledged models for educational purposes and to enhance the credibility of this thesis in the empirical part. Otherwise these strategies are of no relevance for the scope of this thesis.

2.4   Summary on deployment

The concept of deployment was introduced and investigated in the context of a project phase, a separate project and also as a part of an IT program. A synthesis for deployment was provided and possible lifecycles of a project, a product and a service (with applications) were connected to the deployment concept. It was seen that deployment can be justified as a separate project and that deployment can also be a program with multiple deployment projects.

3 DEPLOYMENT PROJECT SUCCESS FACTORS

Project success seems to be a very intangible subject and no definite indicators exist to define whether a project has been a success or a failure. The classic triple-constraints of time-budget-scope (and/or quality) have been used to meter project success through decades (Jugdev & Müller, 2005) and still many project success researchers rely on those (Standish Group, 1995) & (Sauer, Gemino, & Reich, 2007) when dictating project success or failure. This chapter will investigate some project success studies made and their relevance. Further on, the subjectivity of success is elaborated. Going further, concepts of product success and strategic success of a project are presented to judge deployment success. The topic, project success, is relevant for the empirical study, as it is unclear by which criteria deployment projects are judged. Stating that, it is understood that "project success", if thoroughly studied, would include material for several Ph.D. –studies – hence the topic is merely scratched on the surface to the needed level to cover several aspects of it in the empirical part.

3.1 Project success indicators, factors and studies

Over the years, several studies, some more scientifically rigorous than others, have been conducted on the IT field to measure how projects have been performing. Perhaps the most famous and most quoted is the CHAOS Report by Standish Group, which has been repeated over the years from 1994 to latest 2009 to generate a trend on project success. The original CHAOS Report raised much debate with its dramatic figures of only 16 percent of projects being successful (on time, on budget and with planned functionalities) and 31 percent of projects ending up as failures (cancelled as dramatically impaired) (Standish Group, 1995). Although the figures have improved a little, they are still dark in nature as TABLE 1 (Eveleens & Verhoef, 2010, p. 31) shows:

TABLE 1. Standish project benchmarks over the years (Eveleens & Verhoef, 2010, p. 31)

Year	Successful (%)	Challenged (%)	Failed (%)
1994	16	53	31
1996	27	33	40
1998	26	46	28
2000	28	49	23
2004	29	53	18
2006	35	46	19
2009	32	44	24

The resolution types for the outcomes were (Standish Group, 1995, p. 2):

- Resolution Type 1, or project success: The project is completed on-time and on-budget, with all features and functions as initially specified.

- Resolution Type 2, or project challenged: The project is completed and operational but over-budget, over the time estimate, and offers fewer features and functions than originally specified.

- Resolution Type 3, or project impaired: The project is cancelled at some point during the development cycle.

Interpreting the Chaos figures, alarmingly few project are completed as successful. Chaos report has been criticized much in scientific and practitioner literature and much of the critic is valid. Once published, it created an atmosphere of IT project turmoil and mentality of IT project holistic failure. Important is to bear in mind that the Chaos figures consider only the triple constraints of time, schedule and scope. Many studies and authors state that judging projects on those constraints is a very limited approach and other metrics should be implemented today, e.g. Jugdev and Müller (2005), Thomas and Fernández (2008) and Agarwal and Rathod (2006).

Similarly dark figures can be found from several reports like the KPMG's survey on IT project management issues sent out to 1450 public and private sector organizations in 1997. At first glance a staggering 61percent of projects were considered failed. Again, it is important to understand the resolutions for "failed":

- The project budget was overrun by 30 per cent or more; and/or

- the project schedule was overrun by 30 per cent or more; and/or

- the project was cancel[l]ed or deferred due to its inability to demonstrate or deliver the planned benefits (Whittaker, 1999).

As with the Standish group report, the findings should be read with discretion. Deeming a project as "failed", if it is 30 percent over the budget presents things in a very black and white and scandal-seeking light. Also, clear scientific shortcomings can be found, for example, from the Chaos report. Eveleens and Verhoef (2010, p. 31) show one by stating: "[...]a project that's within budget and time but that has less functionality does not fit any category". They go very far in criticizing the results and deem them useless by claiming that :"[...] the Standish figures for individual organizations don't reflect reality and are highly influenced by forecasting biases. Because the underlying data [Standish does not reveal the raw data, nor exact research method] has an unknown bias, any aggregation of that data is unreliable and meaningless" (Eveleens & Verhoef, 2010). Further in their critic Eveleens and Verhoef (2010, p. 31) highlight one aspect, inspected in more detail in Sections 3.2 and 3.3, namely the context-dependency of the results – Standish results do not consider project context, like usefulness, profit and user satisfaction.

On the positive side, quite recent studies show more positive project success figures, although the same performance variances of schedule (time), budget (cost) and scope were used. A good example is a study by Sauer, Gemino and

Reich, which investigated the success of 412 U.K. project managers in their projects. Budget overspends were in average 13 percent, schedule misses 20 percent and scope under-delivery of 7 percent (2007). Although Sauer et al. state that the project managers in question were experienced, the figures are far more encouraging than those of the Chaos and KPMG reports and are based on more rigorous scientific study. For comparison to Chaos, the key findings of Sauer et al. are presented in TABLE 2 (2007, p. 80). In the Chaos report, the perhaps most significant figure is the percentage of canceled projects, which was 24 percent for 2009 (Eveleens & Verhoef, 2010). Compared with the 24 percent in the Chaos, Sauer et al. found a corresponding figure of "canceled" (abandoned) projects of only 9 percent (2007). Also, promising figures were presented in the category of "good performers", which enlisted for 60 percent of total and, in addition, 7 percent of projects completed as "star performers" in Sauer et al.'s categorization. Other categories were "Budget challenged"(5%) and "Schedule challenged"(18%).

TABLE 2. Sauer et al.'s five project types and their performance variances (Sauer, Gemino, & Reich, 2007, p. 80)

We used a non-hierarchical clustering method to combine similar projects based on the three performance variances: schedule, budget and scope. Variances were measured as a percentage of targets as originally planned. A schedule variance of +34% therefore indicates the project was 34% later than originally planned. The results suggested separating the 412 IT projects into 5 basic types described below.

Performance Variance	Type 1: Abandoned Projects n=38	Type 2: Budget Challenged n=21	Type 3: Schedule Challenged n=74	Type 4: Good Performers n=249	Type 5: Star Performers n=30
Performance Variances (Actual as % of Originally Planned) – 100%					
Schedule	N/A	+34%	+82%	+2%	+2%
Budget	N/A	+127%	+16%	+7%	-24%
Scope	N/A	-12%	-16%	-7%	+15%

Size Characteristics of IT Project Types
Four indicators of project size were measured: budget, effort, duration, and team size.

Size Characteristics	Type 1: Abandoned Projects	Type 2: Budget Challenged	Type 3: Schedule Challenged	Type 4: Good Performers	Type 5: Star Performers
Budget (Median, in £000s Average in £000s)	£1,000 £24,232	£625 £8,978	£500 £12,513	£450 £6,106	£2,000 £12,119
Effort (Average Person Months)	798	557	212	89	170
Duration (Average Elapsed Time in Months)	17.4	20.0	13.0	11.2	15.3
Team Size (Effort/Duration)	35.7	17.7	12.9	7.3	9.8

Although the Standish results, the KPMG results and the results of many more reports can be questioned and many come short in their scientific rigor, it remains a fact that these reports tell us that there is room for improving IT project performance and all effort (like this thesis) to improve the situation should be most welcome. Also, as it will be covered in the upcoming Sections 3.2 and 3.3, there is room for a wider perspective into project success and success criteria – perhaps we have outlived the period when projects were to be judged on time, schedule and scope solely (Jugdev & Müller, 2005).

3.2   Subjectiveness of project success

According to Hartman and Ashrafi (2002) *stakeholders (*individuals or organizations that are actively involved in the project, or whose interest may affect as a project execution or project completion (Project Management Institute (PMI), 2004)) generally do not have a clear idea on the success factors of a project and that they have differing views on what success constitutes of. How can a project be seen as successful or a failure objectively, if it is judged by a number of subjective views without consensus on the criteria? Considering the above, Myers' (1994) notion that success would be achieved when results are perceived as successful by stakeholders seems not adequate enough. This is reflected by Thomas and Fernández (2008) in their notion that human beings do have a tendency of underestimating challenges and overestimating their own capabilities and thus stakeholders might perceive a partial failure in a project, which was, in fact, successful in achieving almost near-optimal results.

It is obvious that there are no common, agreed, criteria for project success (Thomas & Fernández, 2008). Rathod and Agarwal (2006) even suggest that it is unwise, in absence of tools for deciding the targets, to deem any project a success or a failure. Further, on the subjectivity of the outcome, they elaborate that success criteria can be perceived from project internal aspect (i.e. time, scope and cost) and external aspect of the stakeholders (i.e. customer satisfaction) (Agarwal & Rathod, 2006). Adding more to the stack of shortcomings, it is noted that project success perception is changing over the project and product lifecycle (Jugdev & Müller, 2005).

Even with this little evidence in the literature, it is quite clear that project success is ultimately always a matter of perception and based on the values of the stakeholders and their expectations on the project. The best advice to tackle the subjective nature of project judgment is provided by Turner (2004, p. 350):

"Success criteria should be agreed on with the stakeholders before the start of the project, and repeatedly […] throughout the project". This is in compliancy with Association of Project Management summarizing (2000, p. 18): "It is essential that project's success criteria be clearly defined and agreed before significant development is initiated.".

3.3 Project success vs. product success

As shown in Section 3.2, project success is an ambiguous subject (Jugdev & Müller, 2005) and it is volatile to the perceptions of the stakeholders involved (Hartman & Ashrafi, 2002). It is also always contextual and has different success criteria (Association for Project Management, 2000). One more dimension, which needs to be considered for the empirical part of the thesis is, which is more important from the deployment perspective: project success or product success (the author uses "product" loosely here, describing the outcomes, the beneficial change that the project is to deliver via introduction of IT artifacts)? Juglev and Müller (2005, p. 19) put this into wording with: "[…] if project success is limited to the variables of time, cost, and scope – and the links to the product/service value are missing – then project management is perceived as providing tactical (operational) value and not strategic value". Understanding this dilemma, some highlights for understanding the "project vs. product success" are investigated.

"The operation was a success, but the patient died" – an often heard saying, but it can be also used in the context of project vs. product success to understand the paradigm. A good example, used by Jugdev and Müller (2005), is the Sydney Opera House: 15 years to build and over 14 times over the budget, yet considered a great success and an engineering masterpiece. So, in the context of Cooke-Davies' (1990) categorization, the Sydney case would have been a great product success but a miserable project management failure:

- *Project management success* is measured on traditional meters like time, cost and quality, and

- *project success* [=product success] is measured on the project objectives.

This "product success" can be also viewed from the angle of the value the project presents to the strategic mission of the company (product success), whereas "project success" is technical performance of the project (Cleland & Ireland, 2002). This can be translated to a fact that a project can reach project management success without business success and vice versa (Thomas & Fernández, 2008). Similar view is presented by Espinosa, DeLone and Lee (2006) with their view of IT project success being a combination of project implementation success and system success. For the purposes of the empirical part, a criteria coding created by Thomas and Fernández as a result of their mini-case study on project success criteria from 36 industry companies is utilized. It is presented in TABLE 3 (Thomas & Fernández, 2008, p. 736):

TABLE 3. Example of success criteria and their categorization (Thomas & Fernández, 2008, p. 736)

Success criteria	Category		
	Project management	Technical	Business
On-time	X		
On-budget	X		
Sponsor satisfaction	X		
Steering group satisfaction	X		
Project team satisfaction	X		
Customer/user satisfaction	X	X	
Stakeholder satisfaction	X	X	
System implementation		X	
Met requirements		X	
System quality		X	
System use		X	
Business continuity			X
Met business objectives			X
Delivery of benefits			X

The example in TABLE 3 well illustrates the dimension of project-product success differentiation, although it splits the "product" success further down to "technical" and "business" success criteria. The criteria listed highlight several

success criteria, which are less often encountered in literature: sponsor satisfaction, business continuity, project team satisfaction and steering group satisfaction (Thomas & Fernández, 2008).

As with overall project success, the distinction between project and product success seems to be contextual and unique in nature (as projects are also). Nevertheless, it is important to recognize the existence of these success dimensions (project-product). Also, there seems to be a sure shift towards a project success criteria definition culture, where project managers are given more room for manoeuvring within the project boundaries and consider items like client's overall objectives (product), user community, interfaces to neighbouring projects and organizational targets (Jugdev & Müller, 2005).

3.4   Summary on project success

Project success was perceived as an intangible subject without definite indicators and being subjective in nature. Still many reports and success criteria research consider time, scope and budget as most important success criteria and/or indicators. Many of the presented findings in those studies do not have much scientific value and are almost scandal-seeking in nature. Nevertheless, the fact remains that there is much room for improvement in IT project performance for achieving success.

4 CHANGE MANAGEMENT

It was noted in Subsection 2.1.1 that projects are instruments of change. Without change there is no progress (Office of Government Commerce (OGC), 2007a). Change is not merely about technology, even in IT projects – it involves organizations and requires awareness, cultural change, motivation and many more aspects inevitably associated with change (Office of Government Commerce, 2007c). Human nature has the characteristic of usually resisting the change, especially, if it affects them personally (Milošević, Martinelli, & Waddell, 2007). Several studies already show that effective management of the change is an important (even the most important) key success criteria when determining project success or failure (Shields, 2001); (Jeston & Nelis, 2006); (Office of Government Commerce (OGC), 2007a); (Beldi, Cheffi, & Dey, 2010) & (Boonstra, 2006).

Although significant research on both project- and change management has been executed, there has been very little engagement between these two and very little recognition from change management community on the value of projects in implementing change (Gareis & Huemann, 2008). This chapter will drill into reactions to change, management change and relevance of change management in IT projects. Understanding change and its management is crucial for successful deployment, this is highlighted for example by Markus (2004, p. 4) as the ultimate risk with inefficient change management and current situation IT projects: "One major risk in technochange—that people will not use information technology and related work practices—is not thoroughly addressed by the discipline of IT project management, which focuses on project cost, project schedule, and solution functionality".

4.1 Managing the change and human reaction to change

On the nature of change management Boonsta (2006, p. 39), quoting Bennis (1984), suggests that change management is: "[...] *management of meaning, or with attempts to establish the credibility and legitimacy of particular definitions of problems and solutions with others, and to gain consent and compliance*". As a definition for organizational change management, this definition is sufficient for this thesis. It is well aligned with notions of "meaning" and "consent" for example with Jeston and Nelis (2006) noting that projects often fail because they are focused on one-way communication of the change to the stakeholders and that activities often are just ensuring that the user *can* use the solution, not if they *want* to use it. Similarly on the "meaning"-aspect, Buttle (2009) elaborates the criticality of involving the stakeholders and selling (getting the buy-in) the objectives and vision to the stakeholders to establish the meaning for the project – Buttle refers to this as "internal marketing".

Although the above mentioned definition prepared by Bennis is the main definition used in this thesis, an additional, more popularly understood definition by Slusarenko is used along the one by Bennis. This is to ensure the interviewees conceptual models are in alignment with the ones expected by the this research's setting: "[Change management is a] *process of setting, managing and meeting the expectations of stakeholders, managing awareness and preparedness to ensure project success*" (Slusarenko, 2010).

Numerous models of change and change management exist in the scientific and practioner literature like the ADGAR model by Prosci (Hiatt, 2006) or Lewin's Change Theory (Schein, 1996). It is not within the scope of this thesis to compare the models, but to understand enough of human reaction to change to investigate its impact on deployment. The well established facts about change are that it is usually resisted, whether the factors behind are individual factors (e.g. locus of control), group factors (e.g. group cohesiveness) or organizational

(e.g. threats presented by the unknown, challenges to the status quo)(Mabin;Forgeson;& Green, 2001). For purposes of understanding human reaction to change, a fundamental building block of modern change management and understanding the change reactions is presented in FIGURE 7. It is author's adaptation of Kübler-Ross' emotional grief cycle.
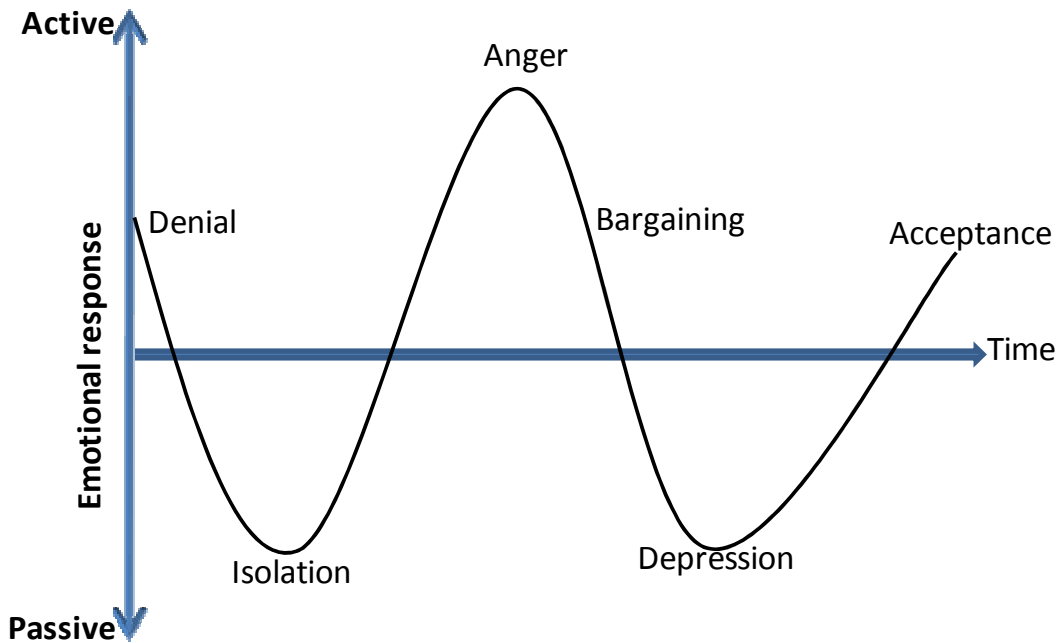
FIGURE 7. The emotional cycle of change (Freely adopted from Kübler-Ross' grief cycle (2009)

Although the model was originally presented based on Kübler-Ross' observations on people with terminal diseases or otherwise dying in the 1970s, it is widely used in cross scientific and practitioners' disciplines to investigate and describe change reaction and many theories and models are based on it.

For all significant changes, people go through the cycle in FIGURE 7, where the graph's illustration of emotional activity level can also be understood as work efficiency. Initial reaction is of *denial*, followed by avoidance of the change by *isolation*. The next phase of *anger* is associated with self blame. Later the "shoot

the messenger"-attitude will follow with *bargaining* the options available. *Depression* will follow as the ultimate facts of the situation are understood and ultimately, with time, *acceptance* of the change will happen. (Kübler-Ross, 2009)

There is a possibility that an individual gets stuck to any of the phases and could end up as a resistor for the change for a long time or never pass through the whole cycle and thus never accept the change. Organizational change management's challenge is to get people to process this curve of change for their own part as quickly as possible and to avoid people getting stuck into phases (such as denial).

This resistance to change is a commonly known phenomenon. Not a new phenomenon either, for example Zander (1950) recognizes it similarly in the 1950s as it is nowadays defined, and refers to it as being considered as individual's (part of an organization) behaviour to protect oneself from effects of real or imagined change. Shields (2001) recognizes the change resistance as responses from stakeholders, which will vary from *denial* (FIGURE 7), *wait-and-see* ("isolation" in FIGURE 7) *active resistance* ("anger" in FIGURE 7), to *excitement and support* ("acceptance" in FIGURE 7). The ITIL framework (Office of Government Commerce (OGC), 2007a) puts a lot of emphasis on change and change management and it recognizes five important ingredients of change: necessity, vision, plan, resources and competence:

> If there is no necessity established, there is lot of resistance from the people; if there is no vision, there is confusion among the employees; if there is no plan, there is chaos in the activities and transition; if there are no/fewer resources, there is a frustration among the employees; and if there is no competence, there is a fear of failure among the employees.

Along the lines of Gareis and Huemann (2008), Slusarenko (2010) states that project management and change management are inseparable and should not be viewed as separate activities, but an integrated process. Shields (2001)

recognizes this strong connection – he states that whether the change impacts are real or imaginary, they must be addressed as it is usually the people issues, not technology issues that make projects to fail. Shields (2001) underlines the importance further and proposes that projects should have an assigned person for primarily change management with tasks including:

- **Assessment**: Assess the readiness of the stakeholders for the change, identify different stakeholder groups to be impacted, identify power structures, and supporters and enemies of the change

- **Involvement**: Actively involve the stakeholders into project (design, testing, rollout, etc.), gather information and advice from stakeholders on processes, requirements, deployment strategies and organizational changes – ensure that input is asked and value opinions

- **Communication**: Ensure two-way communication with stakeholders: project status, newsletters, importance of project (messaged by top management), go-live dates etc – there is no too much communication

- **Documentation**: Making sure that enough documentation on the change and the possible artifacts of the change (tools, processes) exists and it is made available and communicated to the change targets

- **Training**: Make the change targets involved into training delivery and training material assessment to ensure efficient training – make training delivery known, i.e. informed and available to parties in need of it.

Identifying the inseparable nature of project- and change management, it is important to understand the relevance of change management to project management. This is investigated in the following Section 4.2.

4.2   Relevance of change management in project context

"It is clear that there is little benefit from putting in [deploying] new systems that are not used [...] people and processes have to change, and for the good" (Shields, 2001, p. 51). The above quote summarizes the relevance of change management to the mark. If people are not "buying" the change, they will not

adapt it and the investment is lost. The sole exceptional practice of project management in the rational view, regardless of its aggressive use is not enough to make the project success without considering change resistance (to be tackled with change management) (Shore, 2008). Shore (2008, p. 14) argues that: "[...]organizations resisting change and dismissing external threats, may have created an environment in which systematic biases should not be unexpected, even when the application of the traditional tools of project management is vigorously enforced", and "[...]then there are fundamental reasons why project failure should not be an unexpected results".

Several other quotes from the literature suggest that the importance of change management in IT projects is slowly getting recognized. Communication has been a part of the project management methodologies and bodies of knowledge for some time. It is still only one of the parts of effective change management as noted before, and it has to be two-way to provide value (Shields, 2001) &(Jeston & Nelis, 2006). For example, APM's (Association for Project Management, 2000, p. 50) Body of Knowledge promotes communication as very important: "Effective communication with all stakeholders is absolutely fundamental to project success". On the imperative nature of the change, communicating and managing it Beldi et al. (2010, p. 340) comment: "[...]failure to recognize the need for business change [is one of the] key reasons for CRM failures.". They also note on CRM failures that it has been documented that the most of CRM implementation problems have not been technical but related to organizational change and fluctuation (Beldi, Cheffi, & Dey, 2010). Even further emphasizing the importance of change management they note that communication, coordination and top management commitment (achieved through change management) are the keys to project success (Beldi, Cheffi, & Dey, 2010).

The role of the project manager is crucial in ensuring efficient change management (Boonstra, 2006). While managing the change and the stakeholders

efficiently, one is warranting attention from them and this seems to be a crucial success factor throughout the project (Boonstra, 2006). Slusarenko (2010) identifies this by noting that one of the surest ways to fail a project is the project and project manager ignoring the need to keep stakeholders involved and informed - people can only be introduced to change if they feel that their needs are recognized. A notion by Aladwani (2001, p. 266) similarly identifies stakeholder participation as important: "Also, convincing group leaders to effectively participate in the implementation process and make them feel that they are key players (because they are making key decisions) will ensure their valuable commitment. Because of their commitment, leaders of the groups will try to convince their colleagues that the ERP system is to their benefit".

The author returns to Boonstra's comment in the previous paragraph. If and when the role of a project manager is crucial in executing change management (among other tasks) and it is commonly identified (and one of the reasons for this thesis in the first place) that project managers are struggling with their limited resources, especially time, could it then be considered, as commented by Shields in the previous Section about the change manager role, that some of the tasks of project managers were delegated to roles and people having the best expertise for those areas (like scope management or change management) and the needed roles should be defined? This is further discussed in Chapter 8: Conclusions, limitations and future research.

To what extent and how many of the stakeholders then should be engaged and made buy the change is an abstraction and totally context based. Like in Aladwani's (2001) notion of involving group leaders to make the change happen, it could be even just a few individuals, who then convince the rest of the stakeholders to accept the change. Then again, in some situations, if the organization is very homogenous and without clear change champions, it might turn out that most of the stakeholders must be engaged either via getting

involved or otherwise made to buy the change. One could consider the ITIL framework's notion that Vilfredo Pareto's well known 80/20-rule from the 1800s applies to this too. The OGC claims (2007a, p. 172) in the ITIL framework that: "The Pareto Principle of 80:20 is an effective measure – once 80% of the people will let change happen (or even make change happen) you can move on to the next phase; the other 20% will follow". The OGC's principle is provided as an example to underline the author's finding that there is no established, commonly accepted estimation models, but that context-sensitivity is very high and "rules of thumb" like the one by the OGC are as good as any for estimation.

The evidence presented shows that change management is an important, if not even crucial, component of project success. It should also be integrated to project management processes for the whole lifecycle of the project (Jeston & Nelis, 2006); (Shields, 2001); (Slusarenko, 2010) & (Office of Government Commerce (OGC), 2007a). It is good to close the topic with guidance from the ITIL framework (Office of Government Commerce (OGC), 2007a, p. 84) on deployment management: "Ensure that organization and stakeholder change is managed during the release and deployment activities". This notion, made in several literature entries, is the fundamental secondary data finding to later justify the themes and their findings and conclusions in Section 7.7: Theme: scope management vs. change management – which one is more relevant; and in Section 8.1: Concluding research question 1.

4.3 Summary of change management

It was noted that although significant literature exist in both organizational change management and project management, very few studies connect these two, as stated few years back in 2008 in a respected project management journal, *International Journal of Project Management*, as a call for papers on the subject of uniting these two areas. (Gareis & Huemann, 2008). The human

nature is one of resisting change and a model for emotional cycle of change, as being crucial part of understanding organizational change management, was presented. It was also recognized that, in IT departments, project managers far too often focus on technology rather than the changes required for the organization or individuals. Organizational change management was perceived as "management of meaning", which summarizes it well. A note was made that organizational change management and project management should not be separated from the deployment project context, but that organizational change management should be integrated to project management processes as an integrated process extending the project lifecycle. On the significance of organizational change management, it can be summarized that if people are not "buying" the change they will not adapt it and the investment is lost, therefore organizational change management is very important in this research's context.

5 PROJECT SCOPE AND SCOPE MANAGEMENT

Project *scope* and *scope management* exist with several definitions and conceptual models, with no single dominating view. The importance of scope and scope management share the same fate. Project scope is seldom fixed during the project life cycle and it changes over project execution. Boonzaier and Van Loggerenberg (2006) sum it up that the tighter the scope control is, the higher the success chances for the project are. Similarly Lieberman (2001) argues that all projects must face changing requirements and that only with effective scope management those changes can be successfully added to the scope.

In this Chapter the different definitions and notions of project scope and scope management are presented and a synthesis of the findings is provided. Furthermore, the importance of scope and scope management in projects is studied; e.g. Brewer (2005) argues that in a project manager's knowledge portfolio scope management is one of the key competencies to successfully complete projects. Therefore understanding scope management should prove beneficial to parties engaged in project management activities.

5.1   Defining project scope and scope management

Literature is full of differing definitions and conceptual models of *project scope*. Several of those are explored in this Chapter, to first give the reader a view on the variety of definitions and conceptual models, so that the issue can be recognized. After introducing example definitions and models, the author synthesizes the definitions for both project scope and scope management. This synthesis is needed for the empirical part of the study and it will also work as a good overview of scope and scope management. Extended research and holistic literature study is out of the scope of this investigation.

Some scope definitions, like Lieberman's, clearly consider elements of *scope management* instead of purely defining, what is meant by *scope*. Lieberman sees project scope as an outcome of identifying the needs why project has been established and prioritizing those needs (Lieberman, 2001). Lieberman's contribution can be seen in identifying scope as sum of needs for the project in the first place.

Although scope can be investigated from the originating reasons for the project, it must also consider the expected work, outcomes and deliverables. Definitions from Brandon and Suresh consider the deliverables and tasks of the project. Brandon (2006, p. 401) defines scope as: "A description of the project work to be performed in terms of the desired results". On very similar lines is Suresh's (2005) scope definition: "Every IT project is executed with a set of deliverables, and has an expected closure time. Prior to this closure period, there are predetermined set of tasks and activities to complete the project successfully. These tasks constitute the scope of the project".

One more aspect, from which scope can be considered is the work executed in the project. For example Raz and Globerson (1998) identify work breakdown structure (WBS) as playing a central role in project as presenting the total scope of the project.

Except for Suresh's definition, none of the other definitions inspected this far consider any constraints as parts of project scope. In its simplest, only WBS is considered as scope (Raz & Globerson, 1998). The Project Management Institute (PMI) (2004, p. 375), a provider of detailed project methodology, defines scope as "The sum of the products, services and results to be provided as a project". The PMI's project methodology gives a more detailed listing of items, which can be categorized as parts of project scope. In the PMI's Preliminary *Scope*

statement, "definition of project" as per the PMI (2004, pp. 86-87)), the following items can be considered as project scope building blocks:

- Project and product objectives
- Product or service requirements and characteristics
- Product acceptance criteria
- Project boundaries
- Project requirements and deliverables
- Project constraints
- Project assumptions
- Initial project organization
- Initial defined risks
- Schedule milestones
- Initial WBS [Work Breakdown Structure]
- Order of magnitude cost estimate
- Project configuration management requirements
- Approval requirements

According to the PMI's project definition, i.e. scope, *Scope statement* considers also two other *triple constraint* items (additional to scope) time and cost, although the focus is on scope variables such as objectives, requirements, boundaries, deliverables & WBS. For the purpose of defining the scope, identifying the existence of constraints as part of the scope is essential. Section 5.2: Scope's relation to other project constraints and elements will investigate the triple constraints and other scope related phenomena in more detail.

As seen from a short inspection of scope definitions, several different views to the scope exist and none is including all the key elements from others (e.g. reasons for the project, deliverables, work, constraints and so on). WBS (project tasks) or requirements set(s) are not scope in a holistic sense, which involves aspects of business position, constraints, objectives, etc. A synthesis from the inspected literature for *scope* is thus defined in high level as: "*Project scope is the sum of needs for a project's initiation, services or products to be produced, and constraints of schedule and cost*". This definition considers what could be done

(services or products), why it should be done (needs to be solved) and how it can be done (time and cost), i.e. what, why and how.

Now that the project scope has been synthesized, there is also a need to synthesize the definition for *scope management*. Thus scope management can be seen as management of the synthesized scope definition in an IT project.

The PMI (2004, p. 103) defines scope management as the sum of processes needed to ensure a project containing: "[…] all the work required, and only the work required, to complete the project successfully." The PMI further states: "Project scope management is primarily concerned with defining and controlling what is and what is not included in the project." FIGURE 8 (Project Management Institute (PMI), 2004, p. 106) shows the PMI's mapping of processes, tools, inputs and outputs of project scope management as a process flow diagram:

Develop Project Charter 4.1 (3.2.1.1) — Project charter → Scope Planning 5.1 (3.2.2.2) ← Project Management Plan — Develop Project Management Plan 4.3 (3.2.2.1)

Develop Preliminary Project Scope Statement 4.2 (3.2.1.2) — Preliminary project scope statement →

Project scope management plan ↓

Enterprise Environmental Factors →

Scope Definition 5.2 (3.2.2.3) — Approved change requests / Requested changes →

Approved change requests / Approved corrective actions ↑

Organizational Process Assets →

Project scope statement / Project scope management plan (updates) ↓

Create WBS 5.3 (3.2.2.4) — Requested changes →

Project scope statement (updates) / Work breakdown structure / WBS dictionary / Project scope management plan (updates) / Scope baseline ↓

Direct & Manage Project Execution 4.4 (3.2.3.1) — Deliverables / Work performance information →

Scope Verification 5.4 (3.2.4.3) — Recommended corrective actions / Requested changes →

Performance Reporting 10.3 (3.2.4.9) — Performance reports →

Accepted deliverables ↓

Requested changes / Recommended corrective actions / Project management plan (updates) / Project scope statement (updates) / WBS (updates) / WBS dictionary (updates) / Scope baseline (updates) →

Close Project 4.7 (3.2.5.1) ← Organizational process assets (updates) — Scope Control 5.5 (3.2.4.4) — Integrated Change Control 4.6 (3.2.4.2)

FIGURE 8. PMI's process flow diagram on project scope management process (Project Management Institute (PMI), 2004, p.106)

The Association for Project Management (APM)'s (2000) definition in their Body of knowledge, APMBoK, defines scope management as a process of identifying and defining deliverables and work to produce – this being in line with what the PMI defined. Again, similarly to the PMI, the APMBoK states that scope must describe what is included in the project and what is excluded.

Providing synthesis for *scope management* from literature definitions is much easier than providing one for *scope*. Both the PMI (2004) and the APM (2000) are in agreement that scope management is the managing of what is included in the project and what is not – with the PMI (2004, p. 103) further defining: "[…] only the work required, to complete the project successfully". Still, the definitions could be elaborated to highlight the process-nature of scope management and

the fact that scope management starts from the very project initiation and lasts until the final acceptance.

An elaborated definition for scope management is thus: "*Scope management is the cyclic process that extends the project lifecycle , which controls that all the work required, and only the work required, to complete the project successfully is contained in the project execution and no uncontrolled changes are permitted*". This is not in contradiction with several other definitions like those of Webber and Webber (2007, p. 2.9): "Scope management, which ensures that all changes to the project scope are made in a controlled manner and that unofficial 'side agreements' between team members and end users are never permitted". The definition provided was also enhanced with change control aspect from Brandon's (2006, p. 401) scope management definition: "The parts of the overall management processes that deal with controlling the work definition".

5.2   Scope's relation to other project constraints and elements

It is fundamental to understand how scope relates to the other key constraints or elements of the project. Project scope plays a key role in defining the failure/success outcome of some projects. But defining or managing the scope of the project cannot be an activity, which does not relate to other aspects of the project, like schedule and cost that were mentioned in the definition of project scope in Section 5.1: Defining project scope and scope management.

It was mentioned in the introduction of the Chapter that thorough literature review is not in scope of the work. Thus only key findings which elaborate aspects of project scope's interaction with other project key concepts are mentioned.

Of the key aspects, the most important is the heavy dependency of scope, time and cost. Patton (2003, p. 2) sees the interlacing and dependency of scope, time,

quality and resources as a common dogma. "[…] you get 3 of the 4 [resources included]." and when resources are not included in the equitation: "[…] time, scope or quality, you can have any 2 of the 3". This strongly suggests on the heavy dependency of scope and other key constraints of the project: time, money resources and/or in quality. Following the given statement, one understands that changing one of the elements changes all or one of the others. This conclusion is strongly supported by Dekkers and Forselius (2007, p. 387): "[…] there are no scope changes without possible consequences to schedule [i.e. time], budget [i.e. money] and quality or risk level of the project. This is true vice versa as well. If the schedule or budget must be tightened, it may require changing the scope or quality requirements, or increase the project risk".

The above mentioned dependency is not "vanishing" anywhere, whether plan driven or agile project mode is selected. This is indicated by a notion by Sliger (2006) stating that should the project be executed in plan-driven development (traditional project model of defining requirements, fixing scope and executing) or in an agile model (iterative project model), both models still identify the same triple constraints: cost, schedule and scope.

The second key finding, which is of relevance is the notion in "Turner's five-functions of project based management" – model, where Turner (1999, p. 8) draws an image suggesting that *time*, *cost* and *quality* are mere "constraints" of the project in its execution context, which are influenced by project *scope* and *organization* (FIGURE 9):
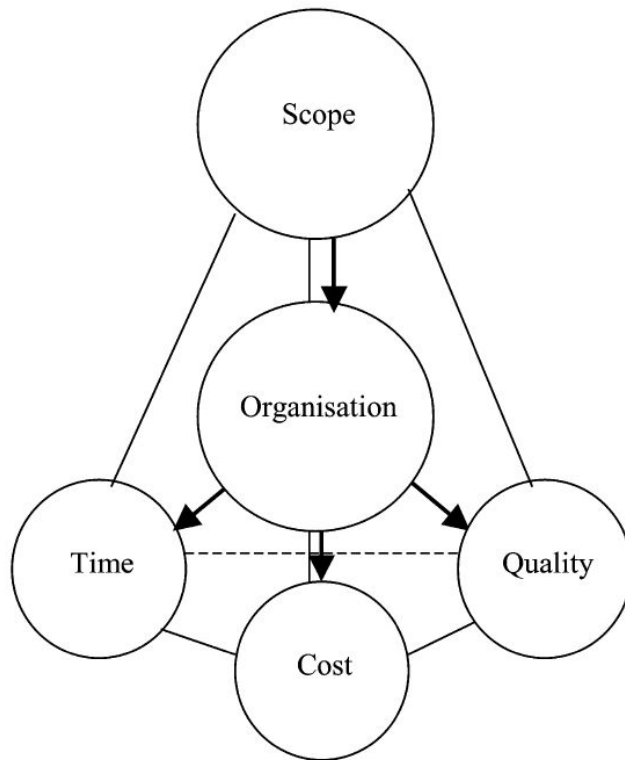
FIGURE 9. Turner's five-functions (Turner, 1999, p.8)

Turner's model explains much in a simple picture, but elaboration is needed. Scope of the project marks the project's work boundaries and it is managed by utilizing both *product* and *work breakdown*s. Product breakdown is the sum of elements the product has, just like work breakdown is. These breakdowns are derived from objectives from vision, mission, facility, team and individual objectives. Further, the scope is detailed by the project *organization*'s resources assigned to the project and it is managed via the organization's breakdown of the required experience and skills of the project members. Turner is considering scope and project organization as key functions of project management, thus time, cost and quality would only be constraints, which the organization and scope influence. In Turner's model, the arrows from scope to organization and further from organization to constraints (time, cost, quality) present this relationship as well as the solid lines from scope to time, cost and quality. The dotted line between time and quality are to include cost also and represents the

constraints as a group, which is influenced by the project scope and organization. (Lee-Kelley, Leong, & Loong, 2003, p. 584)

5.3 Scope's and scope management's relevance in projects

In many case studies and theoretical researches of IT projects the scope or management of it has been seen as either an important success factor (where scope is defined and managed efficiently) or a reason for failure / a reason for hindering the project execution and endangering the project (where scope is not well defined and managed). Several studies are presented next to demonstrate the perceived importance of scope, but especially the importance of scope management for project success.

In Brown's (2004) case study on enterprise resource planning (ERP) implementation at Heald College the first implementation attempt fails and Brown analyzes that some of the key failure factors are the lack of scope definition and scope control. On the second try of the ERP implementation, the needed changes were gap-analyzed, documented and used as items to create an agreement of scope of the project. The second try was successful (other factors than scoping affected the outcome, but scoping was seen as an important success factor).

In his identification of 40 root causes to troubled IT projects, Smith (2002) identifies many root causes that are derived from project scope. Smith claims that poor definition and management of scope are major reasons for project failures. Smith (2002, pp. 240-241) lists the following root causes as being outcomes of poor scope definition and/or management:

- Buyer failure to define clear project objectives, anticipated benefits and success criteria
- Buyer failure to define and document requirements (functional and non-functional)

- Vendor failure to invest enough resources to scope the project prior to contract
- Vendor failure to define project tasks, deliverables and acceptance processes
- Failure to ´freeze´ the requirements baseline and apply change control

Syed Hussain, Rashid, Ahmad and Syed Hussain (2007, p. 13) point out in their study of effective software management that among five causes recognized for project failure there are two that directly relate to scope and scope management: "[c]ontinuous changes in scope" and as an important part of scope "[i]ncomplete, and ambiguous requirements". A similar view is presented by Kassab et al. (2007) as they argue that only via realistic assessment (definition) of project scope in form of user requirements, can a project expect commitment.

In their study of using Turner's five-functions in IT services projects, Lee-Kelley, Leong and Loong (2003) identify a finding from the literature (quoting mainly Turner and Dey) that instead of focusing on the three factors (or constraints) of time, cost and quality, project managers would do well by focusing mainly on project definition and scope. This notion is further supported by Akinsola, Potts, Ndekugri & Harris (1997) in their finding that project organization, project definition and scope are the major influencing factors for delivering successful project objectives. Along the same lines, but also considering requirements and their estimation, are Kassab, Daneva and Ormandjueva (2007) claiming that only via making a realistic assessment on project scope can a project meet commitment and thus have the possibility for success. According to them such an assessment is heavily dependent on availability of knowledge on user-defined requirements of the project, their effort estimates, priorities and risks.

Relevance of the scope can be also investigated via inspecting the attributes of scope itself. For example, one dimension of project scope is the size of it, i.e. what is included in scope contra what the organization executing the project is

capable of accomplishing. Lieberman (2001) states it very black-and-whitely that should the project scope be too large, the organization will be overwhelmed and the project fails – on the other hand, should the scope be too narrow, the resulting system is unfit for the desired purpose.

Although the relevance of scope and management of it are almost the same thing, it is also good to inspect the relevance from the "action perspective". This meaning the execution of scope management, not just inspecting scope as a static entity.

"Defining and managing the project scope influences the project's overall success", argues the PMI (2004, p. 107). On process area of controlling the project scope, the PMI argues that it is concerned with influencing the factors creating project scope changes and the impacts of these changes. Further the PMI maintains scope management being responsible for ensuring that proposed changes are processed through change control process. An important function of scope management, by the PMI, is scope management processes' responsibility to manage actual project changes (not related to organizational change management) and integrating the changes with other controlling processes. The PMI states that uncontrolled changes are often referred as "scope creep". Scope creep is the undesired byproduct of a badly managed project scope, often leading to major difficulties in projects or being a reason for project failure.

According to Dekkers and Forselius (2007), scope management is more important than any other of the individual areas and processes in projects. To support this argument they provide information that 60-99 percent of all defects latent in production software could be attributed back to the requirements phase (software project specific statistics). Taylor (2001) supports Dekkers and Forselius by identifying in his project failure research that the requirements

definition, change management and scope management are the main reasons for a project to fail.

More emphasis on scope management's importance is provided by Man and Leung as they explore the success factors of CRM projects. They highlight that the most important success factor from an interview of 38 CRM professionals was scope management (Man & Leung, 2008). From the field of ERP implementations, a notion is presented that the single biggest reasons for ERP failures have been the project's inability to define the project scope according to business needs (Carton, Adam, & Sammon, 2008). Similar highlighting can be perceived from a finding about software project success criteria: "Software professionals with different responsibilities concerning software projects appear to a have a consensus on *scope* as the most important success criterion"(Agarwal & Rathod, 2006, p. 366).

Further listing some literature findings, in their literature review of "Project control" Rozenes, Vitner and Spraggett (2006, p. 7) have summarized the findings from several studies examining the success factors of projects. Among the findings from several studies they list: "[…] clear [project] goals, management support, ownership, a control mechanism, and communication", of which "clear goals" and "a control mechanism" can be translated into aspects of scope ("clear goals") and scope management ("a control mechanism").

An IT project always has a customer. The customer can be an internal organization or another company. Especially, if the customer is an external company, the project team is often responding to pressure to please the customer by allowing new requirements into the project (Lieberman, 2001). In cases where these new requirements arrive in a haphazard nature and lead to features not planned nor discussed outside the controlled scope, the project

schedule is ruined more efficiently than by any other single cause (Lieberman, 2001).

In a field study by Globerson and Zwikael (2002) ,which evaluated the extend the project manager was involved in the planning process and the quality of the process, the effort that project managers put to scope planning and definition was encouraging. The interviewed 282 project managers and other persons responsible for project management activities were, most of the time, delivering the major products of the process area (in case of scoping processes (scope planning, scope definition) the major products were *project deliverables* and *work breakdown structure*). On the used scale from 1 to 5 (1 meaning major product being hardly ever obtained and 5 meaning major product being obtained quite frequently) scope processes scored 3.8, being surpassed by time processes (3.9) and integration processes (4.0) from the PMI's nine areas of expertise. This study indicates that project managers are aware of the relevance of scope and the importance of scope management. Nevertheless, as other studies show, the actual scoping and scope management still has lot to improve.

5.4   The role of scope management in a project

In the definition of scope management it already turned out that scope management is a process area in the project lifecycle. It strongly interacts with other project process areas (Dekkers & Forselius, 2007); (Association for Project Management, 2000) & (Project Management Institute (PMI), 2004). Dekkers and Forselius (2007) identify that scope management is overlapping and interacting with all other 8 knowledge areas of the PMI's PMBoK: *integration management, time management, human resource management, risk management, cost management, procurement management, quality management* and *communication management*. Similar strong interaction is demonstrated by the PMI (2004) in PMBoK, where, for example, the project charter, an output from scope definition – process, works as key input for other process areas: project time management, cost

management, quality management, communication management, risk management, procurement management and integration management. Further emphasizing the notion of strong interaction, Dekkers and Forselius (2007) state clearly that scope management has a central role in an IT project and that scope management has strong relations to several of the knowledge areas, of which they identify time, cost, quality, and risk management. Visualization of this overlapping and dependency is presented in FIGURE 10 (Dekkers & Forselius, 2007, p. 3)

FIGURE 10. Positioning of scope management in PMBOK knowledge areas (Dekkers & Forselius, 2007, p. 3)

The more thoroughly the scope definition is done in the very initiation phase of the project, the better success possibility the project has (Project Management Institute (PMI), 2004). Similarly, argued by Baccarini (1999), it is critical for succeeding to have early project definition (scope).

The above mentioned efficient scope management's improvement of planning estimation (cost, schedule) can be better understood, when applied in the dimension of time, as constantly diminishing variance in estimations as the project proceeds. In the context of a software project this is demonstrated by Vogelezan's (2007, p. 2) statement that: "It is impossible to know how much software project is going to cost or how long it will take exactly until all the features the project needs to deliver are fully understood". Vogelezang visualizes this by his own adaptation of COCOMO "Cone of Uncertainty" shown in FIGURE 11:
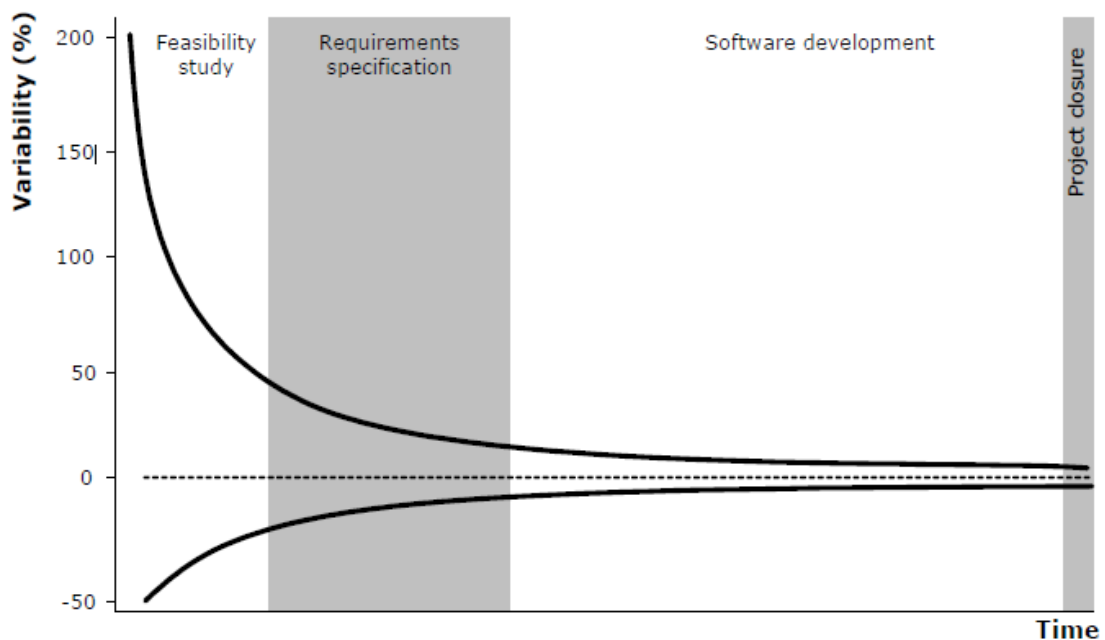
FIGURE 11: Vogelezang's Cone of Uncertainty (Vogelezang, 2007, p. 3)

Unlike the COCOMO cone, Vogelezang uses an asymmetric cone toward the upper limit, where the project uncertainty is at its highest. Similar models are numerous in literature, but their applicability is hard to extend to deployment projects, which seem to be different in nature from software development projects as this research will later show. Only few exist for any estimation effort of deployment projects and the ones existing, e.g. Watson's and Cook's "Parametric Estimating Model for Technology-driven Deployment Projects" are

quite at the very first steps of their evolution. For example, they hardly cover and consider the human factor of the deployment, which is recognized later in this research as a critical factor.

For deployment activities models, theories and estimation frameworks are far from those provided for software development (although it is not a perfect world either). A good example is an initiative project from 2008, driven by the International Organization for Standardization, which started to drive for standardized estimation data collection, analysis and data model for systems used for software project's estimation (Forselius & Käkölä, 2009). The author was not able to identify similar, "over the borders" initiatives for deployment projects, where the already suggested design theories would exist and standardization initiatives had started. Only several commercial and proprietary tools can be identified without any serious standardization attempt. Most promising of them is a parametric estimation tool *SEER for IT* (Galorath, 2008) from Galorath. It has support for lifecycle covering also deployment activities. It is a part of the SEER-SEM software family, which is quite recognized in the industry for example aerospace, manufacturing and banking. The SEER-SEM lists quite recognized users as Boeing, Ford, Lockheed Martin, Siemens, US Department of Defense etc. Alas, its knowledge base data, although printable by the user, is available only via the tool. Although the algorithms are also proprietary, they are made available to the user to enhance the credibility of the tool.

Differing from traditional, plan-driven or waterfall project management models and methodologies, agile methodologies embrace scope changes and thus the approach to scope management is quite different from, for example, regarding the PMI and the APM, which drive for fixing the scope. The philosophy is one where scope changes are expected to happen during the project lifecycle and changes to scope are deemed as positive events – fixing is done on cost and

schedule, not scope (Sliger, 2006) & (Patton, 2003). The mapping of scope management processes between the PMI's PMBoK and agile development is provided by Sliger. Sliger (2006, p. 5) has developed the following mapping of PMI processes vs. agile processes for plan driven development project managers to understand agile:

Initiating ⟶ Envisioning

Planning ⟶ Speculating

Executing ⟶ Exploring

Controlling ⟶ Adapting

Closing ⟶ Closing
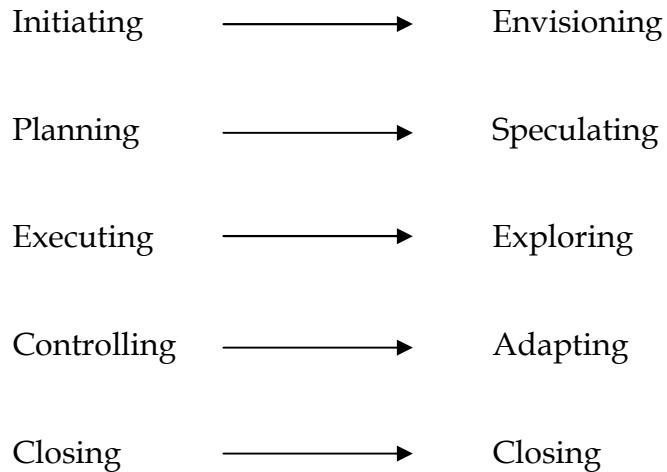
Scope definition is thus not a "onetime process" taking mostly place in the initiation of the project as it is by the PMI and the APM. In agile methods scope management is an ongoing process: scope change control, which happens with each of the development iterations the project has, the needed scope changes are identified and handled as product backlog (Sliger, 2006). A note should be made, though, that Sliger's adaptation model from the PMBoKs plan driven world to Agiles iteration encouraging world is quite simplified. Perhaps the best context for it is to isolate it to software production, for what Sliger intended the mapping for. Also, where the PMI's PMBoK Executing process-group is the "production" of the IT artifact with all the resources available, the "Exploring" of Agile might leave an impression of "trying this and this out" for ages, but it is aiming for the very same outcome, the actual production of the artifact. That being said, the author would claim that much of the work done in the PMI's Executing process area is actually combined in Exploring and Adapting of Agile and thus those mappings should not be considered 1:1 and that Control

processes of the PMI do not play as significant role as Adapt in the Agile mode of operation does.

The importance of scope and management is well demonstrated. One of the troubles the author has perceived from his working environment is that project managers are more and more taxed with responsibilities and driven by tight schedules. This was the reason why scope management and change management are compared in this thesis – lack of time and need to concentrate on critical factors. Another option that the author sees is to reduce the workload of the project manager by delegating some of the responsibilities to other roles. A scope manager is one, and it does already exist and is driven forward, for example, by the Finnish Software Measurement Association (FiSMA). The role of the Scope Manager is elaborated by Dekkers and Forselius (2007, p. 388):

> "southernSCOPE reported that some of the most effective scope managers are software measurement practitioners whose expertise also includes software development and subject matter expertise. 'Metrics experts observe and measure without any vested interest and as such provide unbiased and independent assessment of the project risk, quality and status. The measurement results support these observations…The Scope Manager provides metrics based project governance. We have found this approach to be very successful in objectively quantifying key project attributes to enable informed decision making with respect to project estimates and project risk…The Scope Manager is typically a metrics specialist who has excellent skills in business analysis, project estimation and functional size measurement. They need to be independent of the project team and not be connected to either the IT developers or the business client. They have to be able to report the status of the project objectively without bias, to a management level that has the authority to proceed, change direction or cancel the project."

Immediately, such a role definition can not be used as such in the context of a deployment project. The above role definition focuses on software development and creation and it would need to be adjusted heavily to fit the deployment context. The author still considers that such a modified role, helping project manager out with scope activities and ensuring a project's or a program's

consistency in "what we are to do" would be beneficial. The author will return to this subject in Chapter 8: Conclusions, limitations and future research. It is also identified in the context of organizational change management that separate roles could be beneficial to the project or program (Section 4.2).

5.5   Summary of scope and scope management

Findings from literature on definitions of scope and scope management were investigated and a synthesis for both was provided. From extensive literature found on relevance of scope management for project success, it is concluded that the topic has been researched a lot and the importance of scope management is much highlighted. Very few studies suggest scope management not being relevant for projects. These main findings are investigated further in the empirical part of the research in context of IT deployment projects.

6 EMPIRICAL RESEARCH

This thesis has been based on the author's intuition that the research subject is relevant, up-to-date and the research is bringing value to both practice of science and industry practice. A strong convergence can be found in the research, as several domains and concepts (scope, change and management of those, deployment projects and project success) need to be knitted together to understand larger motivations and causes.

In the empirical part of the research these seemingly stand-alone concepts and constructs are merged together, their interconnections are investigated, differences in importance are identified, and research questions are answered. The empirical part is based on the qualitative research method using the approach of semi-structured interview, theme interview. This Chapter will also elaborate the background of the empirical research, the methods used, introduce the interviewees and explain the data gathering process and analysis of the data.

6.1   Used research methodology and approach

A general guideline for empirical studies, suggested by Hirsjärvi and Hurme, is used for executing the research. It comprises of four main stages (Hirsjärvi & Hurme, 2008):

1) Defined research problem/question (select one with relevance for the community, not one which is just "easy" to cover);
2) Further study of the research problem/question, further refining it and looking into literature and existing research;
3) Collecting the data and analyzing it (observation, interview, databases, ...); and
4) Forming the conclusions and reporting the findings.

The research method used in the empirical part is qualitative and the approach was selected, as the target in the research is to consider also the past of the interviewees, the evolution in their expertise, and their notions of the research phenomena (Hirsjärvi & Hurme, 2008). Unlike often with quantitative research, there is no expectation for clear causal relationships between the interview themes and questions, although many are identified in the research.

Interview, as the data gathering approach, was selected for several reasons, the first being that the subject of the research with its many dimensions and domains is so complex that trying to gather data via, for example, surveys would be very error prone and require a lot of elaboration in the questions, and most probably result in a very low response rate and mixed data. The second important reason is that there is a need to get further clarifications and "why so"s from the respondents and there is a need to deepen the received answers with opinions and free elaborations. The interview technique used can be categorized as half-structured interview, *theme interview* as per Hirsjärvi and Hurme (2008), which is based on *focused interview* presented by Merton, Fiske and Kendall. Most fundamentals of focused interview are well suited for this research as it provides the following characteristics (Merton, Fiske, & Kendall, 1956, pp. 3-4):

- It is known in advance that the interviewees have experienced certain situation;

- The researcher has researched certain phenomenon's presumably important aspects, structures, processes and holistic nature;

- Based on the analysis, researcher has reached certain assumptions about the ruling factors of the researched situation and causes of it to ones experiencing it; and

- The interview is directed on the subjective views of the interviewees, those that the interviewer has earlier researched.

Most of the characteristics fit well into this research's design and purpose, although in this research, one similar "certain situation" (experimentally produced) does not exist. The situation (experiencing deployment projects), although it could produce similar conceptual models from all interviewees, is still a situation, which varies a lot. Because of this "shortcoming" of a clinically correct focused interview, theme interview is chosen. Although very similar to focused interview, it still allows all the interviewees' experiences, thoughts, beliefs and feelings to be researched without an experimentally produced "certain situation" (Hirsjärvi & Hurme, 2008). This interview approach can also be recognized as "the general interview guide approach"(Hirsjärvi & Hurme, 2008).

6.2   Background of the respondents and the interviews

Five deployment project experts were selected for the theme interviews. Respondents were selected by the author recognizing their work history as considerable in deployment activities and especially large scale IT deployment projects and programs. The background details (experience in IT projects, number of deployment projects, sizes of projects, …) on the experience of the respondents is provided in detail in Section 7.3: Theme: experience of the respondent in IT and respondents' project history. Although currently working in varying roles like IT Product Manager(s) (Respondents 3 and 4), Production Manager (Respondent 1), Program Manager (Respondent 2) and Directing Consultant (Respondent 5) and not directly managing projects (except respondent 5), they either involve themselves into project work or own project portfolios. Hence, their past roles as Deployment project managers and the current ones rather provide a wider perspective than exclude the other.

All the respondents have experience both from small scale projects, like maintenance releases of IT systems to large scale and complex introduction of

IT artifacts, like Product Data Management (PDM) system creation and deployment to 10 000+ users globally and across department borders. Other examples list creation and deployments of 15 000+ user requirement management system, similar scale error and change management systems, creation and deployment of a large telecom company's online stores, etc. To summarize, all respondents have experience in endeavors very few of us have the opportunity to participate in.

On the selection criteria, the author has also acknowledged personally that the selected persons have been very successful in their deployment activities. This fact even further assured the author on the selection choices as he wanted to investigate, what successful deployment project experts value (e.g. project success or product success and what criteria drives them onward) when they drive the projects for success.

Possible shortcomings of the selection criteria are explained in the following Section 6.3: Reliability of the selected interview approach. At this point, the author wants to recognize the fact that the interviewees are considerably experienced, they are perceived as being successful, and they all have looked at the deployment projects from several roles. Therefore it is quite possible that selecting respondents with different criteria, e.g. "not experienced", "not being successful" and "no other experience than project manager experience" would produce different data than the one presented in this study. This was acknowledged already at the very planning of this study and the selection criteria was formed already back then to include persons the author has proven experience with, who are recognized in the work community as successful deployment experts and who have experience in large deployments (>1000 stakeholders).

All respondents were asked for their permission and were given a very easy and discreet option to back off from the interview. Care was taken not to build any assumptions or expectations about the interview considering the topic of the study. The respondents were given general description of the interview; that it considered IT deployment projects and their management. They were also told that they were selected based on their perceived experience on the subjects covered in the study. Even the title of the research was "kept secret" to ensure minimal bias on the answers. The respondents were told that the study does not compromise any laws and is kept strictly confidential and they are free to end the interview at any point, if they felt it being inappropriate. All the interviewees accepted the interview. Two of the project manager interviews needed to be conducted in two parts due to time constraints. The second round (missing themes and questions not covered in the initial interview) was conducted via phone calls. As the expectations for the interviews were already established and a good mode of operation was created, it was very easy to finish and conclude the second round. The phone calls were recorded as were the initial interviews.

The ambience for the interviews was selected as open and informal. The interviews were conducted in the respondents' work place, homes, a restaurant, in the interviewer's home and generally in places, which would not pose any feeling of a too formal ambience to keep the conversations easy and fluent. Seating was organized in a way that both the interview and interviewee could see each other's face and expressions well, which is important for the feedback. Although the interviewees were told that their brutal honesty and as detailed and correct as possible answers are expected and valued, the interviews were still carried out in a very friendly and open atmosphere and in a very open-minded mindset. The interview meetings were started with casual chatting about "this and that" to take the edge off and to reduce any possible excitement of the interview situation itself. Average interview duration was 1 hour, 19

minutes, and 50 seconds. Details on duration(s) as well as interviewee profiles can be seen in APPENDIX II - Profiles of the interviewees'.

To aid the verbal interview, author created a slide deck, which had elaborations of the questions in forms of figures and pictures and also definitions of some terms, which were synthesized during the theoretical study. The slide deck is attached as APPENDIX I - Interview slides.

The author was left with impression that all the interviews were held in a open an honest spirit. It was also perceived that the interviewees did their best to be as precise as possible with their answers to provide maximum feedback. They also provided a lot of opinions and elaborations on their answers, which can be perceived as a sign that they were interested in the interview and thus probably provided their best possible answers. They showed their expertise with criticism and by enhancing the presented themes.

6.3 Reliability of the selected interview approach

Apart from the time consuming nature of interview approach, it is commonly recognized to have the shortcoming that the interview itself has the risk of possessing several error sources. These errors or faults can derive from the interviewer or the interviewees. As an example, the reliability of the interview can be compromised by the interviewees tendency to provide socially "correct" answers and please the interviewer. (Hirsjärvi & Hurme, 2008).

Another possible reliability factor is the selection of interviewees. It is possible, although it has been consciously avoided by the author, that the author has selected such interviewees, which he recognizes as respondents favorable to his opinions and views and thus produce research data, which is favorable to the researcher.

The third possible shortcoming is the number of interviewees. How can the author say that an adequate number of people have been interviewed? The answer is: "He can't". There is no scientific nor commonly agreed minimum or maximum number of interviewees a theme interview should have (Hirsjärvi & Hurme, 2008). As the interviewees all are unique in nature, there is always a clear possibility that saturation of answers (i.e. answers from further interviewees would not probably produce new data) has not been reached, although the author is quite confident that the key findings are already presented in the answers of the interviewed people. Of the five interviewees all shared opinions on the critical research questions. The author deems this as a relevant finding in the study. A fact, which would not be altered by 10+ more interviews, even if the 10+ next interviews would produce different results. The findings still remain significant. The author concludes that the accuracy for the purposes of this research is adequate and the data gathering committed is of good scientific rigor when considering the availability of the author's resources. Still, the possibility for undiscovered, fruitful data is recognized as a possible shortcoming.

The fourth possible hindrance for the reliability is the experience of the interviewer with interviewed persons. Although the author has executed several interviews (formal and informal) during his professional career, he is not a professional interviewer. Hence it is possible that the author has missed one or several new aspects to the researched topic when interviewing the respondents. One possible outcome of this would be too early and incorrect notion of reaching saturation and not interviewing further interviewees.

Lastly, the interviewees and the interviewer were acquainted to each other, which is not commonly favored (neither forbidden) in interview guidelines (Hirsjärvi & Hurme, 2008). For the purposes of this research, as it compromises between several areas of expertise and requires acknowledged high experience

in deployment projects, the possibility, within the resources of the author, was to identify the interviewees from his personal network to ensure an adequate level of expertise and know-how from the respondents. The author does seethe positive side of this: he has a long track record with the respondents, he has perceived their experience, success and doings both by self-observation but also via organizations´ colleagues' shared perceptions of the respondents. Also knowing the respondents well in person gives the author a good insight into whether the responses of the interviewees are consistent with their history (as the author knows much of their backgrounds) but also, if the respondents are providing the interviewer the truth and being honest (not automatic, but as the author has a social background with the interviewees, he claims that he is better judging if the respondents are twisting the truth than an interviewer not knowing them in person that much).

6.4   Summary on research approach

The selected empirical research approach was presented and justification for the selection elaborated. Interview and especially theme interview was seen as the research method to collect the primary data from the presented respondents. The main reasons for the interview was that the concepts investigated are very complex in nature and the possibility for misunderstanding without effective dialog is high. Also, it was needed to get elaborations and discussion on the themes from the respondents, not statistical data.

7 INTERVIEW STUDY ANALYSIS

This Chapter will continue from Chapter 6, where the settings and background of the research approach and interviewees were elaborated. The analysis of the primary data collection (interviews) is presented with some of the themes and questions already reduction processed and initial findings provided (interpreted via meaning).

The data collected is thematized and coded and this thematization and classification of data is first explained. Later the identified themes are covered one by one, each in its own subsection with relevant coded entities attached to the theme.

At the conclusion Section of each theme the author will provide his personal experience on the theme. This is to reflect the findings to his own extensive project management knowledge and will further enhance the reliability of this research. The author will also, when needed, ground the findings back to the theoretical part of the thesis, i.e. the literature study of Chapters 2 through 5. A further concluding analysis is provided in Chapter 8: Conclusions, limitations and future research.

7.1   Analysis approach

The initial analysis of the data collected has already taken place at the interviews themselves. The interviewer made remarks of relevant entries and summarized some of his interpretations to the interviewees, who then commented, whether the summary and interpretation was what they meant or corrected the interpretation. The approach could be classified as "self corrective interview" (Hirsjärvi & Hurme, 2008).

Apart from the initial analysis started during the interviews, the analysis was executed after the collected primary data was decomposed. The author proceeded immediately after the primary data decomposition to the analysis of it, trusting his intuition. Not all the primary data was transcribed (circa 25% was), but mostly the classifications and interpretations were made from the recordings directly and analyzed and reported.

The reasoning practiced during the analysis phase can be considered as inductive reasoning. A considerable amount of interpretation of meaning has been applied and is always speculative in nature. This being especially the case, as the author had a personal view to the research subject and he is interpreting the data from this personal angle (Hirsjärvi & Hurme, 2008).

7.2 Identified themes, classification and coding of data

Although the approach utilized is that of a semi-structured theme interview, the author had quite a clear classification, or sub-themes, identified in his interview frame. The findings are classified under these identified sub-themes in the analysis. The used classification/sub-themes can be summarized as follows and they follow the general structure of the research's literature structure:

- Experience of the respondent in IT and respondents' project history
- Scope management in IT deployment projects
- Human/organizational change management in IT deployment projects
- Deployment project success criteria and curve of uncertainty
- Scope management vs. change management – which one is more relevant

A certain order of presenting the sub-themes is also applied. In the interviews the general idea was to first get the respondents own free "definition" of each item (e.g. "scope management", "change management") and then cross check

those against the synthesis the author had provided (APPENDIX I - Interview slides). Then, only after this alignment, the most relevant questions concerning this research were posed. The order of themes getting discussed in the interviews corresponds to the order of the theme Sections following.

Data coding varies from a sub-theme to another and is question/area specific. The used coding is elaborated with each usage. Generally, no numeric scale is used, but if the importance of phenomena is investigated, for example, then comparison is done via comparable terms (e.g. "not important, somewhat important, important, and very important").

One classification is to be mentioned first, because it applies to all the themes presented later. That is the deployment's role as either a standalone project or a phase in a project execution as it was introduced in Section 2.3. Three respondent interviews included such a question, where the answer to this classification can be coded, i.e. if they felt that the interview items applied to both of the cases (phase or project). Classification for this is **appliance in deployment project and deployment as a project phase** and it was coded with *yes/no* – further respondents elaborated with open comments. Enquiring this was done at the very closing of the interview. Analysis is that all the interview themes and their findings apply to both deployment project and deployment as a phase of a project. 3/3 respondents responded *yes*. Although the findings apply to both cases, there are still differences identified, where focus and importance of project activities varies. PMs' comments elaborate this – starting with PM1:

> *In principle yes, although some flavor twists do exist, e.g. if we're talking about a small maintenance project phase or a massive new system deployment* [as a project]. *But still the same fundamentals apply and must be taken care of e.g. "this sort of thing is coming" and "this is the meaning of the change" etc.*

PM2 identifies that deployment as a project tends to do things in more detail and for better outcome:

> *Deployment as a phase of project where there is create too, does not always "walk the last mile". But when we're having deployment as a project more details are addressed and the last mile also taken care of better and things are done in more finished manner […] communication, training are in shape and in overall more than "just the minimum" is done.*

Lastly, PM3 underlines that activities done as a separate project take better care of stakeholder management and facilitate better communication and dialogue between provider and customer organizations:

> *Yes, apply to both, whether it is a phase or a project. I figure it is better to keep it as a separate project, 'cause then it* [deployment] *will work as a link between the receiving organization and create activities … so that it* [deployment] *understands to "translate" the message to both directions.*

7.3 Theme: experience of the respondent in IT and respondents' project history

To understand the answers later provided by the respondents, it was important to first get background information on the respondent. It was also important to validate the respondents' background as *subject matter experts* on the researched topic. From the theme several classifications were produced:

- IT experience (divided further to general IT experience and experience as an IT project or program manager);
- Project work history (overall project work history and history from perspective of being the manager for deployment projects/programs); and
- Sizes of the deployment projects the interviewees had experienced.

7.3.1 IT experience

All the respondents registered at least 10 years of **IT experience**. This means that they had been involved in work assignments where information technology was a central part of their work profiles and project work had been a common mode for executing changes. Whether the experience was acquired as managing role, technology specialist, trainer etc. was not investigated as being not relevant for the research. Neither are the company profiles investigated, although it can be mentioned that 3/5 stated that they had worked in small, medium and large companies, where 2/5 had their experience from a large company. All respondents registered global and international work experience.

The maximum IT experience was 15 years, with average being 12 years. **Experience as an IT project- or program manager** was averaged as 5.6 years with the most experienced interviewee counting 10 years and the least experienced informed 2 years of project manager experience. A note should be made that the latter mentioned experience is only counting, if the respondent was officially responsible with the role of project or program manager. Several notions were made by the respondents that they had actually been (in practice) managing the projects, with person NN being official, but not the effective (operational) project manager (in which case the years/projects did not count).

7.3.2 Project work history

It was relevant to investigate the respondents' project work history to further validate the interviewees fit for the research. This was enquired by first investigating the number of project or programs they had been involved in with some role – this was coded as **overall project work history**. High value of 140 and low value of 10 was recorded with the average being 42 projects or programs. Ignoring low and high value, average n was 20. Of these projects or

programs average of 38, low of 8 and high of 140 had been deployment projects or deployment programs. High and low ignored average scored 14 projects or programs. These deployment projects or programs were further coded into being parts of an IT program. An average of 11 projects was counted, low 5, high 18 and excluding low and high did not impact the average significantly (11,5).

Experience as a **manager for deployment projects/programs** (subset of total project work history) was similarly investigated with low of 1, high of 15 and the average of 7. With the high and low values being ignored, the average scored 6.

7.3.3   Sizes of deployment projects/programs

To understand better the nature of the deployment projects and programs, some sizing factors were enquired from the respondents. They were asked for the (estimated) average sizes of the deployment projects or programs they had been involved with three coded factors: **number of personnel in the project/program** (TABLE 4, "n P"), **number of the stakeholders** (TABLE 4, "n SH") involved in the project/program and **duration** (months) **of the project/program** (TABLE 4, "n D"). The respondents were asked to provide averages they could pinpoint to ranges of values provided.

Common personnel count in deployment projects seem to fall into the category of 10-50. It was commented to be closer to 50 than 10. The analysis is summarized in TABLE 4.

TABLE 4. Number of personnel in the project/program

Respondent	n P 1-10	n P 10-50	n P 50-100	n P >100
PM1			X	
PM2	X			
PM3		X		
PM4		X		
PM5		X		

The number of stakeholders (anyone affected by the beneficial change) (TABLE 5) seem to be rather high with 1000-5000 being trade-off from the interviews. The respondents' best estimate average was asked, but they registered in their own history many endeavors with larger stakeholder figures. PM5 commented:

> *Over 10 000 has been many, 5-10000 has been many, 1000-5000 has been few [...] then smaller than that... well, the current one is in the first category. So hits to all categories but 50-100 that does not score.*

TABLE 5. Average number of stakeholders in the projects or programs

Respondent	n SH 1-50	n SH 50-100	n SH 100-500	n SH 500-1000	n SH 1000-5000	n SH 5000-10000	n SH >10000
PM1					X		
PM2			X				
PM3				X			
PM4					X		
PM5						X	

Deployment projects can be considered rather long in duration with common category of 6-12 months and even there, the bias is more on the 12 months than on 6 months. This is shown in TABLE 6.

TABLE 6. Average duration (months) of the project or program

Respondent	n D 1-3	N D 3-6	n D 6-12	n D 12-18	n D 18-24	n D 24-30	n D 30-36	n D >36
PM1				X				
PM2		X						
PM3			X					
PM4			X					
PM5			X					

Longer deployment projects than 12 months seem to take place quite often. PM5 elaborated this with:

> *One over 36 ... then 12-18 can be found, 6-12, 3-6 and average the cross is in 6-12.*

This was further backed up by PM3 commenting on the durations:

> *From the ones mentioned smallest ones have been 3-6months and the biggest ones have been … largest one has been – how many months there are in five years? Commonly they have been around one year.*

7.3.4   Summary of theme's findings

The target of the theme was to investigate, whether the respondents' experience and background with deployment projects can be considered adequate for the purposes of the research. The author is convinced that the respondents have solid experience for providing subject matter expert opinions and views to the researched subject.

Another target was to investigate the nature of the deployment projects and programs. Judging from the large figures of deployment activities as projects the justification of projectizing deployment activities mentioned in Subsection

2.3.2.1 seems to be backed up by the figures here. It is also shown that deployment projects are quite often part of IT programs; 11 out of 38 and ignoring the low and high values 11 out of 14 deployment projects were part of a complex change program (see Section 2.2).

It was found out that deployment projects or programs involve a rather large number of stakeholders (common 500-1000) and that the projects take rather long to execute (6-12 months, closer to one year). Judging from these figures it seems that deployment projects are established for deployment activities, which involve large number of stakeholders, take rather long to finish, and involve some complex IT endeavors and, where deployment is part of the program execution. For small deployment activities, there seem to be seldom need for separate projects, but deployment activities are rather executed as part of a project creating the beneficial change and the IT artifacts.

Findings correspond to the empirical experience of the author quite well. With all the major deployment projects the author has witnessed the duration of at least six months with a considerable number of stakeholders, counted rather in thousands than hundreds. For deployments, where there have been only dozens or hundreds of stakeholders, the deployment has usually been executed as part of the project creating the technical IT artifacts.

7.4   Theme: scope management in IT deployment projects

The foundation of this theme was to build a "basis" for the main research question with investigating that the respondents have equal understanding of scope management as this research. Their definitions were asked and the relevance of scope management was enquired. Later respondents were asked, if they use any recognizable body of knowledge, methodology or framework for scope management in their daily work. Lastly, the respondents were to tell, if scope management is recognized on the needed level in their work

environment (colleagues etc.) and if scope management has a different role in create-style projects. Following classification is used:

- Scope management in IT deployment projects
- Relevance of scope management in deployment projects
- Usage of recognizable models and tools for scope management
- Acknowledgement of scope management in working environment
- Role of scope management in create projects

7.4.1 Scope management in IT deployment projects

Before showing the respondents any clue what the interviewer had defined for scope management in IT deployment projects, their own opinion was asked. The author was satisfied with the answers, because they were tuned in enough detail with the authors definition of scope management that it was reasonable to proceed with the interviews. Common answers for scope management were identified through the activities relevant to scope management: defining the end user population, needed roles, impacted business processes, geographical scope (local, global), impacted organizations, what the change is, and drawing the line between what is in and what is out. PM2 defined scope management in deployment projects as:

> *Defining precisely what is done and what is out, if one thinks that in deployment;*
> *do all worlds end users belong to the scope, when thinking of a global organization*
> *– is it then all the possible users, who might eventually use the system, or the key*
> *users of the organizations' [...] in what roles these people are, in what point in time*
> *the responsibility is switched to the use –phase.*

PM1 also highlighted the user group and geographical aspect, but also brought in the business processes and that scope management should consider the handover phase also in a sense that it ensures requirements towards use-phase are considered and implemented:

In practice it is definition of what the user group is, then of course the geographical needs, if it is targeted globally or to only certain reason and naturally what are the most relevant business processes it should [(project)] fulfill. [...] Also [identifying] *the requirements towards service management and ensuring that they are taken into use.*

In general, all the respondents recognized the most relevant characteristic as classification: **what is getting done and what is not** with simple binary *0/1* - coding with 5/5 of respondents' definitions coded as *1* (meaning they complied to the characteristic).

7.4.2   Relevance of scope management in deployment projects

To investigate whether the notions made from secondary data in Section 5.3 about the relevance of scope management do generally apply to deployment projects, it was asked, how did the respondents see scope management's relevance in deployment projects. Although an open ended question, classification of **scope management relevance** is investigated with range of *not important, of low importance, somewhat important, important, very important* and it can be translated to the Likert (1..5) scale. The most frequently selected value was *very important*, with PM1: "important", PM2: "important", PM3: "very important", PM4: "very important" & PM5: "very important".

PM1 highlights the importance of agreed and written scope:

It is quite relevant.. because usually at that point we start to go the wrong direction, if it is not done. And usually, in quite many occasions, yellers appear and if we have nothing agreed in written, then the whole thing will get out of control.

PM3 recognized that there is a clear connection between scope and change management and commented that if scope management fails, then change management will get difficult:

I see that it is very important, because if we do not exactly do know what we are deploying, it then makes change management very challenging. And if you do not know your change management target, then it is very difficult get the buy-in from

the management. So when you'll have your 15 minutes of fame to get that project or change sold, it [scope] *then has to be very clear.*

PM5 commented that quite often scope management is not formally executed in deployment projects and loose scope management will lead later to unclarity of what the project was supposed to do:

What is expected to happen is defined via scope management – directly. And if it is not explicitly within scope management then it will be implicitly there and later on there will be like "how this is like this" and why this is not like that, or were you not supposed to handle this too –like cases. It is quite common for deployment projects that it is not done formally and it will implicitly within the project's other many domains.

7.4.3 Usage of recognizable models and tools for scope management

Once the relevance of scope management was investigated, it was interesting to know whether the project managers used any industry or academic recognized bodies of knowledge, frameworks or methodologies in their daily work for scope management (e.g. the PMI's process area of scope management (Section 5.1)). A classification of **tool and model usage for scope management** is presented with *yes/no* options and in case of *yes* a coding for usage frequency is provided with *very seldom, now and then, 50%yes-50%no, most of the time & systematically*. In this thesis' context, a *model* can be perceived as for example a process (can include a tool) from a body of knowledge or a framework, whereas a *tool* can perceived as at its simplest a checklist, which is used more than once.

PM4 commented on the formal tool usage for scope management:

It [scope management]*, has been more a gut feeling, no structured way of doing it.*

The above comment summarizes most of the interviewees' answers. One (PM5) recognized that his working practices would be based on the PMI's PMBoK, but even so, practices were very much changed and adapted to the PM's personal needs and were not formal in that sense. For the frequency of usage he

commented that it is *systematic* in nature. PM1 identified that his scope management practices are adopted from agile and he's using them *most of the time*. 3 out of 5 PMs did not recognize using any tool for their scope management.

This finding reflects a finding that the organizations the respondents have been working have not been able to provide such a tool or model for efficient scope management that the respondents would have embraced. The same analysis is reflected in Subsection 7.4.4, where it is commented that there is no formalized, company-wide, practice (model, tool) for managing project scope. The finding is quite alarming and can on its own behalf also explain some of the challenges the respondents have encountered in project scope's liability.

Although no formal or best practice tools were used by the PMs, the most still recognized that if they lead an external customer project and the customer had some company internal project working practice, they tried to adapt to that and do for example checklists and reviews required by the company working practice for scope management. PM2 identified this as:

> *Mostly like the customer's project practices require. I use nothing formal, usually I do like some 3th degree cross interrogations: "Is this all, are you sure this is all". […] And if customer then has some review system in use or document template, those then pretty much already from the topics tell what needs to be done and recognized.*

7.4.4 Acknowledgement of scope management in working environment

As the interviewees' opinion on scope management and its importance was investigated, it was interesting to get their opinion, if their view is shared in their working communities. This was done by asking whether the respondents felt scope management was recognized in their working communities with enough relevance. This classifies as **acknowledgement of scope management in working environment** and is coded with *highly under-acknowledged, under-*

acknowledged, on needed level, over-acknowledged, highly over-acknowledged. If the answer would prove out to be either of the under-acknowledged, then it was asked, what would be the outcome, if scope management's relevance and importance would be more promoted in the working community.

Of the answers (open ended) 3/5 belonged to category *under-acknowledged* and 2/5 were *highly under-acknowledged*. PM5 commented on the *under-acknowledged*:

> *Scope management is not done as formally as it perhaps should be done. It is not understood enough.*

PM4 summarized:

> *Clear problem, awareness should be raised. It would lead to better planning.*

On the *highly under-acknowledged* PM2 said:

> *Should be promoted, definitely. It would mean that there would be less surprises and it would reduce the fine tuning at the project end and I believe that it'd improve customer satisfaction – that we'd know straight from the start that what we are getting.*

The items, which promoting of scope management would bring in the organization listed among: customer commitment to project (PM1), on time delivery (PM1 & PM5) better planning (PM5, PM2 & PM3) and more focus (PM3).

7.4.5   Role of scope management in create projects

For better understanding the success criteria commonly promoted with successful scope management for project success, the author wanted to enquire, whether the respondents felt that the role of scope management would be similar in create-projects (creating the beneficial change, like software or a process) as in deployment. Classification is **Role of scope management in**

create projects and it is coded with *yes/no* and if *no* was answered, then the difference was further asked to be elaborated. 2/5 coded as *yes* and 2/5 as *no* – one answer cannot be coded to either category.

PM2 elaborated the difference of scope management in create-type projects as:

> *A create project from the scratch, it must be more in detail defined what is to be done and when one thinks about a deployment project either deploying an off-the-shelf product or a product output from create project.. there is less fine tuning in deployment.* [...] *In create, scope has management has a more important role.*

On the other hand, a comment from PM5 on the similar role of scope management, whether it be create or deployment project:

> *Well, the principal question or thing is still the same that the impacts will be similar the more in detail it* [scope management] *is done – topics perhaps are different.*

An additional analysis of many comments showed that it was promoted in several comments that create-project and deployment-project must share the scope and understand each other's scope definitions, otherwise radical "what is to be done" mismatch would be evident.

7.4.6   Summary of theme

It was found out that all the respondents shared compatible understanding of what scope management is about and could put it to context of deployment projects. Furthermore, it was shown that scope management was seen as very important for deployment projects. Although the interviewed identified the importance, they felt that their working communities did not recognize it in an adequate level and some education and awareness raising should be done. Contrary to the explained importance of scope management, only two of the project managers used any models or tools (process model, framework, BoK, ...)

for their scope management work. Again, in one comment it was said that promoting formal scope management would improve the organizations' understanding of scope management and its relevance. It is found out that even in the very large IT organizations where the respondents have worked, there has been no recognized (or the tools and models have been of not adequate benefit) and embraced scope management models and tools and thus some of the respondents have adopted their own models and tools. The author thinks that here is a good change for "leading by example" for the interviewees and promote the need for models and tools.. Lastly, yes/no's were even on the nature and relevance of scope management, whether it is different in create-style projects than in deployments.

On the personal record of the author the findings are compatible, there is no contradiction between the findings of the interviews. As the author had recognized in his professional years the significance and scope management, whether in create or deployment projects, the theme was raised as a key theme in the interview. The author recognizes also that the use of formalized tools for scope management in deployments is more uncommon than common. The author's own scope management exercises are loosely based on the PMI's scope management knowledge area with much simplification and adjustments from working experience. In his projects, the author has always put the needed effort into forming the scope statements for the projects (or programs) and communicating those as the main message alongside the project/program's mission and target. On the personal record, the author feels scope management being a bit more important process area in create projects than it is in deployments.

Ignoring the relevance of scope management has lead to several dramatic over times in the author's record of projects and programs he has witnessed. This has happened both in create and deploy projects, but the effect has been more

dramatic in create projects. Usually, when deployment projects are struggling with their scope, they are given more time and effort to correct the situation than create projects, where the decision to terminate the "work with no direction" is more hastily done. Why this is, is most probably due to a fact that in deployments there is often already significant investment made either to a create project creating the IT artifacts or some IT artifacts being sourced from an outside vendor or provider. Deployments are then given more chances to correct their direction and even some heavy non-planned extensions to projects cost and time are allowed to see the IT artifacts getting deployed.

7.5 Theme: human/organizational change management in IT deployment projects

Similarly to Section 7.4, this theme also aims to validating that the respondents are aligned enough on the concept of human/organizational change management in the context of IT projects. Getting this understanding is crucial for the primary research question and this fact is reflected in the later theme: "Scope management vs. change management – which is more relevant". The author tried his best to interview both scope and change management as equal conceptual models and thus 1:1 questioning model is used as with scope management. Hence the classifications are:

- Change management in IT deployment projects
- Relevance of change management in deployment projects
- Usage of recognizable models and tools for change management
- Acknowledgement of change management in working environment
- Role of change management in create projects

The interviewees were asked for their definitions of change management, usage of tools and models for formal change management, working environments

recognition of change management's relevance and lastly, if change management plays a different role in create-style projects than in deployments.

7.5.1 Change management in IT deployment projects

As with scope management, before showing the interviewer's definition on change management to the respondents, the interviewees' opinions and definitions were asked. It turned out that the conceptual models of the interviewees were very similar in nature and also quite compatible with the author's model. Hence the interview could proceed with all the respondents.

The relevant classification was that of **identifying the change and its impacts and making the stakeholders buy the change** and that was positive (1) finding with 0/1 coding for all the respondents. Common entities (or activities) mentioned at least three times in definitions were: *communication, showing meaning, making people buy the change and getting acceptance for the change, countering the change resistance.*

PM2 saw the change management as the way from the current to the future state and identifying the changes taking place and selling the changes:

> *It is about understanding the current state, what it should be, how do we get to future state from the current state, be it like a system change, there are changes in processes, changes in people, changes to roles* [...] *Getting people to buy the new thing.*

PM4 elaborated that:

> *It is so traditional, when starting a deployment project that the change resistance is considerable. All people are by default against the change.* [...] *"we do not want" and at that point one can start talking about deployment change management, to get the people to believe that the change is good.* [...] *Convincing that one should hop along to this change.*

PM5 noted that the holistic change is the sum of personal changes that people are feeling. Change management is also about helping people to advance in their own change curve:

> *Leading and managing the change people are experiencing. That change is the sum of changes individuals are experiencing and that is the holistic change and change management is controlling it. [...] communication, making the change visible, educating people so that they can advance in their own change curve, where there are different phases [...] Training of course, and making people to participate, meaning that people, who are affected by then change are made to participate for example to scope management, so that right items are in the scope.*

7.5.2 Relevance of change management in deployment projects

To reflect against the secondary data findings of change management's relevance (Section 4.2) to deployment projects, the interviewees were asked to elaborate their view on the relevance and importance. Classification of **change management relevance** was investigated with scale of *not important, of low importance, somewhat important, important, very important*. The answers were coded to these values and they ended up as PM1 (very important), PM2 (very important), PM3 (important), PM4 (important) and PM5 (very important). 4/5 of the project managers commented on the relevance that if change management is not done adequately, then the whole change can be for nothing and not taken into use. It can be classified as **inadequate change management can fail the project** (coding yes/no) although not explicitly asked, the notion is still very relevant – even without one PM making an ad-hoc comment. Overall, 2/5 coded change management as *important* and 3/5 as *very important*.

PM2 made it quite clear-cut-here in her comment about the relevance of change management:

It means that if it [change management] *is not done, then what ever one is doing will never get deployed to use. That it does not matter how good the system is, if people are not starting to use it.*[...] *getting people to buy the new thing.*

PM1 found out that change management is also much relationship management and selling of change:

> *Well, in my mind it has very important relevance, because if you do change management wrong, then the whole change is for nothing. 'Cause if customers do not understand why this is done, they don't for certain take it into use or if one does the change to them by ****** [cursing] to them and as by stating things, then the outcome will be most probably the same. So, in a way you are a salesman there and if you do not take care of that customer, then most probably it will not buy anything from you.*

PM5 wanted to underline the relevance of change management over scope management, although he coded both "very important":

> *Very, very important, in the worst case, if change management is not done right, then the whole deployment will fall there. None of the defined targets* [scope] *are met. So, the mistakes made in scope management can at their worst mean that some things are not getting done or people are not totally happy, but in my mind this* [change management] *is even more important.*

7.5.3 Usage of recognizable models and tools for change management

As the relevance was investigated, the usage of any recognizable models used as tools for change management were enquired. The respondents were asked, if they use any methodologies, frameworks or bodies of knowledge to guide them through the challenges of change management. Classification for this is **tool and model usage for change management** and is coded with *yes/no* and if *yes*, then further coded to tool usage frequency of: with *very seldom, now and then,50%yes-50%no, most of the time & systematically*. One of the interviewees was not asked this in such a way that the answer could be coded to yes/no, but out from the rest four, two recognized using a tool and two did not.

On the usage of change management tools and models PM5 commented:

> *Yes, from one training – even in the training it was not called a model though, but that change management is more like a craft. That it is not science , but a craft like one of blacksmith's That learning I have implemented. [...] I use it systematically.*

PM2 commented on lack of any "formal" change management tools that it is like a hipshot and an intangible thing:

> *No, it is more of a hipshot. Of course there are some best practices, like that I have trained a lot and training plans being much change management – something come from there, but more I feel like that it is a spiritual thing.*

On the frequency of usage PM5 coded *systematically* and PM1 as *most of the time*. Although "formal" change management tools are missing from many PMs they still recognize that they have adapted some best practices, which one PM called "hidden knowledge". Similar to finding from scope management tool and model usage, there seems to be a major gap in availability and usage of formal and efficient tools and models for change management in the organizations the respondents have worked with.

7.5.4 Acknowledgement of change management in working environment

The interviewees were asked how well they felt that their work community had recognized change management and if there was room for improvement, then what would be the outcome of that improvement. This classifies as **acknowledgement of change management in working environment** and is coded with *highly under-acknowledged, under-acknowledged, on needed level, over-acknowledged, highly over-acknowledged*. Should the answer be either of the *under-acknowledged* ones, a question would follow, what would be the outcome, if change management's relevance and importance would be more promoted in the working community.

All the five project managers' comments can be coded to *under-acknowledged* value. Some examples on how the project managers felt that raising the understanding and knowledge of relevance in their work communities could affect the organization, are provided here. PM3 starting:

> *Not understood enough, and commenting with a deep breast voice of experience, it is not about that you have a perfect excel in your hands, where you have all, all, the names in the world and those classified right. The excel will die, before it is ready. It is more about living change management, from one person to another.* [...] *And it* [change management activities] *must be* [executed] *often enough, otherwise people will forget.*

The elaboration on the comment needs to be made, as it concerns the tools usage covered in Subsection 7.5.3. Analysis is that there are tools available, but they are of such low usability or relevance that they are not used as such. As commented by PM3, it is not about the "perfect tools" either, but tools, which are usable and efficient. As with comments from Scope management-theme's tool usage (Subsection 7.4.3), it seems that the tools available are often modified by the experts for their own purposes and ease of use.

PM4 continues with possible positive organizational effects, if change management would be understood and done right, people would attend training:

> *Well, let's take an example that if the change management is not done right and then we are giving these training sessions, which are about this change... people will not attend them. So, in my mind they have not understood what the change means, otherwise they should be attending. And later, after the change, they are crying after the training.*

One last comment on the positive effects of raising the awareness of change managements relevance from PM1:

> *Under acknowledged* [...] *It would mean less change resistance, so that the customers would really know when this* [change] *is coming and more important, why it is coming* [...] *I still see the motivation factor as so very important.*

7.5.5   Role of change management in create projects

There is very little academic and practioner literature on how relevant change management is for project and even less information on change management and deployment projects. Hence it was relevant to understand, if change management is similar in create-projects as in deploy projects and could the few literature entries be applied commonly to both project types or are the project types so different that generalization is a clear error. A classification of **Role of change management in create projects** was created, and it is coded with *is similar to deployment / is not similar to deployment* and if *is not similar to deployment* was answered, then the difference was further asked to be elaborated.

For two PMs there was no such answer which could be coded, as the theme interviews were only semi-structured – they did not cover all the specifics in each interview the same way. Although 3 out of 5 interviewees' opinions can only be coded to this classification, the author still deems the analysis as of relevance. 3 out of 3 commented that the role of change management is not similar in create and deployment projects. Two short comments quoted partly here – PM4 starts with:

They are not similar. In create projects change management's role is totally different. People as stakeholders in create projects usually do what they are required to do, they are not that needed to get "convinced" to hop along.

And PM5 summarizes with a short:

Yes, yes, is different absolutely. It [change management] *is altogether differently highlighted in deployment. In create I see that it is more about scope management, which I see as part of executing efficient change management.*

 The comment is analyzed that PM5 sees a strong dependency of scope and change management for either, especially change management, to be efficient and even possible. This "duality" he perceives is more emphasized in scope

management in create projects, where the role of change management is secondary and the other way around in deployment projects.

7.5.6 Summary of theme

The analysis convinced the author that respondents did have a very similar conceptual model about change management in deployment projects. This model was similar to the author's so all the interviews proved out to be relevant for the main research question. Likewise scope management, change management was seen as very important for deployment projects. All the five project managers commented that there was room for improvement in their working communities for raising the awareness and importance of change management and that their organizations could benefit significantly from this change in working practices. Like in scope management, the usage of tools for change management was on a rather low level. Again, this underlines the fact that even in large organizations the respondents have been working in, recognized tools and models do not exist or they are of such a low benefit that they are not used. As the availability of models and tools is rather good, the author recognizes this (just like in scope management) as a major shortcoming in ways of working of the represented organizations. As one respondent commented, one of the main properties of such models and tools is the simplicity of use. The analysis would thus also suggest that much effort should be put to test the tools and models in project management everyday work and make them efficient there, not just on paper and in theory.

Of the answers, which could be coded, all stated that the role of change management is different from the one in create projects. Hence, directly applying any critical success factors or indicators, which consider scope, time cost (especially scope) to deployment projects, which emphasize change management can be hazardous, as seen in Sections 7.6 and 7.7.

As with scope management the findings from the interviews are rather confirming the author's personal hypothesis on the topic than exploring something dramatically different. This can be perceived as confirmation to the authors red thread of this thesis and confirm that the research setting has been rightfully set up for the upcoming findings.

In the author's working environments, the relevance of organizational change management is very much under-acknowledged and significant training and best practices should be applied. The author will elaborate this need and possible approach in Chapter 8: Conclusions, limitations and future research approach. The author has witnessed, although not lead, several large scale program and projects being cancelled as they have not invested enough to organizational change management. Their scope definition and organization of work has been "ok", but the selling of the change and IT artifacts has not been properly executed and the organizations supposed to use the artifacts have refused to do so and stuck to their old working habits.

If organizational change management differs in create and deployment projects the author's personal judgment is that yes, they do and dramatically so. It does not mean that the organizational change management would be indifferent in create projects, it definitely is important. One could perceive that organizational change management is, for example, about the activities to get organizations to promise to invest money and time to upcoming create activities for creation of the IT artifacts. Still, the organizational change management is, in the author's opinion, of lesser relevance in create projects and it does not play such a significant role as in scope management. This statement is a generalization and it considers "most of the times" view. Exceptions naturally do exist and that must be considered. The author considers the mentioned as a rule of the thumb and applies at least Pareto 80:20 in his personal experience (i.e. in at least 4 out of 5 create projects scope management has been of more relevance than change

management.) Hence, if the research question of this thesis *would have been* from create project's point of view, the author would have personally voted for scope management being more important than organizational change management. This personal opinion was especially the reason, why the items of scope and change management had been introduced in this thesis, both in the theory and empirical part.

7.6 Theme: deployment project success criteria and curve of uncertainty

Whichever way around the main research question would turn out later in the research, it would still be of importance to understand the criteria on which deployment projects are judged. This would probably help to synthesize the big picture later on, because the identified success criteria and whether product or project success is more important, all help to understand the roles of scope and change management in deployment projects.

The following main classifications were identified and will be covered next:

- Product success or project success
- Five most important success criteria
- Project failures and curve of uncertainty

7.6.1 Product success or project success

The findings from Section 3.3 were summarized for the interviewees and they were asked which one is more important for them: product or project success. This is classified as **PM product/project success choice** and is pair comparison of importance of the mentioned two success possibilities. The interviewees were not posed with black-and-white, "if you have to choose one, which one would you choose"-approach, but rather questioned, which one they aim themselves

to achieve. Further elaborations on why the picked success possibility was picked and was considered more important were asked and presented here.

In the same context it was also enquired which one, project or product success, was more important for the organization. This was further divided into part of the organization, which is using it for business purposes and classified as **business success criteria** and coded as *product ,project* which tells whether business organization values product or project success more. A similar comparison was made for IT organization (deploying the change) and classified as **IT success criteria.** Also, an elaboration, if cost/time/scope slippages are commonly accepted, if the change otherwise is deployed and taken into active use, was asked. This was classified as **cost/time/scope slippages allowance** and coded as yes/no (if they are accepted or not) and some elaborations were asked on the choice.

On the PM's personal product or project success choice coding provided 5 out of 5 "product". So, although being project managers and many times judged based on the classic time/cost/scope criteria, all the project managers interviewed still promote that they aim for the product success, and project success is secondary. Some elaborations starting with PM2:

> *Sometimes I have felt honestly bad, being part of some project, in whatever role, and if then nobody uses it* [change] *then it feels a bit like: there goes my time wasted.*

PM3 feels that it does not matter whether the project has been completed successfully in project success sense. It is still a "fail", if nobody uses the system on the intended date, whereas it does not matter if the date slips and people start to use it:

> *In deployment, it is definitely that it is getting used. It's in my opinion a failed deployment, if project-wise all has been completed successfully and all criteria*

fulfilled, but nobody uses the system or nobody understand to use it, then it is a fail.

PM4 further elaborates on possible project scope/schedule/cost slippages:

Product success [is more important]. *In project some of the areas [scope, time, cost] can fail but product success is reached, on the other hand it does not really matter in the end, if the project only has been a success. [...] For example, when I was a project manager, I often rebelled considerably against for example of given dates that the project need to be ready. I don't feel like things like that having that much importance than what the outcome* [product] *then is. Of course there must be some deadline, otherwise project ends up an eternal project, but even then I considered the product success more important.*

On the **business success criteria** all the respondents answered with an answer, which can be coded as *product*, i.e. all the project managers feel that product success is more important for the business organization using the change and that project success can be sacrificed for reaching product success. On the **IT success criteria** 3/5 answered project and 2/5 felt like product success should also be the target of IT organization and that IT drives for that, although appreciating the project success. It was also noted that both are important for the IT and it is a delicate balancing of project and product success for IT organization to earn its place and not, for example, getting outsourced.

PM2 comments on the product success importance and the disconnection of IT and business targets:

If it is IT, then project success is more important and if the organization is "real people" [business] *then the success of the product and the deployment is more important. For business it is more important that it* [change] *is being used and for IT it is sufficient that it was done. [...] and perhaps IT is easier to lure to make the same again after two years.*

On a similar tone, PM4 comments on this disparity:

IT usually is after the project success, for business it is the product success. [...] a conflict of interest, rather serious one too.

PM1 states that the product should be the priority, whatever the organization is:

> *Most of the whining usually comes from that "this is crap". So, I feel, although they* [organizations] *claim whatever, they will still in the end judge what comes out of the pipe, i.e. the product. This product success view should be the starting point on the both sides* [IT and business organizations].

As the project vs. product success criteria were enquired from the organizational perspective, it was still further clarified with a somewhat overlapping question to validate the earlier announced comments. This was done with the **cost/time/scope slippage allowance**, which got 5/5 *yes*. PM2 stated that if change management is done properly, these slippages are getting accepted:

> *If change management is done properly, i.e. key persons in the receiving organization know what it [slippage] is about, for example that we must do some more bug fixing or testing... [...] overall, if the receiving organization knows the reasons, then slippages will get accepted. Naturally the target state should be that these* [slippages] *do not happen.*

PM3 made a good notion that the change receiving deployment organization has most probably already invested money and resources, and thus deployment is more easily getting things forgiven, as the already invested money would otherwise be all wasted:

> *Yes, they* [slippages] *are accepted. Because once reaching deployment, most probably something has been created or bought something, which costs some sum of money. Then one must estimate that how much the creation or purchase cost versus that someone takes it into use two months late of schedule versus the case that we bought that thing, but nobody took it into use on the date we said it should be taken into use.*

Last, one rather interesting and anti-dogmatic note from PM5:

> *Well, these practical lessons learnt from project management is that you always need to sacrifice one of the three* [scope, time, cost] *[...] in practice you cannot finish a project without sacrificing one. [...] yes, they* [slippages] *are accepted.*

7.6.2   Five most important success criteria

For further understanding the nature of deployment projects and how they should be judged for success or failure (apart from project-product success), the respondents were asked to pick five success criteria they thought deployment projects should be metered on. These were classified as top success criteria and the coding used was freely adopted from a set of coding created by Thomas and Fernández (2008) as a result from their mini-case study on project success criteria (Section 3.3). Interviewees ranked five criteria starting from the most important (1) and proceeding to the least important (5). The results are presented in TABLE 7:

TABLE 7. Respondents' priorization of success criteria

Criteria[1]	PM1	PM2	PM3	PM4	PM5	RANK
On-time (P)						
On-budget (P)						
Sponsor satisfaction (P)						
Steering group satisfaction (P)	4		3			V[2]
Project team satisfaction (P)			2			
Customer/user satisfaction (P/T)[3]	2	1	1	1	1	I
Stakeholder satisfaction (P/T)					3	
System implementation (T)						
Met requirements (T)				2	5	V
System quality (T)		5		3		
System use (T)	5	2				IV
Business continuity (B)			4	4		
Met business objectives (B)	3	4	5	5	4	III
Delivery of benefits (B)	1	3			2	II

The scoring is not directly compatible for comparison with Thomas and Fernández's study as their article did not elaborate different coding in a way that no room for interpretation would not have been left. As highlighted already, "system implementation", for example, can mean either creation of the system (e.g. software package and processes) or deployment of (taking into use). And as there is no "standard" coding for success criteria and variations

---

[1] In *Criteria* following categorization is used: P=*Project success* criteria, T=*Technical success* criteria and B= *Business success* criteria

[2] *Steering group satisfaction* and *Met requirements* share the 5th place with equal scoring

[3] Both *Customer/user satisfaction* and *Stakeholder satisfaction* belong to two categories: *Project* and *Technical*

are as many as raindrops in the rain, the coding provided by Thomas and Fernández was considered adequate for the purpose, as it captured at least the commonly used criteria and considered system use and business benefits too.

For the record, the respondents were elaborated on the following coding entries:

- *Customer/user satisfaction, Project team satisfaction, Steering group satisfaction* and *Sponsor satisfaction* – all of these entities are stakeholders. They could be subgroups of *Stakeholder satisfaction* and should be considered as such. Thus the respondents were given a change to highlight one or several of the stakeholder group's satisfaction as more important as others or the general *Stakeholder satisfaction.* The other way around, by prioritizing general *Stakeholder satisfaction* they could imply that the overall satisfaction of all groups is more important than satisfaction of one or several stakeholder groups.

- *System implementation* was explained as the technical deployment of the system, e.g. installations of client software, functioning technology, response speed of the system, availability of the system etc.

- *System use* in this context is a criterion, which can be metered immediately after the deployment and stakeholder usage of the system or it could be metered for example six months from go-live to provide metrics on how actively the system is used. It could be translated to a part of product success.

- *Met business objectives* is explained as providing the planned financial savings, reduction of staff for manual work, supporting the company strategy targets or, for example, making business compatible with another company for an effective merge.

- *Delivery of benefits* is explained as more from operational perspective than *met business objectives.* It is about efficiency of operations, simplifying

processes or system landscape, better availability of information and information quality etc.

Analysis of the results gives a clear result that project managers consider *project management* success criteria as only the third important category (Section 3.3) after *business* success criteria and *technical* success criteria. Customer/user satisfaction is considered the ultimate success criteria (categorized as *Technical* criteria as the PMs commented that they mean delivery satisfaction, not project), which should be used for analysing deployment project success/failure. As an example, quote on the priorization, PM1's reasoning is the following:

> *Number one is the delivery of benefits, means that I still look this from the business perspective so that the end user or customer does not necessarily know what is its own good. Anyways, it follows as a good number two, this customer and user satisfaction, because if that is not there, then number one does not happen. And then I had as third this business objectives that I started from a thought that some benefit must be gotten out of to justify the project itself. Then, if those customers are not happy, they'll not use it and the benefits do not get realized... so this number one and two were kind of not easy to decide, which one is more important. As a four I have then this steering* [steering group satisfaction], *means that you still have to get that E4* [milestone] *or E5 or with what the* ****[cursing] *you want to call that project closure, so that we have the approval from sponsors and stakeholders for the thing, i.e. as they stamp it approved, then it is done and the bigger picture gets recognized through them. For number five I take that system use, because I'm interested to hear afterwards, if it was just passion of the moment or what* [system being used].

The results strongly support the findings from project/product –success analysis (Subsection 7.6.1) that although the respondents are/have been IT project managers, they still value business and product success more important than the project success. The author speculates that this tendency is only built over time as the project managers experience and view on projects and business becomes more extensive. This is supported by several open comments from the interviews, when asked, if their view on success criteria has changed over the years. PM4 summarized this:

> *Yes* [they have changed], *earlier as a project manager I concentrated on-time, on-budget, those that I do not consider...yes, they are important, but then again are not, if we are metering real success [...] secondary stuff, if we get something done.*

Similar reasoning and meaning can be interpreted from the comment by PM5:

> *Yes, sure* [has changed] [...] *This understanding has changed over the years, so that in the beginning I had kind of project management success –view on things, to get the projects done on the schedule and budget and everyone would be happy – it just isn't so.*

7.6.3 Project failures and curve of uncertainty

A certain understanding can be already formed from the earlier chapters' comments on product success importance and the role of project management success on how they might impact the deployment project over-cost, over-time and limited scope deliveries. The traditional "holy triangle" (scope, time & cost) foundations are less important and can be sacrificed for "greater good", i.e. the product success. This seems to be characteristic for deployment projects.

To understand how do deployment projects then fail, the respondents were asked to reflect their own project history. They are enquired on how many of the deployment projects or programs that they have participated have failed (been quitted) and how many deployments have encountered at least 30% cost or time overrun (could be considered as "failed" in Chaos report). The first is classified as **total failures**, the latter as **overrun**. The participants were also asked, how many of their own lead deployments (being a project or program manager) exercises have failed – this is coded as **own failures**. Additionally some reasons for overruns are asked. TABLE 8 shows a summary of the analysis:

TABLE 8. Project failures and overruns

Respondent	Total n of depl.	N of failed depl.	N own depl.	N of own fails	N of over-	% of all fails	% of own fails	% of overruns
PM1	10	3	2	0	10	30	0	100
PM2	140	20	10	0	45	14	0	32
PM3	9	2	7	2	3	22	28	33
PM4	10	5	1	0	5	50	0	50
PM5	23	0	15	0	9	0	0	39
TOTAL	**192**	**30**	**35**	**2**	**72**	**15,6**	**5**	**37,5**

A noticeable 15.6 percent of the reported deployment projects have failed. This was judged on a criteria that the project was cancelled or the project was executed as a project success, but the change was not used after few months of project closing (product success failed). Hence, the pure project failure would be lower, but for purposes of this study the view with product success included was more interesting. The ~16 percent, although not as high as in the Chaos report (Section 3.1), where the latest 2009 failure rate was 24 percent, still means that every 6th deployment project fails. The latest Chaos report showed 44 percent of projects being *challenged* (time or cost overruns or only part of scope realized), which is not very far from the analysis of this research: ~38 (37.5) percent.

Only one project manager reported that a project or program lead by him has failed in this sense. It is almost probable that similar cases could be discovered from other respondents (now reporting "*0*") with further questions. For the purposes of this research that is not of high relevance, but still mentioned as a possible credibility issue on the figures. Among reasons for project/program failures the respondents listed: customer organization changed, business case

lost, economic conditions, complexity, portfolio priorization, failed change management and failed communication.

Some reasons for deployment project overruns were already mentioned as the main reason being the product success focus. Additional reasons were gathered from the interviewees and listed: scope changes, vendor incapability to deliver, overall project complexity, failed change management, technology challenges and optimistic estimates.

Perhaps the nature of deployment projects with rather high stakeholder counts and the need to win the human change resistance play a crucial part in estimation accuracy of deployment project indicators (time, cost, % of success, etc.) at given time in project execution. To gain an understanding on this and thus reason one more factor for project overruns, the author created a modified version of Vogelezang's Cone of Uncertainty (FIGURE 11), which Vogelezang has adapted as a linear version (to show more asymmetric nature of underestimating vs. overestimating) from the well known COCOMO model. The author's Model of Perceived Deployment Uncertainty is shown in FIGURE 12:
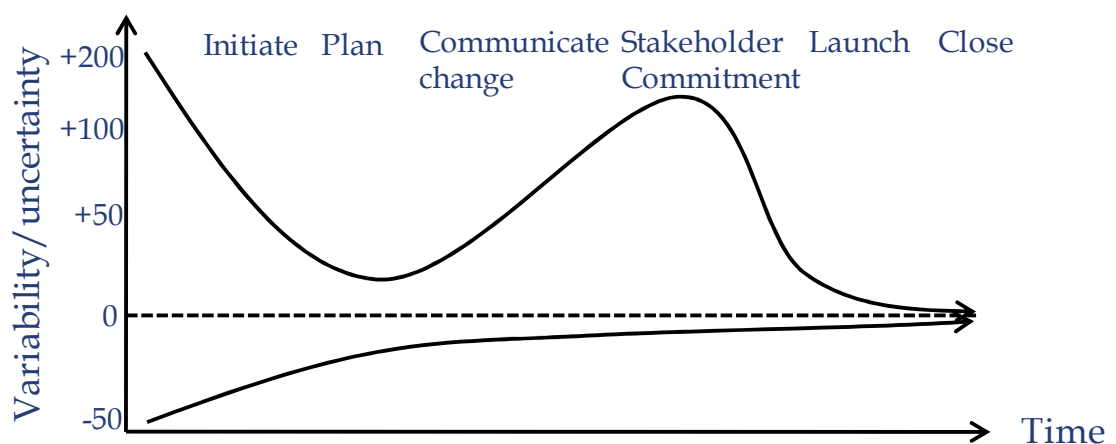
FIGURE 12. The author's Model of Perceived Deployment Uncertainty

The model covers a single deployment project and *Launch* corresponds to the situation where the IT artifacts are taken into the planned scope of usage (deployed as planned). The model does not cover the time after project closure, as it is about project estimation. *Stakeholder commitment* is the point in time that the steering bodies and/or the receiving organization have given their grant to launch the IT artifacts. Model anticipates that the critical mass of the stakeholder community is presented via steering functions, i.e. steering representatives have heard the voice of the to-be user community from their own organization's perspective at minimum. Hence they (steering representatives) are presenting also the critical mass of the whole stakeholder community. Full scale introduction of the artifacts to the entire stakeholder community takes place immediately after *Launch.*

The model is explained as chronologically advancing from left to right. At the very left, when deployment is *Initiated*, the uncertainty of indicators and success is at its widest (the area on the left between the curves (high and low) is the level of uncertainty). As the project proceeds to *Planning,* the uncertainty reduces and estimation of duration, cost etc. varies less than in the *Initiate* phase. Where the model differs dramatically from Vogelezang's COCOMO-adjusted model (where uncertainty continuously diminishes) is that the author's model suggests that at the stage of *Communicating the change* to the main stakeholder groups and driving to get the *Stakeholder commitment* (e.g.that critical 80% as per the ITIL (Section 4.2)), the level of uncertainty actually raises for a period of time. This temporary raise in uncertainty is due to uncertainty of people and the difficulty to estimate and understand human behavior. At the *planning* phase there might be a very good and solid plan, also covering organizational change management and stakeholder communication and ultimately, looking retrospectively at the *closure* of the project, those estimations might have been pretty good. Nevertheless, after even most rigorous and solid planning, the uncertainty raises as human factors come into play. This means

that even if *planning* has considered all possible toleration to change and what could happen, the planners and project people are, for a moment, unsure of how the organizational change management will present its outcomes. If estimations are asked during the *stakeholder commitment* "phase", they most likely differ more than in the *planning* phase. This is due to the undecided outcome of human factors and the duration still needed for deciding one way or another cannot be reflected from plans, as human factors are complicating the estimation and presenting a factor, which has more fluctuation than the ones in plans.

Although the IT artifacts introduction is not over after the *stakeholder commitment* has been won and the grant to *launch* the artifacts into use is given, the estimation accuracy dramatically gets better, once the human factors are mostly "won". Another raise in the graph could be presented after the *Launch*, as the whole stakeholder population perceive the IT artifacts and could resist them after introduction as well. This is not covered in the model. The main focus is on showing fluctuation of accuracy at the main organizational change management activities phase.

Once the change has been "sold" to the main stakeholder population and the project can consider change resistance as mostly won, the estimating accuracy dramatically improves again and the level of uncertainty drops. An absolute correct estimation does not exist until project closure. The angle of attack of the curves, as well as the values provided (-50, 0, +50, +200) are just for the purpose of an example and only demonstrate the phenomena. The positive values correspond to overestimates and the negative to underestimates (human nature tends to overestimate own skills and underestimate challenges (as noted earlier) and that is why the curves are not symmetric in the beginning).

The model was shown to the correspondents with a model provided by Vogelezang (APPENDIX I - Interview slides) and they were asked to pick the curve which best describes deployment project uncertainty, which is classified as **perceived deployment uncertainty** and is a pair comparison of the two models (coding *A, B*). 5 out of 5 selected *A*, which is the author's curve. The analysis then is that uncertainty/estimation models commonly used in software development projects do not fit as such to deployment projects, where great deal of uncertainty is arising from human factors, which are very difficult to estimate. This notion might also explain some of the overruns and even failures of deployment projects. Respondents commented on the perceived deployment uncertainty and the curves with following, starting with PM1:

> *I feel that "A" curve is the one, but perhaps as with not so strong angle of attack on the curve.*

As did PM1 comment on the angle of attack of the upper curve, so did PM2:

> *"A", it always creates some turbulence* [change]*, in real life I'd say that it is the A curve. It can be a quick peak, but something there always happens.*

PM3 addressed the change resistance directly as the reason for the uncertainty level raise:

> *It is quite clearly the "A", because, when planning has started, we "know" that it will be finished in some period of time defined. Then we start to communicate the change and in the beginning everybody is "yes, yes, yes", until they understand that the change will impact them too and then comes the "la resistance". Then followed is getting the stakeholder commitment and it can be a point, where the whole project can fail, so you must work hard there.*

7.6.4   Summary of theme

Deployment project success criteria was investigated to later better understand the findings behind the main research questions. All the interviewees commented that they personally consider product success more important than

project success. Some had even renegaded in their own IT organizations, in cases when project success had been priorized, compromising the product success. A conflict of interest was identified between the providing IT organization's project targets and those of the receiving business organization's targets.

Further enhancing the fact that product success is more important for the interviewees they were asked to pick five most important deployment project success criteria. The results show that the PMs valued the most *business*-category criteria with *technical* –category criteria following. The least important were those which belonged to the *project*-category. One single criterion was above all the others: *customer/user satisfaction*.

Lastly, the project failure rates and overruns were investigated presenting almost 200 deployment projects or programs. Although failure percentages were lower in the covered deployment projects than they are in studies like the Chaos report, still a considerable 15.6 percent had failed. Some 37.5 percent had experienced overruns of cost or time. In addition to the already presented and enquired failure/overrun reasons, the author presented the interviewees with his Model of Perceived Deployment Uncertainty. This model considers especially the need to win the human challenge, the change resistance. It suggests that when people are in play the uncertainty of the project estimation (time, cost, success%) raises for the period, until the change resistance can be considered won (e.g. 80% of stakeholders accepting change).

The author's personal views are much in alignment with the findings and thus, once again, more enforce the hypothesis that the author had, once he planned to conduct this study. The author shares the similar "learning experience" as many of the interviewees, that in his early career years the project success criteria was mostly highlighted. As experience accumulated more and more, the

importance of success criteria shifted and the author started to invest heavily on the product success, rather even compromising his personal bonuses and credibility as the project or program manager (if deemed with scope, time, cost) than to create or deploy IT artifacts, which would not become efficiently used and which would not realize the business benefits for which the artifacts had been introduced in the first place.

Selection of this theme to the interviews is thus once again from the personal experience as a finding from personal growth in expertise and the values associated with this personal change. The literature study in Section 3.3 put these personal findings into the context of project and product success and then it was possible to research those in this empirical part. Hence executing this research has educated the author much more and put words to meanings, which earlier did not have solid concepts, and names for the concepts the models already existed.

Justification of this theme lies also in the fact that the author had a strong feeling that the success criteria and indicators used commonly (e.g. time, cost and scope) fitted badly to the deployments the author had lead or otherwise witnessed. There must have been some fundamental "thing", which made the metrics and the criteria not to fit and thus lead to situation, where much of the success research found in literature could not be applied to deployment. It was the strong drive to meet product success and the willingness to take personal "damage" to see some relevant change to happen rather than execute project properly by the book and get done with it, no matter what the end results would be six months from projects closure. Also in his roles as service manager or solution manager, owning project portfolios, the author has encouraged the project managers under his guidance to aim more for the product success as main priority and worry about the project success criteria discussion later. This reflects the fact that the receiving organization (which the author has also

represented) has been willing to commit more money and resources to see the product success rather than quit the whole project due to dark looking cost and schedule estimates, which were not in line with the original ones.

7.7 Theme: scope management vs. change management – which one is more relevant

Although all the subthemes presented and analyzed are important in themselves, they all are building blocks for this theme and the ultimate research question. This theme covers the comparison of scope and change management as a pair comparison between the two.

The theme also investigates, whether change management should be an integrated, project lifecycle long, process or a set of tasks in some project phase or several phases. Lastly, a topic of separate project role, *Change Manager*, is looked into. Hence, the uppermost classification for the theme is:

- Pair comparison: scope management and change management
- Change management being integrated project process
- Change Manager as a project role

7.7.1 Pair comparison: scope management and change management

Much of this research's literature study have been ensuring that the author's conceptual model is in line with the one found in science and practice. Most of the interview questions have been targeted to ensure compatibility of the interviewees conceptual models of scope management and change management to the ones of the author. All this has driven the work done to the ultimate research question presented: "Which one is more relevant in context of IT deployment projects for project to succeed– *scope management* or *organizational change management*"?

This was classified as **higher relevance for reaching success** and it was coded with two options: *scope management* and *change management*. Open comments for the reasoning behind the choice were also asked. All the respondents (5/5) selected *change management*.

Being the main research question, snippets from all the five elaborations are presented. PM1 starts with underlining the importance of "selling the change":

> *Change management* [is more important] *in the end game. Because with that you sell the product forward and if you sell it poorly it does not matter whether it is the world's best possible thingy or whatever you are deploying, process or stuff, so if you fail that change selling, then it will never be used and there is no realization of any benefits.*

PM2 stresses the importance of getting people to join the change:

> *That's a tough one* [the question]. *I'd still say that change management [is more important], because you need to get the people along. If scope management is done well, change management is then easier, but as a I need to choose, I'll say change management* [is more important].

The existence of exceptions, where for example a solid delivery date might be more important than change management, is recognized by PM3, although he sees change management as generally more important in most of the cases:

> *It kind of depends on what you are deploying. If you're deploying some financial tool, which has fixed delivery date, e.g. tied to quartal reporting* [then time plays critical role]. *If again it is some R&D or similar system, which is getting renewed or totally new one introduced, then it is change management. So that it was solidly brought to use and saw as a good thing instead of trying to push something out with some given schedule and change management does not happen. In my own experience, I'd say change management* [is more important].

Selling the change is also the topmost thought of PM4:

> *Earlier* [in the interview] *I ranked scope very important and change management important, but I'd anyways say that change management* [is more

important]. *It is because the selling still is very important, to get the people really take the change into use.*

PM5 inspects the relevance via making a negation of the issue, i.e. if either of the presented options fails, which one has more dramatic consequences:

> *Well, thinking through a negation – failure of change management has more dramatic consequences that failing scope management, so I think that change management* [is more important].

Judging from the flow of the many conversations (having long pauses, extended thinking, self reasoning aloud and similar activities), the selection between the two was not easy. For the coded part, no room for the author's misjudgment of the answers (wrong interpretation) was left and the interviewees were asked to clearly present their choice from the two. Care was also taken by the interviewer not to, even remotely, build an expectation of the answer and to remain as objective as possible.

7.7.2   Change management being integrated project process

The relevance of change management was revealed when the respondents were asked, what is the nature of change management from project lifecycle point of view. Is it a process extending the project lifecycle integrated to the underlying project management practices or is it a separate phase or phases in a project, i.e. covering only part of the project lifecycle.

Classification of an **integrated process extending the project lifecycle** was provided and coding was provided and coding *yes/no* is utilized. All the respondents saw change management as covering the very total of the project and being an integrated process, i.e. 5/5 *yes* answers. Few open comments are presented starting with PM1:

> *The whole ****ing* [cursing] *project long, because you need to sell it in the very beginning already, so that people cannot come to you and start buggering you*

that "who has decided this" and "why have I not been told". And to my experience, none of the projects have gone as planned, and that means for example that good and constant communication is more than needed.

PM4 takes a philosophical view on the topic:

It [change management] *goes through the whole deployment project, it is needed all the time … constant* [in nature], *the change management … nothing else stays but the change.*

7.7.3   Change Manager as a project role

A notion was made in the literature study that a separate project role, Change Manager, might facilitate project success by ensuring efficient execution of change management activities. A set of activities/responsibilities were introduced to the respondents (as per Shields (2001) in Section 4.1) described as an assigned project role responsible for them. The interviewees were asked to comment whether such a dedicated project role would facilitate project success or not. Furthermore, the role was assigned to a company internal person or an external person (e.g. consultant) to understand better the significance of the person being internal and recognizing the company ways of working, organization and politics or being an external person without recognized the company status.

For the internal person the classification used is **internal Change Manager helping project to succeed,** and for the external; **external Change Manager helping project to succeed**. Both are coded as: *not at all, slightly, moderately, significantly & greatly*. Results are shown in TABLE 9:

TABLE 9. Change Manager -role impacting deployment project success

Respondent	Internal Change Manager	External Change Manager
PM1	significantly	moderately
PM2	significantly	moderately
PM3	significantly	significantly
PM4	significantly	slightly
PM5	significantly	moderately

A person working from inside the organization with an assigned role (the tasks and responsibilities) of a Change Manager would, according to the respondents, make a significant difference in project success. The role being fulfilled by an "outside the company" person is not seen as effective and efficient approach, although moderate help for facilitating deployment success can be recognized. Some open comments further open this internal vs. external view. PM1 elaborates with:

> In my mind that kind of a role would be just needed in that selling role. […] So that when our sponsors, stakeholders and steering group cannot do these practical tasks, then this role would really have its place. […] In many cases the colour of your badge [refers to case, where internal persons have certain colour text in their ID badge and externals are tagged by different colour] plays a major role, so internal significant and external moderate.

PM5 identifies many of the tasks and responsibilities being scattered around the organizations in the working environments he has been involved in and that the role should be defined in the receiving (business) organization:

> The role described here [APPENDIX I - Interview slides] are nowadays scattered around. Some of the sponsors does what should be done, but only in theory. In practice the influencer [the person "selling" the change to ones organization from inside the organization]in the business organization does not do the activities. This role should be defined and facilitated in the receiving

organization, so implicitly yes, significant [project success facilitating relevance].

Although not classified and coded, to further validate the respondents' comments they were asked whether they would be ready to see additional trouble and effort to gain such a Change Manager to their deployment project. In this context the "additional effort" was explained being involved more than normally in the resourcing process. It was referred with, for example, of making the extra effort to convince the persons assigning the project budget to get additional funding for the Change Manager allocation. All the respondents told they were willing to walk the extra mile, especially in case of an internal Change Manager.

7.8 Summary of the theme

This theme was to investigate the main research question by comparing scope management and change management in their relevance to deployment project. All the respondents shared the same view of valuing successful change management more than scope management. This was the key finding of the research as well. On the nature of change management being an integrated process extending the project lifecycle, the same unison of opinions appeared. Lastly, a separate project role, Change Manager, was introduced and its ability to raise project success potential was investigated. It was seen that such a role, as an company internal person, would be a significant help for a deployment project. Even the role filled by an company external person would provide moderate help to finish the project successfully.

When reflecting the key findings from the theme to the experience of the author there is a strong alignment and agreement. Once again, this indicates the research setting has not failed. The author's experience has taught him to invest more on organizational change management than to scope management in

deployment activities. As mentioned, the author sees both as crucial areas but the comparison had to be done to educate communities; which one they should consider more to see their IT artifacts deployed and used successfully.

With regard to the role of a change manager, The author personally feels that such a role occupied in any larger deployment is crucial. If the project manager has the competence and time to execute organizational change management or can delegate it to other project roles and lead efficiently, then the challenge is solved. Still, from experience, the author has seen so many 5+ million euro programs which would have gained significantly, if a specific person would have been made explicitly responsible for the role of a Change Manager, preferable with full time allocation. In few cases, if such a role would have been held by a competent person, it would have most probably saved the project or program. If nothing else, it would have saved a lot of money and extra work put to an inefficient attempt without clear responsibilities of trying to make the stakeholders understand the change imperative, the need for change and how and why they should be involved and taking the change and IT artifacts into use.

On the matter of internal or external roles, the author would definitely prefer an internal role. Being internal gives much more credibility for the Change Manager to speak with understanding of the real implications of the change as a part of the organization. It would also open the doors to top management, where external persons might be ignored and not given the crucially important audiences to convince the management of the importance of the change, so that they could communicate it further to their own organizations.

7.9   Closing the analysis part

The most relevant part of the interview data was classified, coded and the findings were presented in the previous theme Section. A notion should be

made that much more data was gathered in the interviews, but in the author's judgement the data does not provide relevant addition of insight into research questions. Subjects, which were not covered are listed along the covered subjects in the interview frame, which is presented in APPENDIX III – Interview frame.

The analysis part of the research has broken the "big picture" down into smaller pieces, themes, and classified the theme key elements with the appropriate coding, and made a comparison with the secondary research data, the literature study data. The following Chapter 8: Conclusions, limitations and future research will proceed back to the big picture with interpretation of the holistic findings.

8 CONCLUSIONS, LIMITATIONS AND FUTURE RESEARCH

Much of the conclusions from the primary data (interview data) and its analysis were already presented in the interview study analysis in Chapter 7. Possible repetition of those in this chapter would be exactly that: repetition and thus not done. The summaries in the interview theme Section capture well the conclusions drawn. Neither are systematic comparisons to the secondary data (literature study) presented as it is also made familiar in the previous chapter.

This chapter concentrates on concluding the answers to the research questions, although it will additionally provide some further conclusions. It will also comment on the reliability of the study, possible limitations, but also to the significance of the research. Later on, the author will express his personal view on the findings. Lastly, further research possibilities provided by findings of this research are suggested.

8.1   Concluding research question 1

The first, and most important, research question was presented as:

Q1: Which one is more relevant in context of IT deployment projects for project to succeed– *scope management* **or** *organizational change management***?**

On the basis of the literature review there was no clear bias detectable and the both were presented as very important. **In the interview analysis of the IT deployment project management experts' comments this was 100% :"Organizational change management is more important"**. The author was not left into doubt that the respondents would not have had similar conceptual models of the presented concepts as presented in this research and owned by the author. There is also no doubt that the both would play important role in the respondents' professional work. The decision between the two was not easy

for all the respondents and they reasoned their choices quite thoroughly to themselves and the interviewer. Finding is to be considered relevant also as it is contradictory to literature findings mentioned in 1.1: Background. This indicates that the mentioned literature promoting scope management and utmost importance of project scope, should be considered and applied only with care in context of deployment and deployment projects.

The direct implication from this finding is that IT deployment project managers should pay attention to the importance of both, and especially organizational change management, when dividing their precious work hours. This affects directly to the success of the deployment activity.

On a critical eye, the above finding seems to be especially relevant in rather large organizations and things are relevant always only in their own context. For example, if a ultimate deadline for financial accounting system deployment is that it must be in use by a certain date or the company risks its financial stability, then the constraints of time and scope most probably should be more concentrated on. In such a case the necessity (presented earlier) can be easily made so strong that change management is rather easy (use it or get sacked).

Judging from the findings, scope management and organizational change management are two processes which need to be carefully set to right "values", otherwise the project faces great difficulties (or fails). This means that neither cannot exist without the other and there are certain tolerances for both which need to be respected, and crossing either of the tolerance limits puts the project in the danger zone. On the basis of the findings, the tolerance for organizational change management is thus smaller and greater slippages are permitted in scope management. The author visualizes this with an example in FIGURE 13:
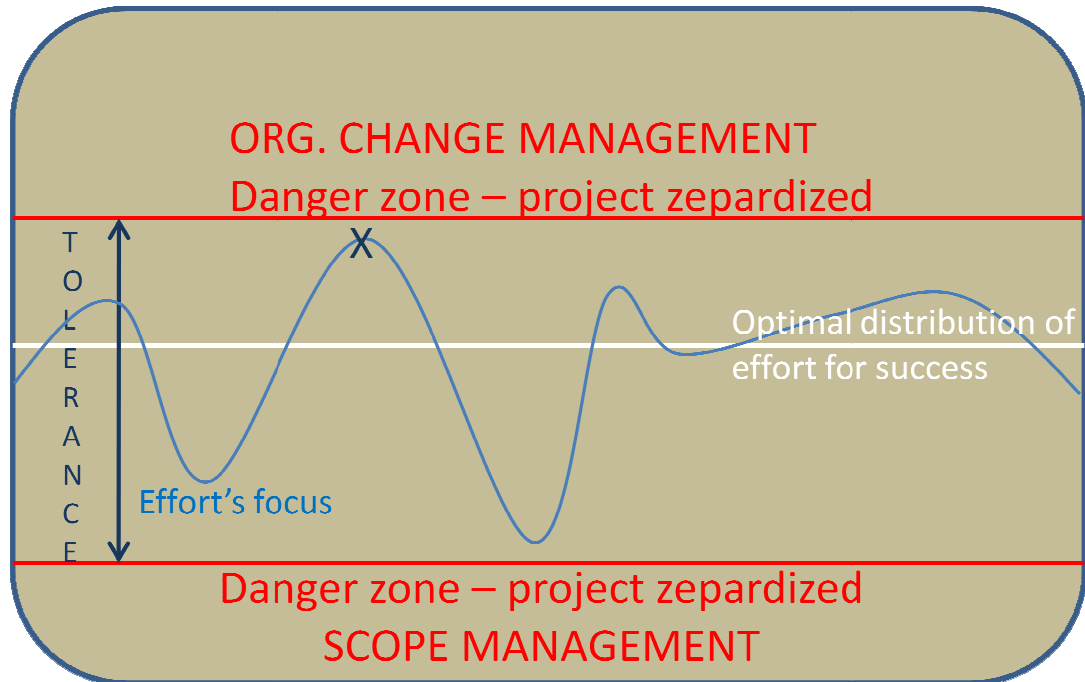
FIGURE 13. Example of scope management and organizational change management tolerance in deployment

The figure demonstrates an example project's distribution of effort put to scope management (*Effort's focus*) and organizational change management – this is presented by the curve. If a project (manager) is ignoring either too much, the success probability (the curve) will get closer to the limits of jeopardizing the project success (the danger zones). The white line presents an optimal balance, where a project (manager) is optimally concentrated on both processes, i.e distributing one's effort optimally for achieving success. Interpretation can be further elaborated by analyzing a single point in project execution time, presented by "X" in the figure. Analysis of the mentioned "X"- point is that the project (manager) has concentrated too much on scope management, neglected change management (close to danger zone limit), and the project is close to serious obstacles and failure. Following the curve further to the right, the project (manager) has identified the situation and concentrated on the change management more. As per the research findings, change management is deemed more important in IT deployments and thus the tolerance is smaller for

it than it is for scope management (presented also by optimal success probability line being closer to the danger zone limit of change management.

8.2 Concluding research question 2

Second research question about understanding deployment project managers motives and what drives them, project success or product success:

Q2: What do IT deployment project professionals consider more important from their own experience: *project success* **or** *product success***?**

The analysis provided data indicating that **All the respondents judge their primus motor being the product success**. This is directly reflected also in the five top success criteria that the respondents were to pick for a deployment project: all of the picked criteria facilitate product success and vice versa. Conclusion is that experienced deployment project managers are willing to jeopardize the indicators and criteria they are usually judged on (scope, time, cost) and drive for achieving product success. This is because they feel that achieving product success and criteria associated with it creates something lasting and achieves the business value initially planned. Some comments also concluded that professionals have personal feelings about the endeavors they are leading and achieving mere project success (with jeopardizing product success and lasting use of the beneficial change deployed) does not satisfy their concept of a "job well done".

It was also noted from several comments that this fact is tied to the level of experience of the project manager, i.e. some of the respondents told that when they were "rookies" they aimed for project success and did not see the bigger picture, but as their experience has accumulated, they now value product success more and their priorization of success criteria has changed over the years.

On the critical side, there are situations when even the most experienced and product success-oriented project manager is forced to appreciate the project success. This is more probable to happen customer projects, where the customer is a company external body (e.g. another enterprise/company) and charging the customer amount to keep the project profitable customer charging is an important factor and even more, if contracts have been tied to certain deliverables on certain dates.

As well as with the research question 1, neither in case of project or product success, the outcomes are black or white. It is possible to fail the project measured with the project criteria and succeed when measured with product criteria and vice versa, but in most of the cases the outcome is certain success in both project and product criteria, e.g. neither was a total success nor a failure but the total project outcome was "ok".

8.3   Concluding research question 3

Last of the research questions was to investigate whether the generalizations of IT project success criteria, critical success factors, reports and statistics can be applied to the context of an IT deployment project:

> **Q3: Can the studies and reports which refer to generalized IT project success/failure factors associated with scope management and/or the triple constraints of time, cost and scope be applied as such in context of IT deployment projects?**

Literature study did not conclude either way as such, and some statistics from literature and those of the empirical study were even quite compatible with some generalizations (like the Chaos report's category of "Challenged"). But considering the "headline statistics", failure/success reasons and criteria widely presented in literature (although many (like Chaos) lacking scientific value),

which often consider scope as key success factor and scope management key process area, this research is stating that they cannot be used as such in deployments. As presented in the analysis part, scope and change management are different in create and deploy projects, have different significance, and relate differently to project success. The statistics commonly seen promote the triple constraint of time, scope and schedule as the main indicators (or extension of these), which have been proven to be secondary drivers and indicators in deployment projects. Also deployment projects aim for product success, not project success, and as per research question 1's findings change management is the more important process area, which facilitates more product success criteria and product success.

It is thus concluded that, in the context of deployment projects, the indicators do not apply as such and care should be taken, when interpreting them to deployment. Focusing on wrong success factors/indictors (like time or cost) and jeopardizing organizational change management can lead to very unwanted results. As with *research question 1 (8.1)*, this should be considered also in context of literature listed in 1.1: Background, as being contradictory to the listed literature. This indicates that the mentioned literature should be considered and applied only with care when considering deployment and deployment projects.

Again, nothing is black-and-white and when interpreting the generalized (or software development project studies etc.) findings in reports/statistics, special care must be applied to understand the criteria they are based on and whether they even consider project success and product success. Naturally, information (not disinformation) is valuable, but it must be put to the right context.

8.4   Other conclusions

The author would like to highlight few additional conclusions. First is "the not presented hypothesis" of the author (motivating for this large research) that the scope management, organizational change management, product and project success and success criteria and available success/failure studies all relate to each other was correct. There is a big picture which is relevant for the large scope of this research. To demonstrate this claim, an example of reasoning is provided: Expert deployment project managers promote organizational change management → this facilitates product success and product success criteria are very different from project success criteria → project success criteria are very commonly used in studies of project success/failure and best practices → generalizations from studies are dangerous in the deployment context.

A conclusion of relevance is the notion of need for shared scope and targets (as well as success criteria) between possible create and deployment projects (e.g. in same IT program). Both contribute to the success and failure of the other and thus the message and target must be shared and projects should be speaking "with one voice" to facilitate e.g. stakeholder satisfaction and general success. This is sadly not promoted in the encountered literature.

Another relevant conclusion is that create and deployment projects do not share similar uncertainty models. The one presented as an alternative example to one of the authors did not apply in the respondents' minds to deployment projects and the author's model was better suited. The author's Model of Perceived Deployment Uncertainty promotes a the following characteristic of deployment projects and change management combined: at the stage of the main organizational change management and seeking their commitment, the level of uncertainty actually arises for a time, until the critical mass of stakeholders have accepted the change.   This length of period is hard to estimate as each

stakeholder "walks" his own change curve as an individual. Statistics could help the estimation, but the fact still is that uncertainty caused by human nature exists and affects the deployment estimation. This could also be one of the reasons for the perceived high over runs in budget and time in deployment projects.

The last relevant finding and its conclusions are that there is a triple disparity with IT organization, business organization and experienced deployment project managers. Few elaborations by the respondents indicated that their IT organizations promote project success where business organizations drive for product success, and in between of the two, deployment project managers aim for product success (which is in contradiction with their own organization main interests). The conclusion is that the role of the IT organization is not an easy one. It must carefully balance between providing enough value for the business to justify its existence and, at the same time, ensure that the other meters it is judged on (time, cost) are adequately in order. This does everything but promote the goal of IT being extension and tool of business strategy. In fact, in some cases it can be more harmful to the company business and the effective (not just in name) ownership of IT project portfolios and budgets should perhaps be in either the business organization itself or an organization "in between " the two to better realize the business benefits of IT. In several large companies, the author has witnessed the fact that although business is "on paper" responsible and owning the IT portfolio, the effective ownership and management of the portfolio is in hands of IT, which executes the portfolio management to its own best vision available – which in many cases does not consider enough the business strategy or the business goals for IT.

A secondary finding and conclusion could be the very disperse selection of definitions and conceptual models existing in the literature for almost all the presented concepts: deployment, scope, scope management, change

management and project success. For all the relevant concepts a synthesis was generated and presented.

8.5 Significance of the findings

The conclusions and findings presented can be considered very significant, as projects and organizations are constantly aiming for raising the number of successful projects and efficiency (business benefits included). It is important to understand the motivators and characteristics of experienced project deployment managers, the nature of the deployment project themselves and the success criteria associated with these. Project managers early in the professional career can learn much from the findings presented and better understand the phenomena in their environment, and the bigger pictures of their company business. They can also divide their limited resources better to underline organizational change management efforts and to ensure that product success will happen to the needed extent, so that the beneficial changes will be used by the stakeholders and business benefits will be realized.

The research is quite unique and encompasses many complex concepts to facilitate the understanding of their dependencies and interactions. A similar study has not been made earlier and the research conducted is expanding both academic doors for further research and it might be saving millions in (if utilized) a single large project heeding the findings and conclusions of the thesis. If the findings and lessons learnt will be adopted by a larger audience (after being validated with further studies) the financial potential is huge.

8.6 Reliability of the study and limitations

Much of the limitations have been explained in Section 6.3: Reliability of the selected interview approach. Scientific rigor in both the research setting and conducting the interviews has been exercised to the author's best ability. Still,

the sampling strategy is limited both in the number of interviewees as their possible self bias on some of the interviewed themes. The author has done his best (with the given resources) to ensure the needed data saturation and receiving as accurate data (not biased) as possible. The company size should also be considered in more detail. Although this research included respondents with a background in companies of several different sizes, the author still feels that the results better fit to large companies than medium-sized or small companies.

Also, as the study is aimed to explore the scope management, organizational change management, success criteria etc. in IT deployment projects, the findings and conclusions should only be viewed in the context of the given research setting. The author does not claim for any outright generalization of the findings and conclusions but, on the contrary, encompasses to critical utilization of the conclusions with the given limitations in mind.

Further study is required to validate the findings with a larger count and more heterogeneous data sources, and more detailed research and interview questions. The author proposes adhering to the interview approach as the concepts are quite complex and survey research can easily lead to misconceptions and corrupted data, if not elaborated to the very last extent of the concept and validated the conceptual model of the respondent.

8.7   The author's professional view on the findings and improvement suggestions

Author would like to use the opportunity to encourage both academic as well as practioner organizations (companies) to put special attention to organizational change management and scope management. Author proposes some concrete steps forward and will later summarize his personal "free word"-thoughts on the findings.

8.7.1 Few suggestions by the author for the improved project success

From my experience and perceptions from the interviews one thing can be considered as a fact: project managers *do* struggle with their time and resources available, be they money, time, people or their availability or their expertise. Project managers need to carefully consider their own time; where they can invest it most efficiently to successfully finish the project. This notion was one of the basis for this research and this research does offer one solid advice and guidance for deployment project managers (e.g. the scope and change management comparison).

Even with the findings presented, deployment project managers will not be sent off to extra holidays. Things will keep on pressing on more and more, time is of luxury. More horizon-considering solutions are needed. They are indicated both in the theoretical part as well as in the empirical part analysis. Project managers *need* to delegate their responsibilities. This is done efficiently *only* by introducing roles, which are taking the burden from the project manager, effectively ensuring one or more "dimensions" of the project (e.g. scope, change, cost, etc.) on behalf of the project manager. These roles then need to be tightly connected to the project manager and information sharing need to be fluent, both informal (notions, comments and best advice) and formal (scope meetings, change management meetings, change boards, creating needed documentation and so on) and working both ways. These roles need to be like the project managers and/or program managers right hand.

A role of Scope Manager was already introduced in Section 5.4 and Change Manager was introduced in Section 4.1. Similar roles *are* needed. The role of a Scope Manager needs revisiting for the purposes of deployment, but the author does feel such a role would benefit deployments a lot. Although surely beneficial, once modified for deployment projects, it could benefit IT programs quite significantly, actually. Such a role, Scope manager, could be working as

the right hand of the Program manager, but at the same time as the close counterpart of project managers in create project(s) and deployment project(s) (and also other projects) in the program. A Scope Manager could work with the role defined by the FiSMA and Dekkers and Forselius (2007) in the program's create project, but the role would also ensure consistent and effective dialog between create and deployment (and other possible project in the program) to ensure that both projects are having the same goals and scope. The author has perceived that, in regrettably many cases (and this was also elaborated in the comments of the interviewees), create and deployment projects do not share the same scope – both are looking at the scope with their own eyeglasses on, without considering the other party. Also a lot of information meant for the other project, which was to be delivered via the other project, is lost. A Scope Manager could ensure that the scope is shared and similar stakeholder communication of the scope is executed. This could be achieved with informal methods, but the author would suggest that in addition to possible change board meetings hosted (possibly) by the Change Manager, this role would establish regular and repetitive meetings with the project core team (program manager, project manager and needed other personnel) to elaborate the common, shared scope, and possible changes since last meeting. This would close much of the gap that the author currently perceives between create and deployment projects. It would also ensure that program management would share the current scope to top management stakeholders possibly being the responsibility of the Program Manager. And it would ensure that the "deals" Program Manager has done with top management are further formally introduced as change objects and shared with the program personnel.

Another suggestion the author wants to make, is for correcting the perceivably bad situation with recognition of importance of both change and scope management. The analysis findings state that the situation is not on the needed

level and the author can only but agree. What could organizations and educational institutions do to enhance the situation? Much.

Organizations: Management must recognize first the importance. This can be done via studies like this and further evidence. The importance must be grounded to money and efficiency lost. This can be demonstrated by few statistics of projects, where good scope and change management practices have been utilized compared to projects lacking those and putting the differences into figures of dollars/euro, time ,resource usage and success outcomes. Once the management has been convinced to recognize the fact and they are willing to pilot the efficient change and scope management tools and models within a limited organization, the education must take place. Education should cover the differences between project types' desired outcomes and how different stakeholders perceive the success. Human reaction to change should be elaborated via models like Kübler-Ross' Grief Cycle (FIGURE 7) (known by that name, although never quoted by Kübler-Ross by that name) and the model should be applied to the context of the organization – how the specific organization tends to react to change, how strong the biases (cultural, emotional etc.) are and how they should be encountered. Good scope management practices should be highlighted (consistent, controlled, informed and communicated & involving the stakeholders). Few people could be already at this stage be introduced to "formal" education of Scope Manager and/or Change Manager with a learning path and an incentive plan attached (money, organizational reorganization, title, job grade and whatever is available).

Even for pilot purposes, the tools and models to be used *need* to be tested in the back office. They need to be easy to use, non-bureaucratic and easily applicable. They must also include indicators, which are adapted for specific project types, like deployment and create. It is essential that the indicators consider the project type as it was found out in the empirical part that deployment cherishes

product success. Tools and models are not only introduced, but their usage is observed and improvement made by professionals using them.

Once piloting is over and the models and the tools are adapted for the organization's purposes, a major change program must be executed formally to deploy the models and tools. Selling factors must include the benefits project managers get from the tools and models (like improved stakeholder commitment, less time wasted pondering scope items and correcting misinformation between stakeholders etc.). It must be also introduced that the usage of the models and tools is observed (in the positive sense) and use of them is promoted via indicators for usage and also users personal incentives (if applicable). It would be beneficial to infiltrate few positive influencers into the community of project and program managers, who tell personal positive experiences of the models and tools over the coffee cup.

Organizations can and must also consider the time of the project managers and their competencies. Similarly as with introducing the models and tools, it can be highlighted to the management that usage of separate (subject expert) roles of Scope and Change Manager are financially beneficial. It must be highlighted, even with showing bad case examples from within the organization, how project managers lack time and or skills to efficiently make the projects as successful and efficient as possible. Any failed projects/programs could be discussed, how they could have been saved by usage of Scope and or Change Manager.

Educational institutions: Institutions like universities and polytechnics must come to understand that the "project manager education" they provide is, naturally, relevant but not creating any "selling criteria" for their students from the industry's perspective. Practices like the PMI's (although not even available for initiatives of project management practice) are so general and neutral that

they can provide very little to challenges like scope and change management in practice. Why can not institutions try to create some competitive advantage for themselves and create learning programs and even minors/majors, where possible specialty roles are recognized and students are concentrating already at this stage to be better professionals of change or scope management than many "so called" professionals of project managers with many years general project management expertise? I would be *very* interested to review a CV, where the recently graduated person is highlighting his/her studies of scope/change management and healthily proud statement of the person that he/she really knows about the area and, especially, its problems. Newly graduates are not of interest to me as an employee (they are usually more time invested than benefit gained), but of such a profile I would be interested and give the person (if otherwise fitting) a chance to prove one's skills. In the very early projects, the person could be working as an assisting role for the senior project manager to get a feeling of real work and get over the initial shock.

I really hope that someone somewhere is brave enough (be it education or business context) to think these opportunities and take the pioneering role in making project life more endurable, successful and interesting for each participating person and organization.

8.7.2   The author's "free word" of the findings

With considerable experience in IT project and program management both in software projects and deployment projects of generally all sizes and complexities, I mostly agree with the perceptions and most of the comments of the interviewees' on the investigated themes. I do recognize my own perception of project success has altered over the years and is nowadays very heavily concentrating on product success getting realized in needed level. At the same time, I respect budgets and time and do my best to balance the optimal outcome of the project and product successes.

Relevance of the organizational change management and getting people to "hop along the ride" has been recognized very early in the professional career and, somehow, I do feel pleased that other professionals feel the same way. Significance, for me, now comes out of this "shared view" with a few other professionals and I can speak of the topics not only as my personal opinions but as something that someone else also thinks alike.

Although reading my "sharing"-experience view on the results, one could start to doubt the findings are being biased, I have to admit that, naturally, I have not agreed with all the interviewees' opinions and have done my best to stay as objective as possible, not putting words to their mouth. A possible reason for so many shared conclusions is that the findings really are a quite general phenomena in the class of project managers the author and the interviewees are presenting. That is to be validated by further research, which is covered next.

8.8   Further research possibilities

Several future research topics can be suggested on the basis of the findings, conclusions, and also the limitations of the study. The following could be considered:

1. Validating the findings of this research – do they withstand a larger audience and more heterogeneous research environments? If not, what are the limits of the findings environment where they do still apply?

2. Qualitative study on the deployment project success criteria. Explore the literature on success criteria (which is evolving constantly) and especially the success criteria used in context of deployment-related activities. Which success set of success criteria deployment could be using and which criteria would fit into which category of deployment activities best? Meaning that a classification of criteria is also needed and

they must be put to context (in applicability) of different type deployment activities, which need to be identified first.

3. Further comparison of the presented key areas of scope and change management with for example the PMI's areas of knowledge – how do scope and change management rank and what could be even more important than these two and why?

4. Investigating the draft model presented by the author for Perceived Deployment Uncertainty. What could be the real values used with the model? An extensive research from many case studies should be executed to (1) validate that the model has needed justification for existence and (2) gather estimation data from the case project from their execution phases on what kind of curves (estimation) would the case studies present and could some generalization be made from them? Derived from the last mentioned possibility - theorizing and creating an estimation model for the deployment human uncertainty – what attributes would it have, how would they be biased and what would be the reliability of such an estimation model? This estimation model would complement the mentioned "Perceived Deployment Uncertainty", by bringing theory and practice into estimating the effect of human change effects. Together with the case studied "Perceived Deployment Uncertainty" these two could provide an effective tool and a model for estimating deployment projects in more detail.

5. As analyzed and concluded, there seems to be a disparity with business goals and IT goals. This leads to a situation where IT is not executing its activities efficiently for realizing business strategy and therefore the best for the company. The indicators and the success criteria are different when perceived from IT and business. Business needs to see product

success happen as the utmost outcome, whereas IT has to balance also with its indicators of time, cost, scope (and other traditional project success related indicators) to justify ITs existence in the company. What would be the metrics and criteria used in IT projects, which would facilitate both project and product success in such a manner that the possible disparity of interests of IT and business would be minimized?

6. Role of the "Scope Manager" in context of deployment projects. It was noted by the author that project managers are overtaxed with their responsibilities and do not even always accommodate the best possible characteristics for efficient scope management. At the same time, scope management is perceived in context of deployment projects. There is a role introduced by the FiSMA for project Scope Manager, but it does not as such fit into the context of deployment projects (as being very create and software creation biased role). How could such a role be either modified from the FiSMA's role or created from a scratch for deployment projects and what would the role be like, where would it fit into project or organization hierarchy and how could it best facilitate the "scope sharing" of create and deployment projects in same program to better make create and deploy work together for common goal?

7. What kind of learning programs and learning paths could be introduced for roles of Change Manager and Scope Manager? More concrete and elaborative work than provided in the context of this thesis is needed to make the roles more alluring and applicable both to education institutions and companies. What could such roles be specifically responsible for in the context of a create or deploy project? How would they sit in the program/project organization? How could one advance in one's learning path on the roles and keep the role interesting, i.e. a learning path is needed and constant challenge must exist.

REFERENCES

Agarwal, N., & Rathod, U. (2006). Defining 'success' for software projects: An exploratory revelation. *International Journal of Project Management , 24*, 358-370.

Akinsola, A., Potts, K., Ndekugri, I., & Harris, F. (1997). Identification and evaluation of factors influencing variations on building projects. *International Journal of Project Management , 15* (4), 263-269.

Aladwani, A. (2001). Change management strategies for succesful ERP implementation. *Business Process Management Journal , 7* (3), 266-275.

Association for Project Management. (2000). *Project management body of knowledge, APMBoK.* UK: Association for Project Management.

Baccarini, D. (1999). The logical framework method for defining project success. *Project Management Journal , 30* (4), 25-32.

Beldi, A., Cheffi, W., & Dey, P. (2010). Managing customer relationship management projects: The case of a large French telecommunication company. *International Journal of Project Management , 28*, 339-351.

Bennis, W. (1984). Transformative Power and Leadership. In T. Sergiovanni, & J. Corbally, *Leadership and Organizational Culture: New Perspectives on Administrative Theory and Practice* (pp. 64-71). Illinois: Urbana.

Bonnal, P., Gourc, D., & Lacoste, G. (2002). The Life Cycle of Technical Projects. *Project Management Journal , 33* (1), 12-19.

Boonstra, A. (2006). Interpreting an ERP-implementation project from a stakeholder perspective. *International Journal of Project Management , 24*, 38-52.

Boonzaaier, J., & Van Loggerenberg, J. (2006). Implementation of a project office to improve on project delivery and performance: a case study. *SAICSIT '06: Proceedings of the 2006 annual research conference of the South African institute of computer scientists and information technologists on IT research in developing countries* (pp. 206-217). South African Institute for Computer Scientists and Information Technologists.

Bower, D., & Walker, D. (2007). Planning Knowledge for Phased Rollout Projects. *Project Management Journal , 38* (3), 45-60.

Brandon, D. (2006). *Project management for modern information system.* London: Idea Group.

Brewer, J. (2005). Project managers: can we make them or just make them better? *SIGITE '05: Proceedings of the 6th conference on Information Technology Education* (pp. 167-173). New Jersey: ACM.

Brown, W. (2004). Enterprise resource planning (ERP) implementation planning and structure: a recipe for ERP success. *SIGUCCS '04: Proceedings of the 32nd annual ACM SIGUCCS conference on User services* (pp. 82-86). Baltimore Maryland: ACM.

Buttle, F. (2009). *Customer relationship management: concepts and technologies.* Burlington: Elsevier.

Carton, F., Adam, F., & Sammon, D. (2008). Project management: a case study of a succesful ERP implementation. *International Journal of Managing Projects in Business , 1* (1), 106-124.

Cleland, D., & Ireland, L. (2002). *Project Management: Strategic desing and implementation* (4th Edition ed., Vol. 1). New York: McGrav-Hill.

Cooke-Davies, T. (1990, 5 1). Return of the project managers. *Management Today - Business Infromation Management* , pp. 119-121.

Dekkers, C., & Forselius, P. (2007). Increase ICT project success with concrete scope management. *Software Engineering and Advanced Applications. 33rd EUROMICRO Conference on 28-31 Aug. 2007* , pp. 385-392.

Espinosa, J., DeLone, W., & Lee, G. (2006). Global boundaries, task processes and IS project success: a field study. *Information Technology and People* , *19*, 345-370.

Eveleens, J., & Verhoef, C. (2010). The Rise and Fall of the Chaos Report Figures. *IEEE Software* , 30-36.

Forselius, P., & Käkölä, T. (2009). An Information System Design Product Theory for Software Project Estimation and Measurement System. *Proceedings of the 42nd Hawaii International Conference on System Sciences* (pp. 1-10). Waikoloa: IEEE Computer Society 2009.

Galorath. (2008, February 7). *SEER for IT.* Retrieved June 30, 2010, from Galorath: http://www.galorath.com/DirectContent/SEERforIT.pdf

Gareis, R., & Huemann, M. (2008). Guest editorial: Change management and projects. *International Journal of Project Management* , *26*, 771-772.

Globerson, S., & Zwikael, O. (2002). The Impact of the Project Manager on Project Management Planning Processes. *Project Management Journal* , *33* (3), 58-64.

Goodpasture, J. (2002). *Managing Projects for Value.* Vienna, VA: Management Concepts.

Hartman, F., & Ashrafi, R. (2002). Project Management in the Information Systems and Information Technologies Industries. *Project Management Journal , 33* (3), 5-15.

Hiatt, J. (2006). *Adkar: A Model for Change in Business, Government and Our Community.* Loveland, CO: Prosci Research.

Hirsjärvi, S., & Hurme, H. (2008). *Tutkimushaastattelu.* Helsinki: Gaudeamus.

Jeston, J., & Nelis, J. (2006). *Business process management: practical guidelines to successful implementation.* Oxford: Elsevier.

Jones, C. (2010). *Software Engineering Best Practices: Lessons from Succesful Projects in the Top Companies.* New York: The McGraw-Hill Companies.

Jugdev, K., & Müller, K. (2005). A Retrospective Look at Our Evolving Understanding of Project Success. *Project Management Journal , 36* (4), 19-31.

Kassab, M., Daneva, M., & Ormandjueva, O. (2007, 28-31 8). Scope management of non-functional requirements. *Software Engineering and Advanced Applications. 33rd EUROMICRO Conference* , pp. 409-417.

Khosrowpour, M. (1999). *Success and pitfalls of information technology management.* London: Idea Group.

Kübler-Ross, E. (2009). *On death and dying: what the dying have to teach doctors, nurses, clergy and their families.* Oxon: Routledge.

Lee-Kelley, L., Leong, K., & Loong. (2003). Turner's five-functions of project-based management situational leadership in IT services projects. *International Journal of Project Management , 21*, 583-591.

Lieberman, B. (2001, 11). *Project scope management: effectively negotiating change.* Retrieved June 30, 2010, from The Rational Edge: http://www.ibm.com/developerworks/rational/library/content/RationalEdge/nov01/ProjectScopeManagementNov01.pdf

Mabin, V.;Forgeson, S.;& Green, L. (2001). Harnessing resistance: using the theory of constraints to assist change management . *Journal of European Industrial Training* , 168-191.

Man, C., & Leung, H. (2008). Challenges in information technology management. In *Challenges in information technology management* (pp. 161-167). Singapore: World Scientific Publishing Co.

Markus, M. (2004). Technochange management: using IT to drive organizational change. *Journal of Information Technology* , *19*, 4-20.

Maylor, H., Brady, T., Cooke-Davies, T., & Hodgson, D. (2006). From projectification to programmification. *International Journal of Project Management* (24), 663-674.

Merton, R., Fiske, M., & Kendall, P. (1956). *The focused interview: A manual of problems and procedures.* Glencoe,IL: Free Press.

Milošević, D., Martinelli, R., & Waddell, J. (2007). *Program management for improved business result.* New Jersey: John Wiley & Sons.

Myers, M. (1994). Dialectic hermeneutics: a theoretical framework for the implementation of information systems. *Information Systems Journal* (5), 51-70.

Nangle, S. (2003). Achieving rapid service deployment in an academic environment. *Proceedings of the 31st annual ACM SIGUCCS conference on User services* (pp. 1-3). San Antonio: ACM.

Office of Government Commerce (OGC). (2007a). *ITIL v3 Service Transition.* Norwich: The Stationery Office.

Office of Government Commerce (OGC). (2007b). *ITIL v3 The Official Introduction to the ITIL.* Norwich: The Stationery Office.

Office of Government Commerce. (2002). *Best practice for ICT infrastructure management.* London: The Stationery Office.

Office of Government Commerce. (2007c). *ITIL v3 Service Operation.* Norwich: The Stationery Office.

Office of Government Commerce. (2007d). *ITIL v3 Service Strategy.* London: The Stationery Office.

Orlikowski, W.;& Iacono, S. (2006). Desperately Seeking the ´IT´ in IT Research: A Call to Theorizing the IT Artifact. Teoksessa J. King;& K. Lyytinen, *Information Systems - The State of the Field* (ss. 19-41). Hoboken: John Wiley & Sons.

Patton, J. (2003). Unfixing the fixed scope project: using agile methodologies to create flexibility in project scope. *ADC'03: Proceedings of the Conference on Agile Development* (pp. 1-6). Salt Lake City: IEEE Computer Society.

Project Management Institute (PMI). (2004). *A guide to the project management body of knowledge (PMBOK), 3th ed.* Project Management Institute (PMI).

Raz, T., & Globerson, S. (1998). Effective sizing and content definition of work packages. *Project Management Journal , 29* (4), 17-23.

Rozenes, S., Vitner, G., & Spraggerett, S. (2006). Project control: literature review. *Project Management Journal , 37* (4), 5-14.

Sauer, C., Gemino, A., & Reich, B. (2007). The impact of size and volatility on IT project performance. *Communications of the ACM , 50* (11), 79-84.

Schein, E. (1996). Kurt Lewin's change theory in the field and in the classroom: Notes toward a model of managed learning. *Systemic Practice and Action Research , 9* (1), 24-27.

Shields, M. (2001). *E-business and ERP: rapid implementation and project planning.* New York: John Wiley & Sons.

Shore, B. (2008). Systematic Biases and Culture in Project Failures. *Project Management Journal , 39* (4), 5-16.

Slack, N., Chambers, S., & Johnston, R. (2007). *Operations management.* Harlow: Pearson Education Limited.

Sliger, M. (2006, March). *A project manager's survival guide to going Agile.* Retrieved June 12, 2010, from Rally Software Development Corporation: http://www.rallydev.com/kportal.jsp?doc=wp15

Slusarenko, S. (2010, March 9). *Change Management and Projects.* Retrieved July 11, 2010, from Project Management Wisdom: http://www.maxwideman.com/guests/change/abstract.htm

Smith, J. (2002). The 40 root causes of troubled IT projects. *Engineering Management Journal* , 238-242.

Stadler, H., & Kilger, C. (2005). *Supply Chain Management and Advanced Planning: Concepts, Models, Software and Case Studies.* New York: Springer Berlin Heiderberg.

Standish Group. (1995). *The chaos report.* Retrieved July 9, 2010, from Projectsmart: http://www.projectsmart.co.uk/docs/chaos-report.pdf

Stenzel, J. (2007). *CIO best practices: enabling strategic value with information technology.* New Jersey: John Wiley & Sons.

Steward, R. (2008). A framework for the life cycle management of information technology projects: ProjectIT. *International Journal of Project Management , 26,* 203-212.

Suresh, B. (2005, January 05). *Scope creep management.* Retrieved July 5, 2010, from Project Perfect: http://www.projectperfect.com.au/downloads/Info/info_scope_creep_mgmt.pdf

Syed Hussain, J., Rashid, K., Ahmad, H., & Syed Hussain, F. (2007). Effective Software Management: Where Do We Falter. *SEPADS'07: Proceedings of the 6th WSEAS International Conference on Software Engineering, Parallel and Distributed Systems* (pp. 13-18). Corfu Island: Word Scientific and Engineering Academy and Society (WSEAS).

Taylor, A. (2001). IT projects: sink or swim. *ITNOW 2000 (42)1 ,* 24-26.

Thomas, G., & Fernández, W. (2008). Success in IT projects: A matter of definition? *International Journal of Project Management , 26,* 733-742.

Turner, J. (2004). Five conditions for project success. *International Journal of Project Management , 22* (5), 349-350.

Turner, J. (1999). *The handbook of project based managemement (2nd ed.).* England: McGraw-Hill Publishing.

Turner, J. (2006). Towards a theory of project management: The nature of the project. *International Journal of Project Management* (24), 1-3.

Van Bon, J. (2005). *Introduction to ITIL.* Norwich: The Stationery Office.

Watson, R., & Kwak, Y. (2004). Development of a Parametric Estimating Model for Technology-Driven Deployment Projects. *Proceedings of the 2004 Winter Simulation Conference* (pp. 1290-1295). Washington: ACM.

Webber, L., & Webber, F. (2007). *IT Project Management Essentials, 2007.* New York: Aspen Publishers.

Whittaker, B. (1999, 7 1). What went wrong? Unsuccesful information technology projects. *Information Management & Computer Security* , pp. 23-29.

Vogelezang, F. (2007, November 5-8). Scope Management - How Uncertain is your Certainty. *International Conference on Software Process and Product Measurement* , 1-16. Palma de Mallorca.

Zander, A. (1950). Resistance to change - its analysis and prevention. *Advanced Management* , 9-11.

APPENDIX I - INTERVIEW SLIDES

INTERVIEW

Project Manager

DEPLOYMENT

- …is = "*A project phase or a project responsible of introducing and taking into use the positive change (technology, operational model, process, …) for the receiving organization and end users, it receives from either other project phases or as an input to deployment project. Deployment is responsible of ensuring needed change communication, training of the user population and approved hand over of the positive change to the validated and ready operations function(s)*"

CHANGE/SCOPE MANAG.

- Scope management = *"The project lifecycle long cyclic process, which controls that all the work required and only the work required to complete the project successfully is contained in project execution and no uncontrolled changes are permitted""*

- Change management = *"management of meaning, or with attempts to establish the credibility and legitimacy of particular definitions of problems and solutions with others, and to gain consent and compliance"* or *"process of setting, managing and meeting the expectations of stakeholders, managing awareness and preparedness to ensure project success"*

PMI – 9 knowledge areas

Area	Rank
Intergr.	
Scope	
Time	
Cost	
Quality	
HRM	
Comms	
Risk	
Procur.	
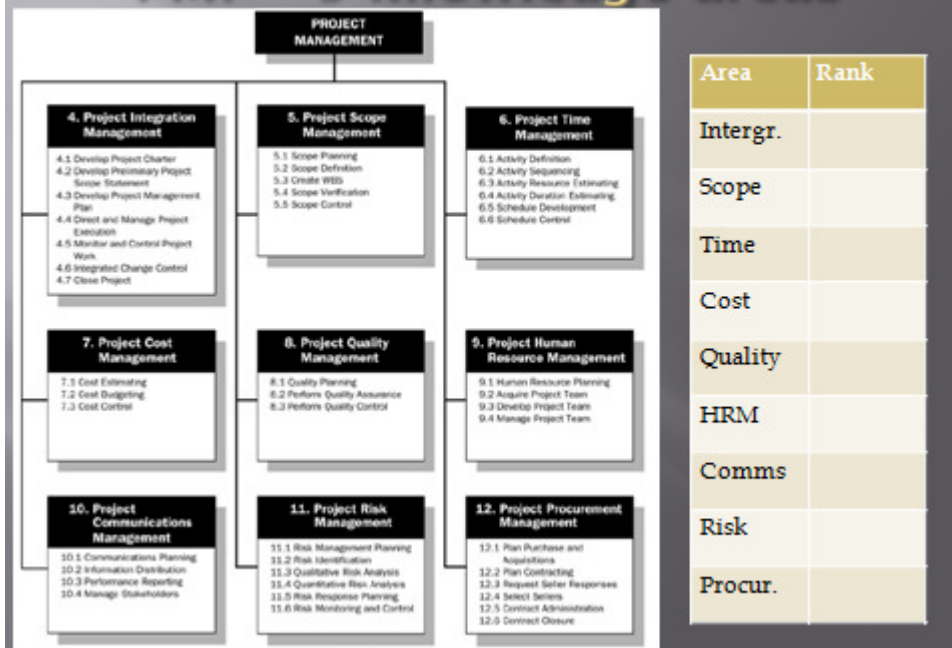
Project vs product success

☐ Project management success = being measured on traditional meters like time, cost and quality

☐ Product success = being measured on the project objectives

☐ In deployment context:
- Project success = on time, on cost and scope
- Product success = solution accepted by stakeholders, and solution in use

☐ Example: Sydney Opera House: 15 years to build and over 14 times over the budget, yet considered a great success and engineering masterpiece. In this categorization case Sydney would have been a great product success but a miserable project management failure

Success criteria

Pick 5 most important in order of releavance for judging deployment project success

Success criteria	Relevance
On-time	
On-budget	
Sponsor satisfaction	
Steering group satisfaction	
Project team satisfaction	
Customer/user satisfaction	
Stakeholder satisfaction	
System implementation	
Met requirements	
System quality	
System use	
Business continuity	
Met business objectives	
Delivery of benefits	

Deployment uncertainty — Which curve best describes deployment project level of certainty on cost, schedule and success?

Change Manager

Project Role: Change Manager

- **Assessment**: assess the readiness of the stakeholders for the change, identify different stakeholder groups getting impacted, identify power structures and supporters and enemies of the change
- **Involvement**: actively involve the stakeholders into project (design, testing, rollout, etc.), gather information and advice from stakeholders on processes, requirements, deployment strategies and organizational changes – ensure that input is asked and value opinions
- **Communication**: ensure two-way communication with stakeholders: project status, newsletters, importance of project (messaged by top management), go-live dates etc – there is no too much communication
- **Documentation**: making sure that enough documentation on the change and the possible artifacts of the change (tools, processes) exists and it is made available and communicated to the change targets
- **Training**: make the change targets involved into training delivery and training material assessment to ensure efficient training – make training delivery known, i.e. informed and available to parties in need of it.

APPENDIX II - PROFILES OF THE INTERVIEWEES'

PM1:

Age and gender: 31, male
Experience (years) in IT: 12
Experience (years) as an IT Project Manager: 2
N of projects/programs that has participated in: 15
N of projects/programs, which are of class: deployment: 10
N of deployment projects/programs, which PM has lead: 2
Interviews: 1st face-to-face 57:26, 2nd phone call 32:44, total: 1h 30min 10sec

PM2:

Age and gender: 31, female
Experience (years) in IT: 10
Experience (years) as an IT Project Manager: 5
N of projects/programs that has participated in: 140
N of projects/programs, which are of class: deployment: 140
N of deployment projects/programs, which PM has lead: 10
Interview: face-to-face 1h 19min 55sec

PM3:

Age and gender: 39, male
Experience (years) in IT: 15
Experience (years) as an IT Project Manager: 11
N of projects/programs that has participated in: 8-9
N of projects/programs, which are of class: deployment: 8-9
N of deployment projects/programs, which PM has lead: 6-7
Interviews: 1st face-to-face 32:21, 2nd phone call 23:32, total: 55min 53sec

PM4:

Age and gender: 34, male
Experience (years) in IT: 10
Experience (years) as an IT Project Manager: 3
N of projects/programs that has participated in: 20
N of projects/programs, which are of class: deployment: 10
N of deployment projects/programs, which PM has lead: 1
Interview: face-to-face 1h 11min 45sec

PM5:

Age and gender: 38, male
Experience (years) in IT: 13
Experience (years) as an IT Project Manager: 8
N of projects/programs that has participated in: 25
N of projects/programs, which are of class: deployment: 23
N of deployment projects/programs, which PM has lead: 15
Interview: face-to-face 1h 41min 26 sec

APPENDIX III – INTERVIEW FRAME

Basic information

- Age
- Experience with IT and projects
- Experience as IT project manager
- N of projects that has participated in some role…
- … of which N has been deployment project or programs …
- …of which N projects have been part of a program…
- …of which N has been lead (officially) by respondent
- Project/program size 1-10, 10-50, 50-100, >100
- Stakeholders 1-50, 50-100, 100-500, 500-1000, 1000-5000, 5000-10000, >10000
- Duration (month) 1-3, 3-6, 6-12, 12-18, 18-24, 24-30, 30-36, >36

Project history

- How many of projects or programs that respondent participated had been terminated or change presented had not been used afterwards…
- … of which N had been lead by respondent
- What have been the main causes
- How many of projects or programs >30% budget or time…
- …and for reasons

Scope management

- What does scope management mean in context of IT deployment
- What is its relevance for deployment
- Does the respondent use any recognized tool (frame, model, best practices, BoK) for managing scope in his daily work …

- …and if yes, how often
- Has scope management been recognized in enough relevance and importance in working environments respondent had worked in…
- …what would raising the awareness bring
- Does scope management have different role in creation projects than in deployment …
- …and if yes, then how is it different
- Has respondents recognition and conceptual model of scope management changed over the professional years and if so, then how

Change management

- what does organizational change management mean in context of IT deployment
- What is its relevance for deployment
- Does the respondent use any recognized tool (frame, model, best practices, BoK) for managing organizational change in his daily work …
- …and if yes, how often
- Has change management been recognized in enough relevance and importance in working environments respondent had worked in…
- …what would raising the awareness bring
- Does change management have different role in creation projects than in deployment …
- …and if yes, then how is it different
- Has respondents recognition and conceptual model of change management changed over the professional years and if so, then how

PMI areas of knowledge

- Rank the areas from most important to least important in context of deployment (personal feel)

Success criteria

- Project vs. product success, which one is more important and why
- And for the organization which ordered the project (change) and why
- What about IT organization and business organization is the importance different and if, then why
- Have the success criteria commonly been agreed prior to project start…
- …are they followed up during the project and adjusted, if not, then why…
- …and would agreeing criteria beforehand contribute to interpreting project success later
- Do deployment projects commonly get forgiven for overruns of cost and time and if yes, then why
- Listing five most important success criteria for deployment (personal feel)
- Have concepts and priorization over the different criteria changed during the professional years
- Models of uncertainty: which one A or B better describes deployment projects and why

Key questions

- Which one respondent sees more important for deployment considering the discussed success criteria and targets, scope or change management and why

- Would a separate role of change manager (explain the role) enhance the deployment and raise success percentage and if yes, then how much and why

- Is there difference in the value of the change manager if role is fulfilled by internal person or external person and what is the difference

- Would the respondent be willing to see extra effort (outside normal allocation process) to locate such a role to your project and would the respondent be willing to justify additional costs of such a role to management

- Is any of the PMI areas of expertise more important than scope or change management

- Is there difference, if the discussed items are in context of a deployment phase in a project or a separate project