Success Factors of Organizational Change

in Software Process Improvement

Dirk Stelzer, Werner Mellis

Software Process Improvement and Practice, Volume 4, Issue 4

14.05.1999

Universitaet zu Koeln, Lehrstuhl fuer Wirtschaftsinformatik, Systementwicklung,
Albertus-Magnus-Platz, D-50932 Koeln, Germany
Phone: +49 221 470 5368; Fax: +49 221 470 5386; email: stelzer@informatik.uni-koeln.de
http://www.informatik.uni-koeln.de/winfo/prof.mellis

Success Factors of Organizational Change in Software Process Improvement

Summary

The management of organizational change is an essential element of successful software process improvement efforts. This paper describes ten factors that affect organizational change in software process improvement initiatives based on the Capability Maturity Model or the ISO 9000 quality standards. It also assesses the relative importance of these factors and compares the findings with the results of previous research into organizational change in software process improvement. The paper is based on an analysis of published experience reports and case studies of 56 software organizations that have implemented an ISO 9000 quality system or that have conducted a CMM-based process improvement initiative.

Key words: success factors, organizational change, software process improvement, CMM, ISO 9000, empirical study

1 Introduction

The term "software process improvement" denotes the "changes implemented to a software process that bring about improvements" [Olson et al. 1989]. ISO 9000-1 defines a process as "a set of interrelated resources and activities which transform inputs into outputs. ... Resources may include personnel, finance, facilities, equipment, techniques and methods" [ISO 9000-1 1994]. Correspondingly, a software process can be defined as "a set of activities, methods, practices, and transformations that people use to develop and maintain software and the associated products (e.g., project plans, design documents, code, test cases, and user manuals)" [Paulk et al. 1993a]. The intent of software process improvement is improving software product quality, increasing productivity, and reducing the cycle time for product development [Paulk et al. 1993b].

Thousands of software companies worldwide have established initiatives to improve software development performance. Various models to facilitate software process improvement are available including the Capability Maturity Model for software (CMM) [Paulk et al. 1995], the ISO 9000 quality standards [ISO 9000-1 1994], TickIT [DTI, BSI 1992], Trillium [Coallier 1995] and

BOOTSTRAP [Kuvaja, Bicego 1994]. North American companies seem to prefer the CMM. European software organizations tend to use the ISO 9000 to enhance their capabilities.

Empirical studies into the success of software process improvement initiatives report about remarkably positive results [Herbsleb et al. 1994; Loken, Skramstad 1995; Stelzer, Taube 1998]. However, most initiatives also have to handle several complex problems [Goldenson, Herbsleb 1995; Herbsleb et al. 1994; Krasner, Ziehe 1995; Loken, Skramstad 1995; Stelzer et al. 1996].

In a survey of 138 individuals in 56 software organizations that have conducted CMM based process improvement initiatives Goldenson and Herbsleb [Goldenson, Herbsleb 1995] found that 26 % of the respondents say that "nothing much has changed" since the appraisal. 49 % say that there "has been a lot of disillusionment over the lack of improvement". 72 % report that process improvement has often suffered due to time and resource limitations. 77 % say that process improvement has taken longer that expected, and 68 % say that it has cost more than expected. 67 % of the respondents say that they need more guidance about exactly how to implement successful process improvement programs.

In a study of 119 German software companies that have implemented an ISO 9000 based quality system 88 % of the respondents say that they consider the improvement initiative successful [Stelzer, Taube 1998]. 68 % claim that the improvement efforts had led to improved product quality and 59 % report about improved customer satisfaction. However, only 44 % say that schedule commitments and 29 % say that cost estimates have been met more often. Only 11 % report about an increased demand for their products and services. 40 % of the respondents say that they are not satisfied with the success of continuous improvement efforts in some parts of their organizations.

Several authors [Curtis 1997; Mattingly, Abreo 1997; Morrell, van Asseldonk 1997; Myers 1996; Quinn 1998; Siddall 1997; Stelzer et al. 1997] suggest that the problems software companies encounter while implementing process improvement initiatives might result from insufficient organizational change management. Therefore, various authors have stressed the importance of organizational change in software process improvement programs [Bottcher 1997; Curtis 1997; Humphrey 1989; Humphrey 1997; Jackson 1997; Mattingly, Abreo 1997; Morrell, van Asseldonk 1997; Myers 1996; Myers 1997; Quinn 1998; Siddall 1997; Stelzer et al. 1997]. Some authors [Coffman, Thompson 1997; Goldenson, Herbsleb 1995; Stelzer et al. 1997] indicate that software organizations usually underestimate the efforts needed to accomplish the change process. McGuire states that " ... even established and successful software development teams may experience uncertainty and disruption when beginning a software process improvement effort ..." [McGuire

1996a]. Nevertheless, organizational change management does not seem to be sufficiently accounted for, neither in the CMM [Curtis et al. 1995] nor in the ISO 9000 standards family [Stelzer et al. 1997]. The CMM tells software organizations "what" to improve. However, the CMM does not specify "how" to effectively implement the practices described in the CMM [Paulk 1998]. The same is true for the ISO 9000 family [ISO 9000-1 1994].

Practitioners that wish to implement process improvement initiatives need a thorough understanding of the factors that affect success and failure of improvement activities. Despite the growing interest in the improvement of software development, however, a profound knowledge of the enablers and inhibitors of software process improvement is still lacking [Herbsleb et al. 1994].

The objectives of this paper are (1) to describe factors that affect organizational change in software process improvement initiatives and (2) to assess their relative importance.

2 A Generic Model of Organizational Change in Software Process Improvement

Although the CMM and the ISO 9000 have various characteristics in common, the two models differ in several dimensions. This paper explores factors influencing organizational change in CMM- *and* ISO 9000-based improvement efforts. Thus, we need a model of organizational change in software process improvement that is independent of the particularities of the CMM and the ISO 9000.

The "Plan-Do-Check-Act" cycle described by Deming [Deming 1992] might be a candidate for such a generic model because the CMM as well as the ISO 9000 quality standards are based on this cycle. However, the issues of changing software processes seem to be underrepresented in this cycle: Only the "Do" element deals with changing the process.

We have therefore decided to choose a generic model of change management described, for example by Beckhard and Harris [Beckhard, Harris 1987] or Nadler and Tushman [Nadler, Tushman 1997].

On the most fundamental level of this model changing software processes consists of two phases: First, analyzing the process and second, changing the process. The result of the first phase is an envisioned (or planned) process, the result of the second phase is an improved (and a new) current process. Figure 1 shows the two phases of software process improvement and their results. [For a more detailed model see, for example, McFeeley1996.]

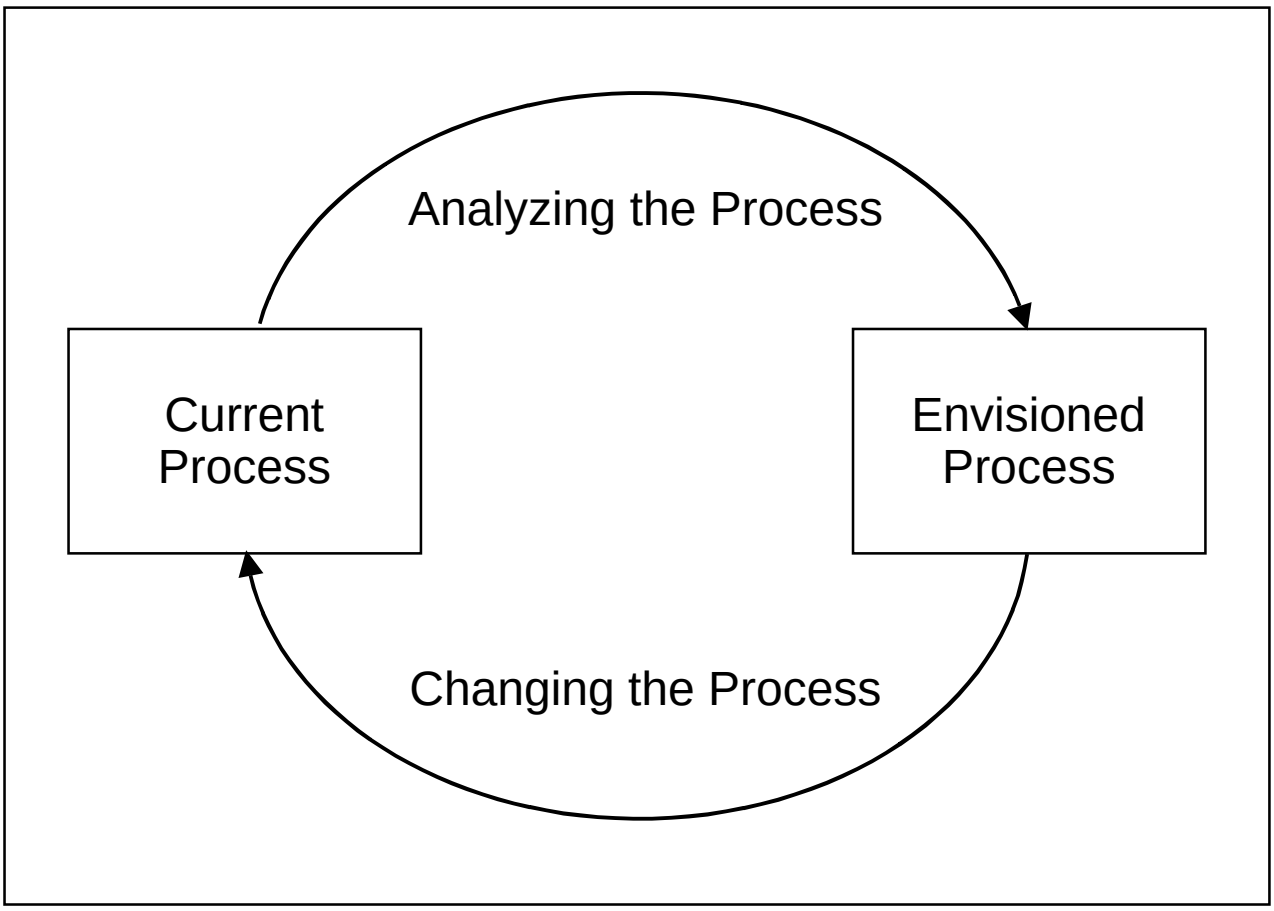

Figure 1: A Generic Model of Organizational Change in Software Process Improvement

The first phase of the model, analyzing the process, consists of two steps: identifying process issues and exploring improvement options.

Identifying process issues reveals strengths and weaknesses of current processes. In a CMM-context identifying process issues may be conducted as software process assessments or software capability evaluations [Paulk et al. 1993a]. In an ISO 9000 context strengths and weaknesses of current processes are identified with the help of quality audits [ISO 8402 1994]. [For a more detailed discussion of process assessments, capability evaluations, and process audits see, for example, Baumert 1994.]

Exploring improvement options aims at developing recommendations, setting priorities and establishing a detailed plan of what to improve in the software process. The current process being assessed or audited may be compared to the key process areas in the CMM or the quality elements in the ISO 9000. The comparison indicates improvement options. According to business objectives, management priorities, corporate culture, and other circumstances the organization sets priorities and designs a plan for improving the current software process.

The result of the first phase in an envisioned (or planned) process, a picture of how the improved software process might look like. However, the envisioned process is useless unless the improvement activities are actually implemented, accepted and executed by managers and employees.

Consequently, the second phase aims at deploying the recommendations to improve the software process. The major objective of this phase is to ensure that the envisioned process becomes the (new and improved) current process. This requires to manage the change of the software process. As already shown in chapter 1, previous research into software process improvement has shown that changing the software process seems to be more critical than the analytical tasks of software process improvement.

Continuous improvement means that the new, improved process becomes subject to the improvement cycle again.

3 Previous Research into Success Factors of Organizational Change

The ability to manage organizational change as a prerequisite for meeting competitive demands has been stressed by various authors [Conner 1995; Hardy, Schwartz 1996; Kanter 1994; LaMarsh 1995; Leonard-Barton 1995; Litwin et al. 1996; Munro-Faure, Munro-Faure 1996; Tushman, O'Reilly 1997].

Numerous studies have explored success factors of organizational change. For example, Kotter [Kotter 1995] analyzed the efforts of more than 100 companies that have attempted to make fundamental organizational changes in order to achieve competitive advantages. Beer, Eisenstat, and Spector [Beer et al. 1990] conducted a comprehensive study of 6 large companies where top management was attempting to revitalize the corporation. Miles, Coleman, and Creed [Miles et al. 1997] grouped various companies according to their success in corporate redesign to meet competitive demands.

These and other studies [see, for example, Decker, Belohlav 1997; Hawley 1995; Masters 1996; Schaffer, Thomson 1992] have identified several factors that influence the success of organizational change efforts. These factors include senior management commitment, setting and communicating objectives of the change efforts, involving technical staff and operating managers in the change process, establishing a sense of urgency, focus on solving concrete business problems, effective measurement techniques, reorganizing employee roles, responsibilities and relationships to address the organization's new structure, monitor and adjust the transition in response to problems in the

change process, removing obstacles to the change process, and anchoring changes in the organization’s culture.

Several authors emphasize that organizational change management is one of the key issues to achieve successful transitions of software processes. Cattaneo, Fuggetta, and Lavazza [Cattaneo et al. 1995] maintain that assessment is not the critical part of CMM-based software process improvement initiatives. However, they stress that introducing these changes into the organization is the challenge. Coffman and Thompson surveyed 41 US Air Force software units that had conducted software process improvement efforts from 1991 to 1996 [Coffman, Thompson 1997]. One of the key issues to help these software units achieve higher CMM levels is to provide support for change management. Craigmyle and Fletcher [Craigmyle, Fletcher 1993] report about the results of a software process improvement program based on the CMM in more than 20 Scottish software companies. Craigmyle and Fletcher highlight that "it is very important to manage changes in line with business activity" [Craigmyle, Fletcher 1993].

The Software Engineering Institute (SEI) has developed the IDEAL model [McFeeley 1996] based on the experience the SEI has gained while working with its clients. The objective of the IDEAL model is to communicate a path of actions that constitute a software process improvement initiative based on the CMM.

Astonishingly, so far only few studies into the success factors of organizational change in software process improvement have been undertaken.

Clark et al. analyzed the organizational change process within a large IS unit at Bell Atlantic [Clark et al. 1997]. The case study is based on first-hand experience of two senior managers leading the organizational transformation, a data collection over a period of 14 months that consisted of interviews with 27 Bell Atlantic employees and managers and a review of numerous internal archival documents. The authors maintain that during the change process various organizational parameters have to be transformed including processes, coordination mechanisms, roles and responsibilities, communication and interpersonal relationships among individuals, reward systems, and people skills.

McGarry et al. [McGarry et al. 1994] describe experiences gained during 17 years with process improvement efforts in the NASA Software Engineering Laboratory (SEL). Numerous software process change efforts have led to a variety of changes of organizational and software engineering practices at SEL. The authors identify the most significant process attributes that distinguish the SEL production environment from the environment before the improvement efforts were started:

Process change has been infused as a standard business practice, measurement is a common part of doing business, change is driven by product and process, not merely process alone, change is guided by development-project experience, and "people-oriented" technologies are emphasized rather than automation.

McGuire [McGuire 1996a; McGuire 1996b] has analyzed factors that affect the transition from a non-process driven software development environment to a process-focused environment under the CMM. McGuire surveyed a team of 64 software development professionals and asked them - among other topics - to evaluate the extent to which they observed various change management strategies being employed by the organization and their managers during the software process improvement initiative. McGuire comes to the conclusion that "change management strategies coupled with appropriate training and information sessions can have a direct effect on the rate of progress a team of software professionals makes in adopting a CMM-based software process improvement program" [McGuire 1996a].

Krasner and Ziehe [Krasner, Ziehe 1995] describe lessons learned at a software process improvement initiative of five US American software suppliers to the semiconductor industry. Krasner and Ziehe report the following major challenges on the road to improvement: staff shortages of software engineers and management positions, lack of commitment and continuity by the leaders responsible for software improvement, lack of definitive software quality goals, making software improvement sufficiently high priority, wavering or sporadic senior management sponsorship, lack of focus on change as a part of the engineering job, lack of will and discipline to change, and little progress in learning how to measure improvement [Krasner, Ziehe 1995]. On the basis of these experiences Krasner and Ziehe define several readiness criteria for software process improvement efforts: sponsorship for both TQM and software process improvement by senior management, an individual identified as the focal point or champion for software improvement, a management steering group that has a vision for software and owns responsibility for achieving that vision, a baseline of its current software process and performance, a quality improvement goal, business incentives for achieving it, and metrics for measuring progress to the goal, a budget for increasing staff skills in software engineering, management, and continuous improvement, one or more focused working groups, staffed and with charters, and a percent of time for software engineers and managers to work on improvement. Other readiness considerations relate to the organization’s culture, its interactions with its customers, and its willingness to contract a software process improvement service provider if there is lack of experience within the organization.

Goldenson and Herbsleb [Goldenson, Herbsleb 1995] conducted a survey of 138 individuals in 56 software organizations that had implemented CMM-based process improvement initiatives. 67 % of the respondents say that they need more guidance about exactly how to implement successful process improvement programs. Goldenson and Herbsleb identify several factors that distinguish successful from less successful software process improvement efforts. The most significant factors of successful improvement efforts include: Senior management monitoring of the improvement initiative, clear, compensated assignment of responsibilities for process improvement, involvement of respected people in process improvement, involvement of technical staff in the improvement effort, adequate amount of staff time and resources dedicated to process improvement since appraisals, and clearly stated and well understood process improvement goals. The most significant factors of less successful improvement efforts include: Organizational politics, "turf guarding", discouragement and cynicism from previous experience, and the feeling among the technical staff that process improvement gets in the way of their "real work".

The empirical studies quoted in the first paragraphs of this section cover change processes in various industries. However, they do not specifically analyze organizational transformations in software companies. The studies by Clarke et al., McGarry et al., and McGuire were conducted in software and information systems organizations. However, these studies cover only one company each. It is unclear whether the results may be generalized to other organizations [McGuire 1996a]. Apart from the surveys conducted by Goldenson and Herbsleb [Goldenson, Herbsleb 1995] and Krasner and Ziehe [Krasner, Ziehe 1995] there is no study investigating factors affecting the organizational change process in a variety of software process improvement initiatives.

The study of Goldenson and Herbsleb is the only study that provides quantitative indicators of the significance of the success factors of organizational change in process improvement efforts. However, this study covers CMM-based improvement efforts only. It is unclear whether the same factors are relevant for ISO 9000-based improvement efforts.

We will, therefore, explore success factors of organizational change in CMM- *and* in ISO 9000-based improvement efforts. Furthermore, we will assess whether there are any significant differences between factors affecting organizational change in a CMM- and in an ISO 9000-context.

4 Identification of Success Factors of Organizational Change in Software Process Improvement

Our research work is divided in two stages: In the first stage we conducted an explorative study of factors that potentially affect organizational change in software process improvement. In the second stage we thoroughly analyzed experience reports of 56 companies that had undertaken process improvement efforts in order to assess the significance of the factors identified in the first stage.

The first stage of our research is subdivided in the following activities:

- A survey of the literature exploring change management issues revealed factors that affect organizational change in various industrial sectors [Beer et al. 1990; Decker, Belohlav 1997; Hawley 1995; Masters 1996; Kotter 1995; Miles et al. 1997; Schaffer, Thomson 1992].
- Interviews with managers in 20 German software companies that had implemented ISO 9000-based software process improvement initiatives enhanced our understanding of factors influencing the success of organizational change in software process improvement initiatives.
- An analysis of experience reports and case studies from a total of 16 European software companies that had implemented ISO 9000-based quality systems [Stelzer et al. 1996] and an initial survey of case studies describing CMM-based improvement initiatives [Billings et al. 1994; Daskalantonakis 1994; Dion 1992; Herbsleb et al. 1994; Humphrey et al. 1991; Wohlwend, Rosenbaum 1994] showed that the factors revealed in the literature survey and the interviews were mentioned in the majority of the reports and case studies but to a differing degree: Management commitment and staff involvement, for example, seemed to be mentioned in more of the reports than other factors. This motivated us to conduct a more thorough analysis of the literature on organizational change in software process improvement.

On the basis of our activities in the first stage of our research we elaborated a synopsis and compiled a list of 10 factors that seemed to influence the success of organizational change in process improvement efforts based on the CMM and the ISO 9000 standards. Table 1 shows the factors. (The factors are listed in alphabetical order. They will be discussed in detail in the following section of this paper.) This list, however, does not give any information on the relative importance of the factors.

Table 1: Factors affecting organizational change in software process improvement efforts

Success Factor of Organizational Change	**Explanation**
Change agents and opinion leaders	Change agents initiate and support the improvement projects at the corporate level, opinion leaders at a local level.
Encouraging communication and collaboration	Degree to which communication efforts precede and accompany the improvement program (communication) and degree to which staff members from different teams and departments cooperate (collaboration).
Management commitment and support	Degree to which management at all organizational levels sponsor the change.
Managing the improvement project	Degree to which a process improvement initiative is effectively planned and controlled.
Providing enhanced understanding	Degree to which knowledge of current software processes and interrelated business activities is acquired and transferred throughout the organization.
Setting relevant and realistic objectives	Degree to which the improvement efforts attempt to contribute to the success of the organization (relevant) and degree to which the objectives may be achieved in the foreseeable future (realistic).
Stabilizing changed processes	Degree to which software processes are continually supported, maintained, and improved at a local level.
Staff involvement	Degree to which staff members participate in the improvement activities.
Tailoring improvement initiatives	Degree to which improvement efforts are adapted to the specific strengths and weaknesses of different teams and departments.
Unfreezing the organization	Degree to which the "inner resistance" of an organizational system to change is overcome.

The objective of the second stage is to explore to what extent the factors identified in the first stage influence organizational change in software process improvement initiatives and to identify the relative importance of each factor. The second stage of our research is subdivided in two activities. First, we analyzed published experience reports and case studies of 25 software organizations that had implemented an ISO 9000 quality system and sought certification. (In the remainder of the paper these reports will be called "ISO cases".) 12 of the organizations were located in the UK, eight in Germany, two in France, and one organization each in Austria, Greece, and the US. Table 2 shows the name and the location of the organizations and references to the published reports.

Table 2: Organizations covered in our study that have conducted ISO 9000-based improvement efforts

Organization	**Location**	**References**
ACT Financial Systems Ltd.	UK	Chambers 1994
Alcatel Telecom	F	Courtel 1996
ALLDATA	D	Kilberth 1997
Anonymous British Software Company	UK	Quinn 1996
Answers Software Service	UK	Walker 1994
AVX Ltd.	UK	Sweeney, Bustard 1997
BR Business Systems	UK	Havenhand 1996
Bull AG	D	Mosel 1996a, Mosel 1996b
Cap Gemini Sogeti	F	Sidi, White 1995
CMS (British Steel)	UK	Hepworth 1994
Danet-IS GmbH	D	Bulski, Martin-Engeln 1997, Klaus 1994
Dr. Materna GmbH	D	Steinke 1997
IBM Deutschland	D	Dette 1996
IDC-UK	UK	Robb 1994
INTRASOFT	GR	Frangos 1995
Logica	UK	Forrester 1996; Forrester, Dransfield 1994
Oracle	UK	Verbe, Robinson 1995
Praxis	UK	Hewson 1996
PSI AG	D	Warner 1997
SAP AG	D	Vering, Haentjes 1995, Dillinger 1994
Siemens AG (Austria)	A	Zopf 1994a, Zopf 1994b
Sybase	US	Macfarlane 1996
Tembit Software GmbH	D	Schroeder, Wilhelm 1996
Triad Special Systems Ltd.	UK	Fulton, Myers 1996
Unisys Systems and Technology Operations (STO)	UK	Clarke 1991

Second, we analyzed experience reports and case studies from 31 software organizations that had conducted CMM-based improvement efforts. (In the remainder of the paper these cases will be called "CMM cases".) The survey covers 19 organizations located in the United States, four British organizations, two Dutch organizations, and one organization each in Canada, France, Germany, Norway, Sweden, and Switzerland. Table 3 shows the name and the location of the organizations and references to the published reports.

Table 3: Organizations covered in our study that have conducted CMM-based improvement efforts

Organization	**Location**	**Reference**
ABB	CH	Menezes, Eschermann 1997
Advanced Information Services (AIS)	US	Seshagiri 1996
AIM Management Group	US	Zahniser, Rizzo 1995
Alcatel Telecom Norway	N	Halsey 1997
Alcatel-Telecom France	F	Courtel 1996
Allied Signal Aerospace	US	Buchman 1996
Applied Materials	US	Krasner, Scott 1996
Boeing Space Transportation Systems	US	Yamamura, Wigle 1997; Fowler 1997; Kness, Satake 1997; Wigle, Yamamura 1997; Vu 1997
Bristol & West Building Society, UK	UK	George 1996
Corning Inc.	US	Johnson 1994
Digital Equipment Corporation, Integrated Office Services Group	UK	Hellmann, Pilgrim 1997
Eastman Kodak Comp.	US	Wiegers 1996a; Wiegers 1996b
Ericsson Netherlands	NL	Keustermans 1996
Hewlett-Packard Software Engineering Systems Division	US	Lowe, Cox 1996
Hughes Aircraft	US	Humphrey et al. 1991; Snyder 1997
IBM Federal Systems Company / Space Shuttle Onboard Software project	US	Billings et al. 1994; Paulk et al. 1995
Lloyds Bank Plc, England	UK	Larner 1996; Larner 1995; Jackson 1996; Hookham 1997
Motorola's Cellular Infrastructure Group	US	Daskalantonakis 1994
Motorola's Government Electronics Division	US	Diaz, Sligo 1997
Odgen Air Logistics Center	US	Oestreich, Webb 1995
Oerlikon Aerospace	CAN	Laporte, Papiccio 1998
Oklahoma City Air Logistics Center	US	Butler 1997; Butler 1995; Herbsleb et al. 1994
Philips Business Electronics, NL	NL	de Jong 1996
PRC, Inc.	US	Hollenbach et al. 1997
Q-Labs	S	Ahlgren, Debou 1997
Raytheon	US	Dion 1992; Dion 1993; Haley 1996
Sacramento Air Logistics Center	US	Westaway 1995
Schlumberger	US	Wohlwend, Rosenbaum 1994
Siemens (Germany)	D	Paulish, Carleton 1994, Mehner, Voelker 1996, Voelker 1996
Texas Instruments DSEG	US	Benno, Frailey 1995
Thomson Training & Simulation, UK	UK	Boulton, Fishbourne 1996

The survey comprises case studies and experience reports that were published between 1991 and 1997. The majority of the texts were published 1995, 1996, or 1997. The case studies and experience reports cover process improvement initiatives in government agencies and organizations in various industries including defense, aeronautics and aviation, telecommunication, finance and electronic systems. The authors are mostly software quality or process specialists, some authors are members of senior management, and some of the case studies are co-authored by consultants.

All case studies and experience reports describe successful improvement efforts. However, on the road to success most organizations experienced several difficulties. The description of these problems and obstacles provide interesting insights into factors affecting organizational change in software process improvement efforts. Most of the CMM cases are more extensive than the ISO cases and provide more detailed information.

We documented which of the factors identified in stage one of our research are mentioned in which of the experience reports and case studies analyzed in stage two. Several case study authors use the same or similar phrases to describe the factors that affect change in their organizations as we do. Other case study authors use a different terminology. In these cases we had to interpret the texts. To eliminate potential bias in interpreting the texts, two authors analyzed each experience report independently of each other. In case of doubt we assumed that the case study authors did *not* intend to mention the success factor as we defined it in our study. The findings of our analysis are documented in Appendix 1.

Subsequently, we examined in how many of the reports each of the factors is mentioned and we computed a percentage. A percentage of x for factor y means that factor y is mentioned in x % of the reports. Finally, we deduced a ranking that shows the relative importance of the factors. The factor with the highest percentage is ranked first, and the factor with the lowest percentage is ranked last. We conducted this analysis (a) separately for the ISO cases, (b) separately for the CMM cases, and (c) for all cases together. Table 4 shows the relative importance and the ranking of the factors. (The factors are listed according to their rank for all cases.)

Table 4: Relative importance and ranking of the factors affecting organizational change in software process improvement

Success Factor	ISO cases (n = 25)		CMM cases (n = 31)		All cases (n = 56)	
	Percentage	Rank	Percentage	Rank	Percentage	Rank
Management commitment and support	84%	1	97%	1	91%	1
Staff involvement	84%	1	84%	8	84%	2
Providing enhanced understanding	72%	3	87%	6	80%	3
Tailoring improvement initiatives	68%	4	90%	3	80%	3
Managing the improvement project	56%	6	94%	2	77%	5
Change agents and opinion leaders	52%	7	90%	3	73%	6
Stabilizing changed processes	52%	7	90%	3	73%	6
Encouraging communication and collaboration	64%	5	74%	9	70%	8
Setting relevant and realistic objectives	44%	9	87%	6	68%	9
Unfreezing the organization	24%	10	52%	10	39%	10

5 Detailed Description of the Success Factors

In the following sections the factors affecting organizational change in software process improvement are discussed in detail.

5.1 Management Commitment and Support

Management commitment and support is the degree to which management at all organizational levels sponsor the change. Management commitment and support is the factor that is mentioned most often. 84 % of the ISO cases and 97 % of the CMM cases notice that management commitment and support is essential for supporting organizational change in software process improvement initiatives.

Software process improvement is not feasible without investing time, money, and effort. It requires modifications of day-to-day management and of staff members' habits. Changing habits often causes resistance of individuals or groups. The necessary investment of time and money and the need to overcome staff resistance are potential impediments to software process improvement initiatives. These obstacles cannot be overcome without management support. Wohlwend and Rosenbaum say that the "key behind the success of the improvement effort lies with the management of the organization" [Wohlwend, Rosenbaum 1994].

Senior management often agrees to improvement initiatives without completely realizing the investment required for the effort. In some organizations top managers assume that the initiative will occur without modification of other commitments. As a consequence, middle management often find themselves caught in a trap. On the one hand they have to modify software processes, on the other hand they have to accomplish project objectives without affecting deadlines, milestones, budget restrictions, and functional requirements [Wohlwend, Rosenbaum 1994]. Consequently, middle management often does not support software process improvement. However, progress is only possible if managers at all levels of the organization support the initiative. Senior managers must commit to developing, implementing and financing appropriate action plans in response to the assessment recommendations. This requires that senior management receives an accurate picture of the time and resources needed to conduct a process improvement initiative.

Senior managers should actively participate in assessment meetings and improvement workshops to demonstrate the importance of the initiative. Active participation and visible support of senior management may give the necessary momentum to the initiative. This positively influences the success of the improvement efforts. Diaz and Sligo emphasize "... senior management sponsorship proved critical to the success of the process improvement efforts. This means not only taking an active interest in the progress of various process improvement initiatives, but also providing funding and time to do the work, and rewarding those who contributed" [Diaz, Sligo 1997].

5.2 Staff Involvement

Staff involvement is the degree to which staff members participate in the improvement activities. Staff involvement is addressed in 84 % of the ISO cases and in 84 % of the CMM cases.

Staff participation is essential in improvement activities because employees must adopt process innovations in their day-to-day activities. If staff members do not buy into the proposed changes the

improvement initiative is useless. Improvement activities promoted only by a department or group external to the people involved in the process is usually not well accepted.

Some software organizations have suffered from a schism between development projects and process improvement activities [Oestreich, Webb 1995]. Staff members in these companies conceived software development and process improvement as separate activities. They did not understand how software process improvement might support their daily work. As a consequence, they complied to the assessment findings only reluctantly and did not commit themselves to the proposed changes. Software developers tended "to get back to real work" [Oestreich, Webb 1995] after the formal improvement efforts had been completed. It was an immense challenge for these organizations to keep the improvement initiative in motion.

Staff members should be involved in the improvement initiative because they have detailed knowledge and first hand experience of strengths and weaknesses of the current processes. Dion emphasizes that "using the skills and experiences of so many project personnel ... guarantees that the resulting process is a consensus that reflects the practical considerations of diverse projects" [Dion 1993]. Software engineers who contribute to the improvement buy into the change and adopt innovations more readily. To ensure grass-roots involvement successful improvement initiatives have established local process teams, special interest groups, training schemes, and forums for the exchange of ideas and for coordinating efforts among project teams [Daskalantonakis 1994; Seshagiri 1996].

5.3 Providing Enhanced Understanding

Providing enhanced understanding to managers and staff members comprises acquiring and transferring knowledge of current software processes and interrelated business activities and objectives. Providing enhanced understanding is mentioned in 72 % of the ISO cases and in 87 % of the CMM cases.

Software development interacts with many other tasks of an organization. Thus, software process improvement cannot be separated from other parts of the business. Wohlwend and Rosenbaum state that "the improvement activities must involve the other parts of the business with which software interacts, namely marketing, hardware development, sales, manufacturing, etc." [Wohlwend, Rosenbaum 1994]. Successful improvement efforts require a common understanding of the organization's mission and vision. Consequently, software process improvement can only be conducted successfully if managers and staff members have a thorough understanding of the entire

business. Managers usually have a general idea of the software process, but they do not have complete understanding of essential details. Employees often do not understand how their work contributes to the corporate mission and vision. Daily routine often prevents them from reflecting on their work from a general business perspective.

Successful improvement initiatives give practitioners the opportunity to better understand the business of their organization. As Daskalantonakis puts it, "this increases their understanding of topics in which they may not have been involved in the past... " [Daskalantonakis 1994]. Documenting current software processes, and conducting internal audits and improvement workshops contributes to acquiring relevant knowledge. Guidelines, checklists, and training schemes help to transfer important information and communicate process innovations to all relevant members of the organization. The process of acquiring, documenting, and distributing knowledge helps staff members to perceive weaknesses and to understand why improvement efforts are useful. The enhanced understanding encourages software engineers to buy into process improvement.

5.4 Tailoring Improvement Initiatives

Tailoring improvement initiatives means adapting improvement efforts to the specific strengths and weaknesses of different teams and departments. The need for tailoring improvement initiatives is emphasized in 68 % of the ISO cases and in 90 % of the CMM cases.

Tailoring increases the compatibility of improvement plans with the existing values, past experiences, and needs of potential adopters. In the beginning of the initiatives, some companies used a standardized and centralized approach. They did not tailor the improvement efforts to particular needs of different sites and teams. For example, some software organizations that implemented ISO 9000 quality systems stipulated quality manuals and process documents for the entire organization. Usually, staff members refused to work with these documents. A considerable number of the companies using a standardized and centralized approach had to repeat their efforts and revise the planning, design and implementation of the process improvement initiative.

Many problems of process improvement concern seemingly minor details. As Humphrey puts it, "it is such details, however, that make the difference between an annoying and inconvenient process and a comfortable and efficient one” [Humphrey 1995]. Tailoring helps to address these details and to fight the problems by implementing incremental changes. These changes often have a significant impact on the improvement initiative. Software engineers begin to see the benefits of the transition,

they realize that the improvement efforts help them with their daily work and, as a consequence, they commit themselves to the improvement program.

Successful improvement initiatives focus on specific strengths and weaknesses of different organizational entities. They encourage local managers and engineers to conduct self-assessments and to create their own action plans for improvements. "This ensures grass-roots involvement in the process and institutionalization of improvement" [Daskalantonakis 1994]. It also helps to fight specific weaknesses and to expand on individual strengths that promise the greatest benefits.

Process improvement activities must clearly and continually demonstrate business benefits to projects. Tailoring helps to implement a process improvement infrastructure that responds to the true needs of the organization. Thus, tailoring contributes to overcoming the potential schism between software development activities and process improvement efforts.

5.5 Managing the Improvement Project

Managing the improvement project means that software process improvement is effectively planned and controlled. Managing the improvement project is addressed in 56 % of the ISO cases and in 94 % of the CMM cases.

Process improvement without project management leads to ad hoc, often inefficient, and sometimes chaotic practices. In the course of the improvement activities several organizations realized that the improvement teams continued to act in a way that resembled CMM maturity level 1 while they were trying to convince the organization that they must achieve level 2 or 3. Westaway, for example, admits "I was also distressed to see the division chief and the project leader continue to act in a Level 1 way while they were telling the organization that we must become Level 2" [Westaway 1995]. Often, the improvement projects had neither specified requirements, nor had they elaborated a formal project plan, defined milestones, or outlined a schedule. Areas of responsibility were not accurately determined, and the initiative was lacking effective interfaces between central process groups and local development teams.

Successful improvement initiatives set up and ran the process improvement project like a software development project. They used existing project management standards, analyzed requirements, defined explicit objectives, established milestones, selected measures for success, monitored progress, and calculated cost benefit ratios to ensure that the effort was really paying off. Larner recommends: "Set up and run each process improvement project like a software development project, make the team use the existing project management standards" [Larner 1995].

By understanding the dynamics of change, and carefully planning its introduction, management can minimize stress for everyone involved. This usually implies that the software process improvement efforts are not immediately introduced at the corporate level. Several successful improvement efforts were tested at a local or site level. This provided the opportunity to improve the improvement activities before the initiatives were extended to the entire organization.

5.6 Change Agents and Opinion Leaders

A significant lesson that the members of the SPI team at Schlumberger learned was that they were "primarily acting as change agents and secondarily as software technologists" [Wohlwend, Rosenbaum 1994]. Change agents initiate and support the improvement projects at the corporate level. They are individuals or teams external to the process that is to be improved. Opinion leaders are competent individuals responsible for initiating, guiding, and supporting the improvement at a local level. They are well respected members of the social system that is to be changed. The importance of change agents and opinion leaders is emphasized in 52 % of the ISO cases and in 90 % of the CMM cases.

In an ISO 9000 context quality managers regularly play the role of change agents. In CMM-based improvement initiatives software engineering process groups often act as change agents. Usually, they initiate the improvement projects, request resources, encourage local improvement efforts, and establish interfaces and communication channels between various groups. They also provide technical support and feedback, publish successes, help to overcome difficulties, and keep staff members aware of the improvement efforts. As Buchman puts it, "the Center for Process Improvement is an agent of change through technology transfer" [Buchman 1996].

Experienced project managers, proficient software engineers, or members of local improvement groups may be opinion leaders in a software process improvement initiative. They often act as advisors, advocates, and communication liaisons. Sometimes they must be able to implement the improvement activities through rough opposition [Larner 1995]. Opinion leaders are indispensable for overcoming the potential schism between software development activities and process improvement efforts. They help to tailor the improvement suggestions to the needs of different project teams and organizational departments.

Several software organizations attempt to eventually disband change agents and devolve responsibility for process improvement to practitioners at a local level. However, at the beginning of

a software process improvement initiative change agents and opinion leaders are indispensable for ensuring that the efforts run smoothly.

[For a more detailed description of the importance of change agents and opinion leaders see, for example, Daskalantonakis 1994; Mattingly, Abreo 1997; Myers 1996; Myers 1997; and Powel 1997]

5.7 Stabilizing Changed Processes

Stabilizing changed processes means continually supporting maintenance and improvement of software processes at a local level. This factor is mentioned in 52 % of the ISO cases and in 90 % of the CMM cases.

Stabilizing changed processes is necessary when permanency of the desired changes is included in the objectives of the initiative. It prevents that an improved software process slides back to the old level. Software process improvement efforts do not always have long lasting effects. On the contrary, benefits are often short-lived. Within a short time, group conduct often slides back to the old habits. Humphrey [Humphrey 1989] calls this phenomenon in a similar context "software process entropy". Software process entropy results from the fact that change is a dynamic process.

Staff members adopting a new process need continuous feedback, motivation, recognition, and reinforcement to stay involved in the improvement effort [Wohlwend, Rosenbaum 1994]. They also need guidance and support to overcome initial problems and difficulties [Paulish, Carleton 1994]. Misunderstandings have to be clarified. Weaknesses must be eliminated to ensure smooth functioning of new processes. To prevent software process entropy successful improvement initiatives provide comprehensive support and encourage staff members to practice the new procedures.

Senior management, change agents and opinion leaders have positively influenced the stabilization of changed processes in successful initiatives. Senior management repeatedly reaffirmed their commitment to the ultimate benefits gained from the change. Opinion leaders communicated problems and benefits to their managers and to software process improvement specialists. They helped to overcome difficulties that software developers perceived when adopting changed processes. Change agents provided continuous feedback and motivation.

5.8 Encouraging Communication and Collaboration

Communication indicates that a strong communication effort should precede and accompany the formal implementation of the improvement program. Collaboration denotes the cooperation of staff members from different teams and departments. Encouraging communication and collaboration is addressed in 64 % of the ISO cases and in 74 % of the CMM cases.

Software process improvement initiatives are often accompanied by rumors, fears, and resistance from staff members. Intensive communication helps to rectify rumors, to preclude misunderstandings, to overcome resistance, and to dispel software engineers' fears.

Successful process improvement initiatives have also encouraged communication among staff members [Buchman 1996; de Jong 1996]. This was achieved, for example, by conducting internal audits and process workshops. Audits and workshops provide an opportunity to discuss strengths and weaknesses of the process initiative, to submit proposals for improving the efforts, and to complain about disadvantages and drawbacks.

Many teams and divisions in software companies suffer from inadequate cooperation with other groups. Software quality-assurance teams, for example, are often not adequately integrated in the process of software development. Staff members in most companies must learn to pull together as a team. Therefore, successful improvement activities emphasize collaboration. Collaborative projects include joint process descriptions, workshops, and special interest groups. Joint activities help staff members discover unexpected similarities in products and processes. Successful improvement initiatives have established effective interfaces of various teams. This helps to exchange experiences of teams doing similar work in different parts of the company. Close cooperation of business units provides natural feedback loops, enhances staff members' understanding and knowledge, encourages people to exploit synergy, and consequently improves productivity and quality. Allied Signal, for example, "has gained significant software process maturity across multiple sites in a short time by collaborating and cooperating in the sharing of tools, processes and expertise" [Buchman 1996].

Intensive communication and collaboration help to create a coherent organizational culture that is necessary for achieving substantial improvements.

5.9 Setting Relevant and Realistic Objectives

Setting relevant objectives means that the improvement efforts attempt to contribute to the success of the organization. Setting realistic objectives means that the goals may be achieved in the

foreseeable future and with a reasonable amount of resources. Setting relevant and realistic objectives is addressed in 44 % of the ISO cases and in 87 % of the CMM cases.

It is essential that staff members understand the relationship between the objectives of software process improvement and revenues, cash flow, or other business results. Mere conformance to a standard, attaining certification, or reaching a CMM level usually is not a relevant goal for staff members. Implementing measures just for the sake of the CMM or the ISO 9000, and not for the sake of quality and productivity, imposes extra burdens on the project teams. Diaz and Sligo recommend: "Emphasize productivity, quality, and cycle time. Avoid process for its own sake" [Diaz, Sligo 1997].

Setting realistic goals means that the improvement objectives should not be too ambitious. For most software companies, for example, it is not realistic to reach CMM level 3 within one year. Managers that set excessive goals usually discourage their subordinates.

Setting relevant and realistic objectives implies to carefully prioritize assessment and improvement areas. Usually, neither all key process areas of a CMM maturity level nor all ISO 9000 quality elements are equally important for the success of the organization. Implementing key process areas or quality elements without prioritization often leads to bureaucracy and excessive efforts.

Some software organizations started the improvement initiatives without defining relevant and realistic objectives. They established long term and fuzzy goals, for example, "implementing a Total Quality Management culture" or "providing superior quality to customers" without further specifying the goals. This did neither motivate interest of staff members nor were the objectives helpful to demonstrate success of the improvement efforts.

The real test of the improvement objectives is the degree to which everyone can make the translation from top management goals to the goals that each person is being asked to achieve. The objectives must be decomposed to specific measures for project managers and programmers. In this way everyone can see how individual and group efforts relate to the organization's success.

Setting relevant and realistic objectives contributes to the clarity of expectations, the degree to which the expected results are shared across all levels of the organization. This helps to direct the efforts towards common objectives, to focus energy, and to motivate people.

5.10 Unfreezing the Organization

Lewin [Lewin 1958] has introduced the importance of "unfreezing the organization" before substantial improvements can be achieved. He emphasizes that social processes usually have an

"inner resistance" to change. To overcome this resistance an additional force is required, a force sufficient to break the habit and to unfreeze the custom. Astonishingly, unfreezing the organization is mentioned only in 24 % of the ISO cases and in 52 % of the CMM cases.

In software companies that have successfully conducted improvement initiatives several factors contributed to unfreezing the organization. First, some employees understood the need for improvement efforts. They realized deficiencies in current processes and they accepted the need for change. In other companies employees had not noticed any problems. They did not accept the need for change, and they were reluctant to participate in the improvement activities [Larner 1995]. Of course this hindered the success of the CMM and ISO 9000 projects. Second, when management committed themselves to the software process efforts they provided momentum to the initiative and created a realistic opportunity to fight the deficiencies. Diaz and Sligo mention: "Staff skepticism can also be an obstacle. Before they buy into a new SPI initiative, most software engineers will wait to see if it truly has management support and staying power" [Diaz, Sligo 1997]. Third, "competitive pressure to improve quality" [Johnson 1994] and "interorganizational competition" [Wohlwend, Rosenbaum 1994] created an awareness for the need to change and helped to unfreeze the organization.

6 Discussion

Our research is based on case studies and experience reports published by managers and process specialists of software organizations. Of course these sources primarily reflect the personal views of the authors of the texts. However, most of the success factors are mentioned in the majority of the papers that we have analyzed. This demonstrates that the factors described in our study do not only represent the personal view of some individuals. However, the factors seem to accurately reflect the state of the practice of organizational change in software process improvement.

A comparison of the success factors identified in our study with the factors discussed in studies exploring transition processes in other industries [see section 3 of this paper] shows that the success factors of organizational change are similar in all industries.

In the following section we will discuss the findings of our study and compare the results with the findings of Goldenson’s and Herbsleb’s survey [Goldenson, Herbsleb 1995].

In the CMM cases 8 of the factors are mentioned in more than 75 % of the reports and all factors are mentioned in more than 50 % of the reports. In the ISO cases only 2 factors are mentioned in more than 75 % of the reports and 8 factors are mentioned in more than 50 % of the reports. The most

plausible explanation for this difference is that the ISO cases are not as lengthy and detailed as the CMM cases. However, one might also presume that organizations implementing ISO 9000-based quality systems on average encounter fewer problems than organizations that conduct CMM-based improvement efforts. As the key process areas of the CMM, especially on maturity level 3 and above, tend to be more challenging than the requirements of the ISO 9000, this might also be a plausible explanation.

Management commitment and support seems to be the most important success factor of organizational change in software process improvement efforts. This factor is emphasized in 84 % of the ISO cases and in 97 % of the CMM cases. This finding confirms the results of the survey conducted by Goldenson and Herbsleb who found that 100 % of the organizations that report marked success of process improvement also report that their managers actively monitor progress of the efforts, and that such management commitment is considerably less in organizations with less successful improvement efforts.

Staff involvement is ranked second in our study. It is mentioned in 84 % of the ISO and also in 84 % of the CMM cases. Goldenson and Herbsleb found that more than 90 % of the organizations that report marked success of process improvement also report that technical staff was involved in the process improvement initiative, and that involvement of technical staff is considerably less in organizations with less successful improvement efforts.

The factors providing enhanced understanding, tailoring improvement initiatives, managing the improvement project, change agents and opinion leaders, stabilizing changed processes, and encouraging communication and collaboration are mentioned in between 52 and 72 % of the ISO cases and in between 74 and 94 % of the CMM cases.

Providing enhanced understanding to managers and staff members is not explicitly addressed in Goldenson’s and Herbsleb’s survey. However, they found that organizations that claim greater success of the improvement efforts are more likely to report that the amount of staff time dedicated to process improvement has been good or excellent. Presumably, the amount of staff time invested in process improvement is interrelated with staff members’ understanding of current software processes, business activities, and objectives. In contrast to our findings, Goldenson and Herbsleb found that detailed understanding of the technical work by senior and middle management does not have a consistent impact on the success or failure of the software process improvement efforts. However, they also found that senior management oversight is quite important.

Tailoring improvement initiatives is also not addressed in the survey of Goldenson and Herbsleb. However, they found that the feeling among technical staff that process improvement gets in the way of their "real" work is associated with less successful process improvement efforts. If process improvement initiatives are not tailored to the strengths and weaknesses of different organizational units staff members might get the impression that process improvement hinders them to do their "real" work. Goldenson and Herbsleb also found that organizational politics and "turf guarding" is common in organizations that report little or no success in software process improvement activities. Organizational politics and "turf guarding" will usually impede adapting improvement efforts to the particularities of different teams and departments. It will also hinder the compatibility of improvement plans with the existing values, past experiences, and needs of different organizational units.

The importance of managing the improvement project is also highlighted in Goldenson’s and Herbsleb’s survey. They found that those organizations that report more success in their improvement efforts are also much more likely to state that in their organizations there is clear, compensated assignment of responsibilities for process improvement. As already mentioned before, they also found that in all successful improvement efforts senior managers actively monitor the progress of process improvement.

Goldenson and Herbsleb found that those organizations that claim greater success in software process improvement are more likely to report that the people involved in the improvement efforts are well respected in their organizations. Presumable these well respected people act as opinion leaders. 52 % of the ISO cases and even 90 % of the CMM cases covered in our study emphasize the importance of opinion leaders and change agents. The role of change agents is not explicitly addressed by Goldenson and Herbsleb. However, they emphasize the importance of senior management commitment, support, and monitoring for the success of process improvement. Senior managers that actively participate in improvement efforts may act as change agents.

Goldenson and Herbsleb found that 26 % of the respondents say that "nothing much has changed" and 49 % say that there "has been a lot of disillusionment over the lack of improvement". This indicates that several organizations encounter difficulties in stabilizing changed processes at a local level. 52 % of the ISO cases and 90 % of the CMM cases in our study highlight the need to stabilize changed processes.

Encouraging communication and collaboration is mentioned in almost two thirds (64 %) of the ISO cases and just short of three quarters (74 %) of the CMM cases. Goldenson and Herbsleb found that

"turf guarding" and organizational politics are associated with less successful improvement efforts. "Turf guarding" and organizational politics will usually impede efficient communication and collaboration among staff members from different teams and departments.

Setting relevant and realistic objectives is mentioned in 87 % of the CMM cases. This corresponds with Goldenson's and Herbsleb's finding that organizations that claim greater success in process improvement efforts are more likely to report that process improvement goals are clearly stated and well understood in their organizations. It also corresponds with the fact that improvement objectives that are too ambitious are associated with less successful improvement initiatives. Astonishingly, in our study only 44 % of the ISO cases mention this factor. One might assume that the low percentage is due to the fact that setting quality objectives is an explicit requirement of the ISO 9000 standards [ISO 9001 1994; ISO 9004-1 1994], and that most case study authors consider setting relevant and realistic objectives to be a matter of course.

Unfreezing the organization is mentioned only in 52 % of the CMM cases and even only in 24 % of the ISO cases. This indicates that unfreezing is not as important for organizational change in software organization as it obviously is in other sectors of the economy. However, one might also suppose that most employees in software organizations have understood the need for substantial changes and that the inner resistance to change is not as high as it is in other organizations.

The results of our study are also comparable to the findings of the research conducted by McGuire [McGuire 1996a; McGuire 1996b] and by Krasner and Ziehe [Krasner, Ziehe 1995]. The fact that research work of other authors using different empirical techniques produced similar results shows that our findings are likely to be valid and reliable.

7 Conclusion

Our research reveals that the factors affecting organizational change in an ISO 9000-context are the same factors that affect change in a CMM-context. The quantitative indicators provided in our study show that there are only minor differences in the significance of success factors in CMM-based improvement initiatives compared to the significance of success factors in ISO 9000-based improvement efforts.

The findings of our study indicate factors that software organizations should consider when they attempt to conduct successful software process improvement initiatives. We believe the success factors discussed here give many interesting insights that will be relevant and applicable to organizational change in most software organizations. If the success factors are implemented they

facilitate the success of improvement initiatives. If they are not implemented - or not implemented correctly - this makes process improvement difficult to achieve, or may even cause failure of the initiative.

In terms of our generic model of organizational change in software process improvement [see section 2] all factors are aspects of the question of *how* improvement activities can be deployed. In other words, changing the software process seems to be at least as difficult as analyzing the process. The fact that an organization has designed an appropriate envisioned process does not necessarily mean that this organization will also be able to improve its software process.

At first glance, the factors discussed in this paper may seem obvious. However, when one looks at the case studies and experience reports a second time it becomes clear that the factors are mostly described as *lessons learned*. This shows that these organizations have obviously not paid enough attention to the implementation of the factors at the beginning of the initiative. Most organizations seem to take the factors for granted. As a result, they do not seem to pay sufficient attention to the management of change in software process improvement. Presumably, this is the reason that makes the factors so critical.

The management of organizational change is obviously not sufficiently accounted for in the CMM and in the ISO 9000 standards. Neither the CMM nor the ISO 9000 family seem to provide adequate support to implement the success factors discussed in this paper.

The Software Engineering Institute has already published IDEAL [McFeeley 1996], a model that supports organizational change in software process improvement initiatives. Astonishingly, there is nothing comparable for ISO 9000-based improvement efforts. Furthermore, IDEAL does not seem to be well known in Europe. Change management should therefore be a central element of future versions of the ISO 9000 standards and other models supporting software process improvement. At least, these models should emphasize the necessity of organizational change management.

8 References

Ahlgren, Magnus; Debou, Christophe 1997. Effective Implementation of CMM Levels 2 & 3. In: Proceedings of the Second Annual European Software Engineering Process Group Conference 1997, Amsterdam, T102c 1-47.

Baumert, John H. 1994. Process Assessment with a Project Focus. *IEEE Software*, March, 89-91.

Beckhard, Richard; Harris, Reuben T. 1987. *Organizational Transitions. Managing Complex Change* (2nd ed.). Addison-Wesley, Reading, Mass.

Beer, Michael; Eisenstat, Russell A.; Spector, Bert 1990. Why Change Programs Don't Produce Change. *Harvard Business Review*, 68, No. 6, 158-166.

Benno, Steve; Frailey, Dennis 1995. Software Process Improvement in DSEG 1989-1995. *Texas Instruments Technical Journal*, 12, No. 2, 20-28.

Billings, Cyndy; Clifton, Jeanie; Kolkhorst, Barbara; Lee, Earl; Wingert, William B. 1994. Journey to a mature software process. *IBM Systems Journal*, 33, No. 1, 46-61.

Bottcher, Peter 1997. Total Quality Management and the Capability Maturity Model (CMM): What's the difference? In: Proceedings of the IRIS 20 conference, Hanko, Norwegen.

Boulton, Jim; Fishbourne, Carol 1996. SPI Lessons Learnt from Getting Started. In: Proceedings of the First Annual European Software Engineering Process Group Conference 1996, Amsterdam, C412 P1-P9 S1-S11.

Buchman, Caroline D. 1996. Software Process Improvement at Allied Signal Aerospace. In: Proceedings of the Twenty-Ninth Annual Hawaii International Conference on System Sciences, Volume 1. Software Technology and Architecture, Washington - Brussels - Tokyo, 673-680.

Bulski; G.; Martin-Engeln, H. 1997. Erfahrungen und Erfolge in der SW-Projektabwicklung nach 4 Jahren DIN ISO 9001 Zertifizierung. In: H. J. Scheibl, editor, Technische Akademie Esslingen - Software-Entwicklung - Methoden, Werkzeuge, Erfahrungen '97, Ostfildern, 403-406.

Butler, Kelley 1995. The Economic Benefits of Software Process Improvement. *Crosstalk - The Journal of Defense Software Engineering*, 8, No. 7, 1-5.

Butler, Kelley 1997. Process Lessons Learned While Reaching Level 4. *Crosstalk - The Journal of Defense Software Engineering*, 10, No. 5, 1-6.

Cattaneo, Fabiano; Fuggetta, Alfonso; Lavazza, Luigi 1995. An Experience in Process Assessment. In: Proceedings of the 17th International Conference on Software Engineering, New York, 115-121.

Chambers, H. 1994. The implementation and maintenance of a quality management system. In: M. Ross et al., editors, Software Quality Management II, Vol. 1: Managing Quality Systems, Southampton - Boston, 19-33.

Clark, Charles E.; Cavanaugh, Nancy C.; Brown, Carol V.; Sambamurthy, V. 1997. Building Change-Readiness Capabilities in the IS Organization: Insights From the Bell Atlantic Experience. *MIS Quarterly*, 21, No. 4, 425-455.
Clarke, A. 1991. Persuading the Staff or ISO9001 without Tantrums. *SQM*, No. 9, 1-5.

Coallier, Francois 1995. TRILLIUM: A Model for the Assessment of Telecom Product Development & Support Capability. *Software Process Newsletter*. No. 2, 3-8.

Coffman, Aaron; Thompson, Kent 1997. Air Force Software Process Improvement Report. *Crosstalk - The Journal of Defense Software Engineering*, 10, No. 1, 1997

Conner, Daryl R. 1995. *Managing at the Speed of Change. How Resilient Managers Succeed and Prosper Where Others Fail*, New York.

Courtel, Daniel 1996. Continuous Quality Improvement in Telecommunications Software Development. In: Proceedings of the First Annual European Software Engineering Process Group Conference 1996, Amsterdam, C309 P1-P9 S1-S11.

Craigmyle, Mac; Fletcher, Irwin 1993. Improving IT effectiveness through software process assesment. *Software Quality Journal*, 2, No. 4, 257-264.

Curtis, Bill 1997. Models of SPI: Getting Beyond Case Studies. In: Proceedings of the Second Annual European Software Engineering Process Group Conference 1997, Amsterdam, C303 1-10.

Curtis, Bill; Hefley, William E.; Miller, Sally 1995. *Overview of the People Capability Maturity Model. CMU / SEI-95-MM-01*. Pittsburgh.

Daskalantonakis, Michael 1994. Achieving Higher SEI Levels. *IEEE Software*, 11, July, 17-24.

de Jong, Lieuwe 1996. Software Process Improvement in Philips Electronics. In: Proceedings of the First Annual European Software Engineering Process Group Conference 1996, Amsterdam, C307 P1-P9 S1-S12.

Decker, Diane; Belohlav, James 1997. Managing Transitions. The forgotten ingredient in change management. *Quality Progress*. No. 4, 1997, 93-97.

Deming, William Edwards 1992. *Out of the Crisis: Quality, Productivity and Competitive Position.* Cambridge University Press, Cambridge.

Department of Trade and Industry, British Standards Institute (eds.) 1992. *TickIT - making a better job of software. Guide to Software Quality Management System Construction and Certification using EN 29001*. Issue 2.0. London - DISC TickIT Office.

Dette, Wolfgang 1996. Einfuehrung eines QM-Systems nach DIN ISO 9001 in der Entwicklung. In: SQS, editor, Software-Qualitaetsmanagement 'Made in Germany' - Realitaet oder Wunschdenken?, SQM Kongress 1996, Koeln.

Diaz, Michael; Sligo, Joseph 1997. How Software Process Improvement helped Motorola. *IEEE Software*, 14, No. 5, 75-81.

Dillinger, Anton 1994. Erfahrungen eines Softwareherstellers mit der Zertifizierung eines Teilbereiches nach DIN ISO 9001. In: BIFOA, editor, Fachseminar: Aufbau eines Qualitaetsmanagements nach DIN ISO 9000, Koeln, 1-24.

Dion, Raymond 1992. Elements of a Process-Improvement Program. *IEEE Software*, July, 83-85.

Dion, Raymond 1993. Process Improvement and the Corporate Balance Sheet. *IEEE Software*. July, 28-35.

Forrester, Mike 1996. A TickIT for Logica. *SQM*, No. 16.

Forrester, Mike; Dransfield Arthur 1994. Logica's TickIT to ride extended for 3 years! *TickIT International*, No. 4.

Fowler Jr., Kimsey M. 1997. SEI CMM Level 5: A Practitioner's Perspective. *Crosstalk - The Journal of Defense Software Engineering*, 10, No. 9.

Frangos, S. A. 1995. Implementing a quality management system using an incremental approach. In: M. Ross et al., editors, Software Quality Management III, vol. 1: Quality Management, Southampton – Boston, 27-41.

Fulton A. M.; Myers, B. M. 1996. TickIT awards - a winner's perspective. *Software Quality Journal*, 5, No. 2.

George, Mike 1996. Using the CMM to Design a RAD Software Engineering Process. In: Proceedings of the First Annual European Software Engineering Process Group Conference 1996, Amsterdam, C308 P1-P8 S1-S9.

Goldenson, Dennis; Herbsleb, James D. 1995. *After the Appraisal: A Systematic Survey of Process Improvement, its Benefits, and Factors that Influence Success.* (Technical report CMU/SEI-95-TR-009). Software Engineering Institute, Pittsburgh, PA.

Haley, Thomas J. 1996. Software Process Improvement At Raytheon. *IEEE Software*, 13, No. 11, 33-41.

Halsey, Fillip A. L. 1997. Approaches to Process Improvement Support. In: Proceedings of the Second Annual European Software Engineering Process Group Conference 1997, Amsterdam, C308a 1-5.

Hardy, Robert E.; Schwartz, Randy 1996. *The Self-Defeating Organization. How smart companies can stop outsmarting themselves*. Reading, Mass.

Havenhand, Ron 1996. TickIT Case Study: British Rail Business Systems. *SQM*, No. 18, 1-6.

Hawley, John K. 1995. Where's the Q in TQM? *Quality Progress*, October, 63-64.

Hellmann, Debbie; Pilgrim, Alf 1997. From Chaos to Control. In: Proceedings of the Second Annual European Software Engineering Process Group Conference 1997, Amsterdam, C407a 1-13.

Hepworth, Brian 1994. Making the best the standard. Users experiences of operating an ISO 9001 compliant quality management system and total quality management culture. In: Proceedings of the Fourth European Conference on Software Quality, Zuerich, 208-223.

Herbsleb, J.; Carleton, A.; Rozum, J.; Siegel, J.; Zubrow, D. 1994. *Benefits of CMM-based software process improvement: Initial results.* (Technical report CMU/SEI-94-TR-13. ESC-TR-94-013), Software Engineering Institute, Pittsburgh, PA.

Hewson, Mike 1996. TickIT Case Study: Praxis. *SQM*, No. 22.

Hollenbach, Craig; Young, Ralph; Pflugrad, Al; Smith, Doug 1997. Combining Quality and Software Improvement. *Communications of the ACM*, 40, No. 6, June, 41-45.

Hookham, Paul 1997. Experiencing SPI at the Sharp End. In: Proceedings of the Second Annual European Software Engineering Process Group Conference 1997, Amsterdam, C307a 1-10.

Humphrey, W.S.; Snyder, T.R.; Willis, R.R. 1991. Software Process Improvement at Hughes Aircraft. *IEEE Software,* 8, July, 11-23.

Humphrey, Watts S. 1989. *Managing the software process.* Addison-Wesley, Reading, MA.

Humphrey, Watts S. 1995. *A Discipline for Software Engineering*. Addison-Wesley, Reading, MA.

Humphrey, Watts S. 1997. *Managing Technical People - Innovation, Teamwork, and the Software Process*. Addison-Wesley, Reading, MA.

International Organization for Standardization, ed. 1994. *Quality management and quality assurance - Vocabulary. ISO 8402*, International Organization for Standardization, Geneva.

International Organization for Standardization, ed. 1994. *Quality management and quality assurance standards. Part 1. Guidelines for selection and use. ISO 9000-1*, International Organization for Standardization, Geneva.

International Organization for Standardization, ed. 1994. *Quality systems. Model for quality assurance in design, development, production, installation and servicing. ISO 9001*, International Organization for Standardization, Geneva.

International Organization for Standardization, ed. 1994. *Quality management and quality system elements. Part 1: Guidelines. ISO 9004-1*, International Organization for Standardization, Geneva.

Jackson, Keith 1996. Relationship with Software Process Improvement (SPI) Supplier. In: Proceedings of the First Annual European Software Engineering Process Group Conference 1996, Amsterdam, C304 P1-P6 S1-S5.

Jackson, Keith 1997. Practical Implementation of Process Improvement. In: Proceedings of the Second Annual European Software Engineering Process Group Conference 1997, Amsterdam, C308a 1-10.

Johnson, Albert 1994. Software Process Improvement Experience in the DP/MIS Function. Experience Report. In: Proceedings of the 16th International Conference on Software Engineering. IEEE Computer Society Press, Washington - Brussels - Tokyo, 323-329.

Kanter, Rosabeth Moss 1994. Mastering Change. In: Sarita Chawla, John Renesch (eds.): Learning Organizations - Developing Cultures for Tomorrow's Workplace, Portland, Oregon, 71-83.

Keustermans, Jef 1996. Reaching CMM level 3 - the Ericsson Telecommunicatie B.V. case. In: Proceedings of the First Annual European Software Engineering Process Group Conference 1996, Amsterdam, C404 P1-P10 S1-S10.

Kilberth, K. 1997. Einfuehrung eines prozess-orientierten QM-Systems bei der ALLDATA. In: H. J. Scheibl, editor, Technische Akademie Esslingen - Software-Entwicklung - Methoden, Werkzeuge, Erfahrungen '97, Ostfildern, 377-392.

Klaus, Hans-Georg 1994. Zertifizierung eines Softwareherstellers nach DIN ISO 9001 -Voraussetzungen, Ablauf, Vorgehensweise-. In: BIFOA, editor, Fachseminar: Aufbau eines Qualitaetsmanagements nach DIN ISO 9000, Koeln.

Kness, Steven P.; Satake, Mark S. 1997. A Level 5 Organization Looks at the Personal Software Process. *Crosstalk - The Journal of Defense Software Engineering*, 10, No. 10.

Kotter, John P. 1995. Leading Change: Why Transformation Efforts Fail. *Harvard Business Review*, No. 2, 1995, 59-67.

Krasner, Herb; Scott, Gregory 1996. Lessons Learned from an Initiative for Improving Software Process, Quality and Reliability in a Semiconductor Equipment Company. In: Proceedings of the Twenty-Ninth Annual Hawaii International Conference on System Sciences. Volume 1. Software Technology and Architecture. Washington - Brussels - Tokyo, 693-702.

Krasner, Herb; Ziehe, Ted 1995. Lessons Learned from the Semiconductor Industry Initiative for Improving Software Process, Quality, and Reliability. In: Proceedings of the First World Congress for Software Quality. American Society for Quality Control (ASQC), San Francisco, CA, session D, pp. 1-20

Kuvaja, P.; Bicego, A. 1994. BOOTSTRAP - a European assessment methodology. *Software Quality Journal*, 3, No. 3, 117-128

LaMarsh, Jeanenne 1995. *Changing the Way We Change. Gaining Control of Major Operational Change*. Reading, Mass.

Laporte, Claude Y.; Papiccio, Nicola R. 1998. Software and Systems Engineering Process Development and Integration at Oerlikon Aerospace. *Software Process Newsletter*, No. 11, Winter 1998, 10-17.

Larner, Chris 1995. Practical Experiences and Lessons Gained by a Major European Bank from a Software Development Process Improvement Programme based on the Capability Maturity Model. In: Proceedings of the First World Congress for Software Quality. American Society for Quality Control (ASQC), San Francisco, 1-11.

Larner, Chris 1996. Practical CMM Implementation Experiences - Or - A Question of Ownership. In: Proceedings of the First Annual European Software Engineering Process Group Conference 1996, Amsterdam, C304 P1-P12 S1-S16.

Leonard-Barton, Dorothy 1995. *Wellsprings of Knowledge. Building and Sustaining the Sources of Innovation*. Boston, Mass.

Lewin, Kurt 1958. Group decision and social change. In: Readings in social psychology. 3rd ed. Holt, Rinehart, and Winston, New York, 197-211.

Litwin, George H.; Bray, John; Lusk Brooke, Kathleen 1996. *Mobilizing the Organization. Bringing Strategy to Life.* London et al.

Loken, Cecile B.; Skramstad, T. 1995. ISO 9000 certification - Experiences from Europe. In: Proceedings of the First World Congress for Software Quality. American Society for Quality Control (ASQC), San Francisco, CA, session Y, 1-11.

Lowe, Douglas E.; Cox, Guy M. 1996. Implementing the Capability Maturity Model for Software Development. *Hewlett-Packard Journal*, 47, No. 4, 1-11.

Macfarlane, Malcolm L. 1996. Eating the elephant one bite at a time. *Quality Progress*, No. 6, 89-92.

Masters, Robert J. 1996. Overcoming the Barriers to TQM's Success. *Quality Progress*, 29, No. 5, 53-55.

Mattingly, Joe; Abreo, Capt. Rene 1997. Midmanagers' Role in Software Process Improvement. *Crosstalk - The Journal of Defense Software Engineering*, 10, No. 1.

McFeeley, Bob 1996. *Ideal: A User's Guide for Software Process Improvement.* Handbook CMU/SEI-96-HB-001. Pittsburgh, PA.

McGarry, Frank; Pajerski, Rose; Page, Gerald; Waligora, Sharon; Basili, Victor; Zelkowitz, Marvin 1994. *Software Process Improvement in the NASA Software Engineering Laboratory.* Technical Report CMU/SEI-94-TR-22. ESC-TR-94-022. Pittsburgh, PA.

McGuire, Eugene G. 1996a. Factors Affecting the Quality of Software Project Management: An Empirical Study based on the Capability Maturity Model. *Software Quality Journal*, 5, No. 4, 305-317.

McGuire, Eugene G. 1996b. Initial Effects of Software Process Improvement on an Experienced Software Development Team. In: Proceedings of the Twenty-Ninth Annual Hawaii International Conference on System Sciences. Volume 1. Software Technology and Architecture, Washington - Brussels - Tokyo, 713-721.

Mehner, Thomas; Voelker, Axel 1996. Prozessassessment und -improvement - Erfahrungen aus der Praxis -. In: Heinrich C. Mayr (ed.): Informatik '96: Technische Beitraege und Praxisprogramm, Klagenfurt, 117-123.

Menezes, Winifred; Eschermann, Bernhard 1997. Setting up SPI in a Multi-Cultural and De-Centralised Engineering Company. In: Proceedings of the Second Annual European Software Engineering Process Group Conference 1997, Amsterdam, C306a 1-9.

Miles, Raymond E.; Coleman, Henry J.; Creed, W. E. Douglas 1997. Keys to Success in Corporate Redesign. *IEEE Engineering Management Review*, 25, No. 2, 5-15.

Morrell, Mike; van Asseldonk, Wilko 1997. Dealing with the Underworld - Accelerating SPI. In: Proceedings of the Second Annual European Software Engineering Process Group Conference 1997, Amsterdam, T102b 1-53; P1-P11.

Mosel, Henning 1996. Erfahrungen mit einem zertifizierten QMS im Bull-Softwarehaus. In: BIFOA, editor, Fachseminar: Von der ISO 9000 zum Total Quality Management? Koeln, 1-25.

Mosel, Henning 1996. Vier Jahre Zertifikat und was sonst noch notwendig ist. In: Software-Qualitaetsmanagement 'Made in Germany' - Realitaet oder Wunschdenken?. SQM Kongress 1996, Koeln

Munro-Faure, Malcolm; Munro-Faure, Lesley 1996. *The Success Culture - How to build an organization with vision and purpose*. London.

Myers, Charles R. 1996. Managing Resistance to Software Process Improvement. In: Proceedings of the First Annual European Software Engineering Process Group Conference 1996, Amsterdam, T104.

Myers, Chuck 1997. Dependence to Influence: Developing and Nurturing Effective Sponsorship. In: Proceedings of the Second Annual European Software Engineering Process Group Conference 1997, Amsterdam, T201b 1-29.

Nadler, David A.; Tushman, Michael L. 1997. Competing by Design. Oxford University Press, New York – Oxford.

Oestreich, Paul C.; Webb, David R. 1995. The Race to Level 3. *Crosstalk - The Journal of Defense Software Engineering*, 8, No. 6.

Olson, T.; Humphrey, W.; Kitson, D. 1989. *Conducting SEI-Assisted Software Process Assessments.* Technical Report, CMU/SEI-89-TR-7, Pittsburgh.

Paulish, Daniel J.; Carleton, Anita D. 1994. Case Studies of Software-Process-Improvement Measurement. *IEEE Computer*, 27, No. 9, 50-59.

Paulk, Mark C. 1998. Using the Capability Maturity Model for Software to Drive Change. In: Tor Larsen, Eugene McGuire (eds.): Information Systems Innovation and Diffusion: Issues and Directions, Idea Group Publishing, Hershey, USA - London, UK, 196 – 219.

Paulk, Mark C.; Curtis, B.; Chrissis, M.; Weber, C. 1993. *Capability Maturity Model for Software, Version 1.1.* Technical Report, CMU/SEI-93-TR-024, Pittsburgh.

Paulk, Mark C.; Weber, C.; Garcia, S.; Chrissis, M.; Bush, M. 1993. *Key Practices of the Capability Maturity Model, Version 1.1.* Technical Report, CMU/SEI-93-TR-025, Pittsburgh.

Paulk, Mark C.; Weber, C.V.; Curtis, B.; Chrissis, M.B. 1995. *The Capability Maturity Model: Guidelines for Improving the Software Process*. Addison-Wesley, Reading, MA.

Powel, Karen L. 1997. Successful Software Engineering Process Groups. *Crosstalk - The Journal of Defense Software Engineering*, 10, No. 2.

Quinn, Belinda 1996. Lessons Learned from the Implementation of a Quality Management System to meet the Requirements of ISO9000/TickIT in two small Software Houses. In: Fifth European Conference on Software Quality - Conference Proceedings, Dublin, Ireland, 305-314.

Quinn, David P. 1998. A Good Idea Still Requires a Good Implementation. *Crosstalk - The Journal of Defense Software Engineering*, 11, No. 1, 1-5.

Robb, C. 1994. From quality system to organisational development. In: M. Ross et al., editors, Software Quality Management II, vol. 1: Managing Quality Systems, Southampton - Boston, 99-113.

Schaffer, Robert H.; Thomson, Harvey A. 1992. Successful Change Programs begin with Results. *Harvard Business Review*, 70, No. 1, 80-89.

Schroeder, Markus; Wilhelm, Rudolf 1996. Flexibilitaet staerken. Erfahrungen beim Aufbau eines QM-Systems nach ISO 9000 in einem kleinen Softwareunternehmen. *QZ - Qualitaet und Zuverlaessigkeit*, 41, No. 5, 530-536.

Seshagiri, Girish 1996. Continuous Process Improvement - Why Wait Till Level 5? In: Proceedings of the Twenty-Ninth Annual Hawaii International Conference on System Sciences. Volume 1. Software Technology and Architecture. Washington - Brussels - Tokyo, 681- 692.

Siddall, Don 1997. Understanding and Managing Resistance. *Software Process Newsletter*, No. 9, Spring 1997, 10-12.

Sidi, Jacqueline; White, David 1995. Implementing Quality in an International Software House. In: Proceedings of the First World Congress for Software Quality. American Society for Quality Control (ASQC), San Francisco, CA, Session W, 1-13.

Snyder, Terry R. 1997. A Quarter Century of Software Process Improvement. In: Proceedings of the Second Annual European Software Engineering Process Group Conference 1997, Amsterdam, C405a 1-14.

Steinke, Stefan 1997. Erfahrungen bei der Einfuehrung und Verbesserung eines QMS. In: SQS, editor, Software-Qualitaetsmanagement 'Made in Germany' - Modeerscheinung oder Daueraufgabe. SQM Kongress 1997, Koeln.

Stelzer, Dirk; Mellis, Werner; Herzwurm, Georg 1996. Software Process Improvement via ISO 9000? Results of Two Surveys Among European Software Houses. *Software Process - Improvement and Practice*, 2, No. 3, 197-210.

Stelzer, Dirk; Mellis, Werner; Herzwurm, Georg 1997. A Critical Look at ISO 9000 for Software Quality Management. *Software Quality Journal*, 6, No. 2, 65-79.

Stelzer, Dirk; Taube, Frank 1998. *Stand der kontinuierlichen Verbesserung in der Softwareentwicklung. Ergebnisse einer empirischen Untersuchung bei Softwareunternehmen in Deutschland, deren Qualitaetsmanagementsystem nach ISO 9001 zertifiziert wurde. Studien zur Systementwicklung des Lehrstuhls fuer Wirtschaftsinformatik der Universitaet zu Koeln. Band 13.* Koeln.

Sweeney, A.; Bustard, D. W. 1997. Software process improvement: making it happen in practice. *Software Quality Journal*, 6, No. 4, 265-273.

Tushman, Michael; O'Reilly Charles A. 1997. *Winning through Innovation: a practical guide to leading organizational change and renewal.* Harvard Business School Press, Boston, Mass.

Verbe, S.; Robinson, P.W. 1995. Growing a quality culture: a case study - Oracle UK. In: M. Ross et al., editors, Software Quality Management III, vol. 1: Quality Management, Southampton - Boston, 3-14.

Vering, Matthias; Haentjes, Volker 1995. Ist ISO 9000 ein geeignetes Werkzeug fuer Process Engineering? Ein Erfahrungsbericht aus der SAP-Entwicklung. *m & c - Management & Computer*, 3, No. 2, 85-90.

Voelker, Axel 1996. Why Maturity Matters. In: Proceedings of the First Annual European Software Engineering Process Group Conference 1996, Amsterdam, C305 P1-P15 S1-S16.

Vu, John D. 1997. Software Process Improvement Journey from Level 1 to Level 5. In: Proceedings of the Second Annual European Software Engineering Process Group Conference 1997, Amsterdam, C404a 1-16.

Walker, S.D. 1994. Maintaining your quality management system - what are the benefits?. In: M. Ross et al., editors, Software Quality Management II, vol. 1: Managing Quality Systems, Southampton - Boston, 47-61.

Warner, A. 1997. Der Weg von der Qualitaetssicherung nach ISO 9001 zum Qualitaetsmanagement in einem Systemhaus. In: H. J. Scheibl, editor, Technische Akademie Esslingen - Software-Entwicklung - Methoden, Werkzeuge, Erfahrungen, Ostfildern, 407-423.

Westaway, Tom 1995. How we achieved level 3. *Crosstalk - The Journal of Defense Software Engineering*, 8, No. 5.

Wiegers, Karl E. 1996a. *Creating a Software Engineering Culture.* Dorset House, New York.

Wiegers, Karl E. 1996b. Software Process Improvement: Ten Traps to Avoid. *Software Development*, 4, No. 5.

Wigle, Gary B.; Yamamura, George 1997. Practices of an SEI CMM Level 5 SEPG. *Crosstalk - The Journal of Defense Software Engineering*, 10, No. 11, November.

Wohlwend, Harry; Rosenbaum, Susan 1994. Schlumberger's Software Improvement Program. *IEEE Transactions on Software Engineering*, 20, No. 11, 833-839.

Yamamura, George; Wigle, Gary B. 1997. SEI CMM Level 5: For the Right Reasons. *Crosstalk - The Journal of Defense Software Engineering*, 10, No. 8, August.

Zahniser, Rick; Rizzo, Dan 1995. Software Process Reengineering. Getting to 'Level 3' with Teamwork. *American Programmer*, 8, No. 6, 36-43.

Zopf, Siegfried 1994. Ein Erfahrungsbericht zur ISO 9001 Zertifizierung. *Softwaretechnik-Trends*, August, 15-16.

Zopf, Siegfried 1994. Improvement of software development through ISO 9001 certification and SEI assessment. In: Proceedings of the Fourth European Conference on Software Quality, Zuerich, 224-231.

Insert Appendix 1 here